DATABASE SECURITY AND DETECTION INFERENCES: AN OVERVIEW OF WESLEY CHU'S DATABASE PROJECTS

Nicca Lewis

Department of Computer Science and Information Technology

University of the District of Columbia

Abstract:

Throughout companies, government departments, and doctors' offices, database systems are used. This particular system stores and retrieves sensitive information such as social security numbers, financial statements, and highly classified data. Organizations with sensitive data in their hands need to be secured using different security techniques and policies. In order to secure the data on a computer, they need to implement techniques like access control, auditing, authentication, encryption, etc; however, malicious users are still breaking into companies' data. Needless to say, companies are not implementing poor security techniques but hackers are just getting smarter and smarter. This is where IT employees need to find new techniques or enhance previous ones.

The proposed research paper will examine Wesley Chu's database projects. His projects deal with protecting databases by using inference detection. This system protects against sensitive data being accessed by malicious users using the detection system.

Keywords: Database, Security, and Detection Inferences.

1. Introduction

In recent years, companies and organizations have become centered around a new technology in which information can be easily stored and retrieved in the blink of an eye. It helps with companies keep track of their customers and financial statement exclusively on the computer system. This new technology is referred to as database systems. Because in the past decades, companies have expanded their database systems, there is a need to improve the systems' security. There is a lot of volatility in the database system that can be improved through the development of newer, more modern software that could ensure system security.

Database security is used to deny access to unauthorized users by process or techniques. Unauthorized users are defined as malicious users, user's misuses, or mistakes made by authorized individuals or process. Business and non-profit organizations are the focal point of database systems. They are collectively storing sensitive information on their computer systems, such information as medical records, social security numbers, credit

cards, bank accounts, and other relevant personal data. Database security protects the breaching of information by malicious users.

According to Wikipedia, "Traditionally databases have been protected from external connection by firewall or routers on the network perimeter with the database have been protected from environment existing on the internal network opposed being located within a demilitarized zone" [3]. There are other security protection techniques and protocols, which have been implemented to secure the system, such as access control, auditing, authentication, encryption, and integrity control. Securing the system with the aforementioned techniques can be difficult, mainly because attacks are being made in accessing the system. Malicious users are using different methods and technical tools to gain access to the database system. For example, in 2002 hackers breached into personal database and financial information 265,000 workers that included Governor Gray Davis. The databases held their names, social security, and payroll information. Some attacks on systems, including the hackers that breached the workers database, are "accessing a series of innocuous information and employing inference techniques to derive sensitive data using that information to access the system" (A. Philip). Malicious users are using the weakness of the inference channel to deploy of database system.

This is the single most important motive why companies and organizations fear utilizing database systems; knowing that hackers are finding it easier to breach their systems, especially with this underlying issue with the inference channel. Companies are using their Information Technology (IT) manager and other specialists to find new techniques and processes to improve the system. It is a never-ending process implementing new procedures since companies and non-profit organizations are increasing their central databases by placing their goods and services on-line.

This research survey paper examines a new technique that will probably be placed in the database system as a security control. The new technique dealing with the inference problem in the system will be discussed in this paper with the assistance of the research project done by Computer Science Professor Wesley W. Chu of University of California. His project is based on the use of a detection violation system that protects sensitive information within a database. The development of the detection system is centered on the directory site. It provides an "image" so as allow scalable and systematic sound inference to vary depending on the channels. According to his research, "construction of a semantic inference model (SIM) which represented all possible channels from any attribute in the system to the set of pre-assigned sensitive attributes" [2]. The attribute contains related attributes that include data dependency, database schema, and semantic related. The SIM model helps the inference detection system by allowing them to keep track of a users' query history. The project focuses on multiple users, rather than individual users because researchers want to have a real life scenario from which to acquire accurate date during an experiment.

The remainder of the survey research paper is sectioned as follows. It will discuss previous projects and papers that dealt with this dilemma in regards to the database system. However, in the midst of reviewing the previous projects on the issue, there will be a discussion on inference channel as a whole. Background information on the problem will be elaborated on which will include descriptions of various multi-level

database inferences, how they are derived from the system, and the overall generalities of the inference channel.

2. Background

Inference channeling in Database systems happens when sensitive information can be revealed by a single user that has low-level classified information. Basically meaning unauthorized user(s) can somehow input a series of non-sensitive information or what others consider low-level information can breach into highly classified database systems. For example, a database administrator from a military transportation system created a table containing cargo information. Each row contains information on shipments of all in and out going flights, such as the air carrier identification numbers and listings of shipment contents.

