Online Advertising Security: Issues, Taxonomy, and Future Directions

Zahra Pooranian, *Senior Member, IEEE,* Mauro Conti, *Senior Member, IEEE*, Hamed Haddadi, *Member, IEEE* and Rahim Tafazolli, *Senior Member, IEEE*

***Abstract*—Online advertising has become the backbone of the Internet economy by revolutionizing business marketing. It provides a simple and efficient way for advertisers to display their advertisements to specific individual users, and over the last couple of years has contributed to an explosion in the income stream for several web-based businesses. For example, Google's income from advertising grew 51.6% between 2016 and 2018, to $136.8 billion. This exponential growth in advertising revenue has motivated fraudsters to exploit the weaknesses of the online advertising model to make money, and researchers to discover new security vulnerabilities in the model, to propose countermeasures and to forecast future trends in research.**

Motivated by these considerations, this paper presents a comprehensive review of the security threats to online advertising systems. We begin by introducing the motivation for online advertising system, explain how it differs from traditional advertising networks, introduce terminology, and define the current online advertising architecture. We then devise a comprehensive taxonomy of attacks on online advertising to raise awareness among researchers about the vulnerabilities of online advertising ecosystem. We discuss the limitations and effectiveness of the countermeasures that have been developed to secure entities in the advertising ecosystem against these attacks. To complete our work, we identify some open issues and outline some possible directions for future research towards improving security methods for online advertising systems.

***Index Terms*—Online Advertising Systems, Security, Ad Fraud, Click Fraud, Taxonomy.**

I. Introduction

OVER the past few years, the widespread adoption of the Internet has led to the emergence of a new form of online business – i.e., *online advertising* – to make money through this means. A significant financial pillar of the Internet ecosystem is provided by online advertising (from websites and mobile apps) [1]–[4].

Many companies such as Google and Microsoft have increased their investment in online advertising to improve their revenue and sales. According to the report in [5], Google's income from advertising grew 51.6% between 2016 and 2018, to $136.8 billion. It was expected that this revenue reached nearly $203.4 billion by 2020 and will continue to increase over time. Also, mobile advertising has become one of the fastest-growing industries with the advent of smartphones [6]. Millions of mobile applications are registered in various application platforms such as Google Play Store, Apps Store, etc., which contain at least one advertising library that allows mobile advertising [7]. According to [8], in 2019, total mobile advertising spending worldwide has reached $189 billion and will surpass $240 billion by 2022.

Online advertising uses the same mechanisms that are applied to manage other "traditional" advertising channels, such as newspapers, radio or TV, but is much more creative in providing targeted and personalized advertisements [9], [10]. Thanks to the rise of the Internet and online advertising, sales of TV and radio advertisements have stagnated, and those of newspaper advertisements have dropped. Fig. 1 shows a comparison of global ad spending by medium [11].

Online advertising provides profit for all the components of the system, such as publishers, advertisers, and advertising network (ad network). Given the high profits involved, the online advertising system is an obvious target for fraud. Hence, several attacks on the current online advertising market have been identified that have targeted various entities in the market, such as hacking campaign account [12], click fraud [13], inflight modification of advertising (ad) traffic [15], and malvertising [14].

The inherent lack of transparency and complexity of the online advertising ecosystem give rise to higher risks, and an adversary can easily exploit these aspects to engage in fraudulent activities and launch an attack on the system. Ad fraud can occur in various forms and may involve fooling different components of the online advertising ecosystem to make money. For instance, dishonest publishers may deceive advertisers into paying an extra fee, or hackers could hijack an advertising slot to gain revenue for themselves.

In view of the factors described above, the success and popularity of the online advertising ecosystem depend primarily on the level of security that can provide against such malicious threats. The considerations mentioned above motivate the current work in terms of studying security issues in the online advertising market and essential related techniques.

Survey Organization: The remainder of this article is structured as follows. We begin by discussing work that gives a high-level overview of related online advertising survey, which help define the contributions of this paper (Section II). In Section III, we explain the differences between the online advertising system and traditional advertising networks, introduce the terminology used, and describe its architecture. Section IV presents our proposed taxonomy of vulnerability on the online advertising system. We also discuss the goal of these attacks, the revenue model, and the primary components

Zahra Pooranian and Rahim Tafazolli are with 5G & 6G Innovation Centre (5GIC & 6GIC), Institute for Communication Systems (ICS), University of Surrey, Guildford, UK (e-mail: {z.pooranian, r.tafazolli}@surrey.ac.uk)

Mauro Conti is with the Department of Mathematics, University of Padua, Padua, Italy (e-mail: conti@math.unipd.it)

Hamed Haddadi is with the Faculty of Engineering, Imperial College London, London, UK (e-mail: h.haddadi@imperial.ac.uk)

Manuscript received 19 May 2020; revised 6 Dec 2020, 17 May and 16 July 2021; accepted 2 October 2021

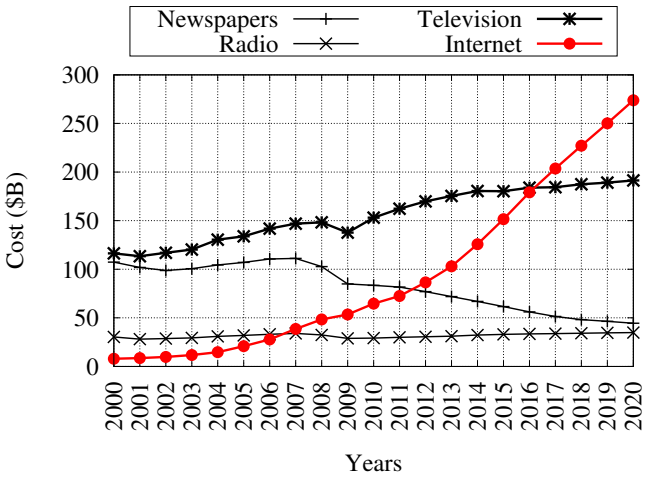

Figure 1: Global ad spending by medium.

targets. In Section V, we categorize and discuss various security solutions identified in the literature and present a preliminary overview of the advantages and disadvantages of the use of these solutions in online advertising systems. In Section VI, we highlight several open challenges for future research in online advertising systems. We conclude our work in Section VII.

II. Related High Level Articles and The Scope of This Survey

We begin this section by discussing the related online advertising survey in Section II-A, which help define the scope and contributions of this paper in Section II-B.

A. Surveys on Online Advertising Systems

Numerous existing works have discussed general aspects of online advertising systems. Most of the early works focused on issues relating to the economic aspects of advertising [16], [17], [18], [19], [20], the literature review on online advertising [21], challenges in online advertising [22], theoretical or analytical assessments of sponsored searches [23], [24], [25], and especially analyses of privacy threats and protection mechanisms [26], [27].

The authors of [28] investigated a wide range of mobile ad frauds and developed a comprehensive taxonomy for the research community. Budak et al. [29] showed that one of the leading sources of threats to the online advertising system is the widespread use of ad-blocking software and third-party platform tracking. The study in [7] presented a comprehensive survey of the existing literature on the privacy risks of targeted advertising, together with solutions to these challenges. The papers in [30] and [31] surveyed click fraud attacks and analyzed solutions for mitigating them, with a focus on the threats affecting the PCC revenue model. In [32], the authors provided a summary of the current state of click fraud, including typical forms of attack and their countermeasures. Although they discussed the impact of click fraud on different types of revenue models, less attention was paid to other kinds of ad fraud in online advertising systems. The survey in [33] gave a detailed summary of all the forms of web and network fraud that can be detected by data mining techniques, and reviewed some approaches for identifying fraud in real time. The study in [34] focused on the visualization of the online advertising ecosystem and future trends in programmatic advertising; although the authors aimed to address a gap in the literature by developing a standard for the visualization and explanation of the digital advertising ecosystem, the survey did not include any discussion of the security aspects of online advertising systems. The survey in [35] addressed the security problems faced by online advertising, including the corresponding countermeasures in the literature. However, our survey paper covers online advertising systems from various angles as follows. First, we examine threats in the system from three dimensions, depending on whether the page content is targeted by fraud, ad traffic, or user actions. Therefore, we classify advertising fraud into three main categories: placement fraud, traffic fraud, and action fraud. In particular, we explain how an adversary can exploit the risks in online advertising systems and conduct ad fraud for each type of attack. Second, we provide a detailed discussion of the goals of the attacks, the revenue model that is the attacker's target and the primary component targets (see Table II). Third, we have provided the pros and cons of the countermeasures techniques and how they could combat attacks. Finally, in the future section, we have discussed four major aspects of research to support online advertising systems' security, reliability, and efficiency.

B. Our Scope

This article presents a survey that primarily targets the security issues and challenges of online advertising systems and reviews the related fundamental concepts. From a security perspective, it presents a comprehensive taxonomy of well-known ad fraud. It also categorizes several security mechanisms that have been proposed in recent years to cope with and mitigate the existing security challenges in the online advertising industry. In particular, our classification focuses on the goals of attacks, the revenue model, and the primary component targets.

The research papers and books we mentioned previously did not address security issues with an emphasis on online ad fraud in this area. There is therefore a need for a concise survey to provide a reader who is planning to undertake research in this field with a classification of online ad fraud, along with an exhaustive review of the corresponding countermeasures. In brief, the essential contributions of the survey are as follows:

- First, some essential background knowledge is presented, including the differences between traditional and current online advertising systems, the terminology used, and the existing architecture of online advertising (e.g., web and mobile). The goal is to enable new readers to gain the required familiarity with online advertising systems and its underlying technologies, such as revenue models and the payment of commissions.
- We present a detailed taxonomy of the current security threats to online advertising. We investigate several possibilities, including both theoretical and practical vulnerabilities, that fraudsters can use to launch an attack on

the online advertising industry. In addition, we present a detailed discussion of the goals of these attacks, their impact on the particular revenue models, and the primary component targets.
- We review several cutting-edge solutions that address security threats to online advertising systems, and explain the advantages and disadvantages of each solution.
- Finally, we identify a number of open challenges and future research directions in the field of online advertising, with particular attention to the security aspects.

To the best of our knowledge, there are no existing surveys that have reviewed and summarized the existing security vulnerabilities and outlined future research directions in the realm of online advertising systems. Motivated by this consideration, the main goals of this study are threefold: (i) to help the reader to understand the scope and consequences of the security threats and challenges in the domain of online advertising systems; (ii) to estimate the potential damage associated with these threats; and (iii) to highlight paths that are likely to lead to the detection and containment of these threats. From a practical perspective, our research aims to raise awareness in the online advertising research community of the urgent need to prevent various attacks from disrupting the healthy online advertising market.

III. Overview of Online Advertising Systems

We begin this section with an explanation of the most widely used terminology associated with the online advertising ecosystem, including terms used in the remainder of the present article in Section III-A. Section III-B provides a brief introduction to the online advertising ecosystem, especially in terms of its main components, the interactions between them, and the technologies they support. To gain insight into how online advertising networks operate, we discuss the current ads delivery workflow on the web in Section III-C. We highlight the similarities and differences between mobile advertising and web-based advertising ecosystems in Section III-D. Then, we discuss the ad delivery workflow on mobile platforms in Section III-E. Next, different methods of targeted advertising and the most common types of ad campaign (revenue model) are described in Sections III-F and III-G, respectively. Finally, we explore how advertisers pay commission fees to commissioners and publishers in Section III-H, and describe the main security goals in the online advertising system in Section III-I.

For ease of reading, in Table I, we list the all abbreviations used in this paper.

Table I: List of abbreviations and corresponding descriptions.

Abbreviation	Description
Ad Network	Advertising Network
Ad	Advertising
DSPs	Demand-side Platforms
SSPs	Supply-side Platforms
HTTP	Hypertext Transfer Protocol
RTB	Real-time Bidding
CTR	Click-Through Rate
ROI	Return on investment
CM	Cookie Matching
SDK	Software Development Kit
UI	User Interfaces
GDPR	General Data Protection Regulation
CPM	Cost per Impression Mile
CPC	Cost per Click
CPA	Cost per Action
CIA	Confidentiality, Integrity and Availability
ISP	Internet Service Provider
MITM	Man-In-The-Middle
DNS	Domain Name System
CFC	Click Fraud Crowdsourcing
MTA	Mail Transfer Agent
TTP	Trusted Third Party
CGI	Common Gateway Interface
GBF	Group Blooms Filter
TBF	Timing Blooms Filter
NMF	Non-negative Matrix Factorization
SeqGAN	Sequence Generative Adversarial Generative
MLE	Maximum Likelihood Estimation
CSBPNN	Cost-sensitive Back Propagation Neural Network
ABC	Artificial Bee Colony
SLEUTH	Single-publisher attack dEtection Using correlaTion Hunting
ML	Machine Learning
CFXGB	Cascaded Forest and XGBoost
CAPTCHA	Completely Automated Public Turing test to tell Computers and Humans Apart
HMSM	Hidden Markov Scoring Model
HMM	Hidden Markov Model
SVM	Support Vector Machine
TLS	Transport Layer Security
IoT	Internet of Things
AI	Artificial Intelligence
AR	Augmented Reality
5G	5th Generation of Mobile Internet
DLT	Distributed Ledger Technology
API	Application Programming Interface

A. Terminology

In this subsection, we define some essential terminology related to online advertising ecosystems, as used throughout this paper.

- An *advertiser* is a party who is willing to show a product, service, or event to the user via advertisements, in order to promote sales or attendance. Advertisers typically pay (or buy traffic from) an ad network to display their advertisements in the advertising space on publishers' websites or phone applications. The publisher also receives a percentage of this fee.
- A *publisher* (such as The New York Times or CNN) is an entity that receives money (via selling traffic) from advertisers by displaying their advertisements to users through its web pages (or mobile app).
- A *user* is an individual who visits a publisher's web pages.
- An *advertising network* (such as Google, Yahoo, Google AdSense, Media.net, or PulsePoint) also known as a *commissioner*, is part of an ad exchange. It acts as a broker between the advertiser and the publisher to manage the interaction between them [36], and is responsible for finding suitable spaces to present advertisements on publishers' websites for advertisers. They may also buy or sell ad traffic (as ad requests), either internally or together with other ad networks.
- An *ad exchange* (such as DoubleClick [37], AdECN [38], or OpenX [39]) is a graph of the advertising networks that

allows the advertiser and publisher to serve advertisements more effectively within an advertising space.

- *Demand-side platforms (DSPs)* are components that work for advertisers, that is, for the actors who generate the demand for advertising services [31]. DSPs work on behalf of advertisers in front of ad exchanges, helping advertisers choose the right audience and media to display their ads. By gathering demand, DSPs can increase selection and effectiveness for advertisers.
- *Supply-side platforms (SSPs)* act on behalf of publishers to provide advertising space to advertisers. SSP offers publishers an optimized strategy for managing their ad inventory.
- *Ad servers* are a type of web server (or platform) that is used to host the content of an online advertisement and distribute this content on digital platforms such as Facebook, Quora, Twitter, etc.
- An *advertising request* (ad request) is a query, in the form of Hypertext Transfer Protocol (HTTP) traffic, that is triggered by a web user's impressions or clicks, and calls an ad server to display an ad to the user.
- *Creative* content is associated with the actual advertising message (e.g., an anchor tag, an Adobe Flash animation, text, or images) in the ad slot displayed to the user. The process of linking an ad message to an advertiser's website is called *click-through* [40].
- An ad server enumerates a *click* event when a user clicks on an ad.
- An ad server counts an *impression* event whenever the content or ad page is loaded for the user. Clicks and impressions generate two different events, which are handled separately in the online advertising system.
- An *auction* is a competitive process that runs within the ad exchange. It is designed to allow each advertiser to bid for advertisement space, where the highest bidder is permitted to place an advertisement in the slot. An auction aims to generate more profit for publishers. In general, the time taken to complete the entire process is on the order of 100 ms.
- After an auction, ad networks may perform *arbitrage* to increase their revenue. To initiate arbitrage, the ad network must run a new and independent auction by buying and reselling traffic from the publisher.
- An *ad campaign* is a method that emerged to help advertisers to decide how much to pay when their advertisements are displayed. We discuss the most common forms of ad campaigns in Section III-G.
- A *banner* is a space on a page that displays a message from the advertiser.
- *Real-time bidding* (RTB) is one of the critical technologies to make a profit for online advertising and allows advertisers to compete in real-time auctions to display their ads [41]. Therefore, when a user visits a website, his impression is sold to the advertiser (or DSP), which offers the highest price in a few milliseconds. Moreover, the bid request messages received by DSPs contain user information (tracking data) to help them adjust ads based on user preferences and decide on a bidding strategy. In this way, the goal of RTB is twofold: to provide a personalized experience to users through targeted advertising and to maximize the profits of the entire advertising ecosystem.

B. An Overview of Web Advertising Ecosystem

Not surprisingly, advertising techniques have evolved over time with the growth of the Internet, and online advertising has become one of the biggest and most profitable Internet businesses. The main idea behind online advertising is to provide an advertiser with a cost-effective, easy, fast, and flexible way to promote and sell their products through the Internet to suitable customers. In this way, it can maximize revenue, click-through rate (CTR[1]) or return on investment (ROI[2]) of the advertising campaign [43]. There are several significant differences between current online advertising and traditional advertising (e.g., via television, radio, and newspapers). For example, traditional advertising uses massive broadcast advertisements without considering the user's interests; in contrast, online advertising can deliver advertisements to targeted users based on their interests and browsing behavior, regardless of geographical barriers.

Fig. 2 shows the high-level overview of an online advertising ecosystem. Existing architecture can be more complex and dynamic than this design. However, the scheme relies on integrating three main components: an advertiser, a publisher, an ad exchange (e.g., multiple ad networks). Thanks to technologies such as RTB, the ultimate goal of these components is to show the right ad to the right user at the right time. The former two components show the demand and supply aspects of an online advertising service's economic model. The interaction between such players is usually done by an intermediate infrastructure, called ad exchange. To participate in the ad bidding, publishers and advertisers connect to the ad exchange network via SSPs and DSPs to conduct auctions and manage bids, respectively, then finally deliver the ads to various media platforms (such as a third party website, a search engine results page) [44]. Users whose data and requests are the basis of decisions made for online advertising services are a passive part of this infrastructure because they do not make money from this billion-dollar business.

C. General Operation of Web Advertising

By showing the main components of the online advertising ecosystem, we now give a brief description of how ads are delivered on the web.

The process of ad serving in an online advertising system is illustrated in Fig. 3. The process is initiated when a user requests (e.g., an HTTP request) calls for an advertisement to be served by the publisher (step 1-2). Following this, the publisher (with the help of SSPs) asks the ad exchange to fill the ad on the visited page that best matches the user's profile and has the best price (step 3). The ad exchange starts an auction between multiple advertisers (with the help of DSPs) by sending the "*bid request*" (with user data) to

[1]The CTR is the number of clicks an advertiser (i.e., publisher or ad) gains as a proportion of the impressions [42].

[2]ROI is an indicator used to measure the efficiency of an investment.

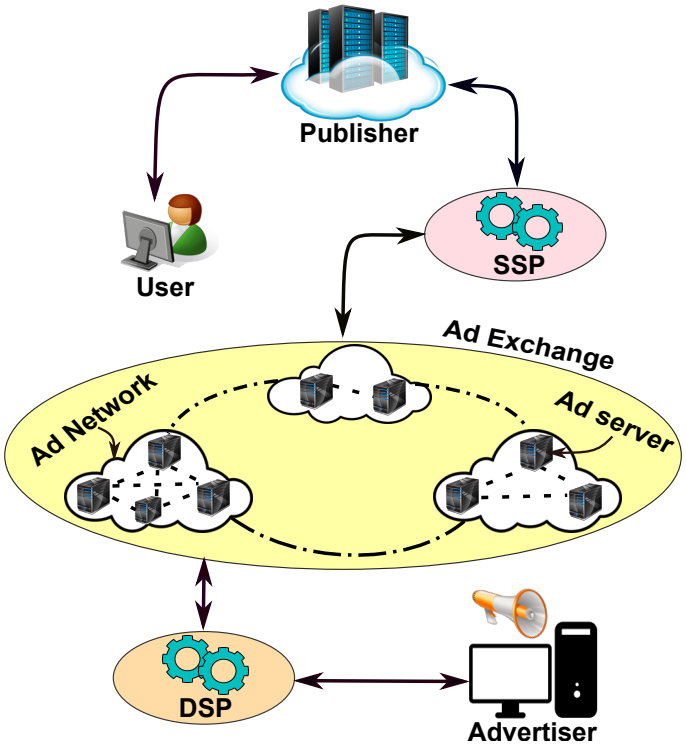

Figure 2: High-level overview of online advertising ecosystem.

