D. Project Description

1 Introduction

The insider threat. *Insider attacks* are an extremely serious and pervasive security problem. For obvious reasons, an insider attack has the highest damaging potential for an organization. According to a 2010 U.S. Secret Service/CERT/Microsoft E-Crime report [36], 67% of the respondents reported that insider attacks are the most costly and damaging type of attacks. About 51% of respondents to the survey who experienced a cyber security event experienced an insider attack. The severity of the problem has prompted the organization of many recent meetings and workshops on the topic such as NSA Workshop [5], ACM CCS Workshop [4], MIST [2, 6, 7], SEI Training Workshop [8], Dartmouth and Columbia Workshop [1], and the RAND Study [20]. Insider attacks such as *masquerading* and *privilege abuse* are well-known threats in the financial, corporate and national defense domains [24, 161]. Attackers may abuse *legitimate* privileges to conduct *snooping* or *data-harvesting* [142] with malicious intent (e.g., espionage). The Bradley Manning incident in late 2009 exemplifies the severity of insider attacks. Manning, then an intelligence analyst for the U.S. Army based in Iraq, (allegedly) leaked a myriad of classified documents containing national defense information to the website Wikileaks, including the now infamous "Collateral Murder" video, the "Afgan war documents," the "Iraq war documents," various military logs and diaries, and more than 250,000 diplomatic cables by 260 embassies and consulates in 180 countries [95].

Detecting, mitigating, or recovering from insider attacks is extremely challenging. First, coming up with a useful *definition* of an insider attack is already difficult. Generic definitions which capture most insider attacks, such as the RAND's report definition [20], are too vague and too wide in scope to be useful in designing and implementing a practical attack detection and mitigation tool. Second, separating *normal* behaviors from (inadvertently or not) *malicious* behaviors by an insider is difficult, even for human inspectors. Third, even if one can precisely characterize normal user behavior, say, by statistically profiling access patterns, these profiles keep changing over time. The detection engine has to be able to smoothly adapt to these normal behavioral shifts. Fourth, the attacker – being an insider – will probably have knowledge of the profiling strategies and thus s/he can adapt to the profiling accordingly. Last but not least, it is very difficult to obtain meaningful and realistic data to validate insider detection and mitigation approaches. In domains where the insider threat is a dangerous problem, such as the banking or military sectors, data are sensitive and confidential and thus hard to come by.

This proposal plans to face the above challenges with the following key components. (1) Narrow down the enormous scope of the problem to a manageable yet still meaningful domain; (2) Propose a novel methodology for user profiling and statistical model building; (3) Partner with a large financial institution; (4) Integrate a unique combination of expertise from the PIs, ranging from machine learning, databases, to security, and algorithms.

Our focus. All of the challenges above are manifested in hundreds of examples reported in the studies by CERT and the U.S. Secret Service [161]. The results indicate that a multi-pronged approach involving the understanding of psychological, organizational, technical aspects of the problem, along with their temporal correlations is probably most effective. While this recommended multi-front framework certainly makes sense, it is too vague to be directly translatable into a technical solution. Furthermore, modeling and validating psychological and organizational aspects is beyond the scope of a computer system research project. Consequently, this proposal focuses only on specific technical aspects of the problem. Even in the purely technical domain there is still a massive ground to cover.

In particular, we will focus on formulating and devising a solution to the insider attack problem on relational database management systems (RDBMS). We devise concrete plans to validate and evaluate our solutions by intimately collaborating with a large financial institution to build a prototype insider threat detection engine operating on their operational, consumer, and commercial RDBMS. We chose RDBMS as the technical target for three reasons: (1) we want to narrow down the enormous scope of the problem to a more manageable level so that real and measurable progress can be made, (2) RDBMS is one of the most pervasive pieces of technologies in use today, which are also highly susceptible to insider attacks, and (3) we have a novel idea for insider attack detection in RDBMS which has been demonstrated in a recent work [94].

We believe that the most effective method for dealing with the insider threat problem is to statistically profile normal users' behaviors (i.e., database access patterns) and change the "threat level" appropriately when a user deviates from his/her routines. For example, a good statistical profiler should be able to detect many types of sabotage attacks, quick data harvesting attacks, or masquerading attacks because the statistical computing footprints of those actions should be significantly different from the day-to-day activities. The idea of profiling users for insider threat detection in particular and security-related anomaly detection in general is certainly neither new (see, e.g., [50, 82, 145]) nor

surprising (attacks *are* anomalous). The novelty lies in the answers to two critical and intertwined questions: (1) *how do we model and construct user profiles?* and (2) *how to use the profiles to detect insider attacks?* Our proposed research is centered around these two questions, built upon an idea called the *query-semantics approach*.

Main idea: data access pattern reveals query semantics. Perhaps the most natural user "profile" is (the syntax of) the set of SQL queries a user issues daily to a database, or more concretely some feature vectors representing past SQL queries. (See, e.g., [71].) This approach has the advantage that the query processing phase of the attack detection system is computationally light. However, in a recent paper [94] we have demonstrated that this syntax-centric view is ineffective for database insider threat detection. On the one hand, queries may differ widely in syntax yet produce the same result, causing the syntax-based detection engine to generate false positives. On the other hand, syntactically similar queries may produce vastly different results, leading to high false negative rate.

Our main idea and also our conviction is that the best way to distinguish normal vs. abnormal (or good vs. malicious) access patterns is to focus on *what* the user is trying to access – the result of the query itself – rather than just *how* s/he expresses it, i.e., the SQL queries. Two syntactically similar queries should be considered different if they result in different tuple sets. In other words, this approach values the *semantics* of the queries more than their *syntax*. When a malicious insider tries to acquire *new* knowledge about data points and their relationships, the data points the insider accesses are necessarily different from the *old* accessed points. To illustrate the idea of looking at the data access region to capture query semantics, consider two queries

```
Q1)   SELECT p.product_name, p.product_id FROM PRODUCT p WHERE p.cost = 100 AND p.weight > 80;
Q2)   SELECT p.product_name, p.product_id FROM PRODUCT p WHERE p.cost > 100 AND p.weight = 80;
```

It should be obvious that for non-trivial databases the results of Q1 and Q2 are very different. Yet, a syntax-based approach such as the one used in [71] will extract *exactly the same* syntactical feature vector from Q1 and Q2 thus consider them identical. Syntax analysis, even if very detailed (taking into account the topology of the query parse tree, distinguishing between '=' and '$>$' in the above examples) is difficult given the expressiveness of the SQL language. Syntax analysis involves determining *query equivalence*, that is difficult to perform correctly. In fact, query containment and equivalence is NP-complete for conjunctive queries and *undecidable* for queries involving negation [23]. Our approach is to peek into the data access regions of queries; therefore we bypass the computational intractability of syntax analysis.

Our recent work [94] showed that the query-semantics approach is very promising. In particular, we devised a scalable feature extraction method where each user's query is represented by a vector in a space whose dimension is only dependent on the database schema, independent of the number of returned tuples (the size of the data region accessed by the query). These feature vectors are then used to train specifically chosen machine learning algorithms, including k-mean clustering and support vector machine (SVM). Preliminary results indicated that our learners can be accurate in detecting several types of attacks, including data harvesting and role masquerading. Our tests were based on an actual graduate admissions database which we have access to. Obviously, a typical graduate admissions database and the damages resulting from "insider attacks" on such databases are not significant enough to demonstrate that our solution is effective in a "real-world" setting. For this reason, we have sought out to an industrial partner with important data, large databases, and a *much* greater loss potential.

Industrial Partner. Our supporting partner for this proposal is *M&T bank*, a very large financial institution headquartered in Buffalo. (See supporting letter from Mr. John Walp, Administrative Vice President and Chief Information Security Officer of M&T Bank.) M&T bank offers a wide range of financial services, including consumer and commercial banking, securities trading, as well as investment and insurance. Many of the attacks/incidents happened at the bank that the M&T security team was allowed to reveal to us were insider attacks. Here are three examples. The first incident which incurred a loss of $2.5 million involves a home equity line of credit (*HELOC*) wire transfer fraud. Several customers with large HELOC balances were setup in Web Banking to allow the funds from their lines of credit to be transferred into their regular Web Banking accounts. This was accomplished by the fraudsters social engineering the telephone banking center (TBC) staff *and* by knowing the exact date the customer's account had been opened. Some of the impacted customers had been with the bank for over 20 years and the bank's analysis indicated that even the customers themselves were extremely unlikely to remember those dates. The exact opening dates of users' accounts were only viewable within one system in the bank. In the second incident, a trader based in the stock trading unit initiated thousands of transactions without customer permission in order to drive up his commissions. The loss from this incident was estimated at $650 million. In a third incident, an insider ran human resource database queries in an attempt to find out how much everyone in the Technology department was making, all the way up to the CTO. The perpetrator was able to access the system and run queries with no restrictions or alerts. The perpetrator was only caught when they revealed their findings to other staffers. All three incidents involved insiders, and could have been

detected by a query-semantics-driven approach such as ours. In the first incident, the act of viewing account opening dates is abnormal in the sense that the insider was looking into a corner of the data space that normally is not accessed. In the second incident, the volume of data movement was the indicator. The third incident is quite similar. In fact, our prior work [94] was designed specifically to catch "browsing" attacks like this one. In summary, even though the query-semantics approach has not been tested "in the wild," we strongly believe that the approach has great potential. In fact, after learning of our approach, the M&T security team fully agreed with it. Eric Ayotte, Vice President and Manager of Network Security at M&T, has agreed to serve as a consultant for this project, and his team has also committed to provide logistical support for this project if it is funded.

Outline of proposed research. While the preliminary work was encouraging, there is still much to be accomplished. (1) The false positive and false negative rates for data harvesting and masquerading attacks are still high, ranging from about 10% to 18%. Furthermore, as alluded to earlier, the test database was not economically significant nor sufficiently large to validate our approach. There remains considerable ground to cover in terms of statistical modeling of the problem to improve the failure rates and in terms of realistic evaluation and prototyping. (2) A significant challenge that [94] did not address is that of a dynamic database, with data insertions, deletions, and updates. Updates to the database will cause the user profiles to be "shifted" statistically, rendering learned models eventually obsolete. (3) In modeling the insider attacks, in [94] we were only able to detect attacking queries if a single query imposed a relatively significant change in the statistical signature as compared to normal queries. For data harvesting, e.g., this deviation needs not hold if the attacker is smart: he can "spread out" his attack over time such that individual queries are not abnormal, but the collection of queries within a certain time window is. Correlations between temporally nearby query results are a fundamental piece of information to be taken advantage of. (4) Even when a group of near-by queries can be deemed anomalous, the attacker can mix and match attacking queries with normal queries to camouflage the attack. (5) An important piece of information which was not specifically modeled in [94] was the correlation between the syntax of a SQL query, other contextual features such as source IP address, duration, time-stamps, and the data-region of the query result. In a masquerading attack, e.g., one can envision the attacker trying to mimic the syntax of a normal SQL query to obtain a different query result. By exploring the correlation between the input query and the output tuples, we should be able to detect subtle anomalies.

It should be emphasized again that data access region features are not the only features available for profiling users. For example, we will also take advantage of syntax-based features, and more behavioral indicators such as the source machine, inter-arrival time between queries, session durations, and database transactions. However, the "data as semantics" idea will be the driving force. The project is roughly divided into *four* thrusts, each of which addresses some combination of the above issues in a coherent manner. What follows is an outline of *what* to be done in each thrusts, leaving the detailed discussions of *how* we plan to accomplish the tasks to Section 3.

- **Thrust 1.** We aim to work on designing new "baseline" learning models to significantly reduce the false positive and false negative rates. The models are called "baseline" because in this thrust we will work with features that are generic to RDBMS, not tied to a specific domain such as finance; also, we will work with relatively static databases where updates are infrequent. The new models will be hierarchical and mixture models, which are much more expressive in terms of capturing non-unimodal profiles. At the same time, we will also build a prototype to evaluate and feedback into our models using several databases in active use at M&T bank with the help of their security team. The main threat models used for evaluation are data harvesting, role masquerading, and privilege escalation which we will define and develop extending the outline shown in our prior work [94].

- **Thrust 2.** Once a collection of reasonably good baseline learning models has been designed and the software prototype has established a pipeline for model validation and feedback, we will delve deeper into the more specific financial domain, where the generic baseline models may not provide adequate performance in terms of false positive and negative rates. To address this issue, we will incorporate expert domain knowledge into the models in at least two ways. First, to increase the detection rate, the M&T security team will help with red-teaming exercises, i.e. they will "attack" our prototype software. The attacks essentially apply years of expertise gathered by the security experts, allowing us to tune "knobs" in the models. Second, to reduce the false positive rates the domain experts can also teach us the operation rules at the bank explaining why some accesses with abnormal access patterns are in fact normal. We will present later specific statistical approaches for improving the "false positive fatigue" issue, one of the hardest problems in intrusion detection.

- **Thrust 3.** Modeling access patterns into data-regions of a static space is expected to be easier than modeling access patterns into a "moving" space. When the database is updated, the data points that it possesses are moved,

deleted, or inserted. In a sense there is a dynamical system that our models have to capture. For example, if the normal behavior is to move things at a slow pace (say, salary increases in small percentage), then a fast pace dynamics should raise a red flag. Modeling such dynamics in a discrete space is a challenging *and* exciting statistical problem, which should have applications elsewhere. Since a temporal dimension is considered and modeled here, it is natural to attempt to deal with stealthy attacks too.

