Enhanced Security Approaches for Data Protection: Managing Consent, Data Breach, and Asset Inheritance

A thesis submitted to Indian Statistical Institute
in partial fulfillment of the thesis requirements for the degree of
Doctor of Philosophy in Computer Science

Author: Ram Govind Singh

under the guidance of

Dr. Sushmita Ruj
University of New South Wales, Sydney
&
Dr. Sabyasachi Karati
Indian Statistical Institute Kolkata

Cryptology and Security Research Unit
Indian Statistical Institute
203 Barrackpore Trunk Road
Kolkata, West Bengal
India - 700 108

July 2024

Dedicated to all friends

Declaration of Authorship

I, **Ram Govind Singh**, a student of Cryptology and Security Research Unit, of the Ph.D. program of Indian Statistical Institute, Kolkata, hereby declare that the investigations presented in this thesis are based on my works and, to the best of my knowledge, the materials contained in this thesis have not previously been published or written by any other person, nor it has been submitted as a whole or as a part for any degree/diploma or any other academic award anywhere before.

Ram Govind Singh
Cryptology and Security Research Unit,
Indian Statistical Institute, Kolkata 203 Barrackpore Trunk Road,
Kolkata, West Bengal, India - 700108

List of Pubications/Manuscript

1. Ram Govind Singh and Sushmita Ruj. “A Technical Look At The Indian Personal Data Protection Bill.” in *“Arxiv Preprint”*, 2020. Available at: https://arxiv.org/abs/2005.13812.

2. Ram Govind Singh, Ananya Shrivastava, Sushmita Ruj . “A Digital Asset Inheritance Model to Convey Online Persona Posthumously.” In International Journal of Information Security, pp. 983-1003. Springer, 2022. DOI: 10.1007/s10207-022-00593-8

3. Ram Govind Singh, and Sushmita Ruj. “Encoding of security properties for transparent consent data processing.” In 2023 IEEE Guwahati Subsection Conference (GCON), Guwahati, India, 2023, pp. 01-08, DOI: 10.1109/GCON58516.2023.10183463.

4. Ram Govind Singh and Naveenkumar D. “ESUL analyzer for inceptive threat identification and mitigation.” In 2023 Third International Conference on Secure Cyber Computing and Communication (ICSCCC), Jalandhar, India, 2023, pp. 409-414, DOI: 10.1109/ICSCCC58608.2023.10176536.

5. Ram Govind Singh and Naveenkumar D. “Are we undermining data breaches? Protecting education sector from data breaches.” In 2023 IEEE International Conference on Computer, Electronics and Electrical Engineering and Their Applications (IC2E3), Srinagar, Uttrakhand, India, 2023, pp. 1-6, DOI: 10.1109/IC2E357697.2023.10262570.

6. Ram Govind Singh, and Sushmita Ruj. “Evaluation and assessment of data breaches for robust compliance under DPDPA.” SPRINGER ICISS, 2023. Manuscript Submitted.

Acknowledgements

As we move forward, we look at life more cherished through the unforgettable memories and experiences that we gain in the journey. I believe words not enough to express gratitude and acknowledge the support I have received throughout this journey. For me, this is an honour and an enriching experience.

I would like to express my gratitude to Dr. Sushmita Ruj, my supervisor, for her unwavering support, guidance, encouragement, and motivation. Her wealth of knowledge has been instrumental in my successful research and thesis writing. Having a supervisor who consistently supports me, both academically and personally, is truly a privilege. I extend my sincere heartfelt thanks to my co-supervisor, Dr. Sabyasachi Karati, for the administrative guidance and support provided. The assistance in navigating administrative processes has been invaluable, contributing significantly to the smooth progress of various tasks during my Ph.D. journey.

I wish to offer my gratitude to my course teachers, Prof. Palash Sarkar, Prof. Rana Barua, Prof. Sushmita Sur Kolay, Prof. Sandip Das, Prof. Mridul Nandi, Dr. Debrup Chakraborty, Dr. Arijit Bishnu, and Dr. Sushmita Ruj. I am really fortunate to have had such mentors during my Ph.D course works. These courses will always remain precious in my academic journey.

My Ph.D. thesis would have remained unfinished without the generous support of my collaborators, and I owe a profound debt of gratitude Dr. Ananya Shrivastava and Naveen Kumar. I have reached out to them for discussion and research ideas. They have consistently displayed kindness by offering invaluable insights and engaging in discussions whenever needed.

Life lacks fulfillment without friends, especially the ones who bring consistent supports. Consider myself fortunate to have friends like Prabal Banerjee, Avishek Majumder, Mostafizar Rahman, Samir kundu, Soumya Das, Nayana Das as my constant support team. Thanks for providing guidance, endless discussion on any topic, and cheering me whenever I felt disappointed. They made my stay in ISI Kolkata both inspiring and enjoyable. A healthy mind resides in a healthy body, and I could not have aced this journey without the support of these guys. Special thanks go to all of my friends and seniors in ISI Kolkata. Thanks to Subham, Ayush, Hemant, Kunal Sharma, Satyendra Singh Rajput and many more friends from Indian Computer Emergency Response Team (CERT-In) who helped me at different times in different ways to complete this journey.

I express my heartfelt gratitude to the non-academic staff of ASU, CSRU, the accounts section, and the Dean's office, who have assisted me with several logistics and official work.

I wish to thank my teachers and friends from my alma maters, IDCL intermediate college Jasra, Prayagraj; SVN intermediate college Naini, Prayagraj; Bundelkhand Institute of Engineering and Technology (BIET), Jhansi; Maulana Azad National Institute of Technolgy (MANIT), Bhopal. This list would be incomplete without thanking Kushal Sahu, Kaushal Kishore, Sunil Kumar, Vipin Yadav, Anuj Gupta, Amit Kumar, and Arihant Banthiya. I have literally tortured them with my incessant complaints and how they kept on encouraging me and acted as an anchor through the tough days. I wish to thanks all my colleagues of United Institute of Technology.

Last but not the least, I will remain indebted to my family for their continuous support, love, and help. Without their encouragement, inspiration, and motivation, I would not have got the mental strength to pursue a career in academics amidst the societal stereotypes and prejudices.

Date: 15th July, 2024

Ram Govind Singh

Abstract

The journey of the Indian data protection framework started in 2018 with the introduction of the initial draft as "Personal Data Protection Bill (PDPB-2018)". Subsequently, a revised draft PDPB-2019 was introduced. This went through revisions as PDPB 2021 and Digital Personal Data Protection Bill (DPDPB-2022). Finally, it was passed as "Digital Personal Data Protection Act" (DPDPA, 2023). The framework emphasized on protected data processing while the user's privacy is honored.

In this thesis, we look at the technical aspects in DPDPA and suggest ways to address the different clauses of the bill. We have analyzed four components: a) *user's consent* that states the nature and scope of consent-based data processing, b) *right to access/right to nominate* to assure the right to nominate someone as a nominee, c) *data breach* to enable appropriate technical measures to prevent and analyze data breach. d) *storage/logging* to preserve and evaluate various logs that strengthen security posture and incident response. Enhanced approaches have been explored under each obligation for stronger data management and processing aligning with the framework.

In analyzing user's consent, we have described that encoding of requisite security and privacy properties will ascertain stronger consent processing. We formalize these properties as Proofs of Consent (PoC) and categorized them into three layers. The acquisition of a higher layer will minimize adversarial risks and ascertain greater transparency. Next, we have proposed a model Shielded Consent Manager (SCM) using blockchain and other cryptographic primitives for retrieval of consent to grant permissions to access android resources. Further, following the right to nominee obligations, we have proposed a model Digital Asset Inheritance Protocol (DAIP) using CertificateLess Encryption (CLE) and Identity Based System (IBS) to convey the user's online persona efficiently to the descendent after his death. DAIP allows the nominee to successfully retrieve the asset after the user's demise, even if a nominee is uninformed regarding the asset. Then, we have proposed the system model of a Data Breach Incident Assessor (DBIA) aiming for breach assessment. It helps in the validation of a threat actor's claim, understanding the root cause of a breach, analyze the scope of the compromise, and provide analysis according to the regulation. Finally, an End System URL Analyzer (ESUL) to analyze the URL based logs in end system is presented.

The simulation and result analysis is done for each of the above approaches. We show that enhanced security approaches can help to realize the obligations in DPDPA, thus ensuring robust data management and processing.

Contents

List of Figures

List of Tables

LIST OF ACRONYMS AND ABBREVIATIONS

Throughout the thesis, we use some acronyms and abbreviations and we describe those common notations here.

Expansion	**Acronyms/ Abbreviations**
Data Fiduciary	DF
Data Processor	DPR
Service Provider	SP
Data Principal	DP
Personal Data Protection Bill	PDPB
Digital Personal Data Protection Bill	DPDPB
Digital Personal Data Protection Act	DPDPA
Third Party	TP
Data Protection Authority of India	DPAI
Data Protection Board of India	DPBI
Data Breach	DB
Incident Response	IR
Consent Manager	CM
Digital Asset Inheritance	DAI

Chapter 1

Introduction

Privacy is the ability of an individual to make choices and preferences on their personal matters. It is a *fundamental right* in the Indian constitution [121]. The *fundamental rights* are a set of rights that require a higher degree of protection from the government irrespective of a person's caste, race, religion, gender, or place of birth. In 2017, the right to privacy was reaffirmed as a fundamental right under the constitution of India [26]. The judgment by the Supreme Court of India declares: *"The right to privacy is protected as an intrinsic part of the right to life and personal liberty under Article 21 and as a part of the freedoms guaranteed by part III of the constitution"*. Under *Right to privacy*, "Any individual, group, or organization can exclude themselves or information about themselves, thereby representing themselves selectively".

Data privacy (a.k.a. information privacy) is the preferences of an individual to decide when, whom, and how much data may be revealed to others. Typically, organizations collect data from a small amount to a large extent with/without the user's knowledge. The collected data is used for data analysis, marketing, automated decision-making, and profiling and is shared with other third parties. The organizations are expected to preserve user data privacy during processing. They use different mechanisms to achieve this, such as encryption, authentication, authorization, access control, etc. However, data may not always be protected, for instance, the massive data breach on GoDaddy [98]. The threat actors exfiltrate data regularly. The exposed data are posted for sale on DarkWeb and appear on public blogs such as BreachForums. Even if a data breach is prevented, it does not ensure privacy. A data fiduciary may not always be trusted and may disclose personal data to third parties surreptitiously for monetary benefits [160]. Therefore, the existing system does not affirm data is indeed protected.

The primary intent is to affirm: "*How a service provider (SP) can comply to enable users to opt for adequate choices & preferences before sharing personal data, and how the shared personal data can be processed promising pertinent security & safeguard, honoring user's privacy, and applying necessary protection?*". The defense should be achieved throughout the data life cycle. It should also prevent risks such as breach, inappropriate disclosure, and adversarial uses.

The above concerns demand an urgent requirement to design a legal framework to regulate data processing entities to comply with secure and protected data processing. To ensure this, the central government of India had established a committee led by Justice B. N. Srikrishna to study the existing challenges and to provide its recommendation for establishing the legal framework. The objective was *"to ensure the growth of the digital economy while keeping citizens' data secure and* *protected"*. The committee submitted its report, and based on it, a draft protection known as *The Personal Data Protection Bill-2018 (PDPB, 2018)* was introduced by the Ministry of Electronics and IT (MEITY), which was the first step towards India's data privacy journey.

After the draft version 2018, a modified version was introduced in Lok Sabha, a lower house in the Indian parliament, named as "The Personal Data Protection Bill-2019 (PDPB)" [141]. This research was started primarily aiming at the PDPB 2019 Draft. We summarize the proposed draft bill and critically analyse it and its associated challenges. Then, we discuss modifications proposed in subsequent bill drafts and the passed act.

1.1 Brief description of PDPB-2019 (Draft)

Data is any information, opinion, facts, and concepts, for instance, health data, biometric data, genetic data, financial data, etc. The bill categorizes data into three parts:
a. Personal data : It is the data about or relating to a *person* who is directly or indirectly identifiable.
b. Sensitive personal data : It may include health, financial, biometric, etc. These are more sensitive and require stronger security and safeguards.
c. Critical personal data : The *personal data* needs more protection and would be processed within India only. The specific data that fall under each category was expected to be defined in the future.

Figure 1-1 shows the primary entities of the PDPB framework. The entities involved are:
i) ***Data Principal:*** A party who owns the data.
ii) ***Data Fiduciary (DF):*** Any person, including the state, a company, any juristic entity or any individual who alone or in conjunction with others determines the purpose and means of the processing of personal data.
iii) ***Data Processor:*** A party, either a data fiduciary or any other party who processes the data on behalf of the fiduciary.
iv) Auditors: entities accountable for data audits.
v) Data protection officer (DPO): Point of contact person related to data protection matters.
vi) Consent Manager: To manage consent records.
vii) Data Protection Authority of India (DPAI) : Entities accountable to regulate the Act.

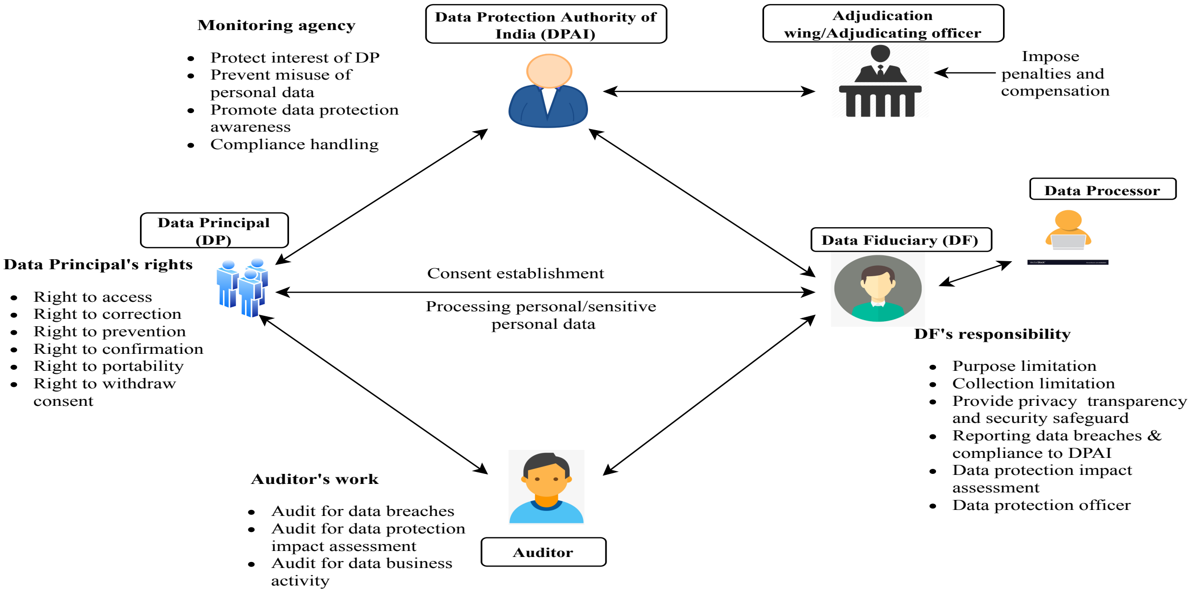

Monitoring agency
Protect interest of DP
Prevent misuse of personal data
Promote data protection awareness
Compliance handling
Data Protection Authority of India (DPAI)
Adjudication wing/Adjudicating officer
Impose penalties and compensation
Data Processor
Data Principal (DP)
Data Principal's rights
Right to access
Right to correction
Right to prevention
Right to confirmation
Right to portability
Right to withdraw consent
Consent establishment
Processing personal/sensitive personal data
Data Fiduciary (DF)
DF's responsibility
Purpose limitation
Collection limitation
Provide privacy transparency and security safeguard
Reporting data breaches & compliance to DPAI
Data protection impact assessment
Data protection officer
Auditor's work
Audit for data breaches
Audit for data protection impact assessment
Audit for data business activity
Auditor

Figure 1-1: Primary entities defined in PDPB-2019 (Draft)

1.1.0.1 Summary of constituents

Table 1.1 depicts a broad categorization of obligations for the bill. The bill specifies clauses such as grounds for the processing, consent, user's rights, data breach, definition and duties of regulatory entities, etc. We discuss and highlight a few significant clauses.

DF has to collect, process, and share personal data for purposes that should be *clear, specific, reasonable, and lawful* (as per Article 4) and shall be processed only for reasonable and *specified purposes* (as per Article 5). The DF will identify and define such objectives before the data collection and shall disclose them to the DP at the time of collection. DF has to perform *fair and reasonable processing* of data and has to collect minimum data as possible according to the *"collection limitation"* requirements (as per Article 6).

DF has to provide *the necessary notice* (as per Article 7) to the DP before collection and processing. The notice will contain a few requisites, e.g., the purpose of collection, consent form, nature of data being collected, information about any cross-border transfer, etc. DF also needs to maintain the quality of personal data during processing. Data must be accurate, complete, and not misleading (as per Article 8).

Consent is one of the legal bases for data processing and is required before the commencement of processing (as per Article 11). Consent would be *free, informed, specific, and clear*. DF will be responsible for keeping proof that consent has been obtained from DP before the processing. The bill restricts the processing of *sensitive personal data*. The processing of such kind of data will require *explicit consent* (as per Article 11(3)). It means that DF has to draw attention to the DP about the sensitivity of personal data, the reason why such collection is necessary, and the possible consequences. The DP will also have to acknowledge this explicitly. A *consent manager* will be designated by DF, who will be accountable for recording and managing all the consent obtained from DP and will work as a point of contact for DP and DPAI. The DPAI shall publish periodically a list of personal data that would be considered sensitive personal data (as per Article 15).

Articles	**Details**
Article 1-8	Objective and establishment of grounds for processing of personal data
Article 9-10	Data retention policy and accountability
Article 11	Definition of consent and explicit consent
Article 12-14	Norms for processing of personal data without consent
Article 15	Criteria for consideration of personal data as sensitive personal data
Article 16	Grounds for processing of personal and sensitive personal data of children
Article 17-21	Establish the rights of data principal
Article 22-24	Privacy, transparency, consent manager and security safeguards
Article 25-32	Significant data fiduciary, data breach, data audit, data impact assessment and data protection officer
Article 33,34	Grounds for processing critical personal data and transfer of personal data outside of India
Article 35-40	Mandate about processing of personal data for other purposes like security of state, for law or legal processing, journalistic purpose, research or statistical purpose etc.
Article 41-56	Establishment, responsibilities and power of data protection authority of India
Article 57-85	Covers penalties, liability, establishment of appellate tribunal and execution of other offences
Article 86-98	Miscellaneous power of central government, grounds for framing digital India policy and norms for processing biometric data

Table 1.1: Organization of articles of PDPB-2019 (Draft)

The *processing of children's data* will be in such a manner that protects children's rights (as per Article 16). DF has to verify the age of the children and obtain consent from the parent or guardian before processing it. DPAI will specify the procedure and the appropriate mechanism to conduct age verification under this regulation. Further, based on the nature of commercial websites, online services offered, and the volume of children's data being processed, DPAI will categorize a few DF as *guardian data fiduciary*. They are prohibited from profiling, target-based advertising, or other activities that may harm the children.

PDPB provides various rights to DP such as *the right to confirmation and access* (as per Article 17), *the right to correction and erasure* (as per Article 18), *the right to data portability* (as per Article 19), and *the right to be forgotten* (as per Article 20). DP may ask what data is being processed by DF in its summary and request that DF update, alter, correct, or erase the personal data. He may also request DF to prevent or restrict further disclosure of personal data.

DF has to take the necessary steps to design a system that asserts *privacy by design* (as per Article 22), *transparency* (as per Article 23), and security safeguards (as per Article 24).

DF may conduct *data protection impact assessment* before the commencement of the processing (as per Article 27) and will maintain up-to-date records. The record may contain activities such as details of essential operations, descriptions of impact assessment, and reviews (as per Article 28). Also, an *audit* may be conducted periodically by an independent auditor approved by DPAI to assess data processing policy, identify risks, and evaluate data protection impact assessment (as per Article 29). DF shall assign a *data protection officer* (as per Article 30) to assist in matters related to data processing and work as a point of contact on behalf of DF. The *data breaches* (as per Article 27), which may harm the data principal, shall be reported to DPAI by DF.

PDPB puts multiple restrictions on *cross-border transfer of personal data* (as per Article 33). Sensitive personal data can be transferred outside India, but DF has to store at least one copy within India. The DPAI will determine and approve which kind of sensitive personal data can be transferred. It will be based on the nature of data, India's international relations with other countries, and international agreements. Further, *critical personal data* shall be processed within India.

Based on the nature, volume, and severity of data processed, a DF or group of DF can be categorized as *significant data fiduciary* (as per Article 26). Every *social media data fiduciary* designated as an important data fiduciary may voluntarily confirm the identity of users who use their service within or from India (as per Article 28).

1.1.1 Criticality in PDPB-2019 (Draft)

The Draft PDPB-2019 had a few obligations that raised concerns. For instance:

The transfer and localization of data The bill states sensitive personal data can be transferred outside India, but the data fiduciary has to store at least one copy within India. This localization of data raised controversies in the community.

We studied thoroughly to determine if such restrictions are required. If yes, then what are the possible reasons? As per our point of view, the main reason might be that holding data within Indian territory will help in better analysis, processing, and compliance with the regulations. We argue the above with the following reasons:

- *For easy compliance of the regulation.* An investigation may require data during analysis to comply with some legal proceedings. If the storage is located outside the border, it may be denied (as per the policy of a country), or the investigation may be delayed.
- The framework is more concerned about the geo-location availability of the data and the protection of sensitive and critical personal data.
- To conduct data audits efficiently.
- To ensure a stronger availability of data and logs to analyze data breaches.
- The availability of a few categories of data outside might have significant risks, e.g., government data.

