MindYourPrivacy: Design and Implementation of a Visualization System for Third-Party Web Tracking

Yuuki Takano*, Satoshi Ohta*, Takeshi Takahashi*, Ruo Ando*, Tomoya Inoue†
*National Institute of Information and Communications Technology, Tokyo, Japan
†Japan Advanced Institute of Science and Technology, Ishikawa, Japan
E-mail: ytakano@wide.ad.jp, sota@nict.go.jp, takeshi_takahashi@ieee.org, ruo@nict.go.jp, t-inoue@jaist.ac.jp

***Abstract*—Third-party Web tracking is a serious privacy issue. Advertisement sites and social networking sites stealthily collect users' Web browsing history for purposes such as targeted advertising or predicting trends. Unfortunately, very few Internet users realize this, and their privacy has been infringed upon because they have no means of recognizing the situation. In this paper we present the design and implementation of a system called *MindYourPrivacy* that visualizes third-party Web tracking and clarifies the entities threatening users' privacy. The implementation adopts deep packet inspection, DNS-SOA-record-based categorization, and HTTP-referred graphical analysis to visualize collectors of Web browsing histories without device dependency. To demonstrate the effectiveness of our proof-of-concept implementation, we conducted an experiment in an IT technology camp, where 129 attendees discussed IT technologies for four days, The experiment's results revealed that visualizing Web tracking effectively influences users' perception of privacy. Analysis of the user data we collected at the camp also revealed that MCODE clustering and some features derived from graph theory are useful for detecting advertising sites that potentially collect user information by Web tracking for their own purposes.**

***Keywords*-Security, Network Monitoring, Data and Knowledge Visualization, Web Mining**

I. Introduction

Social networking sites (SNSs) are usually free of charge to use and earn revenue from advertisements (ads) by collecting user information for targeted ads. Although users can use the services for free, and SNSs can profit by leveraging user information, user privacy can be compromised. Targeted ads are usually generated by gathering and mining information. These processes may result in the risk of tracking, usually referred to as third-party Web tracking. During this process, many of the third-party ad sites, external to the original sites the users visit, get to know which users have visited which site and when. Of greater concern is that tracking by SNSs can associate real names of individuals with the tracking information they collected. The number of third-party Web tracking sites is growing [1] each year, making online privacy a significant issue.

We therefore propose *MindYourPrivacy*, a visualization system for tackling the privacy issue caused by third-party Web tracking. *MindYourPrivacy* clearly shows users' Web tracking information by using a tag cloud technique, and it provides graphical files for Web tracking analysis. Moreover, it is designed to be device-independent through adoption of a deep packet inspection technique. Because not only personal computers but also mobile devices are targeted for Web tracking, Web-tracking counter-technologies must be device-independent.

To demonstrate the system's effectiveness, we conducted an experiment at the Widely Integrated Distributed Environment (WIDE) camp in the autumn of 2013, a four-day workshop on Internet Technology held in Japan and attended by 129 people. We collected and analyzed users' traffic. Our analysis of the results revealed that MCODE clustering and some features derived from graph theory are useful for detecting ad sites, which may potentially collect user information by Web tracking for their business purposes. We also conducted a questionnaire-based user study, which indicated that visualizing Web tracking effectively influences users' perceptions of privacy.

The remainder of the paper is organized as follows. Section II presents related works on Web tracking, detection techniques, and measurement studies. Section III explains the design and implementation of *MindYourPrivacy*, describing why *MindYourPrivacy* is required and how it achieves visual display of Web tracking. Section IV describes the results of an experiment we conducted using *MindYourPrivacy*. Section V discusses limitations of our system, future studies, and the future view of Web tracking protection. A conclusion is given and future work is described in Section VI.

II. Related Work

A. Web Tracking Mechanisms

There are assorted Web tracking techniques, which can be essentially classified into stateless or stateful tracking. Stateless tracking sets identifiers for distinguishing users by using means such as cookies and ETags. Stateful tracking uses the fingerprint of Web browsers or operating systems to identify users. In general, Web trackers adopt multiple ways to track users. Mayer et al. [2] performed a thorough analysis of Web tracking and reported its mechanisms.

Roesner et al. [3] discussed third-party Web trackers by SNSs such as Facebook and Twitter; these Web trackers can associate real names of individuals with the tracking information they collect. Their tracking is performed by using supplied social widgets, such as "Like" or "Tweet" buttons. They also reported on tracking mechanisms and their effectiveness.

978-1-4799-3503-1/14/$31.00 ©2014 IEEE

B. Web Tracking Detection Techniques

The above Web tracking techniques are rather simple, but constructing countermeasures to them can be difficult. One simple solution is to disable JavaScript, but this could significantly deteriorate the quality of users' Web experience since many Websites use JavaScript to boost users' online experience. Moreover, some sites are not correctly displayed because the use of JavaScript is assumed. Schemes that can maintain the user experience need to be considered.

One approach for that is *ShareMeNot* [3], which swaps a link to known data-collection sites such as Facebook. When users click the link, the original link is activated. In this way, links contained by images or widgets don't leak user information. This approach can also maintain the functionality of the original link since the link can be activated only when the user clicks on it. This function was implemented on the user terminal. Note that this scheme uses a predefined list of trackers, and users need to specify trackers' domains.

Lightbeam [4] is a Firefox add-on for visualizing Web tracking by showing a connection graph of Websites. It helps to find who is watching users' browsing histories, but it depends on a specific browser.

