Anonymity and Severity Analysis for Data Leakage Detection

Jay Velasco
Fordham University: Cybersecurity

***Abstract*— Data leakage is the loss of data that may result in the damaging of a company's reputation and productivity. The number of records that were compromised often measures the severity of a data breach. There are many threats that exist that may result in the unauthorized release of information. For example, medical and government information is commonly released in generalized versions. The data publisher may not be aware of other data sets that can be used to perform linkage attacks. Measuring the severity based on privacy and frequency of attributes can help prevent a leakage from occurring. The release of information may be intentional or unintentional. Organizations may monitor for data leakages and provide alerts on certain criteria. Specifying alert criteria can help a company better utilize its time and resources. Providing data models to measure the severity of leakages can aid in the detection process. Severity can be based on many factors and should consider how much an attacker can infer about a subject from the leaked data. We propose KL-Severity to address the scoring of all data classification, incorporating the diversity and distribution of sensitive attributes and an analysis on the effect of different privacy metrics on L-Severity. We determined that our approach of using K-Anonymity does not have large difference on the severity of a record in comparison to the Distinguishing Factor. However, considering a diverse and well-represented set of sensitive attributes can increase the accuracy of measuring severity.** *(Abstract)*

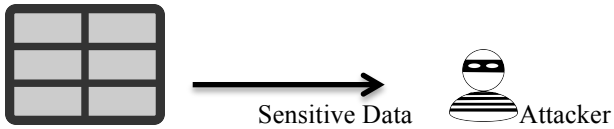
Keywords-Data Leakage Detection, Privacy Enhancing Technology

I. INTRODUCTION

Ponemon Institute performed 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. [6] Each record has an average cost of $158. [6] The healthcare industry had the highest cost of $355 per record. [10] Data is growing at a rapid pace, .5% of all data is analyzed and this amount is decreasing. [13] To address the growing data, technology is changing and more investments in modern data infrastructure are being made. The investments are to improve data analytics, which includes real-time data processing and visualization. As data increases and technology advances, so will the variety and severity of attacks. Privacy and sensitivity of all attributes will contribute to the severity of a data breach. The cost for an organization to be prepared for a breach is fixed. The increase per person that an organization spends on security has gone up 15% since 2013. [10] The cost can be attributed to investments in resources and Data Leakage Prevention (DLP) technologies. The quicker an organization can respond to 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. Leakages caused by cyber criminals are more expensive and harder to detect than those caused by system or human errors. The ability to detect and respond can be increased through measuring severity. Providing an accurate model of severity for sensitive attributes can help determine the allocation of security resources, prevent false positives and avoid intentional and unintentional data leakages. False positives are events that alert the attention of a security professional. He or she must then take time to investigate the event to find out that the security event is not an issue. The investigation incurs costs on the company and redirects the professional from other events that may truly be data breaches.

Regulations have been increased to protect the confidentiality of users. The European Union Agency for Network and Information Security (ENISA) handles information and network security throughout the European Union (EU). Guidelines set by ENISA are to Prepare, Detect, Notify and Respond to a security incident. [14] Accuracy and severity must be measured when handling confidential user information. Data must remain anonymous and privacy preserving techniques should be applied. Obtaining user consent when handling sensitive data may be required. Data handling must be done with intent to fulfill a purpose.

Many variables are involved when measuring the severity of a data breach. The criticality of the data can be determined on sector-based analysis. For example, a breach of confidential data of an organization can negatively impact their stock price. [5] Goel et al. saw a 5% decrease in stock price when a company is a victim of a confidential data breach. Data breaches not involving confidential information had no effect on the stock price. Organizations lost 2.1% of their market value within a timespan of 2 days from disclosure. [5] The ability to consider other classifications and dimensions when measuring severity can help an organization place emphasis on what the organization considers their crown jewels



The number of records leaked and the cost of a breach have a positive correlation. However, the severity of what is leaked may vary. For example, the disclosure of a specific disease may impact the life of an individual worse 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 severity scores. A disease, such as HIV, can have a major impact on the life of a subject if that information is disclosed. A drug may also increase the severity of a record because the disease can be inferred based on the treatment. L-Severity places a weight on the number of rows leaked to allow flexibility on the influence on severity. For example, 10 records that reveal patients have the cold virus may be considered less severe than 5 records that reveal patients with HIV. Adjusting the weight of the number of records leaked may provide a different result. Similar logic is proposed in KL-Severity allowing each data classification to be weighted. Existing conditions when a leak occurred may also impact severity. Records leaked maliciously can remain undetected longer, which increases the negative impact on an organization. Other factors that increase severity are linkages and the frequency of an attack. Linkages are the relationships that are publically available that can be used to reveal sensitive information. This research analyzes the impact of different privacy metrics on the severity of a data leakage.