The above justification may support the localization concerns. However, it was not clear how the objective may be achieved. This uncertainty arises due to both the globalization of Internet technology and practical implementation constraints.

Lack of clarity in definition: The framework specifies many terms e.g. *significant data fiduciary, guardian data fiduciary, or social media data fiduciary*. Technologically, it would require enormous efforts to define each term accurately when the nature and size of data are changing regularly. Similarly, the boundary between *personal data, critical personal data, and sensitive personal data* is hard to decode due to increased data categories in the era of AI, machine learning, and big data.

Breach reporting : Any breach in personal data shall be reported only to DPAI, and it can be reported to the data principal in case of high risks. The choice given to DF for not reporting a data to the data principal raised concern.

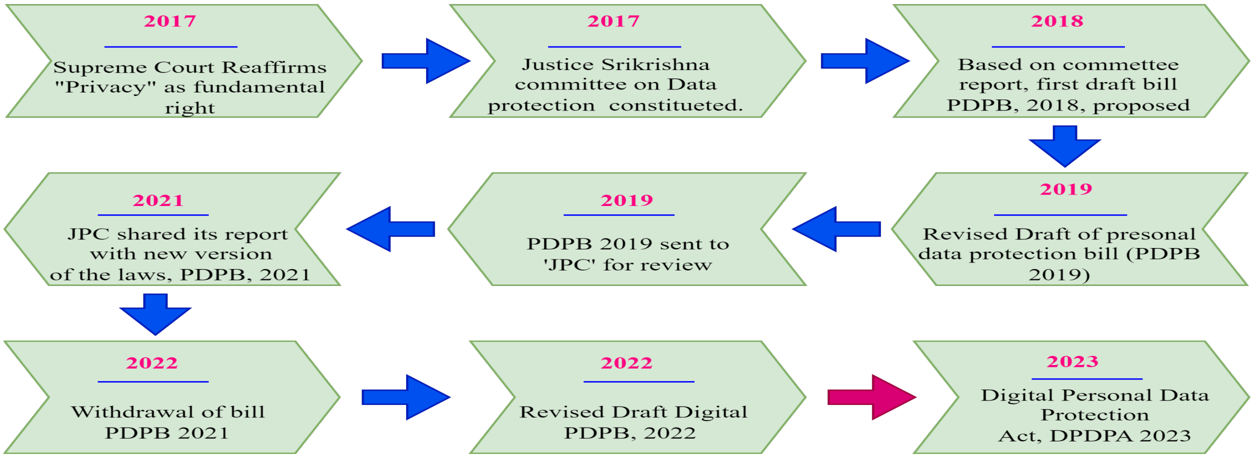

2017
Supreme Court Reaffirms "Privacy" as fundamental right
2017
Justice Srikrishna committee on Data protection constitueted.
2018
Based on commettee report, first draft bill PDPB, 2018, proposed
2019
Revised Draft of presonal data protection bill (PDPB 2019)
2019
PDPB 2019 sent to 'JPC' for review
2021
JPC shared its report with new version of the laws, PDPB, 2021
2022
Withdrawal of bill PDPB 2021
2022
Revised Draft Digital PDPB, 2022
2023
Digital Personal Data Protection Act, DPDPA 2023

Figure 1-2: Time line of the journey of Digital Personal Data Protection Act (DPDPA, 2023)

1.2 Form PDPB 2019 to Digital Personal Data Protection Act (DPDPA 2023)

The PDPB-2019 was sent to a Joint Parliament Committee (JPC) with members from both houses for review and suggestions after its introduction in the Lok Sabha. The Joint Parliament Committee disclosed their reports, including the revised draft of the bill as "The Data Protection Bill, 2021 (DPB, 2021)".

The DPB 2021 was withdrawn on 3^{rd} August 2022, and the Ministry of Electronics and IT (Meity) has released a new draft of the bill as the Digital Personal Data Protection Bill, 2022 (DPDPB, 2022) [6] and opened for public consultation and comments. In August 2023, a further revised draft that replicates and modifies the DPDPB 2022 (Draft) was introduced in parliament. This was passed by both houses and published in the Gazette on 11^{th} August 2023 as *Digital Personal Data Protection Act, 2023 (DPDPA, 2023)* [62]. The timeline journey of the act is shown in Fig. 1-2.

1.3 Digital Personal Data Protection Act (DPDPA 2023)

Many of the obligations proposed in the draft of PDPB-2019 and DPB-2021 were strapped down. DPDPA-2023 consists of the obligations of DPDPB-2022, along with a few modifications and the inclusion of additional constituents. The objective of DPDPA is to ensure that data processing respects individual's privacy rights.

The act removes ambiguous terms such as *sensitive and critical personal data* and uses a single definition as *personal data*. It highlights consent, the lawfulness of data processing, user rights, data protection impact assessments, data protection officers, the Data Protection Board of India, and other clauses. It specifies significant penalties in case of data breaches. The act removes the clause of *localization of data* and puts different restrictions categories such as *barred transfer of personal data to some specific countries*. A detailed discussion on DPDPA-2023 and the differences between PDPB-2019 and DPDPA-2023 is provided in Chapter 3.

1.4 Other Regulations

We also discuss a few other regulations studied in this research:

1.4.1 CERT-In guidelines

The regulation guideline aims at effective incident response and storage of artifacts & logs. The Indian Computer Emergency Response Team (CERT-In) has released directions under sub-section (6) of section 70B of the Information Technology Act, 2000, relating to information security practices, procedure, prevention, response, and reporting of cyber incidents for Safe & Trusted Internet [7].

The guideline emphasizes the storage of artifacts & logs of various critical devices within the cyber-infrastructure. As per this "*All service providers, intermediaries, data centers, body corporate and Government organizations shall mandatorily enable logs of all their ICT systems and maintain them securely for 180 days. The same shall be maintained within the Indian jurisdiction*". These logs will help legal and incident response compliance under the IT Act.

1.4.2 GDPR and differences

Many countries have introduced a data protection framework or legislation to protect user's data. General Data Protection Regulation-European Union (GDPR-EU) is the most familiar one that works as a legal tenet to protect user's data within EU territory [74].

PDPB was modeled after GDPR and went through multiple revisions. It contains various regulations and directions aligned with GDPR. A few differences also exist between both, mostly in exercising various obligations such as data breach penalties, cross-border transfer, definitions of significant data fiduciary and consent manager, consent collection mechanisms, and exercising user's rights. Detailed differences are provided in Chapter 3.

To best comply with the regulation, many works have been started and done in the state of the art, primarily aiming for GDPR. Their primary goal was translating legal obligations into technical solutions using various technological and engineering tools. The research works covered the following five major categories: (i) on the challenges and limitations of GDPR [83, 39, 66, 73]; (ii) on the properties of GDPR like the right to forget, and how it can be achieved [144, 139]; (iii) how changes should be done in the current system that can fulfill the requirement of obligations of GDPR [58, 78, 83]; (iv) the possible future architecture of the system complying GDPR standard; and (v) design and implementation of consent.

1.5 The Objective of Research

We study the regulation and identify two attributes:

- All versions of draft bills from 2018 to 2022 and the Act DPDPA-2023 have aim to set the legal provision for organizations to carry out data protection complied processing honoring the privacy of individuals throughout the data life cycle.
- A set of technological enhanced methods are necessary to satisfy the regulatory expectations. Organizations must modify their data processing activities and privacy policies to comply with the legislation.

The regulation will ensure legal binding and penalties, while technical enhanced methods will ensure stronger data security, safety, and privacy aligned with the framework.

Data Protection's Goal = Legal binding through legislation + Technological Compliance

This research aims are to study and analyze enhanced methods to satisfy the regulatory expectations and to enable stronger data management and processing aligned with the DPDPA framework. The study on DPDPA is selected due to the following reasons:

- As discussed in Sec 1.4.2, much research has already started aiming for compliance with the GDPR framework. However, minimal research work has been done in the context of DPDPA. This is because the bill is under process and in the draft stage. The DPDPA also requires research from a compliance perspective.

- There are a few differences between GDPR and DPDPA. Many GDPR-aligned research studies cannot trivially apply to DPDPA. For instance, nature and scope of consent definition vary in both framework, thus requires an independent study.
- From technological implementation aspects, several challenges must be solved in GDPR and DPDPA.

We emphasized the followings:

- What are the obligations' properties that need stronger technical compliance?
- What technologies might be required to satisfy the conditions given in the act?
- How can organizations implement the data management standards mentioned in the act?
- How to make it easier for individuals to verify that the terms and conditions are honored?
- How to make it easier for law enforcement agencies and regulatory authorities to validate claims in the case of a dispute during the enforcement of the policies?

1.6 Problem formulation: Designing of systems for technological compliance of DPDPA

With the above objectives, this research aligns the technical aspects of the framework and suggests ways to address the different clauses of the bill. The study includes all versions of the bills (draft one and passed act DPDPA 2023) and mainly explores cryptographic and security solutions for enhanced data management and processing.

Total four obligations (User's consent, Right to nominate/ Right to access, Data Breaches, and Data Storage/Logging) are considered from the bill; refer Fig. 1-3. Each obligation concerning technological compliance is studied in detail.

Analysis of each vertical is done primarily in two parts. First, clauses in the vertical are identified, the security and privacy requirements are derived, and a problem statement is formulated. Second, appropriate techniques have been described against properties derived from the previous step. The simulation and result analysis are done for the models. Below, we describe the obligations studied [141, 6, 62, 7] and formulate the problem.

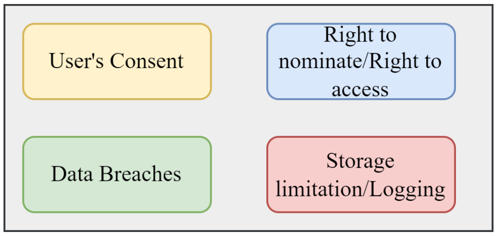


Figure 1-3: DPDPA obligations selected to study

1.6.0.1 User's Consent

PDPB states the nature and scope of the consent. DF must notify the data principal regarding the data collection and should obtain consent (Article 5(1)). The notice to the DP shall include the purpose, the manner for exercising consent, and the manner of making a complaint to the board (Article 5(1) clauses i-iii). Further, Article 6(1) states: *"consent shall be free, informed, specific, unconditional and unambiguous"*. Data principal also has the *right to withdraw their consent* (as per Article 6(4)), right to access (Article 11), right to correction and erasure (Article 12), and right to grievance redressal (Article 13). Next, Article 6 (10) states: DF *"shall bear the burden to prove that a notice was given and the consent was established with DP"* to prove the transparency. Finally, the data fiduciary has to implement an entity named *consent manager* through which he can record, manage, and process consent compliance (Article 6(7, 8)).

Compliance-based consent processing after data protection regulations has been one of the active parts of the research in the last few years. The existing research primarily emphasizes consent properties, challenges, study on the properties of cookies consent, and approaches of compliance methodologies. It includes: ontology-based consent and semantic interoperability [33, 136], Blockchain-based consent platform [145, 176], consenting to different IoT devices [89, 133, 49], consent validity [46], dark and bright pattern of consent [130, 81], commodification of consent [182], uninformed consent, consent notices and effective consent enforcement [56, 182, 117, 131], consent driver and obstacle, consenting communications, purpose and necessity and challenges in adopting algorithmic consent [115, 112, 96, 142, 76], consent properties (uninformed consent, consenting through user interface and dialogues, affirmative consent, consent with icon and link text) [175, 135, 97, 85], consent compliance and processing(auditing consent, generating data set from consent, usable and auditable web consent, consent awareness through gaming and graph, consent compliance verification) [39, 57, 99, 146, 107], cookies consent and web privacy [58, 137, 44, 123, 95, 101, 91, 84, 45, 106, 114, 61], and consent use case models [63, 69, 150].

The existing techniques of consent processing are not transparent and do not adhere to data protection goals. Data fiduciary may misuse the collected data for purposes other than specified in the consent. A problem statement is studied on consent properties and solved in Chapter 4.

1.6.0.2 Right to access/ Right to Nominate

Article 11(1) states the DP has the right to obtain a summary of personal data, the list of DFs and DPRs processing its data, and any other relevant information. Also, as per Article 14 (1), each data principal has *right to nominate* someone as a nominee. It states: "*A DP shall have the right to nominate, in such manner as may be prescribed, any other individual who shall, in the event of death or incapacity of the DP, exercise the rights of the DP in accordance with the provisions of this Act and the rules made thereunder..*".

The earlier work in the direction of "Digital Asset Inheritance (DAI)" was concentrated on the issues and challenges of existing laws and policies. The uniform digital asset law proposed under the Revised Uniform Fiduciary Access to Digital Assets Act (the "RUFADAA") authorize personal representative of descendants to *access* digital assets [179]. However, this accessibility does not imply *inheritability* (transfer of digital asset after death); hence, an adequate law and policy is needed for effective inheritance [149, 104, 134]. A successful digital asset transfer should also include other perspectives like legalization by countries, proper practical planning, and the analysis of failure consequences [122], [53, 94]. It should be handled similarly to other assets but using a better-defined legal system [108, 77]. Many works have analyzed whether the country's existing laws and regulations are applicable to inherit the digital asset [125]. Recently, a few asset inheritance models have been proposed using Blockchain, such as PassOn [168], DigiPulse [166], SafeHaven [169], TrustVerse [170]. However, all these have poor technical descriptions and have many existing challenges, such as applying only to a few asset categories, the possibility of recreation of the incorrect key by a nominee, or a weak method of death confirmation.

Following the right to nominee obligations of DPDPB, a problem statement is studied and solved in Chapter 5. We have formalized the different categories of digital assets and defined the various security goals, required functionalities, and necessary entities to build an asset inheritance model. We have also proposed a new protocol named digital asset inheritance protocol (DAIP) using certificateless encryption (CLE) and an identity-based system (IBS) to convey the user's online persona efficiently to the descendant after his death.

1.6.0.3 Data Breaches

The framework specifies the service provider's responsibility to process the data lawfully and comply with the regulations. Data breach is an essential clause in the bill and has acquired the utmost attention. Article 8(5) states, "DF or DPR shall implement security and safeguards measures.." adhering to the framework to protect the data to prevent data breaches. In case of a data breach, DF or DPR shall notify the board and each affected DP (as per Article 8(6)). The provision of penalty is also mentioned by the framework, which includes penalty in case of failure to provide reasonable security and safeguards causing data breach or failure to notify the board or affected data principal in the event of data breach (as per Article 33(1)).

The data breach is studied from various aspects such as: [119, 127] discuss why we should care about the targeted data breach and prevent businesses from breaches. The [152] explain the consequence of a breach, the role of data breach disclosure[36], preventing data breach in higher education [184], data breach prevention of COVID-19 data [59], the role of top management in data breach security [155]. Few studies have been done on data breach risk prediction [68] and prediction of the possibility of breach [38].

As high penalties may imposed in case of a breach, service providers should design and implement systems with adequate security mechanisms to prevent it. Also, as prevention may not always be guaranteed, and a breach may happen, a corrective evaluation is necessary to minimize penalties and mitigate future risks. The state of the art does not emphasize reactive strategies much after a data breach. Therefore, we explore methods that can help in the correlation, validation, and assessment of data breaches and their impact. We studied the problem in detail in Chapter 6.

1.6.0.4 Logging

The cyber security regulation emphasizes [7] that enabling different kinds of logs is mandatory for legal compliance, cyber security analysis, and incident response.

Different applications and methods are already available over the internet to collect the browser's historical evidence and its analysis [118, 128, 147, 32]. However, many systems run in cyberspace as standalone or small network segments with limited users. The systems are not equipped with adequate network-level security monitoring solutions. The synchronization of threats accessible to the individual, along with their impact and mitigation steps, maybe more helpful in threat detection. In Chapter 7, we study a problem on how the notable threats can be easily communicated and shared with users and how the log analysis may be efficient for early detection of incidents.

1.7 Our Contribution and Organization of Thesis

The rest of the thesis chapters are organized as follows: In chapter 2, we describe a detailed list of surveys on literature work. The rest of the chapters are organized as follows:

- **Chapter 3: Background and technological compliance necessity**. In this chapter, we have analyzed various obligations such as DP's consent, data collection, data processing, security by design, transparency, and data audit. For the above analysis, we have described how cryptographic (such as encryption, signature schemes, zero-knowledge proof, etc) and other solutions (such as anonymization, de-identification, access control, etc) can be used for technical implementation. We also highlight challenges where existing solutions can not provide efficient compliance. For such challenges, advanced methods have to be explored.

- **Chapter 4: Transparent consent data processing**. This chapter describes that encoding requisite security and privacy properties will ascertain stronger consent compliance. We formalize these properties as Proofs of Consent (PoC) and categorized them into three layers. Acquiring a higher layer will minimize adversarial risks and ascertain greater transparency. Based on this, we have proposed a model-shielded consent manager (SCM) using Blockchain and other cryptographic primitives to retrieve consent to grant permissions to access Android resources. SCM includes parameters as per the framework, satisfies the security properties such as integrity of consent, non-deniability by users, auditability of logs in data processing, and provides finer visualization of user's consents.

- **Chapter 5: Digital asset inheritance**. In this chapter, we have formalized the different categories of digital assets and defined the various security goals, required functionalities, and necessary entities to build an asset inheritance model. We have also proposed a new protocol named digital asset inheritance protocol (DAIP) using certificateless encryption (CLE) and an identity-based system (IBS) to convey the user's online persona efficiently to the descendant after his death. DAIP allows the nominee to successfully retrieve the asset after the user's demise, even if a nominee is uninformed regarding the asset. We, then, provide rigorous security proofs of various properties using a real-world–ideal worlds paradigm. Finally, we have implemented the DAIP model using PBC and the pycryptodome library. The simulation results affirm that it can be practically efficient to implement.

- **Chapter 6: Data breach assessment**. In this chapter, we propose the system model of a Data Breach Incident Assessor (DBIA) for breach evaluation that can assess and respond to data breach incidents (if it happens) within the organization. DBIA helps validate a threat actor's claim, understand the root cause of a breach, and analyze the scope of the compromise for mitigation of security gaps and robust compliance under DPDPB. The design of DBIA is simulated as a security information event management system. The simulation results and discussions show the necessity and efficacy of the model.

- **Chapter 7: ESUL analyzer**. In this chapter, we propose the model of an End System URLs Log (ESUL) analyzer for URL-based threats present in standalone systems. The model continuously analyzes the user's browser history logs of the End system (ES) and announces the list of malicious URLs, if visited previously, based on a received adversarial list. This early threat identification from log data will help end users learn about threats, perform incident response, and minimize their impact. It also assists users with relevant advisory and best practices. The model is simulated using a phishing database library, and the results describe its efficacy.

- **Chapter 8: Conclusion and future work**. We summarize the existing research and discuss future work.

1.8 List of Manuscripts and Publications

1. Ram Govind Singh and Sushmita Ruj. "A Technical Look At The Indian Personal Data Protection Bill." in *"Arxiv Preprint"*, 2020. Available at: https://arxiv.org/abs/2005.13812.

2. Ram Govind Singh, Ananya Shrivastava, Sushmita Ruj. "A Digital Asset Inheritance Model to Convey Online Persona Posthumously." In International Journal of Information Security, pp. 983-1003. Springer, 2022. DOI: 10.1007/s10207-022-00593-8

3. Ram Govind Singh, and Sushmita Ruj. "Encoding of security properties for transparent consent data processing." In 2023 IEEE Guwahati Subsection Conference (GCON), Guwahati, India, 2023, pp. 01-08, DOI: 10.1109/GCON58516.2023.10183463.

4. Ram Govind Singh and Naveenkumar D. "ESUL analyzer for inceptive threat identification and mitigation." In 2023 Third International Conference on Secure Cyber Computing and Communication (ICSCCC), Jalandhar, India, 2023, pp. 409-414, DOI: 10.1109/ICSCCC58608.2023.10176536.

5. Ram Govind Singh and Naveenkumar D. “Are we undermining data breaches? Protecting education sector from data breaches.” In 2023 IEEE International Conference on Computer, Electronics and Electrical Engineering and Their Applications (IC2E3), Srinagar, Uttrakhand, India, 2023, pp. 1-6, DOI: 10.1109/IC2E357697.2023.10262570.

6. Ram Govind Singh, and Sushmita Ruj. “Evaluation and assessment of data breaches for robust compliance under DPDPB.” SPRINGER ICISS, 2023. Manuscript Submitted.

Chapter 2

Related Work

2.1 Users privacy and data protection

The privacy of the user's data has been discussed for decades. Cavoukian has proposed the idea of seven foundation principles that may be implemented to achieve privacy by design in the system [50]. Earlier development of the privacy-oriented application was user-centric. It means that the application should have privacy-oriented features so that users can control their privacy and secure personal data. Currently, the accountability is shifted toward Service Providers (SP). SPs are encouraged to enable services in such a manner that can maximize user's privacy and data protection.

The earliest privacy laws were formulated by France (1978) [12] and Canada(1983) [21]. Later, Australia (1988) [22], New Zealand (1993) [23] and USA (HIPAA,1996) [65] have also formulated privacy acts to set regulations to collect, use, disclose, and share the personal information. Similarly, California's new privacy law, California Consumer Privacy Act(CCPA) [5], gives rights to the consumer to take back control over their information from the business.

GDPR[74] is the most popular one, a worldwide recognized framework, and regulated in the European Union. PDPB [141] is also one step towards data protection. Very little has been discussed about PDPB in the state of the art, as the bill was in the draft stage. Since PDPB was introduced after GDPR and has a good deal of similarity, we have described some of the earlier work done for GDPR.