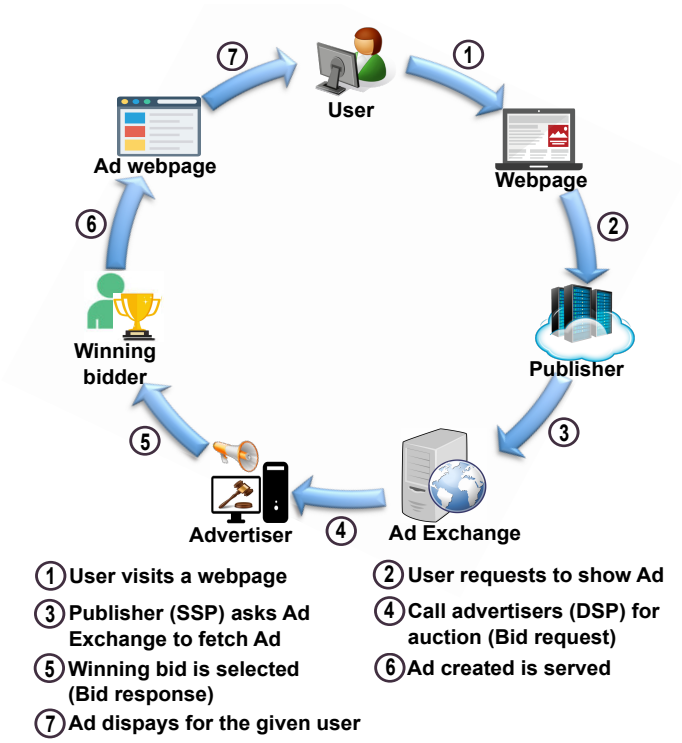

Figure 3: The process of serving ads in an online advertising system.

determine which can make the most profit for the publisher and, consequently, the whole network [45] (step 4). Both RTB and cookie matching (CM) mechanisms help the online advertising system ensure the most impact on users (which is to the benefit of advertisers), with the most benefit to publishers. RTB enables advertisers to bid for the chance to display an ad on a web page loaded by a user's browser.

Along with the bid request, ad exchanges send the following data about the user: URL of the page visited by the user; page subject category; IP address of the user or parts of it; and other information about their web browser. Ad exchanges widely use cookies and advertisers to collect and share such information, thus improving the advertising targeting process's accuracy [46]. Also, advertisers can use cookies to build a user's profile with information about their purchasing habits and browsing history for a future auction. Such information, through CM technology, helps advertisers and DSPs decide whether and how much to bid for a click or an impression. If an advertiser interested to show their ad, then send the price to the ad exchange. After such a process, the highest paying advertiser (winning bidder) wins the auction (step 5), and its ad is served and displayed to the user (steps 6-7).

D. An Overview of Mobile Advertising Ecosystem

Mobile advertising uses many traditional web advertising infrastructures, such as (mobile) user, (app) publisher, ad network, and advertiser, to deliver ads. An advertiser has the exact role of advertising in web advertising. An ad network is a trusted intermediary platform between the advertiser and app publisher to manage them. An app publisher (also called a developer) is an entity that publishes apps in the app market. The mobile user downloads applications from the application market and uses them on smartphones. Also, in this ecosystem, SSP and SSP are connected through an ad network.

In both web and mobile advertising systems, the ad library receives content embedded in a webpage or mobile app from ad providers and displays it on a webpage or mobile app interface. Then, the ad provider pays the publisher based on the number of clicks and impressions by the user. Despite the similarities, there are differences. For example, a website includes JavaScript code for displaying ads, while a mobile application has a custom software development kit (SDK[3]) embedded for loading JavaScript code and ads in a particular component like WebView[4] in Android [48].

E. General Operation of Mobile Advertising

In the mobile platform, ads are embedded as an advertising library to display ads in applications. The process of ad serving in mobile system is illustrated in Fig. 4. The advertiser distributes ads to display advertisements to mobile users through the ad exchange (step 1). When a publisher wants to display ads, he must register on the ad exchange to receive the ad library (SDK library) (step 2). The library typically provides an API for embedding ads in the User Interfaces (UI) of the app publisher and fetching, presenting, and tracking ads (step 3). The device ID is usually used to identify the publisher who wants to embed those ads uniquely. Users download

[3]The SDK usually consists of a pre-compiled ad library with required dependencies [47].

[4]WebView displays web content directly in Android applications without directing the user to the web browser.

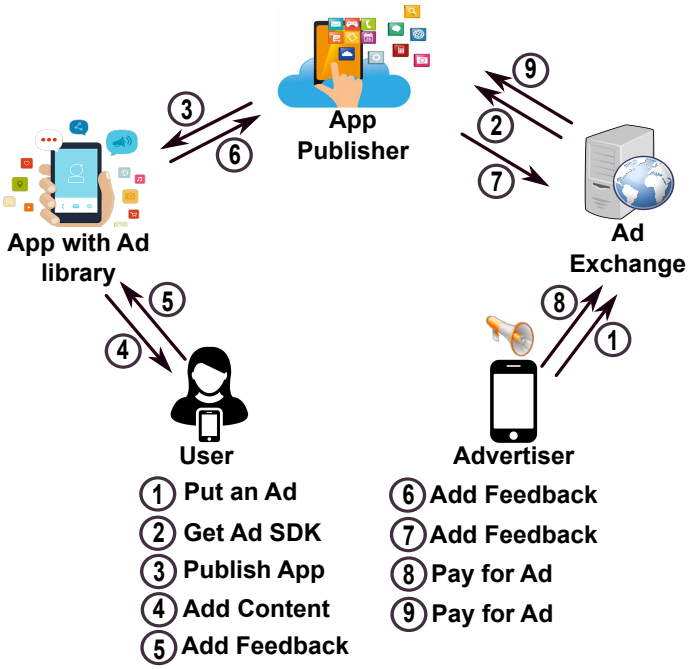

Figure 4: The process of serving ads in mobile advertising system.

the app and run it on a smartphone. Once the program starts, the ad library fetches ad content (step 4) and sends feedback (impression and clicks) to the ad exchange (step 5-7). Advertisers pay advertising networks and app publishers based on the number of impressions or the number of clicks (step 8-9).

F. Targeted Advertising

The most obvious difference between online advertising and a traditional approach is that the former displays advertisements to the customer based on their interests, while in the latter, advertisements are massively broadcast without considering the customer's interests. Ad networks use ad targeting methods to increase their income, and in this way can display advertisements based on the user's preferences. The three most popular types of ad targeting can be categorized into *contextual*, *behavioral*, and *location-based* approaches.

In the contextual approach, advertisers display relevant advertisements by focusing solely on the content of the web page being viewed by the user [49]. A behavioral targeting strategy allows advertisers and publishers to utilize information from the user's browsing history (e.g., by monitoring the behavior of the user on the Internet) to customize the types of advertisements they are served. Whenever an individual visits a website, all of the relevant information, including the pages visited, the period of time spent on each page, the links that are clicked on, and the things that are interacted with, are stored in a profile linked to that visitor [50]. Based on the data in these profiles, publishers can show related advertisements to visitors that match their habits. In a location-based targeting, location-specific advertisements are delivered to potential users; this technique is particularly useful for mobile advertising [51].

Although targeted advertising can be profitable for the advertiser, collecting consumer information raises their privacy concerns. Hence, governments enact more data privacy regulations. For example, the European Parliament has adopted the General Data Protection Regulation (GDPR [5]) to increase the transparency on how individuals data is used and stored. GDPR is an opportunity to build trust among marketing consumers. Becoming a widespread and trusted brand is critical to maintaining power in today's increasingly competitive world. Using data privacy as a core principle in business enables the business to establish an honest and humane relationship with its customers and partners. On the other hand, stricter regulation can limit advertisers' targeting advertising and slow down progress in the advertising ecosystem. Hence, there must be a trade-off between GDPR rules and targeted advertising to have a healthy dynamic in the advertising ecosystem.

G. Revenue Models

In this subsection, we discuss how entities in the online advertising network generate revenue.

Typically, publishers agree to display an advertiser's advertisements and share the keywords used by the advertiser in their website, charging a commission fee for the action(s) generated by the user. This agreement includes a contract made by a broker (also called an Internet advertising commissioner) between publishers and advertisers. The commissioner also controls the advertisers' budget, to avoid over-spending [52]. As soon as the advertiser pays the publisher the commission fee, it displays links determined by the advertiser on its website [53].

The general models [40], [51] used by publishers to make money through advertising are determined based on the numbers of impressions, clicks, and actions. We explain each of these types of revenue model in detail in the following subsections.

1) Cost per Impression: This model is favored by publishers and was developed based on traditional advertising systems. A metric called *Cost Per impression Mile* (CPM) is often used to measure the cost per impression, where the advertiser' payment to the ad network is calculated based on the cost of 1,000 views of an ad.

To enable a better understanding of how the commissioners process the receiving impression traffic, Fig. 5 illustrates this process. The steps are as follows: (1) a user requests a website; (2) in response, the publisher displays the requested website in the user's web browser; and (3) the user's browser redirects to the commissioner's web server (the commissioner does not repeat advertisements since it stores the recent advertisements shown to the user in browser cookies). In steps 4 and 5, the commissioner allows the user's browser to redirect to the advertiser server. In step 6, the commissioner loads the advertisement into the user's browser.

2) Cost per Click: In the Cost Per Click (CPC) model, the advertiser pays the publisher based on how many times a viewer clicks the ad on the publisher's web page. Many search

[5]The GDPR is a provision in the European Union Data Protection and Privacy Act.

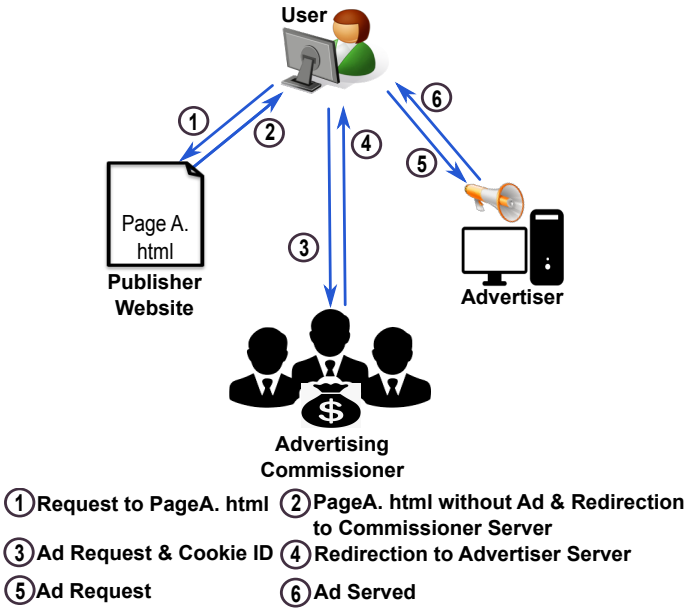

Figure 5: Impression traffic model used in online advertising systems.

engines, including Yahoo, Microsoft, and Google, prefer to use the CPC model. The reason for that is because a user clicking on an ad is a strong signal of interest; as such, CPC guarantees a better return on investment than CPM, where advertisers pay for their advertisements to be shown without counting on any implicit feedback from users.

The click traffic model is the approach that is most similar to the impression traffic model. However, the process starts when a user clicks on a hyperlink on the publisher's site. Then, the user is redirected to the commissioner's server. The server then logs the click for accounting purposes. After that, the server of the advertising commissioner redirects the user's web browser to the web page related to the advertiser.

3) Cost per Action: In general, the CPC charging model is considered to be a specific case of the Cost Per Action (CPA) model, in which the publisher is paid whenever a user-generated click leads to a predefined action being performed, e.g., filling in a form on the page, signing up, registering, or downloading an item corresponding to the ad. Advertisers prefer to deploy this type of cost model since they only pay the publisher for specific actions. Although this approach has advantages for the advertiser, it also has some drawbacks. It is challenging to implement, especially in the case of complex actions, and the publisher is less interested in applying this model since dishonest advertisers may deflate the number of actions to pay a lower commission fee (See Section IV-B2).

Some software also help advertisers and publishers to generate revenue by these models. One of them is adware (or advertising-supported software), which uses pop-up messages or unclosable windows to display ads. The first group of adware is known as *shareware*. This is designed for consumers who are unwilling to pay for specific software, and numerous ad-supported software, games, and utilities have been distributed as adware. This type of software automatically displays advertisements in the form of annoying pop-up messages, and users have an option to disable these advertisements if they buy a license key. The developer uses the adware to recover the costs of development, maintenance, and upgrading of the software. Also, this approach allows consumers to use the software free of charge or for a low price. The second category can be thought of as a kind of *spyware* [54]. This group stealthily collects information on customers by spying on them to serve advertisements embedded in websites. In formal terms, these applications generate revenue for developers by tracking the user's Internet surfing habits to display advertisements associated with the user.

Despite the advantage of adware to make a profit for the developers, it can endanger users' privacy. The adware can collect the required information by continually monitoring the search toolbars of browsers without the user's awareness or permission. In extreme cases, the adware sells this private information to other entities without the awareness or permission of the user. There is a solution to combat adware and enhance the online advertising system's security and privacy, which is called ad-blocking. Ad-blockers are applications that help users passively stop pop-ups ads and banners from displaying in their browsers [55]. AdBlock, AdAway, and AdGuard are ad-blocking add-on (or browser extension) software that can be added to the browser to prevent adware.

Early digital advertising efforts developed intrusive formats such as pop-up ads or autoplaying audio/video ads. This led to consumer demand for ad blockers, applications that allow users to passively block advertising from showing up in their browsers

H. Payment of commissions in online advertising systems

In this subsection, we briefly explain how advertisers pay commission fees to commissioners and publishers.

When advertisers receive valid traffic generated from impressions or clicks, they have to pay the publisher. The commissioner also earns a fraction of this income. If the advertiser uses a similar scheme to pay the publisher, then the commissioner's percentage will be calculated at a fixed rate. For example, in the case where an advertiser pays a publisher per click (or impression), and the publisher receives the money based on the number of clicks (or impressions), then the commissioner receives a fixed payment.

However, an advertiser may pay based on the number of sales, while the publisher earns per click (or impression). This practice is known as an *arbitrage campaign* [52]. In formal terms, an arbitrage campaign is one where the advertiser uses different payment metrics to pay the commissioner and publisher. In an arbitrage campaigns, the commissioner should ensure that its share of the profit from the advertiser is more than the publisher's payment; otherwise, the commissioner loses money. In reality, advertisers prefer to pay based on sales, while publishers prefer to receive income according to the number of impressions or clicks. Hence, Internet advertising schemes are mainly arbitrage campaigns. However, some advertisers may prefer to pay on the basis of clicks or impressions for product branding.

I. Main Security Goals in Online Advertising System

Cybersecurity aims to protect a company's digital assets against cyber attacks. Cybersecurity can be achieved by using appropriate security controls to provide several security features such as deterrence, prevention and detection of cybercrime. The primary goal of cybersecurity for each system (e.g., online advertising system) is to ensure three principles, including confidentiality, integrity and availability (CIA) of data and services. The CIA is essential in cybersecurity because it provides essential security features, helps avoid compliance, ensures business continuity, and prevents reputational damage to the organization. Confidentiality refers to protect information from unauthorized access. The online advertising system is responsible for protecting consumers' private information. Integration refers to protecting data against deletion or modification by unauthorized individuals or systems to ensure the accuracy and consistency of information. Any tampered data injected into the online advertising system can interrupt the system functionality. The term availability refers to ensuring that the system is available only for permitted users whenever required. Lack of availability in the system can cause severe damages.

IV. Vulnerability of Online Advertising Systems

Online advertising systems are vulnerable to various types of attacks, and in this section, we present a taxonomy of current attack methods.

Fig. 6 illustrates the proposed taxonomy. The taxonomy sets out the major fraudulent activities in response to questions about who does what and how. "Who" responds to question about whether the fraud is created directly from human or nonhuman users. "What" intends to classify fraud based on the target revenue model. "How" responds to how a fraudster performs fraudulent acts for certain types of fraud. Based on the above considerations, we identify three dimensions for classifying advertising fraud. The classification mainly depends on whether the fraud targets ad placements, ad traffic, or user actions. The first type of fraud is placement fraud to manipulate/modify the publisher's websites or the content displayed on users' devices to increase impressions or clicks [56]. The second type, called traffic fraud, aims to create fake traffic to increase the number of impressions or clicks generated from separate sites or locations. For example, fraudsters can increase impressions and clicks on publishers' websites by using botnet or crowd. The third type is called action fraud that targets users' actions in order to generate revenue. For example, attackers may hire people to download or send forms to convert or send fake cookies to receive commissions as affiliates using robots. We classify online advertising attack methods into four main categories: hacking campaign account [57], click fraud, inflight modification of ad traffic, and malvertising.

A. Hacking Campaign Account

The threat of hacking in online advertising arises due to unauthorized access to campaign accounts [12]. In search engine advertising, companies aim to attract customers by improving the visibility of their advertisements in results pages. Online campaigns can quickly adapt information in their ad campaigns, which are more flexible, targeted, and tailored than traditional marketing campaigns. The flexibility and time savings of online campaigns guarantee that the transaction processing will be fast. An example of this is AdWords [58], a tool developed by Google to allow advertisers to create online campaigns in only a few minutes. However, despite all the above advantages to online business, online campaigns face with many challenges, including security and privacy.

We illustrate this with an example. Consider the case where an advertiser creates an AdWords account. Users navigate via the web to run search queries, and advertisements can be presented on the websites of publishers or on the search engine network. If an adversary takes control of the advertiser's AdWords account to launch an attack, this is known as hacking campaign account. The consequences of campaign accounts being hacked include blocking, limited access or unauthorized entry to the account of the advertiser. The availability of short-term online campaigns will also be limited. These results may lead to significant reputational damage, loss of money, and violations of user privacy. Fig. 7 illustrates the hacking of an advertiser's AdWords account.

B. Click Fraud

Online advertisements help to develop a healthy Internet, since they provide financial support for the online businesses. The emergence of *click fraud* (also known as malicious clicks, or click spam [59]) therefore poses a serious security risk to the Internet ecosystem. Click fraud refers to cybercrime activity that is carried out either manually (using human clickers) or automatically (software-supported) to generate fraudulent clicks on the advertisement to make illegal profits.

Fraudulent clicks can damage the health of online businesses, since these clickers can increase their profits or deplete the advertising budgets of their competitors. They achieve this by clicking on advertisements with no actual interest in the content.

In the manual approach, fraud consists of hiring a group of people to increase fraudulent traffic, while automatic click fraud attack is usually based on the use of *botnets*[6] [60]. Malicious software called a "*clickbot*" [61] is one example of this use of botnets to generate fraudulent clicks automatically [62], [63]. Using a clickbot to launch a click fraud attack is more efficient than the manual type of attack, since it can perform automatic clicking over a time period of several minutes to avoid detection. We categorize click fraud into two types, *crowd fraud* and *conventional ad fraud*, as described the following subsections.

1) Crowd Fraud: The emergence of *crowdsourcing* [64] has led to a novel form of fraud in online advertising, since it can broadcast a large number of tasks to a numerous online

[6]To build a botnet, a botmaster (an entity that controls the botnet remotely) needs a network of software robots – i.e., bots – that are run independently and automatically.

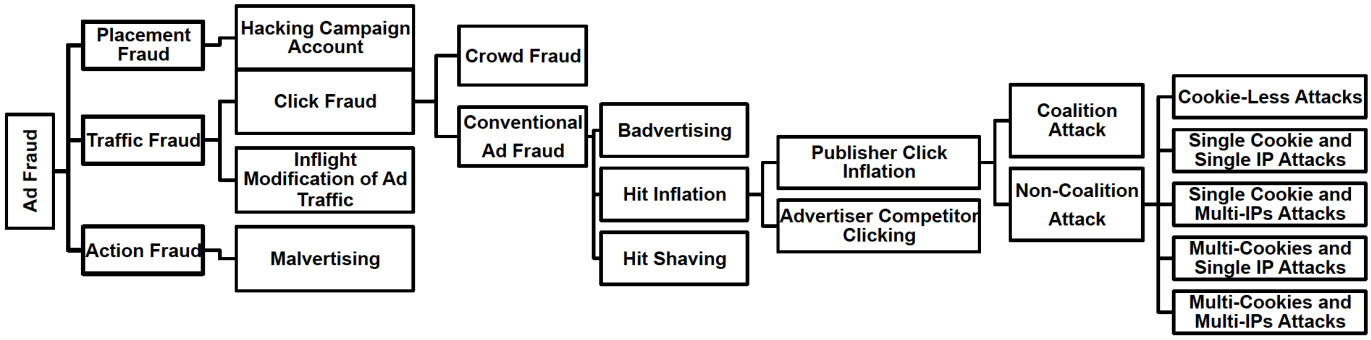

Figure 6: Proposed taxonomy of ad fraud attacks in online advertising systems.

