# Social Security Numbers: Identification is Not Authentication

**There are many contexts in which Americans may be required or expected to share their Social Security Numbers (SSNs). What are the security issues around the use of your SSN as an identifier?**

David Harley, BA CISSP FBCS CITP
Director of Malware Intelligence

# Table of Contents



This paper is based on and expands a blog originally published on July 12, 2009.[1] Thanks to Charles Jeter for drawing my attention to this absorbing research topic!

# Introduction

Would you be happy to use your phone number as your password?

Well, people use all sorts of weak but memorable passwords and passphrases (and, even worse, use the same one for all their services). Table 1 demonstrates clearly that for many people, memorability trumps security, so maybe you would use your telephone number, license plate number and so on.

**Table 1: The 15 Most-Used Passwords**

| | |
|---|---|
| 1 | 123456 |
| 2 | password |
| 3 | 12345678 |
| 4 | 1234 |
| 5 | pussy |
| 6 | 12345 |
| 7 | dragon |
| 8 | qwerty |
| 9 | 696969 |
| 10 | mustang |
| 11 | letmein |
| 12 | baseball |
| 13 | master |
| 14 | michael |
| 15 | football |

Source: http://www.whatsmypass.com[2]

But what if you were required to use your telephone number as your password for a given service (and you knew that if you gave a *fake* telephone number, the service would be withheld)? I guess most people would be a bit worried if anyone who knew (or could obtain) their listed phone number would automatically also have access to their password.

Of course, there are probably no services that require you to give your phone number as your password. But there are quite a few services that do require U.S. citizens to given their Social Security Number (SSN) at some point in order to access them, by way of identification, and I very much doubt that many of them will accept a fake SSN.

So how secure is your Social Security Number? If your answer is: "Very. I only ever give it to organizations who are entitled to know it," that may not be as safe as it sounds. There are generic issues here you should be aware of:

- Some legitimate, convenient-to-subscribe-to organizations may require your SSN who are, nevertheless, not "entitled" to it.
- The difference between legitimate and illicit organizations (or their web content, URLs and so on) is not always as pronounced as you might think; otherwise, we wouldn't have to worry about phishing.

However, there is another critical issue here, and that involves an essential distinction between identification and authentication.

## I Authenticate, Therefore I Am

A very common example of an identifier is a username or account name; it's a way of telling a computer system who you are. What it doesn't do is prove that you are that person.

Here's a simple illustration of the difference between authentication and identification that goes back centuries (or more) in military history.

> "Halt! Who goes there? Friend or foe?"
>
> "Friend."
>
> "Advance, friend, and be recognized."

That's a pretty generic challenge, of course. Nowadays, a sentry is likely to be required to be pretty sure about the identity of the "friend" before allowing him to enter, and will have to check an ID card, a log of expected visitors and so on.

So let's look at a modern, more specific identification/authentication scenario that you're almost certainly familiar with, but may not have thought about in those terms (unless, of course, you're a system administrator, security geek or something similar).

## Mail Model

Anyone with whom I communicate from a particular email account knows my account name on that particular mail service (anonymizing services apart), and there are many ways to gain access to data, such as the name of the email server I have to go through in order to send or receive email. So, not only do I have to identify myself to the system so that it knows which mailbox to access, I also have to authenticate myself to the system to prove that I am indeed the person I say I am, and am *entitled* to have access to that mailbox. For email services, the most common authenticator is a password, though there are many alternatives used in other scenarios.

An authentication factor is a procedure used to verify an individual's identity and check that he has access rights to the system. A password or passphrase falls into the class of "knowledge factors." In this case, I prove my right of access through something I know, such as a password or passphrase, or a safe combination.

Identification is important in security and administrative terms because it represents the *accountability* of the person identified. If you know my username on a computer system, you can check on what I do on that system through an audit trail, and I can therefore be held accountable for those actions. Authentication (from the Greek αυθεντικός, meaning genuine) confirms that identity as authentic.

### Authentication Factors

While we won't consider them further here, there are two other main classes of authentication factors:

- Ownership (or technical) factors are examples of "something you have" rather than something you know. They include identity cards, passcode tokens and physical keys.
- Inherence factors are examples of "Something you are (or something you do)"; they include biometric devices like fingerprint or iris scanning, or algorithms to measure behavioral characteristics such as typing rhythm.

