BMC Medical Research Methodology

BioMed Central

Research article

Open Access

Threshold protocol for the exchange of confidential medical data

Jules J Berman*

Address: Cancer Diagnosis Program, National Cancer Institute, Bethesda, Maryland, USA

Email: Jules J Berman* - bermanj@mail.nih.gov

* Corresponding author

Published: 11 November 2002

BMC Medical Research Methodology 2002, **2**:12

Received: 16 October 2002
Accepted: 11 November 2002

This article is available from: http://www.biomedcentral.com/1471-2288/2/12

Abstract

Background: Medical researchers often need to share clinical data without violating patient confidentiality. Threshold cryptographic protocols divide messages into multiple pieces, no single piece containing information that can reconstruct the original message. The author describes and implements a novel threshold protocol that can be used to search, annotate or transform confidential data without breaching patient confidentiality.

Methods: The basic threshold protocol is: 1) Text is divided into short phrases; 2) Each phrase is converted by a one-way hash algorithm into a seemingly-random set of characters; 3) Threshold Piece 1 is composed of the list of all phrases, with each phrase followed by its one-way hash; 4) Threshold Piece 2 is composed of the text with all phrases replaced by their one-way hash values, and with high-frequency words preserved. Neither Piece 1 nor Piece 2 contains information linking patients to their records. The original text can be re-constructed from Piece 1 and Piece 2.

Results: The threshold algorithm produces two files (threshold pieces). In typical usage, Piece 2 is held by the data owner, and Piece 1 is freely distributed. Piece 1 can be annotated and returned to the owner of the original data to enhance the complete data set. Collections of Piece 1 files can be merged and distributed without identifying patient records. Variations of the threshold protocol are described. The author's Perl implementation is freely available.

Conclusions: Threshold files are safe in the sense that they are de-identified and can be used for research purposes. The threshold protocol is particularly useful when the receiver of the threshold file needs to obtain certain concepts or data-types found in the original data, but does not need to fully understand the original data set.

Background

Many countries have implemented laws regulating the uses of confidential medical records. In the United States, restrictions on the research uses and electronic transfer of confidential medical records are covered by two Federal Regulations: The Common Rule (Title 45 Code of Federal Regulations, Part 46, Protection of Human Subjects) [1] and the Standards for Privacy of Individually Identifiable Health Information, Final Rule (usually referred to under the broader act, the Health Insurance Portability and Accountability Act, HIPAA)[2]. Researchers who wish to use confidential medical records must fully comply with these two sets of regulations. Both regulations permit the use of preexisting records for human subject research when the records are rendered harmless (through de-identification). Both regulations specify that deidentified records

can be used for research purposes without obtaining informed consent from patients. The ability to conduct research without obtaining patient consent is crucial for studies using large numbers of pre-existing patient records.

The purpose of this paper is to describe and implement a threshold protocol that can facilitate the exchange of medical information useful for research purposes. A threshold protocol is a cryptographic technique that splits information into pieces, none of which contains sufficient information to re-create the original text [3]. The protocol permits the original information to be reconstructed from some number of the derived pieces (the threshold number). Threshold protocols have been used since antiquity, commonly appearing as plot devices in adventure novels. A map to buried treasure is divided among the central characters, a puzzle is reconstructed when five missing pieces are assembled, measured turns of the combination lock are distributed to three untrustworthy co-conspirators, matching rings in a set are destroyed, etc.

Methods

A generalized confidentiality problem can be presented as a negotiation protocol between Alice and Bob.

Bob has a file containing the medical records of millions of patients. Alice has secret software that can annotate Bob's file, enhancing its value many-fold. Alice won't give Bob her secret algorithm, but is willing to demonstrate the algorithm if Bob gives her his database. Bob won't give Alice the database, but he can give her little snippets of the database containing insufficient information to match patients with records.

Bob prepares an algorithm that transforms his file into two threshold pieces. Piece 1 is a file that contains all of the phrases from the original file with each phrase attached to its one-way hash value. A one-way hash value is a character string composed of a fixed number of seemingly random characters selected by a mathematical algorithm that cannot be reversed [3]. The one-way hash has two important properties: 1) a phrase will always yield the same hash value when operated on by the one-way hash algorithm, and 2) there is no feasible way to determine the phrase by inspecting or manipulating the hash value. This second property holds true even if the hashing algorithm is known. Bob will give Alice Piece 1.