II. Previous Work

A. K-Anonymity

Sweeney et al. proposed K-Anonymity, which requires quasi-identifier values to occur at least k number of times within a q-block. A q-block is a grouping of tuples that will have the same set ofvalues for their quasi-identifier attributes. Quasi-identifier attributes can be used in combination to reveal a unique entity. 87% of individuals can be identified by their 5-digit zip code, gender and date of birth. [3] K-anonymity protects against inference and linkage attacks. Sensitive attributes can be breached through unintended disclosures. Data that is retrieved in a single query may not violate the K-Anonymity rule. However, when data is combined with other queries the linkage can disclose sensitive information. Generalized data can be released and unintentionally disclose information about individuals. Inference attacks involves linking the released attributes to other data sources. The government and medical industries commonly release information unaware of other related data sets. Security of data can protect against a direct data breach, but not from information leaked through inference.

Previous research has been done on upholding privacy in released tables. For example, statistical databases are released to provide data for research, data mining or fraud detection. A technique to generalize the data involves adding noise, which can damage the integrity of the information. Multi-level databases (MDB) store data in different classifications. Data is divided into higher and lower classified information. A vulnerability that can be found in a multi-level database is when lower classified information is used to infer higher classified records. A way to mitigate this vulnerability is through architecting a strong database design. However, the replication of data after being disclosed cannot be controlled. Data that leaves the original source can be copied and manipulated many times after. There is little to no oversight of handling the data once the data has reached multiple receivers. To avoid a breach, all sensitive data can be suppressed, but this technique can decrease the value of the data.

K-Anonymity is susceptible to different attacks. An unsorted matching attack occurs when positions of the tuples in each generalized table match the private table. To prevent this attack, randomly sorting the data can be done. Complementary released table vulnerability is when two generalized tables form a linked table. The linked table is used to combine quasi-identifier values to uniquely identify rows in the private table. The complementary released table vulnerability can be addressed by using the quasi-identifiers of the original table. Another technique is to base the new generalized table after the original that was released. When basing the new generalized table from the original table, no value should be more specific than the values in the original table. For example, if the original table generalized their zip codes to 0213*, the new table should not be more specific with 02139. Another vulnerability is a temporal attack. Temporal attacks occur when new data is added to the private table over time and a new generalized table is released. Linking the original released table with the newly released table can reveal unique rows. A way to avoid the leakage is to create the generalized table based on its' original version.

Table 2.A: 2-Anonymity

	Race	Birth	Gender	ZIP	Problem
t1	Black	1965	m	0214*	short breath
t2	Black	1965	m	0214*	chest pain
t3	Black	1965	f	0213*	hypertension
t4	Black	1965	f	0213*	hypertension
t5	Black	1964	f	0213*	obesity
t6	Black	1964	f	0213*	chest pain
t7	White	1964	m	0213*	chest pain
t8	White	1964	m	0213*	obesity
t9	White	1964	m	0213*	short breath
t10	White	1967	m	0213*	chest pain
t11	White	1967	m	0213*	chest pain

[1]

Table 2.A displays a table that conforms to 2-Anonymity. However, there are records that do not have well represented sensitive values. For example, if Alice is a black female born in 1965, Bob can conclude that Alice has hypertension. Rows t3 and t4 have a higher level of severity than diversified rows like t1 and t2 when applying the KL-Severity model.

B. L-Diversity

Ashwin Machanavajjhala et al. proposed two attacks on K-Anonymity, homogeneity and background information attacks. An attacker can discover sensitive attributes when the data is not diverse enough. A homogeneity attack leaks information

[1] Sweeney et al.

due to the lack of diversity in the sensitive attributes. An attacker may have background knowledge, which can be used to infer sensitive information. Ashwin Machanavajjhala et al. proposed Bayes-Optimal and L-Diversity. Bayes optimal is an algorithm that works under the assumption that the data publisher and adversary know the complete distribution set of sensitive and non-sensitive attributes. L-diversity provides privacy without the data publisher knowing how much background information an adversary may have. Although Bayes optimal covers a wider scope, it is not practical in use. It is unlikely that the adversary and data publisher have the complete knowledge of the sets of sensitive and non-sensitive attributes. L-Diversity is the more practical algorithm when measuring privacy.