- **Thrust 4.** Along with statistical modeling, the evaluation of the models has to be done through a prototype software system interfacing with the RDBMS. This will require us to address issues related to efficiently logging database state and user behavior, as well as providing low-latency responses to online classification requests. One solution to both problems might be to perform some or all of the feature selection, model training, and feedback looping in the database itself. Selecting an appropriate distribution of effort between these two domains is critical to ensuring the efficiency and responsiveness of any practical prototype.

Intellectual merit. First, the insider threat problem is a notoriously hard security problem to pin down, as is well-documented [20, 36, 161]. We believe that our narrowing down the problem's scope to RDBMS in itself is a useful conceptual step, which not only helps design a solution, but also paves the way for feasible independent benchmarking and validation. The ubiquity of RDBMS makes our problem domain highly relevant, not only to insider threat detection specifically but also to anomaly detection in databases as a whole. Second, coming up with learning models and algorithms for capturing users' behavior, taking into account data updates and temporally correlated query results, is itself an important problem. For example, in addition to profiling user behavior from a security standpoint, we can also use user profiles for caching and query optimization [48, 65]. Third, our problem domain provides an important and challenging ground for applications and further developments of a number of areas of active interest in both statistics and machine learning communities, including hierarchical and nonparametric modeling and structured prediction. Although our focus is detection of insider threats, the statistical learning techniques developed here will be general and will be useful for other applications as well. Compared to existing anomaly-based intrusion detection solutions, the major novelty of this proposal lies in the data-driven semantics approach and new statistical modeling ideas for the problem. Fourth, good threat models and assessment methodology for insider attack mitigation in RDBMS is a contribution to the security research community at large. All insider threat detection engines need realistic and systematic threat models for benchmarking and performance comparisons. Finally, by integrating ideas from statistics, machine learning, algorithms, databases and security, this project will introduce new issues and develop new techniques that have the potential of making a tangible and profound impact on these fields.

PIs Ngo and Upadhyaya have been working together on the insider threat problem for about 7 years. They and their students have developed a combinatorial insider threat model for organizational networks [32], analyzed its complexity [32] and also developed a GUI tool and threat analysis algorithm based on the model [57] as part of an earlier DARPA project. They collaborated on the recent preliminary RAID'2010 paper [94] which proposes the data-centric idea and illustrates its potentials for detecting several types of insider attacks. PI Nguyen has been working on the development of models and learning algorithms for detection problems, including detection of anomalies and change points in networks and other distributed systems in the last 7 years (e.g., [67, 125, 133]). He also has considerable expertise in hierarchical modeling and nonparametric statistics for structured data [121, 122, 128]. PI Kennedy has been working on online monitoring tools for the past 7 years, including real time analytics and monitoring [13,73,75], as well as with online validation of the correctness/security of critical systems [137]. He also has 4 years of experience in building systems that interface databases with models and uncertain data [74, 76, 77].

Broader impacts. Accurate insider threat detection in RDBMS will undoubtedly be widely useful in business enterprises, academia, finance, and national defense sectors. Collaborating with an industrial partner on a real problem that they care about on real data sets that are in active use has the potential to make the research impact immediate. The research will be conducted under the aegis of the UB's NSA- and DHS-certified center of excellence (CAE) in Information Assurance whose mission includes the involvement of minority and women students in research. In the last 10 years of its existence, the center, directed by one of the PIs, has mentored three undergraduate minority students and three doctoral women students among tens of other MS and PhD students. The center has been conducting cyber security awareness workshops as a means to attract students to work in cyber security and efforts will be made to fulfill minority recruitment through the center. The center has reached out to high schools and middle schools by offering these awareness workshops at their levels. The topic of this research will be a significant addition to the security awareness workshops. Research results will be reported via technical papers and disseminated through a new research webpage at the CAE, and through conferences and journals and the center's future workshops.

This multi-PI research uniquely connects databases, security and machine learning, opening up new research topics for PhD students in computer science and engineering. The results of our research will be brought into classroom first through the graduate seminar courses offered by the PIs at their respective institutions and after maturation, will be integrated into the first level graduate courses on Computer Security and Database Systems in the form of lecture modules and lab projects. In addition, the new statistical algorithms developed in this research will also be recommended for teaching in the machine learning classes in our undergraduate and graduate programs.

2 Core Ideas, Preliminary Results, and Main Research Questions

Core Ideas. Consider an RDBMS where there are multiple users who regularly access the database in accordance with their daily tasks. We monitor the users' interactions with the database and construct *user profiles* based on past interactions. Depending on the application domain, scalability and/or security concerns, instead of profiling single users we can profile users in separate groups based on their roles. For example, in the graduate admission database (*GradVote*) we used in [94], there are three roles: *faculty*, *staff*, and *committee chair*.

The atomic unit of interaction includes a SQL query (its syntax), the data region it accesses, updates, inserts, or deletes as represented by the returned tuple set, new tuple set, or changes to existing tuples (its semantics), along with contextual attributes such as timestamps, the issuing GUI/screen, the query source machine's IP address and port numbers, etc. All these information can be used to form a *feature vector* representing the atomic unit of user interaction with the RDBMS. Computing a feature vector is called *feature extraction*. A user profile will be built from a temporally sensitive history of feature vectors associated with a (group of) user. The *insider threat detection engine* consists of statistical learning *models* on the feature space and detection *algorithms* working in concert to monitor activities and adjust threat levels, raising an alarm whenever a user deviates "significantly" from his/her usual RDBMS interaction footprint. The alarm does not have to be binary; it could be a threat/confidence level, and thus the classifiers can be "soft" [92].

While it is not too hard to envision how to extract features from the query syntax (SQL, say) and contextual attributes, it is not clear how one might extract features from the data region being accessed or updated. First, the number of features we can extract from the contextual attributes and query syntax are data independent; hence, they will not blow up the problem's dimensionality. (There is a fixed and relatively small limit on the length of a SQL query.) On the other hand, the set of tuples returned by a SELECT SQL statement is data dependent, and thus is potentially unbounded; We do not know at model design time how large the database is. Second, the existence of queries that modify the database makes it even harder to model the data access/update behavior.

We address the difficulty by looking at the problem in the following way. For the sake of clarity, let us ignore the query syntax and contextual features and concentrate on the semantic features in order to briefly present the intuition behind our approach. Let's visualize a database as a single relation, called the *Universal Relation* [44, 93], incorporating the attribute information from all the relations in the database. Each query by a user accesses, modifies, inserts, or deletes a subset of tuples in the universal relations.

For further simplicity, let us first concentrate on the case when there is no modification query. This is true for relatively static databases such as a human resource database. When there are infrequent changes, one can always re-train the statistical models. We will outline some strategies for dealing with frequent update queries in the next section. The universal relation is a set of data points in a very high dimensional space called the *data space*. The dimension of the space is the total number of different attributes in the universal relation. Each query accesses a region, or a set of points (tuples), from the data space. A user profile has to accurately represent a collection of regions that the user accessed in the past. For example, a bank teller might access tens of non-business customer records every day, mostly on checking accounts. The major problem, as alluded to earlier, is that each accessed region might be too large; one cannot simply concatenate all data points in the region into a feature vector.

In [94], we proposed the following idea. For each query we compute a statistical "summary" of the data region the query accesses. The summary for a query is represented by a vector called the *S-vector*, whose dimensionality is fixed regardless of the size of the corresponding data region. The S-vectors lie in the *feature space*, which is related to but very different from the data space. Past queries (i.e., normal queries) from a user can be intuitively thought of as a "cluster" in the feature space. (We emphasize that the term clustering is used here to convey the intuition behind the approach, but we will not solely apply clustering-based learning algorithms.) When a new query comes, if its S-vector "belongs" to the user's cluster, it will be classified as normal, and abnormal otherwise. A key question is, of course, how to construct the S-vectors and design statistical models to capture user profiles based on them.

Preliminary results. In [94], S-vectors are composed of real-valued features, each representing a statistical measurement computed from the data region that the query accesses. *Intuitively, an S-vector is a statistical summary of the semantics of the query.* Specifically, each attribute of the universal relation contributes a number of features to the S-Vector according to the following rules. Each *numeric attribute* contributes the statistics *Min*, *Max*, *Mean*, *Median* and *Standard deviation*, where the statistics are computed over all or a specifically or randomly chosen subset of the returned tuples. For *non-numeric attributes* such as *char* or *varchar*, the standard statistics do not make sense. For *categorical attributes*, for instance, the typical route is to expand a k-value attribute into a sub-vector of k binary-valued attributes and compute statistics on them as usual. However, the expansion of categorical attributes may result in an S-vector that has far too many dimensions, affecting the time-performance of the learner. We compromised by replacing each categorical attribute with two numeric dimensions representing the *total count* of values, as well as the number of *distinct values* for this attribute in the query result.

The dimensions of the S-vectors are completely determined by the database schema and independent of the number of tuples returned by the corresponding query. Note that although this work did not take into account syntax and contextual features of queries, its results are already encouraging. We performed our evaluations on GradVote, a database at SUNY Buffalo used for evaluating graduate school applicants. In addition to testing the approach described above, we also experimented with several other strategies for constructing S-vectors.

The first type of insider attacks we examined was the *role masquerading attack*. Here, the test database GradVote has three major user roles: *chair*, *faculty*, and *staff*. We compared this approach with the syntax-based approach from [71]. After an extensive set of experiments, it was determined that the k-mean clustering algorithm consistently worked better than the approach from [71]. The second type of attack that we examined was *data harvesting*. Our query-semantics method attained a much higher detection rate than the syntax-centric approach. In fact, the syntax-centric approach of [71] does not work at all for this attack type. The focus here is on detecting syntactically similar queries, but differ in output data (data-values, output volume, or both). This is a significant query anomaly since, in a typical attack, a minor variation of a legitimate query can output a large volume of data to the attacker. This may go undetected and may be exploited for the purpose of data-harvesting. In other attack variations, the volume of the output may be typical, but the data values may be sensitive. We designed a couple of clustering-based outlier detection models under the L_2 and L_∞ norms, which have good overall detection rate and suffer from 15-20% false positive rates.

Main research problems. The core ideas presented above and the preliminary works opened the door to a host of interesting and challenging problems. The following are not independent research problems as the answers to them have to work in harmony to address the main issues described in the four thrusts described in the Introduction. Table 1 explains how they fit into the four thrusts.

1. *Better feature extractions.* Beside the obvious objective of dimensionality reduction to make the problem computationally feasible, feature extraction (when done correctly) also allows us to *improve* the accuracy of our learning models because "noisy" information can be discarded via feature extraction. The S-vector construction in [94] exhibits unimodal behaviors which are likely to be unrealistic. We need better feature extraction methods that retain more information.

2. *Incorporating domain knowledge of experts.* The M&T bank security team has many years of expertise in identifying security holes in their banking system, and their databases in particular. They are committed to helping us to construct a taxonomy of the types of access patterns that could be considered threats *and* non-threats, and will help us to perform red-team exercises. It is extremely unlikely that a generic learning model without domain knowledge will provide us with a satisfactory answer. Hence, it is crucial that we build our learning models with "knobs" that can be tuned using feedbacks from experts, and with exceptions that can be used to filter both the false positive and false negative cases.

3. *Reducing the false positive and false negative rates.* The current false rates of 15% to about 20% of the generic models in [94] are still too high, though better than the syntax-based approach. To reduce the false positive rate we plan to design more expressive models, separated for different threats. In particular, hierarchical and mixture models which capture a much wider range of activities will be a focus.

4. *Dealing with dynamic databases.* The clustering strategies in [94] outlined above worked relatively well for database that are static. When the database is dynamic, in addition to accessing a region in the data space each query can also insert new points into the space, move points from one part of the space to another, or

Table 1: How the problems fit into the four thrusts

Problem	1	2	3	4	5	6	7	8	9
Thrust 1	X		X				X	X	
Thrust 2	X	X	X			X	X	X	
Thrust 3	X	X		X	X	X	X		
Thrust 4						X		X	X

remove a region of the space. The main question is how to "summarize" such queries, extract user profiles from data updates. On the one hand, one can think of data updates as a collection of mappings from the data space to itself. On the other hand, there is a strong sense of a dynamical system at work where temporally related movements of data points form a pattern of user profile. Both modeling strategies pose distinct challenges. Learning transformations is a highly interesting question which should have applications elsewhere (perhaps in robotic and motion learning).

5. *Modeling and detecting attacks spanning over time.* A closely related problem to the dynamic data update problem is to model and detect more stealthy attacks. The experiments in [94] were conducted in settings where we want to detect single query attacks. A smart attacker will spread out his attack over some longer window of time (or window of queries) so that individual queries are not abnormal but collectively over some period of time they are. By making finer use of many other contextual features available in a database setting, such as transactional operations and timings, source machine, query inter-arrival times, session durations, and the likes, we should be able to detect slower attacks. Note that even when a group of near-by queries can be deemed anomalous, the attacker can mix and match attacking queries with normal queries to camouflage the attack.

6. *Making more sense of syntax and contextual features.* In a masquerading attack, e.g., one can envision the attacker trying to mimic the syntax of a normal SQL query to obtain a different query result. By exploring the correlation between the syntax of the input query and its surrounding contextual attributes such as source machine and IP address, time of day, etc. and the output tuples, we might be able to detect subtle anomalies. Even the ordering of the returned tuples is also an important piece of information. (A faculty member, for example, might constantly be ranking candidates in the graduate admission database by GRE scores.)