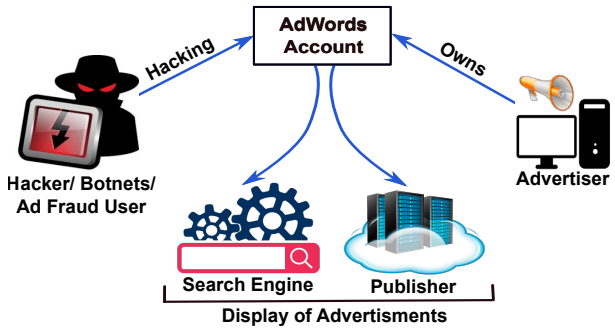

Figure 7: Hacking campaign account attack in online advertising network.

workers. Due to the openness of crowdsourcing systems [65]–[67], a crowd of workers can easily be recruited via malicious crowdsourcing platforms to perform an attack against a competitor or to increase their advertising expenses. There are many differences between automatic fraudulent behaviors (conventional fraud), and frauds carried out by humans. For example, a vast number of workers via crowdsourcing platforms can be involved in human-generated fraud, while automatic fraudulent traffic can be deployed relatively few machines. A difficulty also arises in differentiating normal and no distinct traffic induced by real humans from the noisy traffic generated by machines. Methods used to detect conventional fraud therefore fail to identify these human-generated frauds. The phenomenon of exploiting a group of real humans to increase fraudulent traffic in online advertising is termed *crowd fraud* [68].

2) Conventional Ad Fraud: In contrast to crowd fraud, which is carried out by large numbers of attacking machines, normal and no distinct click behaviors by each web worker, the limited fake traffic generated by each web worker, conventional forms of advertising fraud often have specific features in terms of individual behavior patterns, with few sources and large amounts of traffic. In this regard, the detection of conventional fraud is more straightforward than crowd fraud [68].

We divided the conventional advertising frauds shown in Fig. 6 into three categories: *badvertising*, *hit shaving*, and *hit inflation*. A brief overview of how these attacks are carried out on online advertising ecosystems is given below.

- **Badvertising.** Gandhi et al. defined badvertisement as a kind of camouflaged click fraud attack on the advertising industry [69] that silently and automatically generates click-through on an advertisement when users visit the website. This attack can not only remain undetected by web publishers, but also does not compromise the user's machine. Unlike a traditional malware-based click fraud attack [70], badvertisement is a stealthy offense in the form of a malicious mutation of spam and phishing [71] attacks, except that this attack targets the unaware advertiser as the victim rather than an individual. This is very worrying, since it is easier for an attacker to deceive an individual into visiting a web page than to damage a machine with malware. This attack artificially and stealthily increases the number of clicks on ad banners hosted by the fraudster or unaware associates to generate more revenue for the attacker through advertising. The revenue generated in this way is transferred from the advertiser to the hosting websites by the fraudster. Badvertisement has two main components: (i) delivery, which either transfers consumers to corrupt data or corrupt data to consumers; and (ii) execution, which automatically and invisibly displays advertisements to a targeted user. This stealth attack can be accomplished by corrupting the JavaScript code that is downloaded and executed by the client's browser to publish sponsored advertisements [72]. Online advertisement systems typically work by placing a JavaScript snippet file into a publisher's web page. Whenever a user visits this page and downloads an advertisement from the ad server, the JavaScript file will be executed. Downloading the ad causes the frame in the JavaScript file to be rewritten with the HTML code required to show the advertisement. The publisher relies on the click-through payment process to count the number of times the user clicks on the link to the ad provider's server. Finally, the user is referred to the ad client's website. This scenario is illustrated in Fig. 8.
Badvertisements run extra malicious scripts to automatically deploy clicks. In a nutshell, after running the script code and rewriting the frame, the malicious script parses the HTML code and compiles all links. It then changes the web page to embed an HTML *iframe*. If the user decides to click the link, the *iframe* will be activated in the background, and loads its content to exploit the user (Fig. 9).

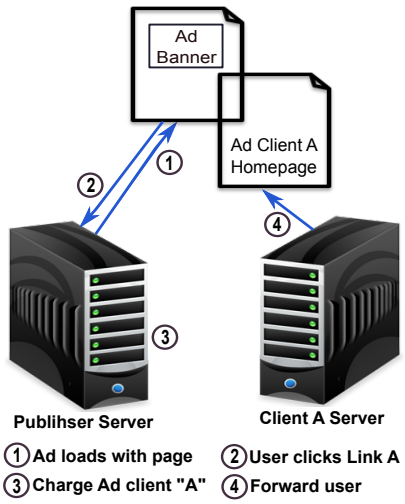

Figure 8: Typical online advertisement services.

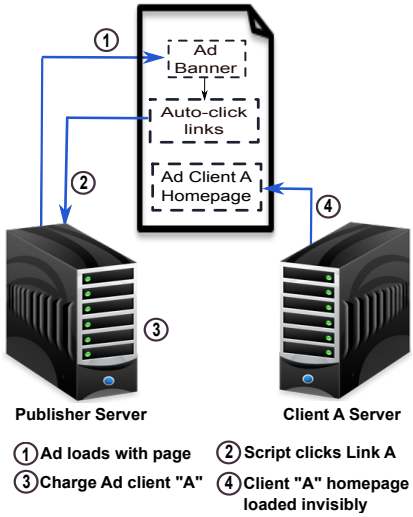

Figure 9: Auto-clicking in a hidden badvertisement.

- **Hit Shaving.** As previously mentioned, advertisers often prefer the CPA model to pay the publisher based on the desired user action, rather than for each click on their ad. However, the CPA model is vulnerable to hit shaving (also called *deflation fraud* [73]). In this attack, a fraudulent advertiser undercounts the real transactions to pay a lower commission fee.
  Before describing how the hit shaving attack is applied in an advertising network, we need to give an overview of the mechanisms used in click-through payment programs.
  Advertising has become a pivotal technology on the Internet, as confirmed by the growth of click-through payments. The main entities involved in click-through payment programs are the user who views the page and clicks on a link, the referrer who exposes advertising material to the user, and the target site running the click-through payment process.
  A click-through payment system works as: we suppose that there are two websites A and B, and that A can refer the user to B. Hence, whenever B receives a referral from A, B has to pay the webmaster[7] of A for this reference. In more detail, when a user views web page A and clicks on a link that refers the user to web page B, then A should receive money from B. In other words, the user has "clicked-through" A to reach B. The use of a click-through payment program by the webmaster of B leads to an increase in traffic to the website, since other websites display links to B. However, since the underlying infrastructure of this structure is based on the HTTP protocol, it is exposed to attack.
  For a better understanding of how this mechanism is vulnerable to fraud, we review the procedure used to exchange HTTP messages (see Fig. 10) during a click-through event. As illustrated in Fig. 10, when users view a web page from site A (called the referrer), the HTTP procedure is executed. Site A includes a link to site B (called the target), and agrees to take part in the process of click-through payment to site B. The customer's browser sends a request to load the page from site B when the link is clicked. Site B can identify the site from which the requested web page originated (i.e. where the user are is being referred from) simply by checking the referrer field in the HTTP header.
  The previous explanation should reveal that the click-through payment system has the potential to be exploited for fraud. The problem arises from the lack of communication between A and B after the user clicks on the link. A cannot verify how many times its web page has referred users to the targeted page, and as a consequence, B is able to omit some of the click-through events from the referrer, in a scheme called *hit shaving*. In addition, although the referrer site can detect that the target site has shaved its referrals, it cannot provide proof of this to a third party. A can also conduct fraud against B by generating false requests in order to increase the payment from B, and this is called *hit inflation*. In brief, hit shaving is a form of fraud by a dishonest advertiser who can undetectably change the number of clicks received from a publisher in order to pay a lower commission fee [52], [74].
- **Hit Inflation.** This is a fraudulent activity performed by an adversary to inflate the hit count, in order to boost revenue or hurt competitors.
  In [75], a sophisticated type of hit inflation attack is defined

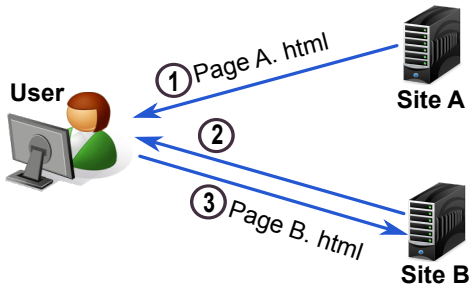

Figure 10: Workflow for a click-through system. Step 1: user retrieves Page A. html from site A (referrer site). Step 2: user clicks on a link in site A and requests the page from site B (target site). Step 3: Page B. html on site B will be uploaded for the user.

[7]The webmaster is the person controlling the content served to the user.

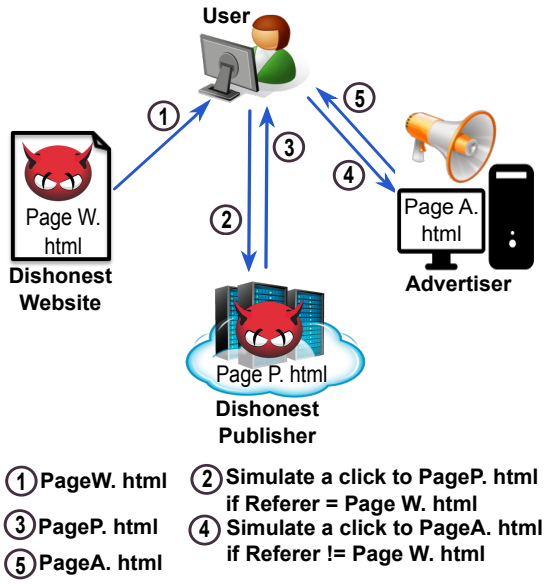

Figure 11: Hit inflation attack on online advertising network.

that is very hard to detect. Fig. 11 illustrates this attack scenario, which involves an association between a fraudulent website (W) and a fraudulent publisher (P), where W uses a script code to silently divert a user to P. The scenario starts when a user simulates a request or click to fetch page W. html from W (step 1). However, the user is redirected to page P. html (step 2). P has two forms of the web page: a manipulated form and a valid form. P will show a manipulated web page to the user when the referrer field in the HTTP request shows W (step 3) and clicks the ad by itself without knowing the user. Otherwise, P will direct the user to the valid web page, and the user is free to either click on the ad or not (step 4, 5).

Publishers and advertisers are the two entities in online advertising systems that are the major sources of inflation attacks. The two most common types of hit inflation attack are called *publisher click inflation* and *advertiser competitor clicking*. We briefly illustrate both types of attack below.

Publisher click inflation. In publisher click inflation, a dishonest publisher is motivated to artificially inflate the click-through count (without real interest in the content of the advertisement) to obtain more income from ad networks. As discussed earlier, if the advertiser wants to present its advertisements on the publisher's website, the publisher enters into a contract with the broker (commissioner). The publisher then gains income from advertisers through the user-generated traffic that they send to websites of advertisers. Obviously, the more clicks the publishers earn, the more money they generate. Consequently, this opens the way for malicious publishers to create illegal revenue by increasing the numbers of clicks, impressions, and actions on their websites.

Publisher click inflation attacks can be classified into two categories: *non-coalition* and *coalition* attacks [76]. The former is performed by a single publisher (one fraudster) who solely generates traffic to its resource(s), while the latter involves a coalition attack among a group of publishers who share their systems. If we can detect both categories of attack, we can claim that the problem of hit inflation is solved.

Launching a coalition attack has several benefits for fraudsters. Firstly, the possibility of fraud detection decreases because the attackers do not need to reuse their resources to generate more attacks [77], making it difficult for detection algorithms to identify the relationships (e.g., the relationships between the cookie IDs and IP addresses of the resources generating traffic and the sites of fraudsters) between each fraudster and all the attacking machines. Secondly, the cost of launching an attack is reduced by sharing resources rather than increasing the number of physical resources.

The study in [52] classifies non-coalition attacks according to the number of IPs and the cookie IDs of the system, and the way in which the commissioners recognize the machines of the surfers (potential Internet customers). When customers visit a website, this traffic has certain fixed characteristics which are different from automatic traffic, and typically involve relationships between IP addresses and cookie IDs. Hence, if fraud detectives find inconsistencies between the cookie IDs and the IP addresses, they can investigate manually by selecting a subgroup of the publishers to detect the attack. On the other hand, when dishonest publishers want to launch the attack, they can leave a false fingerprint for the relationship between the IPs and cookie IDs in order to confuse the detection mechanisms.

The attack can be launched by one or multiple IPs, and these addresses may be associated with no, one or multiple cookie IDs. There are therefore six possible types of attack based on combinations of IPs and cookie IDs, as follows.

1) **Cookie-Less Attacks.** A fraudster can launch cookie-less attacks in at least two known ways. Firstly, there is the option for the attacker to turn off cookies on the system(s) which plan to launch the attack. Secondly, a fraudster can employ commercial services called network anonymization, which are designed to protect the privacy of users [78] and to block third party cookies to give more cookie-less traffic.
2) **Single Cookie and Single IP Address Attacks.** In this type of attack, a dishonest publisher can employ a script to launch an attack from one machine with a fixed IP and one cookie ID. The author in [79] provided an example of this type of script.
3) **Single Cookie and Multiple IP Addresses Attacks.** Attacks of this type are more widespread among fraudulent advertisers than fraudulent publishers, since changing the IP address of the attacking machines is more convenient than changing the cookie ID. The commissioner shows the most profitable advertisements to Internet customers that have not recently been displayed. In addition, if repeating the same cookie sends to the commissioner, as a consequence, the same advertisements display to the users. Hence, a dishonest advertiser can start the attack by visiting the publisher's website and continuing until

the broker shows advertisements from its competitors. The fraudster then stores the cookie ID with the intention to continuously applying the ID to force the broker to show the advertisements from its competitors. In this way, it can simulate clicks on advertisements in order to drain its competitors' advertising budgets.

4) **Multiple Cookies and Single IP Address Attacks.** An attacker can perform this type of attack in various forms. The simplest method is to connect different systems to the Internet via a single router, and then execute various scripts on the systems. In this way, the attacker can simulate receiving traffic with several cookie IDs but a single IP address. However, this type of attack is not economically viable. This attack suffers from a resemblance to the regular Internet traffic problem, in which different customers connect to the Internet with various cookie IDs using a single IP address through an Internet Service Provider (ISP).
   In the second form, in order to make the attack more comprehensive and sophisticated, the attacker can connect several machines to the Internet via an ISP with a similar IP. To reduce the impact of this malicious attack and defraud the detection algorithms, a dishonest publisher can combine fraudulent traffic with regular traffic.
5) **Multiple Cookies and Multiple IP Addresses Attacks.** Performing and detecting this class of attack is difficult. The malicious publisher uses various valid cookies and IPs. The attacker can perform this type of attack by using the cookies and IPs in multiple forms. In the most simple form, which is not economically viable, the attacking publisher has access to various machines with different accounts with ISPs. Another method is to use botnets, such as spyware and Trojans. The aim of using a botnet [80] is to simulate impressions and clicks on the website of the attacker by sending the proper HTTP requests while exploiting the cookies and IPs of legal users. The traffic generated in this way is very similar to regular traffic.
   This type of attack can be considered a more sophisticated version of some of the above examples. Suppose that the publisher has access to different legal cookies and IPs, such that IPs can generate random or can be pre-assigned. Then, whenever a cookie ID and a pre-assigned IP is used in the attack, the attack can be considered a more sophisticated version of the multiple cookies/single IP attack that uses multiple IPs. In contrast, when the IP is selected randomly, this results in the use of identical cookies for different IPs. This attack can also be considered a more sophisticated version of the single cookie/multiple IPs attack with multiple cookies.

Advertiser competitor clicking. In this attack, malicious advertisers carry out hit inflation attacks against their competitors to drain their advertising budgets. In the case where competitors have limitations on their daily advertising budget to participate in bidding, fraudsters can increase the probability of their advertisements being displayed by winning the auction.

More generally, the consequences of fraudulent traffic include reducing the reputation of the commissioner and attracting fewer advertisers, and also may lead to extra fees or penalty payments for advertisers [81], [82].

C. Inflight Modification of Ad Traffic

In [15], a new form of ad fraud was presented that involves the inflight modification of advertising traffic (also called a Man-In-The-Middle (MITM) attack). An well-known example of this type of fraud is the Bahama botnet, which allows malware to force compromised machines to show surfers altered advertisements, and to change the results of searches [87]. The key difference between this attack and traditional click fraud is that in the latter case, ad networks can gain income from fraudulent clicks, while inflight modification of ad traffic can allow either traffic or income to be diverted from the ad networks to the attacker's server.

In the Bahama botnet, compromised systems direct users to a malicious site that looks identical to real Google search results. In this case, the attacker leads the user traffic to another site of the attacker's choosing, such as a fake website, by corrupting the translation of the Domain Name System (DNS) on the infected systems. For example, when a compromised user clicks on advertisements on Yahoo or Google, they are silently redirected to a server that is under the attacker's control. Consequently, the domain name/hostname "Google.com" (or Yahoo.com) translates to an IP address that belongs to the attacker and not to Google (or Yahoo).

Moreover, a viewer can enter a query into the input box that appears to belong to the Google server, but the traffic is in fact redirected to the poisoned server. The user is sent back (malicious) results for the given query from Google, i.e. results that are different from the real ones. Clicking on these fake results leads to the click-through payment program being triggered, and thus to advertisers receiving money, meaning that click fraud has taken place. In the case of Bahama botnet, income is diverted from main ad networks to smaller publishers and ad networks.

The adversary can also use botnets of compromised wireless routers rather than compromising the users' systems [88]. In this scheme, the wireless router, which is hacked by malware, is converted to a bot. The botnet master can then give instructions to launch an inflight modification of traffic attack to transmit traffic through the router. Many public hotspots operate on this model by providing users with free Wi-Fi while embedding advertisements in the users' traffic to earn more money.

Inflight modification of ad traffic has a drawback in that if a user clicks on the displayed advertisements, profit is generated for the fraudster rather than the legal ad network. Hence, this attack weakens the network industry model. It is worth noting that there are other catastrophic effects of these attacks in terms of the security of end-users (as it leads to malvertisement rather than legitimate advertisements), and also a loss of reputation and income for legal advertisers.

D. Malvertising

The primary goal of the online advertising system is to reach users, and these entities are therefore more vulnerable to threat in this system than the others.

When a user navigates the Internet and visits different websites associated with a single advertiser, the same cookies are allocated to the user. In this way, the ad provider can track the user's online activities by compiling the information from the cookies without the user's permission or consent. The consequence of this tracking is that the user's privacy is violated. Moreover, users can be involved in fraud (e.g., click fraud) without realizing. Malvertising (malicious advertisements) is another fast-growing security threat on the web that can infect users [15].

Malvertisement is a platform for distributing malware by injecting malicious code into legitimate ad networks. Malware can be categorized as worms, viruses, Trojans, rootkits, ransomware, botnet, etc [83]. Malicious ads take advantage of browser vulnerabilities to infect the victim's system, persuade users to download and to install malicious software or redirects users to websites they have not planned to visit (in the case of ransomware) [84].

As previously mentioned, there are several entities involved in an online advertising system, making it a complex network. This complexity and the use of multiple redirections between different components allows attackers to embed malicious content (e.g., malicious advertisements) in places that publishers and ad networks would not expect. For example, an report by Blue Coat [85] shows that JavaScript code can be served by an ad server to inject a hidden iframe tag into a benign site instead of fetching legitimate advertisements. In this scheme, the iframe commands the browser of the victim to silently interact with a malware server, allowing a PDF exploit file to be downloaded. Both publishers and advertisers in the online advertising ecosystem have the potential to launch a malvertising threat; for instance, an advertiser can easily inject a malicious ad into a legal ad network to trigger malvertising. As a result, the advertising network may deploy those advertisements on publishers' websites, and users will then access them by clicking. Moreover, publishers can insert malicious content into their sites to indirectly cause a consumer to install malware. In this scheme, users even do not need to click on advertisements to activate malware. One of the most common forms of malvertising is flash-based advertisements [86], in which an Adobe Flash File (also referred to as a SWF) that contains malicious script is abused by criminals to run arbitrary commands. Creating advertisements with animation and sound in an SWF file allows the advertisers to attract a greater audience, and this means that Flash is vulnerable to being used in malicious attacks. It is therefore clear that attackers can spread malicious advertising via Flash, which is known as "malvertising" [86].

