March 19, 2014

Federal Trade Commission, Office of the Secretary
Room H-113 (Annex J)
600 Pennsylvania Avenue, NW
Washington, DC 20580

Re: **Spring Privacy Series: Mobile Device Tracking, Project No. P145401**
Comments of the Electronic Frontier Foundation

Dear Ms. Koulousias and Ms. Anderson:

Thank you for the opportunity to submit comments for this workshop, as well as the opportunity to participate in person.

Summary

Mobile devices and other technical artifacts are now easy and cheap to track. Location tracking is a particularly invasive and risky form of tracking because it has almost unbounded potential to reveal sensitive private facts about individuals' lives across many different domains. The technical means of tracking are usually almost accidental—they are not typically design goals of the technologies in question, nor are they features that consumers want or even know about. Instead, tracking technologies are usually unintended consequences of technical design choices. We can articulate avoidance of tracking capabilities as a desirable property that technical systems ought to have in order to avoid unintentionally revealing information about their owners' whereabouts.

Existing commercial analytics systems that use location tracking of mobile devices are only the tip of the iceberg because of their potential to be combined with other data sources to answer more kinds of questions about individuals and to merge online and offline activities into a single profile. Although some data practices are dramatically less invasive than others, we should not lose sight of the underlying reality that location tracking is commonly nonconsensual and the result of piggybacking on unintended features of technologies. Services that make use of information about individuals' locations can be implemented in more privacy-protective ways, and in ways that give individuals ultimate control over who knows their location and for what purposes.

The sensitivity of personal location information

Location privacy, particularly against systems that can monitor a person's location automatically over a long period of time, is a crucial aspect of personal privacy. A person's whereabouts and movements are sensitive not only in themselves but for all of the other kinds of sensitive information that they directly and indirectly reveal. An analysis of location data about a person may reveal or implicate

- intimate relationships
- medical privacy
- religious beliefs and practices

- legally privileged interactions, such as attorney-client relationships
- sexual activities and interests
- associational privacy and social networks
- attendance at events, including protests[1]
- physical safety, such as when someone is a victim of stalking

Concretely, location data can answer questions like: Do these people know each other? Are these people having an affair? Was this person speeding on their commute? Does this person frequent strip clubs or bars? Who attended this protest? (Who came from another city in order to attend this protest? How long did they spend in this city?) What place of worship does this person attend? Have these people stopped dating recently? Has this person visited an abortion clinic or an oncologist? How did this person's habits change—or not change—after they got married or divorced?

At first glance, location data gathered by observing the proximity of a device appears to be anonymous because it will refer to a device identifier rather than to a person's name. Some people developing tracking systems have emphasized the idea that they are tracking devices, not people. But in practice the distinction is tenuous. Research supports the proposition that it's relatively easy to deanonymize such data; for example, the combination of a person's home and workplace is likely to be completely unique, and so observing that a particular device spends the night in a certain location and the work day in another is virtually always enough to identify an individual.[2] Once the anonymity of data is lost, it can't be restored easily, so the true level of uncertainty about the person or people associated with a device tends to go down over time. As we discuss below, different databases may record facts that would be particularly intrusive in combination with one another. What's more, some analytics providers may actively seek to identify device owners by various means, and concerted efforts to do so are likely to succeed.

It's easy for people to undervalue their location privacy in the short term. Human beings are naturally forgetful, and we often can't remember clearly where we were a week or a month ago, let alone perceive patterns in our activity or envision how those patterns could become interesting to someone else. Machines are less forgetful and can easily retain facts about a person's whereabouts from months or years ago that the person has completely forgotten. Location privacy may also become more important in retrospect, such as if a person runs for office, acquires a stalker, or becomes embroiled in litigation.

The spread of trackable devices

Devices increasingly determine and make use of awareness of their own locations using positioning technologies such as GPS. There are significant privacy concerns about the uses to which this

---

1 See Sean Hollister, “A Lesson From Ukraine On Cell Phone Metadata,” *WBUR Here & Now*, January 24, 2014, available at <http://hereandnow.wbur.org/2014/01/24/ukraine-metadata-lesson> (describing Ukrainian government's use of cell phone location records to determine who was present at a “mass disturbance”); BitsBytesRights Blog, “You Were Identified As a Participant in a Mass Disturbance,” January 27, 2014, available at <http://bitsbytesrights.org/you-were-identified-as-a-participant-in-a-mass-disturbance/> (showing text message sent by government to protesters' phones).