C. Measurements

Several works on measurement have also been reported. Roesner et al. [3] also reports measurement results on the most popular trackers. They collected a set of data from the top-500 Websites according to Alexa Internet, Inc. [5], and clarified which domains are the top 20 trackers. Balachander et al. [1] reported measurement results of third-party domains. Their measurements revealed that Google, Omniture, and Microsoft were the top-three third-party domain holders in 2008. They also proposed a categorization technique using a DNS SOA record we adopt in our system. Carlos et al. [6] reported measurement results of WebCrawler. They crawled 15 months worth of data and obtained 240,000 pages from over 24,000 domains across 47 countries. Gill [7] analyzed Web tracking economics and the impact of blocking ads. They reported that advertising revenue decreased by 30% if the top 5% of users who contribute the most to advertising revenue block ads.

III. Design and Implementation

We designed and implemented *MindYourPrivacy* to show Internet users which companies or organizations collect users' Web browsing history. In this section we explain the design and implementation of *MindYourPrivacy*, which shows and visualizes collectors of private information. The implementation is device-agnostic and can be freely downloaded from GitHub [8].

A. Design Principle

We have two design principles: 1. The implementation must be independent of browsers and devices and 2. the analysis results must be accessible and comprehensive.

1) Independence from Browsers and Devices: Users' Web browsing traffic must be analyzed to identify the organizations collecting users' data. There are two approaches for the analysis—a browser add-on (including a plug-in) or a deep-packet-inspection-based approach. Lightbeam [4] utilizes the functionality of Firefox add-ons and visualizes a connectivity graph of Websites visited by the user. However, browser add-ons are available only for specific browsers. Implementing add-ons for every browser is costly owing to the wide variety of browsers (e.g., Mozilla Firefox, Google Chrome, Opera, and Microsoft Internet Explorer). Moreover, the existence of various OSs or devices such as Linux, Windows, MacOS, and smartphone OSs such as Android and iOS complicates the problem. In contrast, the deep-packet-inspection-based approach has a substantial advantage for deployment and offers reduced implementation costs because of browser and device independence. We therefore adopt such an approach to support heterogeneous browsers and devices.

2) Accessibility and Comprehensiveness of the Analysis Results: MindYourPrivacy provides analysis results in the form of an HTML file via an HTTP server to facilitate users' access to them. Users are therefore not required to install any specific applications or browser add-ons to access and use the results. Users thus only need to have a Web browser to access the results. To supply an easy way to understand the results, *MindYourPrivacy* expresses results visually in tag cloud format. Figure 1 shows an analyzed result of a Web browsing history as presented by the Web user interface provided by *MindYourPrivacy*. The details of Figure 1 are discussed after this section, but for now notice how easy it is to identify which sites are potentially collecting users' browsing history because noticeable sites are expressed in a large font. In this case, we notice that Facebook, Twitter, Google, and Hatena[1] potentially collected the browsing history. In addition to tag cloud expression, *MindYourPrivacy* provides users with a Graphviz [9] *.dot* file and a Cytoscape [10] *.sif* file for more sophisticated analysis. Users can download each file via *MindYourPrivacy*'s Web user interface.

B. Web Tracker Identification Methodology

1) HTTP Referrer Graph Analysis: The HTTP referrer header [11] has source URL information. If URL_α is called by URL_β, the value of the HTTP referrer header of URL_α becomes URL_β. Images and other types of files for Web tracking are usually embedded in targeted sites. These are automatically or manually loaded and then Web browsing information is sent to tracker sites. Because tracking sites tend to be embedded and referred by many sites to obtain Web browsing histories of Internet users, we detect suspicious sites tracking users by finding sites referred by many other sites from the HTTP referrer graph.

2) Domain Aggregation: To show users which organizations track them, we aggregate domains as either second or third level. For example,

[1]Hatena is an IT and Web-oriented company in Japan.