The biggest challenge of the data protection framework is implementing technological solutions to satisfy the legal expectations of the obligations. The effort involves stakeholders such as legal experts, law enforcement agencies, software architects, developers, requirement analysts, and security and privacy experts collaborating coherently [50, 82]. The work done in different areas has been categorized in Table 2.1. There is excessive antagonism between the development style of products in the software industry and the legal tenets of the data protection framework. A dichotomy between GDPR standards and system design perspective has been discussed in [156]. GDPR obligations like data storage, data deletion, and data reuse are challenges in the real world. Gruschka *et al.* [83] have discussed data protection challenges while processing big data and suggested using anonymity and de-identification techniques while processing big data. Esteve [66] has discussed the use of personal data by Google and Facebook for advertisement and business purposes and how this will affect the protection of personal data under GDPR. Fuller [73] argues why privacy is a failure till now. The debate is whether a company should collect data freely to provide a service or impose a fee for the service, in turn, to protect privacy. Similarly, [39] has analyzed GDPR challenges and limitations concerning whether it is for a purpose or a necessity.

Table 2.1: Related work in the area of personal data protection

Area	**References**
Challenges and limitation in achieving goal of Data Protection	[156], [83], [39], [66], [73], [82]
Architecture	[159], [92]
Data Protection Properties	[144], [139], [41], [175]
Review of implementation of GDPR Policy in the system	[58], [78], [83]

After implementing the data protection framework, each data fiduciary has to change its system according to the framework's legal requirements. Hjerppe [92] and [159] have analyzed and proposed software development models and threat models that can be implemented as per the obligations of GDPR. In the other category, few works have analyzed the properties of the framework. For instance, [139] has examined the importance of the "right to forgotten" covenant of GDPR. Similarly, [144] has discussed the deletion of data stored in the blockchain from the perspective of the "right to forget" of GDPR. [161] have analyzed the efficacy of the present system after GDPR. [58] has examined the cookies privacy policy by various websites after the implementation of GDPR; [78] has described the use case of a cyber trust project for the security of the smart home environment. They have shown that processing personal data in the cyber trust project follows GDPR's policy. Likewise, [83] has analyzed the use case of two projects "SWAN" and "OSLO". They have shown that the GDPR policy is implemented effectively to process big data, and proper anonymization and de-identification have followed.

This work is aligned with PDPB. We have described technical challenges for a few obligations and argued that properly implementing cryptographic and security methods could enable stronger compliance. We also discuss how the encoding of security solutions can be useful in satisfying the goals of PDPB.

In the following sections, we discuss the studied state of the art aligning with obligations such as consent, right to nominee (asset inheritance), and data breaches.

2.2 Consent processing

It is a new algorithmic era where some regulatory framework will regulate personal data. Consent is defined as one of the basis for data processing. Compliance-based consent processing after data protection regulations has been one of the active parts of the research in the last few years. The Table 2.2 shows a few works associated with the consent processing. The existing research primarily emphasizes consent properties, challenges, study on the properties of cookies consent, and approaches of compliance methodologies. A summary of these are highlighted below:

Consent and ontology: Aiming towards a better design of consent that can aligned with the regulatory framework, a few research studies have proposed a consent model based on ontology [136, 70, 57]. This semantic consent model may fulfill GDPR requirements using the open vocabulary of ontology to comply with provenance, process, permission, and obligation properties and can be used for a structured representation of consent. The ontology-based consent may also be used in inter-operable heterogeneous environments [33]. For this, the ontology should define the roles of different entities, mechanisms for collecting consent, permissions for users and data owners, and protocols for sharing data with third parties. Then, it should define a standardized approach to achieve interoperability, access control, and data sharing based on the diverse nature of data categories and varying levels of access control owned by different entities in this heterogeneous ecosystem. The proposed ontology model defines the parameter satisfying the above, draws inspiration from GDPR requirements, and facilitates semantic interoperability, the federation of deployments, and the development of privacy-preserving applications. The ontology model was analyzed for simulated IoT federations ecosystem. However, depending on the association with real-world entities and the purpose of data processing, this privacy-enhanced ontology can be easily adapted for use in generic IoT systems.

Consent under GDPR requires information such as how the consent was obtained, the temporal parameter, and how the consent was modified over time. The existing techniques do not include all the information. The model of GCOnsent proposed in [136] uses ontology-based modelling. Firstly, it identifies attributes and information related to consent. Then, it uses OWL2-DL, an ontology-based library, to represent consent complying with GDPR and having the properties of provenance. This means that the properties may be used to validate the consent processing. The ontology-based model does not discuss if the service provider is malicious and generating incorrect data.

Consent and blockchain: Blockchain [4] is a distributed ledger that may be used to store and process data while providing immutability, verifiability, and transparency. Few studies state the blockchain may be used when parties establish consent [145, 176, 174]. However, using blockchain in consent processing is very early and requires further restructuring and redesigning.

Consent in IoT environment: IoT devices have extensive functionalities and pose a significant privacy risk. The model [89] performs a privacy study on IoT devices available at home, assuming a scenario of how to consent to devices in a common location such as a house or office. The model generalizes the scenario to design a probability density function that indicates the probability of consenting to devices based on the sensors and the user's awareness preference.

A discussion overview on practical IoT data collection and sharing approaches is provided in [133]. It defines the four levels (category) of consent as general consent, general consent with specific conditions, general denial with particular conditions, and general denial. However, the model does not provide an algorithmic model for consent retrieval in IoT settings with these categories.

The [49] mentions that all necessary data collected from IoT devices should be communicated to data fiduciary and all data subjects in their range. The consent mechanism is discussed using two scenarios: i)direct communication between data collecting devices and a device carried by the user, such as a smartphone (working as a gateway device). It is claimed that the "direct declaration" may enable transparency and assumes consent may be performed locally without internet connectivity. However, the direct communication mode has challenges as all devices must declare their presence and have privacy policies enabled. The communication range of privacy policies should match the operational range of the device collecting data. ii)registry-based solution: to enable transparency, the devices should declare their presence through a registry. The declaration may include privacy policies, duration, location, range of collecting devices, and any other information the law desires. The registry must be managed correctly, up to date, and accurately, and it may be designed in a centralized or decentralized manner.

Consent validity, pattern & observations: Few researchers analyzed policy of the current consent concerning the validity of context [46], demonstrated the influence of pop-ups, and described other observed patterns [130].

Consent is a complex philosophical principle that relies on the person giving consent fully possessing the fact. The work [46] explores the philosophical background of consent, examines the circumstances that are the point of departure for debate, and attempts to understand the impact of the growing influence of information and a data-driven economy. Further, the work in [130] evaluates the dark patterns after GDPR and analyzes scraping consent pop-ups and their influences. A web scrapper is designed to collect consent management platforms (CMP) visual elements, interac-

Table 2.2: Related work under consent processing

Area	**Description**	**References**
Ontology based consent	Consent ontology, semantic interoperability of consent	[33, 136]
Blockchain and consent	Blockchain based consent platform and consent manager	[145, 176]
Consent in IoT	Consenting to different IoT devices, informed consent in IoT, enhancing transparency and consent in IoT	[89, 133, 49]
Consent validity, pattern & observation	Consent is valid?, dark and bright pattern of consent, commodification of consent, uninformed consent, consent notices, effective consent enforcement	[46, 130, 81, 56, 182, 117, 131]
Consent challenges	Driver and obstacle of adoption of consent, consenting communications, purpose and necessity, challenges in adopting algorithmic consent	[115, 112, 96, 142, 76]
Consent properties	Uninformed consent, consenting through user interface and dialogues, affirmative consent, consent with icon and link text	[175, 135, 97, 85]
Consent compliance and processing	Auditing consent, generating data set from consent, usable and auditable web consent, consent awareness trough gaming and graph, consent compliance verification	[39, 57, 99, 146, 107]
Cookies consent and web privacy	Cookies analysis, cookies tracking and bypassing consent, evaluation of privacy notices, corralling cookies using CookieMonster, CookierEnforcer, cookie based tracking	[58, 137, 44, 123, 95, 101, 91, 84, 45, 106, 114, 61]
Consent use case models	Privacy CURE platform, privacy dashboard analysis, consent management platform	[63, 69, 150]

tion design elements, and text keywords. The scrapper extracts attributes such as notification style (banner or barriers), types of consent (implicit or explicit), user's action (consenting or visiting), navigation/reloading/scrolling/closing/clicking, existence of accept/reject button, etc. A statistical analysis is performed over the collected parameters to evaluate compliance status.

Similarly, the [81] analyzes the dark and bright patterns in cookie consent requests. Dark patterns in consent collection are (evil) design nudges that manipulate users' actions through persuasive interface design. The model analyzes the effect of four common design nudges (default, aesthetic, manipulation, obstructive) and the user's consent decision. To evaluate dark and bright patterns, the study is done with a few hypothesis categories, e.g., users will be more likely to choose the privacy-unfriendly/privacy-friendly option (compared to privacy-friendly) when the privacy-unfriendly option is pre-selected and visually more salient or the alternative (privacy-friendly) the option is obstructed. The other hypothesis is that participants report lower perceived control over their data when the privacy-unfriendly/privacy-friendly option is pre-selected and visually more salient, or the alternative (privacy-friendly) option is obstructed. A questionnaire was created, and an experiment was done with a sample size of 228 users. It was concluded that the service providers are not enabling meaningful choices to the users, thus not complying with the law.

Solid projects decouple data services and applications from data storage. An Open Digital Rights Language (ODRL) and other specialized policies have been applied to extend Solid's authorization mechanism in [56] to incorporate the consent process. Another work described in [182] mentions that consent may be considered an asset by firms and may be traded across organizations. A study is done from the perspective of commodification of consent. It is evaluated that the user's consent as legal provisions between the user, publisher, and third parties will change the distribution of revenue and shares.

A concern of consent complying with GDPR is raised in [117]. The study evaluates the difference between opt-in methods of consent and the user's growing interest in using automated extension tools to opt out of the consent. The work also does a statistical analysis by providing a coupon for the website if users have the Willingness To Sell (WTS) cookies consent. It shows that many people were deviant and willing to sell the data. In [131], four key characteristics "freely given, specific, informed, and unambiguous indication" are selected from GDPR. The study was done on 10000 websites for 26 days using a Python simulation. The simulation automatically crawls the website as the HTML structure is analyzed to determine consent criteria. The criteria from the website, such as the existence of consent buttons, absence of consent buttons, separate consent, accessibility to the privacy policy, and absence of consent agreement, have been used.

Consent challenges: The impact of the user's consent dialogue is studied in [115] as consent properties create a design space for consent dialogue. The service provider aims to maximize click rates and provide positive consent decisions even with the risk of users accepting more purposes than intended. The work analyzed the user's consent decision deviation based on several choices and the presence of a highlighted button ("select all"). The statistical analysis shows the impact using four hypotheses as the user consent dialogue has a highlighted button "select all" then: i)if it selects more purposes than necessary; ii) whether users regret their decisions after opting "select all" choices; iii) the user feels website is more deceptive with all choices; iv) consent with multiple purposes and choices requires more efforts. The statistical analysis is evaluated on the parameters of Perceived Deception (PDE), Perceived Difficulty (PDI), and Privacy Attitudes (PA). In the cookies scenario, PDE is defined with three parameters: PDE1 (the website is dishonest towards its users), PDE2 (tries to mislead users towards selecting cookie settings that they do not intend to select), and PDE3 (the website makes use of misleading procedures so that users select cookie options which they do not intend to opt). Also, Perceived Difficulty (PDI) is defined by three parameters: PDI1 (it was incomprehensible), PDI2 (it was frustrating), and PDI3 (it was easy) to select cookies settings. The Regret (RE) attribute is defined by three parameters: RE1 (I regret my choice of cookie settings), and RE2 (I would change my cookie settings if it were possible). RE3 (I am satisfied with my choice of cookie settings). Privacy Attitudes (PA) are defined by three parameters: PA1 (it is important for me to protect my privacy online), PA2 (the privacy is impaired), and PA3 (the user is concerned that cookies are impairing his online privacy). The participants were divided into deception, reduced choices, and a control group. The analysis is done to evaluate if the deception group and the control group differ in the number of purposes they effectively agreed to, the difference in regret, and the differences in privacy attitude.

WHOIS database records information of domain registrants. After GDPR, certain registrant information is redacted before being disclosed to the public. The study in [112] aimed to quantify the changes and evaluate the impact of data redaction on other applications relying on WHOIS data. The analysis is done on 1.2 billion WHOIS records collected for two years. An application named GCChecker is created that assigns compliance scores to the domain. The study also quantifies which portion of records are found to be redacted. For instance, it has been concluded that 60% WHOIS data provided also redact non-EEA records.

A framework Data Protection and Consenting Communication Mechanisms (DPCCMs) [96] aims to standardize consent processing mechanism to allow users to express their choices and preferences, to manage their online consent complying regulatory frameworks, and to improve dialogue-based current consent retrieval methods (as current model assumed problematic). The work provides existing challenges with two open proposals, i.e., GPC(Global Privacy Control) and

ADPC (Advanced Data Protection Control). GPC is based on unary signals and has a single state of expression usually initiated by the server. ADPC is a bidirectional communication mechanism that can be initiated by either websites or users. It can express multiple distinct values regarding the purposes for which consent is given or withheld and object to direct marketing and legitimate interest. This study specifies technical specifications of both models and mentions the existing challenges in achieving DPCCMs goals.

Another framework for consent processing known as "Transparency and Consent Framework (TCF)" is proposed. One proposal under this framework is to use a Consent management platform (CMP) to collect and manage consent. The use of the CMP platform is growing and being adopted by industries. CMP gathers users' consent on behalf of SP. Later, SP and other publishers can use this consent to process personal data. After GDPR, many CMP platforms became available in the market and were enlisted in the global vendor list (GVL), a list of vendors approved by EU-GDPR that allows the CMP platform as a service. The study in [142] describes TCF, GVL, and CMP and highlights compliance risk and lack of standardization. The analysis is based on the interview with the representatives of GVLs and mentions the existing challenges, doubts, and market pressures about TCF's compliance.

A well-defined consent requires technological means. The study done in [76] maps out the difficulties in applying traditional consent models to data-driven algorithmic systems satisfying the goal of consent processing. The work studies how effectively the obligations can be mapped as algorithm consent. For instance, one of the consent criteria is consent should be valid and informed. It is argued that the consent criteria are hard to reconcile in the era of big data and AI because valid consent implies that users understand the facts and consequences of the consent process. Similarly, the properties of consent derived from GDPR and the challenge in adoption as an algorithmic form are discussed. It is suggested that the gaps between theoretical consent processing and algorithm processing can be filled using methodologies such as bottom-up data governance, rethinking design choices, etc.

Consent properties: Few of the research evaluate consent properties. The work [175] analyzed the properties of consent notices. Multiple properties have been extracted from the consent notice, such as the size of the notice, the position of notice(top left, top-right, bottom-left, bottom-right), choices (no option, confirmation only, binary, category-based notice (e.g., slider) vendor-based notices), nudges and dark patterns, hyperlink to additional information, etc. The analysis performs three experiments for 82890 unique website visitors: i) the position of the notices; ii) several choices, neutral position vs nudging; iii) the impact of (Non) technical language and privacy policy link.

Following the TCF, GPC, and ADPC proposal, the study in [135] evaluates consent properties and urges redesigning the consent processing issues through signals and user-side dialogues. This can be achieved: i) with the use of automation through privacy signals to better govern consenting processes and to reduce “consent fatigue”; ii) with the generation of consent dialogues on the user side and its practicalities for both websites as well as users and agents (e.g., web browsers).

The idea of affirmative consent is proposed in [97], which is built on five core concepts: consent is voluntary, informed, revertible, specific, and unburdensome. Based on these principles, it is urged affirmative consent is both an explanatory and generative theoretical framework for consent processing. The work in [85] discusses the impact of consent properties in choices and preferences; if consent notices, use the sign of special symbols such as toggle, dollar, text link, and triangles as privacy choice indicators. The results suggest the necessity to implement and use privacy choice indicators.

Consent compliance and processing: To evaluate GDPR compliance of consent, a formal model-based approach proposed in [39] identifies the purpose associated with a business process and depicts how it can be used to audit and validate the privacy policies. The study defines the data collection process as an “inter-process communication” among parties (e.g., vendor, customer, business partner, etc.) and shows the audit process using an IPCs graph.

As a compliance, service providers must ensure that they comply with their policies. One approach to verify compliance is to perform an audit of logs. The research in [57] emphasizes that compliance can be validated through the in-time generation of the data set before even analyzing the logs. It uses RDF and OWL ontologies to annotate schema, allowing it to generate declarative mappings that transform (relational) data into RDF driven by the annotations. This can be used to create compliance data sets by altering the mapping results. Using RDF and OWL allows the implementation of the entire process in a declarative manner using SPARQL. All components are integrated as a service that further captures provenance information for each step. The approach is demonstrated with a synthetic dataset simulating users (re-)giving, withdrawing, and rejecting their consent. It is argued that the model facilitates transparency and compliance verification from the start, thus reducing the need for post hoc compliance analysis.

For compliance of consent, [99] propose the idea of “consent receipt”. It is a kind of proof that the service provider will have to store to show consent was established. The study mentions the feasibility of how consent receipt can be implemented. For example, they are using consent dialogue, with the use of Amazon Alexa, or with the acceptance of privacy policy. For compliance, another study done in [146] proposes the model of consent awareness through gamification and graphs. It is shown that the knowledge graph models can depict consent in readable graph format and provide a unified consent model to all parties associated with data processing. The analysis also describes that a gamification interface can raise individuals' awareness about legal compliance. [107] analyze GDPR consent compliance for websites sending marketing Emails. Emails received by registration on 5000 websites have been used for study and evaluated a potential violation of privacy during compliance.

Cookies consent and web privacy: A few works analyze cookies and their distinct parameter in the context of consent and user privacy. For instance, [58, 137] has analyzed the impact on privacy when cookies are collected and tracked by the data fiduciary. The work [44] proposed a browser extension, called "CookieBlock". It uses the machine learning algorithm to enforce GDPR cookie consent at the client. The extension automatically categorizes cookies by using only the information mentioned in the cookie itself and automatically sets the consent for users based on user choices and preferences. Further, an analysis is performed on the top 116 websites of the European Union in [123], and the existence of an asymmetry is identified when consent is collected through different platforms, e.g. websites, browsers, and mobile apps. It has also been identified that many websites start collecting users' data when the website or app starts running, even without waiting for users' consent.

Earlier cookies were used to preserve the state of users, but nowadays, they have more business purposes, such as user behavior monitoring, profiling, and tracking. The research proposed in [95] uses the data set from the Cookiepedia portal. The data set has four necessary categorizations: performance, functionality, and targeting/advertising cookies. Initially, an analysis is performed over the top 20k websites from the Alexa list to evaluate how the website's cookies are classified under four categories, and identified that only 22% model "CookieMonster" is proposed, which can categorize cookie data into one of the four categories mentioned above with more than 94% F1 score and less than 1.5 ms latency.

On cookies consent, a model "CookieEnforcer" is proposed in[101] that automatically discovers cookie notices, processes them, and disables all non-essential cookies. To implement this, the consent cookie notices are rendered from HTML, supplied as a machine-readable task, and produced output as a set of clicks to be made.

The growth of the CMP platform is studied in [91]. The analysis crawls 161 million unique domains and concludes CMP adoptions have significant growth, which doubled from June 2018 to June 2019 and then doubled again until June 2020. It is also analyzed that CMP adoption is more prevalent in popular websites. Further, a discussion is made on how choices and preferences vary between two popular CMPs.

The evaluation of cookie consent interfaces to identify dark patterns and users' awareness about choices, etc, is performed in [84]. Dark patterns are design practices used to get fewer privacy-protective options. They could lead to users unknowingly consenting to data collection or failing to exercise their preferred privacy choices. Dark patterns and manipulation of data and consent are also studied in [45]. A survey is analyzed to measure the impact of GDPR on World Wide Web cookie banners and privacy policies in [106]. Similarly, the theory of "prospective Consent" and its effect on the framing cookie consent decisions is analyzed in [114]. Prospect theory describes how users' behavior changes based on risk factors. The risk involves accepting or denying cookies consent, access, or selling personal information, etc. The research study shows "how the slant of a cookie consent banner and the framing of a banner" impacts users' decisions. Next, how the pattern of cookies-based tracking of Facebook changes between 2015 and 2022 is examined in [61]. It was summarized that cookie policy implementation is incomplete.

Consent use case models: A customized user consent collection interface has been designed as "CURE" (Consent reqUest useR Interface) in [63] allowing users to submit the choices based on the choices complying GDPR. The impact and usefulness of Google's privacy dashboard are explored in [69]. It is deducted that these privacy dashboards are suitable for users to manage their privacy choices and preferences. Whether CMP should be treated as a data processor or data controller under TCF, GVP, or ADPC [150]. Multiple scenarios have been explored wherein CMPs process personal data and may be considered controllers.

This proposed work described how consent establishment should be designed using security and privacy goals and PoC. The fulfillment of PoC will ensure strong compliance in the data processing.

2.3 Asset inheritance

The existing state-of-the-art focuses research on digital asset inheritance from different facets, and Table 2.3 depicts a few categorizations.