E. Putting All Together

This section explains, for each type of attack, how an adversary can exploit the risks in online advertising systems and conduct ad fraud. We review the methods that fraudsters use to gain money from online advertising systems, present a brief analysis of the goals of these attacks, and identify which revenue model is the adversary's goal and which components of the online business system could be the primary targets. To aid in comprehension, the results of this comparison are presented in Table II.

The technical reasons for the threat are mainly due to aspects of human or professional weakness or negligence. In general, by applying regulatory frameworks in the industry, at least some security issues can be addressed. However, due to cost and lack of awareness, the online advertising security framework does not seem sufficient to combat threats.

In Fig. 12 , we consider three dimensions to identify the relationship between threats, which players in the system are the attacker's target, which revenue models can be affected by the threats, and the CIA target. For example, click fraud can affect advertisers and publishers in the system. It can also compromise the confidentiality and integrity of the system. All revenue models, including CPC, CPA and CPM, can be affected by click fraud.

Among the attack, only Inflight modification of ad traffic attack can be applied to all components of the ecosystem. The hacking campaign account and malvertising only affect the advertiser and the user, respectively. Click fraud can endanger both the advertiser and publisher. Among the CIA principles shown on the left side of Fig. 12, only confidentiality can be compromised by all threats. Hacking can affect availability of the system. The integrity of the system can be compromised by click fraud, malvertising, and IMAT. All the revenue models can be infected by click fraud, while Inflight modification of ad traffic can only impact on CPC model.

V. COUNTERMEASURES FOR ONLINE ADVERTISING ATTACKS

In this section, we discuss several approaches proposed in the literature to combat various types of attacks on online advertising systems. Table III summarizes the existing detection methods for online advertising systems and gives a preliminary overview of the pros and cons of using these methods.

A. Countermeasures to Hacking Campaign Account

When Google AdWords [91] was launched in 2000 and quickly became Google's primary source of revenue, it soon became a rich source of targets for ad fraud attacks.

Various reports and forums have discussed the fact that the majority of Gmail address and passwords are used to hack campaign accounts. The different approaches used to hack Google AdWords accounts can be categorized as (i) brute force login attacks; (ii) email spoofing; and (iii) malware and spy tools for obtaining user account information [92]. When fraudsters enter to a campaign account, they can duplicate campaigns. Attackers can also generate enormous numbers of clicks and redirect destination URLs to other companies [93], [94].

One of the more straightforward options for preventing this type of attack is to select strong passwords. The security and protection of an account can be increased by choosing a

Table II: Summary of attacks, description, attack goal, revenue model goal and primary component targets in online advertising system.

Attack	Description	Attack Goal	Revenue Model Goal			Primary Component Targets			
			CPC	CPM	CPA	Advertiser	Publisher	User	Ad Network
Hacking Campaign Account [12]	Unauthorized access to campaign accounts	Hacker aims at taking over control of advertiser's account	✗	✗	✗	✓	✗	✗	✗
Crowd Fraud [68]	Malicious behaviors by humans against competitors for specific targets	Increase fraudulent traffic	✓	✓	✓	✓	✓	✗	✗
Badvertising [69]	Utilizing malicious JavaScript code to publish invisible automatic advertisements in the user's browser	Increase the number of clicks	✓	✗	✗	✓	✗	✗	✗
Hit Shaving [89]	Dishonest advertisers claim that they received less traffic than in reality	Dishonest advertisers omit to pay commission on some of the received traffic to the publisher	✓	✗	✓	✗	✓	✗	✗
Hit Inflation [90]	Artificial inflation of the actual amount of traffic	Economic advantage from over-counting the numbers of transactions	✓	✓	✓	✓	✓	✗	✗
Inflight Modification of Ad Traffic [15]	Infecting the system to show altered search results along with modified advertisements to the users	Generate revenue fraudulently for ad networks and publishers	✓	✗	✗	✓	✓	✓	✓
Malvertising [15]	Perpetrators inject malicious code into legitimate online advertising networks to spread malware	Malicious code, eventually, attempts to redirect users to malicious websites	✗	✗	✗	✗	✗	✓	✗

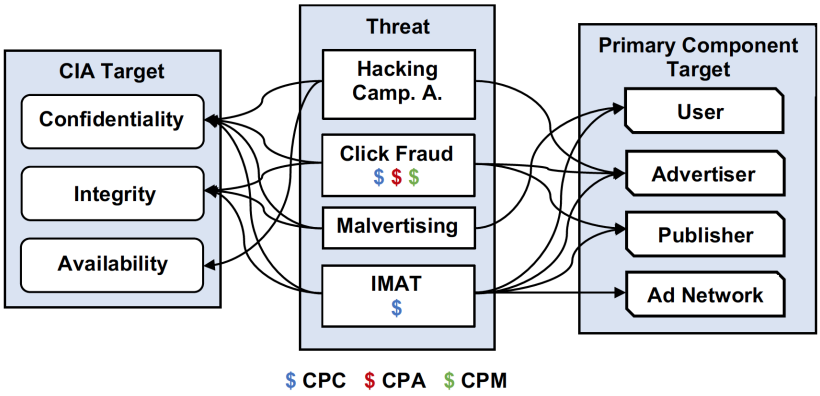

Figure 12: The linkage between threats, CIA target, and primary component target. CPC, CPA, and CPM are the revenue models. IMAT := Inflight Modification of Ad Traffic, Camp. A. := Campaign Account.

complicated and lengthy password combining letters and numbers with special characters, and by changing the password regularly. It is also possible to monitor and control browsers with phishing filters [95], especially when connecting to unsecured WIFI connections and signing in to Google accounts. Industries and business owners should have a contingency advertising plan for monitoring their revenue trends to handle the drop in revenue caused by fraudulent campaigns.

To detect hacking attacks in online advertising, a daily check can be carried out of accounts to guarantee not only the cost-benefit ratio and performance of the campaign but also to protect the campaign from reputational damage and loss of income. It is vital to monitor and analyze the performance of each AdWords campaign on a daily basis [12].

B. Countermeasures to Click Fraud

In this section, we first discuss countermeasures for crowd fraud and then countermeasures for conventional ad fraud.

1) Countermeasures to Crowd Fraud: The techniques typically used in business markets for crowd fraud detection mainly emphasize human interactions, including prior knowledge of malicious queries and principles associated with filtering. These approaches are costly, and tend to become invalid quickly because web workers may change their patterns of behavior to avoid detection.

To address these problems, the authors of [68] investigated the group behaviors associated with crowd fraud, and found that compared with the individual actions of each worker, which may involve considerable noise, group behaviors were more continuous.

In formal terms, these authors discovered certain typical feature distributions and network functions of crowd fraud that can be effectively applied to detect this activity. They noted the following aspects: (i) moderateness: crowd fraud sometimes targets advertisers or queries with medium hit frequencies; (ii) synchronicity: Internet users participating in crowd fraud can classify into coalitions [96] via which they typically target a distinct collection of advertisers and execute the fraud quickly; and (iii) dispersivity: surfers involved in crowd fraud may search for an irrelevant series of topics and click advertisements from different industries simultaneously.

Based on the attributes mentioned above, the authors of [68] introduced an efficient solution for crowd fraud in search engine advertising, which was divided into three phases: *constructing*, *clustering*, and *filtering*. In the constructing phase, they deleted irrelevant data from raw data logs of queries that did not meet the moderateness condition (e.g., either markedly small or large hit frequencies) to create a surfer-advertiser bigraph in which each edge referred to a single unique click history and included aspects such as search queries and hit

times. Then, they built a surfer-advertiser inverted list for this bigraph for the next phase. In this list, each entry referred to the click history for each unique surfer. In the clustering phase, they described the sync-similarity between click histories to discover coalitions of surfers, indicating synchronicity.

Next, they converted the coalition detection system into a clustering problem that could be solved through a nonparametric clustering algorithm (such as DP-means [97]). After the clustering phase, the percentage of finding coalitions was high, and this caused false detections and therefore false alarms. For instance, in some business domains such as healthcare or games, regular Internet users with related interests may repeatedly click on the same advertisements to receive similar services. Hence, using infiltering, they created a filter for clusters based on the dispersivity to eliminate false alarm clusters.

Since this method does not require tuning of any parameters, it can be applied in real scenarios to find an infinite number of coalitions without human interaction. The authors also built a parallel version of their detection method (by parallelizing the nonparametric clustering algorithm) to make the system more scalable for massive web searching. The results of this experiment validated the accuracy and scalability of their approach. Although the proposed algorithm was capable of detecting crowd fraud, however, it failed to prevent this fraud. Moreover, evaluating the accuracy of the algorithm was hard due to the difficulty of collecting fraud data.

The crowd-sourcing click fraud detection model in [98] was based on a clustering analysis. In order to analyze the data, the study defined the distinct features of denseness, moderateness and concentricity. The model consisted of three phases: preprocessing, group detection and post-processing. In the preprocessing phase, queries that were less likely to be fraudulently clicked were removed. In the group detection phase, a crowd-sourcing click fraud group was treated as equivalent to a cluster, and a DPMeans clustering method was used to detect malicious groups. In the post-processing phase, demand clicks checked by mistake were filtered. Although the model achieved good convergence and extensibility, the complexity of this method was excessively high.

A novel crowdsource based system presented in [99] called Click Fraud Crowdsourcing (CFC) addresses the problem of crowd fraud in the mobile system by protecting both advertisers and ad networks. The method consists of four components: ad banners, ad network APIs, advertisers' website, and CFC component, which acts as a click fraud detection engine. The program can monitor the user's duration on the advertiser's website and simultaneously collect request data for several ads related to the different ad network, publisher and advertiser. The results showed the proposed method when detecting click fraud, compared to the solutions offered in the literature, while maintaining a high true positive rate (0.9), offers a false positive rate (0.1).

2) Countermeasures to Conventional Ad Fraud: Countermeasures for different types of conventional Ad fraud discuss in the following.

- **Countermeasures to Badvertising.** A successful badvertisement stealthily and artificially generates automatic clicks on advertisements when users visit a site hosted by a fraudster, and can persist unseen by auditors from the ad provider. It does not require any specific technical knowledge to run this kind of attack, and any illegal webmaster can perform it [69].

At first glance, it may seem easy to detect this attack by controlling the CTR from the intended domain, but this is not always the case. For example, the attacker can generate both click-throughs and non-click-throughs by manipulating the traffic in the damaged page, while the customers correlated to those types are not informed of the advertisement. It should be noted that the owners of the site who earn income for a "badvertiser" may not be aware of their participation in running the attack. For example, the owners of a domain may be pretending not to know of the existence of an attack, or may be fooled by a corrupt webmaster. The former case corresponds to a phishing attack [100].

Developing tools for the discovery and prevention of frequent click fraud attacks is a major aim of industries in this field. AdWatcher [101] and ClickProtector [102] are two well-known companies that try to detect and prevent such attacks. The most common attack types are malware-based, which use automated scripts, individuals hired to deplete their competitors' advertising budget [103] or proxy servers to generate fake clicks. These attacks can be detected by tracking the IP addresses of the systems that generate the clicks or by distinguishing the click registered domains. Companies try to identify aspects such as duplicate clicks for a specific ad by a single IP address or irregularities in the traffic history, and to carry out careful analyses. However, a badvertising attack cannot easily be detected using these approaches, and there is a pressing need for other types of mechanisms to detect and prevent this attack.

The countermeasures discussed in [69] involve the construction of a ad code to detect an attack when preventing it is not possible. These methods can be divided into two types: *active* and *passive*. Active methods are used to detect click fraud, while passive methods are used to monitor the progress of a click fraud.

In formal terms, an active client-side solution is based on interactions with search engines, the execution of public searches, and visits to the resulting sites. It can carry out web surfing in a manner similar to the user. An active mechanism can conceal its status such that an agent cannot recognize it as a robot, and can present itself as a real user to the servers in order to interact with the agent and other entities.

In contrast, passive client-side approaches monitor the actions performed by users that lead to a click. It is possible to trap requests for advertisements by virtual execution of JavaScript code, and any attempt to display a specific web page in a way that it should be occurring after a click can be considered a fraudulent request. It should be pointed out that although this solution can be used against automatic click-fraud, it cannot be applied to protect a system against a type of attack that first creates a significant delay and then performs a click fraud. The only way to do this and to capture a delay is to let the virtual machine randomly select scripts for generating a delay.

We should recall that long delays are not preferred by attackers, since their session might be disconnected from the target before they can generate a click on the website. Passive client-side methods can be included with security toolboxes or anti-virus programs.

Another form of passive scheme is an infrastructure component. That can detect click fraud by shifting traffic, identifying candidate traffic and mimicking the system of the user receiving the packets. Example applications of infrastructure component schemes include an ISP-level spam filter and Mail Transfer Agent (MTA).

We can conclude from a performance analysis that if a client-side detection mechanism is installed only by a small proportion of customers, these attacks become entirely unprofitable.

- **Countermeasures to Hit Shaving.** The author of [104] explained that the rationale behind all inflation and deflation fraud (also called hit shaving) is a lack of knowledge. In both attacks, the entities who perform the fraud may under- or over-count transactions for financial gain, and it is difficult for the victim to prove the damage that arises. As a consequence, a general technique for detecting these frauds is to collect information relating to the victim's claim.

  For example, in the case of deflation fraud, the authors of [104] proposed the use of an online Trusted Third Party (TTP) as a mediator to facilitate interactions between two parties. To detect deflation fraud, the publisher must collect as much information as it can, based on the advertiser's claim. In a nutshell, the more info a publisher can gather, the stronger the detection scheme. The disadvantage of this solution is that it cannot be applied to the online advertising ecosystem. Similarly, in Google's AdWords, the publisher directly monitors the transactions. The methods mentioned above suffer from a lack of scalability and efficiency, since the publisher can interfere in the business operation of the advertiser and in turn with the TTP.

  In [73], an efficient and flexible mechanism was proposed to relax the security solution slightly. The authors point out that there is a certain level of tolerable counting error for the publisher if they miss some transactions. Their mechanism involved a novel deflation fraud detection scheme that applied cryptography and probability-based techniques with the following features: (i) the publisher can detect deflation fraud with a high probability of success, and the security parameters can be tuned by the publisher to provide a balance between cost-effectiveness and security assurance; (ii) under these conditions, the web publisher can estimate and detect the expected number of transactions on a large scale; (iii) although a transaction takes place only between advertiser and users, the proposed scheme is easy for end-users, since they are not required to keep any secret information; (iv) the costs (such as computation, communication, and storage) of this method are all constant, making the scheme efficient and scalable.

  The proposed hybrid method does not require the cooperation of a third party, and retains the simplicity of the current advertising system. The publisher also has the option to tune the security parameters to balance the security and cost of the model. The drawback of the proposed scheme is the need for manual tuning of parameters by the publisher.

  Although there are many click-through payment mechanisms on the web, the publishers cannot verify whether they have received payment for each click-through to the target site. This allows for hit shaving, in which the target sites can avoid paying the publisher sites for some click-throughs.

  The study in [89] proposed some rapid and straightforward approaches to enable referrers to track the number of click-throughs, allowing them to be aware of how much money they are owed. These methods included ways of creating web pages and Common Gateway Interface (CGI) scripts that offer the referrer webmasters a greater ability to monitor the numbers of legal clicks, and also which pages the users click. They implemented these approaches by placing upper bounds and lower bounds on referrals. These are effective techniques that do not require awareness or cooperation by the webmasters of the sites to which the referrals are made. The authors also explore more aggressive approaches for cooperating with the providers of click-through mechanisms, to allow webmasters to more accurately control the number of click-throughs. Although this second group of approaches requires cooperation by the webmasters of the click-through payment programs, it does not need trusted webmasters, since any failure to cooperate is quickly detectable. This is a robust solution: a referrer can discover this fraud after 20 times probe even if the target shaves only 5% of the commission. However, this method is not always feasible, for example if the target website sells expensive items. In this method, referrers are expected to report their payments for leads and sales correctly, with the help of the target sites. Although techniques presented here are mainly invisible to the web user, their main disadvantage is the communication overhead for implementing the protocol, which causes it to be an inefficient and inflexible scheme.

- **Countermeasures to Hit Inflation.** Due to the nature of hit inflation attacks, they are an important concern for advertising commissioners [105]. Most research to date has focused on publisher fraud, since this can also be generalized to advertiser fraud. In the following, we therefore concentrate on publisher fraud unless it is specifically necessary to investigate advertiser fraud. We start with examples of classical approaches to inflation fraud detection, and give an overview of cryptography-based methods. Finally, we argue that the application of statistical analysis to streams of traffic is the most appropriate way to detect hit inflation.

  1) **Classical Approach.** Classical fraud detection, also called offline fraud detection, employs a variety of metrics to evaluate publishers according to the quality of traffic to their websites [74]. The quality of traffic can be measured by its adaptation with normal network traffic. In classical detection methods, brokers can store the total traffic in databases and validate the quality (based on certain metrics) of the stored traffic using complex SQL scripts.

     One of the most appropriate metrics is the CTR of the advertisements, which is constant across websites of

the same type [79], while advertisements of different types have different CTRs on identical sites. If the website automatically visits and clicks, consequently, not only produce similar CTRs for the advertisements but rather the CTR of the displayed advertisements deviates from the normal values. Commissioners can develop this technique to monitor the behavior of advertisements by loading empty advertisements into the websites of publishers and checking clicks on these false advertisements. However, classical metrics have several problems. They are not efficient metrics, since fraudsters can easily circumvent traditional tools, and can fool classical detection tools by abusing the site architecture [52] of a specific publisher to model the network metrics of advertisements and gain information about the parameters of the advertisements displayed on their website.

A lack of scalability is the second problem. It should be noted that the average impressions per second currently received by the commissioner is 20K, corresponding to 70M records that need to be stored in a database per hour. It is clear that executing SQL scripts to compute these metrics will lead to a decrease in database performance, and commissioners therefore execute them only periodically. Moreover, the updating of these metrics is also not scalable. Each click on an ad in any site may mean that the statistical parameters and the ranking of the website need to be recalculated.

Thirdly, the classical approach was developed before Internet advertising reached maturity, and hence represents the standard conflict between advertisers and commissioners. Traffic that does not adapt with the network metrics may be legal, although it will be low-quality traffic. Since classical methods are unable to detect malicious intent, they omit legitimate traffic with low quality.

2) **Cryptographic Approach.** There are various cryptographic methods in the literature that can replace offline measures. The central idea behind these is to change the industry standard to give fraudulent publishers less chance to conduct fraud [106]. For example, in [107], a simple model involving e-coupons was developed. In this model, the advertiser exploits cryptographic algorithms to produce coupons and distributes them to the publishers. Then, the publishers redistribute the coupons to users, who can use these cryptographic coupons to purchase items from the websites of advertisers. Web advertisers favor this model because it is based on pay-per-sale. Most publishers prefer to be paid based on the number of clicks or impressions, since this relates to the load on their servers.

   Conversely, advertisers can exploit the model to receive a vast amount of clicks or impressions, which are essential to increase awareness of their brand. The authors claim that the proposed model meets most security and safety requirements; however, the model is vulnerable to hit shaving attack by advertisers.

   The solution proposed by Goodman [108] is to replace the current pay-per-click scheme used in online advertising with a pay-per-impression system. This approach does not involve a monetary cost to the advertiser for click fraud, since they are no longer paid per click. The authors of [109] suggest a cryptographic technique for changing the CPC model to CPA in which valid clicks are identified rather than invalid clicks being removed. This model guarantees the legitimacy of the clicks received by advertisers through a TTP. However, this model requires sharing information between third parties, which is not possible due to the security restrictions in modern browsers.

   Other cryptographic methods rely on assistance from users to identify fraudulent traffic from regular traffic. Different groups of protocols using basic cryptography methods have been introduced to count the total number of visitors viewing a website [89], [110]. One framework requires users to register with a broker, from which the user receives a token from the broker to use free services on the website of the publisher. The broker also shares the corresponding token with the publisher to allow them to recognize registered users. In this way, each time a user visits the publisher's website and sends a token-based authentication to the publisher, access is granted to that free service. The user updates the publisher via a hash function when an authentication token is sent to the publisher. Since publisher cannot predict the cost of the next visit (but can verify the value of the token), the number of user visits stored in the last token is sent back to the publisher at accounting time.

   There are some limitations to this framework. Firstly, it presumes that the users trust brokers to download code to run the hash function [111] and communicate with the publishers' servers. Secondly, it suffers from a lack of scalability, since numerous hash functions are required (one for each user). Thirdly, this scheme needs brokers to identify users uniquely in order to be effective, although exposing personal information on the users to the brokers violates the user's privacy. The last problem can be handled by user registration in the broker's website (by exposing the user's personal information). Brokers can also track and monitor the behavior of users by downloading spyware [112] onto their systems.

