On Privacy Weaknesses and Vulnerabilities in Software Systems

Pattaraporn Sangaroonsilp
University of Wollongong
New South Wales, Australia
ps642@uowmail.edu.au

Hoa Khanh Dam
University of Wollongong
New South Wales, Australia
hoa@uow.edu.au

Aditya Ghose
University of Wollongong
New South Wales, Australia
aditya@uow.edu.au

2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE) | 978-1-6654-5701-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICSE48619.2023.00097

***Abstract*—In this digital era, our privacy is under constant threat as our personal data and traceable online/offline activities are frequently collected, processed and transferred by many software applications. Privacy attacks are often formed by exploiting vulnerabilities found in those software applications. The Common Weakness Enumeration (CWE) and Common Vulnerabilities and Exposures (CVE) systems are currently the main sources that software engineers rely on for understanding and preventing publicly disclosed software vulnerabilities. However, our study on all 922 weaknesses in the CWE and 156,537 vulnerabilities registered in the CVE to date has found a very small coverage of privacy-related vulnerabilities in both systems, only 4.45% in CWE and 0.1% in CVE. These also cover only a small number of areas of privacy threats that have been raised in existing privacy software engineering research, privacy regulations and frameworks, and relevant reputable organisations. The actionable insights generated from our study led to the introduction of 11 new common privacy weaknesses to supplement the CWE system, making it become a source for both security and privacy vulnerabilities.**

***Index Terms*—Privacy, Vulnerabilities, Threats, CWE, CVE, Software.**

I. Introduction

Technologies and digitalisation are rapidly emerging in our society. People significantly rely on software applications and computing devices in their daily lives, consequently leaving their traceable digital activities, contributions and communications on those digital devices across the Internet [1]. Even when people are not using software, data about their normal life activities may also be collected by software applications through the ubiquity of IoT and GPS devices, surveillance cameras, face recognition apps and so on. Thus, our privacy is under constant threat in this current digital age. In fact, privacy invasions and attacks have been increasing significantly in recent years [2, 3]. For examples, a cyber crime in the U.S. in 2018 exposed 471 million personal records, and a breach of a national ID database in India leaked over 1.1 billion records including biometric information (e.g. iris and fingerprint scans) [4]. Those incidents and threats raise an urgent need for privacy to be deeply integrated into the development, testing and maintenance of software applications.

Although security and privacy are often discussed together, they are *not* the same [5]. Security often refers to protection against the unauthorised access to software applications and the data they collect and store. On the other hand, privacy relates to protection of the individual rights to their personally identifiable information in terms of how those personal data are collected, used, protected, transferred, altered, disclosed and destroyed [6]. For example, security controls are put in place to ensure that only people with credentials have access to a software application in a hospital. However, if anyone with valid credentials can see patient health records using this software, then privacy is not protected. This example demonstrates that security can be achieved without privacy, however security is an essential component for privacy protection.

Cyberattacks, either in the form of security or privacy attacks, are often formed by exploiting *vulnerabilities* or *weaknesses*[1] found in software systems. For instance, the infamous WannaCry ransomware attack exploited a vulnerability in Microsoft Windows systems, while the Heartbleed vulnerability in OpenSSL has made millions of websites and online platforms across the world vulnerable to cyberattacks. To prevent similar attacks, efforts have been put into understanding and publicly disclosing vulnerabilities so that developers can identify and fix them in their software applications. These efforts have resulted in the widely-known CWE and CVE systems [7, 8].

However, there have been very little work (e.g. [9, 10]) in identifying privacy vulnerabilities. A system which specifically records common privacy vulnerabilities does *not* exist yet. Thus, software developers often rely on the CWE and CVE systems to learn about privacy-related weaknesses and vulnerabilities. However, it is not clear to what extent privacy concerns are covered in those systems, and whether privacy receives adequate attention (which it deserves). To answer these questions, we have collected *all* 922 weaknesses recorded in CWE and 156,537 records registered in CVE to date, filtered out non privacy-related records and further analysed the shortlisted records that are privacy-related. We have found only 41 and 157 privacy vulnerabilities in the CWE and CVE systems respectively. The coverage of privacy-related vulnerabilities in both systems is very limited, only 4.45% in CWE and 0.1% in CVE. (**Contribution 1**)

The next questions we aimed to explore are what privacy threats are covered in those privacy-related vulnerabilities in the CWE/CVE systems and if they are adequately cover

[1] The two terms are often interchangeable. Hereby, we will use vulnerabilities to refer to both of them.

1558-1225/23/$31.00 ©2023 IEEE
DOI 10.1109/ICSE48619.2023.00097

the privacy threats raised in both research and practice. To answer these questions, we have conducted an explanatory study on the privacy engineering literature, privacy standards and frameworks (e.g. ISO/IEC 29100), regulations in different jurisdiction, including the European Union General Data Protection Regulation (EU GDPR), California Consumer Privacy Act (CCPA), Health Insurance Portability and Accountability Act (HIPAA), Gramm-Leach Bliley Act (GLBA), the U.S. Privacy Act (USPA) and the Australian Privacy Act (APA), and relevant reputable organisations (e.g. OWASP [11] and Norton [12]). This explanatory study informed us to develop a taxonomy of common privacy threats that have been raised in research and practice. The taxonomy is built upon the existing well-known privacy threats taxonomy [13]. Multiple raters/coders then examined all 41 and 157 privacy vulnerabilities in the CWE and CVE systems, and mapped them to this taxonomy. The Cohen's Kappa coefficient, used to measure the inter-rater agreement, was obtained at 0.874 and 0.875 for the CWEs and CVEs respectively, an almost perfect agreement, suggesting the strong reliability of the classification. We found that the existing privacy weaknesses and vulnerabilities reported in the CWE and CVE systems cover only 13 out of 24 common privacy vulnerabilities raised in research and practice. Many important types of privacy weaknesses and vulnerabilities are *not* covered such as improper personal data collection, use and transfer, allowing unauthorised actors to modify personal data, processing personal data at third parties, and improper handling of user privacy preferences and consent. (**Contribution 2**)

These actionable insights led to our proposal of 11 new common privacy weaknesses to CWE[2]. These new CWE entries cover the areas of privacy threats that have been raised in research and practice but do not exist in CWE yet. To further confirm the relevance and validity of our proposal, we extracted real code examples from software repositories that match with the new CWEs. Our contribution follows the CWE's true spirit of a community-developed list, and will enhance the CWE system to serve as a common language and baseline for identifying, mitigating and preventing not only security but also privacy weaknesses and vulnerabilities. (**Contribution 3**)

The remainder of the paper is structured as follows. Section II discusses related existing work in security and privacy vulnerabilities in software applications. The identification of privacy-related vulnerabilities in CWE and CVE is presented in Section III. Section IV discusses the taxonomy of privacy threats and how the privacy-related vulnerabilities in CWE and CVE systems cover those privacy threats. Section V presents a new common privacy weakness proposal. The threats to validity of our study are discussed in Section VI. We conclude and discuss future work in Section VII. Finally, we provide the details of a replication package and instructions on how to access it in Section VIII.

II. Related Work

Several systems have been established to standardise the reporting process and structure of common vulnerabilities (e.g. CWE [7], CVE [8] and OWASP [11]). However, identifying the root causes of the reported vulnerabilities is still a time-consuming and expertise-required process [14]. Recent work employed information retrieval, data mining, natural language processing, machine learning and deep learning techniques to characterise vulnerabilities reported in CVE and CWE systems [14, 15, 16]. Li et al. developed a vulnerability mining algorithm to characterise software vulnerabilities [15]. It first created a Vulnerability Knowledge Discovery Database (VKDD) by extracting content from CWE, CVE and National Vulnerability databases. It then employed the semantic model to select terms that are relevant to software vulnerabilities from the VKDD. The association and classification rules were later determined to identify and classify the vulnerabilities. Liu et al. built a classification model for detecting vulnerable functions in source code [16]. The deep neural network with bidirectional long short-term memory (LSTM) was employed to learn high-level representations of the program's abstract syntax trees. The study also proposed a fuzzy-based oversampling method to mitigate the class imbalance between vulnerable and nonvulnerable code. However, privacy vulnerabilities were not addressed in those previous work.

Recent work have also studied and made use of the CWE and CVE systems. For example, the work in [17] collected CVE records with their associated CWEs and code commits. The collected information was then analysed to produce insightful metadata such as concerned programming language and code-related metrics. This work can be applied in multiple applications related to software maintenance such as automated vulnerability detection and classification, vulnerability fixing patches analysis and program repair. Galhardo et al. proposed a formulation to calculate the most dangerous software errors in CWE [18]. They used this formulation to identify the top 20 most significant CWE records in 2019. Again, these prior work only focuses on security vulnerabilities.