The earlier work concentrated on the issues and challenges of existing laws and policies of digital asset inheritance systems. The uniform digital asset law proposed under the Revised Uniform Fiduciary Access to Digital Assets Act (the "RUFADAA") authorize personal representative of descendants to *access* digital assets [179]. However, this accessibility does not imply *inheritability* (transfer of digital asset after death); hence, an adequate law and policy is needed for effective inheritance [149, 104, 134]. A successful digital asset transfer should also include other perspectives like legalization by countries, proper practical planning, and the analysis of failure consequences

[122], [53, 94]. It should be handled similarly to other assets but using a better-defined legal system [108, 77]. Many works have analyzed whether the country's existing laws and regulations are applicable to inherit the digital asset. For instance, a study has been done on Estonian government regulation to know in what context their laws are applicable for an heir to make him eligible to access the account of a deceases to download data [125]. A similar study has also been done on the United Kingdom's regulation [87]. Another essential concern for a robust DAI model is the *privacy* of user's data. The regulation needs to fix how the assets should be passed to the descendent while maintaining the privacy of the digital asset owner [93]. For instance, to preserve privacy, data that is financially worthwhile, publicly published, or beneficial in the future may be transferred (with the condition that privacy concerns no longer hold in the future). Any other digital asset that violates the user's privacy must be destroyed [126]. Also, the concern regarding postmortem data privacy, data ethics, and property rights of the user after death has been addressed in [88, 138]. In all these studies, we observe that digital asset inheritance is a growing concern in society, and reform in the law and some practical solution is required for this indispensable subject.

Table 2.3: Related Work under DAI

Area	**References**
Digital asset existing law, policy, issues and challenges	[108], [122], [53], [149], [77], [179], [94], [104], [134]
DAI Problem studied from country perspective	[40], [140], [125], [87],
Organization Policy	[120], [37], [47], [103], [132], [148], [151]
Technological models	[166], [169], [170], [168], [51],
Users privacy	[93], [126], [88], [138],

The possible steps regarding the practical implementation of DAI are being analyzed in [120, 132]. The organization's public policy determines the asset's future in the current system. For instance, Facebook uses a *postmortem data management* policy that preserves the deceased's memory, allows posting about the user's death by nominated friends, and grants memorializing practices [47]. Similarly, Google uses an inactive account manager to inform and delete the account of the dead. Twitter allows retrieving the user's data by their successors from its portal. In the absence of public policy within the organization, the future of the user's account is determined by the company's "Terms of Service Agreement" (ToSA) [103]. The asset may still be denied if ToSA is not implemented correctly [148]. However, these are not full-fledged inheritance models. They may allow access to the user's accounts to a certain extent but do not implement a dedicated *inheritance* model. Thus, an efficient model is necessary to implement DAI effectively.

Designing technical implementation of asset management and asset inheritance is at a very initial stage. Recently, a few asset inheritance models have been proposed using Blockchain, such as PassOn [168], DigiPulse [166], SafeHaven [169], TrustVerse [170]. In DigiPulse, the user can store sensitive data at the DigiPulse portal in the encrypted form and hash of files on the blockchain to maintain integrity. The encryption key can be distributed among nominees using secret sharing schemes. PassOn converts various assets into tokens and stores them on the blockchain. The tokens can be transferred to the family member in case of the user's death. SafeHaven uses a secret sharing scheme to distribute cryptocurrency secret keys among nominees. The original key can be reproduced when each nominee submits their share, including the share of a trusted party known as Trusted Alliance Network (TAN). TrustVerse is an estate planning and asset management system that allows users and their family members to create a smart contract using blockchain for an asset. The smart contract will be executed when a threshold number of family members will confirm the death. However, all these have a poor technical description and have many existing challenges, such as being applicable only for type 3 category assets, the possibility of recreation of the incorrect key by a nominee, or a weak method of death confirmation.

In this proposed work, we have designed a DAI protocol. Any user can manage all four categories of assets he wants to transfer, and the nominee can inherit them successfully after the user's demise.

2.4 Data breach and analysis

Both GDPR [74] DPDPB-2022 [6] define a data breach as one of the prominent obligations. The earlier data breach focused on impacting the institute's reputation. Still, it later shifted to more monetary purposes [162, 100]. Data breach analysis and appropriate response are always challenging for an organization [15, 10]. Statistics of data breaches over the time are given in [86].

The data breach is studied from various aspects such as: [119, 127] discuss why we should care about the targeted data breach and prevent businesses from breaches. The [152] explain consequence of a breach, the role of data breach disclosure[36], preventing data breach in higher education [184], data breach prevention of Covid-19 data [59], the role of top management in data breach security [155]. Few studies are done from the aspect of data breach risk prediction [68] and prediction of the possibility of breach [38].

The technological implementation is necessary for robust compliance of obligations of the bill [157]. Enabling logs is a critical approach for compliance and incident response [7].

The [111] provides a systematic review of existing SIEM models and states the futuristic scope. A customized SIEM can be designed as per the needs of the system. A visualization aspect of a data breach is described in [110].

The data breach is less studied from an incident response perspective. One study is done on data breach identification and notification of data breach incidents [183], and another is related to improving the incident response process in health care [90].

2.5 Logs and threat identification

Incident response is a well-known process in cyber security. Several activities are defined in incident handling, both proactive and reactive [52].

The [43] emphasizes the necessity of a structured incident response method and a maturity model to deal with incidents. The model [172] discusses the lack of security awareness in handling incidents and studies the relationship between security, system, situational awareness, and user's ability to detect, evaluate, and respond to threats. Both works are primarily emphasized on a structured network. However, as we move into a new age of the smart world, we must also apply suitable incident-handling methods for end users.

The necessity becomes more relevant when we study multiple APTs targeting individuals of different sectors. They target collecting credentials, dropping malware, or hosting malicious apps for fraudulent purposes [13, 28, 158]. They run various campaigns to achieve more victims [35]. The phishing exploits are reaching more citizens after the Pandemic [42]. The major concern is these URLs exist for a short duration and become obsolete. Therefore, victims should be alerted, and they should sanitize themselves. Many threat-sharing mechanism provides threat feeds [178]. Still, they have several limitations, such as feeds being not free, not correctly synchronized and designed mainly for structured networks, and having no well-defined mechanism to integrate with end users. Thus, synchronizing threats accessible to the individual and their impact and mitigation steps may be more helpful in threat detection.

Similarly, evaluating the historical data may be vital in early incident mitigation. The latest guidelines emphasize the same and advise monitoring the logs for better incident response [7]. Most log analyses focused on structured networks [164].

The browser's history is already being analyzed from various perspectives. Different tools, techniques, and methods are already available over the internet to collect the browser evidence and its analysis [118, 128, 147, 32].

The explored ESUL model is moderately different as it continuously collects the history matches with an updated list of malicious URLs and emphasizes individual systems.

Chapter 3

Background and Technological Compliance Necessity

PDPB-2019 was introduced in Lok Sabha in 2019 by modifying the draft bill PDPB-2018 and sent to the Joint Parliament Committee for review and further discussion. The bill raised several concerns, including the following significant considerations: i) localization and cross-border transfer of data and ii) the narrow boundary between personal data, sensitive personal data, and critical personal data (because defining these are very difficult as the nature and category of data are growing in the era of data science, machine learning, and AI). iii) the applicability of the definitions of consent, explicit consent, and deemed consent iv) data breach reporting.

The DPDPA-2023 eliminates many ambiguous clauses by tweaking restrictions. Few of the conditions are provisioned as ”rules as prescribed” by the compliance bodies. We highlight a few of the fundamental changes adopted in the current version of the act.

3.1 Changes adopted in DPDPA-2023 from PDPB-2019

The DPDPA-2023 dilutes multiple criteria that were available in previous drafts. The significant modifications include removing the terms sensitive and critical personal data and localizing data clauses. For the transfer of data outside India, the regulation may specify a list of countries or territories where the transfer of personal data will be prohibited. The bill also removes the concept of trust score, rating, etc. The PDPB 2019 contained terms such as social media significant data fiduciary, guardian data fiduciary, etc. All these definitions are removed in DPDPA. Similarly, a provision of penalties is kept if DP deviates from duties. Some processing criteria, such as collection and purpose limitations, have not been specified directly. However, it may be indirectly associated with other constituents. Table 3.1 highlights a few of the modifications adopted in the Act.

S.No	**Changes adopted**	**Details**
1.	Data categorization	Removal of categories such as sensitive and critical personal data. Followed a single definition as “personal data”.
2.	Data Transfer	Will be prohibited in some countries or territories as prescribed.
3.	Consent notices	Scopes are relaxed e.g. does not include list of third parties and storage limitation clauses etc.
4.	Processing criteria	restrictions relaxed e.g. definitions such as storage limitation, purpose limitation, collection limitation are omitted and processing are covered with new definitions such as “law full processing” and “legitimate uses”.
5.	Publicly available data	Excluded from applicability of the act e.g. social media posts.
6.	Duties of data principal	Duties of data principals are included such as not to impersonate, not to suppress any material information, not to register false or frivolous grievances, fine up to 10k Rupees
7.	Significant data fiduciary	Removed definitions such as social media significant data fiduciary, guardian data fiduciary etc. and used only the modified definition of “significant data fiduciary”.
8.	Data breach reporting	The definition is modified, now the breach shall be reported to the board and each affected data principal
9.	Rating	During audit, the definitions of rating and trust score of a data fiduciary are removed
10.	Penalty	The provision of penalty is modified. For instance, penalty is increased in case of data breaches, and penalty is also kept against violation of duties of data principal.

Table 3.1: Key changes adopted in DPDPA-2023 from PDPB-2019

3.2 Differences between GDPR and DPDPA-2023

GDPR is a widely recognized framework around the globe. PDPB is aligned with GDPR, along with a few key differences in exercising various obligations. For instance, DPDPA defines the concept of *consent manager*, an individual nominated by the organization as a point of contact to facilitate, comply, and lodge grievances related to consent processing. Similarly, DPDPA sets the role of the data protection officer in the Indian context. GDPR uses slightly different nomenclature of data fiduciary and data principals such as data controller and data subjects. A few differences exist under the implementation of various rights, the definition of significant data fiduciary, and obligations of data breach reporting. Table 3.1 highlights some differences between GDPR and DPDPA.

3.3 Technological Compliance Necessity

The prime expectation from the bill is stronger compliance. This can be achieved only if the service providers implement technological methods to validate compliance. This thing we urge with two scenarios discussed below using obligations: i) **Consent** and ii) **Data collection**.

We extract the possible deviations if a service provider does not comply with obligations well. We describe the technological methods that, if implemented correctly, may minimize the deviation and provide stronger technological compliance.

The description for a few other major obligations is also studied and mentioned in detail [157].

3.3.1 Consent notices

Existing method of consent retrieval

The figure 3-1 shows the current model of establishment of consent. DF provides a form F to the data principal along with either of the two options. In the first one, he has to accept or reject the consent. In such cases, all the terms and conditions are enclosed in the consent form. In the second case, the data fiduciary provides opt-in methods that allow users to select a few choices and preferences. After selecting appropriate preferences, the data principal can return the document X. Few organizations allow modifying the consent. To do this, the data principal can request to modify his choices. He can re-submit it as document X' after making the necessary changes.

Detail	**GDPR-EU**	**DPDPA-2023**
Entity	Data subject, Data controller	Data principal, Data fiduciary
Cross border transfer	May be transferred without any restriction, if commission/authority assure appropriate safeguard and protection	Prohibited in some country or territory as per description
Data breaches	Data controller may inform to data subject, in case of high risk	Data breach shall be reported to the board and each affect data principal
Rights	rights to data portability and rights to object	Not specified
Nominee	Not specified	Nominate someone to exercises data principal's rights
Data minimization	Specified	Not specified
Children consent	Approval of parental authority is required to process children personal data for children below 16 years	Children data shall be processed after verifiable consent from parent/guardian
Children data processing	Not specified	Data fiduciary shall be barred from tracking, behavioural monitoring and target based advertisement.
Right to access	Data subject may obtain copy of personal data being processed	Data principal May obtain summary of personal data, identity of fiduciary and list of third parties with whom data shared along with any other related information
Consent Manager	Not specified	Record and manage consent of data principal
Significant data fiduciary	Not specified	shall be defined based on no of users, volume of data processed or using some other criteria as specified

Table 3.2: Comparison of GDPR-EU and DPDPA-2023

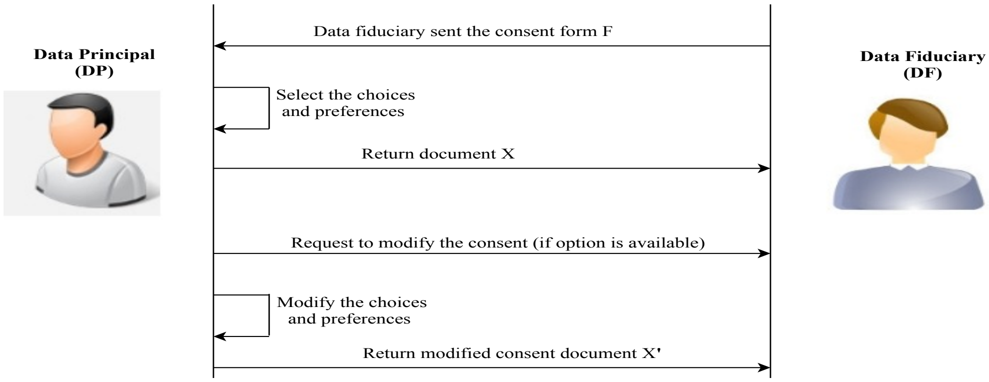

Data Principal (DP)
Data Fiduciary (DF)
Data fiduciary sent the consent form F
Select the choices and preferences
Return document X
Request to modify the consent (if option is available)
Modify the choices and preferences
Return modified consent document X'

Figure 3-1: Current model of consent establishment between data principal and data fiduciary

Limitations of existing consent
The current consent retrieval does not fulfill many of the consent properties. For instance, both the accept/reject method and the opt-in method of collection of preferences do not guarantee correctness because:
(1) Since consent is managed solely by the data fiduciary, he can modify it without the user's approval, or he can prove that the consent is obtained from individuals even without collecting it.
(2) There is no way of verifying whether the processing is done according to the terms and conditions provided in the consent.
(3) It cannot prove whether the processing was done based on recent consent or if the consent was modified.

3.3.2 Security and privacy requirement of consent

The bill presents consent from a regulation perspective. This means that parties are questionable in court if they deviate from the terms and conditions in the consent form. Regulatory action may be taken after an event and requires shreds of evidence.

This evidence can be (possibly) encoded within the software to ensure that the consent properties are implemented accurately and transparently. We call such encoded evidence as ***security and privacy requirement*** that must technically implement the consent. This will prevent data fiduciary from deviating from the legal contract (as in the consent form). It will help resolve disputes if they arise, or it can be presented to the jurisdiction of the data protection authority for any regulatory compliance.

At a very high level, the following are extracted properties considering technological security and privacy requirements in designing consent:

1. Implement methods to prove that free, informed, specific, and clear consent is implemented.

2. Implementation of methods to prove consent is established with the data principal. If the data principal denies consent establishment, the fiduciary can legally prove that consent was established.

3. Implementation of methods to maintain the records of the list of third parties with whom data is shared.

4. Implement methods to maintain the records that personal data has been shared with stated third parties only (this property requires a transparent audit).

5. Implementation of methods to prove withdrawal of a consent. He should maintain proof that the processing has stopped after consent withdrawal. Such proofs are necessary because data fiduciary may keep this data longer for monetary purposes.

6. Implementation of methods to include with whom, when, how, and why the consent was established with data principal.

7. Implement methods to allow the data principal to change his choices or preferences. If the data principal approves such changes, further data processing must be done as per the new consent. Data fiduciary should implement methods to prove that modified consent is recorded and processing is done on modified consent.

8. Data fiduciary may update consent notices from time to time (e.g., due to a change in the data fiduciary's policy or DPDPA requirements). The users will get an option to update their consent preferences. The data fiduciary should implement methods to keep proof of modifications.

9. Implementation of methods to prove that he is not automatically collecting personal data without the user's consent.

10. Implement methods to validate data. The fiduciary is not using data for other purposes.

11. Implementation of methods to validate third parties are processing data as per the contract with the data fiduciary.

12. The third party has to keep the source of data if it is not collected directly from the data principal. The implementation of methods to validate the origin of personal data. Such proof is necessary because no one knows how data is processed behind the wall [156].

13. If the data principal is withdrawing consent, the same will also be communicated to third parties. Implementing methods to validate third parties also stopped processing after consent was withdrawn.

3.3.2.1 Technological remedies

We discuss a high-level overview of how technical methods can be used to implement this.

To satisfy item 2, we must encode **undeniable consent property** during consent collection. One way to achieve it is with the use of *digital signature [105]* schemes, which will work as a non-repudiation technique for consent. The signature on the document will ensure two things: 1) DF can prove that he has obtained consent before data processing, and DP can not deny it; 2) No party other than the data fiduciary can have such proof (if implemented correctly with a combination of other cryptographic techniques). To modify the consent (item 7) or withdrawal (point 5) of the consent, the data fiduciary can ask the user to re-submit another form with different signatures.

Items (5,7, and 8) describe the implementation of methods aligning modification or withdrawal of the consent. Modification in the consent can be initiated either by the data principal or data fiduciary. It can also be initiated when changes happen in the organization's privacy policy or the policy of DPDPA standards. The users can submit a modified consent form along with the new signature. Now, the DF has old and recent copies of the consent (**the dilemma of old and new consent**). Both consents are valid. This may allow DF to be malicious. For instance, he can delete either copy of the consent and do processing based on the remaining one. In such a scenario, DP can not prove the malicious behavior of the data fiduciary. Therefore, *keeping both old and new consent does not force DF to process the personal data based on recent consent*.

To distinguish between old and new consent, blockchain-based [4] methods can be used. Blockchain is a tamper-proof, immutable, verifiable ledger used to record transactions. Transactions recorded on the blockchain are transparent. Proof of consent can be recorded in any public blockchain, for example, [67]. Since all the consent will be available on the public chain, both parties can agree on the latest consent. This will also ensure parties resolve the dispute (if any) through the regulatory authority.

There are few consent properties where implementation may be difficult using existing techniques. For instance, the data fiduciary is collecting data without consent (item 9), the data fiduciary is sharing data with unauthorized parties (violating consent conditions) (item 11), or the data fiduciary is processing data even after consent has been withdrawn (Item 5).

The literature has not thoroughly discussed technical solutions to complying with consent.

Though many authors claim [186, 185, 154] that Blockchain is a panacea for this, the authors are skeptical. The primary reason is the difficulty of proving what operations are performed on data. Maintaining an audit log is insufficient, as it cannot keep track of activities performed by unauthorized users. Even a data fiduciary might choose not to log some of the events, mainly when it shares data with unauthorized third parties or performs malicious operations.

The more advanced technical method could be explored to solve the existing challenges of proof of consent. For instance, a more advanced cryptography-based method could be developed to verify the consent, such as *third party audit [180]*. The audit scheme will increase transparency and validation.

3.3.3 Data collection

Data can be collected after retrieving the consent. In the PDPB-2019, many sections cover data collection properties, such as Articles 5, 6, and 9, which correspond to the *"purpose, collection, and storage limitation"* respectively.

DPDPA does not directly specify the obligation *collection limitation* and *storage limitation*. However, it is specified indirectly under Section 6. For instance, the **collection limitation** may be derived from the obligation specified in Section 6. It states: *" The consent given by data principal ... processing for the specified purpose and be limited to such personal data as is necessary"*, thus data collection and purpose limitation.

We study the data collection specifications with the following properties:

1. Purpose limitation.
2. Collection limitation (by following specified purposes and restricting the category of personal data being collected).
3. Storage limitation (duration of personal data storage).
4. Sharing with third parties.
5. Security and safeguard (during collection, share, and storage).

3.3.3.1 Collecting data for specified purposes

The primary accountability from the data fiduciary is to prove that collected data is only for specified purposes, thereby minimum. It is necessary because data fiduciary can have a strong monetary incentive by collecting enormous amounts of personal information. Sometimes, sensitive data like health information, credit card numbers, or medical records [129] are collected even without the user's knowledge. However, it is still a technical challenge, *how to justify whether collected data is only for specified purposes?*. Here, We discuss one approach to improve the data collection process to achieve the goal of a stronger specified purpose.

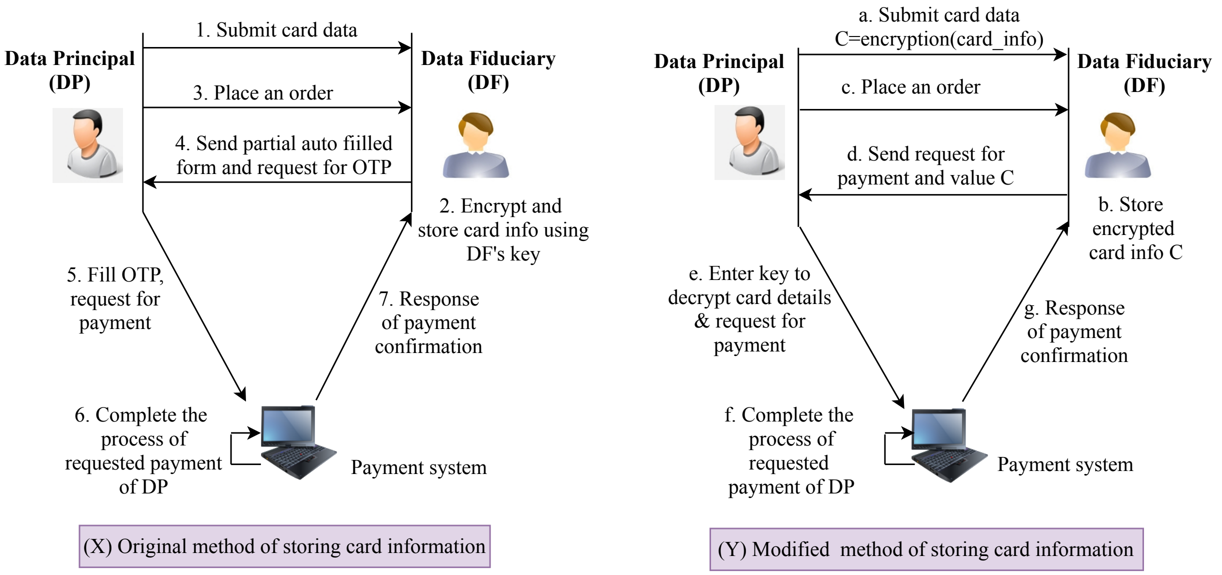
Enrichment in data processing methodology: Data fiduciary should change their existing data processing methodology. A few technical changes in the processing will not only fulfill the goal of data collection but also serve the same business purpose. We have justified our argument using two use cases. Both use cases indicate different observations. The first discusses the requirement of technology change in processing, while the second concerns the necessity of collaboration among data processing parties. Currently, data fiduciary collect multiple sensitive personal information in both cases. We have explained that neither it is providing minimum data collection, nor it is providing data protection. Later, we showed that the same objective, “minimum data collection and necessary data protection,” can be achieved by just a few modifications in their processing activities.