2 See Philippe Golle and Kurt Partridge, “On the Anonymity of Home/Work Location Pairs,” available at<https://crypto.stanford.edu/~pgolle/papers/commute.pdf> (describing the degree of uniqueness of the combination of a person's home and place of work at varying levels of precision of measurement, from county to census block); Arvind Narayanan, “Your Morning Commute is Unique,” available at <http://33bits.org/2009/05/13/your-morning-commute-is-unique-on-the-anonymity-of-homework-location-pairs/> (analyzing this research).

information is put, and about mobile device users' ability to make sure software they use won't collect and report this information in ways they don't want.

But for the purposes of the present workshop, we're most concerned with the ways in which devices can be recognized from the outside—observing when a particular device enters and leaves a particular vicinity, or even determining its fine-grained location from moment to moment. Technologies that recognize devices from the outside are particularly concerning because they can often operate passively, surreptitiously, and without any meaningful ability for a device owner to know that tracking is taking place. We can expect location monitoring to be enabled by a wide variety of technologies:

- Payment systems, including contactless credit card and electronic payment systems and electronic fare and toll collection systems[3]
- RFID tags for inventory control, theft deterrence, or physical access control
- Communication devices such as mobile phones, laptops, and tablets (via their self-reported positions or via monitoring transmissions from wifi, Bluetooth, GSM, or CDMA radio interfaces, among other possibilities)

Tracking can be performed by actual infrastructure operators[4], by peer devices[5], by those impersonating an infrastructure operator[6], or by a third party merely listening to radio communications[7]. Each of these represents a potential threat to user privacy.

Mobile device tracking for commercial purposes today is mostly based on wifi or Bluetooth radios, and, in both cases, on observing a device's MAC address. The MAC address, an abbreviation for *media access control address*, is a unique 48-bit number (that is, a number between 0 and 281474976710655) that was programmed into a network interface by the manufacturer at the time of manufacturer, and that distinguishes one device from another on the network.[8] Most devices do not include a way for the user to change the MAC address, so it's likely to be the same for the lifetime of the device. The MAC address is included in *every* transmission made by the associated radio transmitter, and is not scrambled or encrypted, so it can be read by anyone in the vicinity, not just the intended recipient or base station. Because MAC addresses are also transmitted by devices within *probe packets* that attempt to determine whether specific networks are in range, they can be observed even when a device is not actively connected to any wifi network. Devices cannot detect whether their MAC addresses are being observed or recorded. The effect is that a MAC address on a wifi device is somewhat akin to a license plate on a car—yet a license plate that can be read from any angle, and even through walls!

Commercial location tracking mostly relies on the infrastructure case (wifi operators, at large or small scale, and electronic payment systems, observe when a device with a particular MAC address is active

3 See Andrew J. Blumberg and Peter Eckersley, “On Locational Privacy and How to Avoid Losing it Forever,” August 2009, available at <https://www.eff.org/wp/locational-privacy> (describing the privacy-invasive defaults of automated toll and fare collection systems and presenting privacy-protective technical alternatives).

4 For example, a wifi base station operator; a mobile network operator; a toll plaza; an electronic door lock; a retail point-of-sale terminal that supports a wireless electronic payment method.

5 For example, a laptop with a wifi or Bluetooth interface.

6 For example, an *IMSI catcher* (a device which impersonates a cellular base station) or a *Wifi Pineapple* (a device which impersonates whichever wifi base station is requested by a client device).

7 For example, a laptop with a wifi interface set to *monitor mode*, in which the device will receive wireless traffic regardless of the wireless network ID with which the traffic is associated.

8 A MAC address is usually written as six pairs of hexadecimal digits (0-9 and a-f) separated by colons, like 0a:4b:b6:eb:8c:29. We can also say that it consists of six bytes, or 48 bits, of numeric data.

at a particular location—and may well learn the owner's identity through a registration process) or on the third-party case (listening to actual or attempted wifi communications with others and observing the source device's MAC address). Most devices don't provide an easy way for users to resist or avoid either sort of location tracking, as long as the wireless interfaces on the device are active.

