The difficulties of defending against web tracking

Authors
Darrell Newman, MSc (Royal Holloway, 2017)
Geraint Price, ISG, Royal Holloway

Abstract

In this article, we shall look at some of the difficulties that arise while trying to defend ourselves against being tracked as we go about our daily lives on the internet. The content presented here is based on my thesis, *Evaluating the effectiveness of defences to web tracking*, which investigates the many techniques organisations use to track users, as well as taking a close look at how well some of the popular defences perform against tracking. My aim for this article is to introduce you to web tracking, provide an overview of how organisations track users and finally to discuss a few of the difficulties you may face when trying to defend against it. To provide some background before we jump in, we will first examine web tracking and the types of organisations performing it. We will also touch on the downsides of being tracked, some of which can make for uneasy reading, and discuss two of the main techniques used for carrying out web tracking. Finally, from there we'll briefly look at common defences, before moving onto the difficulties involved in trying to defend against being tracked whilst online. Let's jump in! [a]

[a] This article is published online by Computer Weekly as part of the 2018 Royal Holloway information security thesis series `https://www.computerweekly.com/ehandbook/The-difficulties-of-defending-against-web-tracking`. It is based on an MSc dissertation written as part of the MSc in Information Security at the ISG, Royal Holloway, University of London. The full thesis is published on the ISG's website at `https://www.royalholloway.ac.uk/research-and-teaching/departments-and-schools/information-security/research/explore-our-research/isg-technical-reports/`.

Web whoobywhaty?!

For the purpose of this article, the definition of web tracking is the ability to reliably identify a user or device so as to associate online actions and activities with that user or device. More generally, we mean the harvesting of information provided by users - both directly and indirectly - while they visit web sites and use internet based services such as email or social media, and then linking this information to a particular individual, or device. A few examples of what can be collected include the queries you entered into search engines, your email address, date of birth and telephone number used to create a profile on a job site or a social media account, and the postcode you entered to restrict the results to your local geographical area when looking for a new car. The variety of information provided by users can be vast. To help put it in perspective, think about what you do whilst online. Now think about writing down every search term you've ever entered into a search engine, the details of every web site you've ever visited, and the contents of every email you've ever sent or received using one of the many free email providers available. In isolation, much of this information has little meaning. However, now think about it in the wider context of all these pieces of information aggregated together over years and years of internet usage, to produce a profile specifically about you. That sounds quite scary, so perhaps we should add some colour. There's no single, master profile of all your internet habits. Rather, there are multiple, disparate profiles owned by different entities and organisations, some of which become aggregated together, many of which contain the same information. We are

"Collectibles"

- Your queries.
- Your email address.
- Your date of birth.
- Your post code.
- Your phone number.
- ...

creatures of habit, after all.

These profiles are seen as a goldmine by many, and online advertisers are an obvious interested party. Most of us are accustomed to seeing online adverts that appear to know what we're thinking or searching for. These adverts are served based on our online activities and the sites we have visited. The more an advertiser knows about an audience the better placed they are to serve relevant ads, and ultimately sell more products. In order to improve accuracy some advertisers augment profiles by purchasing data from other entities, both online and offline, such as data aggregators. Data aggregators are third parties present on web sites that track users and collect data to sell on. They may provide content or services, such as news articles or an analytical service such as monitoring user interactions to increase product purchases for the site, in exchange for information. Once collected, the information is classified and sold on by category, or as a whole.

Who wants to know?

- Advertisers.
- Data aggregators.
- Cyber thieves.
- ...