Where more than one authentication factor is used (for instance, a "chip and pin" bank card that requires both the possession of the card and the knowledge of the PIN that goes with it), this is referred to as two- or three-factor authentication.[3]

In principle, a Social Security Number (or an equivalent such as a National Insurance Number in the United Kingdom) is an identifier, not an authenticator. It would be unsuitable for a password, because it isn't secret. Many people know (or at least have access to) your SSN, and if one of them was determined to crack your password in some context, he or she might use that knowledge to guess it. But another problem arises if an organization providing some kind of service to you uses it as an authenticator rather than as an identifier.

## SSN Guessing

A paper by Alessandro Acquisti and Ralph Gross[4] for the Proceedings of the National Academy of Sciences claims that an SSN is (or can be) relatively quick and easy to guess; at least, it is if a criminal has access to other information relating to the prospective victim's place and date of birth. In other words, even if a criminal doesn't have access to your SSN, he or she may be able to guess it from information he or she obtains from other sources, even information you make publicly available on Facebook, for instance.

This shouldn't actually surprise you, unless you think of your SSN as being like a password, and hopefully you realize now that that isn't how it works. But how *does* it work?

## Too Much Information (Theory)

The point of a password, though we don't always think of it like this, is to achieve a balance between randomness and convenience. In fact, many sites use a password generator, which simply puts together random characters in a string. If randomness (or entropy, if you want to sound as if you know what you're talking about in conversation with information theorists) was the only consideration, then that might be a pretty good mechanism.

In practice, however, there are a number of other considerations. For instance, whether it's straightforward enough to remember (convenience), the maximum length of the key, and the variety of characters you are allowed to use. For example, a four-digit PIN (Personal Identification Number — yes, I'm afraid that name does rather blur the distinction between identification and authentication!) can be broken very quickly indeed, all things being equal. All you need to do is compile a list of every possible permutation of numbers between 0000 and 9999, and try each one until you get to the one that works.

Of course, this assumes that you can just keep trying until you find the right permutation. If access is locked after three tries, as is often the case with ATMs — Automated Teller Machines — that's rather more secure. The attacker needs to force or trick the target into giving up his PIN, or to intercept an authentication transaction somehow, if he's to stand much chance of breaking in.

Moving away from ATMs, though, if you can use a wider variety of characters (alphanumerics, punctuation and whitespace) and a longer passphrase, that offers you the chance to generate a much stronger key. In a number of blogs on good password practice[5] (and another white paper[3] on the subject currently in preparation), ESET researchers have mentioned a number of strategies for generating stronger passwords. Most of them involve introducing an element of (pseudo-)randomization because the more "random" the passphrase, the harder it is to guess (for humans or for computers).

However, a Social Security Number is not like a password at all. I might have mentioned that before.

## Hey Joe

In fact, an SSN is essentially a database primary key, an identifier that is unique to you and to your individual, personal record in the Social Security Office's database. The most practical way of generating such a key is often to enhance predictability, not to reduce it.

A primary key is often just a numeric value incremented automatically. For example, if the primary key for the first record is 1, the key for the second is 2, the key for the 10th is 10, and so on. This is how records are numbered by default in Microsoft Access if you use the autonumber facility. Table 2 represents a database record. Since I don't have the slightest idea what the SSO's databases look like (in fact, I don't even have an SSN of my own, as I'm not a U.S. citizen), this is a dummy record from a nonexistent database storing details of workers in a nonexistent company.

A Human Resources department might easily use the payroll number below as the primary key for Joe Sixpack, assuming that the payroll number is a unique number given to each new hire as his or her HR record is created. (In such an instance, the number might be generated automatically by the database.) When Joe has a query about his salary, he calls HR; they ask him his payroll number, and are able to go straight to his record in the database. That's because the number is unique to Joe, and even if the company takes on someone else who's name also happens to be Joe Sixpack, the same applies. HR will find the record for the Joe Sixpack who holds that payroll number because that's the primary key.

But what if HR chooses to incorporate more data into the primary key?

**Table 2: Example Database Record**

| | |
|---|---|
| First Name | Joe |
| Surname | Sixpack |
| Job Title | Chief Cook and Bottlewasher |
| Start Date | 1/1/2009 |
| Termination Date | Null |
| Payroll No. | 420 |
| Salary Details | We could share these with you, but then we'd have to kill you. |

For an attacker trying to guess the key for a victim's record, a sequentially generated number used as a primary key might still be quasi-random. If he or she is aiming to exploit a database known to have millions of entries, he or she may not be able to start to guess where in the sequence the victim's identifier is, so all he or she can do is pluck a number out of the air. (In fact, limitations in an autonumbering mechanism could introduce additional complications in this context.)