7. *Constructing threat models categorizing the types of attacks we aim to detect*. The threat models drive the evaluation process. A threat model should be "just right" in the sense that it is not too complex to the point of being intractable, and not too simple to the point of being practically useless. There is a natural tension between false positive and false negative rates. False negative rate can be improved by specifically tailoring the statistical models to specific threats. The M&T bank security team has committed to helping us with this problem.

8. *Building a prototype and evaluation framework.* Computing a feature vector from a given query is an absolutely non-trivial task from the point of view of database engineering. The obvious approach of computing the query, store the results somewhere to extract the feature vector out is both very taxing on the system and perhaps even unnecessary depending on the types of features we need to extract from the query results. Furthermore, when there are updates on the data, it is not clear how one even "stores" the intermediate results for feature extraction. There are also serious privacy issues we need to address when we build, test, and deploy the prototype system at M&T site.

9. *Using database operations for statistical model building.* To address the space and time complexity requirements for feature extraction, model training, we will also study how to use existing database technologies as a "blackbox" to train the models. This is a rather intriguing problem that will have applications far beyond database security.

In the next section, we outline our plan to attack the above problems.

3 Proposed Research Plan

Reducing the false positive and false negative rates is perhaps the most important and the most difficult problems in any intrusion detection system in general and in insider threat detection in particular. Hence, while we plan to address the major research problems presented above, let us summarize right off the bat specifically on how we plan to reduce the false positive rates, before delving deeper into technical details of the strategies provided in sub-sections that follow. In addition to the query-semantics idea, we believe that the combination of strategies outlined below is novel in the intrusion detection and insider threat detection arena.

	Strategies
False positive rate	Separate soft-classifiers for separate threats Enrich model power by considering hierarchical and mixture models Incorporate organizational structure into hierarchical non-parametric or Bayesian model Better feature extraction, incorporating domain knowledge, syntax and contextual features
False negative rate	Separate soft-classifiers for separate threats More concise threat taxonomy Penetration testing and red-teaming exercise done by M&T security team Better feature extraction, incorporating domain knowledge, syntax and contextual features

3.1 Outline of strategy

The main research problems outlined in the previous section are closely related. Our strategy for dealing with them is as follows. At a high level, the query-semantics approach to insider threat detection is a generic learning problem: we seek to learn a collection of mappings $f_y : \mathcal{X} \rightarrow \{0, 1\}$, where $\mathcal{X}$ represents the space of an individual user's database access footprints, and $y \in \mathcal{Y}$ represents the threat signature in question (such as data harvesting, sabotage attack); $f_y(X) = 1$ means that the database access footprint X is considered an insider threat of type y; $f_y(X) = 0$ means no threat. Learning separate (soft) classifiers for separate threats is already very different from the way we dealt with the problem in [94] where one classifier was trained for different threat types; this strategy is expected to improve both the false positive and false negative rates.

As discussed before, standard off-the-shelf machine learning algorithms (e.g., for clustering, classification, regression) are not readily applicable to this problem. A major component of the proposed research is to develop new statistical machine learning techniques in order to achieve state-of-the-art performance by reducing both false positive and false negative rates for insider threat detection. We accomplish this by taking advantage of structural dependencies that exist in the space of DB access footprints $\mathcal{X}$, and structural dependencies among various attack signatures. We anticipate that incorporating these structural dependencies in a principled manner will require nontrivial extensions of existing machine learning technologies, and the development of fundamentally new statistical and algorithmic ideas.

A false positive indicates that a normal behavior was not captured well by the model. In a detection problem the false positive rate can be improved by having more expressive and accurate models for *normal* DB access patterns for all insider users. In particular, we will look for better feature extraction methods and more powerful models. A false negative means we do not understand the threat well. To reduce the false negative rate, we need a more concise characterization of the threats and somehow embed this knowledge in the model. Accuracy can be improved in both cases by obtaining decision functions f_y with a stronger statistical correlation to actual attack signatures (via more accurate threat models). The M&T security team will help us with constructing a more realistic and concise taxonomy of threats that *they* actually care about. We next describe our modeling ideas and then discuss strategies for translating models to actionable detection algorithms.

For dealing with dynamic databases, the feature extracted can no longer be a simple set of tuple statistics. At the very least, we need to indicate the operation type (update, insert, delete, access), along with the statistics of the data to be updated, inserted, deleted, or accessed. Thus, the first step toward dealing with dynamically changing databases is a better feature extraction method. A potentially difficult problem imposed by dynamism in the data is that abrupt changes to the data might throw our model off-guard as the model was trained based on past data. Fortunately, if our model is any good, then it should be able to detect the abrupt (malicious) change in the data. And, expressive mixture models can also deal with abrupt changes by adding a new mixture component.

Coping with attacks that last over a time interval is also difficult. The obvious approach is to enlarge the feature vectors to include a window of past queries, taking into account temporal information such as time of day, inter-query times, and system information such as source machine and the issuing GUI. For smart attacks that combine normal

and malicious queries, we will need a way to model how the attacker's knowledge is accumulated over time, using some Markov-like models. The tight coupling of the threat models and the statistical model will definitely help.

Overall, the main research components toward solving the problems are better (i.e., more expressive) models that include better feature extraction methods, the translation of models into detection algorithms, and finally realistic threat models working in concert with the other two components.

3.2 Feature extraction and models for user profiling

In prior work [94], the information given by user query $X \in \mathcal{X}$ is described by a summary vector of *feature vector* called *S-Vector*, denoted hereafter by $S(X)$. Then $S(X)$ is fed into a standard machine learning algorithm for anomaly detection. Although economical from a computational standpoint, this choice presents detrimental strictures resulting in substantial loss of information on a given user's access profile. First, many components of a deterministic summary vector $S(X)$ tends to exhibit unimodal behaviors, which are typically unrealistic — we expect that any user can exhibit reasonably diverse typical access behaviors depending on different types of normal activities. Second, due to the law of averages many components of $S(X)$ may not have sufficiently distinguishing effects across different users. Neither do they exhibit noticeable changes over time, if abnormal activities are (intentionally) embedded within larger proportions of normal activities. This suggests the need for appropriate expansions of the feature vector $S(X)$, and a richer modeling approach to capture its heterogeneous behavior. For example, the vector of summary statistics $S(X)$ can be augmented with database-specific information about the level of sensitivity of records that a user query attempts to access to. It is here that domain expertise is required to tune meaningful "knobs" of the statistical models.

To obtain more expressive representations of a normal access behavior by each user we will consider the use of mixture models. The idea is to capture potentially multi-modal access patterns: the same user or group of users might perform different types of tasks with distinct access signatures. Key to the mixture model is the notion of a mixing discrete measure (distribution), $G = \sum_{i=1}^{k} p_i \delta_{\theta_i}$, which specifies typical access behaviors parameterized by θ_i's and mixing proportions p_i's. Given a typical behavior θ_i, there is conceptually a conditional probability distribution $P(S(X)|\theta_i)$, and more generally $P(X|\theta_i)$. The use of a relatively low dimensional summary vector $S(X)$ makes it amenable to parametric specifications for the conditional probability distribution. For instance, we can use Gaussian distributions, leaving the means and covariance matrices to be empirically estimated from data. The normal mixture model can capture very complex heterogeneous behaviors when the number of mixing components (k) is large.

We will also aim to exploit statistical sharing and heterogeneity derived from user groupings, which strongly correlate with access privileges and behaviors. For each user u there is a mixture model G_u, herein subscripted by the user index. Specifically, we write $G_u = \sum_{i=1}^{k} p_i^u \delta_{\theta_i^u}$. The data (i.e., access activities) for each user may be sparse. This issue can be overcome, however, due to the observation that users are *a priori* designated into different groups and subgroups with different access privileges and responsibilities. It is expected that users that belong to the same groups (or subgroups) usually share a subset of typical access behaviors. As a result, it is possible to borrow statistical information from other members in the same group to make inferences about the behavior of any individual user, even if we do not have enough data to support for the whole range of behaviors permissible according to his or her privileges. For example, inside members in a bank's database can be subdivided into different groups such as board of directors, team leaders, and employees. Employees may be further subdivided into groups focusing on business loans, consumer finance, credit analysis, human resources, IT support, etc. There are also overlapping subgroups endowed with different privileges. Evidently, the grouping need not be exclusive – some users are associated with multiple group memberships whose range of activities are expected to be more diverse than others.

The above intuitions motivate the development of *hierarchical modeling*, an effective approach that has seen much progress in recent machine learning and Bayesian statistics literature. To simplify our discussion, suppose for now the grouping can be organized as a tree structure: A group may be subdivided into disjoint subgroups, each of which may be further subdivided. A node in the tree corresponds to a subgroup. A user belongs to only one path in the tree. Following the standard hierarchical modeling approach, we envision a mixing distribution $G_0 = \sum_{i=1}^{k} p_{i0} \delta_{\theta_{i0}}$, which represents all typical activity behaviors if we are to pool all users together (regardless of their grouping associations). Each group v, can be represented by a mixing distribution G_v, which varies around the centering distribution G_0. Suppose that group v (consumer finance) is subdivided further into subgroups vw (e.g., general banking, credit cards and mortgages). Then there are mixing distributions G_{vw} associated such subgroups, and the G_{vw} can be assumed to vary around G_v. Finally, a user that belongs to subgroup vw is endowed with mixing distribution $G_{vwu} = \sum_{i=1}^{k} p_i^{vwu} \delta_{\theta_i^{vwu}}$.

Hierarchical modeling, either in a parametric (cf. [138]), or a nonparametric Bayesian setting (e.g., [158]) is quite natural for a tree-structured organization of user profiles, especially if subgroups are non-overlapping and statistically

exchangeable with one another in the same level of the tree structure. In reality, both assumptions may be too restrictive for the databases that we consider. In a recent work [121] we proposed a nested hierarchical Dirichlet process framework that no longer requires the exchangeability among groups. In particular, the mixing distributions G_v associated with groups v's need not be centering around the same G_0. This approach will be extended to accommodate the modeling of profiles for users who are associated with more than one subgroups.

Incorporating domain knowledge of privileges and typical behaviors. The usefulness of the model hinges on the ability to incorporate effectively domain-knowledge of the database and its usage, according to the known organization structures, its entailed privileges and rules. Organization structures can be encoded in the construction of model hierarchy, as already mentioned above. Members from the same designated groups are expected to have statistically similar access profiles. The knowledge about privileges and rules is crucial in designing relevant features (variables) that define the model. For instance, a member in the IT support groups may have access to the computing accounts by others, but not bank-wide accounts. The credit analysis group has partial access to bank-wide accounts, while the corporate finance group has access to the credit analysis group's records but only limited access to individual banking accounts. These accesses can be represented by a collection of features that are essentially count variables of accesses or attempted accesses within a certain time window. These variables are time-stamped and also indexed by locations of database which are designated by users' privileges – they are represented by $\theta_i^u, \theta_i^{wu}, \ldots$ and components of $S(X)$.

By virtue of nonparametric Bayes, the number of typical access behaviors is left unbounded and can be learned directly from data. This is a desirable feature in our approach to detecting unanticipated anomalies, because fixing k may inject undesirable bias in the estimates of typical normal behavior, negatively impacting both false positive and misdetection rates. In a nonparametric Bayesian approach, as more data (queries) are taken into account, more mixing components (clusters) may be introduced into the (unsupervised) learning and inference. A new component is introduced because it is significantly different from existing components. The addition may be attributed to an unusual activity that is potentially part of an imminent or on-going attack, but it may also be a relatively new but perfectly legitimate access pattern that can eventually be verified with more supporting data. It is not (yet) necessary to raise a flag at the addition of a new mixing component — this important task lies in the design of detection (prediction) algorithms that we describe in the next section, where we also describe how to incorporate structure dependence existing in threat models.

3.3 From models to detection algorithm

Given a query pattern $X \in \mathcal{X}$, which is represented in terms of summary vector $S(X)$, we also obtain additional latent features, denoted herein by $\theta(X)$. ($\theta(X)$ contains both typical behaviors and associated proportion probabilities). At the simplest level, both $S(X)$ and $\theta(X)$ can be fed into a standard anomaly detection algorithm. For concreteness, we will use a state-of-the-art one-class support vector machine (SVM) (see, e.g., [143] for basic ideas), although other methods will also be considered. Central to the SVM is the notion of a kernel function $K(X, X')$. We define $K(X, X') = K_1(S(X), S(X')) + K_2(\theta(X), \theta(X'))$ to combine kernel function for both"raw" data $S(X), S(X')$, and latent features $\theta(X), \theta(X')$. Depending on the mixture model specifications, statistics of the latent features, which may also be random, can be obtained. Thus, the similarity measure for $\theta(X), \theta(X')$ may be evaluated by its expectation $\mathbb{E}K_2(\theta(X), \theta(X'))$, which can be obtained as part of the posterior inference in the mixture models described earlier.