Each block of quasi-identifier groups, or q-blocks, should have at least *l* frequency of sensitive values. The frequency of the sensitive attributes can protect against knowledge an attacker may know. Ashwin Machanavajjhala et al. proposes two algorithms to define well representation of sensitive values called Entropy and Recursive L-Diversity. Entropy diversity ensures that each q-block has well represented groups of sensitive attributes. The more uniform a q-block is, the higher the entropy. Recursive diversity is an algorithm that measures the frequency, but is implemented differently. Ashwin Machanavajjhala et al. proposes other algorithms to handle non-sensitive attributes, which involve variations of entropy and recursive diversity.

Table 2.B: 3-Diversity

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	1305*	≤ 40	*	Heart Disease
4	1305*	≤ 40	*	Viral Infection
9	1305*	≤ 40	*	Cancer
10	1305*	≤ 40	*	Cancer
5	1485*	> 40	*	Cancer
6	1485*	> 40	*	Heart Disease
7	1485*	> 40	*	Viral Infection
8	1485*	> 40	*	Viral Infection
2	1306*	≤ 40	*	Heart Disease
3	1306*	≤ 40	*	Viral Infection
11	1306*	≤ 40	*	Cancer
12	1306*	≤ 40	*	Cancer

[2]

Table 2.B is following 4-Anonymity and 3-Diversity. The table may have stronger privacy in place, but there is not a way to quantify the severity row by row. Also, the disclosure of the disease may have a different impact on an individual's life. For example, the disclosure of heart disease and cancer can have a different affect on an individual that has a viral infection.

[2] Ashwin Machanavajjhala et al.

C. L-Severity

Table 2.C

Job	City	Gender	Disease	Medication
Lawyer	LA	Male	HIV	Vitamins
Lawyer	LA	Male	Heart Attack	Aspirin
Lawyer	LA	Male	Migraine	Paracetamol
Lawyer	LA	Male	Hypertension	Aspirin
M-Score: 2.000				

(a) Case 1.1

Job	City	Gender	Disease	Medication
Lawyer	LA	Male	HIV	ARV
Lawyer	LA	Male	Hypertension	Statin
Lawyer	LA	Male	Heart Attack	b-Blocker
Lawyer	LA	Male	Migraine	b-Blocker
M-Score: 2.000				

(b) Case 1.2

Job	City	Gender	Disease	Medication
Lawyer	LA	Male	HIV	Vitamins
Lawyer	LA	Male	Flu	Paracetamol
Lawyer	LA	Male	Flu	Aspirin
Lawyer	LA	Male	Migraine	Aspirin
M-Score: 2.000				

(c) Case 2.1

Job	City	Gender	Disease	Medication
Lawyer	LA	Male	HIV	ARV
Lawyer	LA	Male	H1N1	Tamiflu
Lawyer	LA	Male	H1N1	Antibiotics
Lawyer	LA	Male	Flu	Antibiotics
M-Score: 2.000				

(d) Case 2.2

Job	City	Gender	Disease	Medication
Lawyer	LA	Male	HIV	Vitamins
Lawyer	LA	Male	Flu	Paracetamol
Lawyer	LA	Male	Flu	Aspirin
Lawyer	LA	Male	Migraine	Aspirin
Lawyer	LA	Male	Migraine	Paracetamol
Lawyer	LA	Male	H1N1	Aspirin
Lawyer	LA	Male	H1N1	Paracetamol
$x = 1$	M-Score: 3.500			
$x = 10$	M-Score: 0.607			
$x = 100$	M-Score: 0.509			

(e) Case 3.1

Job	City	Gender	Disease	Medication
Lawyer	LA	Male	HIV	ARV
Lawyer	LA	Male	Hypertension	Statin
Lawyer	LA	Male	Heart Attack	b-Blocker
Lawyer	LA	Male	Migraine	b-Blocker
$x = 1$	M-Score: 2.000			
$x = 10$	M-Score: 0.574			
$x = 100$	M-Score: 0.507			

(f) Case 3.2 [3]

The M-Score requires sensitivity functions to be defined by domain experts and calculates a severity metric. 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. 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 that are displayed in Table 2.C. Case 1.x[4] 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.

