AP2I: Adaptive PII Scanning and Consent Discovery System

Somchart Fugkeaw
School of Information, Computer, and Communication Technology (ICT)
Sirindhorn International Institute of Technology
Thammasat University
Pathum Thani, Thailand
somchart@siit.tu.ac.th

Ananya Chaturasrivilai
School of Information, Computer, and Communication Technology (ICT)
Sirindhorn International Institute of Technology
Thammasat University
Pathum Thani, Thailand
ananya_tata@hotmail.com

Pitchayapa Tasungnoen
School of Information, Computer, and Communication Technology (ICT)
Sirindhorn International Institute of Technology
Thammasat University
Pathum Thani, Thailand
pitchayapa.t1@gmail.com

Weerapat Techaudomthaworn
School of Information, Computer, and Communication Technology (ICT)
Sirindhorn International Institute of Technology
Thammasat University
Pathum Thani, Thailand
weerapat.tec@hotmail.com

2021 13th International Conference on Knowledge and Smart Technology (KST) | 978-1-7281-7602-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/KST51265.2021.9415803

***Abstract*—Currently, there are several off-the-shelf Personal Identifiable Information (PII) scanning tools available as an assistive tool to find the PII in the network and system endpoints such as server, personal computer, devices, or cloud storage. Most tools have been designed to support General Data Protection Regulation (GDPR), Health Insurance Portability and Accountability Act (HIPPA), California Consumer Privacy Act (CCPA) etc. In Thailand, the Personal Data Protection Act (PDPA) will be enforced in May 2021. Organizations in Thailand are thus now going to be faced with more challenges in relation to security and privacy than they have ever had before. In this paper, we propose a AP2I system which is an adaptive PII scanning and discovery tool for helping the organization to automatically discover and manage PII for enhancing their privacy policy and satisfying PDPA compliance. Our AP2I consists of four key modules including (1) *Converter* converting text to csv format, (2) *PII Scanner* scanning the PII based on the extended function of Presidio, (3) *Consent checker* checking the analyzer and recognizer function of Presidio for the scanning process, and (4) *PII inventory* storing well-organized PII identifying PII records, data subject, and their source. We also conducted the experiment to demonstrate the efficiency of our proposed system.**

Keywords—PII, PDPA, Discovery tool, data subject

I. INTRODUCTION

PII is a primary concern for most organizations due to data privacy or data protection laws or regulations. Therefore, discovering where your employees, customers, or stakeholders' PII data resides is a crucial factor of maintaining compliance and avoiding breaches or loss. Personal Data Protection Act (PDPA) is concerned with the protection of personal data from the unauthorized use, disclosure, collection, and processing. It forces businesses to obtain consent from the data owner or data subjects before using their personal information for any business purposes. Organizations are required to designate at least one individual, known as the data protection officer (DPO), to oversee the data protection within the organization as well as to ensure compliance with the PDPA. According to the Thailand Personal Data Protection Act that will be enforced in May 2021, the organizations are obliged to protect the privacy and provide safeguard to the PII they keep on. The scope for the protection covers PII of their employees, customers and third-party vendors. Although there are tools for the organization to detect PII, most of them are developed to comply with GDPR and not yet for Thailand's PDPA.

PII scanning and discovery can be considered as a foundation stage for successful implementation of GDPR and PDPA. Most of the tools available in the market today can be used for scanning PII in only one platform, which is either file based or web based. Also, the existing tool is an ad-hoc scanning or auto scanning based on the schedule set. The scanning task is exhaustive—hence it incurs expensive computation. To this end, we are motivated to develop a PII scanning and discovery tool that is able to discover PII from more common data sources (files, database, email, and cookies) with optimized scanning and discovery cost.