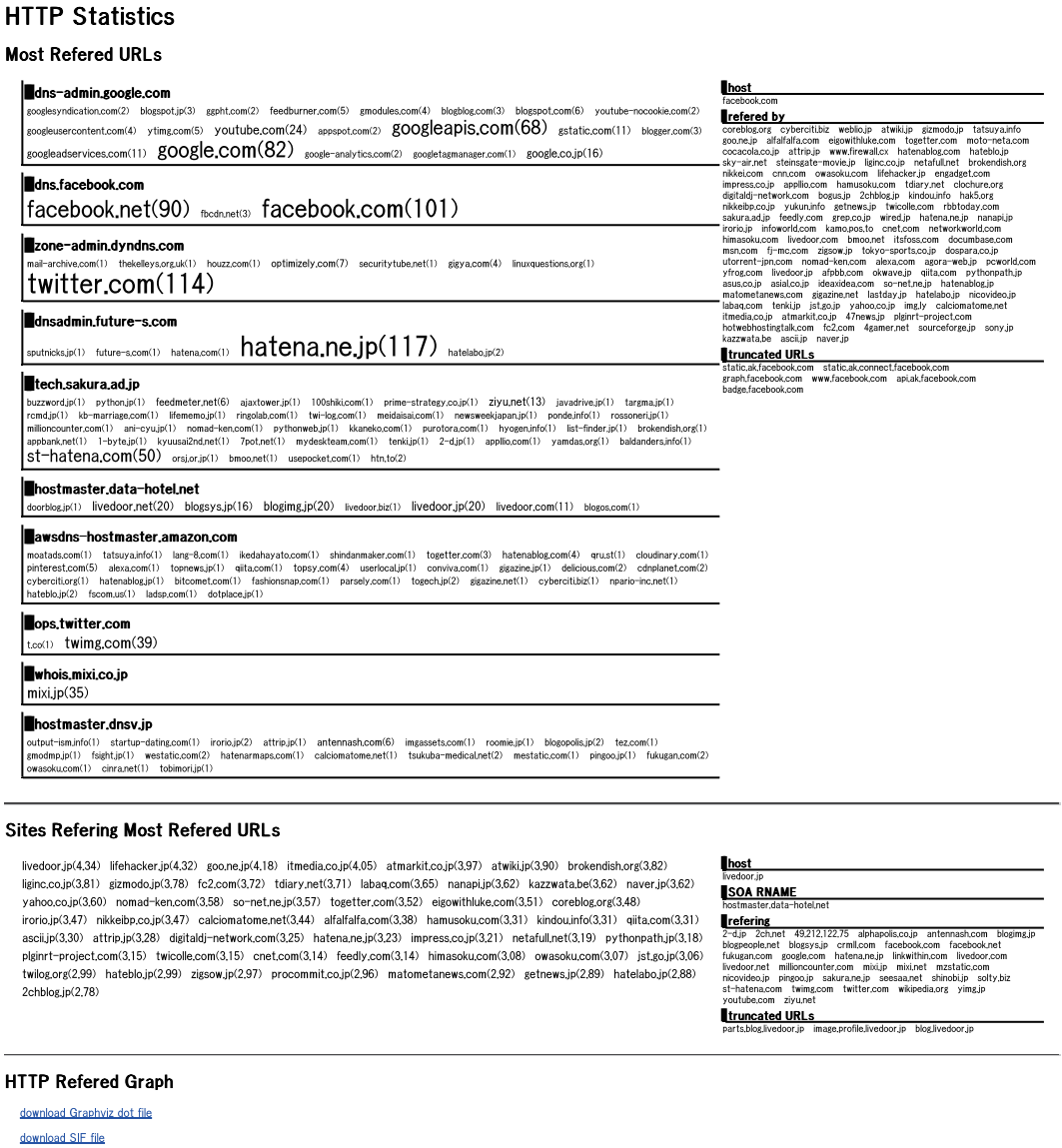

Fig. 1: *MindYourPrivacy*'s Web User Interface

`platform.twitter.com` and `platform0.twitter.com` are aggregated to `twitter.com` as second-level domains, and `s.hatena.ne.jp` and `d.hatena.ne.jp`, whose `.ne` is a country code second-level domain (ccSLD) of `.jp` top-level domain (TLD), are aggregated as third-level domains. By aggregating the domain, users can easily understand to which sites many sites refer or which sites potentially track users.

3) DNS-SOA-Record-Based Grouping: We then aggregate domains by looking up their DNS SOA records [1]. For instance, `facebook.com` and `facebook.net` are aggregated into `dns.facebook.com`, which is their DNS SOA record.

4) Weighted Site Ranking of User Data Leakage: We also tell users which sites definitely leak users' browsing information to tracking sites. To do this, we compute a weighted site ranking of data leakage by using an HTTP referrer graph. The weighted score S_n of a node n is given by

$$S_n = \sum_{i=0}^{L_n-1} \left(\frac{L_{n,i}}{L_{max}} \right)^2, \quad (1)$$

where L_n is the number of outgoing links of the node, $L_{n,i}$ is the number of incoming links to the node, which is linked by the node n's ith link, and L_{max} is the number of incoming links of the node, which has the most incoming links in a graph. The more node n gains outgoing links to nodes linked by many other nodes, the higher S_n becomes.

C. System Architecture and Implementation

Figure 2 shows the system model of *MindYourPrivacy*, which captures users' Web browsing traffic at gateways, analyzes it, and displays the analysis results to users. Users are not required to have any protocol-specific applications or

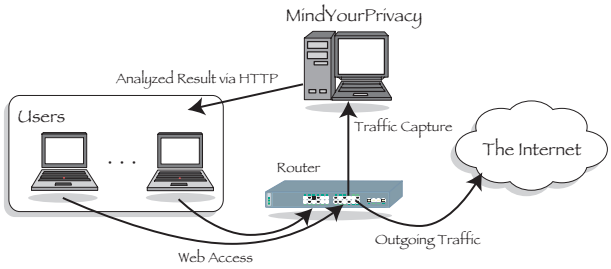

Fig. 2: *MindYourPrivacy*'s System Model

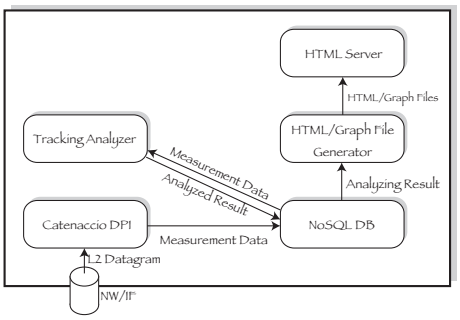

Fig. 3: *MindYourPrivacy*'s Implementation Architecture

configurations. This system is transparent to users, and they can access the Internet without any restrictions.

Figure 3 shows the *MindYourPrivacy* implementation architecture. It consists of five components: Catenaccio DPI, a tracking analyzer, a NoSQL database (DB) of a key-value-type database, an HTML/Graph file generator, and an HTTP server.