The problem of unique identifiers

There are many properties that might make a device unique—ranging from the cookies it's received from web sites[9] to the versions of the software and even fonts it has installed[10] to the distinctive physical characteristics of its hardware.[11] The panoply of potential tracking methods, as well as others yet to be documented by privacy researchers, suggests that trackability is ubiquitous and that mitigating it is challenging. By far the simplest, cheapest, and most reliable way of tracking a device, however, is when the device is *engineered to transmit* a unique and unchanging hardware identifier in a way that can readily be read by others. (MAC addresses and IMEI addresses are among the most important examples of such identifiers in mobile phones, laptops, and tablets.)

When devices contain persistent unique identifiers that are readable by the public, the ability of other parties to engage in location tracking is automatic; they need only create a sensor that observes and records the proximity of devices and their identifiers.[12] Such tracking is especially reliable because there is virtually no uncertainty about the identity of a device, at least if users don't have access to countermeasures that change the identifiers that their devices transmit.

Trackability: unintentional, undesirable, unnecessary

Trackability is often an unintended consequence of technology design. It should be viewed as a design defect because it allows nonconsensual and surreptitious tracking, with the potential to reveal detailed information on the user's whereabouts, activities, and associations against the user's will.

Historically, unique identifiers were often created for engineering purposes that had nothing to do with tracking human beings' movements or whereabouts. For example, unique device hardware addresses were often created for disambiguation purposes for environments in which many devices can be present, or for authentication purposes.[13] The history of the MAC address is instructive. The use of

---

9 See Adi Kamdar, Rainey Reitman, and Seth Schoen, "NSA Turns Cookies (And More) Into Surveillance Beacons," *Deep Links*, December 11, 2013, available at <https://www.eff.org/deeplinks/2013/12/nsa-turns-cookies-and-more-surveillance-beacons> (summarizing *Washington Post* reporting on intelligence agencies' ability to recognize devices and users by observing distinctive identifiers, including advertising cookies, within web traffic).

10 See Peter Eckersley, "How Unique is Your Browser?", Proceedings of the Privacy Enhancing Technologies Symposium (PETS 2010), available at <https://panopticlick.eff.org/browser-uniqueness.pdf> (describing Panopticlick, a system that distinguishes users' browsers by measuring properties such as installed software versions and locally-installed fonts).

11 See J. Lukas, J. Fridrich, and M. Goljan, "Digital Camera Identification from Sensor Pattern Noise," IEEE *Transactions on Information Forensics and Security*, November 2006, p. 205 (proposing forensic means of distinguishing digital photographs' origins based on noise artifacts created by individual cameras' optical sensors); Jakob Hasse, Thomas Gloe, and Martin Beck, "Forensic Identification of GSM Mobile Phones," available at <http://www.dence.de/theme/Cakestrap/doc/Hasse13_GSMMobilePhoneIdentification.pdf> (proposing "a novel method to identify GSM devices based on physical characteristics of the radio frequency hardware" visible in the radio waves emitted by a device).

12 See Seth Schoen, "Location Tracking: A Pervasive Problem in Modern Technology," *Deep Links*, December 11, 2013, available at <https://www.eff.org/deeplinks/2013/12/location-tracking-pervasive-problem-modern-technology> (arguing that location tracking is inevitable when persistent unique device identifiers are transmitted unencrypted).

13 Identifiers that are used for disambiguation purposes often don't need to be persistent (they could change every time a device joins a new network or new environment); identifiers that are used for authentication purposes often don't need

MAC addresses in wifi was adopted wholesale from the MAC address used for wired Ethernet networks. The MAC address was originally invented by Xerox as part of its development of the Ethernet technology in the 1970s; the purpose of the MAC address was to allow computers ("stations") to determine whether a packet of data broadcast on the Ethernet was relevant to them.[14]

At the time of its development, the Ethernet standard was designed for use on wired networks within office environments; it was not seen as relevant to devices meant to be carried around by individuals. Ethernet inventor Robert Metcalfe describes, only a few years prior to his creation of the first Ethernet network, "having the first computer small enough to be stolen":

> I came to work one day at MIT and the computer had been stolen. […] I called up DEC [the manufacturer of the computer] to break the news to them that this $30,000 computer that they'd lent me was gone. […] When I called them, I got a really funny reaction because they sent out the marketing person who'd loaned me the computer. He came with two public relations people. And they thought this was the greatest thing that ever happened. Because it turns out I had in my possession the first computer small enough to be stolen! And they made a big PR deal out of this! […] So I may be famous for having the first computer small enough to be stolen.[15]

Ethernet networks in this era were wired networks containing computers that were not particularly portable, so there was no great uncertainty about the whereabouts of devices (nor likelihood of using devices' movements to track those of human beings). But pressures for backwards-compatibility across multiple generations of wired and later wireless Ethernet technology have preserved the format, length, and uses of the MAC address while adding a previously unforeseen use: watching for the proximity of a mobile device.