The identification numbers of flights in the table cross referenced with another table that contains the origin, destination, flight time, and similar data which appears as the follows:

Flight ID	Cargo Hold	Contents	Classification
1254	A	Boots	Unclassified
1254	B	Guns	Unclassified
1254	C	Atomic Bomb	Top Secret
1254	D	Butter	Unclassified

A low rank officer decides to input some new records into the system regarding an imaginary shipment trying to see if there any top secret cargos shipping out. When the low ranking officer inputs his information and the system reject it, he now has information that was considered classified or top secret shipping information. In a way this issue is like jigsaw puzzle because it can piece together information at one security level to determine a fact that should be protected at a higher security level. The inference channel is somewhat a problem with the access control techniques because IT managers and other specialist want some type of flexible access in the database; however, it causes volatility within a database system. IT managers are trying to correct this issue.

There's an understanding that the inference channel occurs when there are two or more path among the attribute in the database, which has different classification levels. Among using the path, it can be eliminated by the upgrade some other attribute with in the path. This is one way inference channel can occur in the system. There are numbers of inference channels discovered in the course of database security research, according to

Sushil Jajodia and Catherine Meadows's essay [8]. One way inference can occur in discretionary access control system where users are explicitly granted access right to access database. It not obvious task to grants a users the amount access rights the need. In some cases, users are given more rights to database system then they should be given. This type of inference needs to be monitor by looking into the history of user's queries which can be simple to monitor. In other ways this problem can occurs in queries that are based on sensitive data. Assume a user creates a sequence of queries against the database. The queries would help the user to derive an inference, which has a higher classification than the retrieved data.

Here's an example that was given by Jajodia and Meadows to understand, "suppose the database system is classified as rational level" [8]. Within the system, Database manager creates two relations, one that's unclassified called EP (short for Employment); with contain attributes in the system such as EMPLOYEE-Name and PROJECT-Type. There's another relation called PT, standing for Project, with attribute name PROJECT-NAME and PROJECT- TYPE. These attribute are keys to relation of the database system (the existence of the relation scheme PT is classified). This is shown in a SQL query which an un-cleared user has input" [6]. SELECT EP.EMPLOYMEE- NAME From EP, PT WHERE EP.PROJET-NAME=PT.PROJECT-NAME by looking at the query, it shows to be an inference channel even though the unclassified data is being return to the users. This piece of unclassified data still output secret information. Another example to the similar issue; however in this query the user can see the higher classified data. SELECT EP. EMPOYEE-NAME from EP PT WHERE EP.PROJECT-NAME = PT.PROJECT-NAME AND PT.PROJECT-TYPE= SDI.

In a secure multilevel database management system, there's a chance inference channel issues will arise in the system. This is due to the combined data being retrieved from the database with the metadata uses of storages and management. It can lead to a problem in a relational model of multi-level database systems, where a user creates a tuple that have key integrity. Key integrity means every tuple in a relation imbedded a unique key; however, that's not a concern if all the keys in the relation are equally secure. The user, which has a low security, wants to enter a tuple in a relation that's data is classified. The tuple has a similar key that exist already which the database system management need to either delete the existing tuple or tell the user there's a existing tuple with the same key. It can cause a problem with the highly classified data shown to low user, and also the low level user can somehow delete an existing highly classified user in the system, which is unacceptable.

In analyzing different inference in the database system, researchers have expanded their an inference project on eliminating the issue with varies techniques and persuaders. In recent years, researchers have found a method that prevents inference within databases from recurring in the system. By locating inference channel and preventing any occurrence of these types of problems happening in the system. Some have used semantic data modeling to detect the inference channel. It looks into database design and redesigns it to make sure that this type of inference does not occur in the system. The other technique evaluates the database system, which read, update or both by using database transaction to determine if inference has occurred. The technique will either

disable the query or reclassify the query in higher level, only if it discovers an illegal inference.

IT and other researchers extensively studied the inference channel problem over the years, in a manner of getting to the root of the issue. Their studies have brought on various different research projects; either by classifying the different types of inference that occur in the system, or developing new techniques. Many of the earlier works extend their bases of identifying inference to future projects. For example, the Denning research project discusses the view of schema to remove any unauthorized data from the answers to select-only, select-project, and select-project-join queries.

This particular project did not deal with inference problem in the presence of database constraints. Recent researchers took the overview of previous inference work to develop prevention techniques. Within the prevention stage of the research, researchers came up with a new technique that protects the database system from inference channel. Future projects dealt with the notion of using a detection system that will prevent the issue from occurring in the system. There are two different developments of the detection techniques. Each of the two detection techniques is briefly discuss in the previous section.