3) **Data Analysis Approach.** Many advanced data analysis technologies have been developed to alleviate the problems caused by cryptographic methods. The principal aim of these technologies is to find particular patterns that characterize fraudulent traffic. As mentioned above, a broker needs to deal with the conflict between protecting the user's privacy and security, and the best way to address this challenge is to carry out statistical analysis on collected data (such as cookie IDs and IPs) with the help of temporary user identification.

   Cookies do not store any personal information, and the user has the ability to block, accept, or periodically clear them [113]. IP addresses can also be assigned to the user temporarily, and can be shared with other users. There is therefore no reason to change the industry model and to obfuscate the identity of the users when applying

data analysis methods to cookie IDs and IPs, and these methods can detect fraud with high accuracy [90]. The known data analysis approaches to defending against hit inflation are described below.

Detecting duplicate clicks. Since some publishers try to increase the number of clicks on their websites by clicking the same advertisement, some detection techniques rely on searching for duplicate clicks in the clickstream [72], [114]. The detection of duplicate clicks within a short time (for example single a day) raise suspicion for the commissioner.

In classical data analysis techniques, the commissioner can store the total traffic in databases and run complex SQL scripts to find duplicate clicks within a certain period. However, this method suffers from scalability and performance problems. Storing traffic in the database and then checking them to find duplicate clicks is very expensive for commissioners, since they receive a vast amount of traffic (an average size of around 70M records is generated per hour). In a online scenario, a detection scheme also needs to be fast, and should process the total traffic entry within 50μs. Hit inflation detection is therefore a critical part of streaming and sampling algorithms.

To cope with the above problem, Metwally et al. [114] proposed a fast algorithm for detecting duplicate clicks in data streams. Their algorithm relies on original Bloom filters [115] and aims to find click fraud with an error rate of less than 1%. They provide different solutions by considering three types of window, as follows: sliding windows (finding duplicate clicks corresponding to the last observed part of the stream); landmark windows (keeping particular parts of the stream for deduplication); and jumping windows (a trade-off between the first two types).

The results of an experiment on a real dataset show that within one day, one ad was clicked 10,781 times by users with the same cookie ID. Since the method is successful in identifying fraudulent intent, it can be considered a complementary approach to classical schemes that cannot differentiate low-quality from malicious traffic. However, the method has high computational complexity of order O(n), since it needs to keep active click identifications in its memory until they expire.

To address this problem, two algorithms, namely the Group Bloom Filter (GBF) and Timing Bloom Filter (TBF) algorithms, were developed in [72]. The difference between them lies in the number of sub-windows. The GBF can detect click fraud using jumping windows with a small number of sub-windows, whilst TBF achieves this using a large number of sub-windows. These two algorithms involve simple operations and relatively little storage space, with zero false negatives. The error rate of duplicate detection is also reduced to less than 0.1%. Recently, with the development of ML, this approach has been leveraged to find patterns of difference between fraudulent click data streams. Some researchers have claimed that abnormal click-stream traffic is often a simple reuse of legal data traffic, and have tried to identify click fraud by detecting repeating patterns in a given click-stream for an ad. The Clicktok tool used a Non-negative Matrix Factorization (NMF) algorithm to partition click traffic to identify fraudulent clicks [116]. The authors claimed that the proposed solution reached an accuracy of 99.6%. Despite this high efficiency, however, the solution only works on the client-side. A deep learning-based model was used in [117] to build a multi-layered neural network with an attached autoencoder and generative adversarial network to detect click fraud. The authors of [118] proposed a new way to identify inherently hidden patterns performed by fake or malicious users on ad networks. They argued that training machine learning models directly on fraudulent and benign sequences collected from advertising activities were not easy. The majority of ad traffic is non-fraudulent, and data labelling by a human is time-consuming. Hence, they combined a variant of Time-LSTM cells combined with a modified version of Sequence Generative Adversarial Generative (SeqGAN) to generate artificial sequences to impersonate the fake user patterns in ad traffic. They also reduce computational costs by using Maximum Likelihood Estimation (MLE) pre-training and a Critic network for seqGAN training. They claimed that the use of sequences generated by GAN could increase the ability to classify event-based fraud detection classifiers.

Despite the effectiveness of ML methods for detecting fraudulent clicks, because large amounts of data and sophisticated fraudulent ads are constantly involved in the online advertising system, these methods may suffer from labour-intensive feature engineering and the power of detection algorithms. Also, an attacker could easily modify their fraud patterns based on existing fraud detection characteristics and rules to prevent identification. Hence, many studies have focused on graph-based methods for detecting anomalies (e.g., fraud detection) in systems [119]. The study in [120] proposed a weighted heterogeneous graph embedding and deep learning-based fraud detection method (called GFD) to detect fake applications for mobile advertising. The proposed method has three steps: (i) use a weighted heterogeneous graph to display behavioural patterns between users, mobile apps, and mobile ads, and design a weighted meta path to vector algorithm to learn node representations (graph-based features) from the graph; (ii) use a time window-based statistical analysis method to extract attribute-based features from the sample table data; and (iii) propose a hybrid neural network for fuse graph-based features and attribute-based features classifying fake apps from normal apps.

In [121], a cost-sensitive Back Propagation Neural Network (CSBPNN) architecture was implemented for click fraud detection, and the researchers applied an Artificial Bee Colony (ABC) approach to optimize the CSBPNN connection weights and feature selection. However, many types of click fraud do not rely on legitimate streams

of click data, and attackers are also able to construct data streams with patterns similar to legitimate click data streams through the use of fitting classifiers. In this case, the performance of the above systems will be significantly degraded.

Fabricated impressions and clicks. Other solutions collect ad traffic across user IPs and cookie IDs to identify fabricated clicks and impressions. They are based on finding client behavior (e.g., advertisement traffic) that deviates from normal behavior [74], [90].

Cryptographic and classical methods cannot determine the difference between attacks launched by a single publisher and by a group of publishers (also called a coalition attack). In principle, making this difference is the main idea behind the data analysis approach. Although it is easy for coalition attacks to defraud classical methods, data analysis mechanisms have been developed to try to find evidence of these attacks [90].

Metwally et al. [122] designed a scheme to detect the hit inflation attack identified in [75]. They observed that several websites could cooperate to make fake clicks and consequently improve their business interests, and proposed an algorithm named *Streaming-Rules* to detect hit inflation in an online advertising system. This approach relies on discovering the association rules (defined as forward and backward association rules) between each pair of corresponding elements in the stream. This algorithm requires cooperation between ISPs and brokers. An ISP can recognize which websites are generally visited before a particular website, while maintaining users' privacy [123], by analyzing the entire HTTP requests stream. The authors claimed that Streaming-Rules could discover the association between elements occurring in a stream with tight error guarantees and minimal memory usage.

The solution proposed in [122] is not efficient against other coalition attacks, since it is designed to detect the specific attack described in [75]. For example, if each adversary in the coalition attack takes control of the user's system via Trojans, then the adversary can separate the HTTP request stream by ISP, making it impossible to detect the attack using Streaming-Rules. Hence, in [90], an approach was developed to identify different types of sophisticated coalition attacks (e.g., a coalition formed of multiple dishonest publishers) called the Similarity-Seeker algorithm. This detection mechanism relies on analyzing traffic to find similarities in the traffic to websites. Legitimate websites do not have similar traffic, and traffic from similar sets of IPs is therefore suspicious. The original model can discover coalition attacks of size two, and the extended model can find attacks by coalitions of arbitrary sizes. The exploitation of statistical traffic analysis gives more scalability than traditional technologies.

Another method presented by Metwally et al. in [124] called Single-pubLisher attack dEtection Using correlaTion Hunting (SLEUTH) addresses the problem of fraudulent traffic generated by a single publisher via several IPs. This approach focuses on discovering an association between the publisher and the IP address of a machine. However, SLEUTH is only an adequate solution for a botnet that utilizes a vast number of IPs, and assumes that the traffic features of non-fraudulent publishers and IPs are constant. This assumption is not applicable to online advertising systems, where trends are highly temporal.

AdSherlock [125] is another method in this category in client-side that is used for online click fraud detection, and which is based on an ad request tree model. In this type of method, users need to install additional programs on their devices, which is not highly applicable. ClickGuard [126] is an machine learning-based (ML) system for detecting click fraud attacks. This system uses a classifier that is trained based on the motion sensor signals in mobile devices, in order to separate benign clicks from fake ones. However, despite the high accuracy of this novel inference system, it is a client-side application.

Although these solutions have certain benefits, all of them are under the threaten of complicated botnet ad fraud [40]. Many compromised machines are used to modify the IPs and cookie IDs of fraudulent requests.

In [127], the authors described the use of bluff advertisements, an online click-fraud detection strategy that blacklists malicious publishers based on a predefined threshold. This approach was designed to display several unrelated/fake advertisements amongst the user's targeted advertisements, with the expectation that these advertisements will not be clicked on. In addition to monitoring IPs and applying profile-matching and threshold detection techniques, bluff advertisements can create some obstacles for botnet owners who want to train their software. Negative attitudes of users can also be reduced by decreasing the number of precisely targeted advertisements. These considerations motivated the authors of [128] to recommend a technique for advertisers to count the proportion of invalid clicks on their advertisements by generating fake ones. Running bluff advertisements leads to an increase in advertising budgets for advertisers.

A hybrid ML method called Cascaded Forest and XGBoost (CFXGB) proposed in [129] to identify faulty clicks. The proposed model combines two learning models for feature transformation by Cascaded Forests and classification by XGBoost. They have shown that this combination leads to better results compared to using only cascaded forests as classifiers. Despite the high accuracy of the model, it is necessary to adjust the parameters.

All of the above detection methods can only address fraud after it has occurred. The authors of [130] therefore proposed a new automated method for preventing click fraud called clickable Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHAs). In the proposed method, customers complete a simple Turing test [131] and are then diverted to the publisher's site. Although click fraud can be identified based on valid users, the loading of CAPTCHAs requires time and space.

The techniques listed above are related to finding click fraud in web content. There are some other approaches to examine the detection and control of ad fraud in mobile applications. Based on the similarities that we mentioned in Section III-D regarding the web and mobile ad infrastructure, they face similar threats as click fraud [132]. However, despite the similarities, the threats in these systems are in different forms.

Ad threats behave differently on the two platforms because threatening features such as click fraud and malvertising in mobile platforms are different from web ads. For example, the mobile attacker can automatically generate fake ad clicks, while web attackers have to launch click fraud with bots. Based on these observations, the following are some of the methods aimed at detecting mobile click fraud.

The authors of [133] proposed a hidden Markov-based automated classification algorithm to detect fraud impressions in mobile advertising. The algorithm is called the hidden Markov scoring model (HMSM), which is based on the hidden Markov model (HMM) scoring approach instead of a conventional HMM probabilistic model. Although having a large data set in ML-based techniques helps achieve better accuracy, they claimed that the optimal size for training is 5000. Optimal selection of training sample size leads to reduced computation time in training and storage space.

MAdFraud [2] is dynamic testing frameworks that have focused on detecting click fraud on Android by analyzing network traffic. MAdFraud is designed to detect fake URL requests without valid user clicks. They first identify the unique features of mobile that leads to fraud. On the Android platform, at any time, an app runs in the foreground, where the app has a UI. Their first observation was that when an application receives ads in the background, it is probably fraud because the developer of the application receives the ad's credit without showing it to the user. The second one was that when an application clicks on an ad without user interaction, it is fake. They create a new way to detect ad impressions and clicks in three phases automatically: building HTTP request trees, identifying ad request pages using ML, and applying heuristics methods to detect HTTP request trees' clicks. Despite the efficiency of the tool in finding fraud apps, its observation is without user intervention, and it takes time to collect the app process.

MAdLife [134] is another analytics tool in this category that detects the full-screen ad impressions used without any user interaction. This study shows a correlation between click fraud and malvertising and shows 18.36% of malicious ads loaded with click fraud. This tool is developed to monitor the ad traffic generated throughout its lifetime in Android WebView. It first starts by logging the pre-click data in a database. After automatically performing ad clicks with lightweight UI automation, it stores the post-click traffic data in another table in the database. Finally, the tool compares tables to find equality to classify apps as malicious ones. This detection method only considers clicking on WebView and does not consider fraudulent activities that do not involve WebView.

The authors of [135] have created a mobile advertising click bot, called ClickDroid, that periodically clicks on mobile ads. It tries not to be detected by fraudulent mobile clicks on ad networks. To do this, it modifies the device's identifier every time it clicks on a mobile ad. ClickDroid evaluated 100-click fraud on eight popular advertising networks, and only two mobile ad networks detected traffic anomalies, indicating the ad network's inability to detect click fraud. The authors also proposed a method to detect click fraud for Android apps based on tracking the user's clicks from the touch sensor at the kernel level. To do this, they install a middleware framework to gather the sensor output and save it in a separate file. This report can then be used to verify the events generated by ClickDroid based on the difference between the human and software-generated clicks. Assuming that advanced bots cannot bypass the middleware, the system is still susceptible to click-farms.

C. Countermeasures to Inflight Modification of Ad Traffic

The authors of [15] proposed data integrity and authentication tools to ensure end-to-end security for communication to prevent inflight modification. However, the use of these mechanisms has certain disadvantages that make them challenging to deploy on a wide scale. Firstly, authentication tools, such as Transport Layer Security (TLS) protocol, depend on cryptographic processes that impose a high computational cost on servers. In other words, the high security of TLS due to the complexity of the cryptographic algorithms adopted by TLS leads to high computational and energy costs [143]. Secondly, since the authentication mechanism uses digital certificates to activate Web servers authentication, which are expensive since certificate authorities are required carry out authentication of web servers manually. Clearly, if a site has a certificate assigned by a trusted certification authority, a trusted connection can be made that helps browsers to authenticate websites [15].

Web administrators also prefer to use a customized self-signed certificate without relying on third-party certification authorities to avoid the extra cost; however, such self-signed certificates are vulnerable to MITM attacks, and do not provide a reliable solution that allows the web browser to identify the website, and users need to decide whether or not to trust the corresponding website [144]. From the user's point of view, it is complicated to determine the operation of a given certificate and to validate it. As a result, a malicious server can often communicate with users. A notary office can be established to control the consistency of the web server's public keys and to help the user verify self-signed certificates. Although this technique is a new and reliable solution, it has the same limitations as the scheme in [139].

To tackle the above problems, researchers have introduced several alternative approaches to protect Web content effectively [140]–[142]. For example, in [140], the authors present a new opportunistic encryption method for encrypting web communications, involving a secure channel without other host authentication. However, this technique is unable to protect systems against MITM attacks, since the attacker can easily access the certificates used for authentication and replace them to impersonate web page. In other work, the authors of [141] adopted a web-based measurement tool called Web Tripwire to detect inflight changes to websites. This method can inject JavaScript code into the site and monitor the HTTP web page to identify any changes in it. The tool immediately reports any modifications to the web page to both the end-user and the web server. Tripwire is a cheaper tool than HTTPS, which checks the integrity of pages, but is a non-cryptographically secure method. In [142], a secure scheme based on a collaboration

Table III: Comparison on existing detection methods in online advertising system. Camp. := Campaign, Ref. := Reference, Badver. := Badvertising.

Attack	Countermeasure	Advantage	Disadvantage	Ref.
Hacking Camp. Account	Daily checking of the user's account	✓ Protecting from possible financial and reputation losses	✗ Highly time-consuming	[12]
Crowd Fraud	Detection strategies based on human interactions	✓ Strong detection scheme	✗ Labor costs ✗ Becoming invalid quickly due to rapid change in web workers' behavior	[68]
	Substantial randomness solution based on the group behaviors	✓ Robustness, scalable, and reliable ✓ No need to tune parameters manually ✓ Applicable in real-world	✗ Fails in preventing fraud ✗ Difficulty in evaluating the accuracy of the algorithm	[68]
	Detection method based on clustering analysis	✓ Good convergence and extensibility	✗ High complexity	[98]
	CFC	✓ Offer crowdsourcing fraud tool to protect both ad networks and advertisers in mobile platform	✗ Requires the advertisers and ad networks to trust the CFC party	[99]
Badver.	Detecting and preventing badvertisment via active and passive schemes	✓ Preserving user privacy	✗ Needs third-party interaction ✗ Time-consuming	[69]
Hit Shaving	Collecting information	✓ Strong detection scheme	✗ Lack of scalability ✗ Lack of efficiency	[104]
	Using cryptography and probability tools to detect fraud	✓ User-friendly and simple model ✓ No need third party ✓ Constant ad's communications, computation, and storage cost	✗ Need to tune parameters manually	[73]
	Enabling the referrer webmasters to monitor the number of legal clicks	✓ No need awareness or cooperation by the webmasters	✗ Communication overhead	[89]
	Enabling the providers of click-through mechanisms to control the number of clicks	✓ Robust ✓ No need to honest webmaster	✗ Cooperation or awareness by the webmaster	[89]
Hit Inflation	**Classical**: Using a variety of metrics to monitor the quality of the traffic to find fraud	✓ No need third party	✗ Lack of efficiency and scalability ✗ Conflict of interest between commissioners and advertisers	[74], [79], [52]
	Cryptographic: Changing the industry model based on pay-per-sale	✓ Safe ✓ Robust	✗ Vulnerable to hit shaving	[107]
	Cryptographic: Changing the pay-per-click model with the pay-per-impression/ pay-per-action model	✓ Guarantees the legitimacy of the receiving clicks by advertisers through a trusted third party	✗ Sharing the information between the third parties	[108], [109]
	Cryptographic: The assistance of the users to identify fraudulent traffic from regular traffic	✓ Cost saving by free service	✗ Lack of scalability and user privacy ✗ Sharing the information between the third parties	[89], [110]
	Data Analysis: Detecting duplicate clicks: • Original Bloom Filter algorithm • GBF algorithm and TBF algorithm • Clicktok • Deep Learning-based Model • SeqGAN • GFD • CSBPNN-ABC	✓ Less error rate ✓ Requires simpler operations and less storage space/ Low false-positive rate ✓ Low latency ✓ High accuracy ✓ Quick classifier ✓ No labour-intensive feature engineering ✓ High accuracy	✗ Memory waste ✗ Theoretical analysis was made ✗ Work on the user side ✗ Low performance ✗ - ✗ Using one dataset for seven days leads to less robustness and accuracy ✗ Low performance	[115], [72], [116], [117], [118], [120], [121]

Table IV: Continued from Table III.

Attack	Countermeasure	Advantage	Disadvantage	Ref.
Hit Inflation	**Data Analysis**: Fabricated impressions and clicks: • Streaming-Rules algorithm • Similarity-seeker algorithm • SLEUTH • AdSherlock • ClickGuard • Bluff Ads • CFXGB • CAPTCHAs • HMSM • MAdFraud • MAdLife • ClickDroid	✓ Scalability and ability to detect specific hit inflation ✓ Highly scalable ✓ High accuracy & ability to detect complex coalition attacks ✓ High accuracy & Lower overhead ✓ High accuracy ✓ Put some obstacles against the botnet's owner to train their software ✓ Reducing computation time & storage space ✓ High accuracy ✓ Identifying click fraud based on the valid user ✓ High efficiency ✓ Dynamic data analytics approach ✓ No specialized hardware requiredy	✗ Thwarted by sophisticated botnet ad fraud ✗ Under the threaten of complicated botnet ad fraud ✗ Not applicable to online advertising systems ✗ Work on the user side ✗ Work on the user side ✗ Increasing advertisers' budget on advertisements ✗ Less accuracy ✗ Needs to tune parameters ✗ Loading CAPTCHAs needs time and space ✗ Without user intervention & Time-consuming ✗ Lack of trustability & Applicable only for Android WebView ✗ Susceptible to click-farms attack	[122], [90], [124], [125], [126], [127], [129], [130], [133], [2], [134], [135]
Inflight Modification of Ad Traffic	Data integrity and authentication mechanisms	✓ Ensure the end-to-end security of communications to prevent inflight modifications	✗ Lack of scalability ✗ Highly communication cost	[139]
	Using a new encryption method to encrypt Web communications without other host authentication	✓ Highly scalable	✗ Fail to protect the system against MITM attacks	[140]
	Using Web Tripwire to detect inflight changes to websites	✓ A cheaper tool than HTTPS	✗ Non-cryptographically secure method	[141]
	Secure scheme based on the collaboration between ad networks and web servers	✓ Ensure authenticity and integrity of the traffic	✗ Additional charge for publishers and ad networks	[142]
Malvertising	Checking the advertisements regularly and validate their appropriateness by publishers or ad networks	✓ Prevent losses of reputation, traffic, and revenue	✗ Highly time-consuming	[15]
	Install/update anti-malware software by users	✓ Preventing to install malware on the user's machine	✗ Use up a lot of memory & disk space and slowing down the system	[15]
	Applying the game-theoretic approach to formulating the malvertising problem	✓ Data analytics approach	✗ Theoretical analysis was made ✗ Not applicable in real-world ✗ Advertisers' benefits only	[136]
	Detection malvertising by ML	✓ Detecting malvertising in web-based system	✗ Unable to detect unknown attacks	[14]
	UI-based methodology to detect malware	✓ Dynamic data analytics approach	✗ Not applicable to find malware generating by ad network	[137]
	MadDroid	✓ Dynamic data analytics approach	✗ Lack of trustability ✗ Applicable only for mobile	[138]

between ad networks and web servers was introduced to counteract inflight traffic modification. This method is based on the fact that ad networks with digital authentication certificates can ensure the authenticity and integrity of the traffic. However, the implementation of this method imposes a high cost on publishers and ad networks.