The original Ethernet design called for worldwide uniqueness of each device identifier, without anticipating that those identifiers would be visible to strangers (or that networks could be unfriendly or include malicious participants[16]). In 1980, it was most cost-effective to burn in a unique ID for each Ethernet device at the time of manufacture; today, it would be trivial to choose a new random identifier (from a larger set of numbers) each time a device connects to a network. But the legacy of the decades-old design has ended up creating a means of tracking people's whereabouts.

This history is not unique. The simplest path for creating any new communications protocol is typically to include a unique identifier for each device or user, and to allow that device to be transmitted and read by anyone. That means that privacy invasion is the *technical default* unless engineers make a deliberate effort to avoid it. But efforts to avoid tracking can bear fruit—if it's explicitly adopted as a goal and a design requirement.

---

to be readable by the public (they could use cryptographic security means to allow authentication information to be available only to the appropriate party, and not to the general public).

14 See Digital Equipment Corporation, Intel Corporation, and Xerox Corporation, "The Ethernet: A Local-Area Network," September 30, 1980, available at <http://ethernethistory.typepad.com/papers/EthernetSpec.pdf>, section 4.2.2 ("Receive Data Decapsulation checks the frame's destination address field to decide whether the frame should be received by this station") and 6.4.1.2.1 ("The data link controller recognizes and accepts any frame whose destination field contains the physical address of the station").

15 Robert Metcalfe, interview with Joyce Gemperlein and Trevor Getsla, available at <http://www.thetech.org/exhibits/online/revolution/metcalfe/>. Metcalfe invented the Ethernet technology at Xerox just five years after this first-ever computer theft.

16 "Non-Goals: […] There is no attempt to protect the network from a malicious user at the data link level." ("The Ethernet: A Local-Area Network," at section 3.2.)

Proponents of location-tracking can point to ways in which technologies apply people's location data to provide useful services. Familiar examples include satellite navigation, as well as systems that gather data from passing vehicles in order to provide real-time analysis of traffic conditions. We agree that an awareness of location is useful for providing services[17], but it should be an engineering priority to limit, by technical means, the disclosure of the location of device to those situations in which the device's owner has given affirmative, informed consent. *Devices should not be recognizable and trackable under other circumstances*; the ability to detect a device's proximity or whereabouts without explicit consent should be viewed as a security flaw.

Where revealing some information about their physical whereabouts is useful to consumers, as it surely sometimes can be, the ability to opt in to this disclosure by installing certain software is clearly feasible.[18] Indeed, it is already common: a wide range of smartphone apps now collect and transmit location information for specific and, ideally, well-defined and limited purposes.

At least with respect to wifi MAC address tracking, technical remedies to prevent nonconsensual tracking of a device are available. Mobile device manufacturers and operating system developers should make wifi MAC addresses change automatically over time, or give users a simple and straightforward way to change their MAC address and to choose circumstances under which the MAC address will automatically change. They should also consider disabling, or at least allowing users to disable, the transmission of wifi probe packets. New technologies be designed not to transmit unencrypted persistent unique device identifiers.

Legal concerns

Existing commercial practices raise uncertain legal issues. For instance, under federal law it's generally illegal to record or decode "dialing, routing, addressing, or signaling information transmitted by an instrument or facility from which a wire or electronic communication is transmitted"[19] without a court order unless you're a provider of communications service and can fit into a statutory exception.[20]

We're unaware of any relevant case law, but some techniques of carrying out location tracking by capturing addressing data transmitted from smartphones or other wireless devices would seem to run afoul of this statute. Note, however, that this law doesn't have a private right of action, *i.e.*, it doesn't say that an ordinary person can sue someone under it.

Retail tracking could also raise issues under state law. If retail tracking does violate federal law, such violation might be used in combination with a state unfair business practices statute that provides a private right of action. Or such tracking might violate a state constitutional right to privacy. See, e.g., Cal. Const. art. I, § 1; *White v. Davis*, 13 Cal.3d 757 (1975). In *White*, the California Supreme Court explained that "the moving force" behind California's constitutional right to privacy "was a more

17 The engineering need for collecting fine-grained location data can be overstated. For example, some mapping applications download an entire set of map tiles for an urban area ahead of time. Then a user's mobile device can determine its own location and display it on the map without telling a service provider where it is—indeed, without even using a network connection.