1) Catenaccio DPI (CDPI): CDPI is an application-level packet analyzer that captures HTTP and other traffic. Figure 4 shows its layered model. CDPI makes use of libpcap [12] to capture the layer-2 frame from network interfaces in a packet capture plane. The L3/L4 flow controller controls the TCP stream and UDP datagram as flows. The TCP stream controller handles TCP flow, reconstructs the TCP stream, and detects the application-level protocol. The UDP datagram controller handles UDP packets and also detects the application-level protocol. Each application protocol parser parses the TCP stream or UDP datagram and invokes events to the upper layer. For example, the HTTP parser invokes events of *HTTP READ METHOD*, *HTTP READ RESPONSE*, *HTTP READ HEAD*, *HTTP READ BODY*, *HTTP READ TRAILER*, *HTTP READ*, and *HTTP PROXY*. The DB controller receives events from application protocol parsers and stores results in a database. We implemented CDPI in C++, and it depends on libboost [13], libpcap, libcrypto [14], and MongoDB's C++ driver [15]. Its source code is available online [8].

2) NoSQL Database: The NoSQL DB is used for data storage. We adopt a NoSQL database instead of a SQL database since NoSQL databases have better flexibility and are thus suitable for rapid prototyping. Moreover, a key-value-type DB is suitable for HTTP information storage because the HTTP header has a key-value-type structure, and keys

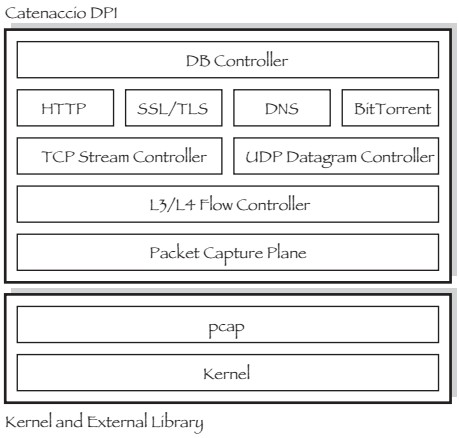

Fig. 4: Layered Model of Catenaccio DPI

cannot be estimated before analyzing the HTTP stream. The implementation uses MongoDB as its NoSQL database.

3) Tracking Analyzer: The tracking analyzer analyzes tracking and leaking sites of users' Web browsing information by using the methods described in Section III-B. It draws out measurement data from the NoSQL DB, analyzes data, and again stores analyzed results into the NoSQL DB. We implemented the tracking analyzer in JavaScript by using the MapReduce [16] function of MongoDB. Moreover, we implemented a DNS inquirer depending on libunbound [17] in C++ for DNS-SOA-record-based domain categorization.

4) HTML/Graph File Generator: This is an HTML and graphical file generator written in Python. Figure 1 shows an example of a generated HTML file; graphical files are provided in *.dot* and *.sif* format.

5) HTTP Server: The HTTP server provides HTML and graphical files to users. We used lighttpd [18], but other HTTP servers can also work. Users can easily access measurement and analyzed data via their Web browser.

D. Web User Interface

Figure 1 shows *MindYourPrivacy*'s Web user interface. It consists of three panes: most-referred URLs, sites referring most-referred URLs, and HTTP-referred graph panes.

1) Most-referred URLs Pane: This pane shows URLs referred by other many URLs. To show which sites potentially collect browsing information intelligibly, a larger font denotes suspicious URLs, and the referred number follows each URL. For example, Figure 1 shows "`facebook.net`(90)," which means `facebook.net` is found to be referred by 90 URLs.

These URLs are categorized by using the DNS SOA records discussed in Section III-B. For example, Figure 1 categorizes `facebook.net`, `fbcdn.net`, and `facebook.com` into `dns.facebook.com`. The category indicates that the same organization uses `facebook.net`, `fbcdn.net`, and `facebook.com`. However, SOA-based categorization has limitations. For example, `awsdns-hostmaster.amazon.com` is a category, but URLs belonging to `awsdns-hostmaster.amazon.com` are

TABLE I: WIDE Camp (2013 Autumn) Attendees

	Working Adult	Student	Total
Male	78	39	117
Female	1	11	12
Total	79	50	129

not operated by the same organization. These URLs are only operated on Amazon's hosting service.

Displayed URLs are clickable, and details of a URL are shown in the upper-right area of the Web user inerface. The area shows sites referring to the URL in clicked and aggregated URLs. In Figure 1, for example, `facebook.com` is referred by `coreblog.org`, `cyberciti.biz`, etc., and `static.ak.facebook.com`, `www.facebook.com`, etc. are aggregated to `facebook.com`.

2) Sites Referring Most-referred URLs Pane: This pane shows which sites definitely leak user browsing data to tracking sites. The leaking sites are listed using the notation of the weighted score of data leakage discussed in Section III-B. In Figure 1, for example, `livedoor.jp`'s leakage score is 4.34 because it is denoted as "`livedoor.jp`(4.34)."

These displayed URLs are also clickable. Users can obtain more detailed information on URLs—referring sites, DNS SOA RNAME, and aggregated URLs—and this is shown at the lower-right area of the Web user inerface. For example, through Figure 1, one can learn that `livedoor.jp`'s DNS SOA RNAME is `hostmaster.data-hotel.net`, it refers `2-d.jp`, `2ch.net`, etc., and `parts.blog.livedoor.jp`, `image.profile.livedoor.jp`, and `blog.livedoor.jp` are aggregated to `livedoor.jp`.

3) HTTP-referred Graph Pane: This pane provides referred graph files in *.dot* and *.sif* formats. Users can download these files from here and analyze or visualize the referred graph by using Graphviz, Cytoscape, etc. Figures 5 and Figure 6 show visualization examples using Cytoscape. Through this sort of visualization users can easily find to which sites many other sites refer.

