A Comparative Analysis of Various Techniques of Data Leakage Detection in Different Domains

Kiran Patil, Harsha Sonune, Soniya Devikar, Vrushali Chaudhari, and Isha Ayachit

Abstract With the steep growth in information technology and its global reach, as well as the common citizen's ever-increasing reliance on technology, data privacy and security have become a major source of concern for individuals all over the world. In today's era, computing devices like virtual servers, databases, physical servers, databases, and many more devices are occupied with confidential data. This paper is an exploratory case study that analyzes the various algorithms and methods proposed across the various domains, and a comparative analysis was done.

Keywords Data leakage detection · Android · Networking · Cloud computing · Machine learning · Guilty agent · Privacy · Watermark · Fake object · Bigraph

1 Introduction

Data is critical and an important aspect as it contains sensitive information related to user privacy, finance, social privacy, etc. This data is a precious asset to companies and organizations for project and business insights. Nowadays, almost all of our data is becoming digital, and because of online sharing platforms, data privacy stands a major risk.

In Jan 2021, about 35 million user accounts data from Juspay was for sale on dark Web. An un-recycled access key was used to steal details of around 35 million

K. Patil (✉) · H. Sonune · S. Devikar · V. Chaudhari · I. Ayachit
MKSSS's Cummins College of Engineering for Women, Pune, Maharashtra, India
e-mail: kiran.m.patil@cumminscollege.in

H. Sonune
e-mail: harsha.khedkar@cumminscollege.in

S. Devikar
e-mail: soniya.devikar@cumminscollege.in

V. Chaudhari
e-mail: vrushali.chaudhari@cumminscollege.in

I. Ayachit
e-mail: isha.ayachit@cumminscollege.in

S. Fong et al. (eds.), *ICT Analysis and Applications*, Lecture Notes in Networks and Systems 517, https://doi.org/10.1007/978-981-19-5224-1_73

consumer accounts, including disguised card data and fingerprints, from a server. In March 2021, around 45 lakh user data was leaked from known organization Air India, whereas Dominos reported 18 crore users' data breach in May 2021.

A single corporation may have access to the personal information of millions of customers, which it must keep private in order to protect consumers' identities. Malicious third parties or hackers can steal this information for their own gain at the expense of ordinary people and businesses' private information. Data can be exposed in a variety of ways. It could be the result of a cyberattack designed to steal data, or it could be an employee purposefully leaking or selling private data to a third party. Domains covered in this paper are Android, ML, deep learning, networking, cloud computing, and fake/guilty agents.

2 Literature Survey

There are many different strategies or techniques that are developed or suggested so far for detection and prevention of data leakage. Here are some of the strategies/techniques used:

Cam recommended uitXROM for the purpose of detecting leakage of sensitive data in custom created Android firmware by inspecting the connection between apps that were pre-installed [1]. This system consists of three major modules: APK extractor (extracts pre-installed apps), APK analyzer (detects sensitive data flow via multi-applications), and path matcher (analyzes data paths based on path entry and path exit points). The results of the experiment show that the system can detect several applications that are already installed that leak sensitive data from over 280 ROMs that are found online or downloaded from the Internet.

The authors of the paper Naik and Gaonkar suggested watermarking to detect data leakage in clouds [2]. Watermarking approach takes data to be conveyed in picture, and a quick response code is inserted. The generated QR code watermark is embedded into cloud data. Current data characteristics are examined to estimate tampering in data, and the guilty agent is identified by taking out watermark and collating watermark data to the agent's details. Existing systems can provide security by encryption utilizing numerous techniques; however, the proposed paradigm provides both security and detection.

Ranchal and others proposed a security enforcement implementing security protocols in composite online services structure [3]. It makes use of active bundles. When the AB engine detects an assault, the EPICS technique enables dynamic data destruction to prevent disclosure, which is not possible with models that treat data as passive entities. The proposed framework is backwards compatible with existing service architecture and meets the real-time needs of Web service interactions.

Rastogi proposed a Uranine technology that monitors Android apps in real time to detect privacy breaches [4]. Upon receiving an app, it is converted by Uranine to a custom IR which can be implemented during runtime for taint propagation. The IR that has been instrumented is then translated to bytecode, following which a fresh

app is created and reassembled to create a new app that can be downloaded and installed on an Android device. Privacy leaks are automatically tracked when the instrumented application runs. Uranine implements Android apps which is why it does not require platform support for tracking information flow. Uranine has good accuracy and a low performance overhead when compared to other practices.