Yang et al. introduced a framework to detect privacy leakage in mobile applications [9]. This study identified several common privacy vulnerabilities in Android such as unintended sensitive data transmission and local logging. Ma et al. discussed a privacy vulnerability in mobile sensing networks which collect mobility traces of people and vehicles (e.g. traffic monitoring) [10]. Although these networks receive anonymous data, it was proven in the study that these data can identify victims. These studies have confirmed the occurrence of privacy vulnerabilities in multiple types of software systems (e.g. web/mobile applications and sensing networks). However, most of the existing studies only focused on security concerns when investigating software vulnerabilities, thus overlooked privacy-related concerns in many contexts.

A number of studies have proposed state-of-the-art taxonomies of privacy threats (e.g. [19, 20, 21]). Solove proposed four categories of privacy threats which cover harmful activities

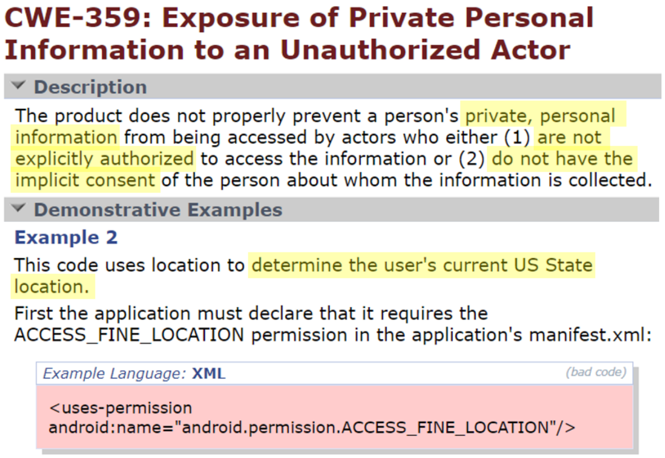
[2] We chose CWE instead of CVE since CVEs specify unique vulnerabilities detected in specific software systems and application, while CWEs are at a more abstract, generic level.

that can violate privacy of individuals [19]. Another well-known standardised taxonomy of privacy threats in software known as LINDDUN was first proposed in [20]. LINDDUN is a model-based technique used to discover the privacy threats in a system [21]. It models the system as a data flow diagram (DFD), maps DFD elements to privacy threat categories, and then identifies and documents the privacy threat type and the DFD element type. Although LINDDUN can be applied to any general-purpose software systems, it requires the details of system description to construct the DFD for threat analysis.

Two renowned organisations have also worked on privacy framework and privacy threat modelling. National Institute of Standards and Technology (NIST) proposed a privacy framework which focuses mostly on providing high-level guidelines that organisations can follow to govern and control privacy risks [22]. It does not address the privacy threats specifically in software systems. Thus, we did not include this framework in our study. The MITRE Corporation has been developing a privacy threat modelling method [23]. However, the MITRE work took a different approach: they identified privacy attacks from the Federal Trade Commission (FTC) and Federal Communications Commission (FCC) closed cases. By contrast, we identified privacy threats from existing privacy engineering literature, privacy regulations/frameworks, and industry sources. This results in several major differences in the two taxonomies in terms of scope of privacy threats, coverage of privacy threats, and classification of threats and (sub-)categories. In addition, our work goes even beyond providing a taxonomy. We provided a detailed, concrete (CWE-ready) description of the common privacy vulnerabilities including how they occur and their consequences, how to detect them and mitigate them, and demonstrative examples. These provide software engineers with more concrete information of privacy vulnerabilities.

III. Identifying privacy-related vulnerabilities in CWE and CVE

Common Weakness Enumeration (CWE) and Common Vulnerabilities and Exposures (CVE) are two well-known systems for publicly known weaknesses and vulnerabilities in software and hardware. The CWE system identifies common categories of flaws, bugs and other errors found in software and hardware implementation, code, design or architecture that could be vulnerable to attacks [7]. The CWE system has three views: i) by research concepts, ii) by software development and iii) by hardware design. Our study focuses on the research concepts and software development views as they are related to software applications. A CWE record consists of a description, relationships to other CWE records, demonstrative examples, mitigations and other relevant attributes. The interested parties can use this information to identify a weakness in their software systems and applications. For example, CWE-359[3] describes a weakness that exposes private personal information to an unauthorised actor (see Figure 1). It also provides demonstrative examples, one of which is a code fragment that exposes a user's location.



CWE-359: Exposure of Private Personal Information to an Unauthorized Actor

▾ Description

The product does not properly prevent a person's private, personal information from being accessed by actors who either (1) are not explicitly authorized to access the information or (2) do not have the implicit consent of the person about whom the information is collected.

▾ Demonstrative Examples

Example 2

This code uses location to determine the user's current US State location.

First the application must declare that it requires the ACCESS_FINE_LOCATION permission in the application's manifest.xml:

Example Language: XML (bad code)

```
<uses-permission
android:name="android.permission.ACCESS_FINE_LOCATION"/>
```

Fig. 1. A screenshot highlights privacy-related information in CWE-359.

CVE is a catalogue of cybersecurity vulnerabilities that may exist in software products, applications and open libraries (e.g. Skype, Mozilla Firefox and Android). These vulnerabilities are reported by organisations that have partnered with the CVE program. Each CVE record describes the details of a vulnerability and specifies affected version of a software, thus it is more specific comparing to CWE records. For example, CVE-2000-1243[4] refers to a privacy leak in version 3.04 of Dansie Shopping Cart which sensitive information such as user credentials were sent to an e-mail address controlled by the product developers.

In the absence of a system which specifically records common privacy vulnerabilities, software engineers and other interested parties often rely on the CWE and CVE systems for privacy weaknesses and vulnerabilities (like the one in Figure 1). However, both CWE and CVE target at cybersecurity, and although security and privacy are often discussed together, they are not the same. Security vulnerabilities are often exploited by unauthorised access to perform malicious actions in software applications. By contrast, privacy vulnerabilities may lead to violations of the individual rights to their personally identifiable information in terms of how those personal data are collected, used, protected, transferred, altered, disclosed and destroyed. Hence, we have explored to what extent privacy concerns are covered in the CWE and CVE systems, and whether privacy receives adequate attention which it deserves.

A. Approach

The privacy vulnerability identification process (see Figure 2) consists of the following steps: (i) obtaining the CWE and CVE lists, (ii) determining a list of keywords and performing a keyword search, (iii) identifying privacy-related criteria and annotating the CWE and CVE records, and (iv) performing privacy vulnerability analysis.

[3] https://cwe.mitre.org/data/definitions/359.html

[4] https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2000-1243

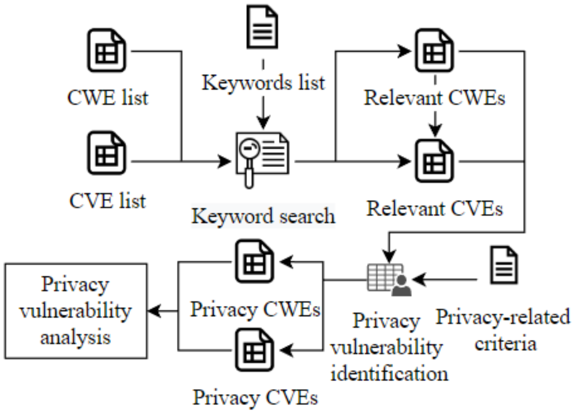


Fig. 2. Privacy vulnerabilities identification process.

To identify the privacy vulnerability in CWE and CVE lists, we first downloaded the whole CWE records in the research concepts and software development views[5] and CVE records[6] from their websites. We examined all the attributes of 922 weaknesses in both research concepts and software development views in the CWE to date. The CVE list contains 156,537 records to date, all of which were examined in our study. We examined all of the following attributes in each CVE record: name (i.e. CVE-ID), status, description, references, phases, votes and comments.

After obtaining both lists, we then performed keyword searches to filter out the CWE and CVE records that do not have privacy-related keywords. We used a search function in Microsoft Excel to examine the keywords in those records. A set of keywords consists of 37 words categorised into 4 groups as follows:

- Group 1: general terms that relate to weaknesses and vulnerabilities in privacy. The terms include privacy, violation, leak and leakage (4 keywords). The term *privacy* generally appears in the CWE and CVE that reported privacy weaknesses and vulnerabilities. On top of that, we also include the terms *violation, leak and leakage* as they are alternatively used to express concerns in the context of privacy (see CWE-359[7] for more details).
- Group 2: terms used to refer to personal data or personally identifiable data. They could be sometimes used interchangeably across regulations, standards and industry sources. The keywords in this group include personal information, personal data, sensitive information, sensitive data, private information, private personal information, personally identifiable information, PII, protected health information, PHI, health information and health data (12 keywords).
- Group 3: terms that relate to relevant privacy and data protection regulations/standards/frameworks. In this group, we select a set of specific well-known and widely-adopted data protection regulations and privacy frameworks mentioned in the CWE records, which includes General Data Protection Regulation, GDPR, California Consumer Privacy Act, CCPA, Health Insurance Portability and Accountability Act, HIPAA, Gramm-Leach Bliley Act, GLBA, Safe Harbor Privacy Framework, ISO/IEC 29100 (10 keywords). In addition, we also include the general terms to ensure that we cover unseen regulations and frameworks which are regulation, data protection, privacy act, privacy framework and privacy standard (5 keywords).
- Group 4: terms that are frequently seen in the privacy policies and literature when discussing personal data protection and user privacy. These include right(s), consent, opt in/opt-in, opt out/opt-out, preference and breach (6 keywords).