Table 2.D Score Matrix

Case	L-Severity	M-Score		
		$x = 1$	$x = 10$	$x = 100$
Case 1.1	2.050	2.000		
Case 1.1	2.900	2.000		
Case 2.1	1.150	2.000		
Case 2.2	2.100	2.000		
Case 3.1	1.950	3.500	0.607	0.509
Case 3.2	2.900	2.000	0.574	0.507

[3] Vavilis et al.

[4] X represents cases 1 and 2

Table 2.D shows the result matrix of the 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. Although the severity is properly adjusted for the impact on an individual, it does not consider the diversity of the sensitive attributes. For example, a q-block with only 1 type of disease should have a higher severity than a q-block with 2 distinct values.

Table 2.E

Job	City	Gender	Disease
Lawyer	LA	Male	Ebola
Lawyer	LA	Male	Ebola
Lawyer	LA	Male	Ebola
Lawyer	LA	Male	Ebola

Job	City	Gender	Disease
Lawyer	LA	Male	Cancer
Lawyer	LA	Male	Heart Disease
Lawyer	LA	Male	Chicken Pox
Lawyer	LA	Male	Rabies

Assuming all displayed diseases have the same severity, using the L-Severity equation both tables would have equal severity scores. However, the less distinct number of sensitive attributes there are, the farther the q-block is from being L-Diverse. The privacy of an individual is considered by applying a Dependency Factor in the L-Severity equation. Vavilis et al. suggests that other severity metrics such as K-Anonymity and L-Diversity be analyzed on the impact on severity.

III. PRELIMINARIES

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

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.

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

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.

$$(c, L) - Diverse = r_1 < c(r_2 + \cdots + r_m)$$

(c, l)-Diversity: Let r_i represent the frequency of a distinct attribute. The most frequency sensitive attribute is r_1. The frequencies are listed in descending order. A q-block is considered (c, 2)-diverse if the frequency of the most occurring sensitive value is less the product of c and the sum of the remaining frequencies.

$$KL - Diversity_L = \sum_{r \in R^{L(b_1,\ldots,b_m)}} KLSENS_r$$

KL-Diversity: Let $L\ (b_1, \ldots, b_m)$ be the rows of the leaked table.

$$KLSENS_r = \left|R^{L(a_1,\ldots,a_n)}\right|^{\frac{1}{x}} \left[\left(\frac{\frac{CS}{DF}}{DivFactor}\right) + WR(QBS_r)\right]$$

KL Record Sensitivity: Let $|R^{L(a_1,\ldots,a_n)}|^{\frac{1}{x}}$ represent the number of leaked rows. DF's definition will be borrowed from L-Severity.

$$CS_r = \forall c_n \left[\sum_{a_i \in A} NS(A_i[r])\right]^{\frac{1}{w_n}}$$

Record Classification Sensitivity 3: Let CS_r be the Classification Sensitivity of a row. For each classification of a row, c_n, record the node sensitivity of the attribute. Let $\frac{1}{w_n}$ be the weight given to the classification.

$$QB_r = \forall q_i \in Q\ q_i[r] = q_i[r_i]$$

Q-Block: Let QB_r represent a q-block that row *r* is a party of.

$$DivFactor = \{r_i | \forall s_i \in QBS\ s_i[r] = s_i[r_i]\}$$

Diversity Factor: The DivFactor, or diversity factor, represents the number of distinct sensitive values for a given q-block. Let QBS represent the set of sensitive attributes that reside in the q-block.

$$WR(QBS_r) = \begin{cases} 1 & if\ (c, L) - Diverse \\ 0 & else \end{cases}$$

Well-Represented Weight: Let WR represent the added weight if a q-block is not well represented.

IV. KL-SEVERITY

K-Anonymity and L-Diversity provide rules to prevent disclosing sensitive information. To the best of our knowledge, there has not been any published severity research that considers the L-Diversity metric and weighted classifications. L-Severity uses a distinguishing factor, but does not evaluate the impact of different privacy preserving metrics on the severity. L-Severity applies scores only to sensitive attributes. We propose KL-Severity that expands on the L-Severity model to include different types of data classifications and to attach a weight to each set. KL-Severity also considers the frequency and number of distinct sensitive attributes in a table. The weight attached to a classification set will allow for more detailed analysis. For example, if we want to place stronger emphasis on the quasi-identifier attributes, we can increase the weight.