Your data is of interest to others as well. Unsurprisingly, personal information makes an attractive target for cyber thieves looking for personal data to use themselves or trade in the darker corners of the web. Many users are too trusting in what is requested from them and tend to provide truthful answers and not question the relevance of what they are being asked for. Indeed, asking users for information is one of the easiest approaches to tracking them. Banking trojans - malicious software or web sites used to steal bank details - are a prime example of this. The authors of such malicious software found a productive approach was to simply inject a number of fields into a web page asking for additional information, such as account number and password. As the new fields did not look out of place in the web page, many users completed the fields and clicked the submit button without a second thought. The lines between web tracking and information security are starting to blur now, so how does this example relate to web tracking? Many attacks have been found to use some form of web tracking to identify individual users or particular versions of software that can be exploited to install some malicious code on the user's device, therefore increasing the chances of success.

Me want cookie! Me eat cookie! Don't be a cookie monster

There are two common approaches to web tracking: stateful and stateless.

From a technical standpoint, tracking users is quite easy. For the sake of this article we will split the techniques into two broad categories: those that store data, generally known as stateful, and those that do not, known as stateless. The most common stateful technique is the use of cookies. Cookies are small text files stored by web browsers to save data between visits to the web page. Web sites use cookies to save information about you, such as your preference settings, or to keep you logged in. In order to do this, a unique value is written to the cookie to identify you. This may be a random, anonymous identifier, or something like an email address or telephone number depending on the information the web site requested as part of the signup process. The cookie is set when you first visit the site, and is sent back to the web server with each request for a resource such as an image or web page, thus identifying you in the request and allowing you to be tracked. For tracking, cookies become a problem when they are set by and exchanged with third parties as this is seen by many as a privacy concern.

Stateful tracking...

... stores data on your machine to track you, such as cookies.

As a contrived example, imagine a third party that provides a general purpose weather widget for web sites. To show the widget on a web site, the site owner adds a small piece of code to load the widget when the web page loads. When the user requests the web page containing the widget for the first time, the widget sets a third party cookie to identify this user. The user later visits more web pages that embed the weather widget. The third party cookie is exchanged each time the widget loads, which provides the widget owner with a history of the web sites the user has visited. The widget owner has tracked the user across all the

sites they visited that contain the weather widget. Social media widgets, such as those that allow the user to like an article or social post, work this way and allow the tracker to properly identify the user by associating the cookie with the user's account. There are many other approaches and nuances related to cookies and stateful tracking, such as highly-persistent cookies that can prove almost impossible to remove, cookies that can come back to life after being deleted, and how internet security policies can be circumvented to allow information to be shared between unrelated organisations. However, the above should be sufficient to provide an understanding of the basic concepts of how stateful web tracking is carried out.

Your (device) fingerprint: the stateless boarder controller

Stateless techniques ...

... create a unique value to identify your device based on its configuration.

Stateless techniques adopt a slightly different approach to tracking by aiming to identify users or devices without needing to store information on client devices. One of the most effective methods for this, and the one we will look at here, is commonly known as device fingerprinting. A device fingerprint is simply a unique value, known as a hash, calculated based on a device's configuration. Now, I appreciate that sentence is slightly abstract, so let's revisit the weather widget example we used in the previous section for a clearer explanation. As we know, the stateful version of the weather widget identified users by storing and exchanging a cookie between the client's browser and the web server. By contrast, when you visit a web page containing the new, stateless tracking version of the weather widget, the widget downloads a fingerprinting script to your device instead of a cookie. The fingerprinting script gathers information about the device, such as the name and version of your browsers software and operating system, details of country and timezone, and even the names of the fonts installed in the browser. All of this information is crunched up and used to produce a unique hash value to identify the device. The fingerprinting script sends the resulting hash value back to the web server, along with the details of the web page hosting the widget. As you browse other web pages containing the widget, this mini-audit process repeats and produces the same hash value for your device. Again, this is sent back to the server with the details of the web page you visited, allowing the widget owner to build a browsing profile associated with your device identified by the unique hash value. Some techniques used by fingerprinting can reach deep into the device hardware making this technique surprisingly effective, even after the device's configuration changes. Device fingerprinting is not the only technique available for stateless tracking. Behavioural biometrics use the biometric traits of the user for identification, such as your typing rhythm and how you interact with the computer's mouse. Sadly, we do not have enough space to cover behavioural biometrics here.

