Anonymity and Severity Analysis for Data Leakeage Detection

Jay Velasco
Fordham University: Cybersecurity

***Abstract*— The number of records that were compromised often measures the severity a data breach. Organizations may monitor for these leakages and provide alerts on certain criteria. Specifying alert criteria can help a company better utilize its time and resources. Providing data models for specific domains to measure the severity of leakages can aid in this process. The severity can be based on many factors and should consider how much an attacker can infer about a subject from the leaked data. This work proposes an approach to evaluate the impact of anonymity on the L-Severity calculation. Often data may come from many sources and is displayed on desktop applications or browsers. An architecture implementation at the application level will be presented to measure severity from multiple containers.** *(Abstract)*

Keywords-Data Leakage Detection, Privacy Enhancing Technology

I. INTRODUCTION

Ponemon Institute conducted a study in 2016 involving 383 companies from 12 different countries. The research found that there has been a 29% increase in in the total cost of a data breach, reaching an average of $4 million. Each record has an average cost of $158. The healthcare industry had the highest cost of $355 per record.

The cost for an organization to be prepared for a breach is a fixed cost. The increase per person that an organization spends on security has gone up 15% since 2013. This cost can be attributed to investments in resources and data leakage prevention technologies. The quicker an organization can deal with a breach will reduce its negative impacts. Stronger data governance, hiring a CISO, having an incident response and business continuity plan can help detect and mitigate data breaches. Data leakages caused by cyber criminals are more expensive and harder to detect than those caused by system or human errors.

The number of records and cost has a positive correlation. However, the severity of what was leaked may vary. For example, one might argue that disclosure of a specific disease can impact their lives more negatively than others if disclosed. Those with expertise within their industry must define the labeling of attributes for a given domain. Vavilis et al. created data models with certain assumptions such as a disease like HIV can have a major impact on the life of a subject if the data was disclosed. The severity of the disease increased as well as its medication to treat it. The medication received a high severity score because it can be used to infer the disease of a subject.

Web Application Programming Interfaces (APIs) are commonly used in web development projects. Web APIs allow a developer to connect to different libraries from a single application. These libraries can come from different sources. Applying privacy and sensitivity metrics at the application level can add an extra layer of abstraction and is more portable. For example, if switching a data source is needed, all the metrics and data models setup will stay the same. This research will evaluate different privacy metrics and its impact on the L-Severity result. At this point, the work will be in the following structure; Section II will review previous work, Section III will perform an analysis of this work's proposal and effort. Section 4 will state conclusions and Section 5 will provide references.

II. PREVIOUS WORK

A. *A Severity-based Quantificaiton of Data Leakages in Database Systems*

Vavilis et al. based their research on the M-Score. The M-Score calculates a severity metric, but has limitation. For example, to calculate the M-Score a Raw Record Score (RRS) is needed. The RRS has a maximum of 1 and the row with the highest RRS is used as the Final Record Score (RS). The RS is then used to derive the M-Score. In order to calculate the RS, there is a Distinguishing Factor (DF) that the RRS is multiplied by. Although not explicitly stated in the paper, DF is set to a constant .5. The M-Score was then calculated against 3 different cases. Case 1.x[1] exposed the min and max limitations and Case 3.x shows although the leaked table in case 3.2 had less records, the diseases overall were more severe. To accommodate this, L-Severity was proposed. L-Severity will aggregate the node sensitivity of each sensitive attribute per row.

$$RRS_r = \min\left(1, \sum_{s_i \in S} f(s_i[r])\right)$$

Raw Record Score: S is the set of sensitive attributes.

$$DF_r^{D(a_1 \ldots\ldots a_n)} = \frac{1}{|R'|}$$

Distinguishing Factor: $|R'|$ Is the number of quasi-attributes within the row. Quasi attributes are attributes that are pre-defined and can be used to identify an entity by linking it to other sources.