In this paper, we propose a PII scanning and discovery tool as a part of privacy management applications for supporting PDPA compliance. The tool is used to detect any PII data that exists in various sources. The scanning method uses regular expressions to detect PII and compare with the rules. After the scanning process, the tool will discover and classify the sensitivity of the data and generate reports to the users. The report shows a list of the PII data discovered, location of the data, graph of the categorized data and severity. Based on our PII inventory verification scheme, data processor or data controller can check the consent status of data files which contain PII against the existing consent retained in the storage. This enables the process of PII discovery and consent management to be done in an efficient and accurate manner.

Most existing PII scanning tools are dedicated for scanning and discovering sensitive data such as personal information, payment data. The tools generally determine where personal and sensitive data is located and what it contains. This helps organizations gain insight to visibility of personal data available in their control and ensure compliance with regulations such as GDPR, CCPA, Payment Card Industry Data Security Standard (PCI DSS), HIPAA and more.

Essentially, this paper entails the fundamental requirement for gathering the PII data. Our project is a part

978-1-7281-7602-4/21/$31.00 ©2021 IEEE

of privacy management software supporting PDPA compliance. Hence, the focus of this paper is first tackling the PII scanning.

The contributions of our proposed approach are described as follows.

1. We propose a PII scanning and consent discovery system applicable for Thailand PDPA compliance. It facilitates adaptive scanning and discovery of PII in any endpoints systems such as database, web, document files, and cookies. The scanning and discovery process is expressively done in regular expression.

2. We develop an automatic consent validation algorithm to check document files containing PII whether they have got consent from the data subject.

3. We introduce a scalable and expressive PII inventory storing PII data set with comprehensive details of PII for further retrieval and analysis. This enhances the ease of integration to any data loss prevention (DLP) tools.

4. We present the implementation of our AP2I prototype system to demonstrate its efficiency and practicality in real cases.

The remainder of this paper is organized as follows. Section 2 discusses related works. Section 3 presents our proposed A2PI system. Section 4 describes the experiments. Section 5 gives the concluding remark and identifies future works.

II. Related Work

Most research works related to PII discovery relied on string matching technique to discover PII and the core focus is dedicated to detecting breaches on PII. For example, AntMonitor [1] and Lumen [2], provide a blacklist of potential PII leaks represented in string and the system then work on deep packet inspection (DPI) on each packet to match any PII strings in headers and/or payload. In Recon system [3], classifiers were trained to detect PII within packets. The traffic was routed to and analyzed at a centralized remote server. In [4], the authors proposed a PII detector for virtual network (PIID VNF) to detect plaintext PII strings embedded in HTTP fields. The system investigates the packets containing PII and then they are marked by the PIID. Depending on the desired system behavior, a PII- containing packet may be either taken into the analysis for further breach detection or instantly discarded by the PIID instance.

In [5], the authors proposed a method that can automatically discover various types of PII in the network system. They developed a list of constraints, or seed rules expressed in regular expressions, based on the format of expected PII data. For PII data types that may not be as easily expressible as a regular expression such as customer names, cities, and regions, seed rules were expressed as dictionaries containing lists of possible values. However, seed rules generated did not cover all cases, formats, and languages.

In [6], the authors proposed SecP2I approach entailing a privacy-preserving PII discovery platform in structured and semi- structured datasets. In their approach, PII are detected based on knowledge base, injection of regular expressions in SQL and NoSQL queries, and a reference base. However, the security protection based on SHA-1 is no longer secure.

In [7], J. Huang et al. presented a PII detection approach working on regular expressions to detect and remove PII from the natural language text in the dataset. Their proposed system is the novel one that focuses on detecting PII expressed in handwriting format. Technically, the system tester reads through the text of the entire dataset and manually marks up all the PII from it. The marked up PII are used to test the implemented Python PII detection program.

Recently, Alizadeh et al. [8] applied natural language processing and machine learning to detect PII data leakage. They used the public dataset which are available contracts in PDF files that were converted to text files using the PyPDF2 Python library. Then, the PII can be extracted and the necessary tokenization as well as learning model over PII can be done with the Natural Language Processing (NLP) and Machine Learning (ML) technique.