The key ingredients of the proposed research is to develop principled methods for incorporating specific features in our threat models to improve the performance of detection algorithms for specific threat signatures. **It should be noted that the term signature is used in this proposal in the statistical sense**. Recall our basic machine learning formulation of achieving a collection of binary classifiers $f_y : \mathcal{X} \to \{0, 1\}$, where y represents a specific threat category in $\mathcal{Y}$. We identify two venues of developments for insider threat detection: (A) To exploit the structure within each threat signature y, so as to develop a ranking-like detection function f_y that maps $\mathcal{X}$ onto a richer discrete space, which represents varying degrees of danger regarding a specific threat category. (B) To exploit the relationship between different threat signatures in order to achieve "transfer learning." Because there may be sharing in attack patterns between different attack categories y, it is desirable to combine the data and inference for all related categories in order to improve the performance for individual threat categories.

Exploiting expert knowledge of attack signatures. To motivate (A), consider a general insider attack sequence consisting of intrusion, privilege escalation and goal [60]. (For an insider attack, the reconnaissance stage in its traditional sense is non-existent). Each stage of the sequence can be considered as a threat. Thus, it would be more useful to produce a detection function f_y taking values in the space of attack stages rather than binary symbols. Recall the three incidents discussed in Section 1: a HELOC wire transfer fraud, a trader's fraud, and a browsing attack. In

either case, the intrusion step may be captured by the latent variables that describe the (ab)normal access behaviors as they are constrained by existing user privilege and associated typical profiles (e.g., mixture means). The level of uncertainty is captured by the posterior of the latent variables, through which we may or may not be able to raise an alarm. According to our threat signature, the attacker is likely to proceed to the next stage, i.e., privilege escalation. This could give us further information and reduce the uncertainty of our inference about the nature of accesses. Indeed, as the volume of the accesses exceeds a threshold, we may be able to raise an alarm regarding the trader's fraud and the browsing attack. The HELOC wire transfer fraud would have been detected, in coupling with the unusual access to a certain part of the database, by the unusually large amount of transfer.

We note that the learning of f_y cannot be formulated as a multi-class classification problem, because there is no inherent relationship among the classes. Regression is also inappropriate. f_y is most closely related to a ranking function [91]. In contrast to standard ranking problems, a challenging issue is related to the lack of training data with labels. Without many labels, a learned detection rule is defined by thresholding f_y, where f_y can be viewed as an approximation (e.g., via compositions of kernel functions defined above) of the likelihood ratio between the empirical distribution and the normal model. To improve the learning efficiency for threat models of interest, a possibility is to use threat models (developed as part of our evaluation) to simulate the attack labels. This also provides a way of evaluating the effectiveness of our threat models later to be applied to real data. Another possibility is to exploit the natural temporal correlation of attack sequences, by considering sequences of queries over a window of time, rather than aiming to rank queries individually. We will elaborate on this in the sequel.

Approach (B) is an extension of (A), by noting that a distinguishing feature of the threat detection problem is the lack of data example that provides support for many different attack categories. Some categories are typically more rare than others and accordingly harder to detect, yet they may also be the most damaging. We propose to exploit the sharing of features/entities across different categories by combining them in a fully integrated learning framework. The goal is to intercept not only attacks for which specific training examples were given, but broader classes of similar attack signatures. We envision the use of a tree structure akin to a decision tree, where each node is associated with an attack stage akin to CAG [32]. Each threat category is associated with a path in the decision tree, allowing paths to share subsequences of nodes. The overall goal is to learn a function which maps one or multiple user queries onto a path or a node in a path of the tree. Intuitively, queries that contain relatively strong evidence should enable the decision function to land on nodes that are closer to the leaves, while nodes closer to the root represent weaker confidence of the prediction.

Both (A) and (B) can be implemented within a single decision making framework: (A) can be viewed as learning individual paths (chains) for each attack categories, while (B) is a more ambitious effort at integrating all these paths together in a decision tree. Moreover, rather making inference on a single user query, we believe that both ideas will be much more effective when multiple queries collected within a time window are considered together.

Dealing with attacks spanning over time. Stealthy insider attacks are rarely detectable based on the analysis of a single access query. One has to string together a sequence of queries collected over certain time windows in order to make credible prediction about an insider's intention. To be clear, the objective is to learn decision functions which map inputs in the form of sequences of queries onto outputs in the form of specific attack signatures. We discussed previously several key modeling, data structure and algorithmic ingredients. The use of a hierarchy of mixture models helps to reduce the dimensions while retaining the heterogeneity of normal behaviors present in the input space. The use of decision chain and tree structures in the output space also helps to accommodate the transfer learning of detection functions and prevents overfitting issues.

At the simplest level, the input is taken as a sequence of activity labels (obtained via the inference with mixture models). Discarding the information on the temporal ordering of the labels, the proportions of activity types provide very useful statistics for making inference about the user's intention. (Temporal correlations may be easily taken into account by using bigram and trigram statistics). Suppose that we have k activities labels, then an input point corresponds to a point in the $k-1$ probability simplex. Threat categories are associated to partitions of the simplex, with partition boundaries defined by detection functions. The partitioning of the output space, however, can be represented by a cascade of increasingly refined decisions via a decision chain/tree described earlier, suggesting the adaptation of structured prediction techniques (e.g., [40, 156]), especially if training data with attack labels are available.

When no such training data are available, we consider taking an unsupervised learning approach. The problem described in (A) has some connection to the problem of part-of-speech tagging, for which a common tool is based on hidden Markov models or other dynamic models designed for sequential data. To be clear, the standard tagging problem presumes the possibility of tagging individual queries with attack stages by exploiting the latent sequential

dependences (e.g., attack stages follow a hidden Markov chain), A notable aspect in our problem setting is that a stealthy attack sequence may be intentionally embedded within other normal activities. It is through accumulation of certain types of activities over time that enable the inference of specific attack stages. We anticipate extensions of the HMM framework in which the transition between hidden stages need to respect the constraints imposed by the decision chain/tree structures. Detection algorithms can be derived from smoothing/filtering algorithms in a dynamic model in a standard way.

Model-free dimension reduction of the query space using divergence measures. We end this section by describing briefly an approach for reducing the immense dimensionality of input space $\mathcal{X}$, which is hitherto accessed only through the summary statistics $S(X)$. For certain threat categories $S(X)$ may not be suitable, because operator S wipes out most relevant information needed for the inference of such type of attack. In this scenario, we need to go back to the conceptual data-driven view of a query X, which represents a probability distribution on the space of records in the DB. It is not possible to specify a full probabilistic model for X (mathematically speaking, X lives in a space $\mathcal{X}$ of unbounded dimensions). However, it is possible to estimate the divergences between probability distributions, using only samples from such distributions. Such model-free divergence estimation methods are readily available [128, 164]. The divergence measures between probability distributions may prove to be useful distance measures for the space of queries. Indeed, the distances between queries can be fed into a nonlinear dimensional reduction algorithm (such as the multi-dimensional scaling method) to obtain a low dimension representation which retains approximate distances between original queries. From here the modeling and algorithms can proceed as before.

3.4 Prototype and Evaluation

The expected outcome of our research project is a practical software system that can be used to detect a large class of realistic insider attacks in RDBMSs, with an emphasis on the finance sector. This system must not only be accurate (i.e., have low false positive and false negative rates), but must also be computationally feasible in real-world deployment. The prototype and the evaluation pipeline will be constructed in parallel with the other research activities for two reasons: development is expected to be a long process, and all basic components of our research need constant evaluation and feedback.

3.4.1 Databases at M&T Bank used for evaluation

Our collaboration with M&T Bank affords us the ability to address both concerns. Together with our industry consultants, we have identified two active databases at M&T Bank that are exemplary of databases at high risk of insider attacks. Both databases are run on a widely used commercial DBMS[1]. The databases are "available" for evaluation in the sense that the M&T security team will be able to help us construct our prototype, anonymize data when necessary, conduct red-teaming attacks and speculate on potential loopholes both in our system and in their system. The first database (DB1) is a human-resource database that is 160GB large, has 2000 tables, and about 50 users. The second database (DB2) manages wire-transfers, is 80GB large, has around 500 tables and 30,000 users. Together, these two databases process a little over 10,000 queries (approximately 1,300 transactions) per day on average. We believe that these databases will be sufficient for validating our initial proof of concept. If the initial proof of concept is successful, we will identify further databases suitable for use in validating our results.

M&T Bank is already engaged in logging basic session information for accesses to these database systems for regulatory reasons, using a commercial database activity monitoring system. We have confirmed the feasibility of sustained logging of the full set of queries issued during each session on both databases. We have also confirmed the possibility of obtaining clean-room access to the databases themselves, which will be sufficient to perform any necessary testing of our prototype.

3.4.2 Evaluation, penetration testing, and red-teaming exercise

The obvious evaluation criteria are to reduce the false positive and false detection rates. These rates are directly influenced by how we "inject" attacks into the system and thus by how we define attacks. The threat modeling component of our research is precisely to address this point. There are common-sense attacks that can be partly modeled using statistical approaches such as data harvesting attacks (where the data regions accessed by individual queries suddenly shift or get too large in volume), role masquerading attacks (where one user's access patterns resemble another). There are known attacks in the finance domain that can be re-played. There are attacks that can only be envisioned by experts who know the target domain very well and some of these attacks will be identified and classified by the M&T Bank security team. They will help perform penetration testing and deeper red-teaming exercise; the knowledge gained from such exercises feeds back to tune our models and creates exception cases for both normal and abnormal

[1] We avoid naming specific software packages at the request of M&T Bank's security team

activities.

We plan to judge detection rate based on three classes of previously identified threat: (1) Threats from previously identified attacks, (2) Threats injected into the trace data by our consultants, and (3) One or more hypothetical simulated attack scenarios, designed by our consultants, and executed by one or more normal users of the exemplar databases. It is important to note, as alluded to earlier, that our classifiers can and will be "soft" classifiers where confidence levels are assessed instead of a binary alarm/non-alarm recommendation. The information flow is not just from the domain experts to us and our model; we expect our model to be able to identify (seemingly) anomalous activities within their system and the experts will be able to explain to us whether such activities are considered normal. We strongly hope that our models will pin-point strange patterns that they have not looked into. In the end, though, the benchmark will still be the false negative and false positive rates, perhaps weighted by the confidence scores of the soft classifiers.

3.4.3 Prototype development

We will begin with a simple prototype based on our prior work. This will allow us to refine our strategy for deploying code alongside and into M&T Bank's production environment, while simultaneously providing a set of baseline metrics (accuracy, processing rate, resource consumption) for later comparison. Our initial prototype will operate in an offline mode, reading in access logs generated by a commercial database logging system already in use at M&T Bank for regulatory compliance. Database state will be initially assumed to be static; The offline processing task will read necessary database state directly from the database during a low-use period. If necessary, the offline processing task will read from a copy of the database state cloned during a low-use period. From this foundation we will further develop the prototype through three stages: (1) Accuracy, (2) Efficiency, and (3) Extensions.

Stage 1. Although efficiency will be a concern, we will focus on the accuracy of our prototype on *static* databases. Our prototype development will start with *Thrust 1* problems where the main aim is to apply several ideas described in the previous section to improve the false positive and negative rates: separate soft classifiers for separate threats, better feature extraction using domain knowledge, mixture and hierarchical models that are much more expressive than the ones used in [94]. We will validate our prototype in this stage by training and testing on recent and historical traces of query activity on the two exemplar databases. Testing will be carried out on a secure testbed machine located within the financial institution's security domain. Whenever necessary, an anonymized copy of the databases and traces will be used for testing and development as well. Although we will gather performance metrics at this stage, our primary consideration will be accuracy (i.e., the detection and false positive rates).

Stage 2. We will address the challenges involved in deploying our prototype into a live production environment. The primary research challenge of this stage will be ensuring that it is possible to classify user activity both in a timely manner, and using only readily available hardware resources. We will explore a range of strategies: (1) Applying batching and shared scan techniques to improve offline classification performance, (2) Developing cost estimation for classification to better allocate compute resources, (3) Identifying fragments of the online classification process amenable to precomputation techniques, and (4) Identifying ways to exploit incrementality in the classification process. We will continue to validate the prototype's performance and accuracy, as in Stage 1. At this stage we also start to consider performance metrics such as memory and disk utilization, scalability, and classification rate. We will compare these results against measurements of the rate and growth of utilization of the financial institution's existing database systems. Last but not least, expert domain knowledge will be incorporated into our prototype in the forms of exception lists and model parameter tuning.

Stage 3. We start evaluation of the prototype on dynamic databases. We will start with a simple attacker model where the entire attack is allowed only a single query. However, in contrast to our prior work, we will treat insertions, updates, and deletions as a fundamental feature of user interaction with the database when designing our classification algorithm. We will examine more sophisticated attacks such as *stealthy attacks* (those spread out over multiple queries) and *coordinated attacks* where multiple attackers each executes a fragment of an attack. The notions of database/financial transactions will be incorporated into the prototype. Transactional properties such as duration, distribution of operations, and roll-backs are potential features. More nuanced deployment issues are considered such as dealing with *service accounts* (which access the database on behalf of other users, e.g., web applications), friendlier GUI for the detection engine, and the tradeoffs between accuracy and time/space complexity of the detection engine.

4 Related Work

Database security. The literature on intrusion/anomaly detection (IDS) is very large, some of which focused directly on databases [14, 22, 30, 37, 50, 64, 70, 72, 81, 88, 90, 130, 163, 168, 171]. There is considerably less research on dealing

with insider threats in general, let alone in databases. While IDS and insider threat detection have similarities, the insider vs. outsider difference in the two problems creates an absolutely non-trivial barrier for trying to adapt IDS solutions to the insider threat problem [14,89]. Cryptographic approaches to attack detection [3,19,49,59,98,150] are both orthogonal and complimentary to our approach, and can be used to provide an additional layer of security on top of our techniques but are not reviewed here.