D. Countermeasures to Malvertising

To avoid malvertising, the authors of [15] suggest checking the advertisements regularly and validating their appropriateness. It is the responsibility of the publishers and ad networks to verify the advertising content (whether active or malicious) by performing regular checks. They should avoid publishing advertisements to end-users if publishers and ad networks become aware of any unexpected or unwanted behavior in the code, such as automated redirections. For example, in June 2009, Google launched an investigative research engine to help ad networks by regularly checking the source code of websites. This search engine is publicly available at www.anti-malvertising.com, and enables ad networks to detect potential malvertising providers. Surfers also need to update/install anti-malware programs on their systems to protect against such risks.

The game-theoretic approach [136] has been applied to formulate the problem of malvertising, and a mitigation strategy was developed based on data analytics that introduced two Bayesian games between the first player, ad network (defender), and second player, the malvertiser (attacker). The model is not applicable in real-world experiments and has advantages only for advertisers. In [14], the authors presented a ML-based method for detecting malvertising at publishers' end. They used 15,000 ads and extracted nine features to train Support Vector Machine (SVM) classification. Their results show that 53% of suspicious ads contain suspicious iFrames. Because the learning-based detection method relies on known attacks patterns, they may not detect unknown attacks.

The authors in [137] used the UI-based methodology to detect malware on the mobile operating system. They claimed that it is essential to find the source of the attack in order to detect it. They pointed out that even the main applications may not be malicious, but the web destinations that the user visits can play an important role in spreading the attacks. Thus, they have found three features to have a successful dynamic analytics tool, including triggering the app-web interfaces, detecting malicious content, and identifying the source of an attack. They performed a simulation for two months in two countries to find malware launched through advertisements and web links in applications. Finding the source of the attack helped them detect the responsible entity in the system.

MadDroid [138] is a dynamic data analytics framework that has focused on identifying malvertising in mobile environments. The method is based on the characterization of devious ad content that leads to lunch malvertising. The ad content downloaded at runtime from trusted ad networks could serve as a channel for attackers to distribute undesirable contents or even malware. Besides the ad content itself, some unwanted payload may be triggered when the user interacts with the ad content. MadDroid records any network traffic and collects content exchanged between mobile advertisers, ad networks, and user devices. The HTTP hook approach is used to build a mapping between ad libraries and ad hosts. This mapping helps the tool to identify ad traffic from all recorded traffic accurately. Applying MadDroid to 40,000 Android apps, they found that roughly 6% of apps deliver devious ad contents. Despite the efficiency of the method in finding malware from an ad content perspective, it lacks trustability since the tool and dataset are not publicly available.

VI. Lessons Learned and Future Research Directions

Following the comprehensive survey presented above of the security aspects of online advertising and related techniques, we now summarize the lessons that can be learned and describe possible future challenges and research directions. Some of these have already discussed in previous sections; however, several further challenges and open research issues are dealt with in brief in this section.

A. Lessons Learned

In many countries, fraudsters have an economic incentive to engage in fraudulent activities and exploit online advertising systems, given the high-risk advertising revenue and the lack of advertising fraud laws. Given that most online services and applications are funded through online advertising revenue interfering with the online advertising business model can have serious consequences. Therefore, there are incentives for stakeholders (e.g., ad networks) to protect advertising revenue and online advertising security. Thus, this work focuses on security issues, which are a crucial element of online advertising.

Despite the rapid growth of online advertising in modern communication networks, there is no comprehensive taxonomy or research documentation to summarize the types of fraud in advertising systems. This paper provided a comprehensive review of fraud activities in online ad systems using a tiered taxonomy to summarize ad fraud at different levels and perspectives. Our paper provides direct answers to key questions such as the main types of fraud in advertising systems, fundamental approaches and features of different types of fraud, and the practical techniques used to detect ad frauds. We provide a detailed description of existing online advertising systems (see Section III) and systems vulnerabilities along with a taxonomy of current attack methods (see Section IV). We have explained who does what and how fraudsters can exploit these vulnerabilities to carry out ad fraud attacks, which we generally divide into three main categories: placement fraud, traffic fraud, and action fraud. For each type of attack, we provide techniques that fraudsters use to profit from ad systems. We discussed the challenges of detecting and reducing advertising fraud and several established countermeasures (see Section V). Table III shows how the various security threats covered in Section IV can be addressed with the available solutions along with the advantages and disadvantages of each.

In particular, we offered various defences to combat click fraud, including the classical approach, the cryptographic

approach, and the data analysis approach. We examined the issue of classical criteria, which is related to the lack of efficiency, scalability and maturity. Then, to solve the problems of the classical solutions, we studied different cryptographic methods. We explained that the main idea of these solutions is to change the industry standard to give fraudulent publishers less opportunity to commit fraud. However, the proposed methods suffer from a lack of user privacy due to the sharing of information between third parties and lack of scalability because multiple hash functions are required. Hence, various data analysis techniques have been developed to reduce the problems caused by cryptographic methods. The primary purpose of these technologies is to find specific patterns that identify fake traffic. We have highlighted that the best way to deal with the conflict between the user's privacy and security is to perform statistical analysis on the collected data (such as cookie IDs and IPs) with the help of temporary user identification. The analysis of IPs and cookie IDs is more privacy-friendly than cryptographic methods. Commissioners can track users based on their cookie IDs and IPs. In the current Internet architecture, the use of cookies and IPs to detect fraud can be a less intrusive technique than methods requiring user login. Despite all the defence mechanisms, much research is still needed to design stronger countermeasures and protect the state of e-commerce. It is mainly because the online advertising system has an open platform and needs real-time trading.

The main conclusion stemming from the results reported in this paper is that fraud detection in online transactions is a dynamic field of research, as fraudsters continually invent new techniques for performing fraudulent transactions which seem to be genuine but which cause losses to businesses. Hence, it can be concluded that the real-time detection of ad fraud is the most critical step in making the whole process more efficient. It should also be mentioned that from an economic perspective, search engines need to find as many fraudulent clicks as possible, in order to maximize the ROI for the advertisers. Many efforts are currently being made to mitigate online advertising fraud, since the online advertising market is in a state of expansion and is working to support the needs of advertisers and online commerce providers. Successful fraud management will provide a competitive advantage to ad networks, and will enable them to provide the highest ROI possible to advertisers.

Due to the various limitations on previous investigations and properties of the current online advertising system, we introduce some possible future research directions towards building a reliable, secure, and efficient online advertising ecosystem in Section VI-B. In Section VI-C, we describe some possible solutions to mitigate each open issue.

B. Future Direction

Fig. 13 shows an overview of four open issues and the corresponding possible solutions. The security, reliability, and efficiency of online advertising systems rely on four major aspects of research, as described below.

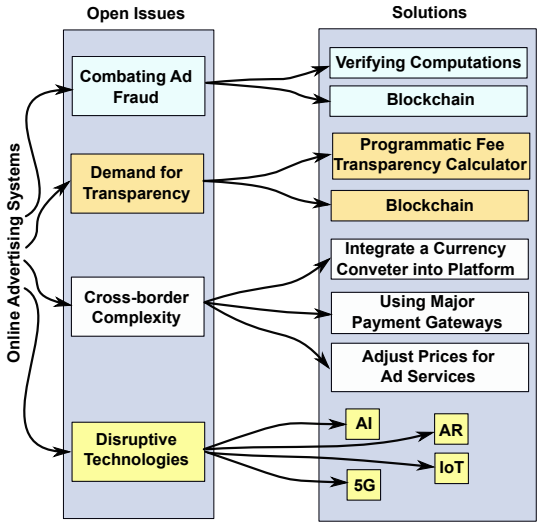

Figure 13: Proposed research roadmap for measuring and optimizing the security of online advertising networks.

1) Combating ad fraud: Although 2021 is expected to be a year of growth, this can be subverted by ad fraud. A report released by Juniper Research states that in 2018, about $42 billion was lost to ad fraud in the online advertising business [145]. It is expected that this amount will grow to $100 billion by 2023 [145]. The damages do not simply involve financial loss, and can affect user privacy and hide the best performing marketing channels. To deal with these damages, growth marketers must consider fraud prevention as a priority. The report claims that attackers tend to apply methods such as domain spoofing to increase the number of clicks by misrepresenting a low-quality site to resemble a high-quality website, rather than using techniques such as app install farms. As a result, it is essential to detect which ad clicks are fake and which are genuine, not an easy task in real-time bidding.

2) Demand for transparency: The report in [146] points out that the majority of the cost allocated to online advertising currently goes directly to waste, due to fraud or off-target audiences. However, there are ways to adapt, and transparency can play a significant role in this. For the entities that are involved in the ad industry, it is vital to know where their banners are served and where their budgets are spent, since if control over the budget allocated to the ad campaign is lost, advertisers will not know what has been spent where. Advertisers and publishers are doing business, and their activities therefore aim to make money, but the fragmentation of this economy means that media customers spend more high-priced than it's worth.

3) Cross-border complexity: This aspect aims to attract and protect global users who require multi-currency pricing options. For example, customers from all parts of the world trust ad providers to give them ad services. However, the payment methods by ad providers are not acceptable. As a result, to gain customer loyalty, ad service providers need to allow them to change money on their side at suitable exchange rates [147]. In this way, they can build a sustainable and secure

platform to execute different multi-currency scenarios.

4) Disruptive technologies: The online advertising industry has been significantly penetrated by technological innovations like the Internet of Things (IoT) [148], artificial intelligence (AI) [149], augmented reality (AR) [150], and 5th generation mobile Internet (or 5G) [151]. In 2018, for example, Google launched a beta experiment involving automatic ad placement on the basis of AI, and publishers' incomes increased by 10% [152]. To gain a competitive advantage in the market to survive, an enterprise needs to adapt to these changes faster than others, and the future of companies who are not ready for the newest technologies is in question.

C. Suggestion of Security Responses

In this section, we propose some responses to the challenges introduced in Section VI-B.

1) Responses for ad fraud: Ad fraud has become a significant concern for everyone involved in the ad industry, and can lead to reductions in trustworthiness and campaign effectiveness, and the siphoning of budgets.

The industry's primary solution for combating all types of fraud is the use of ML to analyze the history of attacks and how they appeared, to help companies predict what will happen next [153]. One of the best and most efficient solutions to prevent ad fraud is to apply sophisticated click validation mechanisms. This increases the workload for fraudsters aiming to steal advertisers' and brands' budgets, and makes it uneconomical for them. In 2019, Adjust [154] proposed a standard based on click validation in which ad channels send impressions with a unique identifier before the click claim is sent.

As mentioned previously, whenever users click on a hyperlink in a publisher's website, the advertiser must pay a fee. The question therefore arises as to how an advertiser can verify that the bill received from the publisher is correct. This poses a challenge and remains an open issue. In this case, our suggestion is to apply *Verifying Computations* without requiring the user to re-execute [155] them. The fundamental theorem behind this is a probabilistic proof system, which is composed of two elements, a prover and a verifier. The prover aims to prove a mathematical assertion (so-called proof) for the verifier, while the verifier checks the proof.

However, in practice, this computational technique is not economically sound. We, therefore, propose the use of a blockchain-based scheme for validation and verification. Using blockchain-based solutions to tackle ad fraud also studied by Kshetri et al. [156]. The concept underlying the blockchain is Distributed Ledger Technology (DLT), which helps various untrusting and distributed agents to transmit data in a trusted, secure, and valid way by providing distributed validation, transparency, and cryptographic immutability. DLT is a technology that starts with the cryptocurrency of bitcoin, followed by smart contracts. The properties of DLTs, such as consensus protocols, provenance and immutability, are used in the model for secure transactions [157]. The consensus protocol is that a network must approve all transactions of all players. Validation algorithms vary from system to system, but the bottom line is that the system only accepts transactions that are made under valid rules. Provenance means knowing the source of the data. Immutability can be defined as the ability to keep data recorded in the DLT unchanged. The problem of trust between entities that do not trust each other can be solved with these three DLT features. Moreover, this is very well in line with the lack of trust in the online advertising ecosystem. The pervasiveness of ad fraud is due to the lack of an intermediary that can track online advertising to increase trust and reduce some concerns. Blockchain transparency helps detect fraudulent traffic and improves ad delivery. With the ability to import transparency to the system and track assets online, blockchain can help the advertising system to reduce ad fraud, if not stop it completely. It is possible to know who did what and when. Recently, a wide range of applications (such as healthcare [158] and genomics [159]) have begun to use the blockchain to guarantee trustworthiness in interactions among untrusting agents. Thus, the blockchain is an appropriate mechanism to ensure trust in cases that require long-running computations. We believe that an important future research direction in using validation clicks to fight ad fraud could be to investigate how blockchain-based validation can be extended and used to ensure effective, trusted verifiable computations.

2) Responses for transparency demand: High levels of transparency play a significant role in building trust between entities in the online advertising system and customers [160], [161]. This also affects the relationship between the publisher and advertiser. One way to help bring transparency over cost is to create a real-time analytic method to follow all activities. In the following, we highlight some other technologies and tools that can improve and guarantee transparency.

- In 2016, IAB released a *Programmatic Fee Transparency Calculator* to add transparency to the collaboration between publisher and advertiser [162]. This tool was designed to help actors in the online advertising market to define and apply cost models differently. In this way, they have the flexibility to enter their planning rates and budgets into the calculator, and then select the available advertising technologies for the campaign. It is essential to mention that the calculation cost model is based on the "% of media."
- The blockchain can provide security and transparency for the transfer of data from advertiser to publisher [163]. It is also possible to do real-time transactions by exploiting blockchain technology, especially when the price is obvious to all participating members of the supply chain. Blockchain transparency can improve the ad delivery process. Advertisers can combine ad data with data provided by ad viewers to increase the effectiveness of their ads. Blockchain has the potential to change the way online ads are paid, sold and measured. It allows advertisers to see if their ads are being delivered or reaching the right customers. Advertisers can track who opened the ad and where is the customer location. A few advertising tools are available to cope with the transparency challenge, including Havas [164] and Apomaya. These platforms aim to support transparency by calculating the fees that media buyers must pay.

3) Responses for cross-border complexity: Engineers have expertise in developing ad software that facilitates multiple-currency and cross-border operations. They are aware of how to create and maintain smart billing services that can support multi-currency payments. We identify some other techniques for coping with the challenges of cross-border complexity as follows.

- One technique is to integrate a currency converter calculator into a pre-built framework. This requires finding an Application Programming Interface (API), such as currencylayer, Fixer, or XE Currency Data, to allow regular updating of currency exchange rates and access to the maximum number of worldwide currencies.
- Another technique is to provide customers with access to different payment gateways, including PayPal, SecurePay, Stripe, Authorize.Net, etc. Offering diverse payment options can help to attract and retain loyal customers [165]. The new digital currency, called Bitcoin, as a peer-to-peer electronic cash system can also help the payment process. This cryptocurrency mechanism can not only solve the double-spending problem [166], but it sets out a new paradigm for performing transactions and exchanging value in an online environment. The interactive and universal features of bitcoin allow marketers to bypass intermediaries, transmit their business content, and reduce costs. For example, retailers typically pay 3% of payment processing to credit card companies, and many online platforms receive sales commissions. Blockchain technology can help brands limit or remove costs and eliminate worthless activities in the middle layer. Thus, the technology can potentially extend the direct relationship between brands and consumers.
- Ad services can also be provided with adjustable prices by considering the average transaction cost across a specific country, since a given amount might be adequate for one country but too high for another.

4) Responses for Disruptive technologies: It is not an easy task to apply cutting-edge technologies when the traditional types work well. For example, it is difficult for an advertiser to change their ad campaigns to the emerging ones. However, in this new era, there is a need to adapt and be aware of the latest technologies, and the domain of online advertising systems is no exception. Emerging technologies such as AI, AR, IoT, and 5G can help ad tech companies in several ways [167]. For example, the role of AI is three-fold. Firstly, the use of AI-based chatbot applications will motivate users to buy products, since a chatbot allows them to ask questions, give commands and receive services in a conversational style [168]. An AI chatbot can read data, analyze complex information and make decisions based on this information. Depending on the customer's question, the system should refer them to a specific social group to demonstrate the items that can be purchased. Secondly, AI provides a method of targeted advertising. Assisted by the application of ML algorithms to big data, AI can automatically sort marketing messages and deliver them to the target users, making ad targeting more accurate and cost-effective [169]. Thirdly, running AI-based algorithms allows ad mediation to be optimized to maximize profits for publishers by finding the best-matched slots for their advertisements. AI-based advertising helps companies in four ways: (i) by displaying personalized advertisements to the relevant customers and minimizing human effort; (ii) by interacting with audiences in a natural way; (iii) by reducing errors using a data-oriented approach for network selection; and (iv) by saving time through automating the process of ad publishing.

Although AR-based marketing is in its infancy, it has become interesting to marketers. Since everyone has a smartphone, advertisement based on AR is now much easier than before. For example, stores can install AR-driven ad applications to send customers popup advertisements to tell them about products, and consequently attract customers to purchase items. AR-based advertising helps companies in three ways: (i) by providing targeted and innovative contextual advertising; (ii) by improving customer experience and making it unique and immersive; and (iii) by boosting customer loyalty through interactive advertising.

A report from the IAB found that around 65% of people in the US own at least one IoT device, and are interested in receiving advertisements on IoT screens [170]. IoT technologies can therefore provide new levels of ad targeting [148]. IoT data can be used to dig even deeper into customers' habits, interests, preferences, and other factors, and allows advertisers to learn more about their customers to create customer personas and targeted ad campaigns. It is also possible to integrate cloud solutions with various gateways to achieve better results in the ad campaign. IoT-based ad software can help advertising companies in three ways: (i) better recognition and prediction of consumers' individual preferences and needs to increase the efficiency and accuracy of target advertisements; (ii) increased user engagement and satisfaction by providing them with valuable information about products; and (iii) improved ad campaign effectiveness.

The arrival of 5G promises to open up substantial new opportunities to advertisers and customers. The possibility of achieving Internet speeds 20 times faster than 4G is an enticing one for both of them. The faster network speeds and high-resolution screens through 5G will allow video resolution to be increased and page loading times to be reduced, creating more interaction between customers and advertisers. These features lead to more customer engagement for e-commerce activities, more time spent on e-commerce websites and more online shopping [171]. Needless to say, to fully exploit the potential of 5G in the advertising industry, all the entities in the industry should prepare themselves before launching 5G. In the following, we identify some of the issues that should be considered.

- **Faster load speeds.** Despite the advent of new technologies, like AI, AR, and 3D modeling, which have revolutionized the ad market, advertisers may not be attracted to online advertising due to issues relating to speed. With the high speeds of 5G, a new era will open up for advertisers to exploit customer profiling, ad creative, targeting, and many more aspects. 5G will increase the speed of a device from 45 Mbps up to a maximum of one gigabit, meaning that

response times will be a few milliseconds, thus leading to a decrease in latency [172]. This can provide a better space for the use of streaming video (or even deeper augmented and virtual reality) to create advertisements [173]. StateFarm reported a 500% increase in mobile ad click-through rates due to its virtual reality ads, indicating the potential of virtual reality ads to grab the audience's attention [174]. It also opens the way for creating video advertisements, giving customers the chance to stop scrolling the web page to watch high-resolution advertisements.