First line of defence: the web browser

Accessible defences

- Browser configurations.
- Browser extensions.

Web tracking has been used almost since the inception of the internet, but the perceived privacy impacts of tracking are by comparison a much more recent concern. This has inspired some web users to completely change their habits and actively try to prevent their online habits being tracked. Just to clarify, we're talking about people wishing to protect their private data and information from being harvested and shared by organisations not known to them, and not about individuals involved in criminal or illegal activities. The most accessible defences concentrate on your web browser, either in the form of configuration or by installing third party extensions. All four of the main web browsers, Microsoft Edge, Google Chrome, Mozilla Firefox and Apple Safari, provide fine grained control over configuration along with some form of tracking protection and private browsing mode. All four also provide the means to extend your browser's functionality by installing small, specialised software modules known as browser extensions.

Popular privacy extensions include Ghostery, Disconnect, uBlock Origin, Privacy Badger and NoScript. At a high level, most privacy extensions are based either on a list, an ability to manipulate the web page being loaded, or learning the difference between good and bad web requests. List based extensions check requested web addresses against a list of known trackers, known as a Tracking Protection List (TPL), and deny requests to those found to be on the list. Extensions based on manipulating your browser either alter the web page as it loads or disable functionality to make your browser more secure. Altering the page as it loads allows the extension to remove potentially harmful elements before they can take effect. Heuristic based extensions learn the difference between good and bad elements of a page by using machine learning algorithms and therefore require a training period to become really effective. Theoretically at least, heuristic approaches have the advantage of being able to identify new and changing approaches to tracking as they emerge, potentially making for an extremely effective defence.

Starting with the browser is the most practical approach for most users. The act of changing browser configuration or installing a privacy extension requires little technical knowledge and is not particularly time consuming. If you are more technically inclined, potential defences are endless. There is a great deal more to defending against web tracking than can be covered here. For example, the effectiveness of a defence such as Private Browsing mode can differ between browsers, while the use of particular technologies such as Tor could potentially result in unwanted attention from the authorities.

Isn't installing a browser extension enough?!

Having multiple options at your disposal can often complicate matters. For example, how do you know which configuration to tweak or browser extension to install? How do you know what you are defending against, or which defences will meet your needs and if they will continue to do so? Earlier we introduced stateful and stateless tracking techniques, but we did not look closely at how they are normally implemented. Entire industries rely on web tracking for revenue generation so it's in the tracker's interest to ensure their approach continues to work under as many varying circumstances as possible. One way this is achieved is by using a multi-layered approach to implementation by combining different tracking techniques.

Trying to defend against such comprehensive approaches can range from easy to almost impossible, depending on what you're trying to achieve. For example, if all you wish for is a speedier internet experience then simply installing a privacy extension will reduce the number of interactions with third parties and therefore help reduce the time web pages take to load. This will also reduce tracking, but not protect against it. At the opposite end of the scale, to completely prevent being tracked and approach anonymity online is practically impossible without deep technical expertise and a considerable amount of time for conducting research, both of which are often scarce. To regain or preserve our privacy, most of us will aim to try and prevent being tracked as much as possible, and this is where the difficulties begin. To make this more manageable, lets look at three key areas: knowledge, implementation and verification.

How do you know you are actually being tracked? A level of knowledge is necessary to be able to identify tracking taking place in the first instance. This initial question naturally leads to more: how do you know which techniques are being used? Or, which extensions defend against which forms of tracking? Or, even that the extensions you install are developed by legitimate organisations and not trackers themselves? Once again, the lines between web tracking and information security are starting to blur. Answering these questions requires an investment in time and effort to undertake research to gain enough knowledge to just get started. Sadly, it doesn't end there. Web tracking evolves as rapidly as technology itself, and staying abreast of the latest techniques requires ongoing reading and research. Shortcuts are few and far between. The academic community contains some exceptional work in this area, making knowledge and experience available without the need to gain it first hand by conducting your own experiments. However, this does not replace the need to locate, read, understand and sometimes verify others' research before making your own decisions on it.