In [87], temporal properties of data are utilized for intrusion detection in applications such as real-time stock trading. Anomaly detection schemes dealing with SQL injection attacks in Web applications were studied in [80,162]. SQL injection attacks are a specific kind of database query anomaly that can be detected by our query-semantics approach in a straightforward manner as shown in our prior work [94]. Data correlation between transactions is used to aid anomaly detection in [66]. Similarly, dependency between database attributes is used to generate rules based on which malicious transactions are identified in [153].

The DEMIDS system [34] detects intrusions by building user profiles based on their working scopes whose features are syntax-based. In [86, 139], syntax-based systems were proposed where SQL statements are used to build "fingerprints" of legitimate transactions. In [25,46,148], database transactions are modeled using some kind of graphs or Petri nets to capture the execution paths of normal activities. Database session identification is the focus of [170] where an entropy-based model was used to profile queries within a session. A multiagent based approach is presented in [135]; relatively simple metrics such as access frequency, object requests and utilization and execution denials/violations are used to audit user behavior. A specific kind of threat, viz. data modification attack is addressed in [132] by preventing write operations that cross a certain threshold value for the data.

Prior approaches in the literature that most resemble ours are [152] and [71]. The solution in [152] is similar in the use of statistical measurements; however the focus of the approach is mainly on detecting anomalies in database modification (e.g., *inserts*) rather than queries. The query anomaly detection component is mentioned only in passing and only a limited set of features (e.g., session duration, number of tuples affected) are considered. The recent syntax-based work in [71] has the same overall detection goals as our work: detection of anomalies in database access by means of user queries. Although the limitations of syntax-based detection schemes have been exposed by our earlier work [94], the approach in [71] will be used in our research as a benchmark for evaluating the performance of our approach. Commercial tools such as DBProtect [61] are now available for securing databases from misuse and abuse but they represent largely audit-analysis based (first generation) solutions.

More closely related to our own approach is a class of auditing strategies that attempt to identify the source of an attack that has already occurred; The attacker is assumed to have full logical access to the data, albeit through an interface that logs their actions. Techniques for dealing with this class of attacker typically use a "secret", or "audit" query to describe a fragment of the database that is considered sensitive, or that has already been identified as having been leaked. A set of queries is considered suspicious if they could have revealed the sensitive data in part [96] or in full [11, 97], where the works of [97] and [96] in particular test for syntactically suspicious queries – queries that could have hypothetically revealed information that would be considered sensitive, making it possible to employ these techniques to stop leaks before they happen. Auditing techniques have also been developed to verify compliance with changing access policies [45], or to determine the source of a leak where the leaked data is known [12]. Most of these techniques, however, rely on having an explicit classification of the sensitive or leaked data, either as a query, or access control policy (e.g., [165]). In each of these cases, insiders authorized by the access control policy or with legitimate reasons to access the sensitive data are immune to scrutiny. Our approach automatically infers each user's expected access pattern, making it possible to detect threats from users who have legitimate access to sensitive data. The approach in [12] has no such restriction, but is rather a forensic tool that compares a log of queries against a dataset that has already been leaked. This approach is complimentary to ours.

Statistical machine learning. Our problem should benefit from existing anomaly/outlier detection solutions. However, the widespread availability of massive amounts of digitized data in almost every imaginable application domain has added many new challenging facets to the basic anomaly detection problem, in addition to the intrinsic difficulty of the problem. For example, network intrusion detection and anomalous computerized stock trades require the anomalies to be detected in milli- (or even nano-) second time scale [42, 129, 151]. Credit card fraud and insurance fraud detection need algorithms to operate on giant data sets [21, 51]. Stealthy worm/virus propagation detection requires highly distributive machine learning algorithms [10, 155]. Botnet detection requires a fundamentally new set of tools and techniques [15, 52].

There are several nice surveys on anomaly detection from a security viewpoint [26, 63, 129]. There are also good references from a machine learning security angle [17, 18]. Textbooks have been written about statistical outlier

detection techniques, which are useful for security-related anomaly detection [16, 140]. The reader is referred to the above surveys and books for further background information on statistical machine learning methods for anomaly detection. All of the existing works with specific application domains in mind (databases, networks, credit card fraud) derived models geared toward solving a particular domain-specific problem and with respect to available training and test data sets. We expect our proposed research to follow the same path. Coming up with statistical models which can capture the intrinsic structure of a new problem domain is absolutely non-trivial, requiring new insights both mathematically and problem-domain specifically. Adopting "off-the-shelf" models does not work as there is a non-trivial barrier for trying to adapt IDS solutions [14, 89].

One of the most challenging aspects of anomaly detection is the unavailability of (plenty of) anomalous data points. For this reason, supervised learning algorithms are much less effective than they can be otherwise. To take advantage of supervised learning still, fault/anomalous injection methods were proposed for some problem domains [9,154,159]. Semi-supervised and unsupervised learning algorithms were also designed for this problem with limited scope [38,39]. Recently, the authors in [99] evaluated the effectiveness of several machine learning algorithms (one-class SVM [144], support vector data description [157], fast adaptive mean shift [35]) in detecting role-based masquerading in document control systems with some notable results; however, their evaluation and validation steps were based on synthetic data and thus the methods' generalization capability is not conclusive.

There are a number of important statistical and learning techniques that prove useful for an anomaly detection problem. Transfer learning is an umbrella term for statistical methods that enable the borrowing of statistical strength from one subproblem or subpopulation to another. This viewpoint is also particularly useful for our detection problem, as we shall elaborate later, since there are common substructures being shared across different types of attacks, as well common normal access behaviors across different users in an organization. For unsupervised learning, this can typically be achieved by hierarchical and graphical modeling methods [69, 84, 169]. The modeling of complex and structured data using mixture models and their hierarchical and nonparametric extensions has seen much progress recently (see [62] for a survey).

From a security perspective, various machine learning frameworks have been proposed to detect intrusions, which can be classified (as done in [41]) into host based, or network based methods. Host based methods can extract features from system calls generated by programs or users (see [39, 47, 64, 83, 149] and references thereof), or even from key strokes [78, 79] and other GUI features [136]. Network intrusion detection has a large literature. For representative surveys and results, see [70, 85, 146, 147].

5 Prior NSF Supports

Prof. Ngo was the PI of an NSF CAREER Award entitled "Designs and Analysis of WDM Switching Architectures" (CCF-0347565, Feb 2004 – Feb 2009). The project developed a complexity model, constructions, and routing algorithms for multi-channel switching networks. The project has produced 13 journal papers (including papers in IEEE/ACM Trans. on Networking and SIAM J. Comput.), and 30 conference papers (including INFOCOM, DSN, SODA, RAID), among others. Notable results include the solution to an 18-year-old open problem in the complexity of multirate distribution networks. He was co-PI on a recent NSF grant entitled "Collaborative Research: Sparse Approximation: Theory and Extensions" (CCF-1161196, Jul 2012 – Jul 2015).

Prof. Upadhyaya's prior NSF supports are 1) IIS-0916612, TC: Small: Online Privacy and Senior Citizens: A Socio-Technical Multi-Perspective Framework for Trustworthy Operations, H.R. Rao (PI), S. Upadhyaya and S. Bagchi-Sen (Co-PIs), 2009-13, resulted in seven journal and 11 conference papers; 2) DUE-0830814, SFS (Scholarship Track): An Interdisciplinary Information Assurance Curriculum, S. Upadhyaya (PI), H.R. Rao, T. Cusick, M. Bartholomew (Co-PIs), 2008-12, supported 11 SFS scholars (including one minority and two women) and resulted in two journal and four conference papers; 3) CNS-0420448, Women and Cyber Security: Gendered Tasks and (In)equitable Outcomes, H.R. Rao (PI), S. Upadhyaya and S. Bagchi-Sen (co-PIs), 2004-08, resulted in nine publications (including several IEEE Journals and Transactions).

Prof. Nguyen's recent prior NSF supports are 1) SI2-SS1: Real-Time Large-Scale Parallel Intelligent CO2 Data Assimilation System; PI: A. M. Michalak; NSF award details: OCI 1047871; 9/2010–8/31/2014. Thus far the project has supported two graduate students, and yielded two publications, two under review for journal publications and several published abstracts. 2) Distributed Detection Algorithms and Stochastic Modeling for Large Monitoring Sensor Networks; PI: X. Nguyen; NSF award details: CCF 1115769; 8/1/2011 – 7/31/2014. This project develops a computational and statistical framework that integrates the distributed computation and communication constraints of the underlying network infrastructure with flexible stochastic modeling and fast computation with spatiotemporal data. Thus far the project has yielded two publications, one journal paper under review, and several published abstracts.

6 Collaboration Plan

This Medium proposal spans two academic institutions and one major financial institution (M&T bank), consisting of four investigators, two consultants and a number of graduate and undergraduate students. This section describes the investigator/consultant capabilities and their roles, the coordination mechanisms across all the investigators/consults, cross-institution scientific integration and a detailed work plan and project milestones.

Investigator credentials and their roles. PI Ngo has worked extensively on combinatorial group testing, switching networks, and algorithms with a focus on database join algorithms. In group testing, he and co-authors discovered several methods for disjunct matrix constructions which are also highly error-tolerant and efficiently decodable in sub-linear time [68, 103, 104, 106, 113, 114]. In switching networks, he developed a complexity model for multi-channel switching networks, proved asymptotically optimal and near-optimal bounds for many classes of multi-channel switching networks [43, 101, 102, 105, 108, 110, 111, 119, 167], and devised new techniques for switching network blocking analysis [100, 107, 109, 115–118, 120, 166]. The paper [118] received the **best paper award** at COCOON'2008. More closely related to this project, he has also been working on several network security problems: combinatorial models for insider threat detection and mitigation, along with developing practical GUI-based tools [31, 32, 32, 53, 57, 94], and modeling and analyzing Internet worm and botnet propagation [54–56, 58]. The paper [112] received the **best paper award** at the 31th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS 2012). The paper gives the first known worst-case optimal algorithm for one of the most important and classic problems in databases: computing the join of an arbitrary set of input relations. PI Ngo will work primarily on the statistical modeling part of the research, but will contribute to the task of modeling and detecting attacks spanning over time and the overall system evaluation and validation.

PI Upadhyaya is the Director of the NSA- (National Security Agency) and DHS-certified Center of Excellence in Information Assurance at UB. He has worked on user-level anomaly detection [33], [160], threat modeling and assessment [32], protecting documents from insider attacks [131], malware detection [27], [29], data/applications security [28] and human centered security [141]. He has participated in panels and international conferences and edited several books. His work on malware detection got the **best paper award** at IEEE MALWARE 2007 [29]. In 2009, he co-edited a book entitled "Annals of Emerging Research in Information Assurance, Security and Privacy Services", published by Elsevier. He is co-founder of the Workshop on Secure Knowledge Management, which has been held every other year since 2004. His research has been funded by NSF, DARPA, AFRL, NSA/ARDA, IBM and Intel Corporation. PI Upadhyaya will work primarily on the threat modeling part of this project but will contribute to the prototype development, system evaluation and validation.

PI Kennedy's research interests include probabilistic databases and uncertainty, realtime analytics, and security issues. In probabilistic databases he has focused primarily on the usability of probabilistic databases, both by reducing the learning curve (Fuzzy Prophet [76]), and by enabling broader applications of probabilistic databases (PIP and Jigsaw [74, 77]). More closely related to this project, he has developed monitoring systems for low-impact validation of correctness and security properties in routing protocols [137]. He was also a primary contributor to the DBToaster project [13, 73], where he led the development of a tool for performing complex realtime analytics over long-lived, rapidly changing data. PI Kennedy will work primarily on dealing with dynamic databases, system efficiency, and any systems issues that arise during prototype development.

PI Nguyen's research interests are in computational and statistical modeling, machine learning and optimization methods for detection with structured data and distributed systems [67, 123, 126–128, 133, 134]. His work on decentralized detection algorithms was awarded the 2007 IEEE Signal Processing Society's **Young author best paper award** [125], and the **ICML's best paper award** [124]. He is also interested in statistical modeling based on hierarchical and graphical models, infinite dimensional function spaces and stochastic processes [121, 122]. PI Nguyen has considerable experience working with real world applications involving texts, images, electronic signals and systems. He will work on the statistical modeling and algorithms that incorporate the structures existing in databases and threat models.

Consultants Eric Ayotte and Manish Gupta from M&T bank have expertise in intrusion detection and analysis and have been working in this area for the past 15 years. They will work with the investigators of the project and supervise students working on the system evaluation, provide data for testing the software, and attend meetings as required for the progress of the project. Eric Ayotte is currently the Vice President and Manager of Network Security at M&T Bank. He and his team will assist in conducting penetration testing and red-teaming exercises, and provide the logistics for evaluation and prototype development.

Management and cross-discipline scientific integration. This project integrates ideas from statistics, machine learning, algorithms, databases and security to address the problem of insider threats in databases. It brings together computer science, engineering, statistics and information science disciplines. Three of the four investigators are with the computer science and engineering department at SUNY Buffalo and the fourth investigator is with the statistics department at University of Michigan at Ann Arbor. The two consultants are from M&T Bank, where Eric is currently the Vice President and Manager of Network Security.