For the open-source PII scanner, CUSpider [9] develoed by Columbia University. It is a forensic file scanning program that can scan Windows desktops and laptops for PII. The scanning result is presented in a list of PII found in the target folders. It also provides a redaction option for a number of file types from within the application.

Nevertheless, all the above approaches are incapable to automatically verify whether the scanned result indicating the document source has got the consent from the data subject or not. Also, to the best of our knowledge, there are no explicit tools specifically developed for Thailand's PDPA where some specific formats of personal data need to be treated differently.

III. Our Proposed System

This section describes our objectives and the construct of our system model.

A. Objectives

Our PII scanning and discovery system aims to renders the automatic and adaptive scanning with high accuracy and efficient consent discovery feature. In order to achieve this goal, we extend the recognizer and analyzer features of the Presidio in the way that the validation scheme of some PII data formats such as citizen ID, address, car license plate no. is developed to satisfy Thailand's PDPA. This enables the scanning process for any data files to be accurate and reliable. Since consent management is one of the crucial parts of the PDPA regulation, we thus incorporate the consent discovery in our scope. Therefore, another objective of our scheme is to discover the consent of the existing PII source and also identify for the missing ones. Our system cooperatively works with the existing consent management system to provide a fine-grained check of the availability of consent. Web service and API connection are generally used to support the consent checking. The ultimate result is to present the list of PII sources in the PII inventory as well as to automatically indicate the existence of their corresponding consent. Based on the use of our system, data processor or data controller can realize a comprehensive set of PII they collect, process, and store. It also allows DPA to take care and manage the PII in a secure and efficient way. For example, as the PII is well retained and indicated, the

administrator can support data export, PII update or revocation requested by the data owner or data subjects.

B. System Model

Our A2PI system model is illustrated in Fig. 1. It comprises four main components: converter, scanner, consent checker, and PII inventory. Basically, our system is designed to be modular for ease of integration to any PII sources and existing consent management system. The details of each system module are described as follows.

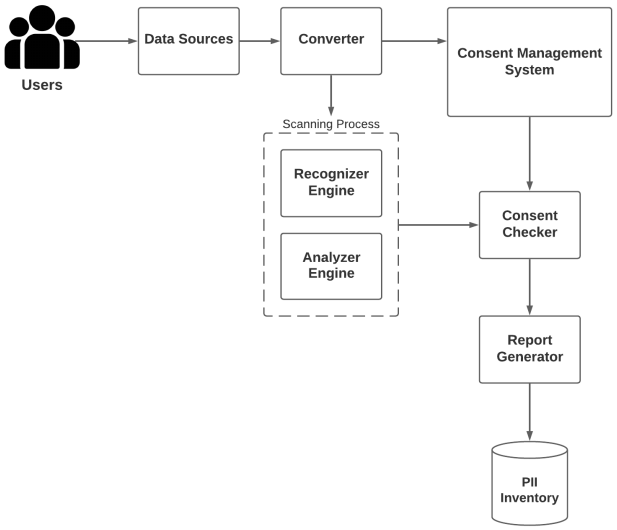

Fig 1: AP2I System Model

1) Converter

Since Presidio is incapable to support various types of data format, we devise a converter to convert any text data such as PDF, Ms. Office, Web pages, and cookies into csv format. Then, CSV data is sent to the PII scanning process. The developed scanner empowers Presidio to be able to scan for PII in other file types. The converter also collects metadata of data sources such as, for databases, table owner, database name, and host or IP address. For file storage, the converter collects file owner and the location of the document. For web pages, we use Scrapy, an open-source web-crawling framework module to work on this part and convert the result to a CSV file and collect URL.

About the converting techniques, there are different handlers for each type of data. For the database, the converter will connect to the database, reach to the table, and dump the table to the CSV file using the database connector module in python. For web pages, scrapy receives the URL as input and starts crawling. If it is successful, Scrapy will save the result to the CSV file.