However, for really big databases, it's often convenient for the administrator to include information in the key that's specifically intended to reduce entropy, by giving more information than simple sequential numbering. For instance, you might want to distinguish between the two Joes by including information about the units in which they work into the primary key. So Joe in catering might have the payroll number 12-420, while Joe in accounting might be 3-779. (I'll resist the temptation to pun on Joe 90.[5]) It appears that the Social Security Office has been doing something rather similar to this consolidation of data into a single field.

## Entropy Entrapment and Naming of Parts[6]

Acquisti and Gross claim that there is a correlation between an SSN and the birthdate of its "owner" that makes it feasible to infer the SSN, given knowledge of that birthdate and assisted by public access to the Social Security Administration's Death Master File. This allowed the researchers to determine SSN allocation patterns based on the ZIP code of their birthplace and the date of issue. Since 1988, the government has been issuing numbers at birth, almost guaranteeing a close correlation between the two factors.[7]

According to the Social Security Office,[8] the nine digits of the Social Security Number are grouped as follows:

- The first three digits represent the Area Number.
- The next two digits represent the Group Number.
- The four digits at the end are called the Serial Number.

(This information seems to have been published in order to make it easier to spot fake SSNs.)

It is pointed out on that web page[8] that "This is an archival or historical document and may not reflect current policies or procedures," so I wouldn't assume that it represents current practice, and Acquisti and Gross may have access to information that I don't have. Still, it's clear that SSNs have not, historically, been random (sequential numbers never are, in principle!).

In this case, the potential exists to determine an individual's SSN algorithmically, if you know how old he is and where he was born. (Note, however, that it isn't necessarily easy or quick to make this determination; success depends on many factors, some of which are discussed below.)

Unfortunately, as I may have mentioned before, information relating to birthdays and geographical location is all too easy to find online nowadays, when so many people make it available on social networking sites. (For a particularly unsettling instance of a Facebook posting where incautiously revealed information had implications for national security in the UK, see Randy Abrams' blog post[9] at http://www.eset.com/threat-center/blog/?p=1281.

Acquisti and Gross say [4] that "Our results highlight the unexpected privacy consequences of the complex interactions among multiple data sources in modern information economies and quantify privacy risks associated with information revelation in public forums."

What they are really talking about here is an example of what we sometimes call a data aggregation or data inference attack. While each datum may be harmless in isolation, the aggregation of data from different sources allows the attacker to infer more information than an unwary target would expect or want. The whole is greater than the sum of the parts.

## Conclusion

The sky is not falling. Knowing your SSN is not necessarily enough to give an attacker control over your finances or steal your identity. It depends on the context and on what other information he or she has.

As the Social Security Office has rightly, if a little evasively, pointed out, Acquisti and Gross have not "cracked a code for predicting an SSN."

What they have done is make it feasible to predict the SSN for some people, given a sufficiency of resources. According to the LA Times,[10] it was able "to identify all nine digits for 8.5 percent of people born after 1988 in fewer than 1,000 attempts. For people born recently in smaller states, researchers sometimes needed just 10 or fewer attempts to predict all nine digits."

While most sites that use SSNs to verify a user's identity will not allow unlimited attempts, a botherder can use multiple machines to access multiple resources such as online credit approval services[11] to test numbers. While botnets are most often associated in the media with spam distribution and distributed denial of service (DDoS) attacks, they can be (and are) used for all sorts of distributed computing tasks like this, if the attack surface is big enough or vulnerable enough.

The Social Security Office has said that it is moving over to a more randomized SSN allocation system. (A pseudonymized approach might be one possibility

for achieving the necessary randomization.) That isn't necessarily going to fix the whole problem though, because:

- It may not impact directly on the fact that SSNs are often used where they're not really appropriate (that is, as authentication rather than as identification) or properly protected, and that's probably not going to change.
- It might not help all the people whose SSNs are vulnerable right now. Generally, government departments are not eager to change identifiers that are supposed to be "for life" for millions of people.
- It's certainly not going to fix the fact that so many people of all generations are increasingly likely to publish data like birthdates on social networking sites that probably shouldn't be so widely known.

