A NEW METHOD OF INFERENCE CONTROL
FOR STATISTICAL DATABASES

By

SHAHID RASHID MALIK

Bachelor of Engineering

N. E. D. University

Karachi, Pakistan

1982

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
December, 1988

A NEW METHOD OF INFERENCE CONTROL
FOR STATISTICAL DATABASES

Thesis Approved:

Huizhu Lu

Thesis Adviser

J Chandler

G. E. Hedrick

Norman N. Durham

Dean of the Graduate College

ACKNOWLEDGEMENTS

I wish to express personal gratitude to my major advisor, Dr. H. Lu, for her intelligent guidance, assistance, and motivation throughout this study. The undertaking of this study would never have been initiated had it not been for her lectures in the Information Storage and Retrieval class.

I want to thank my graduate advisor and committee member Dr. G. E. Hedrick, for his guidance and help not only during the thesis but throughout my stay at Oklahoma State University.

My appreciation for my committee member Dr. J. P. Chandler, whose knowledge and grasp of the subject helped me understand the topic better.

I extend my appreciation to my friend, Troy, without whose help the word processing of this thesis would have become a lot more difficult and a lot less enjoyable.

I reserve a very special thank you to my mother, for her love and prayers, and to my father, for his love, encouragements, understanding, and above all for being what he is.

Many thanks to my sister, sister-in-law, and niece, Mano, to my brothers, especially Khalid, for their love and moral support they all showed that has kept me going in my quest for knowledge, thousands of miles away from my home.

I want to thank my wife, for being so understanding, patient, and for all her love and concern, she showed in her always awaited letters.

Finally, it was you my lovely daughter, Hira, whom I missed every moment of these 28 months. Your cute little face has always been a source of great joy, and your sweet smile has always refreshed me after long and never ending hours of study. With all my love, I dedicate this thesis to you.

TABLE OF CONTENTS

Chapter page

I. INTRODUCTION. 1

II. REVIEW OF RELATED WORK. 6

Control on the Size of Query Set 7
Individual Tracker Compromise 8
General Tracker Compromise. 10
Control on the Overlap of Query Set. 11
Distorting Data or Query Response 12
Random Samples 15
Selection Methods. 16

III. THE NEW METHOD OF INFERENCE CONTROL 20

IV. EVALUATION OF THE NEW METHOD. 25

V. SUMMARY AND CONCLUSION. 36

BIBLIOGRAPHY 38

LIST OF TABLES

Table Page

I. Hypothetical Database. 18

II. Formal Queries and Informal Statements 19

III. Referenced Database. 35

LIST OF FIGURES

Figure Page

1. Logical Flow Diagram of the New System 24

CHAPTER I

INTRODUCTION

Statistical databases typically contain sensitive and confidential information about individuals and enterprises. The information might, for example, be obtained from medical records or from a population census. The purpose of the database is to provide statistical summaries of the information in response to user queries to support activities such as economic planning or scientific research.

The U.S. legal code requires that sensitive information associated with individuals be protected from unauthorized release. At the same time, those data should be available to the public for statistical analysis. No sequence of queries should be sufficient to deduce further information about any individual described in the database. Determining, then enforcing, a policy specifying what information in a database can be given in response to queries is the database security problem [1, 2].

Inference control in statistical databases is an important issue, since many current types of research are dependent heavily upon statistical data that must preserve the confidentiality of individual information. A database whose information may be deduced by a finite mechanism is said to be compromised. Dobkin, et al [3], show that users can compromise databases by simply asking a series of statistical queries.

For example, a database of employee salaries; the salary of each employee is stored along with other key information concerning the employee. Normal system protocol may restrict user to query the system in the following way:

"What is the median salary of {Si} ?" ... (i)

where "{Si}" specifies a set of K employees. An answer to such a query is the value of the median salary, but not who earns it. The database is compromised by the user if the user can determine some employee's salary. In [3], it is shown that even if queries are restricted to the form (i) and the answers are true, it always is possible to compromise the database in (3/2(k+1))+1 queries [23]; where k is the size of median sample and is assumed to be fixed for all allowable queries.

Security is also an issue for operating systems. Unfortunately, the solutions for operating systems are not sufficient to solve database security problems. Most operating system protection mechanisms are "access control mechanisms" [4-13]; that is, they enforce rules about who can perform what operations or access what information. In operating system protection mechanisms, different users have different access rights to the same object, they allow some users to read part or all of the contents of a file and others to alter it in perhaps limited ways. In statistical databases, all users essentially are performing read access. An access control mechanism that only distinguishes between read and alter accesses is not useful.

Another contrast between databases and operating systems concerns queries that involve many data elements. In the operating system, a complex operation can be broken into a set of accesses to individual objects and each access permission is determined independently of the others. In a database, a decision must be made whether the entire query should be permitted in the first place. This decision depends not only on the relationship of data elements being interrogated but also on the query history, the information that has already been divulged to the user.

Newer access control mechanisms take into account the flow of information out of one object and into another as part of the effect of an access. These access control mechanisms incorporate remembering the source from which information encoded in an object was derived [10, 38, 39]. Yet even such sophisticated mechanisms make no interpretation of the contents of the database and have no notion of the history of information already given out. Thus the operating system approach is not sufficient for the database problem [3].

Since operating system security methods are not robust enough for databases, researchers are seeking alternate methods for securing databases. Some of these methods are: controls on the size of query sets, controls on the overlap of query sets, distorting the data or the query responses, random sampling, and selection methods. All these methods are discussed in detail in chapter 2.

Even when modern inference control mechanisms are implemented, researchers observed that compromising a statistical database is fairly easy. The major result of a series of papers is that "compromise is straightforward and cheap" [14], to quote Denning's and Schwartz's conclusion [15].

The demand for statistical database security is higher because relatively more computerized information is maintained in databases than in the recent past. On the other hand, a good solution for this problem is yet to be achieved.

The conclusion of a careful study of past and current research in this area, by the author, is that the database security problems for statistical databases cannot be solved by implying any one security method alone. Therefore a new method of inference control for statistical databases is presented. The new method combines three existing methods. These methods are "restricting the size of the query set", "restricting the overlap of query set", and "distorting the query response". The purpose of using the first two techniques is to make it difficult for the questioner to isolate a single record. If the isolation of a single record occurs in response to a series of queries, then the database system transfers the control to a third layer of protection which generates a very carefully calculated (incorrect but statistically acceptable) response. In this way the information in the isolated record is protected by giving an incorrect answer to the query. The details of the new method are given in chapter 3. Chapter 4 contains the evaluation of the new method and comparison of performance between existing systems and the new method with the help of examples. Chapter 5 contains the conclusion and the suggestions for future work.