3.3.3.2 Use Case 1: storage of debit card/credit card information

Debit or credit card data is sensitive personal information and requires extra protection during processing. The commercial platform provides *ease of doing* facility wherein quick payment could be made by storing the user's card details [71] at the data fiduciary portal. The purpose is as follows: *“It's quicker. By saving your card details, you can save the hassle of typing the complete card information every time you shop at Flipkart. Your card information is 100 percent safe with us. We use world-class encryption technology while saving your card information on our highly secure systems”* [71]. Figure 3-2(X) shows the current model in which data fiduciary stores card information to facilitate quick transactions. The data fiduciary's key generally encrypts the data; hence, it is always available to him. Whenever a user places an order, the merchant sends a payment request. The payment request contains a partially auto-filled form with card information already filled out. Thus, users do not need to submit card details manually. In the next step, the user enters a one-time password (OTP) and offers it to the payment system. The advantage of such processing is that the user would get the ease of doing and quick processing by not entering card information manually.

After the bill's introduction, it is questionable whether such collection and storage are required. If yes, then in which form should the fiduciary store it? Because data fiduciary is non-trustworthy, weak protection of data may reveal card details publicly, or card information can be used for malicious purposes. Data breaches here may lead to severe harm, sometimes appearing in the news when the user's card information becomes available over the dark net. Hence, such storage increases doubts about user data security and privacy. Could the same purpose *quick transaction and ease of doing* be achieved by preserving privacy, protecting card details, ensuring limited data collection, and data minimization? To achieve this, each party involved in the processing should only get the necessary information. The data fiduciary does not need to know the details of the card. It should be visible only to the bank. The specification of the secure electronic transaction (SET) also states that the "order information(OI) information shall be processed by merchant and payment information(PI) by the bank" [54]. Both sections should be processed separately.

Use of more advanced methods: Cryptographic techniques can help to achieve the above goals. For instance, figure 3-2(Y) ensures privacy by preserving minimum data collection and also provides *ease of doing*. Data fiduciary collects card details as a ciphertext $C = Encryption(card\_info)$ encrypted using the customer's key. The key may be anything such as the user's password; since data fiduciary does not know the user's key so it is useless at the merchant's end and would be available at the bank's end instantly. When a user places an order, the data fiduciary can send the payment request along with order details and the ciphertext C. Users can decrypt card information and forward payment requests to the payment system. Since decryption is being done on the user's side, the data fiduciary would not be able to know the card details. It will ensure the goal of data minimization, purpose limitation, collection limitation, storage limitation, and information sharing.


1. Submit card data
Data Principal (DP)
Data Fiduciary (DF)
3. Place an order
4. Send partial auto fiilled form and request for OTP
2. Encrypt and store card info using DF's key
5. Fill OTP, request for payment
7. Response of payment confirmation
6. Complete the process of requested payment of DP
Payment system
a. Submit card data C=encryption(card_info)
Data Principal (DP)
Data Fiduciary (DF)
c. Place an order
d. Send request for payment and value C
b. Store encrypted card info C
e. Enter key to decrypt card details & request for payment
g. Response of payment confirmation
f. Complete the process of requested payment of DP
Payment system

(X) Original method of storing card information

(Y) Modified method of storing card information

Figure 3-2: An example of data collection: original vs modified method

Data Protection Bill encourages data fiduciary to implement such kind of technology. It would enforce data visibility at the right place and with appropriate security and safeguards. Using more cryptographic methods will enhance the confidence of the data principal that he has control over their data and is safe. Such techniques would also provide data processing, storage, and sharing transparency.

3.3.3.3 Use case 2: Prevention of disclosure of sensitive personal information - a PAN card example

What if one data fiduciary enhances their processing methodology while others do not? Sometimes, it requires the participation and collaboration of all parties involved in processing to upgrade their techniques. Through the use case of PAN cards, we have shown that until and unless all the parties with whom data is being shared do not enhance their data processing methods, the goal of data protection can not be accomplished.

PAN card is a unique number assigned to all taxpayers within India. The income tax department keeps track of the individual's tax declaration and income; it can also be used for identity proof. If anyone purchases or avails service of more than the specified amount, he has to disclose the PAN card details to the merchant. The authority can use these details to inspect an individual's tax declaration. It has been observed that the details of PAN cards collected by merchants can be used for malicious purposes such as to purchase benami properties using identity theft [173], to perform fraud payment [116], or to hide income taxes. Consider one example of such PAN detail submission at the commercial websites as shown in figure 3-3. Merchants collect PAN details when users purchase more than the specified limit. Merchants forward information on purchased details to the IT department. The IT authority may verify whether purchasing details violate the regulation of tax disclosure. In the case of fraud/theft, an investigation may be started against suspicious users.

The existing model does not solve the intended goal of data collection: transparency in expenditure and declaration of taxes. In this case, multiple breaches are possible, such as tracking user behavior by the merchant or using card details for impersonation and identity theft. From an authority point of view, details shared by merchants are not trustworthy. Merchants can share inaccurate data, or they can share false information submitted by the users. The authority expects more monitoring and transparency in the purchase activity. However, the current model of PAN details submission neither solves the purpose of any party nor fulfills the collection goals. Furthermore, even if one party, such as a merchant, enhances its methodology to fulfill the data collection goal, the purpose will still not be solved until and unless other parties collaborate.

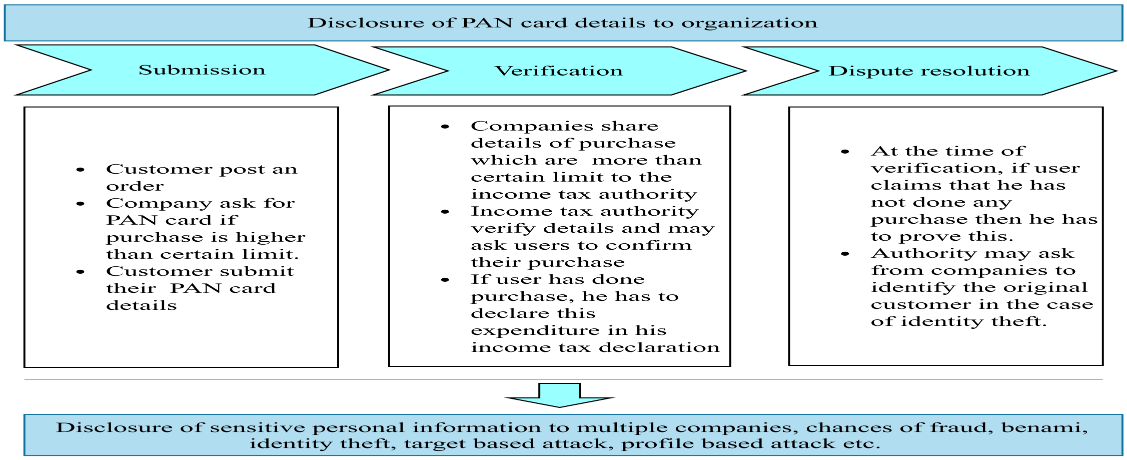

Disclosure of PAN card details to organization
Submission
Verification
Dispute resolution
Customer post an order
Company ask for PAN card if purchase is higher than certain limit.
Customer submit their PAN card details
Companies share details of purchase which are more than certain limit to the income tax authority
Income tax authority verify details and may ask users to confirm their purchase
If user has done purchase, he has to declare this expenditure in his income tax declaration
At the time of verification, if user claims that he has not done any purchase then he has to prove this.
Authority may ask from companies to identify the original customer in the case of identity theft.
Disclosure of sensitive personal information to multiple companies, chances of fraud, benami, identity theft, target based attack, profile based attack etc.

Figure 3-3: Existing model of sharing of PAN information

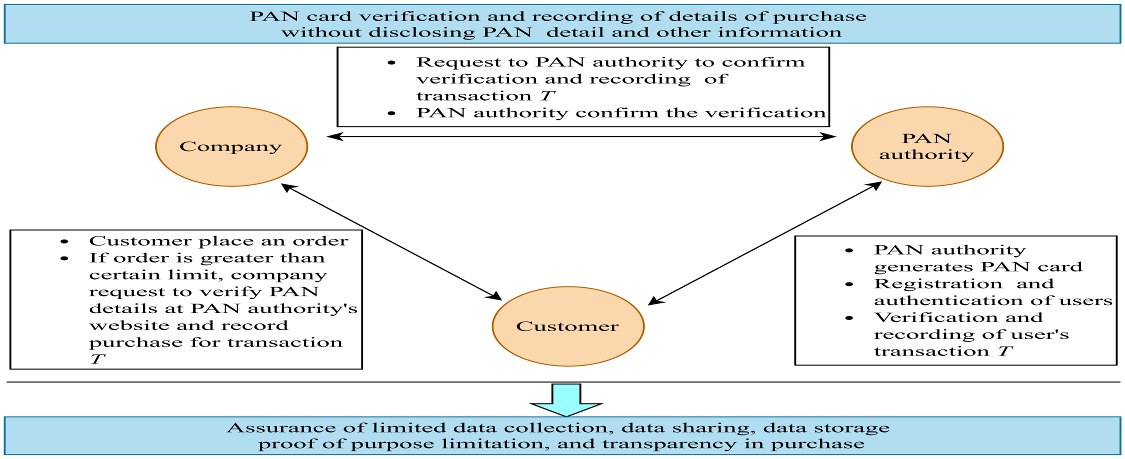

PAN card verification and recording of details of purchase without disclosing PAN detail and other information
Request to PAN authority to confirm verification and recording of transaction T
PAN authority confirm the verification
Company
PAN authority
Customer
Customer place an order
If order is greater than certain limit, company request to verify PAN details at PAN authority's website and record purchase for transaction T
PAN authority generates PAN card
Registration and authentication of users
Verification and recording of user's transaction T
Assurance of limited data collection, data sharing, data storage proof of purpose limitation, and transparency in purchase

Figure 3-4: Modified model of sharing PAN information

Collective change in processing methodology Figure 3-4 shows that if all the parties collaborate and modify their data processing approach, they will solve the above issues. Instead of collecting PAN details directly, data fiduciary (merchants) ask users to log their transactions (purchasing activities) T at the IT portal if it is higher than the specified amount. As shown in the figure, both the user and the merchant can confirm logging of purchase details using appropriate authentication and verification. A comparison is shown in Table 3.3. In the new mode, only the relevant details are getting to every party. The merchant is not collecting any PAN details, hence achieving limited data collection and storage. The merchant does not need to share anything (limited data sharing). The authority logs all the required purchases using proper authentication; hence IT department can achieve transparency and monitoring in the purchase. Moreover, such transactions will eliminate the possibility of impersonation, identity theft, and tax fraud.

Property	**Present method of PAN details processing**	**Modified method of PAN details processing**
Limited data collection	✗	✓
Limited data storage	✗	✓
Limited data sharing	✗	✓
Authentication of user’s	✗	✓
Transparency in processing	✗	✓
Privacy of personal data	✗	✓

Table 3.3: Comparison of existing vs modified model of PAN details processing

3.3.4 Technological compliance

The above study concludes that using enhanced techniques, modification in existing data processing methods, and participation of entities can provide data management models aligning DPDPA and may enable stronger compliance.

In Chapter 4, 5, 6, 7, we formulate the problem statement keeping a view of the above and study the respective one.

Chapter 4

Transparent consent data processing

The existing techniques of consent processing are not transparent and do not adhere to data protection goals. Data fiduciaries may misuse the collected data for purposes other than specified in the consent. Therefore, a robust model is necessary for the framework's consent processing objective.

We have described that encoding requisite security and privacy properties will ascertain stronger consent compliance. We formalize these properties as *Proofs of Consent (PoC)* and categorize them into three layers. Acquiring a higher layer will minimize adversarial risks and ascertain greater transparency. Based on this, we have proposed a model *shielded consent manager (SCM)* using blockchain state channel and other cryptographic primitives to retrieve consent to grant permissions to access Android resources. SCM includes parameters as per the framework, satisfies the security properties such as integrity of consent, non-deniability by users, auditability of logs in data processing, and provides finer visualization of user's consents. The simulation of the contract is done using solidity, truffle, and ganache test network, and the feasibility of practical implementation is analyzed to show the efficacy of the model.

Recalling the background specified in Chapter 3 out of multiple tenets specified in the new regulation, data consent is an empowering and indispensable clause that enables service providers as one of the bases for data processing. The regulation enumerates the service provider (aka Data Fiduciary, DF) has to obtain the user's (Data Principal, DP) consent (permission) before collecting and processing personal data. Consent is a set of choices, preferences, or agreements given by individuals to another. It determines when, where, why, what, and how much information shall be shared with them.

After the framework, the service provider is expected to retrieve the user's consent and perform only the specified purposes mentioned in the consent form. However, the above is not always true. The most trivial problem is that many organizations do not mention their consent policies and collect user data from a small amount to a large extent, even without the user's consent. This collected data may be used for malicious purposes. Further, The current consent retrieval and management model uses *policy form* that lacks standardization, such as DFs usually do not provide opt-in methods (allowing users to select choices and preferences), and those who provide use their methods to implement it. Next, DF may create a fake consent, modify, delete, or update it even without the user's approval.

A few ontology-based consent management models have been proposed for making consent more standardized [33, 136]. They use XML-based ontology to express, validate, and share consent in a heterogeneous environment. However, these are also based on the assumption that data processing parties are honest, which may not be true.

Blockchain technology has also been described as a data management solution [174, 181]. The associated parties can integrate necessary logs during data sharing. Such logging may enable transparency in the system. However, as blockchain is at a very early stage, we need to identify the scope of how blockchain can help in consent processing.

Thus, the existing consent management models are at a very initial stage and do not guarantee data protection. It requires suitable formalization and standardization. To achieve this, we require two components: i) integration of the legal basis of the consent mentioned in the regulation, and ii) a set of correctness proofs that ensure consent properties are complied with, given that a party involved in the processing may be dishonest. The regulation will include legal binding and penalties, while the proof of correctness will ascertain protection. In this work [1], we focus on the second one. Our major contributions are as follows:

- A detailed description of the existing consent retrieval mechanism and its limitations is provided.
- We have analyzed the DPDPB-2022 and formulate requisite parameters, defined as *regulation properties (RP)* to be integrated in consent. Then, we describe the list of security properties against RPs that must be encoded to provide the framework's complied consent processing. These security properties are defined as Proofs of Consent(PoC) and are described as a three-layer architecture.
- We have proposed a model *Shielded Consent Manager (SCM)*. SCM integrates blockchain state channels and other cryptographic primitives to design a stronger transparent consent model to grant permissions to access system resources by an Android application. The model achieves security properties such as integrity of established consent, non-deniability of users, and correct auditable logging of collected data.
- The simulation of SCM is implemented, and the feasibility of practical implementation is analyzed to show the model's efficacy.

The rest of the sections in the chapter are organized as follows. Section 4.1 briefly describes the basis of consent, existing limitations of consent management, and possible deviations, Sect.

[1] The opinions expressed in this work are those of the authors.

4.2 discusses technological formalization, Sect. 4.3 enumerate proofs of consent, and Sect. 4.4 describes the SCM model and implementation.

4.1 Problem formulation

The DPDPB-2023 obligations state the nature, scope, and specify consent as one of the bases for data processing and should be obtained before collecting and processing personal data.

According to Article 5(1) of DPDPA, *"data may be processed based on consent"*. Further, Article 6(1) states: *"consent shall be free, informed, specific, unambiguous .."*. Data principal also has the *right to modify or withdraw their consent* (as per Article 6(7)). Next, as per Article 6 (3), *"consent shall be presented to the DP in a clear and plain language along with the contact details of data protection officer.."*.

The *processing of children's data* will be in such a manner that protects children's rights (as per Article 9). DF has to obtain and verify *parental consent* from parents or guardians before processing it. DFs are prohibited from profiling, target-based advertising, or any other activities that may harm children (as per Article 9(3)).

DF will be responsible for keeping proof that the DP has obtained consent before processing. As per Article 7(9), *"DF shall be obliged to prove that a notice was given by the DF to the DP and consent was given by the DP to the DF.."*. Data will be shared with the data processor and third parties with a valid contract (as per Article 9(9)). Finally, the DF has to implement an entity named *consent manager* through which DP can record, manage, and review the consent (as per Article 7(6)).

4.1.1 Existing consent retrieval methods and its limitations

Currently, the following methods exist in the state of the art for consent retrieval and processing.

1: Privacy Policy Form

The DF maintains a privacy policy statement over the portal. DF asks users to fulfill the provided proforma, which serves as consent (a kind of agreement established between both entities for collecting personal data) while accessing some services. We can consider this submission offering the data principal a document/form F. DF expects DP to read/understand the conditions honored in the form and provide their undertaking. Such consent collection practices widely use the following two formats: a) *accept/reject method* In this, all terms and conditions are enclosed in the consent form, and the user has to accept or reject it. b) *choices and preferences methods* it provides users opt-in methods to select a few choices and preferences. The selection of choices depends on how DF has declared its privacy policy.

The policy form-based consent collection has multiple limitations:
i) It does not include all the directions specified in the framework. ii) Opt-in methods are written as a legal policy document, use narrative languages, and do not define many subjects explicitly, such as the list of third parties with whom data will be shared.
ii) It does not guarantee correctness because DF manages consent solely. In case of malicious DF, he can modify the document without the user's awareness.
iii) There is no way of verifying whether the processing is done according to the mentioned terms and conditions. iv) It can not be proved whether the processing is done based on recent consent if the consent was modified.

2: ONTOLOGY BASED

The new approach proposed in state-of-the-art [136] is the ontology-based consent representation. This model converts consent conditions into variables that can be described as objects, for instance, the name of DF, date of consent obtained, duration, etc. The data is mapped as objects and variables of ontology-based language which enables the processing of consent data in machine-readable format.

It has the following advantages:
i) Collecting user's choices and preferences as a variable format will enable finer visualization and data processing. The DF can utilize such methods to map legal conditions as consent management ontology data management [136].
ii) Such representation may be shared consent permission with distinct parties in a heterogeneous environment [89, 33].

The model has a finer representation. However, this also has many limitations similar to the policy form model:
i) The model assumes that DF is honest.
ii) Modification of the consent form may be done.
iii) Data may be shared with other parties beyond those declared in the consent form.

3: BLOCKCHAIN BASED

Few states of the art have proposed using blockchain to log activities associated with data processing [174] to ensure data protection compliance.

The inclusion of blockchain may provide finer control. But, since both blockchain and data

protection are at a very initial stage, we require more formalization as:
i) How to store large amounts of logs as blockchain is not scalable and storing data is costly.
ii) It needs to refine which information may be logged over the blockchain, given that user data privacy is honored. iii) It needs to identify the possible deviations even after using blockchain. For instance, DF may use data for other purposes not mentioned in the consent form without logging information over the blockchain.

4.1.2 Definition of malicious parties

Without awareness, DF may deviate and process data behind the user's back for other purposes. Therefore, it is necessary to analyze the possible adversarial deviation of parties. This definition will help to design precise consent processing complying with data protection. At a very high level, these are as follows:

- *Data fiduciary:* Consent is solely managed by DF. He can modify, add fake consent, or delete existing users' consent. Apart from this, DF can do the following:
  (a) DF may collect additional data not mentioned in the consent agreement.
  (b) DF may violate the terms and conditions enabled in the consent, such as sharing the data with additional third parties.
  (c) DF can deny the collected user's consent.
  (d) DF may deny holding the personal data collected after consent (This is usually seen in data breach cases).

- *User* The user can deny that consent is given to DF.

- *Auditor* An auditor may perform an erroneous audit during consent or data processing audit.

4.1.3 The ideal case of consent processing

The description of Sect. 4.1.1 and Sect. 4.1.2 urge correctness and transparency in consent processing. We define the following ideal instances expected in consent management complying with data protection in the presence of adversarial entities.

Goal 1: Entities should precisely understand and interpret the legal requirements of consent, derive technical properties, and include all in consent processing.
Goal 2: Entities should include appropriate security and privacy properties and ensure the correct

implementation of these requirements using different techniques.

4.2 Technological formalization of consent processing

We propose a three-layer architecture for consent processing as shown in Fig 4-1. These layers will satisfy the ideal expectations (defined in Sec. 4.1.3) of consent in adversarial settings. Layer 1 describes the properties necessary to include in consent's representation. Layer 2 describes the security properties that need to be encoded to prove the correctness of consent processing. Layer 3 emphasizes on to bind consent and data together. We explain each category below:

4.2.1 Layer 1: Inclusion and representation

INCLUSION We have extracted a set of competency questions based on the context of the consent defined in the regulation. These are denoted as regulation properties (RP), which must be fulfilled correctly to show consent compliance. We have mentioned twenty-four competency questions RP1-RP24, as shown in Fig. 4-2, categorized into seven groups. This classification is done based on the context of the question or type of entity or data associated with the consent. For instance, properties RP1- RP9 represents basic information about the consent, while properties RP10-RP12 represent the time and location information about the consent.