This project plans to support three PhD students at Buffalo and one PhD student at University of Michigan. Each investigator will work closely with a student to address the various components of the research and will meet in a group at least twice a month or as needed. Because of the tight interaction needed among statistics, machine learning, algorithms, databases and security, frequent cross-institution meetings will be necessary between Buffalo and University of Michigan, and occasionally with M&T team. These meetings will be held mostly through Skype conferences. We have also included budget to facilitate team members' travel between M&T site, SUNY at Buffalo and University of Michigan about twice a year.

As M&T headquarter is located in Buffalo we anticipate many physical meetings with the consultants on and off SUNY at Buffalo campus. Several such meetings have already been held in 2012 when we were writing this proposal. Considering the small size of the research team, yearly workshops are not deemed necessary, however, student exchange will be necessary across institutions. PI Nguyen and his students will closely work on improving the feature extraction and modeling parts, and will need to examine the data at M&T site thoroughly. We have allocated sufficient travel funds in this project, a part of which will be used toward student travel as needed. A common shared space on the Internet (such as Dropbox, Google Docs, Assembla) will be created for efficient exchange of project related documents and files. A modest amount is allocated for consultant services in order to compensate for the extra time of the two consultants of this project.

Work plan and milestones. We will roughly follow the order of the four thrusts presented in the Introduction section. In particular, year $1/2/3$ is planned for solving problems pertaining to Thrust $1/2/3$. Problems in Thrust 4 concerning prototype development, evaluation, and system efficiency issues are done throughout the years. The final year is for integration.

Year 1. We will start with designing threat-specific soft classifiers using (nested) hierarchical and mixture models. The fact that hierarchical and mixture models are much more expressive in capturing different organizational roles (thus with different data access patterns) that a single employee might hold at M&T is expected to significantly improve the false positive rates of the preliminary models in [94]. We will evaluate the baseline models using a relatively static database, which is a HRM (Human Resources) database at M&T bank (DB1 in our description). The main threat models used for evaluation are data harvesting, role masquerading, and privilege escalation which we will define and develop extending the outline shown in our prior work [94].

In this phase of the project, we will also gather and pre-process data from the HRM database and the CRM (Cash-Management Wire Transfer) database (DB2 in our description), and anonymize the data (yet retain the statistical patterns) so that students can also work offsite with the data. We will build a prototype software component that allows for our insider threat detection engine to interact with the databases.

Year 2. In order to improve the baseline models, and in particular to address the tension between false positive and false negative rates, we will integrate domain knowledge into our models in the second year. Year 2 will be spent on developing the threat models, algorithms, and the system prototype and its seamless integration with the tool(s) at M&T bank, and conducting experiments for a more realistic validation of our models and algorithms. In particular, we will learn from the dozen or so security audit tools already available at M&T to build our own GUI for the software prototype. It is of crucial importance that we work with M&T security team to build a taxonomy of threat models including known attacks, red-team attacks, and speculative future attacks. The threat models and penetration tests are then used to evaluate our system, and at the same time used to tune our models' parameters. Year 2 will also be spent on carving a deeper understanding of the nature of the data; in particular, much more mathematically involved statistical modeling of the problem will be performed.

Year 3. More detailed experiments will be conducted in this phase to address stealthy attacks and collaborative attacks. The target database, the CRM (Cash Management – Wire) database, will be very dynamic in nature. A new set of feature extraction algorithms and data-dynamic models are to be designed and evaluated. We expect that the statistical models and lessons learned from Year 1 and Year 2 will be integral in addressing the dynamic-database case. The results obtained from this task will be used to refine our threat models and algorithms so that we can develop a

well-tested and validated prototype for possible field deployment.

Year 4. Finally, the last year will be spent on tuning all models, elaborate red-teaming exercises, integrate tightly the prototype software and the statistical models. A summary of the various research tasks and the project milestones is given in Figure 1.

Project Activities & Deliverables	**Inst.**	**2013**		**2014**		**2015**		**2016**		**2017**
Thrust 1. Statistical Modeling		-	9-12	1-6	7-12	1-6	7-12	1-6	7-12	1-8
Feature extraction, Machine learning	BM									
Models, Algorithms, prototyping	BM									
Optimization, refinement	All									
Thrust 2. Domain Knowledge Integration										
Simple attacks	BF									
Advanced threats	BMF									
Integration, Refinement	BMF									
Thrust 3. Dynamic Databases										
Temporal actions	B									
Contextual features	B									
Optimization, refinements	BF									
Thrust 4. Evaluation										
Collection and compilation of data	All									
Experiments, prototyping	All									
Model training, re-building	All									
Persistent Activities										
Publish progress (Website)	B									
Publish Progress (Conference)	BM									
Publish Results (Journal)	BM									
Knowledge to Classrooms	BM									
Involve Students	All									
Institutions: B: *Buffalo,* M: *Michigan,* F: *Financial institution;* **All:** *All institutions*										

Figure 1: Work Plan and Project Milestones

E. References Cited

[1] Insider attack and cyber security. http://www.cs.dartmouth.edu/ insider/program.shtml, 2007.

[2] 1st international workshop on managing insider security threats (MIST 2009). http://isyou.hosting.paran.com/mist09/, 2009.

[3] *Detection and Prevention of Insider Threats in Database Driven Web Services* (West Lafayette, IN, 06/2009 2009).

[4] 2010 ACM CCS workshop on insider threats. http://www.csiir.ornl.gov/ccsw2010/, 2010.

[5] 2010 CAE workshop on insider threat. http://www.dean.usma.edu/caewit/, 2010.

[6] 2nd international workshop on managing insider security threats (MIST 2010). http://isyou.hosting.paran.com/mist10/, 2010.

[7] 3rd international workshop on managing insider security threats (MIST 2011). http://isyou.hosting.paran.com/mist11/, 2011.

[8] Insider threat workshop, software engineering institute, carnegie mellon. http://www.sei.cmu.edu/training/p76.cfm, 2011.

[9] ABE, N., ZADROZNY, B., AND LANGFORD, J. Outlier detection by active learning. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2006), ACM Press, pp. 504–509.

[10] AGOSTA, J. M., DIUK-WASSER, C., CHANDRASHEKAR, J., AND LIVADAS, C. An adaptive anomaly detector for worm detection. In *Proceedings of the 2nd USENIX workshop on Tackling computer systems problems with machine learning techniques* (Berkeley, CA, USA, 2007), USENIX Association, pp. 3:1–3:6.

[11] AGRAWAL, R., BAYARDO, R., FALOUTSOS, C., KIERNAN, J., RANTZAU, R., AND SRIKANT, R. Auditing compliance with a hippocratic database. In *vldb2004* (2004).

[12] AGRAWAL, R., EVFIMIEVSKI, A., KIERNAN, J., AND VELU, R. Auditing disclosure by relevance ranking. In *SIGMOD* (New York, New York, USA, 2007), ACM Press, p. 79.

[13] AHMAD, Y., KENNEDY, O., AND KOCH, C. DBToaster: Higher-order Delta Processing for Dynamic, Frequently Fresh Views. *PVLDB* (2012).

[14] AXELSSON, S. The base-rate fallacy and the difficulty of intrusion detection. *ACM Trans. Inf. Syst. Secur. 3* (August 2000), 186–205.

[15] BAILEY, M., COOKE, E., JAHANIAN, F., XU, Y., AND KARIR, M. A survey of botnet technology and defenses. *Conference For Homeland Security, Cybersecurity Applications & Technology 0* (2009), 299–304.

[16] BARNETT, V., AND LEWIS, T. *Outliers in statistical data*. John Wiley and sons, 1994.

[17] BARRENO, M., BARTLETT, P. L., CHI, F. J., JOSEPH, A. D., NELSON, B., RUBINSTEIN, B. I. P., SAINI, U., AND TYGAR, J. D. Open problems in the security of learning. In *AISec* (2008), pp. 19–26.

[18] BARRENO, M., NELSON, B., SEARS, R., JOSEPH, A. D., AND TYGAR, J. D. Can machine learning be secure? In *ASIACCS* (2006), pp. 16–25.

[19] BOUGANIM, L., AND PUCHERAL, P. Chip-secured data access: confidential data on untrusted servers. In *PVLDB* (Aug. 2002), VLDB Endowment, pp. 131–142.

[20] BRACKNEY, R., AND ANDERSON, R. *Understanding the Insider Threat: Proceedings of a March 2004 Workshop*. RAND Corp, 2004.

[21] BRAUSE, R., LANGSDORF, T., AND HEPP, M. Neural data mining for credit card fraud detection. In *ICTAI '99: Proceedings of the 11th IEEE International Conference on Tools with Artificial Intelligence* (Washington, DC, USA, 1999), IEEE Computer Society, p. 103.

[22] CABRERA, JO A. B. D., LEWIS, L., AND MEHRA, R. K. Detection and classification of intrusions and faults using sequences of system calls. *SIGMOD Rec. 30*, 4 (2001), 25–34.

[23] CALVANESE, D., GIACOMO, G. D., AND LENZERINI, M. On the decidability of query containment under constraints. In *Proc. of the ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS '98)* (1998), pp. 149–158.

[24] CAPPELLI, D. Preventing insider sabotage: Lessons learned from actual attacks. `http://www.cert.org/archive/pdf/InsiderThreatCSI.pdf`, Nov 2005.

[25] CHAGARLAMUDI, M., PANDA, B., AND HU, Y. Insider threat in database systems: Preventing malicious users. In *Sixth International Conference on Information Technology: New Generations* (Las Vegas, NV, 2009), pp. 1616–1620.

[26] CHANDOLA, V., BANERJEE, A., AND KUMAR, V. Anomaly detection: A survey. *ACM Comput. Surv. 41*, 3 (2009), 1–58.

[27] CHANDRASEKARAN, M., BAIG, M., AND UPADHYAYA, S. AEGIS: A proactive methodology to shield against zero-day exploits. *Advanced Information Networking and Applications Workshops, International Conference on 2* (2007), 564–569.

[28] CHANDRASEKARAN, M., SANKARANARAYANAN, V., AND UPADHYAYA, S. J. Inferring sources of leaks in document management systems. In *IFIP Int. Conf. Digital Forensics'08* (2008), pp. 291–306.

[29] CHANDRASEKARAN, M., VIDYARAMAN, V., AND UPADHYAYA, S. J. SpyCon: Emulating user activities to detect evasive spyware. In *IPCCC'07* (2007), pp. 502–509.

[30] CHEBROLU, S., ABRAHAM, A., AND THOMAS, J. Feature deduction and ensemble design of intrusion detection systems. *Computers & Security 24*, 4 (2005), 295–307.

[31] CHINCHANI, R., HA, D., IYER, A., NGO, H. Q., AND UPADHYAYA, S. Insider threat assessment: model, analysis, and tool. In *Network Security*. Springer, New York, 2010.

[32] CHINCHANI, R., IYER, A., NGO, H. Q., AND UPADHYAYA, S. Towards a theory of insider threat assessment. In *Proceedings of the International Conference on Dependable Systems and Networks (DSN)* (Yokohama, Japan, 2005), IEEE, pp. 108–117.

[33] CHINCHANI, R., MUTHUKRISHNAN, A., CHANDRASEKARAN, M., AND UPADHYAYA, S. RACOON: Rapidly generating user command data for anomaly detection from customizable templates. In *20th Annual Computer Security Applications Conference (ACSAC 2004)* (2004), pp. 189–204.

[34] CHUNG, C. Y., GERTZ, M., AND LEVITT, K. DEMIDS: a misuse detection system for database systems. In *Integrity and Internal Control Information Systems: Strategic Views on the Need for Control*. Kluwer Academic Publishers, Norwell, MA, 2000, pp. 159–178.

[35] COMANICIU, D., AND MEER, P. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell. 24*, 5 (2002), 603–619.

[36] CSO MAGAZINE, U.S. SECRET SERVICE, CERT, AND MICROSOFT. 2010 E-Crime Watch Survey. `http://www.cert.org/archive/pdf/ecrimesummary10.pdf`, 2010.

[37] CUPPENS, F. Managing alerts in a multi-intrusion detection environment. In *Proceedings of the 17th Annual Computer Security Applications Conference* (2001), vol. 32.

[38] DASGUPTA, D., AND MAJUMDAR, N. Anomaly detection in multidimensional data using negative selection algorithm. In *Proceedings of the IEEE Conference on Evolutionary Computation* (Hawaii, may 2002), pp. 1039–1044.

[39] DASGUPTA, D., AND NINO, F. A comparison of negative and positive selection algorithms in novel pattern detection. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics* (Nashville, TN, 2000), vol. 1, pp. 125–130.

[40] DAUMÉ, H., AND MARCU, D. Learning as search optimization: approximate large margin methods for structured prediction. In *Proceedings of the ICML* (2005).

[41] DENNING, D. E. An intrusion detection model. *IEEE Transactions of Software Engineering 13*, 2 (1987), 222–232.

[42] DONOHO, S. Early detection of insider trading in option markets. In *KDD '04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining* (New York, NY, USA, 2004), ACM, pp. 420–429.

[43] DU, D.-Z., AND NGO, H. Q., Eds. *Switching Networks: Recent Advances*. Network Theory and Applications, 5. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2001.