CHAPTER II

REVIEW OF RELATED WORK

Database compromise occurs when a questioner deduces, from the responses of one or more queries, confidential information of which he was previously unaware [2]. Researchers have studied methods of controlling compromise but have found for each method either that it succumbs to simple attacks, or that it is impractical to use. Recently the problem of protecting information has been investigated extensively. The survey by Denning and Denning [16] discusses four kinds of safeguards: access controls, flow controls, data encryption, and inference controls. Another survey by Denning [17] deals only with inference controls.

Most of the attacks are based on isolating a single data element at the intersection of several query sets. The confidential value is obtained by solving a system of equations employing the response of these queries. The defense against these attacks are of five kinds: control on the size of query sets, control on the overlap of the query sets, distorting the data or the query responses, sampling from the database, and selection methods. These controls are reviewed briefly in this chapter.

Control on the Size of Query Set

The minimum query size control aims to defend against attacks employing very large or very small query sets. e.g., with a formula, C, that identifies a single record [2, 18]. Let k denote a parameter giving the lower bound on allowable query set size. A query q(C) is not answered unless $k<=nc<=N-k$. where, N is the number of confidential records in the database, C is a characteristic formula which, informally, is any logical formula over the values using the operators AND(.), OR(+), and NOT($\overline{\ }$), and nc is the number of records in the database whose values match with the characteristic formula C. Unfortunately, this control is often subverted easily even for k near N/2 [21], by a simple snooping tool called the "tracker" [15, 19, 20]. The basic idea of the tracker is to pad small query sets with enough extra records to make them answerable, then eliminate the effect of the extra records. Suppose that a questioner, who knows from external sources that an individual I is characterized by the logical formula C, is able to express C in the form $C=(A.B)$ such that queries for the formulas A and $(A.\overline{B})$ are both answerable, then formula $T=(A.\overline{B})$ is the tracker (of I) because it helps the questioner to "track down" additional characteristics of I. The method of compromise is summarized on the next page [15].

Individual Tracker Compromise.

Let C=(A.B) be a formula identifying individual I, and suppose $T=(A.\overline{B})$ is I's tracker. With three answerable queries, calculate:

$$\mathrm{COUNT}(C) = \mathrm{COUNT}(A) - \mathrm{COUNT}\ (T) \qquad (1)$$

$$\mathrm{COUNT}(C.a) = \mathrm{COUNT}(T + A.a) - \mathrm{COUNT}(T) \qquad (2)$$

If COUNT(C.a) = 0, I does not have characteristic a.
If COUNT(C.a) = COUNT(C), I has characteristic a.
If COUNT(C) = 1, arbitrary statistics about I can be computed from

$$q(C) = q(A) - Q(T) \qquad (3)$$

and

$$q(C.a) = q(T + A.a) - q(T) \qquad (4)$$

When COUNT(C) > 1, it may happen that no compromise is possible. The following example illustrates the individual tracker compromise for database of Table I, with k=2.

EXAMPLE 1. The query set size restriction implies that a query q(C) is answerable only if 2<=COUNT(C)<=10. A questioner believes that

C = "F.CS.Prof"

characterizes Dolly, but the restriction k=2 prevents him asking the queries

COUNT (F.CS.Prof)	=	1
COUNT (F.CS.prof.$20KSal)	=	1

to determine Dolly's salary. However, the questioner can make a tracker $T=A.\overline{B}$, where A = "F" and B = "CS.Prof". To verify that Dolly is the only individual characterized by C, the questioner applies eq. (1):

$$\begin{aligned}\mathrm{COUNT(F.CS.Prof)} &= \mathrm{COUNT(F)} - \mathrm{COUNT(F.}\overline{\mathrm{CS.Prof}}\mathrm{)}\\ &= 5 - 4\\ &= 1.\end{aligned}$$

To discover Dolly's salary, the questioner would have to search using repeated applications of eq. (2). If he guessed \$25K, eq. (2) would yield

$$\begin{aligned}\mathrm{COUNT(F.CS.Prof.\$25KSal)} &= \mathrm{COUNT(F.}\overline{\mathrm{CS.Prof}} + \mathrm{F.\$25KSal)}\\ &\quad - \mathrm{COUNT(F.}\overline{\mathrm{CS.Prof}}\mathrm{)}\\ &= 4 - 4\\ &= 0,\end{aligned}$$

revealing that Dolly's salary cannot be \$25K. As soon as the questioner guesses \$20K, eq. (2) yields

$$\begin{aligned}\mathrm{COUNT(F.CS.Prof.\$20KSal)} &= \mathrm{COUNT(F.}\overline{\mathrm{CS.Prof}} + \mathrm{F.\$20KSal)}\\ &\quad - \mathrm{COUNT(F.}\overline{\mathrm{CS.Prof}}\mathrm{)}\\ &= 5 - 4\\ &= 1,\end{aligned}$$

revealing that Dolly's salary is \$20K.

It might seem that the effort to compromise the entire database is very high because the questioner would have to know identifying characteristics of each individual in order to construct a tracker for that individual. However, if a questioner can find any formula whose query set contains at least 2k but not more than N-2k records [16], he can use the formula as a "general tracker" to determine the answer to any unanswerable query of the database [15]. The method of compromise is given below.

General Tracker Compromise

The value of any unanswerable query q(C) can be computed as follows using any general tracker T. First calculate

$$Q = q(T) + q(\overline{T}). \qquad (5)$$

If COUNT(C)<k, the queries on the right-hand side of eq. (6) are answerable:

$$q(C) = q(C+T) + q(C+\overline{T}) - Q. \qquad (6)$$

Otherwise COUNT(C)>N-k and the queries on the right-hand side of equation (7) are answerable:

$$q(C) = 2Q - q(\overline{C}+T) - q(\overline{C}+\overline{T}). \qquad (7)$$

Because at least one of the eqs. (6) or (7) is calculable, q(C) can be evaluated with at most 4 queries beyond the 2 required to find Q. The following example illustrates the general tracker compromise for the database of Table I, with k=2.

