



SURVEILLANCE SELF-DEFENSE

< < [FURTHER LEARNING](#)

What Is Fingerprinting?

Last Reviewed: August 27, 2024

Digital fingerprinting is the process where a remote site or service gathers little bits of information about a user's machine, and puts those pieces together to form a unique picture, or "[fingerprint](#) ⓘ," of the user's device. The two main forms are browser fingerprinting, where this information is delivered through the browser when a user visits remote sites, and device fingerprinting, when the information is delivered through apps a user has installed on their device.

In most cases, fingerprinting is performed by a third-party rather than directly by the site someone is visiting or the app someone is using. As an individual uses their device, a specific third-party tracker may be loaded on multiple apps installed or sites visited. This allows that company to track an individual across their usage of multiple sites they visit or apps they have installed. These trackers get unprecedented insight into the daily activities of the user, including information that is often specific enough to know what a user is doing at any moment and even where they are using their

device.

Fingerprinting and Retargeting

Fingerprinting is done extensively by tracking companies, who use this information to target users with ads or sell that information to [data](#) ⓘ brokers. Digital advertising is a business worth [hundreds of billions of dollars](#). Retargeting, or recognizing a return visitor and marketing materials based on their previous browsing, is a powerful way for marketers to [increase click-through rates](#) and generate revenue. Fingerprinting may also be used for fraud and bot detection, using the same technology that trackers use for legitimate ends.

The most common way to retarget ads on the web is through browser [cookies](#) ⓘ, and apps can use the advertiser ID provided by both iOS and Android. But the user can clear cookies and [advertiser IDs to remove the persistent identifier](#) linked to their particular browser or mobile device. Think of your browsing habits as a string that connects different pins on a board. Each pin represents a site that you've visited, and a tracker can trace that string to see what you've visited in the past. Clearing your cookies is like cutting that string into different segments, and the more often you cut the string, the smaller the segments become. The tracker can no longer see what you've visited in the past.

Fingerprinting creates a string that can't be cut by creating a new persistent identifier. It uses your browser or device characteristics against you, since the identifier is a summary of all the characteristics of your browser or device. Fingerprinting can use all sorts of seemingly mundane details about your device or browser, such as screen resolution, your time zone, [operating system](#) ⓘ version, remaining battery life, and more. The reason why fingerprinting exists is to circumvent the normal controls users have that enable them to control their own browsers. In order to take control of our browsers and devices back, we have to use special tools that resist

fingerprinting.

Effectiveness of Fingerprinting as a Tracking Technique

In order for fingerprinting to be effective for trackers, it has to meet two criteria.

First, it has to be persistent. If the user's fingerprint changes rapidly, there would be no way for the tracker to tell one visit of a user from the next visit. This ability to link visits is essential to determine that it is the same user visiting websites or using apps over time. This persistent identifier is used as a substitute for a cookie, which can be easily deleted by the user. A fingerprint can not be removed, since it does not store anything on the users' machine.

Second, it has to be unique. If two or more users have the same fingerprint, the tracker loses the ability to identify a single individual using fingerprinting. Without this ability, the tracker can not track the user individually and place them in specific marketing categories, such as "baking hobbyist" or "aircraft enthusiast." In our [Cover Your Tracks](#) (previously known as Panopticlick) study of user browsers, launched in 2010, we discovered that the vast majority of browsers satisfied these two criteria.

Within mobile apps, fingerprinting can also grab all sorts of data about your device, ranging from the last time you reboot your phone to what other apps are installed. While it's difficult to enforce, this isn't always allowed. For example, Apple [requires developers](#) to explain why their apps may [need device details often used for fingerprinting](#), but since many of these details have legitimate uses, it's [not always possible to tell if a developer](#) is using the collected data for functionality or for tracking. On Android, many characteristics used for fingerprinting require the app to request specific

