Utility of hashing and salting algorithms in quality improvement studies

Practical way to link patient charts from multiple locations

Malgorzata Kaminska MD CCFP MSc Richard Franke

Owing to the complexity of our health care system, patients access care in various ways and settings, and their medical information is captured in an array of electronic medical records (EMRs) that do not always communicate with one another.[1] Thus, if you want to gather information about patients for a quality improvement or research study, you might find yourself confronted with different sources of unlinked data that require you to find a way to link the data efficiently. Also, this link must not infringe on any patient privacy laws or ethical considerations.[2] This task may appear to be impossible at first, but it can be resolved easily using techniques from computer science: hashing and salting.

Real-life research conundrum

Consider a situation where you wish to study trends in bloodwork of patients in your town over the past decade. These patients would have sought care from different physicians and clinics, and the bloodwork ordered by one physician (or clinic) would rarely or never have been shared with another. As you prepare to gather data from EMRs around town, you ponder how you will link patient-specific data from multiple clinics together.

In this situation, you might have been advised to assign random unique alphanumeric codes (eg, A1A1A1) to individual patients (**Figure 1**). The disadvantage of this is that you cannot conduct studies across multiple physicians or health care settings because, without access to a patient's identifying information (eg, name, date of birth, provincial or territorial health card number), there is no way to assign the same alphanumeric code to that same patient at another location. Thus, a patient who sought care at different clinics would be assigned different codes at each site, and their data could never be linked.

Practical solution

Hashing is a 1-way algorithm that takes data you want to secure (eg, health card numbers) and turns the information into scrambled strings of characters.[3] This procedure has become a key aspect of Internet security and is used in other popular technologies, such as cryptocurrency.[3,4]

Hashing works by using an algorithm on patient data that are highly unlikely to change over a person's lifetime (eg, date of birth, health card number, first name) to create a unique identifier, making it possible to link a patient's data when collected from multiple sites. The study team can pick the personal data to input into the algorithm and, as long as the exact same fields are used each time, the same unique identifier for an individual patient will be outputted at every instance, across time and across different health care settings (**Figure 1**).

How secure is this approach?

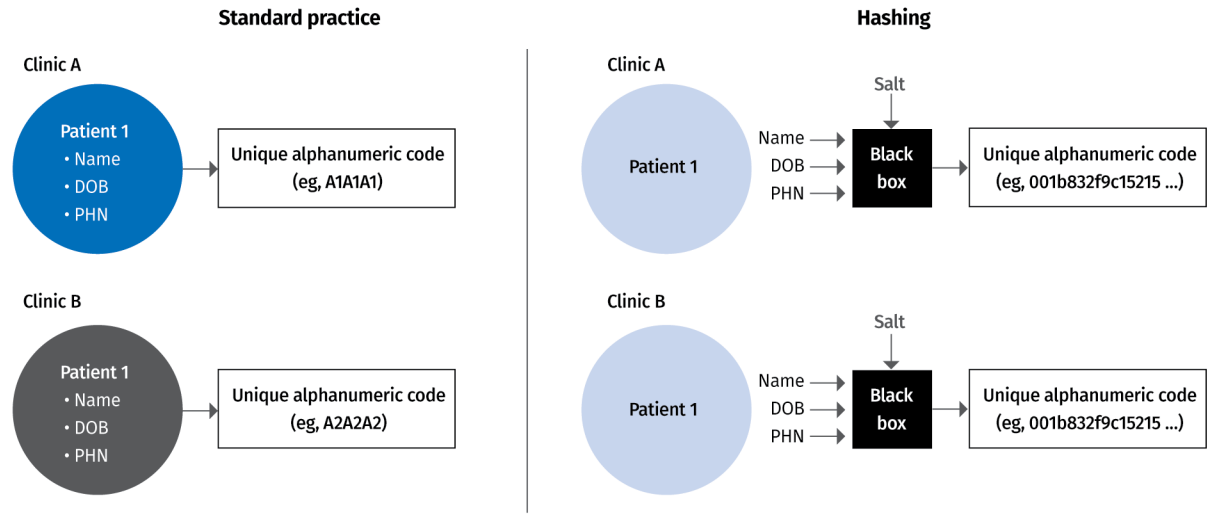
As an example, feeding John (first name), 999-999-999 (provincial or territorial health card number), and 1988 (year of birth) into a hashing algorithm produces the unique identifier 221c5ae9b14d19bd469ca529b41cc102. Trying to reverse engineer this code to obtain the data from which it originated would take 4.54×10^{17} combinations, the rough equivalent of 1 million computer years. However, as computer processing speeds have increased—consider that in the early 2000s a code that took 1 million computer years to reverse engineer could be cracked by a single computer in 1 year in 2023—a *salt* can be added to the hashing algorithm to make the calculations even more complex.[5] A salt is a block of characters that can be chosen or generated that adds trillions more years to any attempted reverse-engineering calculations. Also, these estimates assume that the reverse engineer knows which data fields were used to create the unique identifier. Thus, an additional layer of protection is created based on the *security through obscurity* adage, as it is unlikely that a researcher would reveal those fields.