IV. Experiment

To demonstrate the usability and effectiveness of the proposed system, we conducted an experiment at WIDE camp held during September 10–13 2013.

A. Setup

The WIDE project [19] is a research and development project in Japan aimed at developing a widely integrated distributed environment. It organizes camps every spring and autumn, with many researchers, developers, and students taking part and discussing Internet technologies. Table I lists the breakdown of the camp attendees. There were 129 attendees, most of whom are either IT specialists or students majoring in IT. We have conducted two types of experiments: user traffic analysis and questionnaire-based use analysis.

TABLE II: Top-five Most-referred Sites

Site	# of incoming links
`google-analytics.com`	847
`facebook.com`	437
`twitter.com`	393
`doubleclick.net`	380
`google.com`	356

B. User Traffic Analysis

We captured all the attendee's network traffic, with which we analyzed cookie and ad sites. Figure 5 shows the overall HTTP referrer graph generated by *MindYourPrivacy* and Cytoscape. The graph has a total of 3,966 nodes and 12,941 edges. In this experiment, we obtained 734,194 HTTP requests and 1,661 individual source IP addresses including IPv4 and IPv6. We observed only 40,650 ($40,650/734,194 \approx 6\%$) Do Not Track–flag enabled HTTP requests. Although modern Web browsers provide users with a Do Not Track option, almost no one uses it.

Table II lists the top-five most-referred sites in Figure 5 of the HTTP referrer graph. From this table, we found that GoogleAnalytics, Google's Web site analyzer, has the largest number of incoming links, and SNSs, such as Facebook, Twitter, and Google+, and DoubleClick, Google's ad site, have many incoming links.

We then analyzed the cookies of Twitter and Facebook, the most popular SNSs. `platform.twitter.com` and `www.facebook.com` are used for Web widgets ("Tweet" and "Like" buttons). Table III lists suspected tracking cookies, whose values are random text strings, the number of cookie values we observed, and examples. In total we observed 2,309 and 2,671 requests for `platform.twitter.com` and `www.facebook.com`, respectively. However, we found only about 100 unique values for each cookie, though *fr* of `www.facebook.com` is 397. *fr* thus does not seem to be tracking cookies, and the 100 likely indicates the number of attendees (which was also around 100) or devices. The results reveal that tracking cookies can also be used for per-user analysis and visualization.

We then applied MCODE clustering [20] to the graph in Figure 5 to find further features. This allowed us to observe many ad sites clustered into the rank 1 cluster by MCODE. The following domains were ad sites found in the rank 1 cluster of Figure 6:

> `doubleclick.net`, `amazon-adsystem.com`, `googleadservices.com`, `i-mobile.co.jp`, `advg.jp`, `adingo.jp`, `iogous.com`, `admeld.com`, `criteo.com`.

Ad sites generally tend to collect user information for business purposes. We therefore should be concerned with the privacy issues they present. This discovery should help further analysis and visualization concerning such sites. Table IV lists the feature vector of ads and other sites that appeared in Figure 6.

