Carnegie Mellon
DATA PRIVACY LAB

Towards a Privacy-Preserving Watchlist Solution

An important surveillance problem is the Watchlist Problem, which can be generally described as follows. Government authorities have an explicit list of names of known or suspected terrorists (a "watchlist") they want to locate or merely track among the U.S. population. There are vast numbers of locations the government seeks to query as to whether a customer or patient has appeared in one of these locations bearing an explicit identity appearing on the Watchlist. The idea is to review transactional data (such as that which results from store purchases, hotel registrations, airplane manifests, car rentals, school attendance records, etc). The authorities are to be notified if someone bearing an identity of someone on the watchlist appears at a location. This work examines two challenges lurking within the Watchlist Problem: (1) false matches; and, (2) trail re-identification vulnerabilities. At present, NO solution solves these problems.

False Matches ("Why Ted Kennedy can't fly")

A false match can result because a person bears the same name as a name on the Watchlist, or because a person bears a name that is "similar" to a name appearing on the Watchlist. These fuzzy matches are based on nicknames, intentional misspellings, and systemic ways names are compared. The current method uses Soundex which is horribly problematical.

1. Print the name.
2. Remove all non-letters such as spaces, punctuation, accents and other marks.
3. Remove all occurrences of: A, E, I, O, U, H, W, Y.
4. Remove the second letter of duplicate characters.
5. Remove the second letter of any adjacent letters that have the same soundex number.
6. Convert characters in positions 2 to 4 to a number:
   - B, P, F, V gets 1
   - C, S, K, G, J, Q, X, Z gets 2
   - D, T gets 3
   - L gets 4
   - M, N gets 5
   - R gets 6

   Fill any unused positions with zeros.

In the end, there is always one letter followed by 3 numbers.

Figure 2. The Soundex algorithm. Originally designed to find alternative spellings of names which "may" be related to duplicate records in databases. Examples: Lee is L00 and Bailey is B400.

Limitations:
[1] Despite its objective, some names that sound alike do not always have the same soundex code. For example, Lee (L000) and Leigh (L200) are pronounced identically, but have different soundex codes (because the silent g gets coded).
[2] Names that sound alike but start with a different first letter will always have a different soundex code. For example, Carr (C600) and Karr (K600).
[3] Soundex is based on English pronunciation so it may have limited uses in other contexts.
[4] Sometimes names that do not sound alike have the same soundex code. For example, Powers, Pierce and Price all have the same code (P620).

Given: (1) a set of data holders, where each data holder has a dataset of various transactional attributes about people; and, (2) a central authority, who has a list of suspicious people ("Watchlist"), the goal is to provide a system which identifies occurrences of suspicious people in the transactional data to the central authority without revealing the Watchlist to the data holders or the identities of the subjects of the transactional data not on the Watchlist to the central authority.

Figure 1. The Watchlist Problem.

Trail Re-identification Attack

In prior work, Malin and Sweeney [2000-2003] presented a series of algorithms that re-identify people from the trail of disparate pieces of information left behind. One example is re-identifying people by name using the IP addresses left behind at websites visited.

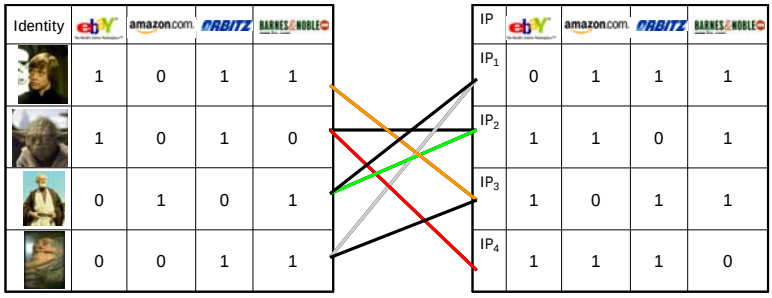

Figure 3. Binary "trails" of sites where person is known by name through purchases or services (left) and sites visited (right).

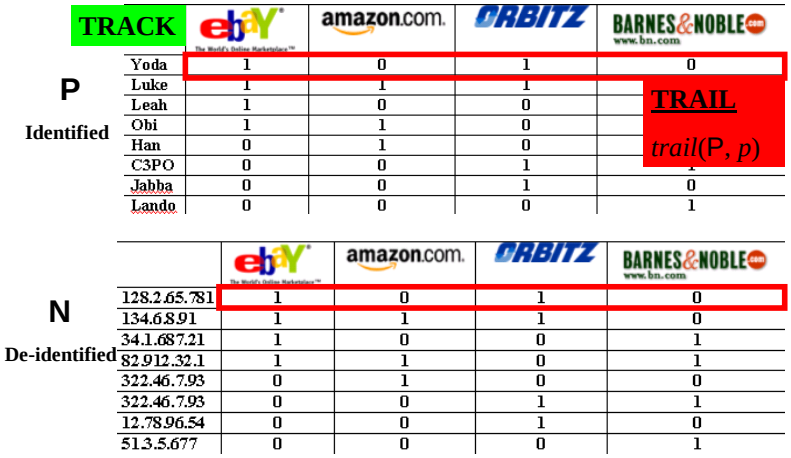

Figure 4. Matrices used by trails re-identifications algorithms for matching identified trails to de-identified trails.

Even when strong cryptographic hashing is used, as proposed by ANNA (Jonas, 2003), trail re-identification vulnerabilities continue. In the ANNA proposal, each person's name is hashed (along with hashes of other possible name spellings, so that rather than a single hashed value, a set of hash values result). These hashed sets are shared and compared rather than actual names. This proposal provides false privacy protection, because credit card transactions, loyalty card use, and others identified information can be used to relate the identities of the hashed sets to the named identities that generated the sets. This works in the same ways names are re-identified to IP trails using trail re-identification above.

Latanya Sweeney
http://privacy.cs.cmu.edu/dataprivacy/projects/watchlist/index.html