Piece 2 is a file wherein each phrase from the original file is replaced by its one-way hash value. High frequency words (so-called "stop" words such as the, and, an, but, if, etc.) are left in place in Piece 2. The use of "stop" words to extract useful phrases from text is a popular indexing technique [4,5]. The list of "stop" words used in the threshold algorithm was taken directly from the National Library of Medicine's PubMed resource [4] and was chosen because it is a publicly available list. Alternate lists of "stop" words have been used for specific indexing purposes [5]. Piece 2 and Piece 1 are used to reconstruct the original text or an annotated version of the original text, using Alice's modifications to Piece 1. The reconstruction algorithm simply steps through all the character strings found in Piece 2. When it encounters a hash-value, the algorithm looks at the list of hash-values in Piece 1 and substitutes the phrase associated with the hash-value back into the Piece 2 file. All other terms in Piece 2 are ignored. This continues until the end of Piece 2 is reached, at which time the Piece 2 file has been restored as the original file (plus any annotations that Alice may have added to the terms in Piece 1).

The following is an example of a single line of μ Bob's text that has been converted into two threshold pieces according to the described algorithm.

Bob's original Text:

"they suggested that the manifestations were as severe in the mother as in the sons and that this suggested autosomal dominant inheritance."

Bob's Piece 1.

684327ec3b2f020aa3099edb177d3794 = >
suggested autosomal dominant inheritance

3c188dace2e7977fd6333e4d8010e181 = >
mother

8c81b4aaf9c2009666d532da3b19d5f8 = >
manifestations

db277da2e82a4cb7e9b37c8b0c7f66f0 = > suggested

e183376eb9cc9a301952c05b5e4e84e3 = > sons

22cf107be97ab08b33a62db68b4a390d = > severe

Bob's Piece 2.

they db277da2e82a4cb7e9b37c8b0c7f66f0 that the

8c81b4aaf9c2009666d532da3b19d5f8 were as

22cf107be97ab08b33a62db68b4a390d in the

3c188dace2e7977fd6333e4d8010e181 as in the

e183376eb9cc9a301952c05b5e4e84e3 and that this

684327ec3b2f020aa3099edb177d3794.

The author has prepared a Perl implementation that has been placed in the public domain. Perl is itself an open-source platform-independent language that is available at no cost. Perl interpreters for virtually every operating system are available at:

http://www.cpan.org

A sample thresholding script and example medical text files can be downloaded at:

http://65.222.228.150/jjb/thresh.tar.gz

Methods for decompressing tar.gz files are freely available and described at:

http://www.gzip.org/#faq6

The text files used were taken from the publicly available medical text, Online Mendelian Inheritance in Man (OMIM) [6]. The complete OMIM exceeds 70 Mbyte and can be downloaded in simple ASCII format. Because this text is publicly available, it is an ideal corpus for testing future thresholding algorithms against the algorithm suggested in this article. Piece 1 and Piece 2 files constructed from the first megabyte of OMIM are included in the distribution file (thresh.tar.gz). Instructions for downloading the complete OMIM text are available at:

http://www.ncbi.nlm.nih.gov/Omim/

The Perl implementation of the threshold file is fast. A text file exceeding 2 MegaBytes was rendered into two threshold pieces in 10 seconds. A Pentium 4 CPU with 480 MegaBytes of RAM was used. Variations on the implementation can substantially slow performance. For example, it may be desirable to exclude punctuation from the extracted phrases added to Piece 1, while preserving the location of punctuation in Piece 2. Likewise, saving case information (uppercase vs. lowercase formatting) would also lengthen execution time. A sample Perl script that preserves case and punctuation is provided in the public distribution file.

As an example of how threshold pieces can be used to enhance the value of the original data records, phrases were autocoded using Concept Unique Identifiers (CUIs) found in the Unified Medical Language System (UMLS) [7]. The UMLS is a standard medical terminology available at no cost from the National Library of Medicine at:

http://www.nlm.nih.gov/research/umls/umlsmain.html

A tarballedgzipped collection of a Perl class library containing methods for autocoding (i.e. automatically assigning UMLS CUI numbers) and scrubbing text can be downloaded as a supplementary file from:

http://65.222.228.150/jjb/parse.tar.gz

Results

Properties of Piece 1 and Piece 2

Piece 1 (the listing of phrases and their one-way hashes)

1. Contains no information on the frequency of occurrence of the phrases found in the original text (because recurring phrases map to the same hash code and appear as a single entry in Piece 1).