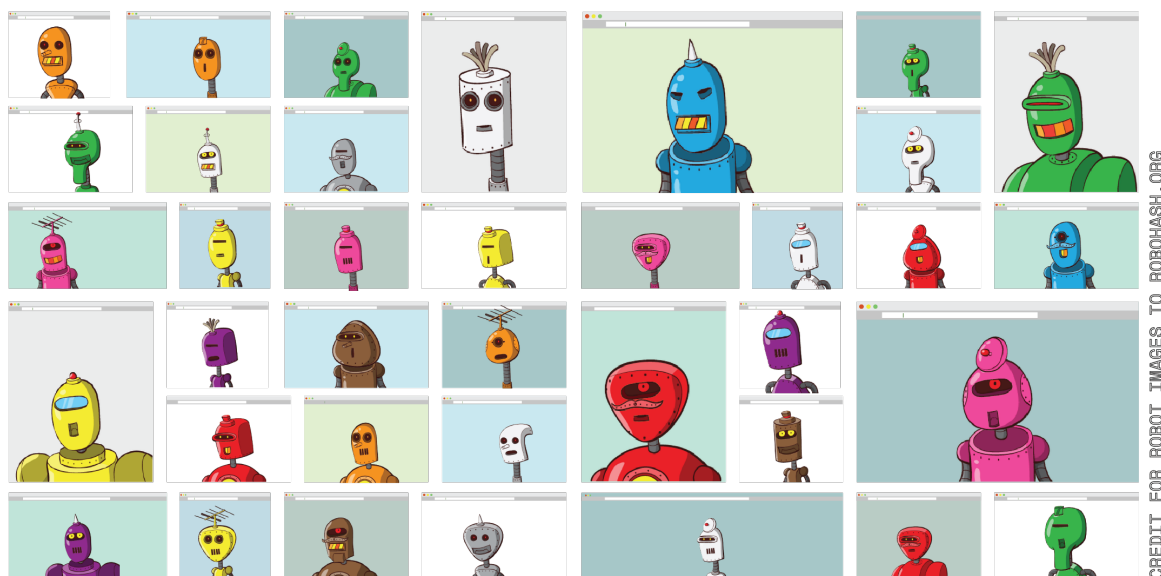
permissions, but a manual review of these permissions is often inscrutable and ignored in favor of simply clicking “allow.” In iOS and Android, it is recommended to limit the number of apps you download and ask if an app really requires the privileges it asks for.

Counter-Fingerprinting Strategies

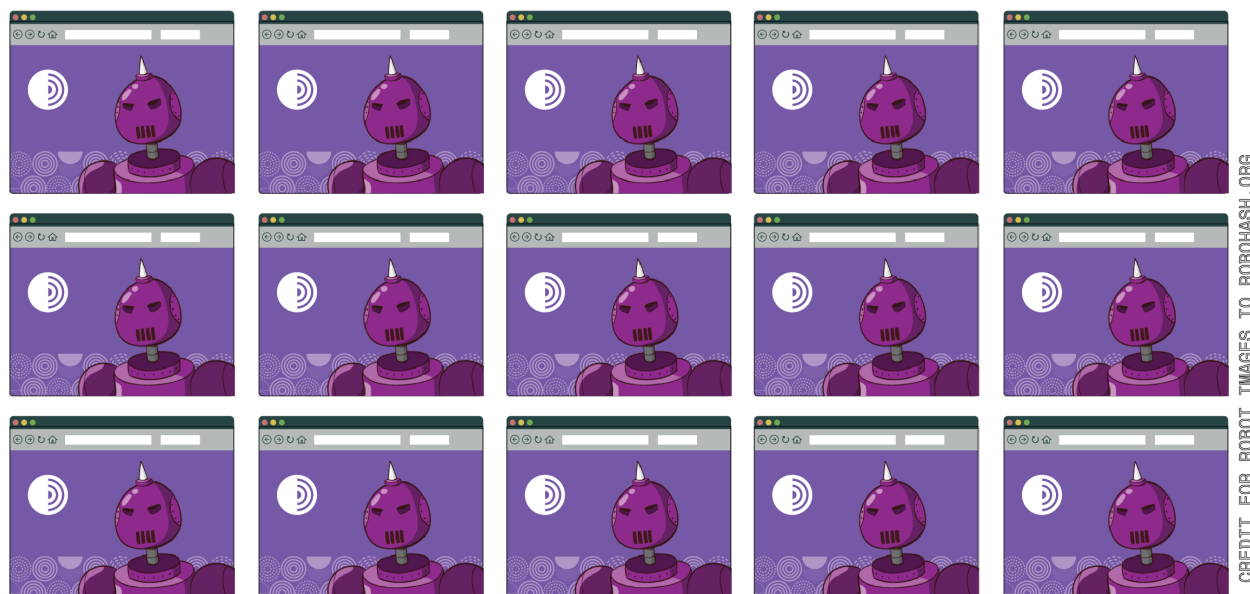
In order for tools for combatting fingerprinting to be effective, they can follow one of two strategies. First, they can attempt to remove one or both of the two criteria above that make fingerprinting effective. Second, they can develop a list of trackers and block each of them from loading in the browser or on a mobile device.