How did it work in real life?

The situation described above is one that our research team encountered while examining bloodwork trends among patients in a mid-sized town in British Columbia. Here are the steps we followed:

1. After agreeing which personal patient data would be inputted, our team's computer programmer wrote a hashing program.
2. Our team's research assistant visited participating clinics and, using the typical EMR reporting function, created reports with the data of interest (ie, bloodwork). These reports included personally identifying patient data.
3. The reports were exported into spreadsheet files, which were collected in a folder on the clinic computer's desktop.
4. The research assistant installed the hashing program on the clinic computer's desktop.
5. The data folder was processed using the hashing program. The program removed personally identifying

Figure 1. Assignment of unique alphanumeric identifiers to a patient using standard practices versus with hashing and salting



DOB—date of birth, PHN—provincial health card number.

data and replaced them with unique identifiers (hashes). Thus, hashed files were created.

6. The hashed files were copied to a flash drive that the research team took from the clinic.
7. The original data reports, hashed files, and hashing program were deleted from the clinic's computer, thus destroying all retrieved personal patient information and ensuring the details of the hashing program remained confidential.
8. Once all hashed files from all participating clinics had been retrieved, an automated search looked across all files for repeated unique identifiers. When these were found, the data belonging to a given unique identifier were linked together.

Once data are hashed, there is no way to identify a specific patient's file, which could be an issue if a clinic had entered or coded data incorrectly. However, if the clinic were to repair the data, the data could be rehashed and exported, with patients each receiving the same unique code they had been assigned previously. Therefore, in our study, we assigned each clinic a unique alphanumeric code and kept track of from which clinic the hashed data had been collected. In this way, we could not only report aggregate results back to clinics about their own patients, but we could also identify affected clinics if any anomalies in hashed data were detected.

Conclusion

By using a hash-and-salt algorithm, we were able to conduct a multisite study through which we retrospectively gathered 10 years' worth of data about patients across multiple physicians and family medicine clinics while keeping patient data linked, despite having removed personally identifiable information.

Dr Malgorzata Kaminska is Assistant Professor in the Northern Medical Program at the University of Northern British Columbia in Prince George. **Richard Franke** is a freelance information technology specialist and an experienced research assistant in Prince George.

Acknowledgment

Financial assistance was received from the Rural Coordination Centre of British Columbia through the Rural Physician Research Support Project grant in the amount of $10,000.

Competing interests

None declared

References

1. Lehne M, Sass J, Essenwanger A, Schepers J, Thun S. Why digital medicine depends on interoperability. *NPJ Digit Med* 2019;2:79.
2. Price WN 2nd, Cohen IG. Privacy in the age of medical big data. *Nat Med* 2019;25(1):37-43. Epub 2019 Jan 7.
3. Gibbs S. Passwords and hacking: the jargon of hashing, salting and SHA-2 explained. *The Guardian* 2016 Dec 15. Available from: **https://www.theguardian.com/technology/2016/dec/15/passwords-hacking-hashing-salting-sha-2**. Accessed 2022 Jul 29.
4. Faife C. Bitcoin hash functions explained: everything you always wanted to know about bitcoin hashing, but were afraid to ask. *CoinDesk* 2017 Feb 19. Available from: **https://www.coindesk.com/markets/2017/02/19/bitcoin-hash-functions-explained/**. Accessed 2022 Jul 29.
5. Education Development Centre. *The beauty and joy of computing. Unit 6 Lab 2: history of computers. Moore's law*. Berkeley, CA: Berkeley University of California; 2020. Available from: **https://bjc.edc.org/bjc-r/cur/programming/6-computers/2-history-impact/2-moore.html?topic=nyc_bjc%2F6-how-computers-work.topic&course=bjc4nyc.html&novideo&noassignment**. Accessed 2023 Feb 27.

Can Fam Physician 2023;69:215-6. DOI: 10.46747/cfp.6903215

Hypothesis is a quarterly series in *Canadian Family Physician (CFP)*, coordinated by the Section of Researchers of the College of Family Physicians of Canada. The goal is to explore clinically relevant research concepts for all *CFP* readers. Submissions are invited from researchers and nonresearchers. Ideas or submissions can be submitted online at **https://mc.manuscriptcentral.com/cfp** or through the *CFP* website **https://www.cfp.ca** under "Authors and Reviewers."