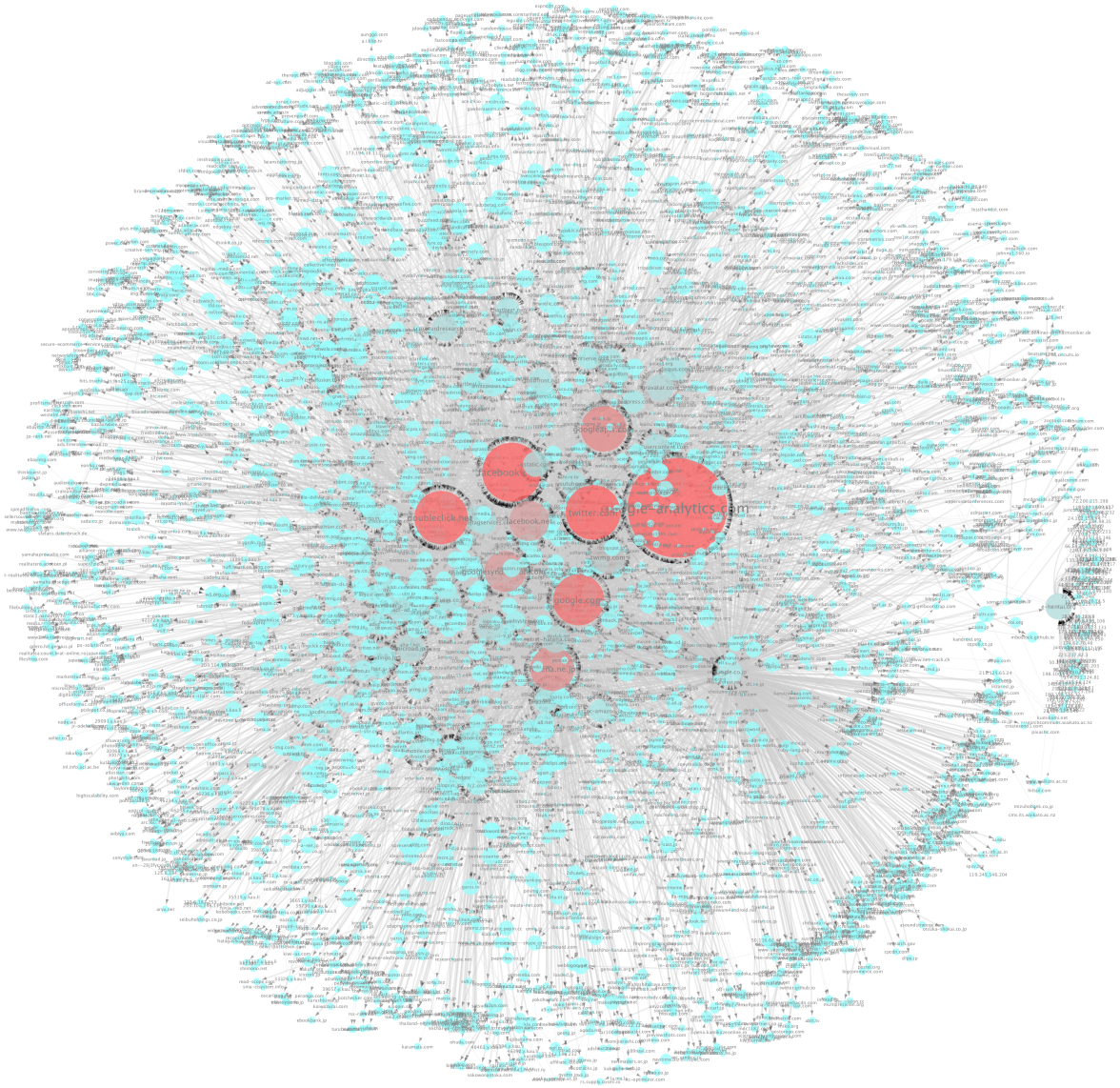

Fig. 5: HTTP Referrer Graph of WIDE Camp (2013 Autumn) Attendees (All)

As shown in the figure, the average numbers of incoming and outgoing links of non-ad sites are quite similar, whereas ad sites tend to have many incoming links but few outgoing links. Therefore the ratio of outgoing and incoming degrees would seem to be helpful in detecting ad sites. Moreover, we found that neighborhood connectivity is quite different between advertisement and other sites. This should also be useful for classification purposes. Further analysis of ad sites needs to be performed but is beyond the scope of this paper. Notice that the unbiased variances listed in Table IV are quite large. This results from the well-known fact that the number of Webpage links follows a power law [21].

C. Questionnaire-based User Analysis

We conducted a questionnaire-based study to identify the users' thoughts on privacy and how *MindYourPrivacy* affects them. We asked three questions to all the attendees and received 56 answers.

Question 1: Are you typically aware of Web tracking?: We first checked users' awareness of Web tracking, finding that 29 are typically aware of it while 27 are not. Thus about 50% of them are not typically aware of Web tracking even though most of them are network specialists. We suspect that the covert nature of Web tracking leads to this result; most Web trackers silently collect user information in the background of main Webpages.

Question 2: Do you take any measures against Web tracking? If so, which?: We then asked how they cope with Web tracking. Subjects can select from multiple-choice answers or provide free responses. Table V summarizes their answers..

It is observed that more than 50% (30 people) take no

TABLE III: SNS Cookies

Site	Cookie	#	Value example
platform.twitter.com	twll	96	l%3D1331765309
			l%3D1333118000
			l%3D1335671305
	guest_id	141	v1%3A131651075276670565
			v1%3A131721129117754955
			v1%3A132349265936441847
www.facebook.com	fr	397	00cR2iyYceFlIPB2P.AWUGBBr8MhcaIM4BALQwG5kdLao.BQ-hz2.kK.FIv.AWU0lIEH
			00cR2iyYceFlIPB2P.AWUSiuGBHhqaESzfy63moRc4g3M.BQ-hz2.kK.FId.AWUXHGm3
			00cR2iyYceFlIPB2P.AWUjl0ipaMJZ9NE_rdSUtbIPc88.BQ-hz2.kK.FIv.AWUTqwp6
	datr	130	1-scUgBUZkbTHDd73b_op-M6
			14UlUmdprx8ANvJbAdiIkIQT
			19_CUe7FcK5cGowt6ArJOWjC
	lu	118	RA0Irow8ICRZYIYPWOWwkHtA
			RAEOvoS3QYv1w5LmXwBruB9g
			RAGC-6X5oD2AuhtcpU2SvQPw

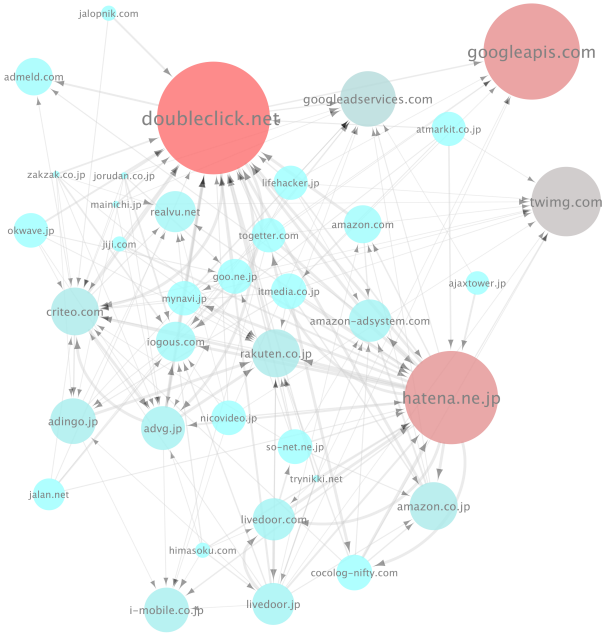

Fig. 6: Rank 1 Cluster by MCODE (include loops = false, degree cutoff = 2, haircut = true, fluff = false, node score cutoff = 0.2, k-core = 2, and max. depth = 100)