There are many tools that attempt to break the persistence of fingerprints in the browser. Some will attempt to randomize the results of certain characteristics, such as a [canvas fingerprint](#) and [AudioContext fingerprinting](#). This can be an effective method for breaking persistence, but it is important to note that a tracker may be able to determine that a randomization tool is being used, which can itself be a fingerprinting characteristic. Careful thought has to go into how randomizing fingerprinting characteristics will or will not be effective in combating trackers.

Browsers can also combat fingerprinting by making all instances of the browser look the same. By making the fingerprinting characteristics on all instances of browsers the same, a particular instance of the browser can not be uniquely pinpointed. This is the method adopted by the anonymity tool [Tor Browser](#).




Most browsers look unique, and trackers can follow them around the web.



Tor Browsers look the same, so trackers can't fingerprint them!

This is highly effective against fingerprinting when done right. Tor Browser has identified dozens of places where work needed to be done to make all their browsers look the same across all characteristics. This is important

because it is easy for a user who is changing individual settings with the intention of throwing off trackers to actually make their browser easier for trackers to identify.

Finally, a tool can develop a list of trackers and block them directly. This is the method employed by many browser add-ons or extensions, such as EFF's own [Privacy Badger](#). By [blocking](#)  trackers, these tools are able to [remove the bulk of the fingerprinting trackers](#) from being loaded in the browser. Third-party trackers, which are the majority of the ones that use fingerprinting to identify users, are not able to identify user browsers.

Though this is a highly effective method to block fingerprinters, it does not completely block the ability to do fingerprinting. Sneakier trackers which haven't been identified yet, or fingerprinting done directly by a site the user is visiting rather than by a third-party, will most commonly be permitted. Blocking known fingerprinters is good enough for most use cases, but does not guarantee strong anonymity.

Don't Customize Your Own Settings to Combat Fingerprinting

Customizing settings on your own to combat fingerprinting rarely results in the intended outcome.

For instance, you might be inclined to change the user-agent string, which identifies the browser and version you are using, to the most common browser user-agent string used across the web. This makes some intuitive sense: using a common user-agent string will result in a more common and less trackable fingerprint, right? In some cases, however, using the most common user-agent string will make you more fingerprintable, not less. This is counter-intuitive: how could choosing a more common metric make one stick out more?

This comes down to how independent your user-agent string is from the other fingerprintable metrics in your browser. For instance, Safari on iOS is actually a fairly non-fingerprintable browser, due to the relative similarity of hardware, software, and drivers across different devices. Most users of Safari for iOS look relatively similar.

But Safari for iOS is not the most common user-agent on the web by a long shot. Let's suppose for the moment that the latest version of Chrome for Windows is the most common user-agent. If the user were to change the user-agent string in Safari for iOS to Chrome for Windows, without changing anything else, they would appear completely unique. They would be the only one who has Safari for iOS results for canvas fingerprinting that also has the Chrome for Windows user-agent.

That's why those that are trying to avoid fingerprinting have to be extra careful that they don't accidentally hurt their privacy instead of helping it. In order to blend in, you have to join a "privacy pool" of other users that have the exact same fingerprint as you across all metrics. To do this, the safest bet is not to change settings individually, but instead to choose something like Tor Browser, Brave, or Firefox which use techniques to make all instances of their browser relatively common.



SURVEILLANCE
SELF-DEFENSE

ABOUT INDEX GLOSSARY CREDITS

DONATE

COPYRIGHT (CC BY)

PRIVACY