We acquired 185 CWE and 1,088 CVE records that contain at least one of the specified keywords. Next, we manually examined each of those records to identify privacy vulnerabilities. A vulnerability is considered as privacy-related if it satisfies one of the following criteria:

1) A weakness or vulnerability involves with any processing of personal data (e.g. collection, use, storage, transfer, alteration, erasure and disclosure).
2) A weakness or vulnerability which may lead to the violations of the individual rights. To extract the individual rights, we first selected a range of well-established data protection and privacy regulations in different domains such as governments, businesses, healthcare and finance (i.e. EU GDPR, CCPA, HIPAA, GLBA, USPA, APA). These regulations have been widely enacted in country- and regional-level, hence they are well respected by organisations worldwide. In each regulation, we went through each article to look for the individual rights of data subjects/patients/consumers. Once we found the individual right, we added it into our list[8].

We went through the shortlisted 185 CWE and 1,088 CVE records to determine the vulnerabilities that meet the above criteria. In addition, we have found that the National Vulnerability Database (NVD) had done some mapping between CVEs and CWEs. Hence, once we have identified privacy-related CVEs, we used this mapping and applied the above criteria to identify additional privacy-related CWE records.

B. Results

We identified 41 and 157 privacy vulnerabilities in the CWE and CVE records respectively. The first 28 privacy-related CWE records were found after the keyword search and manual examination steps. The additional 13 privacy-related CWE records were later added after being identified by the privacy-related CVE records. They cover a wide range of privacy concerns in software applications such as missing personal data protection, improper access control, insufficient credentials protection and personal data exposures. These weaknesses and

[5] The CWE data is available at https://cwe.mitre.org/data/downloads.html

[6] The CVE data is available at https://cve.mitre.org/data/downloads/index.html

[7] https://cwe.mitre.org/data/definitions/359.html

[8] See the file named *"14-rights"* in the data folder in the replication package for the complete list of individual rights and their relevant data protection regulations/privacy acts.

vulnerabilities are not only related to security, but also affect privacy of individuals.

We discuss here a few examples of CWE and CVE records that were classified as privacy vulnerabilities or weaknesses and refer the reader to [24] for a full list of them. CWE-359[9] refers to the exposure of private personal information to an unauthorised actor. Private personal information here includes social security numbers, geographical location, financial data and health records. In addition, this CWE also mentions relevant data protection regulations and privacy acts such as GDPR and CCPA.

Another example is CVE-2020-13702[10] which refers to the rolling proximity identifier used in the Apple/Google exposure notification API beta through 2020-05-29. This vulnerability enables attackers to evade Bluetooth Smart Privacy due to a secondary temporary UID through tracking individual device movement using a Bluetooth LE discovery mechanism. This is a privacy vulnerability since it concerns user's location and relates to the processing of user's location. There are many cases where the reported vulnerabilities are *not* specifically privacy-related. For example, CWE-78[11] enables the attacker to execute arbitrary commands on the operating systems, leading to unauthorised access to operating systems. However, this is not specifically a privacy vulnerability since it does not involve personal data, personal data processing or individual rights.

The coverage of privacy-related vulnerabilities in both CWE and CVE records is quite limited: 4.45% in the CWE system and 0.1% in the CVE system.

IV. Common privacy threats in software applications

This section investigates how those privacy-related vulnerabilities identified in CWE and CVE (see Section III) address common privacy threats in software applications. We first discuss a taxonomy of common privacy threats that we have developed based on an explanatory study of the literature. We then report the threats that have not been covered by the existing privacy-related vulnerabilities in CWE/CVE.

A. Explanatory study

To identify common privacy threats in practice, we performed an explanatory study on the literature of the following three groups: existing privacy software engineering (SE) research, well-established data protection regulations and privacy frameworks, and additional reputable resources. The details of this process are described below.

1) Privacy engineering research: Privacy engineering has attracted an emerging area of research in SE [25]. Our study in this area followed a systematic literature review process proposed by [26] to retrieve relevant papers and conduct a literature survey. This process consists of three phases: planning, papers selection and extracting & reporting.In the planning phase, we defined our study goal which is to identify privacy threats addressed in the privacy engineering research. Our research question is "What are privacy threats caused by software developers and/or relevant parties that were addressed in privacy engineering research?". To achieve the study goal, we developed a review protocol to determine the scope of our study. The protocol consists of four tasks: (i) determining a search keyword and time scope, (ii) selecting SE publication venues, (iii) determining exclusion and inclusion criteria, and (iv) determining a set of questions to identify the privacy threats in the papers. In the papers selection phase, we conducted a search process and applied inclusion and exclusion to the retrieved papers. Finally, we analysed and identified privacy threats in the selected papers and reported the results.

*a) **Search keyword and time scope**:* We used a search keyword *"privacy"* to search in title, abstract and keywords fields of papers in the selected publication venues. Only one search keyword was used as we already performed a search in the specific SE publication venues, thus we aimed to get all the papers that address privacy. In addition, the papers from these venues are peer-reviewed by experienced researchers in privacy and SE area, hence their significant contributions and quality are well received by the research community. We also determined to search for papers that were published in the past 20 years (2001 - 2020). This task was automatically done by the search functions in the academic databases (i.e. IEEE Xplore, ACM Digital Library, ScienceDirect, SpringerLink and Scopus). We have found 1,434 papers related to privacy during the period of 2001 to 2020 (see Figure 3).

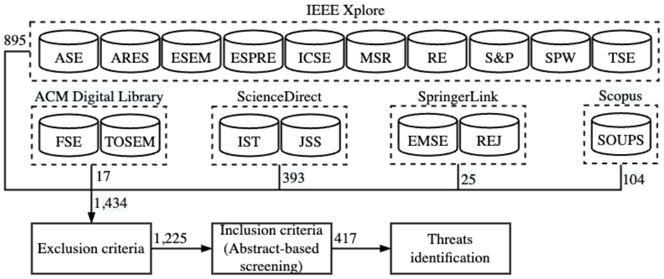


Fig. 3. Number of papers included in each step.

*b) **Software engineering publication venues**:* As privacy has also been widely addressed in other fields of study (e.g. law and social sciences), we scope down the search to get a reasonable number of papers in SE by selecting seventeen highly recognised SE publication venues consisting of 7 conferences, 6 journals, 2 symposiums and 2 workshops. These venues focus on various disciplines in SE, which make them proper candidates for representing privacy in multiple areas. Many of these venues were included in the existing papers that conducted systematic literature review in SE area (e.g. [27, 28, 29]). The full list of venues, along with the number of papers found in each, is included in the replication package.

[9] https://cwe.mitre.org/data/definitions/359.html

[10] https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-13702

[11] https://cwe.mitre.org/data/definitions/78.html

*c) **Exclusion and inclusion criteria:*** We manually performed an inclusion and exclusion task to ensure that the papers retrieved from the automated search process satisfy our study scope. We initially determined a set of inclusion and exclusion criteria to filter out the papers that are irrelevant to our goal. We exclude the papers that either contain insufficient, incomplete or irrelevant information (e.g. records that are journal/conference/workshop introduction, information from program chairs and summaries of keynotes), are duplicate *OR* are secondary or tertiary studies (e.g. existing systematic literature review papers) and posters. For the inclusion criteria, we include the papers which their contribution is related to privacy *AND* software development.

After applying the EC, we excluded 209 out of 1,434 papers. 1,225 papers were passed to the next step. We then applied the IC to the abstracts of the papers. If the papers satisfy both IC, they are included in our study; otherwise they are excluded. Finally, 417 papers were included in our study[12].

*d) **Privacy threats identification:*** To identify privacy threats, we examine the research studies reported in those papers, and shortlist the ones that focus specifically on privacy vulnerabilities and attacks from this list. The papers were analysed by asking the following questions:

- What is a cause of privacy-related problem in software that has been raised in the paper? This cause must involve personal data and is harmful to data subjects.
- Is the identified cause caused by software developers, data controllers/processors, organisations or external parties? This question helps us focus on the privacy threats that are not caused by users. There are papers that investigate user perceptions towards different privacy perspectives (e.g. user perceptions of online behavioural advertising/smart home devices and user confidence in using smartphones). We do not include the privacy threats that are caused by users in this study as it is out of our scope.