2. Contains no information that Alice can use to connect any patient to any particular patient record. Records do not exist as entities in Piece 1.

3. Contains no information on the order or locations of the phrases found in the original text.

4. Contains all the concepts found in the original text. Stop words are a popular method of parsing text into concepts [4,5].

5. Bob can destroy Piece 1 and re-create it later from the original file, using the same threshold algorithm.

6. Alice can use the phrases in Piece 1 to transform, annotate or search the concepts found in the original file.

7. Alice can transfer Piece 1 to a third party without violating HIPAA privacy rules or Common Rule human subject regulations (in the U.S.). For that matter, Alice can keep Piece 1 and add it to her database of Piece 1 files collected from all of her clients.

8. Piece 1 is not necessarily unique. Different original files may yield the same Piece 1 (if they're composed of the same phrases). Therefore Piece 1 cannot be used to authenticate the original file used to produce Piece 1.

Properties of Piece 2

1. Contains no information that can be used to connect any patient to any particular patient record.

2. Contains nothing but hash values of phrases and stop words, in their correct order of occurrence in the original text.

3. Anyone obtaining Piece 1 and Piece 2 can reconstruct the original text.

4. The original text can be reconstructed from Piece 2, and any file into which Piece 1 has been merged. There is no necessity to preserve Piece 1 in its original form.

5. Bob can lose or destroy Piece 2, and re-create it later from the original file, using the same threshold algorithm.

Security

If Alice had Piece 1 and Piece 2 she could simply use Piece 1 to find the text phrases that match the hash-values in Piece 2. Substituting the phrases back into Piece 2 will recreate Bob's original line of text. Bob must ensure that Alice never obtains Piece 2.

The negotiation between Alice and Bob

Bob prepares threshold Pieces 1 and 2 and sends Piece 1 to Alice. Alice may require Bob to prove the authenticity of Piece 1, but Bob has no reason to care if Piece 1 is intercepted by an unauthorized party. Alice uses her software (which may be secret, or it may require computational facilities that Bob doesn't have, or it may require large databases that Bob doesn't have), to transform or annotate each phrase from Piece 1. The transformation product for each phrase can be almost anything that Bob considers valuable (e.g., a UMLS code, a genome database link, an image file URL, or a tissue sample location). Alice substitutes the transformed text (or simply appends the transformed text) for each phrase back into Piece 1, co-locating it with the original one-way hash number associated with the phrase.

Let's pretend that Alice has an autocoder that provides a standard nomenclature code to medical phrases that occur in text. The author has recently described an autocoding algorithm, which is now in the public domain (see Methods section).

Alice's software transforms the original phrases from Piece 1, preserving the original hash values. Phrases from Piece 1 that occur in the Unified Medical Language System now have been given code numbers by Alice's software.

684327ec3b2f020aa3099edb177d3794 = >
suggested (autosomal dominant inheritance=C0443147)

3c188dace2e7977fd6333e4d8010e181 = >
(mother=C0026591)

8c81b4aaf9c2009666d532da3b19d5f8 = >
manifestations

db277da2e82a4cb7e9b37c8b0c7f66f0 = >
suggested

e183376eb9cc9a301952c05b5e4e84e3 = >
(son=C0037683)

22cf107be97ab08b33a62db68b4a390d = >
(severe=C0205082)

Alice returns the coded phrase list (above) from Piece 1 to Bob. Bob now takes the transformed Piece 1 and substitutes the transformed phrases for each occurrence of the hash values occurring in Piece 2 (which he has saved for this very purpose).

The reconstructed sentence is now:

they suggested that the manifestations were as (severe=C0205082) in the (mother=C0026591) as in the (son=C0037683) and that this suggested (autosomal dominant heritance=C0443147)

The original sentence is now annotated with UMLS codes. It was accomplished without sharing confidential information that might have been contained in the text. Bob never had access to Alice's software. Alice never had the opportunity to see Bob's original text.

Implementation issues

Depending on the types of files that need to be converted into threshold pieces, some data preparation may be necessary In particular, when using actual medical records, it may be useful to encrypt or delete specific identifier fields, as listed in HIPAA. Institutions may wish to pre-process files to delete specific words, terms, or character strings from the original file. Methods for scrubbing text would apply equally to scrubbing the phrases in Piece 1. Additionally, institutions may wish to ambiguate the Piece 1 file by adding non-informative text to their original file. This may have some advantage when the original file is small or contains the records of a small number of different individuals.