One of detection methods was led by Thomas H. Hinke and Delugauch, developing detection methods that automate analysis the inference in the database system [4]. The aim was to use these techniques as a detection system in various databases and corresponding semantics. In developing these techniques they used a conceptual graph-based method to detect illegal inference in the system. According to their project, this is mentioned in The Inference Problem: A survey inputs "database entities and activities, relationships between them, domain knowledge, and data sensitivity are represented in the graph. The graph is manipulated by inference rules to derive new inferences. The illegal inferences are detected if there exists a path from unclassified information to classified."

Additionally to their previous work, Hinke also develop several techniques based on the conceptual graph analysis and implemented prototype of the inference detection systems: Merlin, AERIE and Wizard. For example, Wizard takes an input of database schema and other things that will generate inference channel. Here's graph that's displayed in Hinke and Delugauch one of their research projects:
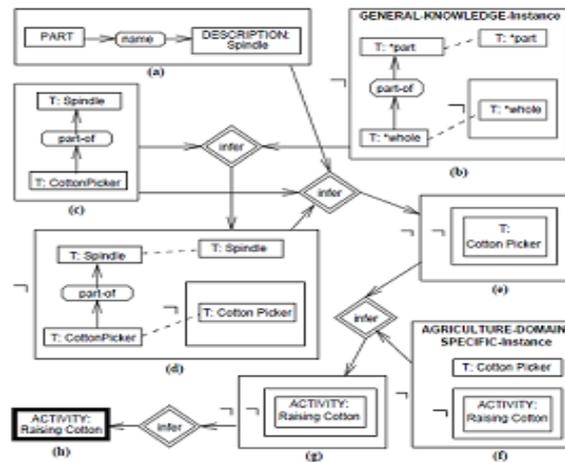
Figure 9: Materialization Of An Activity-Entity Relationship.

In the second detection technique, T.D Garvey developed a tool for database designer to detect and remove specific types of inference in a multilevel database system [7]. This prevents the inference issue to exist in the system. Both detection techniques use schema- level knowledge and do not infer knowledge at the data level.

Each of these projects is used during the database design time instead of run time. However Raymond W. Yip argues against both research that uses schema. Basically pointing out in his research alongside his partner, Karl N Levitt that schema-level inference detection has inefficiency in the technique; within the research he identifies six types of inference rules from the data level, which serve as deterministic inference channel [11]. To insure multilevel database system is secure, a development of an inference controller prototype was developed, which handle inferences channel during query process. Strategies such as rule based inference were applied to this prototype to protect the security.

After the development of the inference prototype, Tyrone S. Toland proposed to develop a system that generate update to the user history file to ensure no query is rejected based on the outdate information. Since this proposal didn't focus on reduce the time spent on examining the history log in computing inference, he decided to use a prior knowledge of data dependency to reduce the search space of a relation and thus reducing the process time for inference.

In previous work, researchers force mostly on deterministic inference channel such as functional dependency. It's an important start in preventing the issue from occurring in this type of database environment; however this will not completely eliminate the issue. Therefore, it is important to examine the outer data environment to truly be able to restrict inference issue. Non-deterministic correlation in data is begin ignore in the inference research community. This can particular occur in a salary data where the

amount of currency range cannot determine the rank. Furthermore, many of the semantic relationships, as well as data types, cannot be specified deterministic environment. To address this concern, Chu created a new approach in database inference, called probabilistic inference. It will treat the detection problem. His work also touches on the detection framework for multiple inferences by collaborative users.

So far, this section presented the base of previous research projects on inference, and the important impact to expand the study to prevent database inference from occurring in the system. This gives way for the next section to discuss, in theory, inference detection in regards to Chu and et al's work in Collective database inference. In the following section, we will explore system tools to analysis development of the detection system, followed by an in depth discussion on the base of his work and findings.