For detecting attacks in huge networks in real time, Wu and others present a method for converting raw traffic vector form to image data form [5]. The image form data has the potential to reduce the number of computing parameters. The study suggests a technique for using the training sample quantity as a basis for determining the weight coefficients of each class's cost function and the purpose of which is to increase the detection of accuracy of the unbalanced traffic dataset. Experiment results indicate that the suggested CNN intrusion detection model performs better than the pre-existing intrusion detection techniques.

Min and others propose TR-IDS, an unique intrusion detection framework that uses both features designed manually and payload features to increase performance [6]. To extract primary features from payloads, it uses two NLP techniques, namely word embedding and Text-CNN. Word embedding maintains semantic relationships among bytes while reducing feature magnitude, and Text-CNN is then utilized to pull out features from all payloads. For the final categorization, it used a complex random forest algorithm.

The technique introduced by Liu and team is a CNN-based multi-classification detection system for network intrusion [7]. The experimental findings were compared to deep learning models like DNN, GRU-RNN, LSTM-RNN, and others using the KDD-CUP99 and NSL-KDD datasets. The CNN-based model enhances recall and accuracy, minimizes false positive rate, and produces improved outcomes for unexpected attacks detection.

Alrawashdeh and Carla Purdy introduced a deep learning-based method for detecting anomalies that has a high accuracy and efficiency rate on the remaining 10% of the testing data from the KDDCUP99 dataset [8]. This method is based on a deep belief network fine-tuned with logistic regression softmax. To increase the overall performance of the network, a multiclass logistic regression layer was implemented and trained with 10 epochs on the upgraded pre-trained data. In addition, they simulated a network with a short training time and minimized the dataset's preprocessing.

Gupta and Singh have introduced a discriminatory criminal model that identifies malicious entities involved in a confidential data breach and provides security to prevent breach of personal data [9]. This requires addressing the issue of data breaches. Based on the assigned data to different clients or agents, this implementation foretells the guilty agent. In the model, the supplier distributes the given data item shared by numerous agents. Bigraph is used to represent these things. We use this graph to find the matrix, which gives us a list of the agents to whom each data object has been assigned. If an agent discloses sensitive data objects to a malicious third party and the distributor later discovers them in an illegal location, the distributor is liable.

Govinda and Divya Joseph have employed the data allocation methods to determine the exact moment and also the guilty agent. This demonstrates how the distributor can use a basic approach of introducing bogus items to prime line identification during the initial distribution phase [10]. These bogus objects that are injected have no relation to the actual data, yet they appeal to the distributed agent as real data. This notion is synonymous with the concept of embedding watermarks. Where there are parallels to be drawn based on the fact that object insertion works in a similar way to hiding a watermark. The distributor may readily identify the agent who is responsible for this with the help of these false things. This method also delivers evidential proofs that accurately identify the guilty agent.

The most famous prevention technique for data leakage in peer-to-peer networks is the data loss prevention technology also known as DLP, but it has some disadvantages. For example, due to the high-detection-error rate, it might prohibit transferring of a normal file externally from the organization, and it can also cause breach of privacy of internal staff for filtering of data. Hence, Chae proposed a system that can prevent internal data loss for employees and customers via a peer-to-peer network [11]. The suggested system can identify the personal data and expel it from the sharing file using the risk factor of privacy data leakage. This procedure also addressed DLP system issues like high-detection-error in data privacy and concerns of invasion of privacy.

To prevent data leakage by internal staff, companies enter the behavior pattern associated with data leaking into the system ahead of time and identify the employee as the staff who released the data, whose behavior pattern is then identified. However, because the data was not entered into the system, the reason for the data leakage cannot be correctly determined if it is released according to the pattern of security log occurrence. Seo and Myung-Ho Kim propose a system that uses convolutional neural networks to identify the cause of data leakage, in which the data leakage pattern is defined as a series of security logs that can appear immediately at the time of data leakage rather than being fed into the system [12]. As a data leakage judgment scenario, these security logs might arise as a result of an association analysis algorithm, i.e., the Apriori algorithm.

The majority of data leakage events occur while employees are performing routine operations, such as sending email that contains sensitive information accidentally. Sensitive information, such as bank records and credit card scores, is sent through email. Wang suggested an email protection system by explaining threats to an email and why it is important to protect emails [13]. The key advantage of building this system on gateway is that it safeguards the company's important information from rivals.