EXAMPLE 2. The questioner, who knows that Dolly is a female CS professor, seeks to discover her salary. To be answerable, a query set size must fall in the range (2, 10), but a general tracker's query set size must fall in the subrange [4, 8]. The formula T = "M" qualifies as a general tracker since COUNT(M) = 7. The questioner applies eq. (5) for counting and summing queries to discover the database size (N) and total of all salaries (S):

$$N = COUNT(M) + COUNT(\overline{M})$$

$$= 7 + 5$$

$$= 12.$$

$$\begin{aligned} S &= \mathrm{SUM}(M;\mathrm{Sal}) + \mathrm{SUM}(\overline{M};\mathrm{Sal}) \\ &= \$101K + \$99K \\ &= \$200K. \end{aligned}$$

The questioner verifies that Dolly is the only female CS professor by applying eq. (6) with counting queries:

$$\begin{aligned} \mathrm{COUNT}(F.CS.Prof) &= \mathrm{COUNT}(F.CS.Prof + M) \\ &\quad + \mathrm{COUNT}(F.CS.Prof + \overline{M}) - N \\ &= 8 + 5 - 12 \\ &= 1. \end{aligned}$$

He then calculates her salary by applying eq. (6) with summing queries:

$$\begin{aligned} \mathrm{SUM}(F.CS.Prof;Sal) &= \mathrm{SUM}(F.CS.Prof + M;Sal) \\ &\quad + \mathrm{SUM}(F.CS.Prof + \overline{M};Sal) - S \\ &= \$121K + \$99K - \$200K \\ &= \$20K. \end{aligned}$$

The above two examples show that tracker compromise is clearly a powerful technique to compromise databases.

Control on the Overlap of the Query Set

The minimum overlap control inhibits the responses from queries that have more than a predetermined number of records in common with each prior query [3]. No efficient implementation of this control is known [21]. Before responding, the query program could have to compare the current query group against every previous one. This control may be subverted by queries that overlap by small amounts (e.g., by solving a system of equations) [3, 22, 23, 24, 25, 40].

An effective method of preventing a clever intruder from isolating a record by overlapping queries is "partitioning the database" [26, 36]. Records are stored in groups, each containing at least some predetermined number of records. Queries may apply to any set of groups, but never to subsets of records within any group. Therefore it is impossible to isolate a record. A variant is called "microaggregation". Individuals are grouped to create many synthetic "average individuals". Statistics are computed for these synthetic individuals rather than the real ones [37].

Partitioning has two severe practical limitations in dynamic databases. First, the free flow of useful statistical information can be inhibited severely either by excessively large groups or by ill-considered groupings. Second, forming and reforming groups as records are inserted, updated, and deleted from the database can lead to costly bookkeeping [21].

Distorting the Data or Query Response

The minimum query size controls and minimum overlap controls give exact answers when they respond. "Rounding" aims to prevent inference by perturbing the responses. Under "direct rounding", the answer to a query is rounded up or down by a small amount before it is released [27, 28]. Rounding by adding a zero-mean random value (noise) is insecure since the correct answer can be deduced by averaging a sufficient number of responses to the same query [21]. Rounding by adding a pseudorandom value that depends on the data is preferable, because then a given query always returns the same response. The method can sometimes be subverted with trackers by adding dummy records to the database [29] or simply comparing the response to several queries in order to narrow the range of values containing the confidential value [30].

A method of indirect rounding is called "error inoculation". This control aims to prevent inference by perturbing or replacing the values stored in records [31]. Like direct rounding, this control attempts to trade accuracy in the statistics for security. One method is to modify the data when the record is created (losing the original data). The problem with this approach is that correctness of the raw data may be essential for other uses; e.g., storage and retrieval of patient's medical records. A better approach stores a "perturbation factor" in the record along with the original data and applies this factor when the data are used in a query [31].

A variation of error inoculation which may not disturb the accuracy of the statistics is "multidimensional transformation" or "data swapping". The values of fields of records are exchanged so that the record for any particular individual is likely to be incorrect, but maintains the same frequency count statistics as the original data. Data swapping reduces the risk of compromise since there is no way of knowing with which individual a disclosed value is actually associated. The problem with this approach is that no efficient method exists either for finding groups of records whose values can be swapped, or for determining whether a valid swap even exists is known. Since exact data swapping is not feasible practically, Reiss [33] suggested a feedback algorithm to find an approximate data swap on a categorical data set. Approximate data swapping is still in an experimental stage, and its computational efficiency has yet to be determined. Furthermore, approximate data swapping is not feasible for noncategorical data such as salary figures [32].

Conway and Strip [34] suggested value distortion, in which the value of a restricted field would be modified by some random quantity before retrieving the value to the query. That is,

$$Vd = Va + Vr$$

where, Vd, is the distorted value. Va, the actual value, and, Vr, a random deviate with a given distribution, d. The + sign in the formula implies that this strategy is applicable only to fields with arithmetic values. One can imagine random distortion of character-valued fields (perhaps by random displacement in the collating sequence), but it is hard to imagine the resulting Vd being of any use whatever.

The distribution, d, is chosen to have an expected value of zero so that Vd is an unbiased estimater of the true value, Va. It is not always obvious what would constitute an appropriate distribution. If Va in the statistical database is symmetric, then the random deviate distribution probably should be symmetric. But if the Va is highly skewed, which is a common occurrence [32], then the choice of the distribution is much more difficult.

Random Samples

All the controls listed above are subverted by a single basic principle of compromise. Because the questioner can control the composition of each query set, he can isolate a single record or value by intersecting query sets. Rounding and error inoculation perturb the responses, but the "noise" can often be removed by averaging responses for carefully selected query sets.

The U.S. Census Bureau has used a technique that responds to queries that involve only a random subfile of the database, rather than the complete database. In "random sampling" the user can apply responses to a set of records no longer selected by him. This prevent inference by depriving him of the ability to isolate a known record. The 1960 U.S. Census, for example, was distributed on tape as a random sample of one record in 1000 [27]. The best snooper would have at best a 1/1000 chance of associating a given sample record with the right individual. This technique is applicable to large databases only. Because a small random sample will be useless, other methods are needed to prevent compromise of small databases.

Selection Methods

Some researchers have considered key specified queries which selects some element from the query set; e.g., the median, the largest, or the smallest data values [3, 23]. In [23], the selection of response to any query is any value within the query set, which need not be the right one. For example, the database in response to a query:

"median salary (Brown, Black, White, Red)"

simply may decide to return White's salary, whether or not White's salary is actually the median salary. The problem with this type of system is that the answer is chosen within the query set and repeated queries with overlap can deduce the right information in about $4k^2$ queries, where k is the query size [23].