We manually examined the selected papers, and added privacy threats identified in the papers into a threat list. During the investigation, if we find a threat that is already included in our list, we annotate the paper with the existing threat and proceed to the next paper, otherwise we add a new threat into the list. Finally, we have found 179 papers that address privacy vulnerabilities and attacks in software development[13]. From these studies, we have identified a range of privacy threats in various applications (e.g. mobile/web applications, mobile sensors, smart home devices, network communications and privacy policy compliance). These will be discussed in details in Section IV-B. The rest of the papers either address the threats caused by users or do not address privacy vulnerabilities or attacks in software systems.

2) Privacy regulations and frameworks: We have studied 7 well-established data protection and privacy regulations and frameworks (i.e. GDPR [30], CCPA [31], HIPAA [32], GLBA [33], USPA [34], APA [35] and ISO/IEC 29100 [6]). We have found that these regulations and frameworks focus mainly on the privacy threats related to the rights of individuals to control and be informed of their personal data processing.

3) Additional sources: We have also included a range of relevant reputable organisations (e.g. OWASP and Norton) on this topic. These sources (e.g. [11, 36]) cover the recent trends of privacy risks and attacks. For example, OWASP identified 20 privacy risks in web applications [37] such as personal data leak, personal data stolen through common cyberattacks (e.g. cross-site scripting and broken session management). Norton also identified a number of social engineering cyberattacks that are privacy-related such as phishing and keystroke logging attacks [12].

In the next section, we will discuss in details the common privacy threats that we have found in our explanatory study of the literature.

B. A taxonomy of common privacy threats

We have built a taxonomy of common privacy threats upon the well-established privacy threats taxonomy described in [13]. The taxonomy was originally proposed by [19]. It covers privacy violations of individuals that are caused by physical activities (e.g. a newspaper reports the name of a rape victim, or a company sells its members' personal information despite promising not to do so). Later, Stallings adapted the concept from [19] to build the taxonomy of privacy threats for information systems. However, Stallings's taxonomy covers privacy threats in a generic and rather abstract level. Thus, we tailored this taxonomy and refined it into a more concrete version that addresses privacy threats in SE. The taxonomy consists of four categories of privacy threats: information collection, information processing, information dissemination and invasions (see Figure 4). Each of these groups contains different subcategories covering relevant harmful privacy threats. The yellow boxes in Figure 4 represent the categories and subcategories included in the original taxonomy.

After identifying the privacy threats in the explanatory study, we classified those threats into two groups: vulnerabilities and compliance. Privacy vulnerabilities refer to technical issues, flaws or errors that lead to privacy exploits in software applications. Compliance addresses the privacy threats related to the individual rights and the governance of personal data. We then expanded the Stallings's taxonomy by mapping the privacy threats in the vulnerabilities group into their relevant subcategories in the original taxonomy (i.e. blue boxes in Figure 4). However, the privacy threats in the compliance group have not been addressed in any groups in the original taxonomy. Thus, we propose the compliance group as an extension to the original taxonomy (see Figure 5). This group is discussed in details in Section IV-B5. The full taxonomy is available at [24].

In our taxonomy, we classified 24 privacy vulnerabilities into 7 subcategories. In the classification process, we mapped a privacy threat into the most relevant subcategory based on the description of each subcategory explained in [13]. The

[12] The bibliographic data of those papers are available in [24].

[13] See the file named *"15-papers-by-threat-categories"* in the data folder in the replication package.

categories, subcategories and their relevant privacy threats are described below. We also provide several examples of sources where the privacy threats were raised or discussed in each category[14].

1) ***Information collection:*** (Sources: [6, 30, 37, 38, 39, 40]) This category concerns privacy threats that occur when collecting personal data from individuals. The *surveillance* subgroup covers vulnerabilities existing in the way software collects personal data such as watching individuals through cameras or CCTVs, listening to individuals, or recording individuals' activities. The privacy threat related to this subgroup occurs when personal data is collected without consent or permissions from individuals (e.g. via mobile sensors).

2) ***Information processing:*** (Sources: [20, 32, 33, 41, 42, 43, 44, 45]) This category covers vulnerabilities related to the use, storage, and manipulation of the collected personal data. It contains four subgroups: identification, insecurity, secondary use and exclusion. The *identification* subcategory addresses a privacy threat that aggregates personal data from various sources and uses it to identify individuals.

The vulnerabilities in *insecurity* subcategory are caused by improper protection and handling of personal data. There are multiple forms of this vulnerability type such as lacking mechanisms to protect personal data, allowing unauthorised actors to access or modify personal data, track individual users and transferring personal data without protection. Personal data is sometimes required to be processed at third parties. This poses a privacy vulnerability since the third parties may apply weaker level personal data protection than the source does. In addition, this subcategory refers to the vulnerabilities where mechanisms to protect personal data are in place but they are not appropriate. Different types of personal data require different methods/techniques and levels of protection. Thus, appropriate protection mechanisms should be used to protect personal data against potential risks. This vulnerability type is often refined into improper techniques/methods and insufficient levels of protection (e.g. weak encryption).

The *secondary use* subcategory refers to the use of personal data for other purposes without consent or not following user privacy preferences. Personal data used or transferred without user permissions, or to an unintended destination is a privacy vulnerability. The personal data used without following user privacy preferences can also cause a privacy vulnerability[15].

The *exclusion* subcategory refers to the failure to provide individuals with notice and input for managing their personal data. These vulnerabilities relate to consent which allows the users to express their agreement on the use of their personal data in software applications. We note that consent handling may seem to be a part of compliance, however this subcategory focuses on the malfunctions that cause vulnerabilities in consent handling. User consent is required when the processing of personal data is not required by laws. Users should also be notified when the conditions on the consent are changed. In addition, users should be allowed to modify or withdraw their consent. One example of this privacy vulnerability is in mobile applications where users allow a specific permission to an app, however the permission is overridden in other apps without their consent [45, 46]. Apart from user consent, user privacy preferences are also important. Privacy preferences enable users to personalise how they prefer their personal data to be managed. This privacy vulnerability type can be in two forms: privacy preferences not provided to the users, and users not be able to modify their privacy preferences.

3) ***Information dissemination:*** (Sources: [37, 45, 46, 47, 48]) This category refers to the privacy threats that lead to the disclosure of personal data to public. The *breach of confidentiality* subcategory covers the vulnerabilities that cause personal data leakage by those who are responsible for personal data processing (e.g. software developers, data processors and third parties).

4) ***Invasions:*** (Sources: [12, 20, 37, 49]) This category addresses attacks that directly affect individuals. The *intrusions* subcategory covers vulnerabilities exploited by common privacy attacks in software applications. There are typically four attacks: web applications, phishing, keystroke logging and smart home devices.

5) ***Compliance:*** We used a bottom-up approach to construct a taxonomy for the compliance category. The privacy threats that address the same concerns are grouped into the same subcategory. We classify privacy compliance into the following three subcategories. The first is *not complying with individual rights* (Sources: [6, 20, 30, 31, 32, 33, 34, 35, 37, 50, 51, 52, 53]). There are 15 individual rights identified in our study[16].

The second is *not providing contact details of a responsible person* (Sources: [6, 30]). This subcategory covers cases where software applications do not provide the contact details of a responsible person or a representative who processes personal data. This is a privacy threat as the users do not know whom to be contacted regarding their personal data processing. The third is *improper personal data breach response* (Sources: [6, 30, 32, 37]). When a breach occurs, a responsible person must notify two key stakeholders: concerned individuals and a supervisory authority. The privacy threat is raised if the responsible person does not communicate the breach incident to the concerned users whose personal data is leaked to the public and the supervisory authority who monitors the personal data processing under the individual rights.

Our taxonomy covers the existing vulnerabilities occurred in a range of activities performed in software systems (e.g. personal data collection and processing). These privacy vulnerabilities have been raised in real world software applications and software development processes. The taxonomy was also developed based on an existing comprehensive privacy threats

[14] See the file named *"4-RQ2-explanatory-study-papers"* in the data folder in the replication package for the full list of papers with their associated privacy threats.

[15] For example, CVE-2005-2512 reports a vulnerability which could result in a privacy leak in mail.app in Mac OS 10.4.2 and earlier in which remote images are loaded against the user's preferences when an HTML message is printed or forwarded.

[16] See the file named *"14-rights"* in the data folder in the replication package for the complete list of individual rights.