[44] ELMASRI, R., AND NAVATHE, S. *Fundamentals of Database Systems*. Addison-Wesley, Reading, MA, 2006.

[45] FABBRI, D., LEFEVRE, K., AND ZHU, Q. PolicyReplay: misconfiguration-response queries for data breach reporting. *Proceedings of the VLDB Endowment 3*, 1-2 (Sept. 2010), 36–47.

[46] FONSECA, J., VIEIRA, M., AND MADEIRA, H. Online detection of malicious data access using dbms auditing. In *Proc. of the 2008 ACM symposium on Applied Computing (SAC'08)* (2008), pp. 1013–1020.

[47] FORREST, S., HOFMEYR, S. A., SOMAYAJI, A., AND LONGSTAFF, T. A. A sense of self for unix processes. In *Proceedinges of the ISRSP96* (1996), pp. 120–128.

[48] GARCIA-MOLINA, H., ULLMAN, J. D., AND WIDOM, J. *Database Systems: The Complete Book*, 2 ed. Prentice Hall Press, Upper Saddle River, NJ, USA, 2008.

[49] GE, T., AND ZDONIK, S. Answering aggregation queries in a secure system model. In *PVLDB* (Sept. 2007), VLDB Endowment, pp. 519–530.

[50] GHOSH, A. K., SCHWARTZBARD, A., AND SCHATZ, M. Learning program behavior profiles for intrusion detection. In *Proceedings of the 1st conference on Workshop on Intrusion Detection and Network Monitoring - Volume 1* (Berkeley, CA, USA, 1999), USENIX Association, pp. 6–6.

[51] GHOSH, S., AND REILLY, D. L. Credit card fraud detection with a neural-network. In *Proceedings of the 27th Annual Hawaii International Conference on System Science* (Los Alamitos, CA, 1994), vol. 3.

[52] GU, G., PERDISCI, R., ZHANG, J., AND LEE, W. Botminer: clustering analysis of network traffic for protocol- and structure-independent botnet detection. In *Proceedings of the 17th conference on Security symposium* (Berkeley, CA, USA, 2008), USENIX Association, pp. 139–154.

[53] HA, D., UPADHYAYA, S., NGO, H. Q., PRAMANIK, S., CHINCHANI, R., AND MATHEW, S. Insider threat analysis using information-centric modeling. In *Advances in Digital Forensics III*, P. Craiger and S. Shenoi, Eds. Springer, Boston, 2007.

[54] HA, D. T., AND NGO, H. Q. On the trade-off between speed and resiliency of Flash worms and similar malcodes. In *Proceedings of The 5th ACM Workshop on Recurring Malcode (WORM 2007), in association with the 14th ACM Conference on Computer and Communications Security (CCS 2007)* (Oct 29–Nov 02 2007), ACM.

[55] HA, D. T., AND NGO, H. Q. On the trade-off between speed and resiliency of flash worms and similar malcodes. *Journal in Computer Virology 5*, 4 (2009), 309–320.

[56] HA, D. T., NGO, H. Q., AND CHANDRASEKARAN, M. Crestbot: A new family of resilient botnets. In *GLOBECOM* (2008), pp. 2148–2153.

[57] HA, D. T., UPADHYAYA, S. J., NGO, H. Q., PRAMANIK, S., CHINCHANI, R., AND MATHEW, S. Insider threat analysis using information-centric modeling. In *IFIP Int. Conf. Digital Forensics* (2007), pp. 55–73.

[58] HA, D. T., YAN, G., EIDENBENZ, S., AND NGO, H. Q. On the effectiveness of structural detection and defense against p2p-based botnets. In *DSN* (2009), pp. 297–306.

[59] HACIGÜMÜŞ, H., IYER, B., LI, C., AND MEHROTRA, S. Executing SQL over encrypted data in the database-service-provider model. In *SIGMOD '02* (New York, New York, USA, 2002), ACM Press, p. 216.

[60] HAINES, J. W., RYDER, D. K., TINNEL, L., AND TAYLOR, S. Validation of sensor alert correlators. *IEEE Security & Privacy 1*, 1 (2003), 46–56.

[61] HERLANDS, A. Arrest the threat: Monitoring privileged database users. In *White Paper* (2007), Applications Security, Inc.

[62] HJORT, N., HOLMES, C., MUELLER, P., AND WALKER, S. *Bayesian Nonparametrics: Principles and Practice*. Cambridge University Press, 2010.

[63] HODGE, V., AND AUSTIN, J. A survey of outlier detection methodologies. *Artif. Intell. Rev. 22*, 2 (2004), 85–126.

[64] HOFMEYR, S. A., FORREST, S., AND SOMAYAJI, A. Intrusion detection using sequences of system calls. *Journal of Computer Security 6*, 3 (1998), 151–180.

[65] HRISTIDIS, V., AND PETROPOULOS, M. Semantic caching of xml databases. In *WebDB* (2002), pp. 25–30.

[66] HU, Y., AND PANDA, B. Identification of malicious transactions in database systems. In *Proc. of the 7th International Database Engineering and Applications Symposium* (2003), pp. 329–335.

[67] HUANG, L., NGUYEN, X., GAROFALAKIS, M., HELLERSTEIN, J., JOSEPTH, A., JORDAN, M. I., AND TAFT, N. Communication-efficient online detection of network-wide anomalies. In *Proc. of 26th IEEE INFOCOM* (May 2007).

[68] INDYK, P., NGO, H. Q., AND RUDRA, A. Efficiently decodable non-adaptive group testing. In *Proceedings of the Twenty First Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'2010)* (New York, 2010), ACM, pp. 1126–1142.

[69] JORDAN, M. Graphical models. *Statistical Science Special Issue on Bayesian Statistics (19)* (2004), 140–155.

[70] KABIRI, P., AND GHORBANI, A. A. Research on intrusion detection and response: A survey. *International Journal of Network Security 1* (2005), 84–102.

[71] KAMRA, A., TERZI, E., AND BERTINO, E. Detecting anomalous access patterns in relational databases. *The VLDB Journal 17*, 5 (2008), 1063–1077.

[72] KENDALL, K. *A database of computer attacks for the evaluation of intrusion detection systems*. PhD thesis, Massachusetts Institute of Technology, 1999.

[73] KENNEDY, O., AHMAD, Y., AND KOCH, C. DBToaster: Agile Views for a Dynamic Data Management System. In *CIDR* (2011), pp. 284–295.

[74] KENNEDY, O., AND KOCH, C. PIP: A database system for great and small expectations. In *ICDE* (2010), pp. 157–168.

[75] KENNEDY, O., KOCH, C., AND DEMERS, A. Dynamic Approaches to In-network Aggregation. In *ICDE* (2009), IEEE, pp. 1331–1334.

[76] KENNEDY, O., LEE, S., LOBOZ, C., AND SMYL, S. Fuzzy prophet: parameter exploration in uncertain enterprise scenarios. In *SIGMOD* (2011).

[77] KENNEDY, O., AND NATH, S. Jigsaw: efficient optimization over uncertain enterprise data. In *SIGMOD* (June 2011), ACM Request Permissions.

[78] KILLOURHY, K. S., AND MAXION, R. A. Comparing anomaly-detection algorithms for keystroke dynamics. In *DSN* (2009), pp. 125–134.

[79] KILLOURHY, K. S., AND MAXION, R. A. Why did my detector do *that*?! - predicting keystroke-dynamics error rates. In *RAID* (2010), pp. 256–276.

[80] KRUEGEL, C., AND VIGNA, G. Anomaly detection of web-based attacks. In *Proc. of the 10th ACM conference on Computers and Communications Security (CCS'03)* (2003), pp. 251–261.

[81] KRÜGEL, C., TOTH, T., AND KIRDA, E. Service specific anomaly detection for network intrusion detection. In *Proceedings of the 2002 ACM symposium on Applied computing* (2002), ACM, pp. 201–208.

[82] LANE, T., AND BRODLEY, C. E. Temporal sequence learning and data reduction for anomaly detection. *ACM Trans. Inf. Syst. Secur. 2* (August 1999), 295–331.

[83] LANE, T., AND BRODLEY, C. E. Temporal sequence learning and data reduction for anomaly detection. *ACM Transactions on Information Systems and Security 2*, 3 (1999), 295–331.

[84] LAURITZEN, S. *Graphical models*. Oxford University Press, 1996.

[85] LAZAREVIC, A., ERTOZ, L., KUMAR, V., OZGUR, A., AND SRIVASTAVA, J. A comparative study of anomaly detection schemes in network intrusion detection. In *Proceedings of SIAM International Conference on Data Mining* (May 2003), SIAM.

[86] LEE, S. Y., LOW, W. L., AND WONG, P. Y. Learning fingerprints for a database intrusion detection system. In *Proc. of the 7th European Symposium on Research in Computer Security (ESORICS'02)* (2002), pp. 264–280.

[87] LEE, V. C., STANKOVIC, J., AND SON, S. H. Intrusion detection in real-time database systems via time signatures. In *Proc. of the Sixth IEEE Real Time Technology and Applications Symposium (RTAS'00)* (2000), p. 124.

[88] LEE, W., AND STOLFO, S. J. Data mining approaches for intrusion detection. In *Proceedings of the 7th conference on USENIX Security Symposium - Volume 7* (Berkeley, CA, USA, 1998), USENIX Association, pp. 6–6.

[89] LIU, A., MARTIN, C., HETHERINGTON, T., AND MATZNER, S. Ai lessons learned from experiments in insider threat detection. In *Proc. AAAI Spring Symp.* (2006), pp. 49–55.

[90] LIU, P. Architectures for intrusion tolerant database systems. In *Proc. of the 18th Annual Computer Security Applications Conference (ACSAC '02)* (2002), p. 311.

[91] LIU, T. Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval 3*, 3 (2009), 225–331.

[92] LIU, Y., ZHANG, H. H., AND WU, Y. Hard or soft classification? large-margin unified machines. *Journal of the American Statistical Association 106*, 493 (2011), 166–177.

[93] MAIER, D., ULLMAN, J. D., AND VARDI, M. Y. On the foundations of the universal relation model. *ACM Trans. on Database Syst. 9*, 2 (1984), 283–308.

[94] MATHEW, S., PETROPOULOS, M., NGO, H. Q., AND UPADHYAYA, S. J. A data-centric approach to insider attack detection in database systems. In *RAID* (2010), pp. 382–401.

[95] MIKLASZEWSKI, J., AND KUBE, C. Manning faces new charges, possible death penalty. *MSNBC* (2011).

[96] MIKLAU, G., AND SUCIU, D. A formal analysis of information disclosure in data exchange. *Journal of Computer and System Sciences 73*, 3 (May 2007), 507–534.

[97] MOTWANI, R., NABAR, S. U., AND THOMAS, D. Auditing SQL Queries. In *ICDE* (2008), IEEE, pp. 287–296.

[98] MYKLETUN, E., AND TSUDIK, G. Aggregation Queries in the Database-As-a-Service Model. D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, E. Damiani, and P. Liu, Eds., vol. 4127, Springer Berlin Heidelberg.

[99] NELLIKAR, S., NICOL, D., AND CHOI, J. Role-based differentiation for insider detection algorithms. In *Insider Threats Workshop* (Chicago, IL, 2010), ACM, pp. 55–62.

[100] NGO, H. Q. A new routing algorithm for multirate rearrangeable Clos networks. *Theoret. Comput. Sci. 290*, 3 (2003), 2157–2167.

[101] NGO, H. Q. WDM switching networks, rearrangeable and nonblocking $[w, f]$-connectors. *SIAM Journal on Computing 35*, 3 (2005-2006), 766–785.

[102] NGO, H. Q. WDM switching networks: complexity and constructions. In *Combinatorial Optimization in Communication Networks*, D.-Z. Du, M. Cheng, and Y. Li, Eds., vol. 18 of *Combinatorial Optimization*. Springer, New York, 2006, pp. 395–426.

[103] NGO, H. Q. On a hyperplane arrangement problem and tighter analysis of an error-tolerant pooling design. *J. Comb. Optim. 15*, 1 (2008), 61–76.

[104] NGO, H. Q., AND DU, D.-Z. A survey on combinatorial group testing algorithms with applications to DNA library screening. In *Discrete mathematical problems with medical applications (New Brunswick, NJ, 1999)*, vol. 55 of *DIMACS Ser. Discrete Math. Theoret. Comput. Sci.* Amer. Math. Soc., Providence, RI, 2000, pp. 171–182.

[105] NGO, H. Q., AND DU, D.-Z. Notes on the complexity of switching networks. In *Advances in Switching Networks*, D.-Z. Du and H. Q. Ngo, Eds., vol. 5 of *Network Theory and Applications*. Kluwer Academic Publishers, 2001, pp. 307–367.

[106] NGO, H. Q., AND DU, D.-Z. New constructions of non-adaptive and error-tolerance pooling designs. *Discrete Math. 243*, 1-3 (2002), 161–170.

[107] NGO, H. Q., NGUYEN, T.-N., AND HA, D. T. Crosstalk-free widesense nonblocking multicast photonic switching networks. In *Proceedings of the 2008 IEEE Global Communications Conference (GLOBECOM)* (New Orleans, LA, U.S.A., 2008), IEEE, pp. ??–??