Quasi-identifier values can impact the severity of a data leakage. For example, a leakage involving diseases can have different impacts on individuals at certain stages of their lives. LT1 and LT2 are two tables with identical attributes except for age. Assuming the severity score for the diseases are equal, the traditional L-Severity model would conclude that these two tables that were leaked have equal severity. However, the severity of LT2 can be debated because the table contains sensitive information regarding a minor.

LT1

Job	Age	Disease
Student	100	HIV

LT2

Job	Age	Disease
Student	10	Ebola

Quasi-Identifier Sensitivity Scores

Quasi-Identifier
f(age < 18) = .8 f(age) = .5

Using the L-Severity model only sensitive attributes are considered without any separation from quasi-identifiers or non-sensitive attributes. Non-sensitive attributes may be misclassified and can give more information than intended. Therefore, attaching a constant score for any added attribute can be beneficial when detecting for data leaks. A limitation of scoring severity is that the domain of these attributes and the classifications must be maintained. If an attribute is misclassified an attacker can target non-sensitive attributes to prevent detection.

JSON Example

{ property_1: value_1 },
{ property_1: value_1, property_2: value_2 }

JavaScript Object Notation (JSON) is a data exchange format. JSON can come in different structures. An object's properties can be parsed. Additional properties can be added without the receiver's knowledge. If the scope of the user's validation does not include checking the properties, the data may make it through into their system or forwarded to another party. This unchecked property can go undetected. If severity is being tracked and the property is not in a maintained score table, an adversary can leverage this to pass or receive sensitive information. Giving a constant score to non-sensitive attributes can be useful when reading unstructured data.

LT3

Job	Age	Disease
Student	*	HIV
Student	*	HIV
Student	*	Ebola
Student	*	HIV

LT4

Job	Age	Disease
Student	*	HIV
Student	*	Ebola
Student	*	Cancer
Student	*	Meningitis

LT3 and LT4 are examples of two leaked tables that we assume have the same severity scores for their sensitive attributes. The difference between LT3 and LT4 is the number of distinct sensitive attributes that are disclosed. LT3 contains 2 distinct values for disease while LT4 has 4. An inference attack on LT3 would be more successful in performing than LT4 because of the lack of diversity in the sensitive attributes. LT3 also does not satisfy the condition of being (c, l)-Diverse. For example, if Alice knew Bob did not have Ebola and knew that Bob attended Hospital A where LT3 was leaked. Alice can infer that Bob has HIV. If LT4 was leaked from Hospital

A, Alice still knows that Bob does not have Ebola. However, Alice will have to deduce amongst three other diseases – Cancer, HIV and Meningitis. The scenario described can be quantified using KL-Severity. Case 1 shows that LT3 has higher severity than LT4. We can assume $c = 1$ and the severity score of all the diseases are the same.

Case 1: Severity of LT3 and LT4

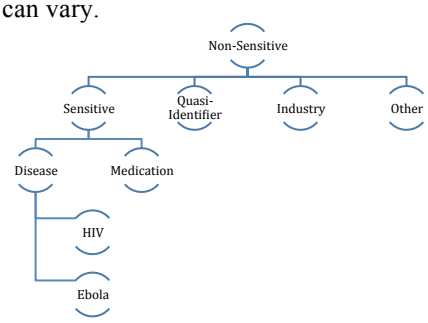
	LT3	LT4
Equation	4 * (20 / 4 / 2) + 1	4 * (20 / 4 / 4) + 0
Severity Score	11	5

V. Discussion

A comparison of the impact of the Dependency Factor (DF) in L-Severity was done against K-Anonymity. This research did not find a significant impact on the severity when alternating algorithms. For example, if a table is conforming to the K-Anonymity rule, the DF can also remain constant or decrease severity with a higher DF metric. Having a higher DF metric will reduce the severity of a row. However, a higher DF score does not guarantee that a leaked table conforms to the K-Anonymity rule. In order for a record to follow the K-Anonymity rule, it must be part of a group of records that is at least *k* in size. After our analysis of M-Score's DF metric and K-Anonymity we concluded that the metrics will have almost equivalent influence on the severity score. An addition that can be added to the overall table is a weight that determines if a table followed the K-Anonymity rule. A check may be desired to avoid linkage and temporal attacks. K-Anonymity is a good baseline for measuring privacy within a generalized dataset. We attempted to measure the impact of considering how far off a group of records were from *k.* The more the number of unique quasi-identifiers a group of records is less than *k,* the higher the severity will be. This correlates with a lower DF metric. For example, if there is only 1 distinguished record out of *n* records, *n/1* is greater than *n/k.* Assuming that *k* is larger than 1.