A research that analyzes the present condition of security monitoring managed by three Korean enterprises and suggests certain risk scenarios about unauthorized access is assessing risk scenarios regarding security monitoring system vulnerabilities with an emphasis on information leakage by insiders. The study has gathered each business's policy on security, systems for security monitoring and the utilized system log for the case analysis. As a result of this research, four risk situations that

are probable to occur in future were discovered, as well as threats that were hard to identify using the current security monitoring system [14].

Time stamps are very important in data leakage prevention for providing permission to access a certain data because the data is confidential during a specific period of time [15]. Because the same data could become non-confidential after the timestamp, a technique for preventing data leakage with time stamps was proposed.

Karaçay and others proposed a protocol for detecting intruders over encrypted Internet data in which the detection model and the entire network are kept private [16]. The authors used a homomorphic encryption algorithm as the attack structure and used a personally evaluated decision tree on the network data. The analysis demonstrates that by regulating a number of parameters such as the list of laws, the numeral characteristics, and the list of categories in feature representation, security, and privacy can be improved. The proposed method improves privacy as well as quality in terms of execution time.

Another method of detecting data leakage is to use a solution that can be delegated and deployed in a reasonably ethical detection environment. Acts to detect data leakage technique named "fuzzy fingerprint" is used to improve data privacy. This procedure is based on a one-way calculation of sensitive data. This enables data owners to securely handle content monitoring responsibilities to DLD providers while keeping important documents protected [17].

3 Observation

Data leakage detection system can be covered under various domains. These domains are guilty/fake agent, networking, cloud computing, Android, machine learning, artificial intelligence, and deep learning.

The guilty/fake agent domain introduces techniques for detecting data leaks produced by malicious entities intentionally or unintentionally transmitting confidential data to unauthorized third parties.

Networking domain includes P2P file sharing and mobile ad hoc network. Using the P2P network, which is fundamentally a direct link between computers. P2P file sharing allows registered users to exchange and share files with fellow peers without the usage of networks. The disadvantage is that it may allow internal personnel and customers' personal information to be leaked.

Cloud computing includes techniques such as watermarking, use of active bundles, and other methods to detect data leakages in cloud and other Web services.

Android mainly focuses on analyzing relationships of pre-installed applications and technologies that monitor Android apps in real time to detect privacy breaches.

Machine learning, artificial intelligence, and deep learning mostly are about intrusion detection systems. This domain does not cover much about data leakage detection but is useful for predicting various kind of attacks like denial of service, remote to local, user to root, and probing for follow-up procedures.

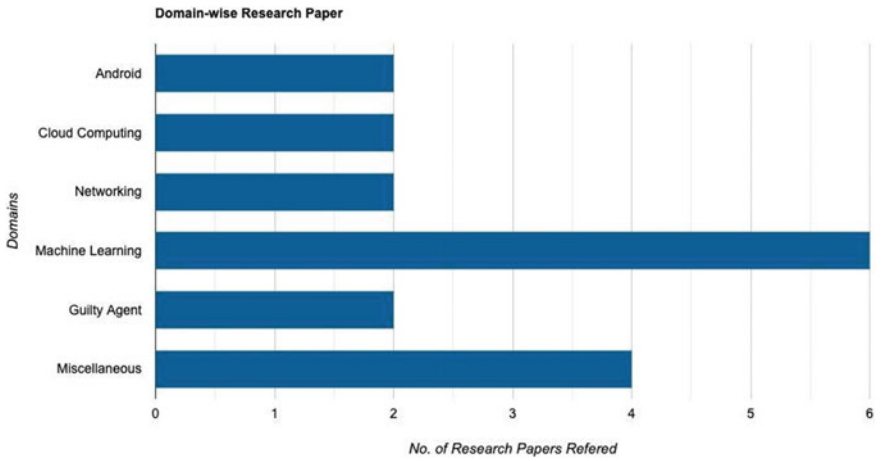

Fig. 1 Domain-wise comparative analysis of research papers taken for reference

Figure 1 represents the following statistics of research papers from various domains:

Android—2
Cloud computing—2
Networking—2
Machine learning—6
Guilty agent—2
Miscellaneous—4.

