FINGERPRINTING

Neal R. Wagner

Drexel University
Mathematical Sciences Department
Philadelphia, Pennsylvania 19104

Abstract. This paper presents a general discussion of the use of fingerprints, especially fingerprinted data. Fingerprinting is classified in four orthogonal ways, and some illustrative examples are given. The basis for a statistical analysis of altered fingerprints is presented, along with an example simulation. The possibility of more subtle fingerprints is discussed.

1. Introduction

Fingerprints are characteristics of an object that tend to distinguish it from other similar objects. Fingerprinting refers to the process of adding fingerprints to an object and recording them, or of identifying and recording fingerprints that are already intrinsic to the object.

Fingerprinting is now in wide use. Typical examples include:

- o Human fingerprints have had numerous applications for centuries.
- o We routinely match typed characters with a specific typewriter, or a fired bullet with a specific weapon, etc.
- o Many objects are now provided with a unique identification string, such as the "serial number" on consumer goods.
- o Similar advertisements are often placed in different magazines with slightly varying return addresses, so that one can determine which magazine gave the best response. (Thus an advertisement in "Scientific American" might have "Dept. SA" as part of its return address.)
- o Some explosives are now manufactured with tiny coded particles that can be found after an explosion. Examination of the particles will yield the place and approximate time of sale [Sci80].
- o Tables of logarithms have been generated with some of the least significant digits altered to protect copyright [Mil78, p.130].
- o Maps are sometimes drawn with slight deliberate variations from reality, in order to identify copiers.

We suggest that fingerprinting might be employed more systematically and on a larger scale than at present. While fingerprinting in itself provides only detection and not prevention, the ability to detect may help deter individuals from committing the acts being detected.

The next section gives a partial classification of fingerprinting, while Section 3 focuses on statistical aspects. Section 4 gives more examples, and Section 5 discusses subtle kinds of fingerprinting.

2. Taxonomy of Fingerprinting

We wish to classify fingerprinting. Let us first identify the types of individuals who might manipulate fingerprints. The distributor is the authorized supplier of fingerprinted objects to users. Users are individuals (or agencies) who are authorized to gain access to the objects. In many cases users will be aware that certain objects are fingerprinted. The opponent is an individual (or agency) who gains unauthorized access to objects (or makes unauthorized use of objects), through one or more users. The opponent might be a user or might have compromised a user.

The distributor's goal is to identify the user or users whom the opponent has compromised. Usually the distributor will examine a copy of an object which has been used in an unauthorized way and will try to employ the fingerprints to trace the object back to a specific user, or to several specific users.

The opponent's goal is to prevent identification of the compromised user or users by the distributor. The opponent will often try to subvert the distributor's task by manipulations and alterations of the fingerprints.

In the remainder of this section we consider four more-or-less mutually orthogonal divisions of fingerprinting.

First orthogonal division:

- o Logical fingerprinting. Here the object to be fingerprinted must be in machine-readable form, such as a file of Ascii characters. The fingerprints will be computer-generated and subject to computer processing.

- o Physical fingerprinting. This is the opposite of logical fingerprinting. Here the fingerprints depend on physical characteristics of the object.

Second orthogonal division:

- o Perfect fingerprinting. These are characterized by: any alteration to the object that will make the fingerprinting unrecognizable must necessarily make the object unusable. Thus the distributor can always identify the opponent by one misused object. (For a physical example, consider a serial number that cannot be removed or altered. One would have a near-perfect logical example if one modified the fourth example in Section 1 by taking out a separate P.O. Box for the return address in each separate magazine advertisement. See Example A in Section 4 for more discussion.)

- o Statistical fingerprinting. These are characterized by: given sufficiently many misused objects to examine, the distributor can gain any desired degree of confidence that he has correctly identified the compromised user. The identification is, however, never certain. (Examples appear in Section 3.)

- o "Normal" fingerprinting. This is a catch-all category for fingerprinting that does not belong to one of the first two categories.

Third orthogonal division: (These categories have to do with the manner in which fingerprints become associated with objects. Some fingerprinting will overlap across several of the categories.)

- o Recognition. Recognize and record fingerprints that are already a part of the object.

- o Deletion. The omission of some legitimate portion of the original object.

- o Addition. The addition of some new portion to the object. We have two subcategories:
  - Legitimate addition.
  - Bogus addition.

- o Modification. A change to some portion of the object.

Fourth orthogonal division:

Usually the fingerprinting on an object will decompose into separate parts, each of which we call a fingerprint.