Having the option to attach weights to different data classifications allows for more detailed analysis. For example, it is possible to weigh privacy higher than other classifications. Classifications can include sensitive, quasi-identifier and user-defined attributes. KL-Severity allows for the flexibility to add *n* number of classifications. An example of a classification that can be added is one that contains rules for attributes that are important to a specific industry. Tables LT1 and LT2 show two leaked tables where LT2 involves a minor. Proper logic must be put in place to capture the scenario. Another approach is to consider the context of each attribute. For example, if a doctor is retrieving sensitive attributes it may be considered more "normal". However, if the janitor retrieved a set of sensitive attributes, this scenario may be considered more severe.

Measuring the diversity and frequency of the sensitive attributes contributes to the accuracy of the severity score. As seen in Case 1 in Section IV, the severity a record is increased when the sensitive attributes are not diversified or well represented. Using L-Severity with the assumption that the diseases have the same score would result in a draw. A security professional would not be able to determine at a high level which leakage is more severe. A worse case would be if a breach remained undetected. If the score resulted in a true negative, a company's reputation can be damaged and consequences can vary.



Classification Model

The classification model shown above is a proposed approach when establishing scores. The root, non-sensitive attributes, would represent the parent objects. The attribute tree's leaves would be the more specific classification, which can be further broken down.[5] Vavilis et al. proposed a similar model to represent severity score. The model allows for nodes to have default severity scores. The severity scores can be overridden for specific scenarios. For example when HIV combined with a common medication that is used for the treatment of the disease can have a specific score assigned when both are present. However, the default score may be different. Having non-sensitive attributes as the root node allows for all attributes to have a constant score. This will prevent an attacker from attempting to leak only non-sensitive attributes. Non-sensitive attributes may be a target because the severity and relationships must be maintained. It is possible that the attributes are not truly represented in a proposed model. Although adding a constant score to a value can be helpful in detecting a breach, it does not protect the damage that was already done going undetected and below a threshold to signify an alert. A shortcoming of the severity measurements is the need to maintain the list of classified attributes with certain scores. Each classification can have a default score, but if a new value appears it will need to be placed in its proper classification. By default the score will inherit what's specified for the root. Future research may want to propose a model to determine what classification a new value might be placed in based on the values of other attributes. A good example could be the placement of new diseases. If a new STD is passed to a system, whether the data was received through a feed or an upload, the value should be properly placed in a sensitive classification. An undesirable scenario would be that it is placed in the non-sensitive classification and unintentionally disclosed. A better approach that can be utilized would be to automatically assume new values are the most sensitive of all attributes. The high severity score would

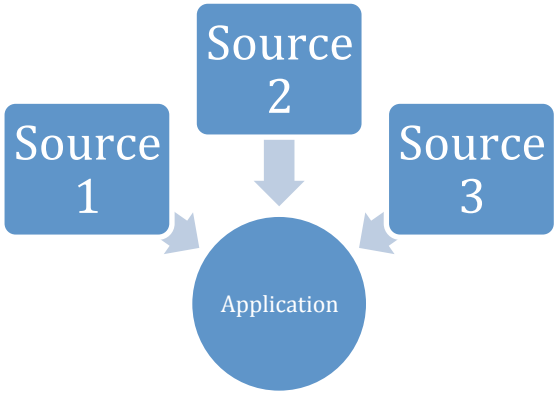
[5] Please see the sensitive attribute node.

result in many false positives forcing the maintenance of the table to be consistent.

A. System Architecture

Normal traffic of an application or system must be defined. Vavilis et al. use examples where data is queried from a system. A baseline of "normal" should be established in order to detect anomalies in severity. Data Leakage Prevention (DLP) technology is correlated with organizations that have teams to prevent data theft. [11] McAfee performed a study where 64% of security professionals within firms that experienced a data breach agreed that the breach could have been prevented if their firm used DLP technology. [11] DLP technology is a top tool in detecting insider threats. A tool that detects the severity of data being retrieved can be helpful when investigating security events. Future work will involve architecting a system that can be used to measure severity on data sets. Finding vulnerabilities in detecting KL-Severity can be done. An interesting test would be to attempt to bypass security controls by retrieving sensitive attributes while keeping within "normal" severity measures. Retrieving the sensitive data in smaller chunks and at higher frequencies can be an attack to perform on KL-Severity.