After the blog on which this article is based was published, it was heavily quoted by SC Magazine,[12] where a reader pointed out that the use of SSNs for authentication as well as identification is described in documents published by the Federal Trade Commission.[13]

The problem here is less a "vulnerability" in the SSN generation algorithm than the misuse of the SSN as an authentication mechanism. In some contexts, it's likely that knowing someone's name, date of birth and SSN really is enough to steal their identity, and it's suggested[12] that even a partial match to an SSN may be enough to "authenticate" a credit applicant.

Unfortunately, short of a dramatic reengineering of society and the national infrastructure, there isn't much an individual can do except to keep his or her SSN to himself or herself as much as possible and to avoid giving out too much personal information in public (and that includes social networking sites!). You might also consider taking more countermeasures such as monitoring your credit status, SSN usage, credit reports, public records reports and other countermeasures that come under the identity monitoring banner.

## References


1. David Harley. "There's Security, and Then There's Social Security," ESET ThreatBlog, July 12, 2009. http://www.eset.com/threat-center/blog/?p=1313
2. http://www.whatsmypass.com/the-top-500-worst-passwords-of-all-time
3. David Harley, Randy Abrams. "Keeping Secrets: Good Password Practice"; forthcoming. http://www.eset.com/download/whitepapers.php
4. Alessandro Acquisti and Ralph Gross. "Predicting Social Security Numbers from Public Data". Proceedings of the National Academy of Sciences of the United States of America. http://www.pnas.org/content/early/2009/07/02/0904891106.abstract
5. http://en.wikipedia.org/wiki/Joe_90
6. Henry Reed. "Naming of Parts," New Statesman and Nation 24, No. 598, August 8, 1942. http://www.solearabiantree.net/namingofparts/namingofparts.html
7. http://blogs.discovermagazine.com/80beats/2009/07/07/researchers-guess-social-security-numbers-from-public-data/
8. The SSN Numbering Scheme. http://www.socialsecurity.gov/history/ssn/geocard.html, Social Security Numbers
9. Randy Abrams. "Social Networking or Social Suicide?" ESET ThreatBlog, July 6, 2009. http://www.eset.com/threat-center/blog/?p=1281
10. Los Angeles Times. "Study says SSNs can be cracked", July 7, 2009. http://www.latimes.com/news/nationworld/nation/la-na-briefs7-2009jul07,0,1044157
11. Hadley Leggett. "Social Security Numbers Deduced from Public Data," Wired, July 6, 2009. http://www.wired.com/wiredscience/2009/07/predictingssn/
12. Dan Raywood. "Claims on code breaking on Social Security Numbers dismissed, although more security needs to be applied," SC Magazine, July 15, 2009. http://www.scmagazineuk.com/Claims-on-code-breaking-on-social-security-numbers-dismissed-although-more-security-needs-to-be-applied/article/140066/
13. http://www.ftc.gov/reports/facta/041209factarpt.pdf; http://www.ftc.gov/os/2008/12/P075414ssnreport.pdf

## ESET Resources

ESET Threatblog (TinyURL with preview enabled): http://preview.tinyurl.com/esetblog

ESET Threatblog notifications on Twitter: twitter.com/esetresearch

ESET White Papers Page: http://www.eset.com/download/whitepapers.php

## Other Resources

"Securing Our eCity": http://www.securingourecity.org/

http://www.informationweek.com/news/security/privacy/showArticle.jhtml?articleID=218400854

http://www.pittsburghlive.com/x/pittsburghtrib/news/pittsburgh/s_632528.html

http://www.idanalytics.com/solutions/whitepapers.html

Leslie McFadden. "Detecting Synthetic Identity Fraud." http://www.bankrate.com/brm/news/pf/identity_theft_20070516_a1.asp

Randolph E. Schmid. "Social Security number code cracked, study claims," Washington (AP), July 6, 2009. http://www.google.com/hostednews/ap/article/ALeqM5iIPvqyHARkUTfz_yQs4a-kBeAsHgD9996BNG0

Chris Jay Hoofnagle. "Identity Theft: Making the Known Unknowns Known." Harvard Journal of Law and Technology, Vol. 21, Fall 2007. http://papers.ssrn.com/sol3/papers.cfm?abstract_id=969441

Stuart Rigot and Uttam Dubal. "Synthetic ID theft," Cyber Space Times, University of North Carolina School of Law. http://www.unc.edu/courses/2008spring/law/357c/001/idtheft/synthetic.htm