- o Discrete fingerprint. An individual fingerprint with only a limited number of possible values. Subcategories are:
  - Binary fingerprint. A two-valued discrete fingerprint.
  - N-ary fingerprint, where $N \geq 2$.

- o Continuous fingerprint. Here a real quantity is involved, and there is essentially no limit to the number of possible values. (For example, a physical continuous fingerprint might be placed on a liquid by varying the percentage of some additive. We will refer to logical fingerprints involving floating point numbers as continuous even though technically there are only finitely many possible values.)

3. Statistical Fingerprinting

In this section we have chosen a specific statistical fingerprinting scenario and a specific algorithm for processing the returned data. This is intended mainly as an illustration of what can be done, since there are many different possible approaches.

Suppose we have n real data values V_1, V_2, ..., V_n and m users. In order to qualify for use in statistical fingerprinting, each value V_j must have an associated delta value $D_j > 0$ with the property that any number in the closed interval $[V_j - D_j, V_j + D_j]$ is acceptable for use by all users.

For approximately 50 percent of the data values, users are provided with either V_j or $V_j + D_j$. In the remaining cases, users get either V_j or $V_j - D_j$. All choices are done in pseudo-random fashion as described in the next section. With this strategy, any coalition of users has at most two versions of the given data value to work with -- one correct and one incorrect, and they do not know which is which. (For our purposes here, we need two acceptable values for each datum, and it does not matter whether one is "correct" and the other is "incorrect".) The version of the jth datum sent to User i is denoted V_{ij}.

Now suppose the data has been misused in some way, and that values V_1', V_2', ..., V_n' return to the distributor. For each i, $1 \leq i \leq m$, we want to test the hypothesis that User i is the source of the returned values. For this purpose we examine

the numbers

$$L_{ij} = (1/D_j)(V_j' - V_{ij}),\ 1 \leq j \leq n, \text{ fixed } i.$$

(The L_{ij}, $1 \leq j \leq n$, are the normalized differences between the returned values and the values given to User i.)

For fixed i, we consider means over two disjoint subsets of the L_{ij}, $1 \leq j \leq n$. We set MH_i equal to the mean of those L_{ij} such that V_{ij} is the higher of the two versions of V_j sent to different users, and ML_i equal to the mean of those L_{ij} such that V_{ij} is the lower of the two versions. Finally we set

$$M_i = ML_i - MH_i.$$

First suppose the opponent made no alteration to the values before they returned as the V_j'. If the opponent got his data from User i, then we will have

$$M_i = MH_i = ML_i = 0.0.$$

If the opponent got his data from some other user, then we expect

$$MH_i \doteq -.5,\ ML_i \doteq .5, \text{ and } ML_i - MH_i \doteq 1.0,$$

where $\doteq$ indicates approximate equality. Thus if an opponent makes no alterations to the data, we expect him to be immediately noticable, unless n is very small.

Now suppose the opponent does alter the returned values. (This is his best strategy.) Then even for large n, it may no longer be true that $MH_i \doteq 0.0$, since the opponent might alter values according to some distribution with non-zero mean. However, we are assuming the opponent cannot tell which values were the larger of the two possible versions and which were the smaller. Thus for large enough n, if User i is the source of the opponent's values, we expect that M_i will be as close to zero as we like.

On the other hand, if User i is not the source of the opponent's values, we expect MH_i and ML_i to be approximately 1.0 apart for large enough n, i.e., we expect to have

$$M_i = ML_i - MH_i \doteq 1.0, \text{ for large n.}$$

We are mainly interested here in illustrating a basic approach and have not done any statistical analysis. Roughly speaking, we can say the following:

For each i, calculate the difference M_i of the two means as above. If for some specific i, M_i is close to 0.0, and for all other k not equal to i, M_k is close to 1.0, then there is evidence for the hypothesis that User i for that specific i is the source of the misused data.

This is a statistical result, so the probability that our hypothesis is correct will get closer to 1 as we take tighter bounds around 0.0 and 1.0, and as we let n get larger. The hypothesis never becomes a certainty.

Note that the opponent cannot protect himself from disclosure by this method. The particular distribution chosen for pertubations of the data, whether random or deterministic, symmetric or skew, does not matter with this method. The standard deviation of this distribution, if made large, only forces a larger n to be used in order to get strong evidence. Notice however that the opponent cannot perturb the data values too far without rendering them useless.