18 The Commission has recognized, however, that consumers can be harmed by misleading commercial practices in this regard and can be tricked into installing software that tracks their location. See *In the Matter of Goldenshores Technologies, LLC*, File No. 132-3087 (adopting consent order to restrain allegedly deceptive practices by firm that marketed a flashlight app with unexpected location-tracking functionality).

19 18 USC § 3121(a) (prohibition); 18 USC § 3127(3) (defining "pen register").

20 18 USC § 3121(b).

focused privacy concern, relating to the accelerating encroachment on personal freedom and security caused by increased surveillance and data collection activity in contemporary society," and that its "primary purpose is to afford individuals some measure of protection against this most modern threat to personal privacy." *Id*. at 774.

Importantly, the California constitutional privacy right protects against private businesses as well as the government: It "prevents government and business interests from collecting and stockpiling unnecessary information about us," partly because "[t]he proliferation of government and business records over which we have no control limits our ability to control our personal lives." *Ibid*. Thus, among the "principal 'mischiefs'" targeted by the constitutional right are "the overbroad collection and retention of unnecessary personal information by government and business interests" and "the improper use of information properly obtained for a specific purpose, for example, the use of it for another purpose or the disclosure of it to some third party." *Id*. at 775.

Privacy and existing commercial practices

We don't believe that broadcasting the location of a device to anyone listening nearby is a feature that consumers would welcome. More broadly, we doubt consumers would expect to be tracked this way, commonly understand how technically straightforward it is, or would find it easy to opt-out. Since wifi tracking, at least, is undetectable and has extremely low barriers to entry, there is a high likelihood that there will always be some location tracking by parties who don't respect or participate in any particular opt-out mechanism.

Participants in the workshop strongly emphasized that their existing analytics applications work well with anonymous data and do not require trying to identify or profile the owner of a particular device, nor link observations of that device to other data sources. We accept that this is true of most of today's retail-oriented location analytics applications. However, we think the temptation to try to combine offline analytics with online analytics databases will prove irresistible in the long run. Firms are likely to conclude that online and offline databases are significantly more valuable in combination than apart.

Some questions posed to the workshop panelists asked how this could be done, or, more broadly, how the owner of a device might be identified. Ashkan Soltani's introductory presentation at the February workshop already offered one example, in the form of a mobile application that simultaneously collected and transmitted MAC addresses along with other personally-identifiable information. The operators of mobile hotspots are also in a position to create substantial databases of device MAC addresses paired with strongly identified online and legal identities. For example, firms that operate paid hotspot networks usually collect name and e-mail address as part of the payment process (and can also see the MAC address of devices that associate with their networks[21]); cable and mobile providers that offer wifi access for their subscribers will ask their subscribers to sign in with a credential linked to their billing information; and even firms like Facebook are now offering branded wifi routers in cafés and other businesses through partnerships with the business owners.[22] All of these businesses have the ability to make a direct observational association between a particular device and a particular strongly identified user.

---

21 MAC address data is used directly by software that authenticates paid users of a wifi network, such as a pay-per-use wireless hotspot. Hotspot providers thus typically possess this data, even where they did not intend to use it for location-tracking purposes.

22 See Jennifer Van Grove, "Why Facebook is giving out free Wi-Fi for check-ins," CNET.com, October 2, 2013, available at <http://news.cnet.com/8301-1023_3-57605745-93/why-facebook-is-giving-out-free-wi-fi-for-check-ins/> (describing how users are asked to sign in to participating wifi networks using a Facebook account and password).

We think that industry will not be able to resist the temptation to combine online and offline observations of user behavior and interactions when it has the technical ability to do so. We already see analytics firms helping web sites learn the specific "venue" from which a particular user (who might well already be personally identified by those sites) is connecting.[23] The incentive for making these sorts of associations is there—on all sides.

Hashing as a means of limiting the sensitivity of MAC address records

Some of the industry participants at the Commission's workshop referred to the use of *cryptographic hashing* by the location analytics industry and advertising industry. The industry views hashing as a privacy-protective technical measure that reduces the invasiveness or sensitivity of information collected about individuals' devices. Informal conversations outside of the workshop indicated to us that some in these industries view hashing as transforming certain records so that they no longer constitute personally-identifiable information (PII), or so that certain kinds of searches or inferences are "not possible."