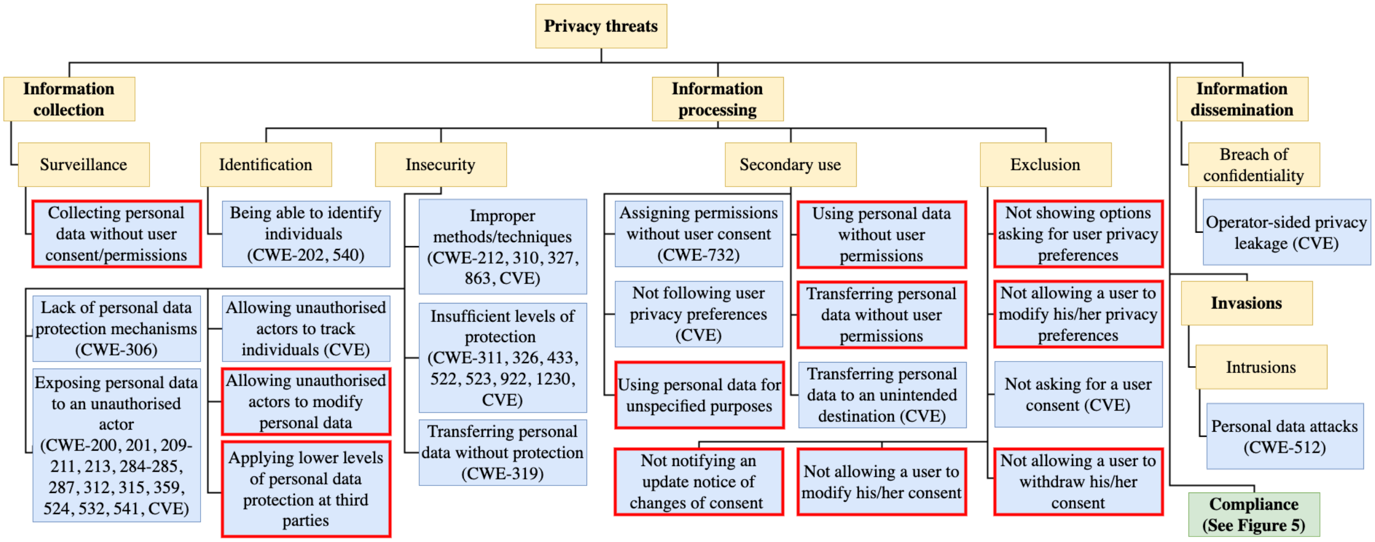

Privacy threats
Information collection
Information processing
Information dissemination
Surveillance
Identification
Insecurity
Secondary use
Exclusion
Breach of confidentiality
Collecting personal data without user consent/permissions
Being able to identify individuals (CWE-202, 540)
Improper methods/techniques (CWE-212, 310, 327, 863, CVE)
Assigning permissions without user consent (CWE-732)
Using personal data without user permissions
Not showing options asking for user privacy preferences
Operator-sided privacy leakage (CVE)
Lack of personal data protection mechanisms (CWE-306)
Allowing unauthorised actors to track individuals (CVE)
Insufficient levels of protection (CWE-311, 326, 433, 522, 523, 922, 1230, CVE)
Not following user privacy preferences (CVE)
Transferring personal data without user permissions
Not allowing a user to modify his/her privacy preferences
Invasions
Intrusions
Exposing personal data to an unauthorised actor (CWE-200, 201, 209-211, 213, 284-285, 287, 312, 315, 359, 524, 532, 541, CVE)
Allowing unauthorised actors to modify personal data
Using personal data for unspecified purposes
Transferring personal data to an unintended destination (CVE)
Not asking for a user consent (CVE)
Personal data attacks (CWE-512)
Applying lower levels of personal data protection at third parties
Transferring personal data without protection (CWE-319)
Not notifying an update notice of changes of consent
Not allowing a user to modify his/her consent
Not allowing a user to withdraw his/her consent
Compliance (See Figure 5)

Fig. 4. Common privacy vulnerabilities in software applications

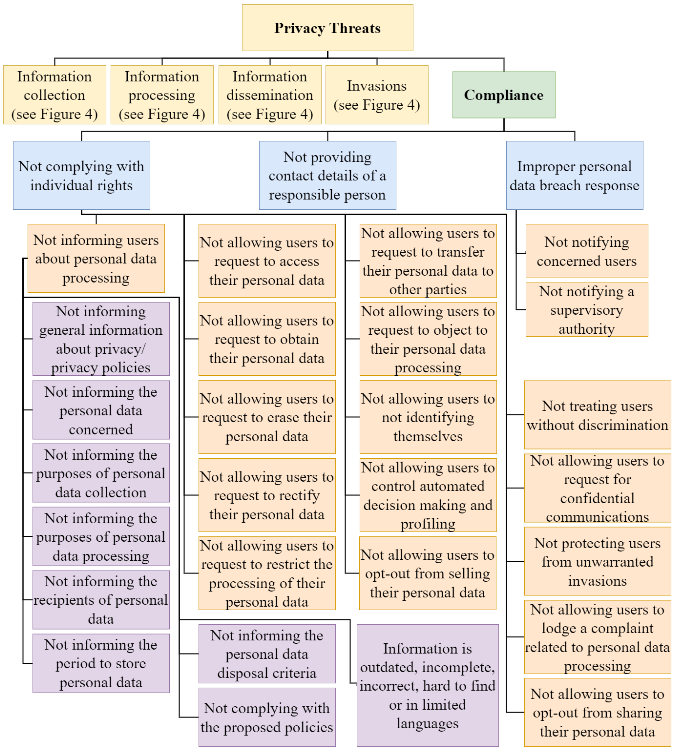

Privacy Threats
Information collection (see Figure 4)
Information processing (see Figure 4)
Information dissemination (see Figure 4)
Invasions (see Figure 4)
Compliance
Not complying with individual rights
Not providing contact details of a responsible person
Improper personal data breach response
Not informing users about personal data processing
Not allowing users to request to access their personal data
Not allowing users to request to transfer their personal data to other parties
Not notifying concerned users
Not notifying a supervisory authority
Not informing general information about privacy/ privacy policies
Not allowing users to request to obtain their personal data
Not allowing users to request to object to their personal data processing
Not informing the personal data concerned
Not allowing users to request to erase their personal data
Not allowing users to not identifying themselves
Not treating users without discrimination
Not informing the purposes of personal data collection
Not allowing users to request to rectify their personal data
Not allowing users to control automated decision making and profiling
Not allowing users to request for confidential communications
Not informing the purposes of personal data processing
Not allowing users to request to restrict the processing of their personal data
Not allowing users to opt-out from selling their personal data
Not protecting users from unwarranted invasions
Not informing the recipients of personal data
Not informing the personal data disposal criteria
Information is outdated, incomplete, incorrect, hard to find or in limited languages
Not allowing users to lodge a complaint related to personal data processing
Not informing the period to store personal data
Not complying with the proposed policies
Not allowing users to opt-out from sharing their personal data

Fig. 5. Common privacy compliance in software applications

taxonomy, and validated with the common vulnerabilities reported in CWE and CVE.

C. Privacy threats covered in CWE/CVE

The next step in our study was to investigate how 41 privacy vulnerabilities in CWE and 157 in CVE (see Section III) address the taxonomy of common privacy threats in Section IV-B. This process consists of the following steps. The first two co-authors (hereafter referred to as the coders) *independently* analysed each of the 41 privacy vulnerabilities in CWE, and classified it into the most relevant privacy threat in the taxonomy. Of the 157 privacy vulnerabilities in CVE, 112 were assigned to a specific CWE by the NVD. These 112 CVEs are automatically classified into the same privacy threat as their associated CWE. The remaining 45 CVEs, which were not assigned a specific CWE[17], were manually classified by our coders. To facilitate the classification step, each coder was provided with a Google Sheet form pre-filled with the privacy vulnerabilities in CWE and CVE and the privacy threats in the taxonomy. The privacy threats were prepared as a drop down list. The coders then selected the vulnerability that is most relevant and best described those CWE and CVE in their view.

We have also employed the Inter-Rater Reliability (IRR) assessment and disagreement resolution processes to ensure the reliability of the classifications. The Cohen's Kappa coefficient is used to measure the inter-rater agreement as it is a perfect measure for a multi-class classification problem with two coders [54]. Once the coders had completed the classifications, the inter-rater agreement was computed. The Kappa agreement values between the coders are 0.874 and 0.875 in CWE and CVE classifications respectively, which both achieve *almost perfect agreement* level [55]. A disagreement resolution was conducted to resolve some small classification conflicts (4 CWEs and 4 CVEs). The coders met, went through them together, discussed and reclassified those vulnerabilities. Thus, the final classification reached the maximum agreement between the coders.