Recently Schierman, Jonge, and Riet have presented a method in which the database refuses to give answer to a query, if the right answer reveals the secret [35]. Even this complex system is not secure, because, the user with the responses of the past queries and a little logical cleverness, can deduce the right answer. It is because of the fact that the database refuses to respond only when a secret is likely to be revealed.

The net result of the above survey is that it is very difficult to protect a statistical database. Every method is unsafe against at least one type of attack. In the next chapter a new method is presented. The new method is a combination of three of the above mentioned techniques with some variations.

TABLE I
HYPOTHETICAL DATABASE OF EMPLOYEES OF A UNIVERSITY

NO	NAME	SEX	DEPT	POST	SAL($K)	DONAT($)
1	ABLE	M	CS	PROF	20	50
2	BOB	M	ENG	PROF	15	150
3	CARY	F	EE	PROF	25	150
4	DOLLY	F	CS	PROF	20	65
5	EDDY	M	STAT	PROF	15	0
6	FLYNN	F	STAT	ADM	23	150
7	GATE	M	MATH	PROF	12	20
8	HOME	M	CS	PROF	16	450
9	IAN	F	CS	STU	6	60
10	JIM	M	STAT	ADM	18	15
11	KATE	F	MATH	PROF	25	100
12	LAMB	M	CS	STU	5	0

TABLE II
EXAMPLES OF QUERIES FOR DATABASE OF TABLE I

FORMAL QUERY	ANSWER	INFORMAL STATE
COUNT (F . CS)	**2**	**# of females in CS Dept.**
COUNT (M . Adm . (EE + Stat))	**1**	**# of male adm in EE or Stat Depts.**
SUM (F + Math; Sal)	**$126K**	**Total of salaries among either males or Math Dept Personnel.**
SUM ($25K Sal; Donat)	**$250**	**Total of donations by persons earning $25K.**
SUM (Donat > $100; Sal)	**$79K**	**Total of salaries of persons donating more than $100.**

CHAPTER III

THE NEW METHOD OF INFERENCE CONTROL

The major cause for the leakage of information from statistical databases is that the user can isolate a single record containing information about a particular individual, by means of a series of queries. The new method for inference control is devised such that it will prevent the leakage of information even when the user isolates a single record successfully. Three layers of protection are incorporated into the new system to achieve the desired results. (i) When a questioner has complete control over the query set, and when the responses are undistorted, then compromise is easy [16]. The principle of this compromise is simple. The questioner finds a formula, C, whose query set count is 1. He can then discover whether the individual thus isolated has any other characteristics, X, by asking "How many individuals satisfy C(AND)X ?" The response "1" indicates that X is characteristics of the individual and "0" indicates not. The basic idea of this protection layer is given below:

> Do not respond to queries for which there are fewer than k, or, more than N-K records in the query set. Where N is the total number of records in the database, and k>0.

The positive integer K in this control is a design parameter specifying the smallest allowable size of a query set. If the query language permits complementation, then a maximum size N-K of the query set must also be enforced, for otherwise the questioner could pose his queries relative to the complement ($\overline{C}$) of the desired characteristics (C). The value of k for this system is kept as low as 2. The above method alone is not sufficient protection from tracker based inquiries; therefore, the second layer of protection is incorporated.

(ii) Tracker based compromises employ groups of records having high overlaps [16]. To protect against trackers a minimum overlap control is taken into consideration. The basic idea of this control is:

> Do not respond to a query that has more than a predetermined number of records in common with any prior query.

The only difficulty in this type of control is that, before responding, the query program would be required to compare the latest query group against every previous one. This can be achieved by maintaining a log of all the queries asked in a session. If there is a chance that the log file can grow indefinitely, then some upper bound can be placed on the number of queries a questioner can ask in one session. No compromise is possible if overlapping of fields are not allowed, but at the same time the task of getting statistics out of database becomes very difficult. Overlapping of two (2) fields is both sufficiently convenient for the users and highly effective against tracker based queries.

Even if the above two techniques are applied to the query program, the isolation of a single record is not preventable completely; however, it takes many more queries and a great deal of user's time to perform the task of isolation than it would without these techniques. A third level of protection is needed and to accommodate this technique in the system the basic idea of (i) is modified slightly:

> Respond to all queries for which there are more than or equal to K, or, fewer than or equal to N-K records in query set. For queries which have fewer than K, or, more than N-K records in the query set, do not give direct response but activate the third layer of protection.

if the query set is one (1) or N-1, the system still responds to the query, but the response given is not from the query set, rather, it is generated by the system randomly. The answers are generated such that every time the same query is asked, the system gives the same response. This can be achieved by taking into account the data stored in the record. This is true even when the data stored in the database in non-numeric. The random number generator can take the non-numeric fields, change them into numeric fields by taking the ASCII or EBCDIC equivalents. The response does not depend upon the full record under isolation, but on the fields which the query references. This way different queries to the same record get different responses, and hence protect the information. Until the data stored in the record is changed, the questioner receives the same response whenever he asks the same query, in the same session or in different sessions, and hence prevent any inconsistency in calculating statistics.

This added protection of giving incorrect answers if a single record is isolated along with the other two techniques can prevent disclosure of information very successfully. In this system the value of K is kept as low as 2, compared to the limit of N/2 of some of the other systems. As the actual data in the database is not distorted in any way, it is possible to incorporate privileged access for any person who has the authority to access the complete database as it is; e.g., a doctor getting a patient's medical history, or the head of a firm getting the personal dossier of his employee due for promotion, etc. The basic logical flow of the system is shown in fig 1. Each of the three system when applied alone could not prevent the deduction of private information, but, when applied together may very well protect the database as shown in chapter 4.