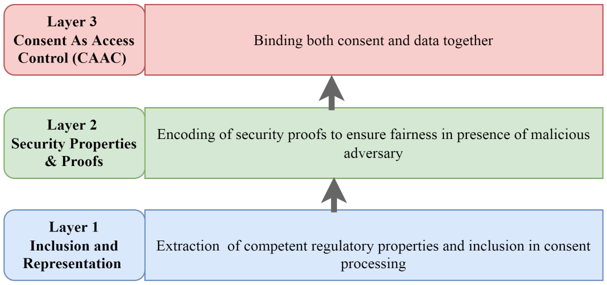

Layer 3
Consent As Access Control (CAAC)
Binding both consent and data together
Layer 2
Security Properties & Proofs
Encoding of security proofs to ensure fairness in presence of malicious adversary
Layer 1
Inclusion and Representation
Extraction of competent regulatory properties and inclusion in consent processing

Figure 4-1: Three-layer requirement of consent formalization to adhere to the goal of data protection

REPRESENTATION The representation of properties helps users understand and interpret the consent format easily. The better representation helps classify the consent information conveniently at a broader level. The following methods may help to provide the consent form in a more structural format:
i) A stronger graphical user interface (GUI).

ii) More options for checkboxes.
iii) Consent in the user's native languages.
iv) A short description and hyperlinks for details.
v) Clarity in the sentences.

Table 4.1: Consent representation

Consent Description
Variables: Time: 01/01/2023, ExpiresOn: 01/04/2023 Location: 192.168.1.100 Where: website Third parties: party A, Party B, ..
Policies When you use our Website, we collect and store your personal information which is provided by you from time to time .....
Security Proofs: Smart contract Address: ”12345” Hashes: ”000000” Signature: ”111111” Nonce: ”abc, pqr”

We segregate the variables, policy, and security parts to represent different segments as shown in Table 4.1. This will help users to understand the consent agreements more clearly. This paper emphasises on layer 2. Hence, the detailed description of layer 1 was omitted.

4.2.2 Layer 2: Inclusion of security properties

To minimize deviation, we must integrate requisite security properties along with Layer 1. These legal evidence encoded with RPs (Fig. 4-2). The table 4.2 enumerates adversarial deviation against RPs, expected security properties, and the list of existing technological methods that may satisfy the requirements. We define each property below:

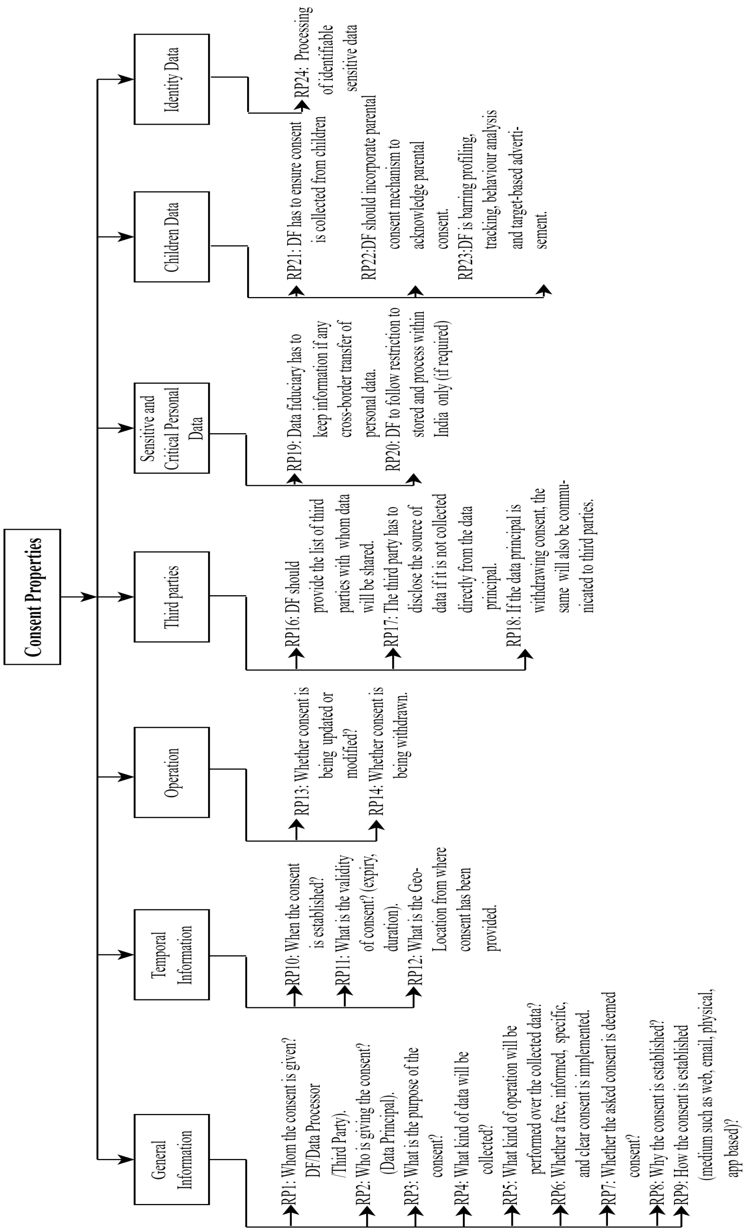

Consent Properties
General Information
Temporal Information
Operation
Third parties
Sensitive and Critical Personal Data
Children Data
Identity Data
RP1: Whom the consent is given? DF/Data Processor /Third Party).
RP2: Who is giving the consent? (Data Principal).
RP3: What is the purpose of the consent?
RP4: What kind of data will be collected?
RP5: What kind of operation will be performed over the collected data?
RP6: Whether a free, informed, specific, and clear consent is implemented.
RP7: Whether the asked consent is deemed consent?
RP8: Why the consent is established?
RP9: How the consent is established (medium such as web, email, physical, app based)?
RP10: When the consent is established?
RP11: What is the validity of consent? (expiry, duration).
RP12: What is the Geo-Location from where consent has been provided.
RP13: Whether consent is being updated or modified?
RP14: Whether consent is being withdrawn.
RP16: DF should provide the list of third parties with whom data will be shared.
RP17: The third party has to disclose the source of data if it is not collected directly from the data principal.
RP18: If the data principal is withdrawing consent, the same will also be communicated to third parties.
RP19: Data fiduciary has to keep information if any cross-border transfer of personal data.
RP20: DF to follow restriction to stored and process within India only (if required)
RP21: DF has to ensure consent is collected from children
RP22:DF should incorporate parental consent mechanism to acknowledge parental consent.
RP23:DF is barring profiling, tracking, behaviour analysis and target-based advertisement.
RP24: Processing of identifiable sensitive data

Figure 4-2: Consent Regulation Properties (RP)

4.2.2.1 Non repudiation

Non-repudiation is defined as the entity submitting, holding, or asserting the information that can not be denied from it. The user may deny consent given, and DF may deny consent collected. Entities must refute the claims by achieving security property non-repudiation in the consent agreement. This can be included with RP1 and RP2 to prove who has established consent with whom.

One way to satisfy non-repudiation is to use *digital signature* schemes [105]. The signature will ensure: i) DF has obtained consent and DP can not deny it; ii) No party other than the data fiduciary can have such proofs (if implemented correctly with a combination of other cryptographic techniques).

Table 4.2: Adversarial consent processing and requisite security properties

S.No	**RP**	**Malicious Properties**	**Security Properties**	**Technological methods**
1.	RP1, RP2	DF or DP can deny consent	Non repudiation	Digital signature
2.	RP3, RP4, RP5	Data used for additional purposes and operations	Access control	anonymization, access control methods
3.	RP10, RP11, RP12, RP13, RP14	The consent may be modified, tampered	Integrity, tamper-proof	Blockchain, future research
4.	RP16, RP17, RP18	Data may be shared with additional parties	Transparency, integrity, and access control	Blockchain, future research
5.	RP19, RP20	data may be stored at different locations	Geolocation securities	Future research
6.	RP21, RP22, RP23	Data may be processed violating the norms	Correct children consent	Audit, parental consent
7.	RP24	Identifiable information may be misused	identifiable consent (access control and anonymization)	Blockchain, privacy preserving identity verification

4.2.2.2 Access control

During consent retrieval, DF has to declare the purpose of consent, the type of data, and the possible set of operations on collected data. In an adversarial setting, the collected data may be used for other purposes, or additional data may be collected violating the consent parameter. Therefore, DF should implement security properties to ensure: i) he is not using data for other purposes (RP3). ii) Additional data is not collected, or collected data is minimal (RP4). iii) only specified operation is performed on collected data (RP5).

One approach to satisfy the above requirements is the integration of distinct access controls. The access control will limit the data collection and operations performed over it. Few access control approaches have been discussed in [157]. For instance, the use of Client Side Encryption (CSE). As CSE encrypts data on the client side, the probability of processing data for additional purposes can be minimized. Similarly, access control methods such as attribute-based, user-based, de-identification, and anonymization [109] will limit the purpose, restrict data collection and operation, and provide finer control.

It should also be noted that restriction on data will not completely prohibit adversarial use because once data is handed over, the possibility of misuse is always possible. Therefore, we require greater access control and consent and data collection restrictions. We have discussed these in Layer 3.

4.2.2.3 Integrity and tamper-proofness

The consent urges DF to include information on time, location, etc. (Ref Fig. 4-2). If DF is not trusted, the integrity of data is unreliable as he can modify the given consent, construct a false consent, or collect data without the user's agreement.

Encoding a few security properties will help restrict data tampering and integrity. It primarily includes the correct data logging and variables to prove correct modification, withdrawal, and prevention of data processing after consent withdrawal.

Blockchain-based technology can provide the solution for many of the above security properties. However, we need future research to design each goal. We have proposed a model *SCM* in Section 4.4 complying with certain integrity properties.

4.2.2.4 Security in third parties data sharing

Consent emphasis is to induce the list of third parties with whom data will be shared. The properties RP16, RP17, and RP18 are associated with it. In case of malicious DF, the compliance may not be congruously followed, and data may be shared with additional third parties not specified in the consent.

We require security properties for:
i) Transparency: data is only shared with parties declared in the consent.
ii) Access control: only pertinent information is shared with third parties.
iii) Integrity: Ensure that consent and shared data comply.

Accomplishment of these properties urges to define methodologies and future research.

4.2.2.5 Geo-location securities

Data fiduciary should maintain security properties to comply with Geo-location transfer of data-basedd restrictions(if applicable). The security properties for this urge to define methodologies and future research.

4.2.2.6 Children Data consent

The DF should maintain security properties for a transparent implementation of parental consent. He should also enable security properties to ensure that he is not performing any profiling, tracking, or target-based advertisement aimed at children.

Many *parental control* methods exist which enable guardians to control children's online behavior [113]. The correct security implementation will enable children to effectively use the internet and prevent them from being a victim of malicious DF [102].

4.2.2.7 Identity collection and consent

Many DFs may collect to obtain users' identifiable information (e.g., biometric, Aadhar, etc.). Identifiable information is sensitive, especially when customer data may be stored globally.

A secure mechanism is necessary for identifiable data processing. Privacy-preserving identification, blockchain, and Aadhar 2.0-based identity verification are the advanced methods that can provide compliance-oriented identifiable data processing.

4.2.3 Layer 3: Consent as access control (CAAC)

Layer 1 and Layer 2 assume consent retrieval and data collection as separate processes. This separation makes many things critical. It leaves a space for an adversarial service provider to misuse the data. For instance, once data is delivered to the hand of the service provider, it may be used for additional purposes and operations. Thus, the processing of advanced levels of consent compliance urges more strong restrictions on consent retrieval and data collection.

Binding: The robust restriction can be achieved by binding data and consent together. As a validation attribute, the control can be hitched with the data itself to fulfill the specified consent criteria. Encoding advanced cryptographic methods may help specify the pertinent binding to achieve this goal. We can understand it from some examples:

1. **Example 1:** Consider a situation where DF wants to operate to compute the threshold age of a person (anonymous age verification).

2. **Example 2** DF wants to consent to perform some computation on data without knowing the actual data (e.g., secure multiparty computation (SMPC)).

The above examples do not ask for data directly. Instead, they perform cryptographic computation. So, if DF asks the user's consent to compute something (say, age) on data, he can perform specified operations without knowing the actual data. Here, the definition of input data itself defines the consent conditions. Consent and data collection are not independent; they are bound together. Such processing will minimize the adversarial impact. SMPC, searchable encryption, and ZKP are any other advanced cryptographic techniques that can be useful in achieving such a goal.

4.3 Proofs of consent (PoC)

The Data Protection Bill presents consent from a legal perspective. This means that parties are questionable in court if they deviate from the terms and conditions in the consent form. Legal action is taken after an event and requires shreds of evidence. This evidence can be (possibly) encoded within the software to ensure that the consent properties of the Layer 1, Layer 2, and Layer 3 are implemented correctly. We call such encoded evidence or proofs as *proofs of consent (PoC)*. When we say "proof for consent properties," it essentially means that the construction of proof would be such that deviation would not be possible by the parties included in the proofs.

4.4 Shielded Consent Manager(SCM)

The Android user permits apps to access Android resources (e.g., camera, storage, SMS, etc). The consent may be at *installation time* or *run time*. In the second category, the system provides *"only this time, always, or not allowed"* options. It allows features to disable the permission manually if an app has not used a resource for a longer period. The Android system also stores an app's logs of accessed resources for some period. The current consent to access Android resources has the following limitations: i) consent has limited choices (thus requires more, e.g., consent may be given for three days); ii) the App can modify the consent agreement and permissions over time without the user awareness (integrity of consent agreement is necessary) iii) Resource access logs may be modified by user and apps (integrity of logs is required). iv) The app may deny consent collected v) Access logs are not stored for a longer period (longer duration may provide better visibility of historical data collected).

With the above gaps and motivation from Sect. 4.2, we propose the model of *Shielded Consent Manager (SCM)* to provide security properties integrity, non-repudiation, and auditability. The model integrates the consent representation format of Fig. 4.1.

We use blockchain state channel [64] and Hash function. A state channel provides a scalable blockchain solution where users may create a smart contract and perform many transactions by writing only the first and last transactions to the blockchain.

Fig. 4-3 contains a description, and the pictorial interaction is shown in Fig. 4-4. Here, a channel is created over the blockchain to establish an agreement. An agreement A is prepared by both user U and service provider (SP) of the App, including consent variables, requisite crypto parameters, and policies. The channel created using smart contract stores the hash of the agreement h_A, and a hash of secrets a_1, b_1. The secrets may be disclosed in dispute or while closing the channel. The state channel includes other parameters $param_U, param_{SP}$ such as account address, initial funds, etc. The channel will work as integrity and tamper-proof evidence for the consent.

The SP can request access to desired resources after the agreement. The android resource manager will first check the status of the consent agreement from the consent manager. If consent is valid, then it initializes variables to log the request. Since parties may tamper with the log, tamper-proof auditable logs are necessary before granting the resource request. Both parties create a log as $(t, log_{pq)}$ for each request t. The auditability and tamper-proof come from the definition of state channel and cryptographic hash as a_1, b_1 are secrets. Therefore variable $log_{pq} = H(b_1||t_U||q||p)$ can not be modified. Both parties may disclose their secrets to resolve the dispute in case of dispute. The detailed description of the proof is omitted.

SCM$[U,\text{SP},\text{R},B]$

A. Consent creation

1. The android systems has resources $R = (r_1, r_2, r_3, ..r_i)$. U and SP selects random number a_1, b_1, and compute $x \leftarrow H(a_1), y \leftarrow H(b_1)$, respectively.
2. SP provides a consent form f containing a well defined consent representation, variable, policies and cryptographic parameters.
3. U fills the form and prepare the agreement A, compute $h_A \leftarrow H(A)$, and share (A, h_A) with SP. SP also compute the same and verify it.
4. Parties U and SP creates and initialize a state channel over blockchain B as : $S \leftarrow open(h_A, param_U, param_{SP}, x, y)$.

B. Resource Access

5. SP request a token from U to access a resource r_j.
6. Android resource manager, verifies if consent is valid. Then, for each request, he generates a token t as: select a random number q, computes $t_U \leftarrow H(a_1||q)$, and send (t_U, q) to SP.
7. SP generates a random number p for each request,computes the logging variable $log_{pq} \leftarrow H(b_1||t_U||q||p)$ and share (log_{pq}, p).
8. The resource manager logs the value (t, log_{pq}, p, q) and provide the resource access.

C: Consent Update

9. DF provides a consent form f' with updated policies. Follow steps 1 to step 3 furthr.
10. Update the state channel as $S' \leftarrow update(S, h'_A, param_u, praram_{SP}, x', y')$.
11. To access resources, follow the Steps 5-8 with the modified value x' and y'.

C: Consent Withdraw

10. The parties may call *close()* operation to close the channel with state S'.

Figure 4-3: Description of *Shielded Consent Manager (SCM)*

The consent update and withdraw option is performed through an update of the state channel and other variables.

The security of variables on the blockchain (such as users' secret keys and the online availability of users while closing the state channel in case of disputes) and pre-image of the hash function will imply the security of the model. The detailed validation is omitted here.

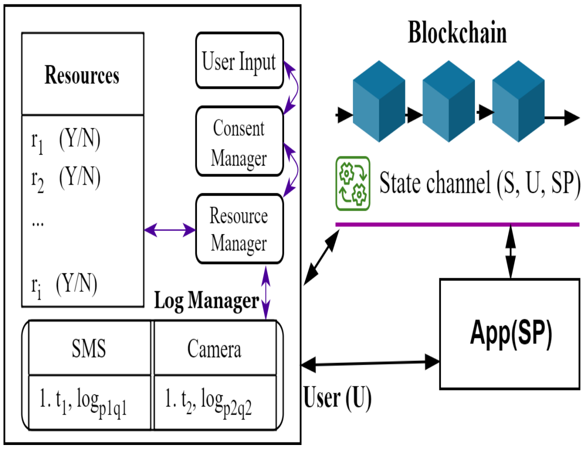


Figure 4-4: Shielded Consent Manager (SCM)

4.4.1 Implementation and practical considerations

We have done the feasibility study of the implementation of the SCM model and simulated the contract code for the consent agreement. We have taken a test file of 1 MB consisting of consent representation in the format as defined in Fig. 4.1. We assume the size of random variables is 32 Bytes. The Cryptographic hashes of files and other variables are computed using SHA256().

The SHA256() and random numbers are simulated on Python ”3.10.10”. The smart contract and state channel are created and deployed using Truffle v5.7.7 (core: 5.7.7), Ethereum client Ganache v7.7.5, and Solidity - 0.8.18 (solc-js). Simulation is set up on the machine running with Windows 10, CPU Intel(R) Core(TM) i7-1165G7 @ 2.80GHz 16 GB RAM.

The user and SP create a state channel named ”ConsentChannel” with funding of initial balance and passing of two arguments (consisting of file hash and a hash of secrets), each of 32 bytes. We have created three functions: Update(), updateChanelBalance(), and Close(). The update function exchanges messages off-chain to update the modified consent agreement values. Both the user and SP sign the update message variable to prevent one party from updating the consent without the

Table 4.3: SCM Simulation results

Parameters	**Values**
Storage overhead	128 (bytes)
Gas used	1177827 unit
Total cost	1.0038703 ETH

approval of another party. updateChanelBalance() is used to update the balance of the channel. Both parties can perform on-chain transactions to close the channel.

The codes are deployed using *truffle*. The client *ganache* is also integrated to deploy the smart contract on the local Ethereum network. The gas and ether cost in deploying the contract is mentioned in Table 4.3.

The model requires additional communication between parties to establish the state channel and log the cryptographic parameters. The storage overhead is computed for each token request and is at least 128 bytes (including one token index, two random numbers, and one hash, each 32 bytes).

The analysis indicates only two transactions are required over the blockchain, and the rest of the processing and storage can be done offline on both sides. The exact comparative model to compare the results is not available in the literature. The works [174], and [145] emphasize storing every consent and data log over the blockchain, which may not be cost-effective. The SCM aimed to optimize the number of transactions over the blockchain and urge to store minimum data. The total cost and gas used for the transaction are slightly high, which may be reduced in future optimization.

Chapter 5

Digital Asset Inheritance

An *asset* is any substance or resource (tangible or intangible) that has an economic value and future benefit. Due to the recent breathtaking development of the Internet, digital communication is now an integral part of people's lifestyles, and such an expansion has generated *digital assets* exponentially over cyberspace. The data associated with email, social media platforms (such as Facebook, Instagram, Twitter, etc.), user's website, blogs, data stored in the cloud (such as pictures, photos, diaries, videos, songs, books), online wallets, coupons, and gifts cards are few examples of digital assets. The *asset inheritance* system assists the dependent in the absence of an asset holder (a.k.a user). People can convey their property in the present or future to one or more living person(s). Currently, organizations authorize inheritance only for financial assets. However, due to digitization, the value and importance of digital data have increased, which appeals to *asset inheritance* and *asset management* of digital assets as well.

The traditional inheritance model of financial assets has been designed under the assumption that the nominee is aware of the asset and can inherit it later after the user's demise. However, this model fails to guarantee inheritance if the nominee is uninformed regarding assets. This can be seen in media reports that banks and insurance companies have many dormant accounts with billions of unclaimed assets [167]. Furthermore, the conventional inheritance model is not directly applicable to digital assets because the nature and category of assets are immensely diversified. For instance, an Internet user's account may not be associated with a real user name (such as an account created on a cloud platform) or an account holding decentralized cryptocurrencies solely managed by the user. These assets may be lost forever if the transfer process is not robust.

