Open Access and Database Anonymization an Open Source Procedure Based on an Italian Case Study

Luca Leschiutta* Giuseppe Futia**

*Nexa Center for Internet & Society – DAUIN – Politecnico di Torino, Italia, luca.leschiutta@polito.it
**Nexa Center for Internet & Society – DAUIN – Politecnico di Torino, Italia, giuseppe.futia@polito.it

Abstract: *The only method, believed to be compliant to privacy laws, to open a database that contains personal data is anonymization. This work is focused on a car accident database case study and on the Italian DP law. Database anonymization is described from a procedural point of view and it is explained how it is possible to complete the whole process relying solely on widespread open-source software applications. The proposed approach is empirical and is founded on the letter of the Italian privacy law.*

Keywords: *Open-Access, Open-Source, Data-Base, Case-Study*

Acknowledgement: *The authors would like to thank the CSI-Piemonte (the Piedmont Consortium for ICT System) for sharing with us the structure of the TWIST database that is the object of our case study.*

Introduction

Data protection laws are one of the biggest impediments to grant open access to databases that belong to Public Administrations. Particularly in Italy, taking into account the Personal Data Protection Code with respect to data dissemination, such concerns are completely justified given the pecuniary and custodial penalties foreseen for incorrect personal data processing.

We don't have enough room here to go into the details of the Italian Data Protection Code nor to analyze other relevant European laws but the general assumption is that, in most cases, it is forbidden to share with the public databases containing personal data. One important deviation from what is stated above comes from Legislative Decree no. 33 of 14 March 2013 about Public Administrations' transparency in which is foreseen that PAs must publish several data such as organigrams and costs.

Notwithstanding the a.m. decree, the principal way to openly share a database is to remove all data that could lead to the identification of the involved subjects. This operation, also known as Database Anonymization, is object of this work. We will face the problem from a very procedural point of view and we'll show how, under certain conditions, all the involved operations can be performed using solely widespread open-source software applications.

This work was developed in the framework of the Open-DAI[1] project. Open-DAI is "Opening Data Architectures and Infrastructures" for European Public Administrations. It is a project funded under the ICT Policy Support Programme as part of the Competitiveness and Innovation framework Programme (CIP) Call 2011.

Our study is based on a real case in which a data base consisting of 352 data fields of car accidents related data (TWIST) needs to be open accessed.

TWIST is owned by Piedmont Region and managed, provided and maintained by CSI[2]

Within the Open-DAI project it is foreseen to open the data base (integrated with other sources) in order to create an application that will use statistical road accident and traffic data to propose better paths to the end user.

Database Anonymization

Hereafter we'll describe a procedure on how to process and anonymize a collection of data that includes personal, sensitive and judicial data. It is worth mentioning that anonymizing a database does not mean to simply throw away the most sensitive information but, in most cases, it is mandatory to retain the capability to restore the removed data at a later time.

The procedure is general purpose and implemented relying solely on common open-source software applications. The actual instructions to operate both on Windows and Linux operating systems are sketched.

An Easy Data-Set Anonymization.

Let's suppose that our data-set is in a single table named *NonAnonymousData.csv* and that the various items have no correlation. Let's proceed with the well-known *Libre Office* suite. In the first step the Identification Data (ID) are grouped on the left of the table and the Non Identification Data (NID) are grouped on the right.

Table 1: Ordered NonAnonymousData.csv

ID1	ID2	ID3	ID4	NID1	NID2	NID3	NID4
Item 1							
Item 2							
Item N							

In the second step: a column containing Anonymous ID (AID) is added. AIDs are numeric strings extracted from a list of 10*N random generated numbers (see below) and saved in *NonAnonymousData.csv*

Table 2: Ordered NonAnonymousData.csv including Anonymous IDs

ID1	ID2	ID3	ID4	AID	NID1	NID2	NID3	NID4
Item 1				1053				
Item 2				1001				
				1057				
Item N				1133				

In the third step the table is stripped of the NIDs and saved as *AnonymousData.csv*

[1] http://www.open-dai.eu/

[2] CSI-Piemonte il Consorzio per il Sistema Informativo piemontese: http://www.csipiemonte.it/en/

Table 3: AnonymousData.csv

AID	NID1	NID2	NID3	NID4
1053				
1001				
1057				
1133				

Depending on the conditions and on the legislation the *NonAnonymousData.csv* file shall be either completely destroyed or kept hidden and safe for a very long time

The operation in which you safely destroy some data is called "*data wiping*" and consists of several passes in which the desired portion of hard disk is overwritten with random data. To perform these operations, On Windows you can use the open source program *Eraser*[3]. On Linux you can use the following commands:

```
> shred NonAnonymousData.csv
> rm NonAnonymousData.csv
```