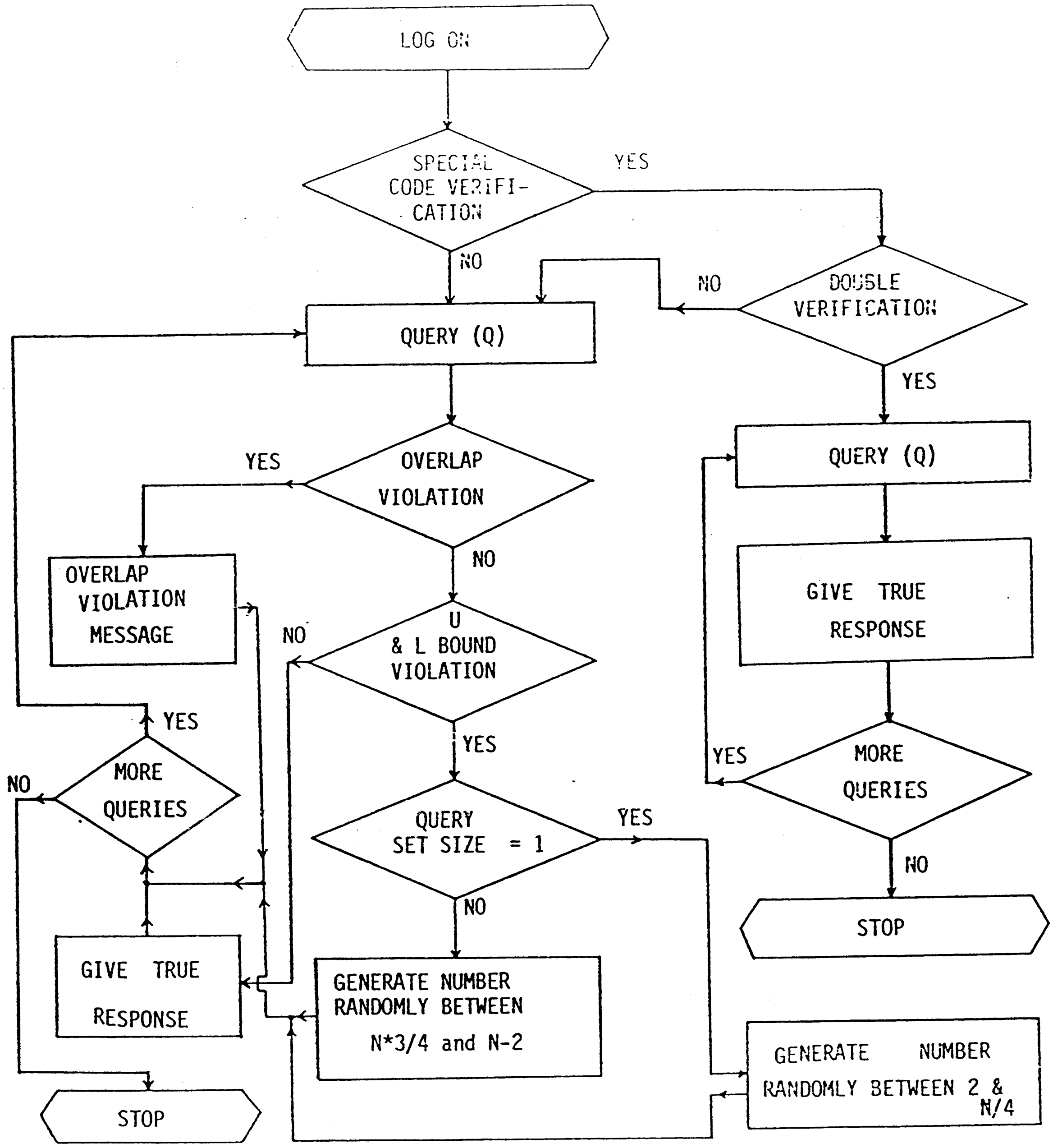

FIGURE 1. Logical Flow Diagram of the New Method

CHAPTER IV

EVALUATION OF THE NEW METHOD

In this chapter, every possible query that may compromise a database is posed on the database of Table I under different systems, including the new method. The response of queries for these systems are noted to compare the superiority of performance among systems. Usually COUNT, SUM, AVERAGE, and MEDIAN types of queries are used to get the statistics from the statistical databases, but since COUNT and SUM queries are used in tracker based compromise, only these two are considered here. The examples of queries for the database of Table I are expressed formally and informally in Table II.

Once again the rules which the new method follows to give response to queries are summerized below.

1) Check whether the current query overlaps more than two fields with any of the prior queries:
   a) if 'NO', go to 2.
   b) if 'YES', the query response will be 'O-V' (overlap violation).
2) Check the size of query set:
   a) if between 2 & N-2, then give true response.
   b) if either 1 or N-1, then go to 3.

3) Check the size of the query set:

   a) if 1, then generate random number between 2 and N/4.

   b) if N-1, then generate random number between N*3/4 and N-2.

The above three rules when applied in the query responses will be pointed out by their number.

Consider the database of table I and suppose that no restriction is posed on queries. A questioner who tries to find Dolly's salary poses the following queries:

```
COUNT(F.CS.Prof)          = 1
COUNT(F.CS.Prof.$20KSal)  = 1
```

in just two queries he finds out the information and compromises the database.

Now suppose the database of table I incorporates the control on the size of the query set and k=2, i.e., 2<=COUNT(C)<=10. The questioner posing the same two queries gets the following response:

```
COUNT(F.CS.Prof)          = ###
COUNT(F.CS.Prof.$20KSal)  = ###
```

As the query set of both the queries is 1, the database refuses to give any response. This refusal of database reveals to the questioner that the query sets have violated either the lower or the upper limit of query set. Since the normal queries are not helpful in this situation, the questioner applies the individual tracker based queries to find Dolly's salary. Since C = (F.CS.Prof), he applies eq. 1 & 2 (page8) and forms A=(F), B=(CS.Prof), and T=(F.$\overline{\text{CS.Prof}}$). He asks the following queries:

$$\begin{aligned}\text{COUNT(F.CS.Prof)} &= \text{COUNT(F)} - \text{COUNT(F.}\overline{\text{CS.Prof}}\text{)}\\ &= 5 - 4\\ &= 1.\end{aligned}$$

He now knows that C uniquely identifies Dolly, and poses two more queries for her salary:

$$\begin{aligned}\text{COUNT(F.CS.Prof.\$20KSal)} &= \text{COUNT(F.}\overline{\text{CS.Prof}} + \text{F.\$20KSal)}\\ &\quad - \text{COUNT(F.}\overline{\text{CS.Prof}}\text{)}\\ &= 5 - 4\\ &= 1,\end{aligned}$$

which reveal that Dolly's salary is $20K.

Now, incorporating the new method of protection into the database of table I, and asking the same query COUNT(F.CS.Prof). The query set of this query is 1, for which the Rule(2b) of the new system is applied. The new system generates the response randomly, say,

COUNT(F.CS.Prof) = 3 (Rule # 3a)

and the response to the query

COUNT(F.CS.Prof.$20KSal) = 'O-V' (Rule # 1b)

because there are more than two fields common in both the queries. The questioner asks further queries and gets the following responses.

