A brief overview of basic inference attacks and protection controls for statistical databases

O. DZIĘGIELEWSKA, B. SZAFRAŃSKI

olga.dziegielewska@wat.edu.pl, boleslaw.szafranski@wat.edu.pl

Military University of Technology, Faculty of Cybernetics
Institute of Computer and Information Systems
Kaliskiego Str. 2, 00-908 Warsaw, Poland

With cyber-attacks on the dramatic rise in the recent years, the number of entities which realize the necessity of protecting their IT assets increases. Individuals are more aware of the potential threats and demand high level of security from the business entities having access to their personal and private data. Such entities have legal obligations to satisfy the confidentiality when processing sensitive data, but many fails to do so. Keeping the statistical data private is a challenge as the approach to the security breaches slightly differs from the classical understanding of data disclosure attacks. The statistical disclosure can be achieved using inference attacks on the not-effectively protected assets. Such attacks do not target the database access itself, i.e. are performed from a perspective of an internal user, but the statistical interface used to retrieve the statistical data from the database records. This paper sums up basic types of inference attacks classifying them in the CVSS standard and provides a series of fundamental countermeasures which can be undertaken to mitigate the risk of performing successful attack.

Keywords: statistical databases, inference control in statistical database, statistical disclosure, data security, data privacy.

1. Introduction

Regardless of the main purpose of the statistical database it always holds sensitive data identifying individuals or entities which provided the input information. Despite the fact that the input data itself must be kept satisfying privacy requirements, the statistical disclosure controls implemented in the database must prevent legitimate system users from gaining unauthorized access to some sensitive statistic about a particular individual using inference attack methods.

It is assumed that the statistical database users cannot access the data repository directly but using statistical interface which allows only aggregation queries, e.g. sum, average, count. The inference attacks use security breaches in the implemented statistical interface restrictions to retrieve the sensitive statistics.

There is no universal method for protecting the statistical processing, however there are statistical disclosure controls which when implemented properly can significantly decrease the risk of releasing confidential data from the database.

2. Related work

The statistical databases security research is almost as long-established as the discipline itself. To the date countless overviews, reviews and critics describing the security approach has been released. One of the earliest comprehensive breakdowns of the security protection methods is *Security-control methods for statistical databases*: *A comparative study* published in 1989 (Nabil, Worthmann 1989) in which the authors focus on the existing security--controls.

With rapidly increasing number of digital data storages, the research in the area of statistical databases has also developed significantly since the early 2000. In 2002 the first conference dedicated to the privacy in the statistical databases was held and since then became a biennial event organized by UNESCO Chair in Data Privacy. Most recent advances in the security aspect of the statistical databases can be followed based on the conference papers [5].

The overviews of the inference control in statistical databases has been published in Advances in Inference Control in Statistical Databases: An Overview (Domingo-Ferrer, 2002) and A Survey of Inference Control

Methods for Privacy-Preserving Data Mining (Domingo-Ferrer, 2008).

3. Proposed statistical disclosure vulnerability rating

The data privacy is the key aspect in the security of statistical databases. As always, the basic Confidentiality, Integrity and Availability triad can be used to measure the risk levels of particular threats, however using more comprehensive approach helps to identify and rank the potential vulnerabilities in a more effective way.

Currently one of the most widely used frameworks for assigning the risk levels is CVSS (Common Vulnerability Score System). Despite its limitations and distinct main purpose, i.e. application for calculating the vulnerabilities' risk levels for the real-life systems, it can be also used to measure the risk levels of the general attack scenarios, as those described in the next chapter, on condition that some assumptions are made. This rating method was selected to point out the crucial security aspects of the attacks and show similarities and differences between them in a standardized form.

The most common division of the statistical databases gives 4 main categories with 2 types in each:

1. Responsiveness
   - Immediate [RI]: the queries are processed in real time.
   - Delay [RD]: the queries are processed with some specified delay.
2. Immutability
   - Dynamic [ID]: the records can be updated multiple times.
   - Static [IS]: the records always remain the same, no alterations are possible.
3. Distribution
   - Centralized [DC]: there is only one central database.
   - Decentralized [DD]: multiple copies of the database are maintained.
4. Purpose
   - Dedicated [PD]: the system serves only the database.
   - Shared [PS]: the system serves the database, the database management system and other installed software.