[108] NGO, H. Q., PAN, D., AND QIAO, C. Nonblocking WDM switches based on arrayed waveguide grating and limited wavelength conversion. In *Proceedings of the 23rd Conference of the IEEE Communications Society (INFOCOM)* (Hong Kong, China, 2004), IEEE.

[109] NGO, H. Q., PAN, D., AND QIAO, C. Constructions and analyses of nonblocking WDM switches based on arrayed waveguide grating and limited wavelength conversion. *IEEE/ACM Transactions on Networking 14*, 1 (2006), 205–217.

[110] NGO, H. Q., PAN, D., AND YANG, Y. Optical switching networks with minimum number of limited range wavelength converters. In *Proceedings of the 24rd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)* (Miami, Florida, U.S.A., March 2005), vol. 2, IEEE, pp. 1128–1138.

[111] NGO, H. Q., PAN, D., AND YANG, Y. Optical switching networks with minimum number of limited range wavelength converters. *IEEE/ACM Transactions on Networking 15*, 4 (2007), 969–979.

[112] NGO, H. Q., PORAT, E., RÉ, C., AND RUDRA, A. Worst-case optimal join algorithms. In *Proceedings of the 31st ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS)* (2012), pp. 37–48.

[113] NGO, H. Q., PORAT, E., AND RUDRA, A. Efficiently decodable error-correcting list disjunct matrices and applications - (extended abstract). In *ICALP (1)* (2011), pp. 557–568.

[114] NGO, H. Q., PORAT, E., AND RUDRA, A. Efficiently decodable compressed sensing by list-recoverable codes and recursion. In *Symposium on Theoretical Aspects of Computer Science (STACS)* (2012). to appear.

[115] NGO, H. Q., RUDRA, A., LE, A. N., AND NGUYEN, T.-N. Analyzing nonblocking switching networks using linear programming (duality). In *INFOCOM* (2010), pp. 2696–2704.

[116] NGO, H. Q., AND VU, V. H. Multirate rearrangeable Clos networks and a generalized bipartite graph edge coloring problem. *SIAM Journal on Computing 32*, 4 (2003), 1040–1049.

[117] NGO, H. Q., AND VU, V. H. Multirate rearrangeable Clos networks and a generalized bipartite graph edge coloring problem. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'2003, Baltimore, MD)* (New York, 2003), ACM, pp. 834–840.

[118] NGO, H. Q., WANG, Y., AND LE, A. A linear programming duality approach to analyzing strictly nonblocking d-ary multilog networks under general crosstalk constraints. In *Proceedings of the 14th Annual International Computing and Combinatorics Conference (COCOON)* (Bejing, China, 2008), Springer, LNCS, pp. 509–519.

[119] NGO, H. Q., WANG, Y., AND PAN, D. Rearrangeable and nonblocking $[w, f]$-distributors. *IEEE/ACM Transactions on Networking* (2008). Accepted for publication.

[120] NGUYEN, T.-N., NGO, H. Q., AND WANG, Y. Strictly nonblocking f-cast photonic switching networks under general crosstalk constraints. In *Proceedings of the 2008 IEEE Global Communications Conference (GLOBECOM)* (New Orleans, LA, U.S.A., 2008), IEEE, pp. ??–??

[121] NGUYEN, X. Inference of global clusters from locally distributed data. *Bayesian Analysis 5*, 4 (2010), 817–846.

[122] NGUYEN, X., AND GELFAND, A. The Dirichlet labeling process for clustering functional data. *Statistica Sinica 21*, 3 (2011), 1249–1289.

[123] NGUYEN, X., JORDAN, M. I., AND SINOPOLI, B. A kernel-based learning approach to ad hoc sensor network localization. *ACM Transactions on Sensor Networks 1* (2005), 134–152.

[124] NGUYEN, X., WAINWRIGHT, M. J., AND JORDAN, M. I. Decentralized detection and classification using kernel methods. In *International Conference on Machine Learning* (2004).

[125] NGUYEN, X., WAINWRIGHT, M. J., AND JORDAN, M. I. Nonparametric decentralized detection using kernel methods. *IEEE Transactions on Signal Processing 53*, 11 (2005), 4053–4066.

[126] NGUYEN, X., WAINWRIGHT, M. J., AND JORDAN, M. I. On optimal quantization rules in some problems in sequential decentralized detection. *IEEE Transactions on Information Theory 54(7)* (2008), 3285–3295.

[127] NGUYEN, X., WAINWRIGHT, M. J., AND JORDAN, M. I. On surrogate losses and f-divergences. *Annals of Statistics 37(2)* (2009), 876–904.

[128] NGUYEN, X., WAINWRIGHT, M. J., AND JORDAN, M. I. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory 56(11)* (2010), 5847–5861.

[129] PATCHA, A., AND PARK, J.-M. An overview of anomaly detection techniques: Existing solutions and latest technological trends. *Comput. Netw. 51*, 12 (2007), 3448–3470.

[130] PORTNOY, L., ESKIN, E., AND STOLFO, S. Intrusion detection with unlabeled data using clustering. In *In Proceedings of ACM CSS Workshop on Data Mining Applied to Security (DMSA-2001* (2001), Citeseer.

[131] PRAMANIK, S., SANKARNARAYANAN, V., AND UPADHYAYA, S. Security policies to mitigate insider threat in the document control domain. In *20th Annual Computer Security Applications Conference (ACSAC 2004)* (2004).

[132] RAGAVAN, H., AND PANDA, B. Mitigation of malicious modifications by insiders in databases. In *ICISS* (2011), pp. 337–351.

[133] RAJAGOPAL, R., NGUYEN, X., ERGEN, S., AND VARAIYA, P. Distributed online simultaneous fault detection for multiple sensors. In *Proceedings of the 7th International Conference on Information Processing in Sensor Networks (IPSN)* (April 2008).

[134] RAJAGOPAL, R., NGUYEN, X., ERGEN, S., AND VARAIYA, P. Simultaneous sequential detection of multiple interacting faults. *http://arxiv.org/abs/1012.1258* (2010).

[135] RAMASUBRAMANIAN, P., AND KANNAN, A. Intelligent multi-agent based database hybrid intrusion prevention system. In *Proc. of the 8th East European Conference (ADBIS '04)* (2004).

[136] REEDER, R. W., AND MAXION, R. A. User interface defect detection by hesitation analysis. In *DSN* (2006), pp. 61–72.

[137] REYNOLDS, P., KENNEDY, O., AND SIRER, E. Securing BGP using external security monitors. *arXiv* (2006).

[138] ROBERT, C. *The Bayesian choice*. Springer, 2001.

[139] ROICHMAN, A., AND GUDES, E. Diweda – detecting intrusions in web databases. In *Proc. of the 22nd annual IFIP WG 11.3 working conference on Data and Applications Security* (2008), pp. 313–329.

[140] ROUSSEEUW, P. J., AND LEROY, A. M. *Robust regression and outlier detection*. John Wiley & Sons, Inc., New York, NY, USA, 1987.

[141] SANKARANARAYANAN, V., UPADHYAYA, S. J., AND KWIAT, K. A. QoS-T: QoS throttling to elicit user cooperation in computer systems. In *MMM-ACNS'10* (2010), pp. 102–117.

[142] SCHNEIER, B. *Secrets and Lies: Digital Security in a Networked World.* John Wiley and Sons, New York, NY, 2000.

[143] SCHÖLKOPF, B., AND SMOLA, A. *Learning with Kernels*. MIT Press, Cambridge, MA, 2002.

[144] SCHÖLKOPF, B., SMOLA, A. J., WILLIAMSON, R. C., AND BARTLETT, P. L. New support vector algorithms. *Neural Computation 12*, 5 (2000), 1207–1245.

[145] SCHONLAU, M., DUMOUCHEL, W., JU, W., KARR, A., THEUS, M., AND VARDI, Y. Computer intrusion: Detecting masquerades. *Statistical Science 16*, 1 (2001), 58–74.

[146] SEKAR, R., GUANG, Y., VERMA, S., AND SHANBHAG, T. A high-performance network intrusion detection system. In *Proceedings of the 6th ACM conference on Computer and communications security* (1999), ACM Press, pp. 8–17.

[147] SEKAR, R., GUPTA, A., FRULLO, J., SHANBHAG, T., TIWARI, A., YANG, H., AND ZHOU, S. Specification-based anomaly detection: a new approach for detecting network intrusions. In *Proceedings of the 9th ACM conference on Computer and communications security* (2002), ACM Press, pp. 265–274.

[148] SHATNAWI, N., ALTHEBYAN, Q., AND MARDINI, W. Detection of insiders misuse in database systems. In *Proceedings of the International Multiconference of Engineers and Computer Scientists* (2011).

[149] SNYDER, D. Online intrusion detection using sequences of system calls. Master's thesis, Department of Computer Science, Florida State University, 2001.

[150] SONG, D. X., WAGNER, D., AND PERRIG, A. Practical techniques for searches on encrypted data. In *SECPRI-00* (2000), IEEE Comput. Soc, pp. 44–55.

[151] SOULE, A., SALAMATIAN, K., AND TAFT, N. Combining filtering and statistical methods for anomaly detection. In *IMC '05: Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement* (Berkeley, CA, USA, 2005), USENIX Association, pp. 31–31.

[152] SPALKA, A., AND LEHNHARDT, J. A comprehensive approach to anomaly detection in relational databases. In *DBSec* (2005), pp. 207–221.

[153] SRIVASTAVA, A., SURAL, S., AND MAJUMDAR, A. K. Database intrusion detection using weighted sequence mining. *Journal of Computers 1*, 4 (2006), 8–17.

[154] STEINWART, I., HUSH, D., AND SCOVEL, C. A classification framework for anomaly detection. *Journal of Machine Learning Research 6* (2005), 211–232.

[155] STEWART YANG AND, J. S., RAJAMANI, H., CHO, T., ZHANG, Y., AND MOONEY, R. Fast and effective worm fingerprinting via machine learning. In *Proceedings of the 3rd IEEE International Conference on Autonomic Computing (ICAC-2006)* (Dublin, Ireland, June 2006). Poster Session.

[156] TASKAR, B., CHATALBASHEV, V., KOLLER, D., AND GUESTRIN, C. Learning structured prediction models: a large margin approach. In *Proceedings of the ICML* (2005).

[157] TAX, D. M. J., AND DUIN, R. P. W. Support vector data description. *Mach. Learn. 54* (January 2004), 45–66.

[158] TEH, Y., JORDAN, M., BEAL, M., AND BLEI, D. Hierarchical Dirichlet processes. *J. Amer. Statist. Assoc. 101* (2006), 1566–1581.

[159] THEILER, J., AND CAI, D. M. Resampling approach for anomaly detection in multispectral images. In *Proceedings of SPIE 5093* (2003), 230-240, Ed.

[160] UPADHYAYA, S., KWIAT, K., CHINCHANI, R., AND MANTHA, K. Encapsulation of owner's intent a new proactive intrusion assessment paradigm. In *Managing Cyber Threats: Issues, Approaches and Challenges*, V. Kumar, J. Srivastava, and A. Lazarevic, Eds. Springer, 2005.

[161] U.S. SECRET SERVICE, AND CERT. The Insider Threat Study. `http://www.cert.org/insider_threat/study.html`, 2008.

[162] VALEUR, F., MUTZ, D., AND VIGNA, G. A learning-based approach to the detection of sql attacks. In *Proc. of the Conference on Detection of Intrusions and Malware & Vulnerability Assessment (DIMVA '05)* (2005), pp. 123–140.

[163] WAGNER, D., AND DEAN, R. Intrusion detection via static analysis. In *Security and Privacy, 2001. S&P 2001. Proceedings. 2001 IEEE Symposium on* (2001), IEEE, pp. 156–168.

[164] WANG, Q., KULKARNI, S. R., AND VERDÚ, S. Divergence estimation of continuous distributions based on data-dependent partitions. *IEEE Transactions on Information Theory 51*, 9 (2005), 3064–3074.

[165] WANG, Q., YU, T., LI, N., LOBO, J., BERTINO, E., IRWIN, K., AND BYUN, J.-W. On the correctness criteria of fine-grained access control in relational databases. VLDB Endowment.

[166] WANG, Y., NGO, H. Q., AND JIANG, X. Strictly nonblocking f-cast d-ary multilog networks under fanout and crosstalk constraints. In *Proceedings of the 2008 International Conference on Communications (ICC)* (Bejing, China, 2008), IEEE.

[167] WANG, Y., NGO, H. Q., AND NGUYEN, T.-N. Constructions of given-depth and optimal multirate rearrangeably nonblocking distributors. In *Proceedings of the 2007 Workshop on High Performance Switching and Routing (HPSR)* (2007), IEEE.

[168] WENHUI, S., AND TAN, D. A novel intrusion detection system model for securing web-based database systems. In *Proc. of the 25th International Computer Software and Applications Conference on Invigorating Software Development (COMPSAC '01)* (2001), p. 249.

[169] WHITTAKER, J. *Graphical Models in Applied Multivariate Statistics*. Wiley, 2009.

[170] YAO, Q., AN, A., AND HUANG, X. Finding and analyzing database user sessions. In *Proc. of Database Systems for Advanced Applications* (2005), pp. 283–308.

[171] ZHANG, N., YU, W., FU, X., AND DAS, S. K. Maintaining defender's reputation in anomaly detection against insider attacks. *IEEE Transactions on Systems, Man, and Cybernetics, Part B 40*, 3 (2010), 597–611.