TABLE IV: Feature Vector of Rank 1 Cluster's Edge (Average and Unbiased Variance)

	#incoming links		# of outgoing links		Neighborhood connectivity	
	avg.	var.	avg.	var.	avg.	var.
ad sites	90.2	12405.4	15.2	3972.9	46.0	3972.9
others	30.2	3972.9	29.7	569.3	130.2	5212.0

measures, and the most popular measure is to use multiple browsers. Although multiple browser usage does not strictly prevent user privacy leaking to trackers, critical privacy leakage will be avoided. A DNT (Do Not Track) flag [22] indicates users' intention of tracking to Websites, but only seven people use this. We consider that this is because of the fact that

TABLE V: Answers to Question 2 (Multiple Choices Allowed)

Question	#	%
No	30	53.6
Use multiple browsers	16	28.6
Enable Do Not Track flag	7	12.5
Do not use SNSs	1	1.8
Disable HTTP referrers	1	1.8
Disable HTTP cookies	2	3.6

the DNT flag is not a technological way of restricting Web tracking; it is just a guideline. Only three people disable HTTP referrers or cookies. This is likely because doing so decreases online usability. We then observed that only one person does not use SNSs. This reveals that SNSs are now a major form of infrastructure. We therefore need to seriously address the pros and cons of SNSs.

The free-form text answers were summarized as follows:

- Use private browsing mode
- Delete HTTP cookies frequently
- Use AdBlock plug-in
- Absolutely do not mind tracking

Modern Web browsers provide users with a private browsing mode to isolate the browsing environment. Some attendees responded that they use this to avoid critical privacy leakages. Some of them frequently delete HTTP cookies. The reason for not disabling HTTP cookies is likely a usability problem. Some attendees also use AdBlock, a browser plug-in that blocks online ads but this does not prevent users from privacy leakage through SNSs. Unlike in previous answers, some attendees answered that they absolutely do not mind Web tracking. Such an answer reconfirmed that feelings toward privacy are quite different among individuals.

Question 3: Did you change your mind about Web tracking after seeing the experiment?: The final question was about user perception of Web tracking before and after the experiment. Table VI lists the answers to this question. Unfortunately, seven attendees did not see our experiment. We assume that they had no interest in Web tracking. Eight attendees also

TABLE VI: Answers to Question 3

Question	#	%
Didn't see the experiment	7	12.5
Didn't mind and won't mind it in the future	8	14.3
Didn't mind but will mind it in the future	12	21.4
Minded but won't mind it in the future	1	1.8
Minded and will mind it in the future	22	39.2
No answer	6	10.7

answered that they do not care about it, bringing the total of those who were not concerned with Web tracking to 15 people ($15/56 \approx 27\%$). Twelve attendees answered that they did not mind Web tracking before but they will mind it in the future. This is the most important result in this paper as it reveals the importance of clearly informing users what is happening while browsing to raise awareness concerning privacy. Meanwhile, 22 attendees also reported that they will mind Web tracking in the future, but they also previously had concern. Since this is a population of computer-literate people, these 22 were already aware of Web tracking. A total of 34 ($34/56 \approx 60\%$) people said that they mind Web tracking. Only one person who had minded Web tracking before answered that he or she did not mind it after the experiment. We assume that this person noticed that Web tracking is not something that is personally critical.

V. Discussion

A. Limitations of Proposed Scheme

The proposed scheme sometimes cannot work as expected. One of the major problems is that there are cases in which the actual operator differs from the registered entity for the DNS. For instance, one can have a private domain name and run his or her own Website under that name. To register the domain name, the person asks a hosting service provider, which then registers the domain name under its own name rather than that of the registrant.

However, the proposed scheme cannot analyze HTTP traffic correctly if the referrer field of the HTTP header is omitted. Some browser configurations in fact remove the referrer field. Nevertheless, this is not substantial issue when measuring all Internet traffic.

B. Toward Further Analysis and Visualization

In the experiment, we analyzed traffic per IP address to visualize per-user Web-tracking status, but more precise per-user visualization needs to be performed. To achieve this, the cookie-based analysis and visualization described in Section IV-B would be helpful. Because cookies are used to identify and track users, it is also important to analyze them and show users what information they contain. We mainly showed users many referred sites in the form of a Web tracker. Nevertheless, as described in Section IV-B, many sites do not refer ad sites, which tend to track users, and clustering sites appear helpful for showing details of ad sites. Further analysis of such sites needs to be performed.

C. Toward Privacy-aware Services

It is common knowledge that well-known domains such as `facebook.com` and `twitter.com` collect huge amounts of data, and this could lead to privacy infringement. The situation is not currently regarded as a social issue because there have been no substantial incidents of privacy right infringement. Nevertheless, when end users become aware of privacy issues, services that collect greater amounts of user information may experience a decline in popularity. In another words, services with a larger circle in Figure 5 will become less popular. We envision a market in which service providers try to minimize the size of the circle to acquire more customers. Though a certain tradeoff between usability and security exists, service providers need to begin thinking in this context about the means to grow the number of customers and increase their satisfaction.

D. Toward Web Tracking Protection for Mobile Devices

ShareMeNot [3] is an efficient browser plug-in for protecting user devices from Web tracking by SNSs, but it is difficult to implement on mobile devices, which have several restrictions that prevent implementation. One major such restriction is the limitations of Web browser extensions. Though various Web tracking detection and protection schemes have been proposed, they are mostly implemented in the form of such extensions. In the case of smartphones, most users can download these extensions from the platform vendor (e.g., Apple and Google). The use of iOS and Android terminals does not permit download of such extensions through these markets. Users cannot freely choose Web browser extension packages and vendors can control which packages are installed. For instance, the browser extension package AdBlock is not available in the Google Play App Store [23], where users usually select and download arbitrary application packages, because Google claims that the package violates its company policy.