The database type influences the risk level for all the general attacks scenarios described in this paper and must be taken into account when analyzing the feasibility of performing a particular attack and calculating its CVSS scores.

For the purpose of this paper only CVSS v3.0 base score parameters will be analyzed as database-type-dependent:

- attack vector (AV)
- attack complexity (AC)
- privileges required (PR)
- user interaction (UI)
- scope (S)

The temporal score will not be taken into account as the security of commercial and non-commercial database implementations is not the subject of the paper.

The confidentiality (C), integrity (I) and availability (A) of the base score measure the impact of a particular vulnerability, therefore they remain independent of the database type until the attack impact analysis phase.

To calculate the CVSS scores of the analyzed attacks, it was assumed that:

- the database can be accessed from adjacent network (affects AV),
- read-only or statistical access is a low privilege (affects PR),
- read and write access is a high privilege (affects PR).

However, it must be noted that the given risk metrics are only considering the general case and in real life systems the scores might be moderately different as additional mitigation or increase metrics might be applied.

4. Basic attack methods

It is assumed that the statistical database is compromised when it reveals a sensitive statistic which allows to identify confidential information of an individual.

The fundamental attack methods for the statistical databases compromise can be divided into two groups:

a) *Static* – The internal state of the database does not change; an attacker infers from the results of the queries performed on the database. Read-only or statistical access is sufficient to perform such attack.
b) *Dynamic* – The internal state of the database changes, an attacker infers from the delta of the results of the same query performed on the original and modified data set. Read-write access is necessary to perform this attack.

4.1. Static

Static attacks are performed by querying the database with specially prepared statements to retrieve information about entries that satisfy some characteristic C and inferring the sensitive information from the results.

The first type of static attack is small and large query set attack [*S1*] which exploit lack of restriction on retrieving statistic based on small query sets, i.e. if a database does not restrict releasing characteristics which identify small group of entities, an attacker can learn a sensitive statistic about a selected entity (Hoffman, Miller, cited in Denning 1982). The attack is performed in the following steps:

1. An attacker knows that the entity E satisfies some characteristic C_1 represented in the database.
2. Queries the database with: $Q_1 = count(C_1)$ to verify the number of entities satisfying the characteristic.
3. Then by adding a new characteristic $C2$ to the query: $Q_2 = count(C_1 \cdot C_2)$ and comparing the results from both Q_1 and Q_2 attacker gains information if E satisfies also C_2.

The same attack is also possible on the large query sets for the databases which permits complementation in the queries.

The second type of static attacks is the linear system inference [*S2*] (Denning, 1982), which bases on generating such queries that individually do not reveal sensitive statistic, but when put into a set of equations, some sensitive statistic about a selected entity can be released. The attack involves solving a system of equations $HX = Q$, where H is a binary values matrix which indicates if a record j has a characteristic C_i:

1. $h_{ij} = 1 \; if \; j \in C_i$
2. $h_{ij} = 0 \; otherwise$
3. $1 \leq i \leq m, 1 \leq j \leq N$

Q is a column vector of values of the known statistics: $Q = [q_1, \dots, q_m]$ and $X = [x_1, \dots, x_N]$ is also a column vector, where m is the number of released characteristics and N is the number of total records taken into account.

One type of linear system attacks is the tracker attacks group. This kind of attacks use padding to add some extra records in the small query sets, to bypass the query-set-size control restriction [*N1*], and later dismisses the padding results to retrieve the sensitive statistic.