2) Scanner

Scanner module consists of two engines: Recognizer and Analyzer. We extend presidio's predefined recognizer to improve the performance of the scanning. Also, Thailand's PII format in some categories are not the same as the predefined recognizer provided by presidio for example: ID number, phone number, and address. The custom recognizer needs to be implemented in order to handle Thailand's PII data. The regular expression is added to match for all possible patterns and the score which represents the accuracy is also given to each pattern to make sure that the recognizer is able to detect PII more precisely.

Our algorithm starts by taking input in type text together with recognizer patterns. The algorithm will use the search for the matching between the input text and the recognizer patterns. If it matches and the result is validated, the result is stored in the array list.

Algorithm 1: Pattern Analyzing

Input: text, regex_patterns
Output: results

```
for pattern in patterns:
    matches = regex_module.find(pattern.regex, text)
      for match in matches:
            pattern_result = generate_result_object(match)
            if validate_result(match):
                  results.append(pattern_result)
return results
```

For example, Thai citizen ID consists of 13 digits, optionally separated by a "-" or white space into 5 groups as x-xxxx-xxxxx-xx-x. Some restrictions are applied to the ID number: the first group cannot be 0 or 9, the first 2 digits of the second group cannot be larger than 77, and the last group needed an extra calculation to calculate for the checksum value. In this case, the most common pattern that we found the ID number is separated by "-" so we can use the regular expression to represent the citizen ID as shown in Fig. 2. As it is the most common pattern, we can identify that this pattern has a strong possibility to be an ID number.

The scanner basically scans through all the input text and

```
\b[1-8]{1}[\-|\s]{1}([0-6]{1}\d{1}|[7]{1}[0-7]{1})\d{2}[\-|\s]{1}\d{5}[\-|\s]{1}\d{2}[\-|\s]{1}\d{1}\b
```

Fig 2: Regular expression of Thai Citizen ID

discovers any 13-digit number that exactly matches on the regular expressions the system provides. The result is given as an array list of the category of the PII, the location, and the accuracy score. The result is passed to the consent checker to check the consent status of that document files or web pages containing PII via the consent management system.

3) Consent Checker

Consent status discovery is another crucial feature of privacy management software controlling the PII. We also proposed a dynamic checking algorithm that is applicable to the consent management system where the list of document files having consent from the data subject is collected. In this paper, we assume that the consent management system is a separate system where all document files and their consent status are stored. Figure 3 displays the process on how the consent status is checked and how final PII data is stored in the PII inventory.