We have programmed a simple simulation of this scheme, with 200 data values (n = 200), 40 users (m = 40), each V_j chosen randomly from the range 10.0 to 100.0, and each D_j equal to 0.5. We assume the opponent uses the following random pertubation strategy:

50 percent of time: round to nearest integer
25 percent of time: truncate to nearest integer
12.5 percent of time: round and add 1
12.5 percent of time: round and subtract 1

Figure 1 gives the first five data values and values provided to five of the users.

j	Value	Value sent to User i, for i = 1	2	3	4	5	Returned value
1	11.5	11.5	11.5	11.0	11.0	11.5	11
2	69.8	69.3	69.8	69.3	69.8	69.8	69
3	41.9	41.4	41.4	41.4	41.4	41.9	41
4	81.9	82.4	81.9	82.4	81.9	82.4	82
5	75.8	76.3	76.3	75.8	75.8	76.3	76

Figure 1.

Then the values of the two means and difference of means for the first 5 users are shown in Figure 2.

i	MH_i	ML_i	$M_i = ML_i - MH_i$
1	-.78	.22	1.00
2	-.97	.38	1.36
3	-.81	.25	1.07
4	-.70	.11	.82
5	-.10	-.33	-.22

Figure 2.

Statistics for users 6 through 40 are similar to those for 1 through 4, and user 5, who was the source of data for the opponent in the simulation, stands out clearly. (The opponent was always visible like this in repeated simulations with these parameter settings. The values in Figure 2

are about the worst that came up. Of course there is always a small probability that this method will mistakenly identify the wrong user.)

Notice that we are using fairly small values for the D_j compared to the pertubations used by the opponent. Notice also that the distribution used by the opponent for perturbations is skewed to the left by approximately 0.125, as is reflected in the values for MH_i and ML_i (skewed left by an average of 0.25 because of normalization).

4. Enhancements and Examples

The type of fingerprinting one should employ will depend on many factors, including characteristics of the object being fingerprinted and the anticipated nature of the opponent.

The same object should always be provided to the same user with the same fingerprints. If the opponent makes no alterations to the fingerprints, then only $\lceil \log_2(m) \rceil$ binary fingerprints will be needed to distinguish m users. The case with no alterations is so easy to detect that we will assume an active opponent. Here we will usually need many more than $\lceil \log_2(m) \rceil$ fingerprints.

In general we prefer pseudo-random fingerprinting, in which the user id and an identification of the portion of the object being fingerprinted are fed in as the seed to a random number generator. The output determines which version of an individual fingerprint to use. Advantages of this approach are the ease of generating the fingerprints, the ease of adding new users, and the fact that minimal data needs to be stored to associate the proper fingerprints with each user. The disadvantages compared to some exact error correcting code for the fingerprinting are that it may take more fingerprints to distinguish users and more processing time to identify the compromised user. Also it is possible for two users to end up with the same fingerprints. This pseudo-random approach is similar to the random codes used to prove Shannon's main coding theorem [Ham80], except that true random codes require large codebooks. Pseudo-random fingerprinting might work best when one rarely needs to seek out an opponent, and when one does not know what fingerprint alterations the opponent might employ.

A hybrid approach will often be useful, in which the user id is encoded directly into a few of the fingerprints to allow immediate identification of the compromised user in case there are no alterations.

One further problem should be mentioned. In a traditional environment a dutiful user will react strongly to the slightest variation in two copies of a classified document. One might avoid difficulties by openly acknowledging the presense of fingerprints. There could also be legal problems if there are multiple versions of an "official" document.

Example A. Consider the example of different return addresses, introduced in Section 1 and mentioned again in Section 2. The fingerprint "Dept. SA" is an obvious one that is easy for an opponent to alter or delete. Separate P.O. Boxes will work nearly perfectly because the opponent cannot even make use of the address without leaving on the fingerprint. In practice we can strive for near-perfect fingerprints by making less obvious changes to the return address, such as altering the name of the person addressed. In general, perfect logical fingerprints can be created if the distributor is in a position to require that the fingerprints be present.

Example B. Suppose we are sending data to NATO allies regarding the sizes of newly-designed weapons systems, like lengths of missiles, sizes of tanks, etc. With some simplification, we could picture this looking like the simulation at the end of Section 3, where lengths are provided in meters to the nearest 0.1 meter, say. Fingerprints are added by perturbing selected items by 0.5 meters. The opponent rounds to the nearest meter and sometimes adds or subtracts an extra meter. (If the opponent did much more to the data, it might be useless when passed on.) With 200 data values, the simulation suggests that we will almost always pick out the compromised user correctly.