Hopefully, conducting some research has paid off and allowed you to make decisions as to which defences to try and implement. Reducing tracking can be achieved in many ways, and in some ways

is quite easy. Examples are to disable JavaScript, disable particular request headers, disable cookies and restrict or prevent requests to third parties, all effective in their own right. Unfortunately, they are also often debilitating or destructive in some form and likely to leave the visited web page heavily damaged or completely paralysed. The point here is that tracking needs to be considered in a particular context, in this case usability. JavaScript is a good example: the internet relies heavily upon JavaScript to provide rich user interaction with web pages, such as slick animations on menus and buttons, as do device fingerprinters for accurately identifying and tracking devices. This shows how having one without the other becomes extremely difficult and results in unexpected side effects, such as reduced usability of the web page. To make our case more complex, multi-layered approaches to tracking would take the lack of JavaScript into account and simply add a tracking pixel to the page to ensure tracking continued.

The final difficulty we will look at is verification. Verification relates directly to the other two topics as there is a need to verify the knowledge you have gained is correct, and also to verify the defences you have implemented are working as expected. Both of these can be tricky, particularly the second one. In my thesis, I opted for a technical approach to verification by conducting experiments to test extensions against different criteria, but appreciate such an approach may not be fitting for all users. This brings us full circle as knowledge and research are once again required to understand what needs to be verified, how to do it and whether the outcomes are acceptable. However, the one thing we do know is that as the tracking landscape evolves and new techniques emerge, failing to verify the effectiveness of existing defences could become a recipe for disaster!

Practically:

- Think about what you are trying to achieve
- If you can, research as much as possible - after all, knowledge is power!
- To help reduce tracking, start with your web browser
- There are many privacy extensions available for all the core web browsers
- When online, think about the information you are being asked for, and if you need to be truthful
- Try and verify your chosen defences work as you expect
- Remember, good defences today may not be good defences tomorrow

Final thoughts

I hope this article has introduced some of the difficulties inherent in defending against being tracked online. My intention was to move away from adopting defences based upon others' recommendations and move towards building your own knowledge, understanding and verification to provide you with a better understanding of what you are trying to defend against and how well you can achieve it. I've highlighted how tracking is often implemented using multiple layers and how this requires a defence-in-depth response. I have also showed that technical implementations without sufficient knowledge and understanding are unlikely to be, or remain, truly effective. Rather than trying to prevent web tracking as much as possible, perhaps we should strive to prevent it as much as *practical*, and instead decide how we wish to compromise on what we cannot control!

Biographies

Darrell Newman has recently completed his MSc in Information Security, graduating with distinction from Royal Holloway University of London. Darrell also won the David Lindsay Prize for his thesis, "Evaluating the effectiveness of defences to web tracking", awarded by the British Computer Society. In addition to an MSc, Darrell has a BSc in Computing and extensive knowledge of Software Engineering with over fifteen years development experience working primarily in the financial sector.

Geraint Price BSc (London), PhD (Cantab) obtained his B.Sc. in Computer Science from Royal Holloway University of London in 1994 and his Ph.D. from University of Cambridge in 1999. His Ph.D. dissertation analysed the interaction between Computer Security and Fault Tolerance. Since then he has worked on various projects including Denial of Service attacks in networks and the future of Public Key Infrastructures, funded by academia and industry. Geraint is a Senior Lecturer in the Information Security Group, and has a strong interest in the practice of information security. He leads the ISG's external engagement activities with business and government. Geraint is a regular attendee, panellist and speaker at a number of industrial fora, including I-4 and the ISF.

Series editor: S.- L. Ng