3. Result of Projects by Chu's Group

3.1 The framework

The research analysis the inference channel detection by using a module tool that combines the data schema, dependency, and semantic knowledge. The modules, semantic inference model (SIM); link all possible inference channel relation among the attributes of the data sources, such as related attributes and entities. Additionally to the SIM module there's a semantic inference graph that constructs the query time of violation in the system containing inference occurs in the database system.
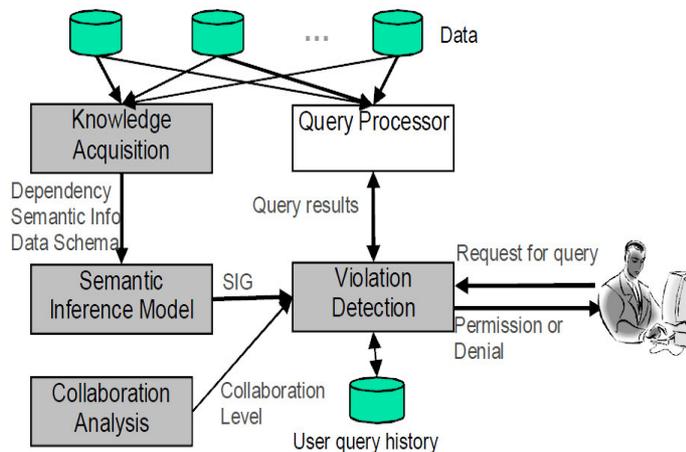

Fig. 1. The framework for an Inference Detection System

In return, allows the inference detection combines the new query request with the request log. It checks the current logs exceeds fixed pre meter of information leakages. According to the work, if there's collaboration according to collaboration analysis the violation detection module will decide whether to module will decide whether to answer a current query based on the acquired knowledge among the malicious group members and their collaboration level to the current user.

The data schema, dependency and other knowledge was extracted into a knowledge acquisition module. This is on the base of users may pose queries and acquire knowledge from different sources. There needs to be an understanding of the three knowledge entail. For data dependency it consist of two things, one is casual relationship and non-deterministic correlation between attribute values. The non-deterministic nature allows the dependence between two attributes to be represented by conditional probabilities. This gives more general representation then deterministic relationships. Non-deterministic data dependences consists of two types.

The first type is called dependency within entity, it lets two attributes such as A and B be stored in an entity. Let us say that B depends on attribute A, then each entity E will be placed with the value of attribute B. However, this is only possible when the value has dependency on attribute A. In order to learn the parameter of dependency within entities from relational data, an sequential scan of the relational table that stores entity E count the occurrence of A, B , and co occurrence of A and B , using the conditional probabilities formula $p_{i|j}=Pr(B=b_i|A=a_j)$. As for the other data dependency, Dependency-between-related-entities allows attribute A to resign in entity E1 and attribute C in E2, in which E1 and E2 are related by R a relation that can be derived from the database schema. The parameters will all the first joining the two entity tables bases on relation R, it will scan and count the frequency of occurrence of the attribute pair in the joining table. "If two entities have an m-to-n relationship then the associative entity table can be used to join the relation entity tables to derive dependency-between-related entities," which is stated in the report [2].

The database schema defines the tables, the fields in each table, and the relationships between fields and tables in a relational database. The database designer uses data dictionary, in which is generally stored to define data schema. The database owners use two keys, primary key and foreign key to specify the entities. The keys represent a relationship between two entities. For example, "If entity E1 has primary key pk, entity E has Foreign key fk, and e1.pk=e2.fk, then dependency- between-related-entities from attribute A (in e1) to attribute C (in e2) can be derived" [2].

Outside information such as domain knowledge can cause inference channel problems in the system. This kind of knowledge with relation among attributes and entities can render a problem in the secure of database, allowing hackers to be able to breach the system. The research does not define semantic knowledge among attributes in the database, which vary with context. The semantic knowledge will be extracted from a large set of semantic queries posed by the users. For example, in the following illustration, "in the WHERE clause of the following query, clauses 3 and 4 are semantic condition that specify the semantic relation 'can land' between entity Runway and entity Aircrafts. Based on this query, this allows the researcher to extract semantic knowledge 'can land' and integrate it into the SIM" [2].

- **Query: Find airports that _can land_ a C-5 cargo plane.**

```
SELECT AP.APORT_NM
FROM AIRCRAFTS AC, AIRPORTS AP, RUNWAYS R
WHERE AC.AC_TYPE_NM = 'C-5' and                          #1
AP.APORT_NM = R.APORT_NM and                             #2
AC.WT_MIN_AVG_LAND_DIST_FT <= R.RUNWAY_LENGHT_FT and     #3
AC.WT_MIN_RUNWAY_WIDTH_FT <= R.RUNWAY_WIDTH_FT;          #4
```

3.2 Research findings

In developing the semantic inference graph, there's a need to reduce
complexity in the evaluating user inference for sensitive attribute. The researcher
develops a map that links SIM to a Bayesian network. The meaning of Bayesian network
is a probabilistic graphical model that represent a set of random variable and their
conditional independencies. By using this module, it gave the research the opportunity to
measure time required for inference evaluation, which was almost constant.   For
example, the calculation of 40 nodes and 28 edges gave an elapse time after a single users
poses a random query of 16 minutes. This was taking by using a Dell desktop running
Window XP with 3.20GHZ CPU and 2GB of ram.