Example C. Consider a medical database with medical histories of individuals. Here a variety of fingerprints might be useful. The individual whose history was recorded could confer with a medically knowledgeable database administrator to decide on various binary and n-ary fingerprints, including trivial variations in the year of a disease or vaccination, and bogus entries or omissions of a medically insignificant nature. Each individual query is fingerprinted according to the query source. This application could create ethical and legal problems because of difficulties in interpreting "medically insignificant." It would be useful to have a legal concept of an "official fingerprinted medical record."

Example D. Consider some text prepared from machine-readable source. Here we could incorporate physical fingerprints in an arbitrarily subtle form -- occasional individual letters could be in a different type face or could have minute variations from normal. If such a text is xeroxed as it is, then it will be trivial to identify the compromised user. (If it is anticipated that only small portions will be xeroxed, one might want to place fingerprints everywhere.) Because of the ease of detecting fingerprints in xeroxed copies, the sophisticated opponent will realize that he must at least re-type any documents. This will eliminate physical fingerprints, but not logical ones, and the logical fingerprints are also easy to detect in xeroxed copies. For these reasons there should seldom be a need for physical fingerprints on text data. One should instead choose n-ary variations in the text for fingerprints.

Example E. Finally consider a piece of software. As copies of programs are prepared for shipment to individual users, it would be easy for an extra small utility to add fingerprints, based on a table

of possible changes. There is so little extra overhead that this should be routine practice. More specifically supppose a user signs an agreement not to further distribute some piece of software received for a nominal fee. If an unauthorized copy is found, and if the fingerprints identify this particular user, then the user could be denied a later release or charged more for it, or even sued or "publicly humiliated." The fear of these possibilities would help enforce compliance with the agreement. In addition, the user might appreciate knowing that his organization has a security problem.

5. Subtle Fingerprinting

As we use more drastic alterations for fingerprints, we expect the resulting fingerprinted object to survive more drastic manipulations in recognizable form. However, it would be better if we could find more subtle fingerprints.

Consider the example of a piece of software from Example E of Section 4. The opponent could easily delete comments, reformat, rename identifiers, and convert some high-level constructs to a lower level. We might find our misused software in compiled form only, as machine code. Even in this extreme case, it is usually fairly easy to locate places in algorithms where there are several simple ways to accomplish the same thing. One could even select changes that will survive the modifications of an optimizing compiler. As a drastic example, if the software employed a binary search, one could use Fibonacci search [Knu73] as a near-equivalent alternative to give a binary fingerprint. It is easy for the software developer to find places for such fingerprints. If an opponent wanted to make a number of changes like these in order to destroy possible fingerprints, he would need to understand the software well enough to write it himself, and still might miss many fingerprints. (See also [DeM78] for a suggested method of placing "dirty, strange fingerprints" on software.)

There is a story, perhaps apocryphal, that a famous mathematician was sent a Russian-language probability text to review. The text turned out to be his own book, translated into Russian. His name, transliterated into the Russian alphabet and then back into the Roman alphabet, was so altered that the reviewing journal did not recognize it as the same.

As another example, one might be able to recognize that a distance figure in miles had been translated to kilometers and back to miles with rounding both times. For instance, if we send out 500 miles, and 497 miles comes back, we might conjecture that the 500 was converted to 804.675 kilometers, rounded to 800 kilometers, the result converted to 497.095 miles, and this rounded to 497 miles.

Along these lines we would like to find more general ways of sending data out into the world and by the alterations in it when it returns, recognize something about the path it had taken.

6. Conclusions

We have presented a case for the routine use of fingerprints on data. If done in pseudo-random fashion, fingerprint insertion can be little extra overhead. There are statistical methods for identifying fingerprints even when an opponent tries to alter them. More subtle fingerprints may still be recognizable after drastic alterations.

Acknowledgement

This work was supported in part through funds provided by Drexel University under its Faculty Development Mini-Grant program.

References

[DeM78] R.A. DeMillo, R. Lipton and L. McNeil, "Proprietary software protection" in Foundations of Secure Computation, ed. by R.A. DeMillo, et.al., Academic Press, 1978.

[Ham80] R.W. Hamming, Coding and Information Theory, Prentice-Hall, 1980.

[Knu73] D. Knuth, The Art of Computer Programming, Vol. III: Sorting and Searching, Addison-Wesley, 1973.

[Mil78] J.K. Millen, "Discussion" in Foundations of Secure Computation, ed. by R.A. DeMillo et.al., Academic Press, 1978.

[Sci80] "After the blast: fingerprinting bombs," Science 80 1, 6(Sept./Oct. 1980), pp.96-99.