If you have to keep the data hidden and safe for a very long time, you must:

1. Cryptograph[4] the file on Windows. This can be achieved by using the open source 7zip[5] program that allows to achieve a strong AES-256 encryption. On Linux you can use the following command:

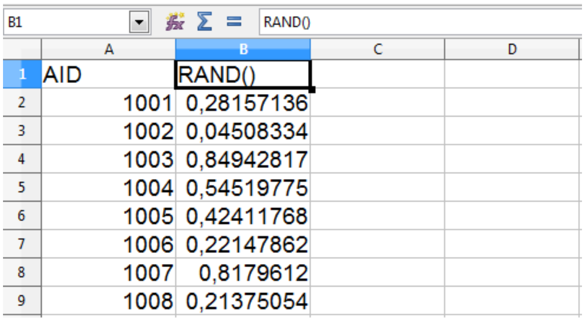
   ```
   > gpg -c NonAnonymousData.csv
   ```

   In both cases, the non-encrypted *NonAnonymousData.csv* file must then be destroyed using the above mentioned procedure and the password must be chosen and preserved with the usual due diligence.

2. The encrypted file must then be backed up to a safe location e.g. a non-rewritable DVD or a WORM (Write Once Read Many) tape

Random AIDs Generation

Hereafter is described an easy procedure to generate N (for simplicity sake we'll assume N=100) unique AIDs with LibreOffice. This procedure might not be the most efficient but does not require any dedicated software nor any high level IT skill:



B1 RAND()

	A	B	C	D
1	AID	RAND()		
2	1001	0,28157136		
3	1002	0,04508334		
4	1003	0,84942817		
5	1004	0,54519775		
6	1005	0,42411768		
7	1006	0,22147862		
8	1007	0,8179612		
9	1008	0,21375054		

Figure 1: random number generation

Open a new Calc spreadsheet and write into column A the subsequent number within 1000 and 1999

1. In column B generate as many random numbers by means of the RAND() function
2. Reorder according to Column B
3. Copy the first 100 AIDs into the relevant dataset column (see Table 3. above).

[3] http://eraser.heidi.ie

[4] A cryptographic software shall have the following specifications:

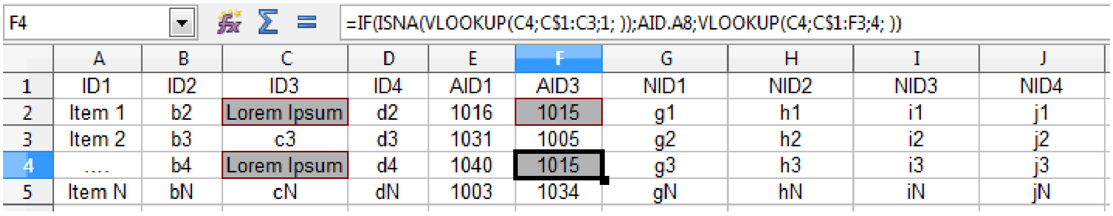
1. Standard file format and cryptography algorithm so that the file will always be recoverable
2. Open source to assure the highest reliability

[5] http://www.7-zip.org/

Advanced Techniques: Repeating Ids

The above mentioned technique works well for a very simple database in which all IDs are unique but, in real cases, is quite common the situation depicted in the ID3 column of the following figure (e.g. in our case study we have repeating hospitals. We do not want to put this information in clear – it would be too easy to track the guilty driver hospital location given the crash location – but we do not want to lose entirely the information)

In this case, should we apply the above described technique, we would lose some correlational information. The solution consist in anonymize and keep the ID3 column using the same ID for repeating data (e.g. in the following figure AID3 is 1015 every time "Lorem Ipsum" is encountered in column ID3)



F4

=IF(ISNA(VLOOKUP(C4;C$1:C3;1;));AID.A8;VLOOKUP(C4;C$1:F3;4;))

	A	B	C	D	E	F	G	H	I	J
1	ID1	ID2	ID3	ID4	AID1	AID3	NID1	NID2	NID3	NID4
2	Item 1	b2	Lorem Ipsum	d2	1016	1015	g1	h1	i1	j1
3	Item 2	b3	c3	d3	1031	1005	g2	h2	i2	j2
4	....	b4	Lorem Ipsum	d4	1040	1015	g3	h3	i3	j3
5	Item N	bN	cN	dN	1003	1034	gN	hN	iN	jN

Figure 2: NonUniqueIDs.csv

In this case, the *Libreoffice* formula

IF(ISNA(VLOOKUP(C4;C$1:C3;1;));AID.A8;VLOOKUP(C4;C$1:F3;4;))

is more complex and some explanations are due: For each ID in column ID3 the formula looks in the above IDs to detect any repetition. If the ID is not a repetition, a new AID is inserted in column AID3, otherwise the same AID used before is inserted in column AID3.