In using the violation detection in preventing inference occurrence in multiple users,
researcher discovers the needed to estimate the effects of the multiple users. This evolves
several experiment studies that found out there are three types' of collaboration levels.
The three are the following: authoritativeness, honesty, and fidelity. Authoritativeness
represent how accurate is the information. In other words, if a provider is knowledge and
has a high class within the task, then the provider can give more accuracy of information.
The honesty show up only if the provider gives truthful information and rely the
knowledge to a recipient. As for fidelity, it measures a percentage of information send
over to recipient due to limitation in communication mode. All three of components give
the research the ability to derive the collaboration level, which tests the detection system.

Within the collective levels, several experiments that were look into which asked several
students to take part in the collaboration test bed. The students had to register on a web
base system and input the their ages, gender, major, year in school, courses taken, GPA
skill, interests, teamwork and ability, social activities, friends in the class, and what
dealing with school. The information gave clues about the information authoritativeness
and certain aspects of the fidelity of the test subject. In the experiment students were split
into five teams. The first team consists of PhD students who had knowledge in database,
which should have good authoritativeness. The second team was made up of good friends
of the PhD students, which provided good communication fidelity. The other three was
made up of random students.

In the first team revealed communication fidelity plays a positive role in determining
collaboration effectiveness. This was done by using the SIG structure base in the
database and allowing the each team to enter fixed number of queries to infer the security
attribute, in the first test. The system computed what teams inputted in the system. The
system then denied any queries that exceed the threshold. In the below figure team 2 had
the highest inference probability, mainly because they held meeting to discuss strategies

in posing queries. The second test, the same experiment was conducting in the second test. In second test resulted in a crease of authoritativeness in each team. With the same fixed queries, there was an additional knowledge of CPTs and threshold of the security attribute. There were six queries denied out of four of the five team due to the excess of the threshold. In the second experiment, they address both authoritativeness and fidelity all together. The experiment was similar in the previous experiment but they used other graduated to conduct the test. The results confirmed that both collaborative levels have positives affect, and there are two types
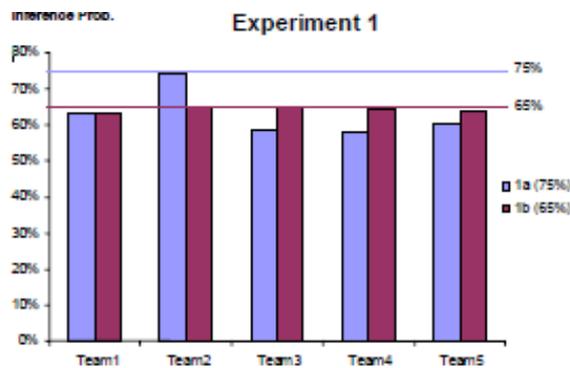
of components in the level. In honesty collaboration experiment, they used proxy to find the result. In this experiment, team members could not exchange information. The proxy can also alter the query answers to control the honesty. The result from the first experiment gave way that the proxy did not send the exact answer to the original query; however it was sent to the parent node of the original query in the inference channel. There was an average of loss of information by answering parent node. As for the three groups it was an average of 0.6107.
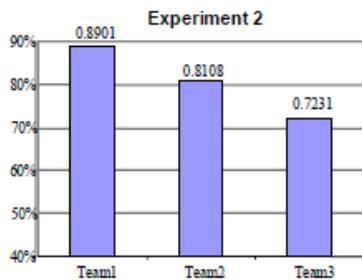

Fig. 14. Inference result for Experiment 2. The inference threshold of the security node was set at 90%.
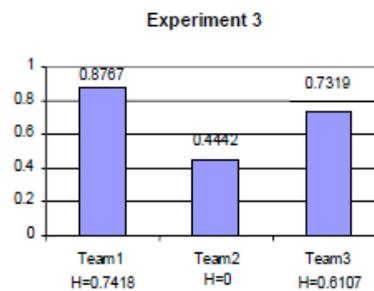

Fig.15. Inference result for Experiment 3. The inference threshold of the security node was set at 90%.