The individual tracker attack [*S2I*] (Schlorer cited in Denning 1982) targets some specified entity E which can be identified by a characteristic C_1 and an attacker want to verify whether E satisfies also characteristic C_2. If the database has implemented *N1* to suppress the *S1* attack, then the attacker must divide the query in such way that the statistic returned by it is permitted by the control and at the same time allows to calculate searched sensitive statistic C_2. The attack is executed in the following steps:

1. Find a decomposition of C_1: $C_1 = C_1^1 \cdot C_1^2$, such that $count(C_1^1)$ and $count(C_1^1 \cdot \neg C_1^2)$ are permitted.
2. Compute $count(C_1 \cdot C_2) =$
$$= count\left((C_1^1 \cdot \neg C_1^2) + C_1^1 \cdot C_2\right) - count(C_1^1 \cdot \neg C_1^2)$$
3. If $count(C_1 \cdot C_2) = 0$, then E does not have C_2. If $count(C_1 \cdot C_2) = count(C_1)$, then it has C_2.

The pair $\{C_1^1, C_1^1 \cdot \neg C_1^2\}$ is called an individual tracker. Individual trackers are different for each target entity; however, it was proven that each database has at least one general tracker [*S2G*] which allows to retrieve every sensitive statistic about each of the entities.

The third type of static attacks are selection attacks [*S3*] in which an attacker selects some values, e.g. minimum, median or maximum, from two or more query result sets, and based on them infers a sensitive statistic (Denning, 1982).

To perform such attack, an adversary must find two queries which retrieve results of two distinct characteristics C_1 and C_2 and which have the same selection value V of some statistic S and a single common record E. Then V will be the value of the statistic S for the record E. The condition can be summarized as follows:

1. $C_1 \cdot C_2 = E$
2. $V = selection(C_1, S)$
3. $V = selection(C_2, S)$
4. $V = S(E)$

4.2. Dynamic

Dynamic attacks assume analyzing changes of the database query results at the database entry modification, the insertion or the deletion of an entry set.

The first scenario [*D1*] is bypassing query-set-size restriction [*N1*] by adding some number of dummy entries which satisfy the original

query or the complementation of the original query, i.e.:

1. $if\ |C| < n, then\ R(C) = R(C) + D(C)$
2. $if\ |C| > N - n, then\ R(C) = R(C) + D(\neg C)$

The next scenario [*D2*], also based on insertions, analyzes the changes of the same query results before and after inserting a new record into the database. The whole process can be described in the following steps:

1. $R_{before} = R(C)$
2. $insert\ i$
3. $R_{after} = R(C)$
4. $R(i) = R_{after} - R_{before}$

A slight modification of the second scenario is analyzing changes in the $R(C)$ statistics after insertion or deletion of a record from a database.

4.3. Vulnerability scores for the attacks

The CVSS base scores set all the described attacks at medium level with the risk values ranging from 4.5 to 5.7.

However, taking into account that confidentiality, integrity and availability of the data in described systems is crucial the requirements for these parameters was set to high in the CVSS environmental score attributes. After adding these requirements, the vulnerability score significantly increased for each attack, with the highest value of 7.5 and high risk value for *S1* and *S2G*.

As previously discussed, the vulnerability score in this paper considers the most general case and despite it can be used as is, it is recommended to adhere the scores in the real-life applications taking into account the real system characteristics and implemented configurations.

Tab. 1. Proposed CVSSv3.0 scores for static attacks

Attack ID	Base score	Vector string
S1	5.7	CVSS:3.0/AV:A/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N
S2G	5.7	CVSS:3.0/AV:A/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N
S2I	4.8	CVSS:3.0/AV:A/AC:H/PR:L/UI:N/S:U/C:H/I:N/A:N/
S3	4.8	CVSS:3.0/AV:A/AC:H/PR:L/UI:N/S:U/C:H/I:N/A:N/

Tab. 2. Proposed CVSSv3.0 scores for dynamic attacks

Attack ID	Base score	Vector string
D1	5.2	CVSS:3.0/AV:A/AC:L/PR:H/UI:N/S:U/C:H/I:L/A:N
D2	4.5	CVSS:3.0/AV:A/AC:L/PR:H/UI:N/S:U/C:H/I:N/A:N