COUNT(F.CS) = 2 (Rule # 2a)

COUNT(F.Prof) = 3 (Rule # 2a)

COUNT(F.$20KSal) = 2 (Rule # 3a)

COUNT(CS.Prof) = 3 (Rule # 2a)

COUNT(CS.$20KSal) = 2 (Rule # 2a)

COUNT(Prof.$20KSal) = 2 (Rule # 2a)

The above queries are the only possible queries the questioner can ask without violating the overlap constraint and with these responses he is not sure what Dolly's salary is, because the value of the formula "F.CS.Prof.$20KSal" can not be determined by the above queries.

Now assuming that the questioner after asking the query COUNT(F.CS.Prof), and getting the response 3, applies the individual tracker based queries to find Dolly's salary:

$$\text{COUNT(F.CS.Prof.\$20KSal)} = \text{COUNT(F.}\overline{\text{CS.Prof}} + \text{F.\$20KSal)} - \text{COUNT(F.}\overline{\text{CS.Prof}}\text{)}$$

Since query $\text{COUNT(F.}\overline{\text{CS.Prof}}\text{)} = 4$ (Rule # 2a) has more than 2 fields common with the other query the other query gets the response,

$$\text{COUNT(F.}\overline{\text{CS.Prof}} + \text{F.\$20KSal)} = \text{'O-V'} \qquad \text{(Rule \# 1a)}$$

Since one of the queries is not answerable, the questioner remains unsuccessful.

It is fair to assume that the questioner applies the tracker queries from the very begining. He first finds the count of female professors in the CS department; i.e.,

$$\begin{aligned} \text{COUNT(F.CS.Prof)} &= \text{COUNT(F)} - \text{COUNT(F.}\overline{\text{CS.Prof}}\text{)} \\ &= 5 - 4 \\ &= 1. \end{aligned}$$

He determines that Dolly is the only female professor in CS department and asks queries for her salary,

$$\mathrm{COUNT(F.CS.Prof.\$20KSal)} = \mathrm{COUNT(F.\overline{CS.Prof} + F.\$20KSal)} - \mathrm{COUNT(F.\overline{CS.Prof})}$$

Since these are the same queries asked above, and at least one of them is not answerable under the new system (rule # la), the questioner remains unsuccessful in deducing Dolly's salary.

In the last, general tracker based queries are posed on the database to find Dolly's salary. In the database of table I, the formula T = 'M' qualifies as a general tracker since COUNT(M) = 7. The questioner applies eq. 5 (page10) for counting and summing queries to discover the database size (N) and total of all salaries (S):

$$\begin{aligned} N &= \mathrm{COUNT(M)} + \mathrm{COUNT(\overline{M})} \\ &= 7 + 5 \\ &= 12. \end{aligned}$$

$$\begin{aligned} S &= \mathrm{SUM(M;\ Sal)} + \mathrm{SUM(\overline{M};\ Sal)} \\ &= \$101K + \$99K \\ &= \$200K. \end{aligned}$$

The questioner tries to verify that Dolly is the only female CS professor by applying eq. (6) with counting queries:

$$\mathrm{COUNT(F.CS.Prof)} = \mathrm{COUNT(F.CS.Prof + M)} + \mathrm{COUNT(F.CS.Prof + \overline{M})} - N$$

but overlap violation once again blocks his way. He then tries to calculate her salary by applying eq. (6) with summing queries:

$$\text{SUM(F.CS.Prof;Sal)} = \text{SUM(F.CS.Prof + M;Sal)} + \text{SUM(F.CS.Prof} + \overline{\text{M}}\text{;Sal)} - \text{S}$$

and once again the database refuses to give response to at least one of the queries due to the overlap violation and hence prevents the disclosure of private information.

The new method is evaluated assuming a single questioner is trying to compromise the database. This system, however, may not be useful if more than one person try to compromise the database for the same information at different times and compute the result by comparing the responses to their queries. This type of (gang) compromise is possible for the tracker based queries only, since they are always in the answerable limits of the databases. Since the tracker based queries have a set patterns, one way to solve this problem may be to put a check on every query asked to the system for its being tracker query. If a pattern is matched, then the possible combinations of the other tracker queries related to the first match are generated by the system internally and, then onward, every query asked by any user within a predetermined time (say one month) is checked for a match.

For individual tracker of example 1, the following queries are asked:

$$\text{COUNT (F)} \qquad \text{....(i)}$$

$$\text{COUNT (F.}\overline{\text{CS.Prof}}\text{)} \qquad \text{....(ii)}$$

Since query (ii) is matched with the tracker query pattern of (A.$\overline{\text{B}}$), the database generates all the possible matching queries:

COUNT (F.$\overline{\text{CS.Prof}}$ + F.Sal) ....(iii)

COUNT (F.$\overline{\text{CS.Prof}}$ + F.Donat) ....(iv)

For general tracker, the following queries are asked:

COUNT (M) ....(v)

COUNT ($\overline{\text{M}}$) ....(vi)

SUM (M; Sal) ....(vii)

SUM ($\overline{\text{M}}$; Sal) ....(viii)

COUNT (F.CS.Prof + M) ....(ix)

Query (ix) matches the tracker query pattern (C+T), therefore the following eqs. are generated by the database:

COUNT (F.CS.Prof + $\overline{\text{M}}$) ....(x)

COUNT (F.$\overline{\text{CS.Prof}}$ + M) ....(xi)

and for the query SUM(F.CS.Prof + M;Sal) the following eqs. are generated:

SUM (F.CS.Prof + $\overline{\text{M}}$; Sal) ....(xii)

SUM (F.$\overline{\text{CS.Prof}}$ + M; Sal) ....(xiii)

Eqs. (iii), (iv), (x), (xi), (xii), and (xiii) are kept in a separate file and all other queries, asked by any user, will be compared to this file for a possible match. There is no response to the matched queries; hence this protects the database.

Now the new method is tested for another database which is used in the paper by Denning D. E. [17]. The database is given in Table III. The questioner knows that "L" is a male director and a board member. To find the number of overdrafts taken by L, he applies the following queries:

COUNT(M.DIR.MEM)	= 3	(Rule # 3a)
COUNT(M.DIR.MEM.OD=50)	= 'O-V'	(Rule # 1b)

because there are more than two fields common in both the queries. The questioner asks further queries and gets the following responses.

COUNT(M.DIR)	= 2	(Rule # 3a)
COUNT(M.MEM)	= 3	(Rule # 2a)
COUNT(M.OD=50)	= 2	(Rule # 3a)
COUNT(DIR.MEM)	= 3	(Rule # 3a)
COUNT(DIR.OD=50)	= 2	(Rule # 3a)
COUNT(MEM.OD=50)	= 3	(Rule # 3a)

Above are the queries the questioner can ask without violating the overlap constraint and with these responses the value of the formula "M.DIR.MEM.OD=50" can not be determined.