1) Results: We have found that all the 41 CWEs and 157 CVEs together cover 13 vulnerabilities in the taxonomy. They are annotated with the corresponding CWEs and CVEs in Figure 4. For brevity, we do not include the CVE numbers in Figure 4, but they are provided in full in our replication

[17] NVD used two special placeholder names for these: NVD-CWE-noinfo and NVD-CWE-other.

package [24]. Exposing personal data to an unauthorised actor, insufficient levels of protection, and personal data attacks are the top three most addressed vulnerabilities in both CWE and CVE. Personal data protection seems to attract a lot of attentions in CWE/CVE with more than half of the CWEs (56.1%) and CVEs (59.87%) vulnerabilities reported, most of which (36.59% in CWEs and 50.32% in CVEs) are related to exposing personal data to an unauthorised actor. There are 19.51% in CWE and 15.92% in CVE reporting vulnerabilities regarding personal data attacks.

There are four types of privacy vulnerabilities that are covered by both CWE and CVE: exposing personal data to an unauthorised actor, insufficient levels of personal data protection, improper methods/techniques for personal data protection, and personal data attacks. There are several types of privacy vulnerabilities that have been reported in CVE, but not in CWE, such as allowing unauthorised actors to track individuals, not following user privacy preferences, and not asking for user consent. 8.28% of the CVEs refer to those types of privacy vulnerabilities, suggesting that those types of vulnerabilities need to be added into the CWE system.

A number of areas that are not covered by the existing privacy vulnerabilities in CWE and CVE are highlighted in red outline in Figure 4. For example, exclusion involves a range of privacy vulnerabilities in failing to provide users with notice of user consent and input about their privacy preferences. User consent and privacy preferences are two essential mechanisms that enable users to control their personal data processing. These sources [6, 32, 56] confirm that user privacy is vulnerable if users are not presented with options to specify or cannot modify their user privacy preferences. Similarly, user privacy may be compromised if the users cannot modify or withdraw their consent, or are not notified about any changes of consent [6, 30, 32, 35, 37]. However, none of them (except not asking for a user consent vulnerability) is covered in existing CVE.

Other areas that are not well covered in existing CWEs and CVEs are privacy vulnerabilities due to insecurity and secondary use. Processing personal data at a third party is a risk since they may apply weaker level personal data protection, particularly mobile [45] and web applications [37]. There are cases that user privacy is violated when personal data is used for unspecified purposes or used or transferred without permissions (e.g. [32, 33, 34, 38, 40, 41]). In addition, although allowing unauthorised actors to modify personal data and collecting personal data without user consent/permissions are serious threats, none of the existing privacy vulnerabilities in CWE and CVE covers this.

The existing privacy weaknesses and vulnerabilities reported in the CWE and CVE systems cover only 13 out of 24 common privacy vulnerabilities raised in research and practice.

V. New common privacy weaknesses

Our study has shown the gaps in the existing CWE and CVE systems in terms of covering privacy vulnerabilities. To fill these gaps, we propose 11 new common privacy weaknesses to be added to the CWE system. We focus on CWE instead of CVE since CVE specifies unique vulnerabilities existing in specific software systems and application, while CWEs are at the generic level, similar to our taxonomy of privacy threats. We followed the CWE schema [57] to define the new CWE entries which include attributes such as name, description, mode of introduction, common consequence, detection method, potential mitigation and demonstrative example. These attributes provide an overview of a privacy weakness in terms of its causes and consequences, and mitigation methods.

Security vulnerabilities can typically be well specified as concrete hardware/software attacks that can be exploited to bypass the system's security. Unlike security vulnerabilities, privacy vulnerabilities seem to be more contextual than security vulnerabilities and its operationalisation may differ depending on context. In our study, we tried to formalise the privacy vulnerabilities as software attacks that are harmful to personal data and violate individual rights. Thus, it is important to provide the detailed description in each attribute in the identified 11 common privacy weaknesses since it defines a specific scenario that potentially leads privacy vulnerabilities.

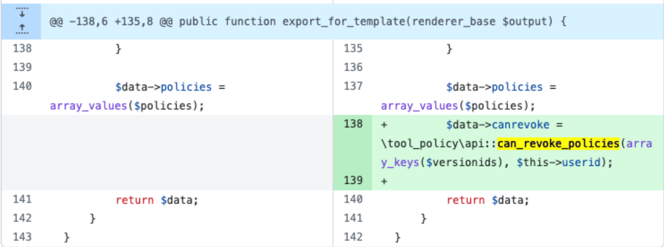
The proposed common privacy weaknesses address four groups of threats: surveillance, insecurity, secondary use and exclusion. The weaknesses in the insecurity group can be detected and resolved by implementing security mechanisms to better protect personal data and user privacy. On the other hand, the other groups of weaknesses require more attention from software development teams on examining privacy constraints involved, and designing relevant functions to respond to those constraints. The software development team needs to determine relevant functions, take privacy constraints into consideration and regularly review existing functions to ensure that they do not violate user privacy. For example, a user may want to change his/her user preferences when a new feature is launched, or a system must notify its users when there is a change to the policy that they have given consent to.

Due to space limit, we present here two examples (see Table I and Table II) of the new CWE privacy weaknesses and refer to the readers to our replication package [24] for the remaining newly proposed CWEs. Table I shows a new common privacy weakness for not allowing a user to withdraw his/her consent. This weakness belongs to the exclusion category. Following the CWE schema, a short summary of the weakness is provided in its name, while a detailed description is provided in the description section. The mode of introduction briefly discusses how and when the weakness is introduced, which in this case is the architecture and design phase.

The common consequence section identifies a privacy property that is violated and an effect that is caused by the weakness. The detection method section describes different methods that the weakness can be detected in software. We also propose methods to mitigate the weakness. It is noted that different phases in software development may pose different privacy concerns. Finally, we provide a demonstrative example of the new weaknesses by extracting code fragments, issue

TABLE I
AN EXAMPLE OF A MISSING CONSENT WITHDRAWAL WEAKNESS.

Subcategory: Exclusion
Class: Not allowing a user to withdraw his/her consent
Name: Missing consent withdrawal
Description: The software forces users to give consent before providing its services. These services may include personal data processing. However, the software does not a provide a function for users to withdraw their consent. The users can only accept consent, but they cannot withdraw their consent when they wish to. This vulnerability seriously violates user privacy.
Mode of introduction: Phase: Architecture and Design. This weakness is caused by a missing privacy consideration about consent management and its related processes, which leads to the missing consent withdrawal function.
Common consequence: The software violates user privacy by not allowing users to express their agreement on the use of their personal data.
Detection method: Method: Manual analysis. Description: A consent management page or window in a software does not show an icon or option to withdraw consent.
Potential mitigations: Phase: Architecture and Design. Strategy: User consent withdrawal consideration. Description: The software development team should consider which points in the software that should provide a user an ability to withdraw consent.
Demonstrative example: Issue MDL-62309 in Moodle reports that the users cannot withdraw consent and cannot enter the site without giving consent. This issue violates user privacy as the users should be able to freely withdraw consent.



```
@@ -138,6 +135,8 @@ public function export_for_template(renderer_base $output) {
138               }                          135               }
139                                          136
140               $data->policies =          137               $data->policies =
        array_values($policies);                     array_values($policies);
                                             138   +           $data->canrevoke =
                                                     \tool_policy\api::can_revoke_policies(arra
                                                     y_keys($versionids), $this->userid);
                                             139   +
141               return $data;              140               return $data;
142           }                              141           }
143       }                                  142       }
```

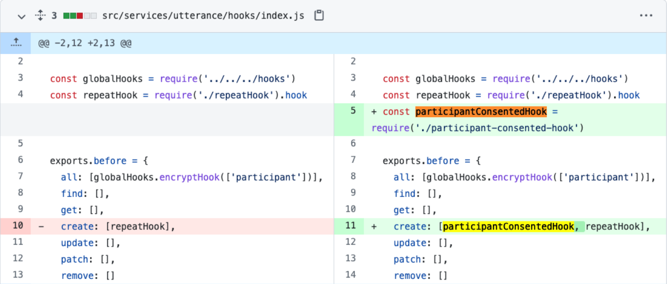
Fig. 6. An example of missing consent withdrawal function in Moodle (Issue MDL-62309).

reports and commits from real software repositories hosted on GitHub[18]. We used the keywords in the weakness class and name to search for relevant code, commits and issues. Once the search results were returned from GitHub, we went through the descriptions and identified the demonstrative example from those results. For example, Figure 6 shows a code fragment extracted from the Github commit *ad5e213*[19] in Issue MDL-62309[20] of the Moodle project. This example demonstrates the existence of the missing consent withdrawal weakness in practice.

Table II presents a new common privacy weakness for collecting personal data without user consent/permissions. This weakness belongs to improper personal data collection category. Figure 7 shows a code fragment confirming the existence of a missing consent check before collecting personal data weakness. The code fragment is extracted from the Github commit *0b09df0*[21] in the HumanDynamic repository [58].

TABLE II
AN EXAMPLE OF A COLLECTING PERSONAL DATA WITHOUT USER CONSENT/PERMISSIONS WEAKNESS.