For example, it appears that some people working in the advertising and mobile analytics industry believe that hashing obfuscates MAC addresses so that analytics firms can recognize particular devices again *when those devices are present*, yet cannot produce a list of all of the devices that were present in the past. To put this another way, the industry hopes to use hashing to be able to answer historical questions about specific devices ("Have we seen this particular device before? Where and when?"), yet not blanket, nonspecific inquiries ("Which mobile phones were present in this store on March 1, 2012?" "Who are all the customers who have ever visited our stores?"), and perhaps also render it more difficult to link location analytics databases with other, unrelated databases.

We emphasize that **naïve uses of hashing do not succeed in providing this kind of privacy protection**. Carefully applied, cryptographic hashing can be useful for achieving particular privacy goals; EFF's own CryptoLog software, for example, provides a way to keep web server logs of historical statistical value but whose individual entries can be made unlinkable across specified time windows[24]. (A web site that uses CryptoLog can arrange that it will become infeasible to determine whether visitor A and visitor B were the same individual if their visits took place within different time windows.) However, **merely hashing an identifier such as a MAC address with a standard hash function does not effectively obfuscate the input value**; such a hashing process can be quickly and cheaply reversed, recovering the original value and eliminating any purported privacy benefit.

Former FTC Chief Technologist Ed Felten explained the problem clearly in a 2012 blog post[25]. Dr. Felten explains that, even though there is no direct mathematical way to reverse a hash, an analyst can nonetheless easily recover a Social Security Number given its hash by trying every possibility.

> The analyst simply guesses my SSN–he enumerates all of the possible nine-digit SSNs and hashes each one. When he hashes my correct SSN, the result will be equal to the

23 See, *e.g.*, Skyhook Wireless, "Advertising," available at <http://www.skyhookwireless.com/advertising/> (describing geolocation services available to web site operators to recognize online visitors' whereabouts "down to a 100-meters radius and below" to help them make associations between "people, places, demographics, and time").

24 The CryptoLog code is available from <https://git.eff.org/?p=cryptolog.git;a=summary>. It relies on hashing visitors' data together with a secret value that is changed every day; since hashes of identical data only match when the same secret value was used, and prior days' secret values are deliberately forgotten, matching records can be identified within a given day but not between one day and another.

25 Ed Felten, "Does Hashing Make Data 'Anonymous'?", Tech@FTC Blog, April 22, 2012, available at <http://techatftc.wordpress.com/2012/04/22/does-hashing-make-data-anonymous/>.

> [obfuscated SSN], so he will know that he guessed right. You might think it would take a long time to run through all of the possible SSNs, but computers are very fast–there are "only" one billion possible SSNs, so your laptop can hash all of them in less time than it takes you to get a cup of coffee.
>
> A clever analyst would do it even faster. He would hash all of the possible SSNs in advance, and build an index that allowed him to recover the SSN from its corresponding hash value in the blink of an eye. Hashing the SSN would offer no protection at all against an analyst who had built such an index.
>
> It should be clear by this point that hashing an SSN does not render it anonymous. The same is true for any data field, unless it is much, much, much harder to guess than an SSN–and bear in mind that in practice the analyst who is doing the guessing might have access to other information about the person in question, to help guide his guessing.

An interesting question for us here is whether particular data fields are "much, much, much harder to guess than an SSN" and hence whether the obfuscation provided by hashing can be reversed in this way simply by guessing every possibility. Dr. Felten's view that one billion possibilities are not a large number for a computer is actually relatively modest, since the Graphics Processing Unit (GPU) in many modern desktop computers can be programmed to perform tens to hundreds of millions of hash operations per second.[26]

Data field type	Example value	How many possible values are there?	Seconds to test all possibilities at 800 Mhash/second	Time corresponds to
ZIP code	20580	5 digits = 10^5 = 10000	0.0000125	12.5 microseconds
SSN[27]	409-52-2002	9 digits = 10^9 = 1000000000	1.25	1.25 seconds
IP address	216.128.243.109	32 bits = 2^{32} = 4294967296	5.36	5.36 seconds