- **Reach across channels.** The efficiency of the online advertising ecosystem depends on the communication technology infrastructure [175]. Speed and access to Internet connections affect the advertising industry's ability to reach consumers. With faster speeds everywhere, more devices will be connected and participate in the online advertising world. Hence, with the introduction of 5G networks, consumers will eventually experience much faster wireless connectivity. 5G can not only help with ad creation but will also help advertisers reach audiences more effectively [176]. The targeting of audiences with low-speed Internet was not straightforward, but 5G will pave the way for creating a range of channels that will enable advertisers to connect directly with consumers.
- **Unlocking identity.** It is worth pointing out that advertisers with more digital touchpoints are more likely to be selected by customers, and should consider this a chance to learn more information about their audiences. Not surprisingly, the most significant impact of 5G will be on the quality and quantity of data in the system, which will allow the advertiser to target and capture the correct audience more effectively [171]. To achieve maximum benefit from the new data, it is vital to make sure all the basis includes the right technology partners, the right infrastructure and the right kind of privacy measures for the use of the data. Moreover, in the run-up to the introduction of 5G, marketers should warn stakeholders to consider the principles of privacy by design when using this valuable data.

The disruptive technologies that have been developed over the last decade promise new solutions and new ways to connect with customers and markets; however, as with any opportunity for growth, these developments are not without risks, and businesses should start considering these now. For example, predictions made by Ericsson show that 29 billion devices will be connected to the IoT by 2022 [177]. This growth in the number of IoT connections will lead to increased security threats. Hence, markets and businesses need to ensure that their IoT-connected devices are safe, that default passwords are not used and that all security updates are installed.

VII. Conclusion

Online advertising is vital in sustaining the economy of the Internet, since each party in the system can gain profit. However, in many countries, there is a lack of legal protection against ad fraud, and given the amount of ad revenue at stake, online advertising has become a target for criminals to gain financial incentives through fraudulent activities.

In this article, we have investigated and discussed the security aspects of the online advertising market. We first gave a brief introduction to the online advertising system, followed by the fundamental concepts that have emerged in relation to the online advertising system. Next, we presented a state-of-the-art study of the various forms of security attacks on the system that arise from the weaknesses of the ecosystem. We then proposed a comprehensive taxonomy of ad fraud to describe these threats in global terms and facilitate cooperation among research hyper refers to deal with ad fraud attacks. We classified the existing solutions that have been proposed in the literature to cope with these attacks, along with the limitations and effectiveness of these solutions. Finally, we presented our view of current research challenges and future directions to improve existing security solutions in the online advertising system.

References

[1] N. Vratonjic, J. Freudiger, M. Felegyhazi, and J.-P. Hubaux, "Securing online advertising," Tech. Rep., 2008.

[2] J. Crussell, R. Stevens, and H. Chen, "Madfraud: Investigating ad fraud in android applications," in *Proceedings of the 12th annual international conference on Mobile systems, applications, and services*. ACM, 2014, pp. 123–134.

[3] L. Song, X. Gong, X. He, R. Zhang, and A. Zhou, "Multi-stage malicious click detection on large scale web advertising data." in *BD3@ VLDB*. Citeseer, 2013, pp. 67–72.

[4] P. Gill, V. Erramilli, A. Chaintreau, B. Krishnamurthy, K. Papagiannaki, and P. Rodriguez, "Best paper–follow the money: understanding economics of online aggregation and advertising," in *Proceedings of the 2013 conference on Internet measurement conference*. ACM, 2013, pp. 141–148.

[5] "Forbes," https://www.forbes.com/sites, 2020, [Online; accessed 16-May-2021].

[6] T. Yao, Q. Li, S. Liang, and Y. Zhu, "Botspot: A hybrid learning framework to uncover bot install fraud in mobile advertising," in *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 2020, pp. 2901–2908.

[7] I. Ullah, R. Boreli, and S. S. Kanhere, "Privacy in targeted advertising: A survey," *arXiv preprint arXiv:2009.06861*, 2020.

[8] "Mobile advertising spending worldwide from 2007 to 2022," https://www.statista.com/statistics/303817/mobile-internet-advertising-revenue-worldwide/, 2021, [Online; accessed 16-May-2021].

[9] V. Yurovskiy, "Pros and cons of internet marketing," *Research Paper], Turiba University Faculty of Business Administration (Latvia). Retrieved from http://www. turiba. lv/f/StudZinKonf_Yurovskiy. pdf*, 2015.

[10] H. Haddadi, P. Hui, T. Henderson, and I. Brown, "Targeted advertising on the handset: Privacy and security challenges," in *Pervasive Advertising*. Springer, 2011, pp. 119–137.

[11] "recode," https://www.vox.com/2017/9/14/16294450/mobile-ad-spending-growth-worldwide, 2020, [Online; accessed 16-May-2021].

[12] A. Mladenow, N. M. Novak, and C. Strauss, "Online ad-fraud in search engine advertising campaigns," in *Information and Communication Technology*. Springer, 2015, pp. 109–118.

[13] J. Linden and T. Teeter, "Method for performing real-time click fraud detection, prevention and reporting for online advertising," Nov. 27 2012, uS Patent 8,321,269.

[14] P. Poornachandran, N. Balagopal, S. Pal, A. Ashok, P. Sankar, and M. R. Krishnan, "Demalvertising: a kernel approach for detecting malwares in advertising networks," in *Proceedings of the First International Conference on Intelligent Computing and Communication*. Springer, 2017, pp. 215–224.

[15] N. Vratonjic, M. Manshaei, and J.-P. Hubaux, "Online advertising fraud," Tech. Rep., 2011.

[16] B. Edelman, M. Ostrovsky, and M. Schwarz, "Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords," *American economic review*, vol. 97, no. 1, pp. 242–259, 2007.

[17] D. S. Evans, "The economics of the online advertising industry," *Review of network economics*, vol. 7, no. 3, 2008.

[18] C. E. Tucker, "The economics of advertising and privacy," *International journal of Industrial organization*, vol. 30, no. 3, pp. 326–329, 2012.

[19] J. Chen and J. Stallaert, "An economic analysis of online advertising using behavioral targeting," *Mis Quarterly*, vol. 38, no. 2, pp. 429–449, 2014.

[20] D. S. Evans, "The online advertising industry: Economics, evolution, and privacy," *Journal of economic perspectives*, vol. 23, no. 3, pp. 37–60, 2009.

[21] H. Choi, C. F. Mela, S. R. Balseiro, and A. Leary, "Online display advertising markets: A literature review and future directions," *Information Systems Research*, 2020.

[22] S. Bostanshirin, "Online marketing: challenges and opportunities," in *Proceedings of SOCIOINT14-International Conference on Social Sciences and Humanities, Istanbul, September*, 2014, pp. 8–10.

[23] G. Aggarwal, J. Feldman, S. Muthukrishnan, and M. Pál, "Sponsored search auctions with markovian users," in *International Workshop on Internet and Network Economics*. Springer, 2008, pp. 621–628.

[24] A. Goldfarb and C. Tucker, "Search engine advertising: Pricing ads to context," 2008.

[25] H. R. Varian, "Position auctions," *international Journal of industrial Organization*, vol. 25, no. 6, pp. 1163–1178, 2007.

[26] J. Estrada-Jiménez, J. Parra-Arnau, A. Rodríguez-Hoyos, and J. Forné, "Online advertising: Analysis of privacy threats and protection approaches," *Computer Communications*, vol. 100, pp. 32–51, 2017.

[27] G. Chen, J. H. Cox, A. S. Uluagac, and J. A. Copeland, "In-depth survey of digital advertising technologies," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 2124–2148, 2016.

[28] F. Dong, H. Wang, L. Li, Y. Guo, T. F. Bissyandé, T. Liu, G. Xu, and J. Klein, "Frauddroid: Automated ad fraud detection for android apps," in *Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, 2018, pp. 257–268.

[29] C. Budak, S. Goel, J. Rao, and G. Zervas, "Understanding emerging threats to online advertising," in *Proceedings of the 2016 ACM Conference on Economics and Computation*, 2016, pp. 561–578.

[30] R. Kayalvizhi, K. Khattar, and P. Mishra, "A survey on online click fraud execution and analysis," *International Journal of Applied Engineering Research*, vol. 13, no. 18, pp. 13 812–13 816, 2018.

[31] N. Gohil and A. D. Meniya, "A survey on online advertising and click fraud detection."

[32] N. Daswani, C. Mysen, V. Rao, S. Weis, K. Gharachorloo, and S. Ghosemajumder, "Online advertising fraud," *Crimeware: understanding new attacks and defenses*, vol. 40, no. 2, pp. 1–28, 2008.

[33] N. Laleh and M. A. Azgomi, "A taxonomy of frauds and fraud detection techniques," in *International Conference on Information Systems, Technology and Management*. Springer, 2009, pp. 256–267.

[34] N. G. andValerio Stallone, "The digital advertising ecosystem–status quo, challenges and trends."

[35] Y. Cai, G. O. Yee, Y. X. Gu, and C.-H. Lung, "Threats to online advertising and countermeasures: A technical survey," *Digital Threats: Research and Practice*, vol. 1, no. 2, pp. 1–27, 2020.

[36] T. Blizard and N. Livic, "Click-fraud monetizing malware: A survey and case study," in *2012 7th International Conference on Malicious and Unwanted Software*. IEEE, 2012, pp. 67–72.

[37] "Doubleclick," https://static.googleusercontent.com/media/\www.google.com/en//adexchange/AdExchangeOverview.pdf, 2020, [Online; accessed 16-May-2021].

[38] "Adecn," https://www.crunchbase.com/organization/adecn, 2020, [Online; accessed 16-May-2021].

[39] "Openx," http://www.openx.com/, 2020, [Online; accessed 16-May-2021].

[40] B. Stone-Gross, R. Stevens, A. Zarras, R. Kemmerer, C. Kruegel, and G. Vigna, "Understanding fraudulent activities in online ad exchanges," in *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*. ACM, 2011, pp. 279–294.

[41] J. Estrada-Jiménez, J. Parra-Arnau, A. Rodríguez-Hoyos, and J. Forné, "On the regulation of personal data distribution in online advertising platforms," *Engineering Applications of Artificial Intelligence*, vol. 82, pp. 13–29, 2019.

[42] D. Jiang, R. Xu, X. Xu, and Y. Xie, "Multi-view feature transfer for click-through rate prediction," *Information Sciences*, vol. 546, pp. 961–976, 2021.

[43] X. Zhao, C. Gu, H. Zhang, X. Liu, X. Yang, and J. Tang, "Deep reinforcement learning for online advertising in recommender systems," *arXiv preprint arXiv:1909.03602*, 2019.

[44] Z. Gharibshah and X. Zhu, "User response prediction in online advertising," *arXiv preprint arXiv:2101.02342*, 2021.

[45] C. Niu, M. Zhou, Z. Zheng, F. Wu, and G. Chen, "Era: Towards privacy preservation and verifiability for online ad exchanges," *Journal of Network and Computer Applications*, vol. 98, pp. 1–10, 2017.

[46] A. Ghosh, M. Mahdian, R. P. McAfee, and S. Vassilvitskii, "To match or not to match: Economics of cookie matching in online advertising," *ACM Transactions on Economics and Computation (TEAC)*, vol. 3, no. 2, pp. 1–18, 2015.

[47] L. Jin, B. He, G. Weng, H. Xu, Y. Chen, and G. Guo, "Madlens: Investigating into android in-app ad practice at api granularity," *IEEE Transactions on Mobile Computing*, 2019.

[48] Y. Imamura, R. Orito, H. Uekawa, K. Chaikaew, P. Leelaprute, M. Sato, and T. Yamauchi, "Web access monitoring mechanism via android webview for threat analysis," *International Journal of Information Security*, pp. 1–15, 2021.

[49] K. Zhang and Z. Katona, "Contextual advertising," *Marketing Science*, vol. 31, no. 6, pp. 980–994, 2012.

[50] J. M. Koran, C. Y. Chung, A. Gupta, G. H. John, H. Yin, L.-J. Lin, and R. Frankel, "Behavioral targeting system," Aug. 6 2013, uS Patent 8,504,575.

[51] R. E. Chatwin, "An overview of computational challenges in online advertising," in *2013 American Control Conference*. IEEE, 2013, pp. 5990–6007.

[52] A. Metwally, D. Agrawal, and A. El Abbadi, "Hide and seek: Detecting hit inflation fraud in streams of web advertising networks," Technical Report 2006-06, University of California, Santa Barbara, Department of Computer Science, Tech. Rep., 2006.

[53] S. Mittal, R. Gupta, M. Mohania, S. K. Gupta, M. Iwaihara, and T. Dillon, "Detecting frauds in online advertising systems," in *International Conference on Electronic Commerce and Web Technologies*. Springer, 2006, pp. 222–231.

[54] N. Vratonjic, "Security, privacy and economics of online advertising," EPFL, Tech. Rep., 2013.

[55] B. R. Gordon, K. Jerath, Z. Katona, S. Narayanan, J. Shin, and K. C. Wilbur, "Inefficiencies in digital advertising markets," *Journal of Marketing*, pp. 1–53, 2021.

[56] X. Zhu, H. Tao, Z. Wu, J. Cao, K. Kalish, and J. Kayne, *Fraud prevention in online digital advertising*. Springer, 2017.

[57] A. Frik, "Economics of privacy: Users' attitudes and economic impact of information privacy protection," Ph.D. dissertation, University of Trento, 2017.

[58] "Adwords," https://www.google.at/adwords/, 2020, [Online; accessed 16-May-2021].

[59] X. Li, M. Zhang, Y. Liu, S. Ma, Y. Jin, and L. Ru, "Search engine click spam detection based on bipartite graph propagation," in *Proceedings of the 7th ACM international conference on Web search and data mining*. ACM, 2014, pp. 93–102.

[60] R. A. Rodríguez-Gómez, G. Maciá-Fernández, and P. García-Teodoro, "Survey and taxonomy of botnet research through life-cycle," *ACM Computing Surveys (CSUR)*, vol. 45, no. 4, p. 45, 2013.

[61] N. Daswani and M. Stoppelman, "The anatomy of clickbot. a," in *Proceedings of the first conference on First Workshop on Hot Topics in Understanding Botnets*. USENIX Association, 2007, pp. 11–11.

[62] M. Kantardzic, C. Walgampaya, B. Wenerstrom, O. Lozitskiy, S. Higgins, and D. King, "Improving click fraud detection by real time data fusion," in *Signal Processing and Information Technology, 2008. ISSPIT 2008. IEEE International Symposium on*. IEEE, 2008, pp. 69–74.

[63] M. Alauthman, N. Aslam, M. Al-Kasassbeh, S. Khan, A. Al-Qerem, and K.-K. R. Choo, "An efficient reinforcement learning-based botnet detection approach," *Journal of Network and Computer Applications*, vol. 150, p. 102479, 2020.

[64] E. Kamar, S. Hacker, and E. Horvitz, "Combining human and machine intelligence in large-scale crowdsourcing," in *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*. International Foundation for Autonomous Agents and Multiagent Systems, 2012, pp. 467–474.

[65] H. Choi, K. Lee, and S. Webb, "Detecting malicious campaigns in crowdsourcing platforms," in *Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. IEEE Press, 2016, pp. 197–202.

[66] A. Doan, R. Ramakrishnan, and A. Y. Halevy, "Crowdsourcing systems on the world-wide web," *Communications of the ACM*, vol. 54, no. 4, pp. 86–96, 2011.

[67] F. Tahmasebian, L. Xiong, M. Sotoodeh, and V. Sunderam, "Crowdsourcing under data poisoning attacks: A comparative study," in *IFIP Annual Conference on Data and Applications Security and Privacy*. Springer, 2020, pp. 310–332.

[68] T. Tian, J. Zhu, F. Xia, X. Zhuang, and T. Zhang, "Crowd fraud detection in internet advertising," in *Proceedings of the 24th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2015, pp. 1100–1110.

[69] M. Gandhi, M. Jakobsson, and J. Ratkiewicz, "Badvertisements: Stealthy click-fraud with unwitting accessories," *Journal of Digital Forensic Practice*, vol. 1, no. 2, pp. 131–142, 2006.

[70] B. Miller, P. Pearce, C. Grier, C. Kreibich, and V. Paxson, "What's clicking what? techniques and innovations of today's clickbots," in *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*. Springer, 2011, pp. 164–183.

[71] T. N. Jagatic, N. A. Johnson, M. Jakobsson, and F. Menczer, "Social phishing," *Communications of the ACM*, vol. 50, no. 10, pp. 94–100, 2007.

[72] L. Zhang and Y. Guan, "Detecting click fraud in pay-per-click streams of online advertising networks," in *Distributed Computing Systems, 2008. ICDCS'08. The 28th International Conference on*. IEEE, 2008, pp. 77–84.

[73] X. Ding, "A hybrid method to detect deflation fraud in cost-per-action online advertising," in *International Conference on Applied Cryptography and Network Security*. Springer, 2010, pp. 545–562.

[74] A. Metwally, D. Agrawal, A. El Abbad, and Q. Zheng, "On hit inflation techniques and detection in streams of web advertising networks," in *Distributed Computing Systems, 2007. ICDCS'07. 27th International Conference on*. IEEE, 2007, pp. 52–52.

[75] V. Anupam, A. Mayer, K. Nissim, B. Pinkas, and M. K. Reiter, "On the security of pay-per-click and other web advertising schemes," *Computer Networks*, vol. 31, no. 11-16, pp. 1091–1100, 1999.

[76] M. Oger, I. Olmez, E. Inci, S. Kücükbay, and F. Emekci, "Privacy preserving secure online advertising," *Procedia-Social and Behavioral Sciences*, vol. 195, pp. 1840–1845, 2015.

[77] C. Kim, H. Miao, and K. Shim, "Catch: A detecting algorithm for coalition attacks of hit inflation in internet advertising," *Information Systems*, vol. 36, no. 8, pp. 1105–1123, 2011.

[78] A. J. Broder, "Data mining the internet and privacy," in *International Workshop on Web Usage Analysis and User Profiling*. Springer, 1999, pp. 56–73.

[79] D. V. Klein, "Defending against the wily surfer-web-based attacks and defenses." in *Workshop on Intrusion Detection and Network Monitoring*, 1999, pp. 81–92.

[80] G. Shaw, "Spyware & adware: the risks facing businesses," *Network security*, vol. 2003, no. 9, pp. 12–14, 2003.

[81] H. Johnston, "Cliques of a graph-variations on the bron-kerbosch algorithm," *International Journal of Computer & Information Sciences*, vol. 5, no. 3, pp. 209–238, 1976.

[82] R. Kannan, S. Vempala, and A. Vetta, "On clusterings: Good, bad and spectral," *Journal of the ACM (JACM)*, vol. 51, no. 3, pp. 497–515, 2004.

[83] A. Qamar, A. Karim, and V. Chang, "Mobile malware attacks: Review, taxonomy & future directions," *Future Generation Computer Systems*, vol. 97, pp. 887–909, 2019.

[84] A. Bermudez-Villalva, M. Musolesi, and G. Stringhini, "A measurement study on the advertisements displayed to web users coming from the regular web and from tor," in *2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. IEEE, 2020, pp. 494–499.

[85] C. Larsen, "Exploiting trust in advertising networks," 2010.

[86] S. Ford, M. Cova, C. Kruegel, and G. Vigna, "Analyzing and detecting malicious flash advertisements," in *2009 Annual Computer Security Applications Conference*. IEEE, 2009, pp. 363–372.

[87] "Botnet caught red handed stealing from google," https://www.theregister.co.uk/Archive/2009/10/09/, 2020, [Online; accessed 16-May-2021].

[88] "Network stealth router-based botnet," http://dronebl.org/blog/8, 2020, [Online; accessed 16-May-2021].

[89] M. K. Reiter, V. Anupam, and A. J. Mayer, "Detecting hit shaving in click-through payment schemes." in *USENIX Workshop on Electronic Commerce*, 1998.

[90] A. Metwally, D. Agrawal, and A. El Abbadi, "Detectives: detecting coalition hit inflation attacks in advertising networks streams," in *Proceedings of the 16th international conference on World Wide Web*. ACM, 2007, pp. 241–250.

[91] "Google company – our history in depth." http://www.google.com/about/company/history/, 2020, [Online; accessed 16-May-2021].

[92] "Lefty g balogh – digital marketing testing ground." http://www.leftygbalogh.com/2011/story-hacked-google-adwords-account/, 2020, [Online; accessed 16-May-2021].

[93] "Moz – blogs: Adwords hackers – what a nightmare." https://moz.com/ugc/adwordshackers-what-a-nightmare/, 2020, [Online; accessed 16-May-2021].

[94] "Google – official adwords community." https://www.de.adwords-community.com/t5/Grundlagen/Hilfe-Mein-Account-wurde-gehackt/td-p/44399/, 2020, [Online; accessed 16-May-2021].

[95] "Google– official blog– insights from googlers into our products, and technology." http://googleblog.blogspot.co.at/2008/04/how-to-avoid-getting-hooked.html/, 2020, [Online; accessed 16-May-2021].