Subcategory: Surveillance
Class: Collecting personal data without user consent/permissions
Name: Missing a consent check before collecting personal data
Description: The software does not check for a user consent prior to personal data collection. This makes the software collect personal data that users have not given consent to (e.g., location and speech).
Mode of introduction: Phase: Implementation. This weakness is caused by missing a consent check before collecting personal data.
Common consequence: The software violates user privacy since users has not given consent/permissions to collect their personal data.
Detection method: Method: Manual analysis. Description: Perform a code check at points of personal data collection.
Potential mitigations: Phase: Implementation. Strategy: Check for user consent before collecting data. Description: The software development team should perform a consent check at every point that collects personal data in the software.
Demonstrative example: A commit 0b09df0 in HumanDynamics repository collects user speech without consent check.



```
@@ -2,12 +2,13 @@
 2                                                        2
 3   const globalHooks = require('../../../hooks')        3   const globalHooks = require('../../../hooks')
 4   const repeatHook = require('./repeatHook').hook      4   const repeatHook = require('./repeatHook').hook
                                                          5 + const participantConsentedHook =
                                                            require('./participant-consented-hook')
 5                                                        6
 6   exports.before = {                                   7   exports.before = {
 7     all: [globalHooks.encryptHook(['participant'])],   8     all: [globalHooks.encryptHook(['participant'])],
 8     find: [],                                          9     find: [],
 9     get: [],                                          10     get: [],
10 -   create: [repeatHook],                             11 +   create: [participantConsentedHook, repeatHook],
11     update: [],                                       12     update: [],
12     patch: [],                                        13     patch: [],
13     remove: []                                        14     remove: []
```

Fig. 7. An example of missing a consent check before collecting personal data in HumanDynamics (Commit 0b09df0).

We propose 11 new common privacy weaknesses to be added to CWE. These will significantly improve the coverage of privacy weaknesses and vulnerabilities in CWE, and subsequently CVE.

VI. THREATS TO VALIDITY

Internal validity. Our method for the extraction of privacy vulnerabilities in CWE and CVE using keywords might not result in the complete list. We also note that we did not consider the variations of the selected keywords (e.g. plural forms). However, we have used several strategies to mitigate these threats such as determining the keywords based on alternate terms described in CWE, and using the frequent terms identified in the studies that performed a large-scale analysis in privacy policies and considering general terms to cover unseen materials (e.g. regulation, data protection and privacy standard).

External validity. Our taxonomy of common privacy threats is constructed and refined based on the existing privacy threats taxonomy. We have put our best effort to ensure the comprehensiveness of the study by examining popular software engineering publication venues, well-established data protection regulations and privacy frameworks, and relevant reputable organisations. However, we acknowledge that there might be other sources of other privacy threats that we have not

[18] https://github.com/
[19] https://github.com/moodle/moodle/commit/ad5e213
[20] https://tracker.moodle.org/browse/MDL-62309
[21] https://github.com/HumanDynamics/rhythm-server/commit/0b09df0

identified yet. We have carefully defined a set of inclusion and exclusion criteria to select the most relevant papers so that we got a reasonable number of papers to be examined individually. Future work would involve expanding our explanatory study to increase the generalisability of our taxonomy for common privacy threats. In addition, classifying the privacy-related vulnerabilities in CWE and CVE into the common privacy threats in the taxonomy involved subjective judgements. We have applied several strategies (e.g. using multiple coders, applying inter-rater reliability assessments and conducting disagreement resolution) to mitigate this threat. Future work could explore the use of external subject matter experts in these tasks.

VII. Conclusions and future work

The increasing use of software applications in people's daily lives has put privacy under constant threat as personal data are collected, processed and transferred by many software applications. In this paper, we performed a number of studies on privacy vulnerabilities in software applications. Our study on CWE and CVE systems found that the coverage of privacy-related vulnerabilities in both systems is quite limited (4.45% in the CWE system and 0.1% in the CVE system).

We have also investigated on how those privacy-related vulnerabilities identified in CWE and CVE address the common privacy threats in software applications. To do so, we developed a taxonomy of common privacy threats, which was extended from the existing work [13], based on selected privacy engineering research, data protection regulations and privacy frameworks and industry resources. We have found that only 13 out of 24 common privacy vulnerabilities in the taxonomy are covered by the existing weaknesses and vulnerabilities reported in CWE and CVE. Based on these actionable insights, we proposed 11 new common privacy weaknesses to be added to the CWE system. We also mined code fragments from real software repositories to confirm the existence of those privacy weaknesses. These newly proposed weaknesses will significantly improve the coverage of privacy weaknesses and vulnerabilities in CWE, and subsequently CVE.

Future work involves expanding our taxonomy to cover additional common privacy threats that may have raised or discussed in other sources. We will also perform a study to characterise privacy vulnerabilities in software applications. This will enable us to develop new techniques and tools for automatically detecting privacy vulnerabilities in software and suggesting fixes.

VIII. Data Availability

A full replication package containing all the artifacts and data produced by our study is made publicly available at https://figshare.com/articles/conference_contribution/icse2023-paper908-replication-pkg/21922731 or DOI: https://doi.org/10.6084/m9.figshare.21922731. There are five folders in the package: i) accepted-paper, ii) code, iii) common-weakness-proposal, iv) data and v) taxonomy. The **README** file that explains relevant folders and files is also included in the replication package.

References

[1] J. Johnson, "Internet users in the world 2021," 2021. [Online]. Available: https://www.statista.com/statistics/617136/digital-population-worldwide/

[2] Statista, "Annual number of data breaches and exposed records in the United States from 2005 to 2020," 2021. [Online]. Available: https://bit.ly/statistaUS2022

[3] Office of the Australian Information Commissioner, "Notifiable data breaches statistics," 2021. [Online]. Available: https://www.oaic.gov.au/privacy/notifiable-data-breaches/notifiable-data-breaches-statistics/

[4] J. Johnson, "Cyber crime: biggest online data breaches as of 2020," 2021. [Online]. Available: https://www.statista.com/statistics/290525/cyber-crime-biggest-online-data-breaches-worldwide/

[5] D. E. Bambauer, "Privacy Versus Security," *Journal of Criminal Law and Criminology*, vol. 103, no. 3, 2013. [Online]. Available: https://scholarlycommons.law.northwestern.edu/jclc/vol103/iss3/2

[6] ISO/IEC, "International Standard ISO/IEC 29100," ISO/IEC, Tech. Rep., 2011.

[7] The MITRE Corporation, "About CWE," p. The MITRE Corporation., 2021. [Online]. Available: https://cwe.mitre.org/about/index.html

[8] ——, "About CVE," 2021. [Online]. Available: https://cve.mitre.org/about/index.html

[9] Z. Yang, M. Yang, Y. Zhang, G. Gu, P. Ning, and X. S. Wang, "AppIntent: Analyzing sensitive data transmission in Android for privacy leakage detection," in *Proceedings of the ACM Conference on Computer and Communications Security*, 2013, pp. 1043–1054.

[10] C. Y. Ma, D. K. Yau, N. K. Yip, and N. S. Rao, "Privacy vulnerability of published anonymous mobility traces," *IEEE/ACM Transactions on Networking*, vol. 21, no. 3, pp. 720–733, 2013.

[11] Open Web Application Security Project, "About Us — The OWASP Foundation," 2020. [Online]. Available: https://owasp.org/about/

[12] C. Stouffer, "What is social engineering?" 2021. [Online]. Available: https://us.norton.com/internetsecurity-emerging-threats-what-is-social-engineering.html

[13] W. Stallings, *Information Privacy Engineering and Privacy by Design: Understanding Privacy Threats, Technology, and Regulations Based on Standards and Best Practices*. Addison-Wesley, 2019.

[14] D. Gonzalez, H. Hastings, and M. Mirakhorli, "Automated Characterization of Software Vulnerabilities," in *Proceedings - 2019 IEEE International Conference on Software Maintenance and Evolution, ICSME 2019*, 2019, pp. 135–139.

[15] X. Li, J. Chen, Z. Lin, L. Zhang, Z. Wang, M. Zhou, and W. Xie, "A Mining Approach to Obtain the Software

Vulnerability Characteristics," in *Proceedings - 5th International Conference on Advanced Cloud and Big Data, CBD 2017*, 2017, pp. 296–301.

[16] S. Liu, G. Lin, Q. L. Han, S. Wen, J. Zhang, and Y. Xiang, "DeepBalance: Deep-Learning and Fuzzy Oversampling for Vulnerability Detection," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 7, pp. 1329–1343, 2020.

[17] G. P. Bhandari, A. Naseer, and L. Moonen, "CVEfixes: Automated Collection of Vulnerabilities and Their Fixes from Open-Source Software," 2021.