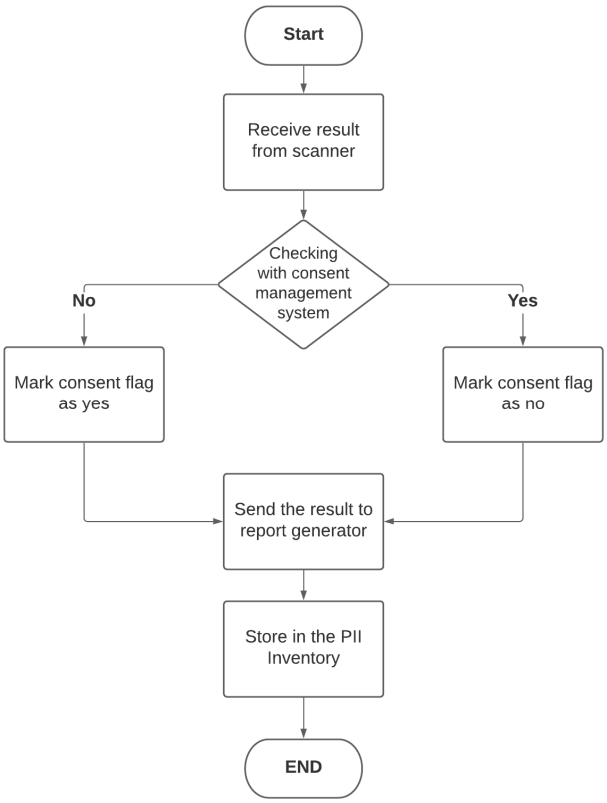

Fig 3: Consent checking process and PII Inventory data flow

As shown in Fig.3, the checker checks whether the PII detected already got consent from the data subject. Consent checker uses the data sources, and file name acquired from the scanning process to check the existing of that file in the consent management system. The consent checker uses a flag to mark whether the PII consent from the data subject by sending API to web service requests for consent checking automatically. We assume that files containing PII that have got consent from data subjects are stored in the consent management system. On the other hand, if the consent checker cannot find the file containing PII inside the consent management system, those PII are not consent from the data subject and they need to be taken care of. If there is a file that matches with the input data, the consent checker will mark that file with 'yes' flag otherwise the data will be marked 'no'. Then, the results will be stored in the PII inventory.

C. PII Inventory

PII inventory is responsible for storing the scanning result and the scanning report which can indicate the location of the sources of PII discovered (URL or directory), the data owner who has the right to access that data sources, the data subject who owns the information, and the consent status that got from the data subject. The inventory provides insight to sensitive data and enable the PII traverse for the data sources and data owner. After the consent checker checks files whether they are consented from data subject, the system stores the record in the PII inventory. We classified the report according to the severity and category of PII so that the data owner can prioritize and manage to get consent of the unconsented data effectively.

IV. EXPERIMENT

This section explains the experiment setting, result presentation, and the comparative result scanning of our system and traditional checking. For the initial stage of our implementation, we aim at testing the functionality of our system to substantiate that our A2PI is able to scan and discover PII from identified sources and the system can check their consent status before all scanned results are systematically retained in the PII inventory. Experiment Setting and Result Presentation

We conduct the experiment on Ubuntu Server 20.04 using Python language to create a scanning service, consent checker, and the GUI interface which serves on the web server. The PII inventory and consent management system are developed in MySQL. The scanning service will be called by users and scan the selected data source. Scanning service uses pattern-based recognizer with the analyzer engine to discover PII entities in the text. Most of the scanning systems are developed from Microsoft's Presidio. In our system, the AP2I server is installed with SSL certificate to enable secure communication between our PII tool and remote devices. Hence, all data transmitted from a remote device or workstation is encrypted based on HTTPS protocol.

For our initial experiment, we developed the system prototype and conducted the test. Specifically, the scanning service is a core module we ran the test. As for the simple test, after the scanning process is successful, the scanning results with its confident score are basically presented on the terminal as shown in Figure 4. Here, the administrator is able to view the results via system console.

```
pipenv run python test.py
[2020-11-15 07:24:12,500][presidio][INFO]Loaded recognizer: ThPhoneRecognizer
[2020-11-15 07:24:12,501][presidio][WARNING][ThPhoneRecognizer]. NLP artifacts were not provided
[type: PHONE_NUMBER, start: 43, end: 54, score: 0.5]
```

Fig.4. Scanning Results

The confidence score for the matched PII is calculated based on the rank of the regular expression in the recognizer. A score between 0 -1 that reflects the likelihood a model has correctly matched a text or word with the pattern. The most common pattern which can easily differentiate PII data from surrounding text is indicated as a strong pattern and the score will be set to 0.7, 0.5, and 0.01 based on the confidence of the regular expression pattern. Some recognizer that needs extra calculation such as Thai citizen ID or credit card number, the confidence score is set to 1 which is the highest automatically if it satisfied the condition which guarantee that it is a PII data in that category otherwise the confidence score is set to 0.

The result will also be used to generate the report where the consent status is automatically checked with consent management system and presented in the report. Figure 5 displays an example of the report. The report shows a scanning result that is also stored in PII inventory including the data owner, location of the source, and the information of PII instances found in the document. Based on the report

generated, the administrator realizes the location of PII sources and how many files have no consent so that further handling to get consent is required.