These estimates are unnecessarily pessimistic in practice, because, in each case, not every value is really possible. Not all ZIP code ranges have been assigned; not all SSN ranges have been assigned; not all IP address ranges have been assigned. Similarly, not all MAC address ranges have been assigned. The range of possibilities for a MAC address value first appears to be 48 bits (2^{48} possibilities), implying a naïve search time of a few days. But with knowledge of which numerical ranges do and do not occur in practice, we can drastically speed up the search. For MAC addresses, the first 24 bits (three bytes) are designated as a manufacturer ID, or Organizationally Unique Identifier (OUI). OUIs are issued to manufacturers of Ethernet devices by a central coordinating body. Although

26 Dr. Felten considered carrying out these hashing operations on a CPU, the microprocessor that controls most of the calculations and operations on a typical computer. It turns out that powerful GPU chips, while somewhat more complex to program and not included in every computer, are dramatically faster at operations of this sort, outperforming CPUs by factors of hundreds to thousands. Thus, where Dr. Felten expected the SSN hash deobfuscation to take the length of a coffee break, we estimate it at only one and a quarter seconds.

27 The example value is the SSN assigned to Elvis Presley, who died in 1977.

16777216 different OUI values are possible, only a tiny fraction of these have ever been assigned to manufacturers; at most about 26000 OUIs are known to have ever been used in practice. For instance, the OUI code C0:84:7A has been assigned to Apple, Inc., so an Ethernet device with a MAC address beginning with C0:84:7A was manufactured by that company. However, no other OUI codes beginning with C0:84 have ever been allocated, so no MAC addresses beginning with C0:84:00, C0:84:01, C0:84:02, etc., would be expected to be encountered in the wild.

Around 0.15% of possible OUI manufacturer codes are known to have been used; an intelligent search strategy can take advantage of this fact and hence consume only about 0.15% of the time that would be required, by considering only the minority of MAC addresses beginning with allocated codes (such as C0:84:7A) and skipping the majority of MAC addresses that begin with unallocated codes (such as C0:84:00). This is akin to considering only ZIP code ranges actually allocated by USPS or SSN ranges actually allocated by SSA. Using this strategy, only about 26000×224 = 436207616000 MAC addresses need to be considered, which should take our GPU about 545 seconds or around 9 minutes. Hence, for a realistic optimized search, the table should be completed as follows:

Data field type	Example value	How many possible values are there?	Seconds to test all possibilities at 800 Mhash/second	Time corresponds to
MAC address *(with valid OUI)*	58:91:CF:2B:09:1A	about 38.6 bits or $2^{38.6} \approx$ 436207616000	545	9 minutes

One expert we consulted doubted whether this efficiency could be fully realized on an actual GPU because programming the GPU to examine discontiguous input ranges slightly reduces its effective speed.

However, even given this limitation, it's clear that GPUs are up to this task and that their use for this purpose is entirely practical. Stanford privacy researcher Jonathan Mayer decided to rent a server from Amazon's EC2 service to see how easily he could actually deobfuscate hashed MAC addresses.[28] (Through this service, Amazon rents computer servers, including some equipped with powerful GPUs, to the general public cheaply for what can be very short time intervals.[29])

Mayer found experimentally that he could recover an original MAC address that had been obfuscated using the standard SHA1 hash function in 12 minutes *using a single rented server* at a cost of $0.65, using only publicly-available software and the official list of OUIs. (Mayer adds that there are many ways to speed up the process and to save the results for fast subsequent re-use when de-obfuscating large numbers of MAC addresses at once.) He also cites other researchers' estimates of the time and resources required to recover the original values of hashed data fields, which are broadly consistent with our own estimates and with his empirical result.

---

28 See Jonathan Mayer, "Questionable Crypto in Retail Analytics," March 19, 2014, available at <http://webpolicy.org/2014/03/19/questionable-crypto-in-retail-analytics/>. Mayer questions, for example, the Euclid Analytics Privacy Statement (version of January 9, 2014), which states that "[h]ashed data cannot be reverse-engineered by a third party to reveal a device's MAC address. This means that anyone who gains access to the database […] would see only long strings of numbers and letters. They would be unable to get any information that could be linked back to a particular device owner."

29 See Amazon EC2, <https://aws.amazon.com/ec2/>.

Dr. Felten's 2012 post concluded that

> There are more advanced uses of hashing that can offer some protection in some settings. But the casual assumption that hashing is sufficient to anonymize data is risky at best, and usually wrong.

We believe this conclusion is clearly correct with regard to MAC addresses.

Sincerely,

Lee Tien
Senior Staff Attorney

Seth Schoen
Senior Staff Technologist