The original file that is actually used by the algorithm can itself be assigned a hash number, as should Piece 1 and Piece 2. These three hash numbers could be saved and used for authentication, book-keeping or tracking purposes in later stages of a data negotiation protocol.

Issues of data space collisions arise when using very large files. A data space collision occurs when two different phrases are assigned the same hash-value by the hashing algorithm. This problem could be handled a number of different ways, including adding a subroutine that looks for collisions, computing an alternate hash value when collisions occur. The easiest way to avoid collisions is to employ a one-way hash algorithm that has a large key. The current implementation of the threshold algorithm uses

md5, which has a 128-bit key [3]. SHA (Secure Hash Algorithm) is recommended for U.S. federal agencies by NIST (National Institute of Standards and Technology) [3]. SHA256, the 256-bit version of SHA, is widely available. Using a 256-bit secure hash would mitigate issues of dataspace collisions for sub-terabyte data sets.

Discussion

Until recently, many researchers who collected confidential or proprietary data had a heavy-handed way of dealing with confidentiality issues: they denied everyone access to their data. As a result, the scientific community had no way of verifying, replicating or extending research conducted by their colleagues. The U.S. National Institutes of Health (NIH), sensing that data hoarding has become an impediment to medical progress, has promoted data sharing by NIH funded scientists. Recently, the NIH has issued a draft statement to emphasize this policy [8].

"The NIH will expect investigators supported by NIH funding to make their research data available to the scientific community for subsequent analyses. Consequently, the NIH will require that data sharing be addressed in grant applications (e.g., in sections related to significance, budget, and the end of the research plan) and in the review of applications. Funds for sharing or archiving data may be requested in the original grant application or as a supplement to an existing grant. Investigators who incorporate data sharing in the initial design of the study can more readily and economically establish adequate procedures for protecting the identities of participants and provide a useful data set with appropriate documentation. "

After the draft statement was issued, research societies requested NIH to develop techniques for data sharing that protect confidential information (i.e., identified patient records or intellectual property) [9]. In particular, concern was expressed that methods for keeping data confidential must conform to the emerging HIPAA privacy standards. Although the NIH has responded with general guidelines for protecting confidential information, no actual protocols, algorithms or implementations have been made available to the research community. The only guidelines that the author has found that in any way resemble a uniform approach to de-identification is the so-called safe harbor list of patient identifiers specified in the HIPAA privacy standards [2]. Technical approaches to medical data de-identification have recently been reviewed [10–13].

It seems obvious that if large amounts of data are to be shared among researchers, implementations are needed that can quickly render large data sets harmless to patients. In order for these implementations to be accepted by the research community, they must be freely available to test, improve or replace (with better algorithms).

What is the value of the threshold negotiation protocol?

The original text has been converted into two pieces, neither of which contain information linking patients to records. There is sufficient information in Piece 1 for Alice to annotate the text and return it to Bob (annotated Piece 1). Bob can reconstruct his original text, including Alice's annotations, thus adding value to his original data, without breaching patient confidentiality. Bob can pay Alice for her services. Alice can keep Piece 1 and use it for her own purposes. Alice can make a large database consisting of all the Piece 1 files she receives from all of her customers. Alice's aggregated Piece 1 database can be used by owners of Piece 2 files to reconstruct their original files (along with Alice's value-added annotations). Alice can sell Piece 1 to a third party, if she wishes. Alice can continually update or otherwise enhance her annotations on Piece 1 and sell the updated versions to Bob and others.

Variations on the threshold negotiation protocol

The same protocol could have been implemented in a multi-party negotiation. Bob may have been a data supplier with no interest in using the data himself. Suppose Carol was interested in Alice's annotations of Bob's file. Bob may have given Alice threshold Piece 1 and Carol threshold Piece 2. Alice may have made her transformation of the phrases in Piece 1 and sent the transformed version of Piece 1 to Carol. Carol could use Alice's transformed version of Piece 1 and her copy of Piece 2 to create a transformed version of Bob's original text. This would only work, of course, if the transformed version of Bob's original file [produced by Carol], contains no confidential information. A variation may involve assigning Bob as the trusted broker, who uses Piece 2 and the transformed version of Piece 1 to create a file for Carol. In this variation, Carol receives nothing until the end of the negotiation and Bob can take measures to ensure that the file that Carol receives is "safe."