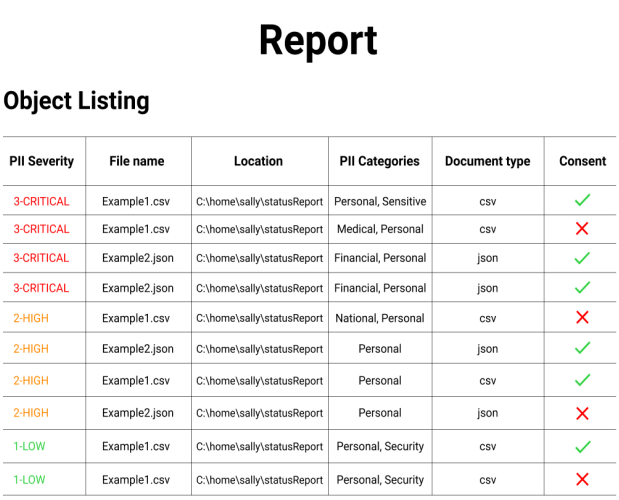

Fig 5: Report showing the detailed scanning result

The report is generated in the form of HTML file which can be opened on web application created by Flask, a python web framework. It is flexible for real-time update on the scanned result list and consent flag in the PII inventory. Any users authorized to submit the scan requests to our AP2I server can view the scan and generate the reports.

List of PII instances stored in the PII inventory is the final output of our system. Figure 6 illustrates the PII Inventory screen shot. In the PII inventory, there is an incremental update on the list if there is any new scanned result. The list of PII scanning results can be also exported as .txt file for further analysis. The inventory maintains a detailed index of all PII instances detected across all batch scans. From this inventory, the user can get insight to the details of PII-related information and can be used to integrate with PII analytics tools. Also, this inventory can be served as a primary source for supporting data breach or data loss prevention mechanism.

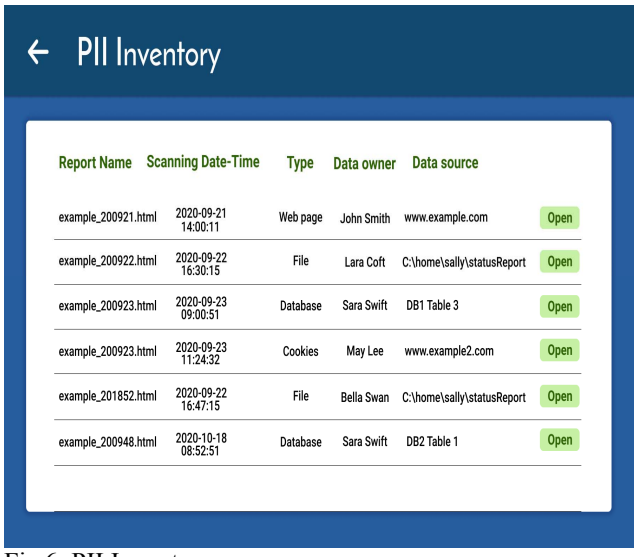

Fig 6: PII Inventory

A. *Comparative Scanning Results*

To substantiate the practicality and efficiency of our AP2I system, we conducted the experiments on computer with an Intel Core I5-7200U using 8 GM of RAM running on Ubuntu server 20.04 to measure the accuracy and scanning performance of our AP2I and manual checking. In the manual checking, a volunteer was asked to search and count the PII shown in the target csv files. In the test, 20 csv files containing PII data in various categories were used to be scanned by our system and manual checking. Each csv file contains about 300 PII instances in average. Table 1 shows the accuracy and speed of both ways.