KL-Severity Architecture



The KL-Severity Architecture proposes a way to consolidate data from multiple unrelated sources in one application. The proposed architecture is beneficial for scaling applications. Implementing severity measurements at a database level does not consider data from other sources unless database links are set up or data from other sources is moved to the targeted database. However, the latter suggestions can only be done if the parties are in agreement with each other in exchanging data. The sharing of data may not be possible between all parties and alternatives would have to be considered. An approach could be to use data structures such as binary and balanced trees can represent the data model's relationships. The proposed architecture allows for more scalability than being limited to one data source.

B. Conclusion

The importance of data leakage prevention is relevant in today's media and influences how we use and collect data on a day-to-day basis. Previous work emphasizes on finding a privacy metric that takes account the entire table. KL-Severity provides these algorithms at the tuple and q-block level.

Providing security metrics at a database level is beneficial, but having the option to do so at an application level can be more robust. Tracking transactions within an application can alert an organization when a possible breach has occurred or in transit. A breach may go undetected until a victim reports a problem or an attacker advertises the data on the black market. Severity is complex and can involve many variables and parameters. These parameters can be based on common characteristics of data or specific to an industry. For example, quasi-identifiers and industry specific classifications can be defined. However, there are few a publications that focus strictly on the severity of the data and the definitions of the impact of values for sensitive, quasi-identifier and non-sensitive attributes. Vavilis et al. created a model to quantify severity by attaching scores to values within a sensitive domain. L-Severity does not separate the different classifications of data. We propose KL-Severity that is a more scalable and accurate model.

Data can be retrieved in different formats from various sources. For example, an API can accept or send JSON that may have varying properties. Due to the unstructured format that the data can come in, unexpected attributes may be passed. Depending on how this data is used, extra information may be leaked or accidentally disclosed to an unauthorized user. An example could be a data dump of values that need to be emailed to another group or data that is passed into another system. This can cause system errors, rejections and leakages.

Future research can involve more work on data classification. Ashwin Machanavajjhala et al. proposed different algorithms to handle *don't care sets* (DCS). DCS represents sensitive attributes that have no effect on severity. Although DCS has no effect on severity, when mishandled can reveal sensitive attributes. KL-Severity can be used with other privacy metrics, such as T-Closeness, to measure severity. Future work will involve conducting tests using KL-Severity with real data.

VI. References

[1] Machanavajjhala, Ashwin, et al. "l-diversity: Privacy beyond k-anonymity." ACM Transactions on Knowledge Discovery from Data (TKDD) 1.1 (2007): 3.

[2] Vavilis, Sokratis, Milan Petković, and Nicola Zannone. "A severity-based quantification of data leakages in database systems." Journal of Computer Security 24.3 (2016): 321-345.

[3] Sweeney, Latanya. "k-anonymity: A model for protecting privacy." International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 10.05 (2002): 557-570.

[4] Vavilis, Sokratis, Milan Petković, and Nicola Zannone. "Data leakage quantification." IFIP Annual Conference on Data and Applications Security and Privacy. Springer Berlin Heidelberg, 2014.

[5] Goel, Sanjay, Christopher Brown, and Hany Shawky. "Measuring the impact of security breaches on stock valuations of firms." Information & Management 46.7 (2009): 404-410.

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

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

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

[9] Only 0.5% of All Data is Currently Analyzed, A.R. Guess, June 10, 2015

[10] Recommendations for a methodology of the assessment of severity of personal data breaches, Data Protection Authorities of Greece and Germany, Clara Galan Manso, ENISA, Sławomir Górniak, ENISA, December 20, 2013

[11] Grand Theft Data - Data exfiltration study: Actors, tactics, and detection, McAfee, 2015

[12] "Data Leakage – Threats and Mitigation." *SANS Institute*. N.p., 15 Oct. 2007. Web. 10 May 2017.

[13] "Data Loss Prevention." *SANS Institute*. N.p., 6 Sept. 2015. Web. 10 May 2017.

[14] Li, Ninghui, Tiancheng Li, and Suresh Venkatasubramanian. "t-closeness: Privacy beyond k-anonymity and l-diversity." *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*. IEEE, 2007.

[15] Stiennon, Richard. "Categorizing data breach severity with a breach level index." *SafeNet Inc* (2013): 3.