[96] Q. Zhang and W. Feng, "Detecting coalition frauds in online-advertising," in *Mathematical and Computational Approaches in Advancing Modern Science and Engineering*. Springer, 2016, pp. 595–605.

[97] B. Kulis and M. I. Jordan, "Revisiting k-means: New algorithms via bayesian nonparametrics," *arXiv preprint arXiv:1111.0352*, 2011.

[98] X. Jiarui and L. Chen, "Detecting crowdsourcing click fraud in search advertising based on clustering analysis," in *2015 IEEE 12th Intl Conf on Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom)*. IEEE, 2015, pp. 894–900.

[99] R. Mouawi, I. H. Elhajj, A. Chehab, and A. Kayssi, "Crowdsourcing for click fraud detection," *EURASIP Journal on Information Security*, vol. 2019, no. 1, pp. 1–18, 2019.

[100] D. Tripathi, B. Nigam, and D. R. Edla, "A novel web fraud detection technique using association rule mining," *Procedia computer science*, vol. 115, pp. 274–281, 2017.

[101] "Adwatcher." http://www.adwatcher.com/, 2020, [Online; accessed 16-May-2021].

[102] "Clickprotector." http://www.clickprotector.com/, 2020, [Online; accessed 16-May-2021].

[103] "India's secret army of online ad "clickers."," https://timesofindia.indiatimes.com/business/india-business/Indias-secret-army-of-online-ad-clickers/articleshow/654822.cms?, 2020, [Online; accessed 16-May-2021].

[104] R. Johnson and J. Staddon, "Deflation-secure web metering," *International Journal of Information and Computer Security*, vol. 1, no. 1-2, pp. 39–63, 2007.

[105] T. Zeller Jr, "With each technology advance, a scourge," *The New York Times*, 2004.

[106] M. Naor and B. Pinkas, "Secure and efficient metering," in *International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 1998, pp. 576–590.

[107] M. Jakobsson, P. D. MacKenzie, and J. P. Stern, "Secure and lightweight advertising on the web," *Computer Networks*, vol. 31, no. 11-16, pp. 1101–1109, 1999.

[108] J. Goodman, "Pay-per-percentage of impressions: an advertising method that is highly robust to fraud," in *Workshop on Sponsored Search Auctions*. Citeseer, 2005.

[109] A. Juels, S. Stamm, and M. Jakobsson, "Combating click fraud via premium clicks." in *USENIX Security Symposium*, 2007, pp. 17–26.

[110] C. Blundo and S. Cimato, "Sawm: a tool for secure and authenticated web metering," in *Proceedings of the 14th international conference on Software engineering and knowledge engineering*. ACM, 2002, pp. 641–648.

[111] R. Khare and A. Rifkin, "Trust management on the world wide web," *Computer networks and ISDN Systems*, vol. 30, no. 1-7, pp. 651–653, 1998.

[112] S. Saroiu, S. D. Gribble, and H. M. Levy, "Measurement and analysis of spyware in a university environment." in *NSDI*, 2004, pp. 141–153.

[113] R. McGann, "Study: Consumers delete cookies at surprising rate," *ClickZ News*, 2005.

[114] A. Metwally, D. Agrawal, and A. El Abbadi, "Duplicate detection in click streams," in *Proceedings of the 14th international conference on World Wide Web*. ACM, 2005, pp. 12–21.

[115] B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors," *Communications of the ACM*, vol. 13, no. 7, pp. 422–426, 1970.

[116] S. Nagaraja and R. Shah, "Clicktok: click fraud detection using traffic analysis," in *Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks*, 2019, pp. 105–116.

[117] G. Thejas, K. G. Boroojeni, K. Chandna, I. Bhatia, S. Iyengar, and N. Sunitha, "Deep learning-based model to fight against ad click fraud," in *Proceedings of the 2019 ACM Southeast Conference*, 2019, pp. 176–181.

[118] L. Jiang, N. S. Sadghiani, and Z. Tao, "Generating multi-type sequences of temporal events to improve fraud detection in game advertising," *arXiv preprint arXiv:2104.03428*, 2021.

[119] T. Pourhabibi, K.-L. Ong, B. H. Kam, and Y. L. Boo, "Fraud detection: A systematic literature review of graph-based anomaly detection approaches," *Decision Support Systems*, vol. 133, p. 113303, 2020.

[120] J. Hu, T. Li, Y. Zhuang, S. Huang, and S. Dong, "Gfd: A weighted heterogeneous graph embedding based approach for fraud detection in mobile advertising," *Security and Communication Networks*, vol. 2020, 2020.

[121] X. Zhang, X. Liu, and H. Guo, "A click fraud detection scheme based on cost sensitive bpnn and abc in mobile advertising," in *2018 IEEE 4th International Conference on Computer and Communications (ICCC)*. IEEE, 2018, pp. 1360–1365.

[122] A. Metwally, D. Agrawal, and A. E. Abbadi, "Using association rules for fraud detection in web advertising networks," in *Proceedings of the 31st international conference on Very large data bases*. VLDB Endowment, 2005, pp. 169–180.

[123] M. S. Iqbal, M. Zulkernine, F. Jaafar, and Y. Gu, "Protecting internet users from becoming victimized attackers of click-fraud," *Journal of Software: Evolution and Process*, vol. 30, no. 3, p. e1871, 2018.

[124] A. Metwally, F. Emekçi, D. Agrawal, and A. El Abbadi, "Sleuth: Single-publisher attack detection using correlation hunting," *Proceedings of the VLDB Endowment*, vol. 1, no. 2, pp. 1217–1228, 2008.

[125] C. Cao, Y. Gao, Y. Luo, M. Xia, W. Dong, C. Chen, and X. Liu, "Adsherlock: Efficient and deployable click fraud detection for mobile applications," *IEEE Transactions on Mobile Computing*, 2020.

[126] C. Shi, R. Song, X. Qi, Y. Song, B. Xiao, and S. Lu, "Clickguard: Exposing hidden click fraud via mobile sensor side-channel analysis," in *ICC 2020-2020 IEEE International Conference on Communications (ICC)*. IEEE, 2020, pp. 1–6.

[127] H. Haddadi, "Fighting online click-fraud using bluff ads," *ACM SIGCOMM Computer Communication Review*, vol. 40, no. 2, pp. 21–25, 2010.

[128] V. Dave, S. Guha, and Y. Zhang, "Measuring and fingerprinting click-spam in ad networks," in *Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication*. ACM, 2012, pp. 175–186.

[129] G. Thejas, S. Dheeshjith, S. Iyengar, N. Sunitha, and P. Badrinath, "A hybrid and effective learning approach for click fraud detection," *Machine Learning with Applications*, vol. 3, p. 100016, 2021.

[130] R. A. Costa, R. J. de Queiroz, and E. R. Cavalcanti, "A proposal to prevent click-fraud using clickable captchas," in *2012 IEEE Sixth International Conference on Software Security and Reliability Companion*. IEEE, 2012, pp. 62–67.

[131] A. M. Turing, "Computing machinery and intelligence," in *Parsing the Turing Test*. Springer, 2009, pp. 23–65.

[132] H. Shaari and N. Ahmed, "An extensive study on online and mobile ad fraud," 2020.

[133] I. Aberathne, C. Walgampaya, and U. Rathnayake, "Novel hidden markov scoring algorithm for fraudulent impression classification in mobile advertising," in *International Congress on Information and Communication Technology*. Springer, 2020, pp. 110–118.

[134] G. Chen, W. Meng, and J. Copeland, "Revisiting mobile advertising threats with madlife," in *The World Wide Web Conference*, 2019, pp. 207–217.

[135] G. Cho, J. Cho, Y. Song, and H. Kim, "An empirical study of click fraud in mobile advertising networks," in *2015 10th International Conference on Availability, Reliability and Security*. IEEE, 2015, pp. 382–388.

[136] C.-T. Huang, M. N. Sakib, C. Kamhoua, K. A. Kwiat, and L. Njilla, "A bayesian game theoretic approach for inspecting web-based malvertising," *IEEE Transactions on Dependable and Secure Computing*, 2018.

[137] S. Suresh, F. Di Troia, K. Potika, and M. Stamp, "An analysis of android adware," *Journal of Computer Virology and Hacking Techniques*, vol. 15, no. 3, pp. 147–160, 2019.

[138] T. Liu, H. Wang, L. Li, X. Luo, F. Dong, Y. Guo, L. Wang, T. Bissyandé, and J. Klein, "Maddroid: Characterizing and detecting devious ad contents for android apps," in *Proceedings of The Web Conference 2020*, 2020, pp. 1715–1726.

[139] E. Rescorla, *SSL and TLS: designing and building secure systems*. Addison-Wesley Reading, 2001, vol. 1.

[140] A. Langley, "Opportunistic encryption everywhere," *In W2SP*, 2009.

[141] C. Reis, S. D. Gribble, T. Kohno, and N. C. Weaver, "Detecting in-flight page changes with web tripwires." in *NSDI*, vol. 8, 2008, pp. 31–44.

[142] N. Vratonjic, J. Freudiger, and J.-P. Hubaux, "Integrity of the web content: The case of online advertising," Tech. Rep., 2010.

[143] R. A. Nofal, N. Tran, C. Garcia, Y. Liu, and B. Dezfouli, "A comprehensive empirical analysis of tls handshake and record layer on iot platforms," in *Proceedings of the 22nd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, 2019, pp. 61–70.

[144] D. Wendlandt, D. G. Andersen, and A. Perrig, "Perspectives: Improving ssh-style host authentication with multi-path probing." in *USENIX Annual Technical Conference*, vol. 8, 2008, pp. 321–334.

[145] "Juniper research," https://www.juniperresearch.com/researchstore/content-digital-media/future-digital-advertising-strategy-report?utm_campaign=pr1_smartcities_technology_apr19&utm_source=businesswire&utm_medium=pr, 2019, [Online; accessed 16-May-2021].

[146] "Nielsen," https://www.nielsen.com/us/en/solutions/capabilities/digital-ad-ratings/, 2020, [Online; accessed 16-May-2021].

[147] "Why your business benefits from using multiple currencies," https://fastspring.com/blog/why-business-benefits-using-multiple-currencies/, 2019, [Online; accessed 16-May-2021].

[148] H. Aksu, L. Babun, M. Conti, G. Tolomei, and A. S. Uluagac, "Advertising in the iot era: Vision and challenges," *IEEE Communications Magazine*, vol. 56, no. 11, pp. 138–144, 2018.

[149] J.-A. Choi and K. Lim, "Identifying machine learning techniques for classification of target advertising," *ICT Express*, 2020.

[150] W.-H. S. Tsai, S. C. Tian, C.-H. Chuan, and C. Li, "Inspection or play? a study of how augmented reality technology can be utilized in advertising," *Journal of Interactive Advertising*, pp. 1–14, 2020.

[151] F. Zhao, "Imagination of mobile media through advertising: Thematic analysis of 4g and 5g ads in china and the us," Ph.D. dissertation, 2020.

[152] "How google's new auto ads use artificial intelligence to automatically boost performance," https://www.marketingaiinstitute.com/blog/how-googles-new-auto-ads-use-artificial-intelligence-to-automatically\protect\discretionary{\char\hyphenchar\font}{}{}boost-performance?fbclid=IwAR1--Iph2iQ6x58DRGmdV-IT1TtC_ZM5SFk78y4rrLzIduEFiBiZdHQymKE, 2018, [Online; accessed 16-May-2021].

[153] "6 essentials for fighting fraud with machine learning," https://www.technologyreview.com/2019/11/18/131912/6-essentials-for-fighting-fraud-with-machine-learning/, 2019, [Online; accessed 16-May-2021].

[154] "Adjust," https://www.adjust.com/blog/why-every-click-needs-an-impression/, 2020, [Online; accessed 16-May-2021].

[155] M. Walfish and A. J. Blumberg, "Verifying computations without reexecuting them," *Communications of the ACM*, vol. 58, no. 2, pp. 74–84, 2015.

[156] N. Kshetri and J. Voas, "Online advertising fraud," *Computer*, vol. 52, no. 1, pp. 58–61, 2019.

[157] M. Asante, G. Epiphaniou, C. Maple, H. Al-Khateeb, M. Bottarelli, and K. Z. Ghafoor, "Distributed ledger technologies in supply chain security management: A comprehensive survey," *IEEE Transactions on Engineering Management*, 2021.

[158] Z. Shae and J. Tsai, "Transform blockchain into distributed parallel computing architecture for precision medicine," in *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2018, pp. 1290–1299.

[159] H. I. Ozercan, A. M. Ileri, E. Ayday, and C. Alkan, "Realizing the potential of blockchain technologies in genomics," *Genome research*, vol. 28, no. 9, pp. 1255–1263, 2018.

[160] "How do we boost transparency in digital advertising?" https://www.exchangewire.com/blog/2019/06/17/how-do-we-boost-transparency-in-digital-advertising/, 2019, [Online; accessed 16-May-2021].

[161] A. Portes, G. N'goala, and A.-S. Cases, "Should digital marketing practices be more transparent? an empirical investigation on the roles of consumer digital literacy and privacy concerns in self-service technologies," in *16th International Research Conference in Service Management*, 2020.

[162] "Iab programmatic fee transparency calculator," https://www.iab.com, 2016, [Online; accessed 16-May-2021].

[163] "How blockchain will dominate the digital advertising industry in 2020," https://www.searchenginewatch.com/2020/02/26/how-blockchain-will-dominate-the-digital-advertising-industry-in-2020/, 2020, [Online; accessed 16-May-2021].
[164] "Havas promises clients programmatic transparency with new offering," https://www.thedrum.com/news/2017/05/23/havas-promises-clients-programmatic-transparency-with-new-offering, 2017, [Online; accessed 16-May-2021].
[165] "Why your small business needs diverse payment options," https://www.business2community.com/small-business/small-business-needs-diverse-payment-options-02041338, 2018, [Online; accessed 16-May-2021].
[166] H. Treiblmaier, "Combining blockchain technology and the physical internet to achieve triple bottom line sustainability: a comprehensive research agenda for modern logistics and supply chain management," *Logistics*, vol. 3, no. 1, p. 10, 2019.
[167] M. A. Camilleri, "The use of data-driven technologies for customer-centric marketing," *International Journal of Big Data Management*, vol. 1, no. 1, pp. 50–63, 2020.
[168] M. Adam, M. Wessel, and A. Benlian, "Ai-based chatbots in customer service and their effects on user compliance," *Electronic Markets*, pp. 1–19, 2020.
[169] W. Zhang, X. Zhao, L. Zhao, D. Yin, G. H. Yang, and A. Beutel, "Deep reinforcement learning for information retrieval: Fundamentals and advances," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 2468–2471.
[170] "The internet of things," https://www.iab.com/insights/connected-devices-internet-things/?fbclid=IwAR1GZZiXonTG2e_nIuu3KpgiPlQcdbfbFxmwgT6CPA35fwn0I_IhPGc1-pk, 2016, [Online; accessed 16-May-2021].
[171] N. Kshetri, "5g in e-commerce activities." *IT Prof.*, vol. 20, no. 4, pp. 73–77, 2018.
[172] M. Sung, J. Kim, E.-S. Kim, S.-H. Cho, Y.-J. Won, B.-C. Lim, S.-Y. Pyun, J. K. Lee, and J. H. Lee, "5g trial services demonstration: Ifof-based distributed antenna system in 28 ghz millimeter-wave supporting gigabit mobile services," *Journal of Lightwave Technology*, vol. 37, no. 14, pp. 3592–3601, 2019.
[173] "5g is coming and will change digital advertising in more ways than you think," https://digiday.com/marketing/5g-coming-will-change-digital-advertising-ways-think/, 2018, [Online; accessed 16-May-2021].
[174] H. Lee, J. Lee, D. Kim, S. Jana, I. Shin, and S. Son, "Adcube: Webvr ad fraud and practical confinement of third-party ads," in *30th {USENIX} Security Symposium ({USENIX} Security 21)*, 2021.
[175] N. Helberger, J. Huh, G. Milne, J. Strycharz, and H. Sundaram, "Macro and exogenous factors in computational advertising: Key issues and new research directions," *Journal of Advertising*, vol. 49, no. 4, pp. 377–393, 2020.
[176] "Five ways 5g will change digital advertising," https://martechseries.com/mts-insights/guest-authors/five-ways-5g-will-change-digital-advertising/, 2019, [Online; accessed 16-May-2021].
[177] "9 biggest risks to disruptive innovation and technology in 2020," https://www.resolver.com/blog/risks-disruptive-innovation-technology/, 2020, [Online; accessed 16-May-2021].

Zahra Pooranian (M'17-SM'21) is a postdoctoral fellow in Network Security at the 5G & 6G Innovation Centre (5GIC & 6GIC), Institute for Communication Systems (ICS), University of Surrey, Guildford, UK. Before joining the University of Surrey, she was postdoctoral at The Alan Turing Institute, London, UK. She was working on Enhancing Security and Privacy of National Identity Systems project. From 2017 to 2019, she was postdoctoral in Network Security at the University of Padua, Padua, Italy. She received her Ph.D. degree in Computer Science Sapienza University of Rome, Italy, in February 2017. She is a (co)author of several peer-reviewed publications (h-index=20, citations=1136+) in well-known conferences and journals. She is an Editor of KSSI transaction on Internet and information systems and Future Internet. Her current research focuses on Machine Learning, Social Network Security, Cloud Security and Cloud/Fog Computing. She was a programmer in several companies in Iran from 2009-2014, respectively. She is a Senior Member of IEEE. For additional information: https://zahrapooranian.github.io/Zahra/

Mauro Conti is Full Professor at the University of Padua, Italy. He is also affiliated with TU Delft and University of Washington, Seattle. He obtained his Ph.D. from Sapienza University of Rome, Italy, in 2009. After his Ph.D., he was a Post-Doc Researcher at Vrije Universiteit Amsterdam, The Netherlands. In 2011 he joined as Assistant Professor the University of Padua, where he became Associate Professor in 2015, and Full Professor in 2018. He has been Visiting Researcher at GMU, UCLA, UCI, TU Darmstadt, UF, and FIU. He has been awarded with a Marie Curie Fellowship (2012) by the European Commission, and with a Fellowship by the German DAAD (2013). His research is also funded by companies, including Cisco, Intel, and Huawei. His main research interest is in the area of Security and Privacy. In this area, he published more than 400 papers in topmost international peer-reviewed journals and conferences. He is Area Editor-in-Chief for IEEE Communications Surveys & Tutorials, and has been Associate Editor for several journals, including IEEE Communications Surveys & Tutorials, IEEE Transactions on Dependable and Secure Computing, IEEE Transactions on Information Forensics and Security, and IEEE Transactions on Network and Service Management. He was Program Chair for TRUST 2015, ICISS 2016, WiSec 2017, ACNS 2020, CANS 2021, and General Chair for SecureComm 2012, SACMAT 2013, NSS 2021 and ACNS 2022. He is Senior Member of the IEEE and ACM. He is a member of the Blockchain Expert Panel of the Italian Government. He is Fellow of the Young Academy of Europe. From 2020, he is Head of Studies of the Master Degree in Cybersecurity at University of Padua. For additional information: http://www.math.unipd.it/~conti/

Hamed Haddadi is a Reader in Human-Centred Systems and the Director of Postgraduate Studies at the Dyson School of Design Engineering at The Faculty of Engineering, Imperial College London. He serves as a Security Science Fellow of the Institute for Security Science and Technology, and is an Academic Fellow of the Data Science Institute. In his industrial role, he is a Visiting Professor at Brave Software where he works on developing privacy-preserving analytics protocols. He enjoys designing and building systems that enable better use of our digital footprint, while respecting users' privacy. For additional information: https://www.imperial.ac.uk/people/h.haddadi

Rahim Tafazolli (SM'09) is a professor and the Director of the Institute for Communication Systems (ICS) and 5G Innovation Centre (5GIC), the University of Surrey in the UK. He has over 30 years of experience in digital communications research and teaching. He has published more than 500 research papers in refereed journals, international conferences and as invited speaker. He is the editor of two books on "Technologies for Wireless Future" published by Wiley Vol.1 in 2004 and Vol.2 2006. He is co-inventor on more than 30 granted patents, all in the field of digital communications. He was appointed as Fellow of WWRF (Wireless World Research Forum) in April 2011, in recognition of his personal contribution to the wireless world. As well as heading one of Europa's leading research groups. He is regularly invited by governments to advise on network and 5G technologies and was advisor to the Mayor of London with regard to the London Infrastructure Investment 2050 Plan during May and June 2014. For more information: https://www.surrey.ac.uk/people/rahim-tafazolli