Tab. 3. Proposed CVSSv3.0 environmental scores for the attacks

Attack ID	Environmental score	Vector string
S1	7.5	CVSS:3.0/AV:A/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N/CR:H/IR:H/AR:H
S2G	7.5	CVSS:3.0/AV:A/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N/CR:H/IR:H/AR:H
D1	6.7	CVSS:3.0/AV:A/AC:L/PR:H/UI:N/S:U/C:H/I:L/A:N/CR:H/IR:H/AR:H
S2I	6.6	CVSS:3.0/AV:A/AC:H/PR:L/UI:N/S:U/C:H/I:N/A:N/CR:H/IR:H/AR:H
S3	6.6	CVSS:3.0/AV:A/AC:H/PR:L/UI:N/S:U/C:H/I:N/A:N/CR:H/IR:H/AR:H
D2	6.3	CVSS:3.0/AV:A/AC:L/PR:H/UI:N/S:U/C:H/I:N/A:N/CR:H/IR:H/AR:H

5. Basic protection methods

The protection methods, also called Statistical Disclosure Controls (SDC), against the described attacks can be divided into two groups (Willenborg, DeWaal cited in Domingo-Ferrer 2008):

a) *Perturbative* – The query result sets are slightly altered before the release in such a way that the computed statistics do not significantly differ from the original result, yet protect the privacy of potentially sensitive statistics
b) *Non-perturbative* – The query result sets are not altered. The protections is achieved by restricting statistics.

Each of the control mechanisms, when analyzed or implemented separately, might not be efficient and appropriate for each type of

the database and prevent existing attack vectors. Therefore, in the real-life systems the protection methods are combined.

5.1. Perturbative

The first perturbative protection method is rounding [*P1*]. This method replaces either original values or original results with rounded values (Domingo-Ferrer, 2008). The rounding function is the crucial decision factor, as the level of security of this control method depends on it. If the method returns predictive values, an attacker might deduct the sensitive statistic from sufficiently large sample.

Data swapping and rank swapping [*P2*] is a transformation which exchanges values between individual records in such a way that the individual sensitive statistic cannot be retrieved but the statistical accuracy of the returned result is maintained. A variant of the data swapping is rank swapping which sorts the values of the characteristic C in ascending order and later the values are interchanged within the scope of its rank. Ranks are defined as percentage ranges of all the C values.

5.2. Non-perturbative

One of the most basic non-perturbative controls is query-set-size control [*N1*] which restricts such results that have less than n or more than $N-n$ records for some positive integer n (Denning, 1982). This way only the easiest attacks are prevented, e.g. *S1*, because this control can be easily bypassed.

Another non-perturbative control is maximum-order control [*N2*], which does not allow queries which includes too many parameters (Denning, 1982). The number of the parameters allowed in a single query should be defined separately for every database in a process which finds the minimum number of attributes that allows to identify a sensitive statistic of a particular entity. The drawback of this method is that it restricts many non-sensitive statistics and with each change in a database the process of finding the maximum of allowed parameters must be repeated.

The suppression mechanism [*N3*] dismisses such values from the query set result which can be categorized as sensitive and those which are non-sensitive, but might allow to derive a sensitive statistic from the retrieved data (Denning, 1982). The suppression criterion used in this method for the $count(C)$query is a minimum query-set size and for the $sum(C)$ query is then$-\ respondent\ k\%-dominance$ rule, i.e. a sensitive statistic is calculated with n or less values which make up more than $k\%$ of the total values.

Another control method is sampling [*N4*] in which different sample records are used to compute the queried statistic, i.e. the sample records of the original query result set are used in the result set released from the database (Domingo-Ferrer, 2008).

Last analyzed protection method is generalization [*N5*] in which several data categories are combined into one more general to increase the number of entities used for calculating the statistic (Rubin, Samarti cited in Domingo-Ferrer 2008). The special case of this control is top-bottom coding used for the categories which can be ranked. In this case, minor categories are grouped in major categories by the rank values, i.e. top values and bottom values are combined in separate groups (Domingo-Ferrer, 2008).