4 Conclusion

There are several ways for leakage of data to happen, and the research papers cover various techniques and methods to detect the data leakage. Data can be leaked by an insider or an employee of the organization, or by a third party agent to whom the distributor shares data with, or by an external attack like hacking. These are some of the ways that are covered by this report as well as methods and techniques that can be implemented as solutions for these problems. According to the troubles faced by an organization or a company, appropriate methods can be integrated within their systems categorized by the various results and output of each technique.

Among the machine learning, artificial intelligence, and deep learning papers, we found that the information-retrieval-based technique is suitable for data leakage detection. This technique is useful for Web data leakage detection for sensitive data. The system monitors the Web and collects information about Web documents according to user's preferences. If a document on the Web appears to be semantically

similar to confidential user documents, the system alerts the user to the possibility of data leaking.

Out of all the techniques in the cloud computing domain, we found the use of active bundles for data leakage detection more suitable. This technique protects data from malicious cloud administrators by providing data integrity and confidentiality, and the method also allows for access control based on roles and attributes.

In the Android domain, uitXROM technique is more suitable for data leakage detection. This approach detects sensitive data leaks in already installed apps on modified Android ROMs.

In the field of networking, wireless networks like MANET are a very unique application due to scalability and mobility. Data leakage detection and reduction (DLDR) with lightweight cryptography is a method for detecting and reducing data leakage via the Internet. Use of lightweight cryptography and S-Max algorithm supports data confidentiality and reduces scope of data leakage.

There are many algorithms and methods proposed to find out the guilty or fake agent who maliciously leaks confidential information and based on the organization's needs and requirements as well as the based on the particular region which is in essential demand for security and privacy, the appropriate method can be utilized effectively from the comparative analysis provided in this analytical case study.

References

1. Cam NT (2017) Sensitive data leakage detection in pre-installed applications of custom Android firmware. https://doi.org/10.1109/MDM.2017.56
2. Naik R (2019) Data leakage detection in cloud using watermarking technique. In: 2019 international conference on computer communication and informatics, pp 1–6
3. Ranchal R (2019) EPICS: a framework for enforcing security policies in composite web services. https://doi.org/10.1109/TSC.2018.2797277
4. Rastogi V, Qu Z, Mcclurg J, Cao Y (2015) Uranine: real-time privacy leakage monitoring without system modification for android, vol 2, pp 256–276. https://doi.org/10.1007/978-3-319-28865-9
5. Wu K, Chen Z, Li W (2018) A novel intrusion detection model for a massive network using convolutional neural networks. IEEE Access 1. https://doi.org/10.1109/ACCESS.2018.2868993
6. Min E, Long J, Liu Q, Cui J, Chen W (2018) TR-IDS : anomaly-based intrusion detection through text-convolutional neural network and random forest, vol 2018
7. Liu G, Zhang J (2020) CNID: research of network intrusion detection based on convolutional neural network, vol 2020
8. Alrawashdeh K, Purdy C (2016) Toward an online anomaly intrusion detection system based on deep learning. https://doi.org/10.1109/ICMLA.2016.167
9. Gupta I, Singh AK (2017) A probability based model for data leakage detection using bigraph, pp 1–5
10. Govinda K (2017) Dynamic data leakage using guilty agent detection over cloud. In: 2017 international conference on intelligent sustainable systems (ICISS), pp 744–746
11. Chae C, Shin Y, Choi K, Kim K, Choi K (2015) A privacy data leakage prevention method in P2P networks. https://doi.org/10.1007/s12083-015-0371-x
12. Seo M (2017) An advanced data leakage detection system analyzing relations between data leak activity. Int J Appl Eng Res 12(21):11546–11554

13. Wang S (2017) Data leakage prevention: e-mail protection via gateway data leakage prevention: e-mail protection via gateway
14. Kim K (2019) A study on analyzing risk scenarios about vulnerabilities of security monitoring system: focused on information leakage by insider. Springer, Berlin
15. Peneti S, Rani BP (2016) Data leakage prevention system with time stamp. In: ICICES, pp 3–6
16. Karaçay L, Sava E (2019) Intrusion detection over encrypted network data. https://doi.org/10.1093/comjnl/bxz111
17. Shu X, Yao D, Bertino E (2015) Privacy-preserving detection of sensitive data exposure. IEEE Trans Inf Forensics Secur 10(5):1092–1103. https://doi.org/10.1109/TIFS.2015.2398363