[18] C. C. Galhardo, P. Mell, I. Bojanova, and A. Gueye, "Measurements of the Most Significant Software Security Weaknesses," in *ACM International Conference Proceeding Series*, vol. 11, 2020, pp. 154–164.

[19] D. J. Solove, "A taxonomy of privacy," *University of Pennsylvania Law Review*, vol. 154, no. 3, pp. 477–564, 2006.

[20] M. Deng, K. Wuyts, R. Scandariato, B. Preneel, and W. Joosen, "A privacy threat analysis framework: Supporting the elicitation and fulfillment of privacy requirements," *Requirements Engineering*, vol. 16, no. 1, pp. 3–32, 2011.

[21] K. Wuyts, R. Scandariato, and W. Joosen, "Empirical evaluation of a privacy-focused threat modeling methodology," *Journal of Systems and Software*, vol. 96, pp. 122–138, 2014.

[22] National Institute of Standards and Technolgy, "NIST Privacy Framework: a tool for improving privacy through enterprise risk management," *NIST Special Publication*, p. 43, 2020. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.01162020.pdf

[23] C. Bloom, "Privacy Threat Modeling," Tech. Rep., 2022.

[24] P. Sangaroonsilp, H. K. Dam, and A. Ghose, "Replication Package," 2022. [Online]. Available: https://figshare.com/articles/conference_contribution/icse2023-paper908-replication-pkg/21922731

[25] S. Gürses and J. M. Del Álamo, "Privacy Engineering: Shaping an Emerging Field of Research and Practice," *IEEE Security and Privacy*, vol. 14, no. 2, pp. 40–46, 2016.

[26] B. Kitchenham, "Guidelines for performing systematic literature reviews in software engineering," *Technical report, Ver. 2.3 EBSE Technical Report. EBSE*, 2007.

[27] F. Ebrahimi, M. Tushev, and A. Mahmoud, "Mobile App Privacy in Software Engineering Research: A Systematic Mapping Study," *IEEE TRANSACTIONS ON SOFTWARE ENGINEERING*, vol. 14, no. 8, 2019.

[28] H. Perera, W. Hussain, J. Whittle, A. Nurwidyantoro, D. Mougouei, R. A. Shams, and G. Oliver, "A study on the prevalence of human values in software engineering publications 2015 - 2018," in *Proceedings - International Conference on Software Engineering*, vol. 12, 2020, pp. 409–420.

[29] A. Bertolino, A. Calabrò, F. Lonetti, E. Marchetti, and B. Miranda, "A categorization scheme for software engineering conference papers and its application," in *Journal of Systems and Software*, vol. 137, 2018, pp. 114–129.

[30] Office Journal of the European Union, "General Data Protection Regulation (GDPR)," Tech. Rep., 2016. [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679

[31] California State Legislature, "California Consumer Privacy Act of 2018," 2018.

[32] Department of Health and Human Services, "HIPAA administrative simplification regulation text," pp. 1–115, 2013. [Online]. Available: https://www.hhs.gov/sites/default/files/hipaa-simplification-201303.pdf

[33] United State Congress, "GRAMM-LEACH-BLILEY ACT," 1999. [Online]. Available: https://www.govinfo.gov/content/pkg/PLAW-106publ102/pdf/PLAW-106publ102.pdf

[34] U.S. Department of Justice, "The Privacy Act of 1974," 1974.

[35] Federal Register of Legislation, "Privacy Act 1988," 2017. [Online]. Available: https://www.legislation.gov.au/Details/C2016C00979

[36] Norton, "Why Your Online Privacy Matters," 2021. [Online]. Available: https://us.norton.com/internetsecurity-privacy-why-your-online-privacy-matters.html

[37] The OWASP Foundation, "OWASP Top 10 Privacy Risks Survey," 2015.

[38] S. J. De and D. L. Metayer, "Privacy Harm Analysis: A Case Study on Smart Grids," in *Proceedings - 2016 IEEE Symposium on Security and Privacy Workshops, SPW 2016*, 2016, pp. 58–65.

[39] K. Lebeck, K. Ruth, T. Kohno, and F. Roesner, "Towards Security and Privacy for Multi-user Augmented Reality: Foundations with End Users," in *Proceedings - IEEE Symposium on Security and Privacy*, vol. 2018-May, 2018, pp. 392–408.

[40] R. Hasan, D. Crandall, M. Fritz, and A. Kapadia, "Automatically detecting bystanders in photos to reduce privacy risks," in *Proceedings - IEEE Symposium on Security and Privacy*, vol. 2020-May, 2020, pp. 318–335.

[41] G. Fisk, C. Ardi, N. Pickett, J. Heidemann, M. Fisk, and C. Papadopoulos, "Privacy principles for sharing cyber security data," in *Proceedings - 2015 IEEE Security and Privacy Workshops, SPW 2015*, 2015, pp. 193–197.

[42] L. S. Figueiredo, D. Molnar, M. Veanes, and B. Livshits, "Prepose: Privacy, security, and reliability for gesture-based programming," in *IEEE Security and Privacy*, vol. 15, no. 2, 2017, pp. 14–23.

[43] G. Venkatadri, A. Andreou, Y. Liu, A. Mislove, K. P. Gummadi, P. Loiseau, and O. Goga, "Privacy Risks with Facebook's PII-Based Targeting: Auditing a Data Broker's Advertising Interface," in *Proceedings - IEEE Symposium on Security and Privacy*, vol. 2018-May. Institute of Electrical and Electronics Engineers Inc., jul 2018, pp. 89–107.

[44] D. F. Some, "EmPoWeb: Empowering web applications with browser extensions," in *Proceedings - IEEE Sympo-*

sium on Security and Privacy, vol. 2019-May, may 2019, pp. 227–245.

[45] X. Zhang, X. Wang, R. Slavin, T. Breaux, and J. Niu, "How does misconfiguration of analytic services compromise mobile privacy," in *Proceedings - International Conference on Software Engineering*, 2020, pp. 1572–1583.

[46] P. Calciati, K. Kuznetsov, X. Bai, and A. Gorla, "What did really change with the new release of the app?" in *Proceedings - International Conference on Software Engineering*, vol. 18, 2018, pp. 142–152.

[47] S. Jana, A. Narayanan, and V. Shmatikov, "A scanner darkly: Protecting user privacy from perceptual applications," in *Proceedings - IEEE Symposium on Security and Privacy*, 2013, pp. 349–363.

[48] Lucia, D. Lo, L. Jiang, and A. Budi, "Kb e-anonymity: Test data anonymization for evolving programs," in *2012 27th IEEE/ACM International Conference on Automated Software Engineering, ASE 2012 - Proceedings*, 2012, pp. 262–265.

[49] G. Chalhoub, I. Flechais, N. Nthala, and R. Abu-Salma, "Innovation inaction or in action? The role of user experience in the security and privacy design of smart home cameras," in *Proceedings of the 16th Symposium on Usable Privacy and Security, SOUPS 2020*, 2020, pp. 185–204.

[50] I. Omoronyia, L. Cavallaro, M. Salehie, L. Pasquale, and B. Nuseibeh, "Engineering adaptive privacy: On the role of privacy awareness requirements," in *Proceedings - International Conference on Software Engineering*, 2013, pp. 632–641.

[51] J. Bhatia and T. D. Breaux, "Semantic incompleteness in privacy policy goals," in *Proceedings - 2018 IEEE 26th International Requirements Engineering Conference, RE 2018*. Institute of Electrical and Electronics Engineers Inc., oct 2018, pp. 159–169.

[52] L. Yu, X. Luo, J. Chen, H. Zhou, T. Zhang, H. Chang, and H. K. Leung, "PPChecker: Towards Accessing the Trustworthiness of Android Apps' Privacy Policies," *IEEE Transactions on Software Engineering*, vol. 47, no. 2, pp. 221–242, feb 2021.

[53] B. Mihaylov, L. Onea, and K. M. Hansen, "Architecture-based regulatory compliance argumentation," *Journal of Systems and Software*, vol. 119, pp. 1–30, 2016.

[54] K. A. Hallgren, "Computing inter-rater reliability for observational data: An overview and tutorial." *Tutorials in Quantitative Methods for Psychology*, vol. 8, no. 1, pp. 23–34, 2012.

[55] A. J. Viera and J. M. Garrett, "Understanding interobserver agreement: the kappa statistic," *Family Medicine*, vol. 37, no. 5, pp. 360–363, 2005.

[56] A. I. Antón and J. B. Earp, "A requirements taxonomy for reducing Web site privacy vulnerabilities," *Requirements Engineering*, vol. 9, no. 3, pp. 169–185, 2004.

[57] CWE Team, "CWE - Schema Documentation - Schema Version 6.5," 2021. [Online]. Available: https://cwe.mitre.org/documents/schema/

[58] Human Dynamics, "Human Dynamics Github Repository." [Online]. Available: https://github.com/HumanDynamics