5.3. Protection mechanisms application

Described protection mechanisms can be applied to mitigate but not eliminate the risk of succeeding in performing attacks from the previous chapter. The table below maps the protection method with the attacks it targets.

The overall risk of a particular attack can be calculated taking into account the number of adequate protection methods implemented in a real-life system and their security completeness level.

Tab. 4. The table shows if a particular control mitigates the risk of each attack

	S1	**S2**	**S3**	**D1**	**D2**
P1		■	■	■	■
P2		■	■	■	■
N1	■	■			
N2		■	■		
N3	■	■	■		
N4		■	■	■	■
N5			■	■	

6. Summary

The paper presents the fundamental protection and attack methods on statistical database systems in a form of cohesive classification which can be used as a starting point of

understanding the importance and deeper analysis of the privacy in such systems. As of today, the commercially used database systems do not provide full statistical data protection components despite the importance the data privacy currently has in the global scale.

7. Bibliography

[1] Denning D., *Cryptography and Data Security*, Addison-Wesley Publishing Company, Inc., Boston, 1982.
[2] Domingo-Ferrer J., "A Survey of Inference Control Methods for Privacy-Preserving Data Mining", in: *Privacy-Preserving Data Mining: Models and Algorithms*, Aggarwal Ch.C., Yu P.S. (eds.), in series: Advances in Database Systems, Vol. 34, pp. 53–80, Springer, 2008.
[3] Domingo-Ferrer J., "Advances in Inference Control in Statistical Databases: An Overview", in: *Inference Control in Statistical Databases*: *From Theory to Practice*, Domingo-Ferrer J. (eds.), LNCS 2316, pp. 1–7, Springer, 2002.
[4] Adam N.R., Worthman J.C, "Security-control methods for statistical databases: A comparative study", *ACM Computing Surveys*, Vol. 21, No. 4, 515–556 (1989).
[5] Privacy in Statistical Databases, http://dblp.uni-trier.de/db/conf/psd/

Krótki przegląd podstawowych ataków sterowania wnioskowaniem i metod ochrony dla statystycznych baz danych

O. DZIĘGIELEWSKA, B. SZAFRAŃSKI

Wraz ze wzrostem liczby cyberataków w ostatnich latach, zwiększa się również liczba podmiotów, które uświadamiają sobie konieczność ochrony swoich zasobów teleinformatycznych. Ludzie są coraz bardziej świadomi potencjalnych zagrożeń i wymagają wysokiego poziomu bezpieczeństwa od instytucji, które mają dostęp do ich poufnych danych. Podmioty przetwarzające dane wrażliwe podlegają procedurom i regulacjom prawnym nakładającym obowiązek zachowania poufności przy przetwarzaniu danych wrażliwych, ale wiele z nich nie robi tego skutecznie. Podejście do łamania zabezpieczeń i utraty poufności w statystycznych bazach danych różni się od klasycznego rozumienia ataków skierowanych na ujawnienie danych wrażliwych i dlatego osiągnięcie pełnej poufności danych statystycznych jest nadal wyzwaniem. Naruszenie poufności danych statystycznych może zostać osiągnięte za pomocą ataków sterowania wnioskowaniem skierowanych na nieskutecznie zabezpieczone bazy danych. Takie ataki nie są kierowane na sam dostęp do bazy danych, są wykonywane z perspektywy użytkownika z prawami dostępu do bazy danych i skierowane na interfejs statystyczny wykorzystywany do pobierania danych statystycznych z bazy danych. Artykuł podsumowuje podstawowe typy ataków sterowania wnioskowaniem, klasyfikując je w standardzie CVSS, i omawia kluczowe metody ochrony, które mogą zostać wykorzystane w celu zmniejszenia ryzyka przeprowadzenia skutecznego ataku.

Słowa kluczowe: statystyczne bazy danych, sterowanie wnioskowaniem, ujawnienie danych statystycznych, bezpieczeństwo danych, poufność danych.