Even more complex is the case described in the following table in which we have cross-correlation between various columns.

Table 3: NonUniqueIDsInMultipleCells.csv

ID1	ID2	ID3	ID4	NID1	NID2	NID3	NID4
Item 1		Lorem ipsum					
Item 2							
		Lorem ipsum					
Item N			Lorem ipsum				

In this case a variation of the above mentioned technique can be used but, since *Libreoffice* is not capable of multidimensional lookups, some code should be written according to the following pseudo-code snippet:

```
flag=false;
for (i=0; i<n: i++){
  for (j=0; j<m: j++){
          if(ID_Matrix[i][j]==ID_Matrix[n][m]{
                  AID_Matrix[n][m] = AID_Matrix[i][j];
                  flag=true;
                  break;
          }
  }
}
```

```
if (flag==false){
  AID_Matrix[n][m]=Next_Availabe_AID(k);
  k++;
}
```

Advanced Techniques: Data Degradation

In some cases, we might wish to retain some information but we do not feel confident that the database, in this way, will have the desired level of anonymization. In these cases data degradation might be performed. No general purpose technique comes to our mind w.r.t. data degradation, but our case study can provide some insight:

- In our database the accident location is provided with extreme accuracy thanks to GPS longitude and latitude data that have the precision of the second of degree (e.g. 45° 03.866′ N) which lets individuate a location with the accuracy of a few meters. If we round up this numbers to the tenth of a minute of a degree (e.g. 45° 03.9′ N), we obtain an accuracy of roughly 10 km
- In our database the accident time is given exactly to the minute (e.g. 10:45 of the 12th of November 2012) . In this case we could drop the information relevant to the minute, the day of the month and we could degrade the month to the season of the year (e.g 10 o'clock, winter 2012).

The De-anonymization Test

Finally we tried a little de-anonymization experiment on our test case: working on a data-set in which obvious fields such as vehicle registration plates, driving license numbers, people's names had been removed, we found out that it is quite easy to find the complete name of involved people (especially if deceased after the accident) relaying on other fields such as:

- Accident location (actually we are dealing with several fields that consent to locate quite precisely where the crash took place)
- Accident time
- Number of injured people
- Number of losses

The trick consists simply to use "*name of the street*", "*date of the accident*" and "*accident*" as search parameters in google to gather several informative piece of news from local papers. In this example the data degradation, described above, would guarantee a much better anonymization.

Conclusions

The analysis of the Italian Personal Data Protection Code would show how daunting is the road that leads to databases open access for a PA. We believe that similar considerations can be extended to other European jurisdictions given the ongoing effort to harmonize the various privacy laws

The only viable solution to this problem is database anonymization and we have provided a step by step procedure that we consider straightforward and cost-effective. The use of open-source software, besides the economic considerations, is also important for the cryptographic point of view.

We have barely addressed the more difficult problem that is how to test if the anonymization level of the database is sufficient. In our test case, we found out that removing all personal data is not enough because with "*reasonable means*" we were able to counteract the anonymization.

The solution of the "de-anonymization party" working on several test cases to break the anonymization effort, if well documented, seems to us in line with the law requirements ("*the means possibly required to effect identification are to be considered disproportionate compared with the damage resulting*").

Several aspects remain open to questioning especially in cases, such as the one we analyzed into this work, in which sensitive and judicial data are involved.

One important question is whether a database of sensitive and judicial data, even if cleaned of any reference to personal information, is still object of the DP code?

Furthermore, should the above be true, opening the database, having in mind that someone might be able to exploit the data for some unforeseen application or service that does not fall in the category of statistical and scientific purposes, is it not contradictory to the provisions that ask to have the consent from the subject for each specific treatment?

About the Authors

Luca Leschiutta

Luca Leschiutta is the IT manager of the Nexa Center for Internet & Society of the Politecnico di Torino. After graduating in Electronic Engineering, he pursued a PhD in Information Technology at the Internet Media Group of the Politecnico di Torino. His fields of study have been image compression and wireless networking. He also taught programming and networking courses. In the past he worked as a reliability engineer at Alenia Spazio in the ISS project. Since 2010 he also works in a similar position at the Human Genetics Foundation of Torino.

Giuseppe Futia

Giuseppe Futia is the communication manager of the Nexa Center for Internet & Society of the Politecnico di Torino. He holds a Master Degree in Cinema and Media Engineering and since 2008 he collaborates regularly with La Stampa daily. At the Center he is in charge of communication and press office. More specifically, Giuseppe's main responsibilities consist of keeping contact with media and following Nexa's Image.