Thus, the critical concern is how the *digital asset inheritance* (DAI) can be modeled. Few works in this direction have proposed the idea of DAI for user-owned assets like cryptocurrency, passwords, cryptographic private keys, etc. For instance, Digipulse [166] stores assets in encrypted form and transfers to the descendants using blockchain technology; PassOn [168] stores assets in the form of a token; SafeHaven [169] uses a secret sharing scheme to distribute cryptocurrency private key; and TrustVerse [170] is designed for asset management using the smart contract. However, we observe that none of these models specify their inheritance design. In particular, it is not known how the asset information will be stored securely, how the user's death will be confirmed correctly, how many nominees will participate, and how dependents will get to know about the asset after the user's demise. These models also fail to explain how the asset transfer will work if the dependent does not know about it.

Transfer and inheritance of digital assets are highly overlooked and have not received the research community's appropriate attention [153]. All the digital assets are worthwhile, and the user would like to transfer them to his family members. Therefore, a simple and robust model must be designed so a nominee can get information regarding the digital assets left after the user's demise. The nominee can inherit it efficiently without any loss, given that he may or may not know about the assets the user owns before death.

In this work, we have solved the above problems. We have formalized the category of digital assets and defined the functionalities and security goals necessary to design a DAI model. We have proposed a digital asset inheritance protocol (DAIP) based on certificateless encryption that stores and manages the user's asset or asset information. After the user's demise, such information can be conveyed to the nominee efficiently and used to inherit the asset.

5.0.1 Problem formulation

DPDPA mentions the obligation of the "right to nominate". As per the Act *"A Data Principal shall have the right to nominate, in such manner as may be prescribed, any other individual, who shall, in the event of death or incapacity of the Data Principal.."*.

It is SP's responsibility to facilitate the right to nominate. However, the nature and category of personal data processed by DFs are diversified, which may make asset inheritance difficult. Using technological models, we explore how the right to nominate can facilitate DPs efficiently in various scenarios.

The research community comprehended the significance of digital asset inheritance and started appraising it from different aspects. To design an effective DAI model, the following needs to be addressed: i) the creation of well-defined state and federal-level digital asset laws and policies [108, 167, 179, 104, 134]; ii) the development of adequate organizational level policies [37, 93, 120]; and iii) the creation of a robust technological model. In this paper, we explore and formalize only the last one, the technical design of DAI, while the first two are out of this paper's scope.

In the existing system, few organizations practice provisional methods of a legacy contract [47] to grant access to the user's account after his death. However, this is not a full-fledged inheritance model. Recently, DigiPulse [166], PassOn [168], SafeHaven [169], and TrustVerse [170] have introduced the notion of asset inheritance and asset management systems. However, these models have only proposed ideas without straightforward design and have the following issues and challenges: i) Designed for inheriting only *user-owned assets* (such as cryptocurrency key, password, or any other data) and does not discuss other categories of digital assets such as organization-owned data; ii) Require excessive nominee participation during creating, storing and managing the asset which increases the overall complexity in terms of design; iii) Uses inaccurate ways of death confirmation such as inactivity period [166], voting [168], weighted voting [168], consent of nominee [170]. All these death confirmation methods make the DAI protocol fail to guarantee the *robustness* property (as well as other security properties) if one nominee denies voting during the death confirmation process; and iv) Does not specify how the secret key required to access the assets will be transferred to the descendants in [166, 168, 170]. Therefore, current asset management models are very early and relatively immature.

The above challenges motivated us to design a model to solve the above limitations. In particular, we ask the following questions:

- What are the different categories of digital assets? How and in which form should all asset categories be managed so they can be inherited efficiently?
- How can a user securely store digital asset information to be conveyed to the descendent after his death?
- What should be the functionalities and properties of the asset inheritance model?
- How can a nominee successfully claim and retrieve all the digital assets after the user's death?
- Can we force each data fiduciary to hand over all the digital assets to the nominee when they are alive?

Integrating a robust asset inheritance model, particularly with the existing platform, may greatly help a nation, especially during COVID-19, where many died and left their assets without informing their nominees. Our contributions are as follows:

Our first contribution is formally modeling the digital asset inheritance (DAI) system. In particular, we have described the different categories of digital assets, entities involved in the system, design functionalities, security goals, and various constraints associated with the DAI model.

Next, we have designed a digital asset inheritance protocol, DAIP, that adheres to the required design functionalities and achieves certain security properties. Our DAI protocol uses certificateless encryption and some basic cryptographic primitives like hash functions to achieve the desired design functionalities and security goals. Our protocol also solves one of the important challenges, namely the identifiability problem, that exists in the organization to verify the identity of a pseudonymous user correctly after the death.

Thereafter, we have given rigorous security analysis of our protocol to prove that it ensures various security properties (such as *asset privacy*) using the real/ideal simulation paradigm of *universal composability* (UC) framework.

Next, we establish that our DAIP is better than the existing ones both in terms of design functionalities as well as securities properties (see Table 5.1 and 5.2 for the details). We have discussed the platform utilization of the DAIP model, applications, benefits to stakeholders and service providers, and how it can be integrated with the existing infrastructure (refer Sect. 5.6.1 for the details). Finally, we simulated and analyzed the implementation of DAIP and showed that the protocol is efficient for many users.

Organization: The rest of the sections in the chapter are organized as follows. We start by modeling the digital asset inheritance (DAI) system in Sect. 5.1, which describes digital assets, design functionalities, and security goals. Thereafter, in Sect. 5.2, we describe various cryptographic primitives as preliminaries. We present the full description of our DAI protocol along with formal security proof in Sects. 5.3 and 5.4. Then, we compare our newly designed protocol with the existing ones, in terms of design and security, in Sect. 5.5. Finally, we discuss the application and practical implementation of DAIP in Sect. 5.6.

5.1 Digital Asset Inheritance System

A user must plan to pass his online persona to his successor after his demise. The model that deals with the transfer of such assets is known as the Digital Asset Inheritance (DAI) system. The construction of inheritance models is very early, and no formal construction is defined yet. This section formalizes the design functionalities, entities required, and security goals necessary to develop an inheritance model.

5.1.1 Digital asset

We define *digital asset* as any valuable data that exists over cyberspace, that a user wants to pass-on to his descendants. The *asset information* is a set of parameters declared by a user during inheritance declaration. It may include user ID, account number, registration number, or any other user-specific variables used to identify and claim an asset by the descendants. Digital asset exists in many forms. At a very high level, we can divide the asset into four major categories:

TYPE 1 (ORGANIZATION MANAGED DATA (OMD)). In this category, data is entirely maintained and managed by the organization (a.k.a. data fiduciary), such as web services, cloud data, songs, and videos. The users can store, update, delete, and manage the data with access control.

Based on the nature of the organization's data and policy, it should register nominee details so that the asset can be transferred to the descendants after his demise. We want to point out that most organizations are not concerned about it currently.

TYPE 2 (ORGANIZATION MANAGED MONETARY ASSET (OMMA)). An organization that manages users' financial assets, having monetary value, comes under this category. It can be local or global currency, bank accounts, insurance, stock market shares, mutual funds, or e-wallets. This category of data mostly belongs to identifiable users.

Users can keep all these account information together so that even if a nominee is unaware, they can know it in the future.

TYPE 3 (USER OWNED DATA (UOD)). These personal data are only known to the users, and the organization's participation is unnecessary. Users' sensitive personal data, cryptographic private keys, passwords, cryptocurrency keys, or non-bankable assets are examples of data that come under this category. It can also include any data that are managed and stored solely by the user itself, such as any personal files, books, songs, etc.

TYPE 4 (MIXED CATEGORY). This category of data consists of two parts. i) a secret key and ii) encrypted data. The user owns and manages the key while the organization stores encrypted data (for example, digital will or data stored with client-side encryption).

The user has to add nominee details at the organization and plan the transfer of the secret key to the descendants.

The user needs to convey either the complete asset (e.g., user-owned data, a cryptographic key) or the asset information (e.g., account number) to the descendants based on the category of data.

Our DAIP is designed for type 1 and type 4 categories of assets. Nevertheless, it can be easily extended for type 2 and type 3 categories of assets.

5.1.2 Entities involved

This section defines the primary entities required to construct the DAI model. The details of each entity, its role, and its behavior are described below:

USER (U) is the owner of the digital asset who plans to pass its asset to his successors. We divide the users into two categories: (i) *pseudonym users*, who have not registered themselves with any real identity at the organization. These users may or may not be identified correctly (for example, email id, Github or Twitter handle); (ii) *identifiable users*, who have registered to the organization using real identity such as bank accounts, mutual funds, etc.

NOMINEE (N) is a legal custodian of the asset. Anyone can become a nominee if they fulfill the legal requirements. Generally, preferred nominees are family members and close relatives, but in certain circumstances, anyone could be a nominee if state or federal laws allow it. The user can define single or multiple nominees against an asset.

ORGANIZATION (O) stores user's data for type 1, 2, or 4 category assets. It shall allow users to declare nominee details. A nominee can communicate with the organization upon the user's death and receive the asset after necessary verification. Based on the nature of the service offered, an organization may register both pseudonyms and identifiable users. Note that the organization must confirm and verify users' identity correctly before transferring an asset to the nominee for the pseudonym users.

CERTIFICATE AUTHORITY PORTAL (CAP) is the portal that confirms the user's death. It can be any trusted entity that shall approve demise correctly. In this model, we have assumed CAP is a central entity authorized by the government within the state, solely responsible for confirmation of death and generation of *death certificate* (defined in a later section) upon the request by their belongings.

NOMINEE DISPLAY PORTAL (NDP) is a dedicated system for efficiently storing and managing asset inheritance data (AID). NDP implementation can be done either using a dedicated centralized system or a decentralized system such as blockchain.

IDENTITY BASED SYSTEM (*IBS*) stores the identity (id_U) of users within the system. IBS could be assumed to be any identity-based system, such as the Aadhar system in India or the SSIN number in the USA. It will help entities to confirm and verify users' identities after demise. IBS has the functionality of an authentication service. Any user(u, id_U) can authenticate himself uniquely and correctly using this service.

5.1.3 Design functionalities

This section defines the core functionalities required to design an asset inheritance model. It includes the following:

AID CREATION. The asset inheritance model should implement functionality allowing users to create *Asset Inheritance Data (AID)*. It may consist of a list of asset or asset information, a list of nominees, secrets, and any other information per the requirement. The nominee will receive AID after the death of the user. If an organization stores the asset, communication with the organization is also required during AID creation.

AID STORAGE. The DAI model should implement functionality that allows users to *store and manage* the AID created in the previous step. Storage functionality shall enable the user to edit, update, or delete AID.

NOMINEE PARTICIPATION. The DAI model should define up to which extent and in what manner nominee participation is required. Note that the DAI model should use the minimum participation during asset creation and execution of the protocol.

DEATH CONFIRMATION. The DAI model should define a robust and accurate method for death confirmation since security, privacy, and AID transfer depend on the correct implementation of death confirmation.

AID RETRIEVAL. The DAI model should allow the nominee to retrieve the AID from the storage functionality after a user's death.

ASSET TRANSFER. The DAI model should implement functionality that allows the nominee to reconstruct secrets and recover asset or asset information. Note that the asset recovery will also require communication with the organization if stored at the organization.

5.1.4 Security goals

This section defines the various security goals required to securely design a digital asset inheritance protocol (DAIP).

ASSET PRIVACY. A DAI model is said to achieve *asset privacy* if AID stored by the user at the NDP portal does not reveal anything before death to NDP and any other third party. Note that after a user's death, an NDP will only learn about the mapping of a pseudonymous user with the real identity. However, the assets' secrets and worth will be revealed only to the nominee.

NON-REPUDIATION. A DAI model is said to achieve *non-repudiation* if the organization cannot deny the asset's holding when the nominee requests it. This definition holds for both pseudonymous and identifiable users and data category type 1, type 2, and type 4.

IDENTIFIABILITY. A DAI model is said to achieve *identifiability* if it fulfills the following conditions: i) the user's identity is not known to the organization before death, and ii) an organization knows the identity of the user *only* after the death. This definition holds only for the data category type 1 and type 4, where the organization provides service to pseudonym users.

Currently, in type 1 and type 4 categories of assets, the users are not registered at the organization with any real identity. Thus, his physical identity can not be known accurately and correctly by the organization. Therefore, there is a need to design a functionality to identify a user correctly after death and transfer the online persona to his descendant.

KEY INHERITANCE PRIVACY. A DAI model is said to achieve *key inheritance privacy* if secret keys are revealed only to the designated nominee assigned against an asset. This property is important because sometimes, a user does not want to disclose all the assets to everyone.

USER PRIVACY. A DAI model is said to achieve *user privacy* if the asset inheritance data stored at NDP does not reveal the user's identity before his death.

ROBUSTNESS. The DAI model is said to be *robust* if the nominee retrieves all the asset or asset information from AID successfully without fail.

CORRECTNESS. A DAI model is said to be *correct* if it fulfills the following: i) the organization can confirm and validate the user's identity; and ii) the organization can correctly validate the nominee before transferring the asset.

5.1.5 Concern of correct death confirmation

Digital asset inheritance protocol requires the assumption of a trusted party who can confirm the death correctly.

Death is purely unpredictable. It does not depend on any physical parameters and real-world constraints. The existing models of asset management use the following mechanisms to confirm the death [166, 170, 169, 168]: i) ***voting:*** In this method, all nominees vote to confirm the death. However, if any nominee denies voting, the asset can not be recovered; ii) ***weighted voting:*** It allows confirmation of death by a threshold number of nominees, each having different weights. Although, if some nominees with higher weights deny voting, the threshold can not be achieved; iii) ***notary:*** is a legal person who confirms the user's death and also stores the secret key which will be used later by the nominee to learn about the asset. However, if the notary denies participating in the protocol, then the nominee can never retrieve this secret key; iv) ***inactivity period:*** In this method, the user's logged-in status is collected from different social media accounts (for example, E-mail, Facebook). The death is confirmed if a user has not logged in since a pre-defined time. However, due to the privacy policy, many websites may not share login details with other organizations. Further, if a user is alive and forgot to update his liveliness status through account login, all his secrets would be revealed to the nominee, and v) ***secret sharing:*** In this method, the multiple nominees hold the shares of the key. After the user's death, nominees submit their claims and recover the key. This method does not validate death correctly, as all the nominees may collude with each other.

The incorrect execution of death confirmation may adversely affect the model. Therefore, we need a *trusted* entity to correctly confirm the user's demise; otherwise, the complete model will fail.

5.2 Preliminaries

Our construction requires various well-known cryptographic schemes, namely, a symmetric-key encryption scheme $\mathsf{SKS} = (\mathsf{SKS.Setup}, \mathsf{SKS.Enc},$
$\mathsf{SKS.Dec})$, a signature scheme $\Sigma = (\Sigma.\mathsf{Setup}, \Sigma.\mathsf{Sign}, \Sigma.\mathsf{Verify})$, a pseudo-random function PRF, and a hash function H. For better readability, the rigorous definitions of these schemes are defined in 5.2.2. In this section, we define the following primitive in detail.

5.2.1 Certificateless Encryption (CLE) Scheme

Definition 5.1 (Certificateless Encryption [34])**.** *A certificateless encryption $\mathsf{CLE} = \{\mathsf{CLE.KeyGen}, \mathsf{CLE.Enc}, \mathsf{CLE.Extract}, \mathsf{CLE.Dec}\}$ is a 5-tuple of algorithms over the setup algorithm $\mathsf{CLE.Setup}$, that works as follows:*

- $\mathsf{CLE.Setup}(1^\lambda) \rightarrow (mpk, msk)$*: It is executed by the key generation center (KGC). On input of the security parameter 1^λ, it returns $(mpk.msk)$, where mpk is the master public key, and msk is the master private key. The master public key is distributed, and msk is kept secret by KGC.*

- $\mathsf{CLE.KeyGen}(mpk, id) \rightarrow (pk, sk)$*: It is executed by the receiver to generate a public/secret key pair. On input (mpk, id), it returns the key pair (pk, sk) where pk is the public key, and sk is the secret key. Note that the public key generated using this algorithm does not need to be authenticated with a digital certificate.*

- $\mathsf{CLE.Enc}(mpk, pk, id, m) \rightarrow c$*: It is executed by the sender to send a message m to the receiver. On input of the parameters (mpk, msk, id), and a message m drawn from the message space M, it returns either a ciphertext $c \in \mathcal{C}$ or $\perp$ indicating error that the public key is not valid for the identity id.*

- $\mathsf{CLE.Extract}(mpk, msk, id) \rightarrow sk'$*: KGC executes it to create partial private key s for the user's identity id. It takes master secret key msk, master public key mpk, and id as a parameter and returns partial private key sk'.*

- $\mathsf{CLE.Dec}(mpk, sk, sk', c) \rightarrow m$*: It is executed by the receiver to decrypt a ciphertext. It takes inputs, the master public key mpk, the receiver's secret value sk, the receiver's partial* *private key sk', and ciphertext c, and returns either a message $m \in M$ or $\perp$ indicating that the ciphertext is invalid. The above algorithm satisfies the following property:*

The above algorithm satisfies the following property:

- *Correctness: For every pair (pk, sk, sk', id) and for every $m \in \mathcal{M}$, the following holds:* $Pr[\textsf{CLE.Dec}(mpk, sk, sk', \textsf{CLE.Enc}(mpk, pk, id,$ $m)) = m] = 1.$
- *Indistinguishability: A CLE scheme is semantically secure against an adaptive chosen ciphertext attack ("IND-CCA secure") if no polynomially bounded adversary $\mathcal{A}$ of Type I or Type II has a non-negligible advantage against the challenger in the guessing game. For details, see [34].*

The concrete instantiation of this scheme is as follows:

5.2.2 SKS, signature, and PRF

Definition 5.2 (Symmetric-key Encryption Scheme [79])**.** *A Symmetric-key encryption scheme $\textsf{SKS} = (\textsf{SKS.Enc}, \textsf{SKS.Dec})$ is a 2-tuple of algorithms over the setup algorithm $\textsf{SKS.Setup}$ and the message space $\mathcal{M}$ that works as follows:*

- $\textsf{SKS.Setup}(1^\lambda) \to k$*: On input of the security parameter 1^λ, it returns sk, where k is the symmetric key to be used in encryption and decryption.*
- $\textsf{SKS.Enc}(k, m) \to ct$*: On input the message m, it returns ciphertext ct corresponding to $m \in \mathcal{M}$ under the symmetric key k.*
- $\textsf{SKS.Dec}(k, ct) \to m$*: On input of the symmetric key k and the ciphertext ct, it returns the message m.*

The above algorithm satisfies the following property:

- *Correctness: for every key $k \in \mathcal{K}$ and for every $m \in \mathcal{M}$, the following holds:* $Pr\big[ct = \perp\ OR\ SKS.Dec(k, ct) = m : ct \xleftarrow{\$} SKS.Enc(k, m)\big] = 1$

Definition 5.3 (Signature Scheme [60, 80])**.** *A signature scheme $\Sigma = (\Sigma.\textsf{Sign}, \Sigma.\textsf{Verify})$ is a 2-tuple of algorithms over the setup algorithm $\Sigma.\textsf{Setup}$ and the message space $\mathcal{M}$ that works as follows:*

- $\Sigma.\textsf{\textit{Setup}}(1^\lambda) \to (mpk, msk)$*: On input the security parameter* 1^λ*, it returns* (mpk, msk)*, where* mpk *is the verification key and* msk *is the signing key.*

- $\Sigma.\textsf{\textit{Sig}}(msk, m) \to \sigma$*: On input the message* m*, it returns signature* σ *corresponding to* $m \in \mathcal{M}$ *under the signing key* msk*.*

- $\Sigma.\textsf{\textit{Verify}}(mpk, m, \sigma) \to b$*: On input the message* m *and signature* σ*, it returns a bit* $b \in \{0, 1\}$*, where:* $b = 1$ *if the pair* (m, σ) *is verified under the verification key* mpk*.*

The above algorithm satisfies the following property:

- *Correctness: For every pair* (mpk, msk) *and for every* $m \in \mathcal{M}$*, the following holds:*

$$Pr\Big[\Sigma.\textsf{\textit{Verify}}(mpk, m, \sigma) \to 1 \\ \mid \Sigma.\textsf{\textit{Sig}}(msk, m) \to \sigma\Big] = 1$$

- *Unforgeability: A signature scheme is* unforgeable under an adaptive chosen message attack*, if for any PPT adversary* $\mathcal{A}$*, and a negligible function* μ*, the following holds:*

$$Pr\Big[\Sigma.\textsf{\textit{Setup}}(1^\lambda) \to (mpk, msk) \\ \wedge \mathcal{A}^{\Sigma.\textsf{\textit{Sign}}(msk,\cdot)}(pk) \to (m', \sigma') \\ \wedge \Sigma.\textsf{\textit{Ver}}(vk, m', \sigma') \to 1 \mid m' \notin M\Big] \; < \; \mu(1^\lambda),$$

where M *is the set of messages submitted by* $\mathcal{A}$ *to the* $\textsf{Sign}$ *oracle.*

Definition 5.4 (Pseudo-random function [79])**.** *A family of function* $F_K : \{0,1\}^n \to \{0,1\}^m$*, indexed by a key* $K \in \{0,1\}^s$ *is said to be a* pseudo-random function *(PRF) if it satisfies the following:*

- *Given a key* $K \in \{0,1\}^s$ *and an input* $X \in \{0,1\}^n$ *there is an efficient algorithm to compute* $F_K(X)$*.*

It satisfies the following property:

- *Indistinguishability: For all probabilistic polynomial time distinguisher D, there exists a negligible function $\mu(\cdot)$ such that:*

  $$\left| Pr_{K \leftarrow \{0,1\}^s}[D^{F_K(\cdot)}] - Pr_{f \in \mathcal{F}}[D^{f(\cdot)}] \right| < \mu(\lambda)$$

  where $\mathcal{F} = \{f : \{0,1\}^n \rightarrow \{0,1\}^m\}$.