The threshold negotiation need not be based on text exchange. The same negotiation would apply to any set of data elements that can be transformed or annotated. The threshold protocol has practical value in instances when the receiver of Piece 1 can perform a useful annotation or transformation of data without acquiring the intact data record. The protocol teases apart the data records and substitutes one-way hash values back into the record. The ways in which individual pieces of data can be transformed or annotated are limited only by the imagination. Sequences of DNA can be annotated with positional mappings or standard nomenclature or homdogy information. A local institution's tissue code could be supplemented with data obtained from a tissue database

containing experimental results performed on the tissue. Disease names can be supplemented with gene expression array data collected on tissues from other patients with the same disease.

Conclusions

A threshold protocol can render medical records harmless by dividing data sets into de-identified pieces. De-identified pieces can be safely distributed and used to enhance the value of the original data set or to share de-identified data with other researchers. The protocol facilitates compliance with Federal regulations that permit the exchange of de-identified data for research purposes and may help implement NIH's proposed data sharing policy. Most importantly, this protocol is one example of how computational methods can be used to solve legal and ethical problems faced by researchers who need access to medical data sets.

Competing interests

None declared.

Authors' contributions

This work represents the opinions of the author and does not represent the policy of NIH or any other U.S. Federal Agency.

All of the supplemental files needed to implement and replicate the algorithms described in the article are listed in Methods and can be freely downloaded from the URLs cited in Methods and References.

Acknowledgements

This work was conducted at the NIH as part of the author's customary work activities, and no specific financial support was received for this work.

References

1. **Title 45 CFR (Code of Federal Regulations), Part 46. Protection of Human Subjects; Common Rule.** *Federal Register* 1991, **56:**28003-28032
2. **Title 45 CFR (Code of Federal Regulations). Parts 160 and 164. Standards for privacy of Individually Identifiable Health Information; Final Rule.** *Federal Register* 2002, **67:**53181-53273
3. Schneier B **Applied Cryptography: Protocols, Algorithms and Source Code in C.** *New York, Wiley* 1994,
4. **PubMed Help.** [http://www.ncbi.nlm.nih.gov/entrez/query/static/help/pmhelp.html]
5. Moore GW, Miller RE and Hutchins GM **Indexing by MeSH titles of natural language pathology phrases identified on first encounter using the barrier word method.** *In: Computerized Natural Medical Language Processing for Knowledge Representation (Edited by: Scherrer JR, Cote RA, Mandil SH) Amsterdam, North-Holland* 1989, 29-39
6. **Omim™ Online Mendelian Inheritance in Man.** [http://www.ncbi.nlm.nih.gov/Omim/]
7. **Unified Medical Language System (UMLS).** [http://www.nlm.nih.gov/research/umls/umlsmain.html]
8. **NIH Draft Statement on Sharing Research Data.** 2002, [http://grants1.nih.gov/grants/guide/notice-files/NOT-OD-02-035.html]
9. **Comment Letter on NIH Data Sharing Proposal from the American Association of Medical Colleges.** [http://www.aamc.org/advocacy/library/research/corres/2002/051102.htm] May 10, 2002
10. Sweeney L **Replacing Personally-Identifying Information in Medical Records, the Scrub System.** *In: Proceedings, Journal of the American Medical Informatics Association. (Edited by: Cimino JJ) Washington, DC: Hanley & Belfus, Inc* 1996, 333-337
11. Moore GW and Berman JJ **Anatomic Pathology Data Mining.** *In: Medical Data Mining and Knowledge Discovery (Edited by: Cios KJ) Berlin, Springer-Verlag* 2000,
12. Quantin C, Bouzelat H, Allaert FA, Benhamiche AM, Faivre J and Dusserre L **Automatic record hash coding and linkage for epidemiological followup data confidentiality.** *Methods Inf Med* 1998, **37:**271-277
13. Berman JJ **Confidentiality for Medical Data Miners.** *Artificial Intelligence in Medicine* 2002, **26:**25-36

Pre-publication history

The pre-publication history for this paper can be accessed here:

http://www.biomedcentral.com/1471-2288/2/12/prepub