Mobile devices are personalized, and a great deal of private information can be obtained by carefully analyzing their behavior. Though it is difficult to ascertain to what extent service providers can obtain such information, the current situation in which users are unaware of this information leakage is of concern because of possible privacy infringement. We thus visualized what parties collect what types of information.

Current schemes assumed to be implemented over end terminals, including *ShareMeNot*, could be extended further and implemented into a device's first-hop router to avoid mobile device restrictions. In this way, the schemes could become agnostic to the end-user devices.

ISPs could also actively work on this issue. Instead of having first-hop routers of individual user devices, routers inside ISPs could implement these privacy-preserving schemes. That could be one direction to pursue, though it would be difficult for ISPs to inspect IP packet payloads owing to privacy law restrictions.

VI. Conclusion and Future Work

The proposed visualization system for third-party Web tracking, called *MindYourPrivacy*, shows users which sites collect and potentially track users. It uses deep packet inspection, tag cloud, DNS-SOA-record-based domain categorization, weighted scoring of data leakage, and graphical analysis techniques. The deep packet inspection technique enables a device-independent means, and tag cloud, SOA-based categorization, and weighted scoring clearly show users the result.

The user data analysis indicated that cookie information is useful for per-user analysis, and the MCODE clustering method is helpful for clustering and finding ad sites. The questionnaire-based user study revealed that visualization of Web tracking effectively influences users' perceptions of privacy, but some people are completely unconcerned about Web tracking. This indicates that thinking toward privacy varies quite substantially among individuals.

In the current design and implementation, per-user analysis is not thoroughly performed, and cookie analysis needs to be applied in the future. In addition to cookie analysis, ad sites, which tend to collect user information, should also be analyzed deeper. We will apply the MCODE clustering and feature vector discussed in Section IV-B to *MindYourPrivacy* to analyze ad sites in the future.

Acknowledgment

We thank the WIDE Project for assisting our experiment.

References

[1] B. Krishnamurthy and C. E. Wills, "Privacy diffusion on the web: a longitudinal perspective," in *WWW* (J. Quemada, G. León, Y. S. Maarek, and W. Nejdl, eds.), pp. 541–550, ACM, 2009.
[2] J. R. Mayer and J. C. Mitchell, "Third-Party Web Tracking: Policy and Technology," in *IEEE Symposium on Security and Privacy*, pp. 413–427, IEEE Computer Society, 2012.
[3] Roesner, Franziska and Kohno, Tadayoshi and Wetherall, David, "Detecting and defending against third-party tracking on the web," in *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*, NSDI'12, (Berkeley, CA, USA), pp. 12–12, USENIX Association, 2012.
[4] Lightbeam for Firefox, "http://www.mozilla.org/en-US/lightbeam/."
[5] Alexa - The Web Information Company, "http://www.alexa.com/," Sept. 2010.
[6] C. Jensen, C. Sarkar, C. Jensen, and C. Potts, "Tracking website data-collection and privacy practices with the iWatch web crawler," in *SOUPS* (L. F. Cranor, ed.), vol. 229 of *ACM International Conference Proceeding Series*, pp. 29–40, ACM, 2007.
[7] P. Gill, V. Erramilli, A. Chaintreau, B. Krishnamurthy, D. Papagiannaki, and P. Rodriguez, "Follow the Money - Understanding Economics of Online Aggregation and Advertising," in *Internet Measuremet Conference 2013*, 2013.
[8] Catenaccio DPI, "https://github.com/ytakano/catenaccio_dpi."
[9] Graphviz - Graph Visualization Software, "http://www.graphviz.org/."
[10] Cytoscape: An Open Source Platform for Complex Network Analysys and Visualization, "http://www.cytoscape.org/."
[11] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, and T. Berners-Lee, "Hypertext Transfer Protocol – HTTP/1.1." RFC 2068 (Proposed Standard), Jan. 1997. Obsoleted by RFC 2616.
[12] TCPDUMP/LIBPCAP public repository, "http://www.tcpdump.org/."
[13] Boost C++ Library, "http://www.boost.org/."
[14] OpenSSL: The Open Source Toolkit for SSL/TLS, "http://www.openssl.org/."
[15] MongoDB, "http://www.mongodb.org/."
[16] Jeffrey Dean and Sanjay Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters," in *OSDI*, pp. 137–150, USENIX Association, 2004.
[17] Unbound, "http://unbound.net/."
[18] Home - Lighttpd - fly light, "http://www.lighttpd.net/."
[19] WIDE PROJECT, "http://www.wide.ad.jp/."
[20] G. D. Bader and C. W. V. Hogue, "An automated method for finding molecular complexes in large protein interaction networks," *BMC Bioinformatics*, vol. 4, p. 2, 2003.
[21] A.-L. Barabási and R. Albert, "Emergence of Scaling in Random Networks," *Science*, vol. 286, pp. 509–512, October 15 1999.
[22] Tracking Preference Expression (DNT) W3C Working Draft 12 September 2013, "http://www.w3.org/TR/tracking-dnt/."
[23] Adblock Plus for Android removed from Google Play store , "https://adblockplus.org/blog/adblock-plus-for-android-removed-from-google-play-store."