[1] X represents cases 1 and 2

$$RS_L = \min_{r \in R^{L(b_1 \ldots b_m)}} RRS_r \times DF_r^{ST(a_1 \ldots\ldots a_n)}$$

Final Record Score: $DF_r^{ST(a_1 \ldots\ldots a_n)}$ Represents the source table.

$$MScore_L = \left| R^{L(a_1,\ldots,a_n)} \right|^{\frac{1}{x}} \times RS_L$$

M-Score: $R^{L(a_1,\ldots,a_n)}$ Represents the number of leaked records. Variable x is defined by an analyst and influences the impact of the number of rows that was leaked.

$$RSENS = DF_r^{ST(a_1 \ldots a_n)} \times \sum_{S_i \in S} NS(S_i[r])$$

Record Sensitivity: NS Represents the Node Sensitivity that is defined in the domain's data model.

$$L - Severity_L = \sum_{r \in R^{L(b_1 \ldots b_n)}} RSENS_r$$

L-Severity: For each leaked row, aggregate the record sensitivity.

Table 1 Score Matrix

Case	L-Severity	M-Score		
		x=1	*x*=10	*x*=100
Case 1.1 Case 1.1	1.700 2.900	2.000 2.000		
Case 2.1 Case 2.2	1.150 2.100	2.000 2.000		
Case 3.1 Case 3.2	1.950 2.900	3.500 2.000	0.607 0.574	0.509 0.507

Table 1 shows the result matrix of M-Score against L-Severity with different values of x. X is only applicable to M-Score. Case 3.1 and 3.2 have different results, L-Severity scores the table in case 3.2 having a higher sensitivity score than what was given in M-Score, .507. Therefore, L-Severity takes account the severity of the entire table and is not limited by the min or max values in M-Score.

B. M-score: A misuseability Weight Measure

The M-Score requires sensitivity functions to be defined by domain experts. M-Score was developed to provide a measurement of misuse. Harel et al. describes four dimensions of what they refer to as misuseability; number of entities, anonymity, number of properties and their values.

III. ANALYSIS

M-Score and L-Severity generalizes the DF. Harel et al. would count the number of records that matched the quasi-identifier. When the distinguishing factor changes the emphasis on anonymity becomes an issue. Those within the healthcare industry may place a higher emphasis on the number of rows and how anonymous their clients are. M-Score's distinguishing factor is a good metric, but others exist such as l-diversity, t-closeness and k-anonymity. M-Score was created to address the threat of an insider attack, but the evaluation of the other privacy metrics in coordination with L-Severity to our knowledge has not been done yet. Wagner et al. provided a survey of over eighty privacy metrics. Privacy metrics can come in many forms such as similarity and diversity and information gain and loss. Information gain and loss determines how much information an attacker can acquire. High information gain is associated with low privacy and vice versa. Similarity and Diversity measures only the privacy of the data itself without considering an attacker. This research will measure the impact of different privacy metrics on the outcome of the result of L-Severity.

Using (Profession, Location, Gender, Disease, Treatment,D_{Score}, $Tscore$, sum(DT)) as the columns of Tables 2 and 3.

Table 2

Case 3.1							
Lawyer	LA	Male	HIV	Vitamins	100	10	55
Lawyer	LA	Male	Flu	Parcetamol	10	10	10
Doctor	NY	Female	Flu	Aspirin	10	30	20
Doctor	NY	Female	Migraine	Aspirin	30	30	30
Teacher	TX	Male	Migraine	Parcetamol	30	10	40
Nurse	NY	Female	H1N1	Aspirin	40	30	35
Nurse	NY	Female	H1N1	Parcetamol	40	10	25
							215
							2.15

Table 3

Case 3.2							
Lawyer	LA	Male	HIV	ARV	100	100	50
Lawyer	LA	Male	Hypertension	Statin	60	60	30
Lawyer	LA	Male	Heart Attack	b-Blocker	70	80	37.5
Lawyer	LA	Male	Migraine	b-Blocker	30	80	27.5
							145
							1.45

Tables 2 and 3 represent case 3.1 and 3.2 applying the L-Severity score with a varying DF. DF has a large impact on the outcome of the results. The results were obtained using profession and location as the quasi-identifier values. See Distinguishing Factor in section 2.A. Analysis into applying different privacy metrics could benefit the performance of the algorithm.