Since the normal queries are useless, the questioner applies the Individual Tracker queries. He forms the tracker as follows:

C = (M.DIR.MEM)

A = (M)

B = (DIR.MEM)

$T = (M.\overline{DIR.MEM})$

He applies eq. (1) from page 8, and gets the following responses:

$$\begin{aligned}\mathrm{COUNT(M.DIR.MEM)} &= \mathrm{COUNT(M)} - \mathrm{COUNT(M.\overline{DIR.MEM})} \\ &= 8 - 7 \\ &= 1.\end{aligned}$$

He finds out that L is the only male director who is also a board member. The questioner now applies eq. (2) from page 8 to find out the number of overdrafts taken by L:

$$\begin{aligned}\mathrm{COUNT(M.DIR.MEM.OD{=}50)} = {} & \mathrm{COUNT(M.\overline{DIR.MEM} + M.OD{=}50)} \\ & - \mathrm{COUNT(M.\overline{DIR.MEM})}\end{aligned}$$

since the queries in eq. (2) have three fields in commom which violates the overlap constraint (rule # 1a) of the new method, the new method responds with 'O-V' for one of these queries.

In the last, general tracker based queries are posed on the database to find L's number of over drafts. In the database of Table III, the formula T = 'M' qualifies as a general tracker since COUNT(M) = 8. The questioner applies eq. (5) from page 10, for counting and summing queries to discover the total number of over drafts (S):

$$\begin{aligned}\mathrm{S} &= \mathrm{SUM(M;\ OD)} + \mathrm{SUM(\overline{M};\ OD)} \\ &= 108 + 3 \\ &= 111.\end{aligned}$$

The questioner tries to apply eq. (6) with summing queries to find out L's over drafts:

$$\begin{aligned}\mathrm{SUM(M.DIR.MEM;\ OD)} = {} & \mathrm{SUM(M.DIR.MEM + M;\ OD)} \\ & + \mathrm{SUM(M.DIR.MEM + \overline{M};\ OD)} - \mathrm{S}\end{aligned}$$

and once again the database refuses to give response to at least one of the queries in eq. (6), due to the overlap violation. This way the disclosure of private information is prevented.

By incorporating the new method of inference control, a statistical database may be made more secure than:
control on the size of query set; because this system can be subverted by tracker queries, and the new method protects the information against these types of queries;
control on the overlap of query set; because this control can be subverted by solving equations of the queries, and the new method is safe against this type of attack. Unlike Partitioning, the new method does not have any problem with the free flow of statistical information and does not need costly bookkeeping for update, insertion and deletion of records in the database;
distorting the data or query responses; because the data stored in the database under the new method is original, unlike direct rounding, and can be used for other purposes. All the statistics returned by the new method are true values, unlike indirect rounding, except for the case where the query set is 1;
and above all, the new method protects the database against the gang compromise, and , that may make it a highly secure method for inference control, compared to any single method.

TABLE III
REFERENCED DATABASE FROM DENNING'S PAPER [17]

NO	NAME	SEX	PROFESSION	MEM	OD	AMOUNT($)
1	A	M	LAWYER	NO	1	10
2	B	M	JOURNALIST	NO	0	0
3	C	M	PRESIDENT	NO	0	0
4	D	M	DOCTOR	NO	2	100
5	E	M	LAWYER	YES	30	50,000
6	F	F	LAWYER	NO	0	0
7	G	F	SENATOR	NO	3	50
8	H	M	LAWYER	YES	25	10,000
9	I	F	DOCTOR	NO	0	0
10	J	M	SENATOR	NO	0	0
11	K	F	JOURNALIST	NO	0	0
12	L	M	DIRECTOR	YES	50	100,000

CHAPTER V

SUMMARY AND CONCLUSION

A new method of inference control for statistical database is presented and evaluated against all types of known attacks to deduce private information from statistical databases. The system is especially evaluated for the most powerful tools of compromise, the trackers, and shown to be highly secure, although not completely secure against gang attacks, over a period of time. The new method is a combination of three already existing methods, namely, control on the size of query set, control on the overlap of query set, and, distorting the response of the query only when the query isolates a single record. The new method always responses to the user queries of any type and hence prevents the guessing of responses by the questioner which leads to compromise.

In the case of gang compromise, the efficiency and security of the system depends on the number of records in the database, and, on the number of fields in the records. The larger the database, the more time system will spend in matching the queries, but the efficiency can be enhanced by maintaining the matching file at very regular intervals.

The new method implemented with the matching file system is feasible for any database of any size. For extremely larger databases having millions of records, the matching file system, however, will need extra care, otherwise the efficiency of the system will be effected adversly.

The new method is only evaluated for COUNT and SUM queries. Possible future work may be to extend the method to handle the AVERAGE, MEDIAN and other queries. Also, the implementation of the new method on an available statistical database system will be an excellent way to find the practicality of the system.

BIBLIOGRAPHY

[1] Haq, M.I. "Insuring individuals' privacy from statistical database users." Proc. AFIPS 1975 NCC, vol. 44, AFIP press, Montvale, N.J., 941-946.

[2] Hoffman, L.J. "Getting a personal dossier from a statistical data bank." Datamation 16, 5 (May 1970), 74-75.

[3] Dobkin, D., Jones, A.K., and Lipton, R.J. "Secure databases: Protection against user influence." ACM Trans. Database Syst. vol. 4, # 1, (March 1979), 79-106.

[4] David, L.S., and Ehud, G. "A unifying approach to the design of a secure database operating system." IEEE Trans. Softw. Eng. SE-10, # 3, May 1984, 310-319.

[5] Reynolds, D., and Henry, G. "The IBM system/38." Datamation, Aug 1977, 141-143.

[6] Date, C. "An introduction to data base systems." Reading, MA: Addison-Wesley, 1977.

[7] Griffiths, P. P., and Wade, B. W. "An authorization mechanism for a relational database system." ACM Trans. Data Base Syst., vol. 1, # 3, (Sep 1976), 242-255.

[8] Linden, T. "Operating system structures to support secure and reliable software." ACM Comput. Surveys, vol. 8, (Dec 1976), 409-445.

[9] Popek, G.J. "Protection structures." Computer, June 1974, 22-33.