TABLE I. ACCURANCY AND SPEED COMPARISON BETWEEN OUR AP2I AND MANNUAL CHECKING

	Average Speed	Accuracy
Traditional way (Manual)	2 minutes 36 seconds	98%
AP2I System	3.20 seconds	100%

As shown in Table 1, our AP2I system outperforms the manual checking in both speed and accuracy. The benefit of our tool becomes more obvious when there are a high number of files to be scanned. Especially, the tool is significantly useful for supporting a large batch scan that analyzes an entire storage or device at once.

V. CONCLUSION

We have proposed the PII scanning and discovery system that automatically and adaptively scan and discover PII in any endpoints systems. It also checks consent status of the data sources to provide the insight PII management. We also enhance the accuracy of the scanner by customizing the recognizer to be able to work with Thai PII data sources as to comply Thailand's PDPA. This schema simplifies data owner's tasks as well as their performance. Finally, we conducted the experiment to show the efficiency of our AP2I system.

For future work, we will work on the parallel processing of multiple instances to decrease the bottleneck that might happen during the convertion process and optimize NLP model to improve performance of the tool. Also, the tool will be implemented to make it easy for the data owner to manage the consent of the existing files in their company. To render more functionality in serving PII scanning on image document, we will consider NLP and Optical character recognition (OCR) technique for the PII recognition. In addition, the access control mechanism will be provided on PII inventory to guarantee the privacy of the PII. Our PII inventory will be further extended to support RESTFUL web service that allows other systems to validate or call the PII. Finally, we plan to extend our implementation for deploying our system to support PII scanning in the cloud where most organizational data are outsourced.

REFERENCES

[1] A. Shuba, A. Le, E. Alimpertis, M. Gjoka, and A. Markopoulou. Antmonitor: System and applications. arXiv:1611.04268, 2016.

[2] A. Razaghpanah, N. Vallina-Rodriguez, S. Sundaresan, C. Kreibich, P. Gill, M. Allman, and V. Paxson. Haystack: A multi-purpose mobile vantage point in user space. arXiv:1510.01419v3, Oct. 2016.

[3] J. Ren, A. Rao, M. Lindorfer, A. Legout, and D. Choffnes. Recon: Revealing and controlling pii leaks in mobile network traffic. In Proc. of the 13th Annual Int. Conf. on Mobile Systems, Applications, and Services (MobiSys), volume 16, New York, NY, USA, 2016.

[4] Sharleen Joy Y. Go; Richard Guinto; Cedric Angelo M. Festin; Isabel Austria; Roel Ocampo; Wilson M. Tan, An SDN/NFV-Enabled Architecture for Detecting Personally Identifiable Information Leaks on Network Traffic, In Proc. Of. International Conference on Ubiquitous and Future Networks, Croatia, 2-5 July, 2019.

[5] Y. Liu, H. H. Song, I. Bermudez, A. Mislove, M. Baldi, and A. Tongaonkar, "Identifying personal information in internet traffic," in Proceedings of the 2015 ACM on Conference on Online Social Networks, COSN '15, (New York, NY, USA), pp. 59–70, ACM, 2015.

[6] A. Mrabet, M. Bentounsi, P. Darmon, SecP2I A Secure Multi-party Discovery of Personally Identifiable Information (PII) in Structured and Semi-structured Datasets, In Proc. Of IEEE International Conference on Big Data, Los Angeles, CA, USA,9-12 December, 2019.

[7] Jiaju Huang; Bryan Klee; Daniel Schuckers; Daqing Hou; Stephanie Schuckers, Removing Personally Identifiable Information from Shared Dataset for Keystroke Authentication Research, In Proc. Of 2019 IEEE 5th International Conference on Identity, Security, and Behavior Analysis (ISBA), Hyderabad, India, 22-24 January, 2019.

[8] Fatemeh Alizadeh; Timo Jakobi; Alexander Boden; Gunnar Stevens; Jens Boldt, GDPR Reality Check - Claiming and Investigating Personally Identifiable Data from Companies, In Proc. Of 2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), Genoa, Italy, 7-11 September 2020.

[9] CUSpider. https://www.columbia.edu/acis/security/ spider/ [Online]