Valvivis et al. performed an experiment analyzing alerts of a Data Leakage Detection (DLD) system. Many alerts appeared and L-Severity was used as a metric to help identify the most critical alerts. The measurement that was used was the False Discovery Rate or FDR. The tool performed the analysis

on queries. Although data retrieval may result in an underlying query, data can come from different sources and in different types. For example, retrieving data from a database and from a file. This data would then be aggregated by the application and displayed in a readable format. The database may have security metrics setup, most legacy systems will not. It may be beneficial and easier on legacy systems to implement the severity measurement at the application layer. This research proposes to create an implementation at the application level that performs these metric calculations from varying sources.

Table 4

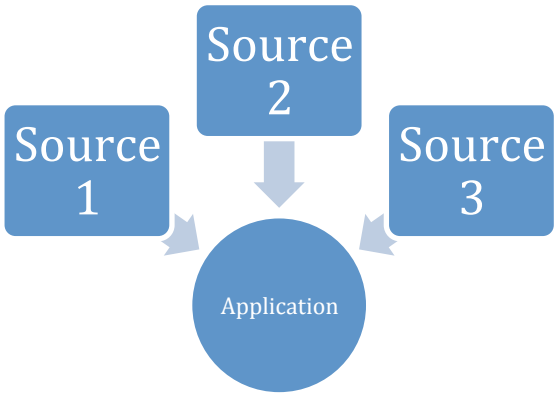

Table 4 shows a single application retrieving data from multiple sources. It can be assumed that database links exist enabling the sources to communicate, but in this example these sources are from 3 different companies. It may not be practical for these companies to create links to each other for one client's application. Source 1 can perform all the metrics that were described in Section 2, but the other data containers may not do the same. Aggregating this data may also be a challenge. Performing the calculations in the application layer can allow the metrics to then be stored in the desired database.

TIMELINE

Table 5 Timeline

	Week 1	Week 2	Writing Phases
Lecture 1	Topic		
Lecture 2	Research Analysis	Research Presentation	Proposal Paper and build slides
Lecture 3	Finalization on privacy metrics	Design application	Analysis on metrics
Lecture 4	Proof of concept of application		Design of application and analysis of metrics
Lecture 5			Design of application
Lecture 6	Finish Paper		
Lecture 7	Finalize paper	Present project	Finalizing/cleaning up paper and build slides

Table 5 represents a timeline of events and goals to complete throughout the course of the semester. Lecture 6's weeks will be the time to finish the paper. Lecture 7's weeks will be reviewing and cleaning up final pieces of the project.

IV. CONCLUSION

The importance of data leakage prevention is relevant in today's media and influences how we use and ingest data on a day-to-day basis. Previous work shows an emphasis on finding a severity metric that takes account of the entire table. However, the result can be impacted by privacy metrics such as the distinguishing factor. Providing security metrics at a database level is beneficial, but having the option to do so at an application level can be more robust. A timeline of events that will take place until the project is completed has been presented. Challenges that are expected is finding, interpreting and attempting to create an improvement based on the privacy metrics that will be used in this research. Designing and creating a proof of concept for the application will be challenging due to time constraints. Future work can focus on applying the proof of concept in an experimental setting or attempting to measure the severity of modifications on sensitive data.

V. REFERENCES

[1] S. Vavilis, M. Petkovic, and N. Zannone, "A severity-based quantification of data leakages in database systems,"Journal of Computer Security,vol. 24, no. 3, pp. 321–345, 2016.

[2] Ponemon Institute. (2016). 2016 Cost of Data Breach Study: Global Analysis. The Ponemon Institute.

[3] Wagner, Isabel, and David Eckhoff. "Technical privacy metrics: a systematic survey." *arXiv preprint arXiv:1512.00327* (2015).

[4] Harel, Amir, et al. "M-score: A misuseability weight measure." *IEEE Transactions on Dependable and Secure Computing* 9.3 (2012): 414-428.