[10] Fenton, J.S. "Memoryless subsystems." Computer J., vol. 17, # 2, (May 1974).

[11] Conway, R., Maxwell, W., and Morgen, H. "On the implementation of security measures in information systems." CACM, vol. 15, (Apl 1972), 211-220.

[12] Purdy, G. "A high security log-in procedure." CACM, vol. 17, (Aug 1974), 442-445.

[13] Evans, A., Kantrowitz, W., and Weiss, E. "A user authentication scheme not requiring secrecy in the computer." CACM, vol. 17, (Aug 1974), 437-442.

[14] Tarub, J.F., Yemini, y., and Wozniakowski, H. "The statistical security of statistical database." ACM Trans. Database Syst. 9, 4 (Dec 1984), 672-679.

[15] Denning, D.E., Denning, P.J., and Schwartz, M.D. "The tracker: A threat to statistical database security." ACM Trans. Database Syst. 4, 1 (March 1979), 76-96.

[16] Denning, D.E., and Denning, P.J., "Data security." ACM Computing Survey, 11, 3 (Sep 1979), 227-249.

[17] Denning, D.E., "Are statistical databases secure?", Proc. 1978 Nat. Compt. Conf., vol. 47, AFIP press, Arlington, Va, 525-230.

[18] Chin, F.Y. "Security in statistical databases for queries with small counts." ACM Trans. Database Syst. 3, 1 (March 1978), 92-104.

[19] Denning, D.E., and Schlorer, J. "A fast procedure for finding a tracker in a statistical database." ACM Trans. Database Syst. 5, 1 (March 1980), 88-102.

[20] Schlorer, J. "Disclosure from statistical databases: Quantitative aspects of tracker." ACM Trans. Database Syst. 5, # 4, (1980), 467-492.

[21] Denning, D.E. "Secure statistical databases with random sample queries." ACM Trans. Database Syst. 5, 3, (Sept 1980), 291-315.

[22] Davida, G.I., et al. "Data base security." IEEE Trans. Softw. Eng. SE-4, 6 (Nov 1978), 531-533.

[23] DeMillo, R.A., Dobkin, D., Lipton, R.J. "Even data bases that lie can be compromised." IEEE Trans. softw. Eng. SE-4, 1 (Jan 1978), 73-75.

[24] Kam, J.B., and Ullman, J.D. "A model of statistical databases and their security." ACM Trans. Database Syst. 2, 1, (Mar 1977), 1-10.

[25] Schwartz, M.D., Denning, P.J., and Denning, D.E. "Linear queries in statistical databases." ACM Trans. Database Syst. 4, 2, (Jun 1979), 156-167.

[26] Yu, C.T., and Chin, F.Y. "A study on the protection of statistical data bases." ACM SIGMOD Conf. Management of Data, Toronto, Canada, (Aug 1977), 169-181.

[27] Hansen, M.H. "Insuring confidentiality of individual records in data storage and retrieval for statistical purposes." Proc. AFIPS 1971 FJCC, vol. 39, AFIP press, Montvale, N.J., 579-585.

[28] Reed, I.S. "Information theory and privacy in data banks." Proc. AFIPS 1973, vol. 42, AFIP press, Arlington, Va., 581-587.

[29] Karpinski, R.H. "Reply to Hoffman and Shaw." Datamation 16, 10 (Oct 1970), 11.

[30] Haq, M.I. "On safeguarding statistical disclosure by giving approximate answers to queries." Int. Computing Symp., 1977, 491-495.

[31] Beck, L.L. "A security mechanism for statistical databases." ACM Trans. Database Syst. 5, 3, (Sep 1980), 316-338.

[32] Liew, C.K., Choi, J.U., and Liew, C.J. "A data distortion by probability distribution." ACM Trans. Database Syst. vol. 10, # 3, (Sep 1985), 395-411.

[33] Reiss, S.P. "Practical data swapping: The first steps." Proc. of Symp. on Security and Privacy, (Apl 1980), IEEE, New York, 38-45.

[34] Conway, R., and Strip, D. "Selective partial access to a database." Proc. of ACM Annl. Conf. (Houston, Tx., Oct 20-22, 1976), ACM, New York, 85-89.

[35] George, L.S., Weibren, D.J., and Reind, P.V. "Answering queries without revealing secrets." ACM Trans. Database Syst. 8, 1, (Mar 1983), 41-59.

[36] Chin, F.Y., and Ozsoyogulu, G. "Security in partitioned dynamic statistical databases." Proc. COMPSAC '79 (Piscataway, N.J., 1979), IEEE, New York, 594-600.

[37] Feige, E. L., and Watts, H.W. "Protection of privacy through microaggregation." Databases, Computers, and the Social Sciences, R. L. Bisco, Ed., Wiley-Interscience, New York, 1970.

[38] Denning, D. E. "A lattice model of secure information flow." Comm. ACM 19, 5 (May 1976), 236-243.

[39] Jones, A.K., and Lipton, R. J. "The enforcement of security policies for computation." Proc. 5th Symp. Oper. Syst. Principles, Oper. Syst. Rev. (ACM) 9, 5 (1975), 197-206.

[40] Schwartz, M. D., Denning, D. E., and Denning, P. J. "Linear queries in statistical databases." ACM Trans. Database Syst., vol 4, # 2, (June 1979), 156-167.

VITA 2

SHAHID R. MALIK

Candidate for the Degree of

Master of Science

Thesis: A NEW METHOD OF INFERENCE CONTROL FOR STATISTICAL DATABASES

Major Field: Computing and Information Science

Biographical:

Personal Data: Born in Mirpurkhas, Sind, Pakistan, November 12, 1958, the son of Rashid Malik and Khalida Rashid Malik.

Education: Graduated from Pakistan High School, Manamma, Bahrain, Arabian Gulf, in July 1974; passed Inter Science Pre-Engineering from Government College, Mirpurkhas, Sind, Pakistan, in July 1976; received Bachelor of science in Electronics Engineering from N. E. D. University, Karachi, Pakistan, in 1982; completed requirements for the Master of Science degree in Oklahoma State University in December, 1988.

Professional Experience : Assistant Engineer in Space and Upper Atmosphere Research Commission of Pakistan (SUPARCO), Karachi, November, 1982, to February, 1983; Research and Development Officer in Inspectorate of Electronics and Instruments (IE & I), Chaklala, Rawalpindi, Pakistan, February, 1983, to October, 1985; Assistant Engineer in SUPARCO, Karachi, March, 1986, to September, 1986.