The development of the inference detection allowed them to examine the users past queries logs, and calculate the probability of sensitive information from answering the posed query. It would be denied only if any query can infer sensitive information which exceeds the pre specified threshold. This also extends in the detection inference in the collaborative users environment. Therefore, any two multiple users would be stored as one knowledge; this will detect their inference toward sensitive. Basically the enters of the user queries, allowed the system not only to check if the query request can be refer to sensitive data above the threshold with query answer, it also check other team members query answer to disable them from inferring sensitive data, by using an N- collaborative case approach. The below algorithm is used to either deny or enable a query request from any multiple users. Additionally using the algorithm, they first need to sort out all N (number) members by the members' inference probability, sensitive attributes and the member highest inference probability as well. Suppose the highest member wanted to get into the current query answer and cannot get into the sensitive attribute threshold.

```
1.   Assume: current query request Q, malicious team M, sensitive data S, threshold of S is T;
2.   List(M) = sort team members M in descending order of inference probability to S;
3.   While(List(M) is not empty) {
4.       m = first member in List(M) with highest inference probability;
5.       max_col = the maximum collaborative level from any member in List(M) to the query requester;
6.       real_col = m's collaborative level to query requester;
7.           If (m integrate answer to Q with max_col can get inference probability < T)
8.           Then {answer query Q; goto end;}
9.           Else
10.              If (m integrate answer to Q with real_col can get inference probability >= T)
11.              Then {deny query Q; goto end;}
12.              Else { List(M) = List(M) – m;}
13. }
```

The researcher could stop all findings of the rest of the team and answer the query question, mainly because no other member of the team has a highest inference. If the highest member can get into database then, Chu and his team can continue on with the next member with the highest inference until a decision can be decided. This is the bases on how they analysis the collaborative user inference detection system.

4. Conclusion

By developing and analyzing the inference channel to give way to improve the statue of database system, just like the previous research projects of Professor Wesley Chu, it gives the database administration a chance to create similar inference detection system and extend their knowledge of the area for better results in the future. In years to come, hopefully inference detection techniques will be created in every database system to tightly secure the system, where any type of unwanted users cannot get access to data so easily.

References

1.   Chapple, Mike. "Database Security Issues: Inference." www. About.com/database security"

2. Y. Chen and W.W. Chu "Protection of Database Security via Collaborative Inference Detection" In IEEE Transactions on Knowledge and Data Engineering (TKDE), Special Issue on Intelligence and Security Informatics, Vol 20, No 8, August 2008

3. "Database security." *Wikipedia, The Free Encyclopedia*. 11 Aug 2009, 22:27 UTC. 11   Aug   2009 <http://en.wikipedia.org/w/index.php?title=Database_security&oldid=307449603>.

4. Delugauch, Harry S., and Thomas H. Hinke. "Wizard: A Database Inference Analysis and Detection   System." In *IEEE Trans. Knowledge and Data Engineering*, vol. 8, no. 1, 1996.

5. Delugauch, Harry S., and Thomas H. Hinke. "Using Conceptual Graphs To Represent Database Inference," Inference Security Analysis, 2007

6. Farkas, Csilla, and Sushil Jajodia. "The Inference Problem: A Survey." ACM SIGKDD Explorations Newsletter 4.2 (2002): 6-11.

7. Garvey, T. D., and T. F. Lunt. "Multilevel Security for Knowledge Based Systems." Proceedings of the Sixth Annual Computer Security Applications Conference, December 3-7, 1990, Tucson, Arizona. Los Alamitos, CA: IEEE Computer Society, 1990. 148-59.

8. Meadows, C., and S. Jajodia. "Integrity versus Security in Multi-level Secure Databases." Database Security: Status and Prospects. Amsterdam: North-Holland, 1988. 89-101.

9. Qureshi, Anique A., and Joel G. Siegel. *The International Handbook of Computer Security*. New York: Amacom/American Management Association, 2000.

10. Petrocelli, Tom. Data Protection and *Information Lifecycle Management*. Alexandria, VA: Prentice Hall, 2005.

11. Yip, Raymond W., and Karl N. Levitt. "The Design and Implementation of a Data Level Database Inference Detection System." Database Security XII: Status and Prospects : IFIP TC11 WG11.3 Twelfth International Working Conference on Database Security, July 15-17, 1998, Chalkidiki, Greece. Boston: Kluwer Academic, 1999. 253-66.

12. Yovits, Marshall C. Advances in Computers. New York: Academic P, 1994.

(Nicca Lewis was a senior student when she completed the paper. Cera Chen at JSPC was the technical editor for this paper.)