Definition 5.5 (Collision Resistant Hash Function [55])**.** *A collision free hash function family $\mathcal{H}$ is an infinite family of finite sets $\{H_m\}_{m=1}^{\infty}$ and a polynomially bounded function $t : N \rightarrow N$.*

A member H_m is a function $h : \{0,1\}^* \rightarrow \{0,1\}^{t(m)}$, and is called an instance of $\mathcal{H}$ of size m.

$\mathcal{H}$ must satisfy the following:

- *Given a value of m, there is a probabilistic polynomial (in m) time algorithm Θ which on input m selects an instance of $\mathcal{H}$ of size m at random.*

- *For any instance $h \in H_m$ and $x \in \{0,1\}^*$, $h(x)$ is easy to compute, i.e. computable in time polynomial both in m and $|x|$.*

- *Given an instance $h \in \mathcal{H}$ selected randomly as in (1), it is hard to find $x, y \in \{0,1\}^*$, such that $h(x) = h(y)$ and $x \neq y$.*

  More formally, for any probabilistic polynomial time algorithm $\mathcal{A}$, and any polynomial P, consider the subset of instances h of size m for which $\mathcal{A}$, with probability at least $1/P(m)$, outputs $x \neq y$ such that $h(x) = h(y)$. Let $\epsilon(m)$ be the probability with which Θ selects one of these instances. Then, as a function of m, $\epsilon(m)$ vanishes faster than any polynomial fraction.

5.2.3 Construction of Certificateless Encryption (CLE) Scheme

The concrete instantiation of the CLE scheme is as follows [34]:

- CLE.Setup$(1^\lambda) \rightarrow (mpk, msk)$: It works as follows:

  1. Generate $(q, \mathbb{G}_{\nVdash}, \mathbb{G}_{\nvDash}, \eth, , \mathbb{H}_{\nVdash}, \mathbb{H}_{\nvDash})$.

     [Here, $\mathbb{G}_{\nVdash}$, $\mathbb{G}_{\nvDash}$ are groups of some prime order q, $e : \mathbb{G}_{\nVdash} \times \mathbb{G}_{\nVdash} \rightarrow \mathbb{G}_{\nvDash}$ is a pairing, generator $g \in \mathbb{G}_{\nVdash}$, $H_1 : \{0,1\}^* \rightarrow \mathbb{G}_{\nVdash}$, $H_2 : \mathbb{G}_{\nvDash} \rightarrow \{\nvdash, \nVdash\}^{\ltimes}$, the message space $\mathcal{M} \in \{0,1\}^n$, and the ciphertext space $\mathcal{C} \in \mathbb{G}_{\nVdash} \times \{\nvdash, \nVdash\}^{\ltimes}$.]

   2. Choose $s \xleftarrow{\$} \mathbb{Z}^*_{\shortparallel}$.
   3. Set $p_0 := s \cdot g$.
   4. return $mpk := (q, \mathbb{G}_{\nVdash}, \mathbb{G}_{\nvDash}, , \mathbb{H}_{\nVdash}, \mathbb{H}_{\nvDash}, \eth, \ltimes, {}_{\shortmid\nvDash})$ and $msk := s$.

- CLE.KeyGen$(mpk, id) \to (pk, sk)$: For an $id \in \{0,1\}^*$, the algorithm works as follows:

   1. Choose $s_k \xleftarrow{\$} \mathbb{Z}^*_{\shortparallel}$.
   2. Set $pk := (X, Y)$.
      [Here, $X := sk \cdot g, Y := sk \cdot p_o = sk \cdot s \cdot g$.]
   3. return (pk, sk).

- CLE.Enc$(mpk, pk, id, \mathbb{M})\to \mathbb{C}$: For a message $\mathbb{M} \in \mathcal{M}$, the algorithm works as follows:

   1. Check $X, Y \in \mathbb{G}^*_{\nVdash}$.
      [Here, $pk = (X, Y)$.]
   2. Check $e(X, p_0) \overset{?}{=} e(Y, g)$.
   3. Compute $Q = H_1(id)$.
   4. Choose a random number $r \xleftarrow{\$} \mathbb{Z}^*_{\shortparallel}$.
   5. Compute $\mathbb{C} = \{\diagdown.\eth, \mathbb{M} \oplus \mathbb{H}_{\nvDash}((\mathbb{Q}, \mathbb{Y})^{\diagdown})\}$.
   6. return $\mathbb{C}$.

- CLE.Extract$(mpk, msk, id)\to sk'$: It works as follows:

   1. Compute $Q = H_1(id)$.
   2. Set $sk' = s \cdot Q$.
   3. return sk'.

- CLE.Dec$(mpk, sk, sk', \mathbb{C})\to$ $\mathbb{M}$: For a received cipher text $\mathbb{C} = (\mathbb{U}, \mathbb{V})$, the algorithm works as follows:

   1. Compute the private key $S_{id} = (sk \cdot sk') = sk \cdot s \cdot Q$.
   2. Decrypt message as follows:
      $V \oplus H_2(e(S_{id}, U))$
      $= V \oplus H_2(e(sk \cdot s \cdot Q, r \cdot g))$
      $= V \oplus H_2(e(H_1(ID), g)^{s \cdot sk \cdot r})$
      $= \mathbb{M} \oplus \mathbb{H}_{\nvDash}((\mathbb{Q}, \mathbb{Y})^{\diagdown}) \oplus \mathbb{H}_{\nvDash}((\mathbb{H}_{\nVdash}(\mathbb{ID}), \eth)^{\sim\cdot\sim\urcorner\cdot\diagdown})$
      $= \mathbb{M} \oplus \mathbb{H}_{\nvDash}((\mathbb{H}_{\nVdash}(\mathbb{ID}), \eth^{\sim\cdot\sim\urcorner})^{\diagdown}) \oplus \mathbb{H}_{\nvDash}((\mathbb{H}_{\nVdash}(\mathbb{ID}), \eth)^{\sim\cdot\sim\urcorner\cdot\diagdown}) = \mathbb{M}$.

5.3 Digital Asset Inheritance Protocol (DAIP)

In this section, we design the asset inheritance protocol DAIP, using the CLE primitive, for type 1 and type 4 asset categories. We consider six entities: user U, nominee N, NDP, CAP, IBS, and the organization O.

Suppose a user U has an account acc_U in the organization O. Digital assets are managed, modified, and updated by U and stored at O. U holds a unique identity number id_U assigned by IBS. He wants to create an *asset inheritance data* AID at NDP to convey his persona $asset_U^O$ to his nominee N after his demise. After the user's death, the protocol should guarantee that N can obtain AID from NDP and retrieve $asset_U^O$ from O.

The protocol Π is a composition of two sub-protocols: (i) *Asset Management*, denoted Π_1, which ensures efficient storage of an asset by the user so that it can be retrieved later by the nominee after the demise of the user; (ii) *Asset Inheritance*, denoted Π_2, which ensures proper delivery of the asset to the nominee post demise of the user.

5.3.1 Asset management protocol

The *Asset Management* protocol, denoted Π_1, allows a user U to store the *asset inheritance data* AID of acc_U so that the nominee can retrieve and inherit it after the demise of the user. To design the protocol, we define the following functions:

- $\mathsf{KeyGen}(\lambda) \to r$: On input the security parameter λ, it returns a random number $r \overset{\$}{\leftarrow} \{0,1\}^\lambda$.
- $\mathsf{PseudoID}(id_U, r) \to \widetilde{id}_U$: On input the identity of user id_U and a random number r, it returns the pseudo-identity of the user $\widetilde{id}_U = \mathsf{H}(id_U \| r)$, where H is a standard hash function. The function maps a user's *unique* identity to a *pseudo-random* identity.

From the design standpoint, the protocol Π_1 consists of 4 stages, all executed by parties U, N, O, IBS, CAP, and NDP. The pictorial and algorithmic descriptions of Π_1 are given in Figs. 5-1 and 5-2, 5-3 respectively.

Stage 0 - Setup Phase: This stage aims to generate initial parameters – public/private key pairs and pseudo-identity – for the parties executing the protocol.

In this stage, the user generates an *asset key* k^O (a function of partial keys: k_1, and k_2^O) for hiding the asset or asset information. Then, he encrypts the partial key k_1 (along with the user

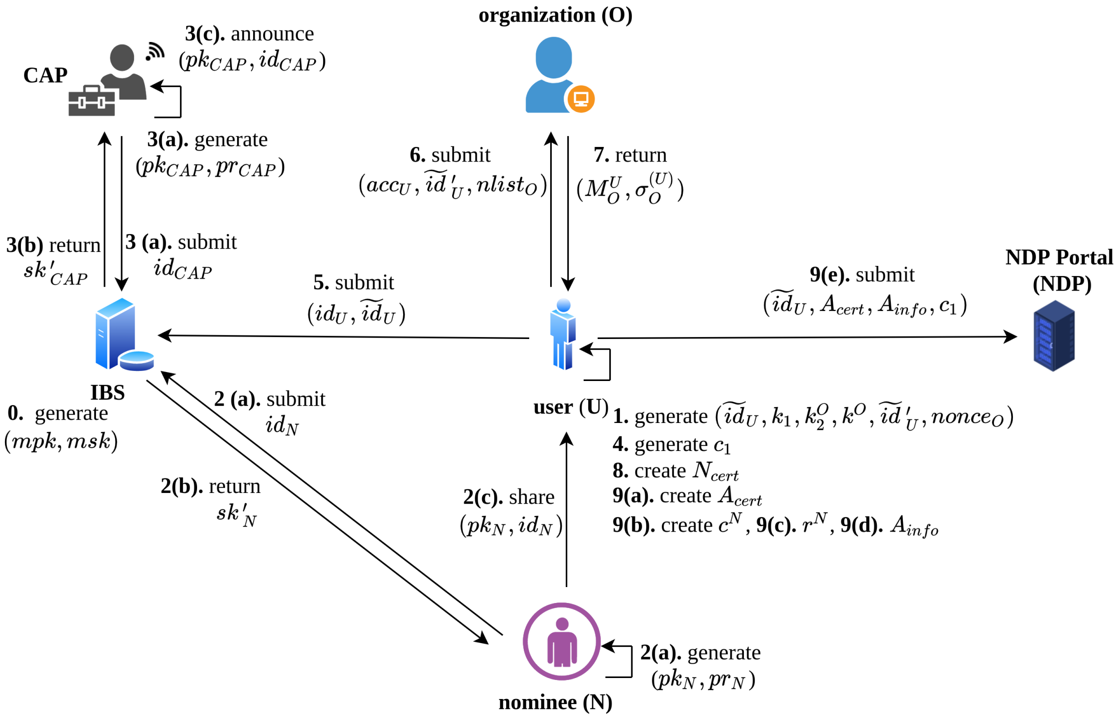


Figure 5-1: Pictorial description of *Asset Management* protocol Π_1.

identity id_U) using public parameters of CAP $(pk_{\text{CAP}}, id_{\text{CAP}})$. The asset key k^O has the following features: i) it will be revealed to the nominee only after the user's demise; ii) it will be revealed to the designated nominee only for recovering the AID.

Our model achieves this using *key distribution* approach. The two partial keys will be stored separately at different entities and later revealed to the nominee to retrieve the asset from the organization. Note that to retrieve the asset, a nominee has to re-compute the *asset key* using these partial keys.

Stage 1 - Registration: This stage aims to register the user's pseudo-identity, generated in stage 0, at IBS. Note that this mapping is known only to the user and IBS. The IBS will reveal this mapping to NDP only after the user's death.

Stage 2 - Generation of *nominee registration certificate* N_{cert} for each organization: The purpose of this stage is to generate the nominee registration certificate N_{cert}. It is a data set generated by the user after registering the nominee details with the organization.

The certificate N_{cert} confirms that the organization will provide all the assets declared in the certificate to the nominee if he fulfills the conditions, denoted $other\,details$, mentioned in the certificate, such as verification procedure, nature and category of data to be transferred, and other

$\Pi_1[U, N, O, \text{IBS}, \text{CAP}, \text{NDP}]$

Stage 0: Setup Phase

(0) **[Generation of master public-private key by IBS]** Identity-based system IBS invokes $\mathsf{CLE.Setup}(1^\lambda) \to (mpk, msk)$ and sends mpk to U, N and CAP.

1. **[Generation of pseudo-identity by user]** User U invokes the following: $\mathsf{KeyGen}(\lambda) \to r$, $\mathsf{PseudoID}(id_U, r) \to \widetilde{id}U$, $\mathsf{KeyGen}(\lambda) \to k_1$, $\mathsf{KeyGen}(\lambda) \to k_2^O$, $\mathsf{PRF}(k_1, k_2^O) \to k^O$.

   [Here, k_1 denotes partial key of the user, k_2^O denotes partial key for each organization, k^O denotes asset encryption key, id_U denotes the identity of the user registered with IBS, and $\widetilde{id}_U$ denotes the pseudo identity of U].

2. **[Generation of key parameter by nominee]** Execute the following:

   (a) Nominee N invokes the key generation algorithm $\mathsf{CLE.KeyGen}(mpk, id_N) \to (pk_N, pr_N)$. It submits id_N to IBS to generate second secret key.
   [Here (pk_N, pr_N) denote public key and private key of the nominee].

   (b) IBS invokes $\mathsf{CLE.Extract}(mpk, msk, id_N) \to sk'_N$ and returns sk'_N to N. [Here sk'_N denotes the nominee's secret key].

   (c) Nominee N shares (pk_N, id_N) with U.

3. **[Generation of key parameter by CAP]** Execute the following:

   (a) CAP invokes $\mathsf{CLE.KeyGen}(mpk, id_{\text{CAP}}) \to (pk_{\text{CAP}}, pr_{\text{CAP}})$. It submits id_{CAP} to IBS to generate second secret key.
   [Here $(pk_{\text{CAP}}, pr_{\text{CAP}})$ denote public key and private key of CAP].

   (b) IBS invokes $\mathsf{CLE.Extract}(mpk, msk, id_{\text{CAP}}) \to sk'_{\text{CAP}}$ and returns it to CAP. [Here sk'_{CAP} denotes CAP's secret key].

   (c) CAP shares $(pk_{\text{CAP}}, id_{\text{CAP}})$ with U.

4. U invokes $\mathsf{CLE.Enc}(mpk, id_{\text{CAP}}, pk_{\text{CAP}}, data_2) \to c_1$. [Here $data_2 = (k_1 || id_U)$].

Stage 1: Registration at IBS

5. U sends $(id_U, \widetilde{id}_U)$ to IBS for storage.

Stage 2: Generation of Nominee Registration Certificate for Each Organization

6. User U sends the triplet $(acc_U, \widetilde{id}'_U, nlist_O)$ to the organization O and request to certify the asset.

   [Here, $nlist_O = \{(N_1, rp_1^{(U)}, 1), (N_2, rp_2^{(U)}, 2), ..., (N_l, rp_l^{(U)}, l)\}$, $\widetilde{id}'_U \leftarrow \mathsf{PseudoID}(id_U, nonce_O)$, $nonce_O \leftarrow \mathsf{KeyGen}(\lambda)$, $nlist_O$ denotes the nominee list, $\widetilde{id}'_U$ denotes the pseudo-identity used with O, and $nonce_O$ denotes the commitment value].

7. Organization o invokes $\Sigma.Sig(sk_0, M_O^U) \to \sigma_O^{(U)}$ and sends $(M_O^U, \sigma_O^{(U)})$ to U.

   [Here $\sigma_O^{(U)}$ denotes signature on message M_O^U and the message $M_O^{(U)} = \langle(acc_U^O, \widetilde{id}'_U, O, nlist_O, other\ details)\rangle$].

8. U generates $N_{cert} = (M_O^U, \sigma_O^{(U)}, nonce_O)$.

Figure 5-2: Algorithmic description of Asset Management protocol Π_1. (a)

Stage 3: Storing Asset Certificate at NDP

9. U executes the following:
   (a) Invokes $\mathsf{SKS.Enc}(N_{cert}, k^O) \rightarrow A_{cert}$ [Here A_{cert} denotes the *asset certificate*].
   (b) For all $i \in [l]$, encrypt the second partial key by invoking $\mathsf{CLE.Enc}(mpk, pk_{N_i}, id_{N_i}, k_2^O) \rightarrow c^{N_i}$.
   (c) Compute $r^N = (c^{N_1}, c^{N_2}, ..., c^{N_l})$ and $N_{receipt} = (r^N, nlist, O, other\ details)$
   (d) Invokes $\mathsf{SKS.Enc}(N_{receipt}, k_1) \rightarrow A_{info}$. [Here A_{info} denotes the *asset information*].
   (e) Register *asset inheritance data* (AID) at NDP portal as, $AID = (\widetilde{id}_U, A_{cert}, A_{info}, c_1)$.

Stage 4: Updation/deletion of Nominee Details at Each Organization

10. U invokes the following:
    (a) Update nominee list $nlist'_O$ by adding, deleting or changing the preference level of nominee.
    (b) Send the triplet $(acc_U^O, \widetilde{id}'_U, nlist'_O)$ to O and request to certify the asset. [Note that $\widetilde{id}'_U$ is same as generated previously in step 5.]
    (c) Execute steps 6-9 with modified values.

Figure 5-3: Algorithmic description of Asset Management protocol Π_1 (b).

terms & condition of transfer.

We want to point out that for asset categories type 1 and 4, the user is not identifiable at the organization. Thus, we have stored a pseudo-identity $\widetilde{id}'_U$ at the organization, and its pre-image id_U and $nonce_O$ are stored at IBS and NDP, respectively. The organization can correctly identify the user by providing these two values by a nominee. Note that these values will only be revealed to the nominee after the user's demise. Also, $nonce_O$ is defined uniquely for each organization; thus, a malicious nominee cannot use the nonce defined for one organization to retrieve assets stored in another organization.

Stage 3 - Storing *asset inheritance data* AID at NDP: The purpose of this stage is to store the *asset inheritance data*(AID) at NDP so that a nominee can later obtain it after the user's demise.

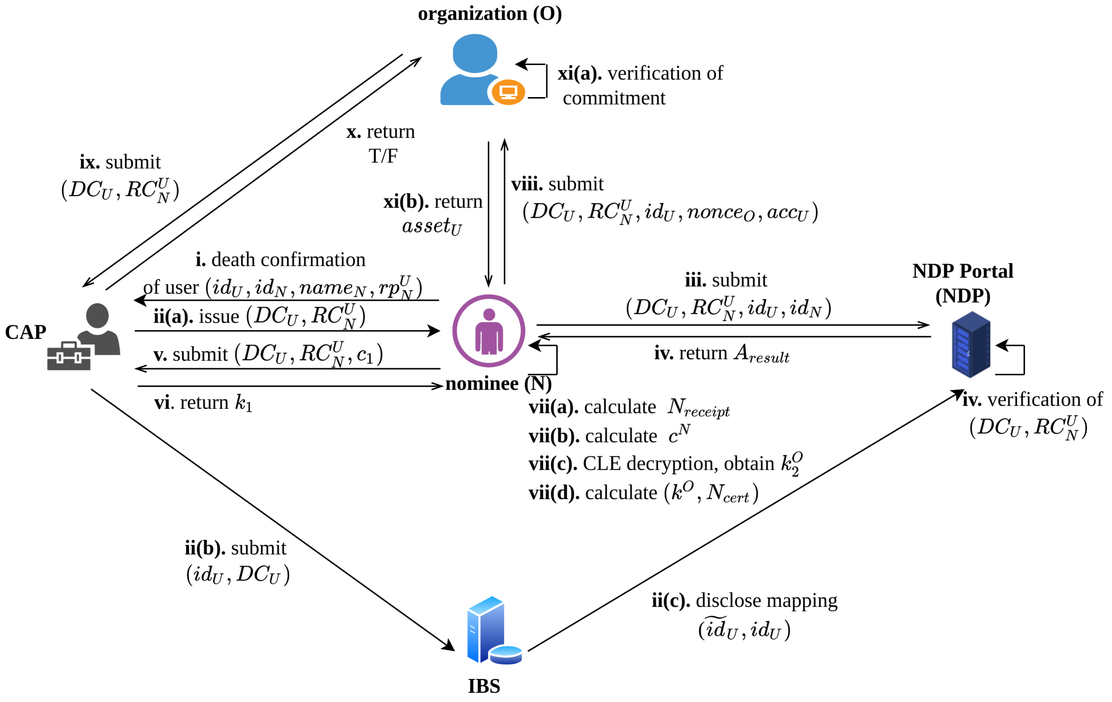
In this stage, the user creates an *asset inheritance certificate* A_{cert} that encrypts the *nominee certificate* N_{cert} using the (symmetric) *asset key* k^O. Then, he encrypts the second partial key k_2^O using the public parameter, denoted (pk_{N_i}, id_{N_i}), of each nominee in the nominee list. Next, he generates *nominee receipt* $N_{receipt}$. The information stored in $N_{receipt}$ allows any nominee to know whether the user has assigned them as the nominee for the asset. This will make the nominee aware of the asset(s) and whether he has to apply for the claim after the user's demise. Note that such disclosure will not breach the asset and user's privacy since the partial key k_1 will not be revealed to the nominee by CAP before the user's death.

Finally, user stores asset inheritance data $AID = (\widetilde{id}_U, A_{cert}, A_{info}, c_1)$ at the NDP. The AID ensures that the descendent can use this data to retrieve the asset from the organization after his demise.

Stage 4 - Updation/Deletion of nominee details at each organization: This stage allows a user to update the nominee details at the organization.

In this phase, a user updates (add/delete a nominee or modify the preference level of a nominee) the nominee list, denoted $nlist'_O$ and sends the tuple $(acc_U^O, \widetilde{id}'_U, nlist'_O)$ to the organization for updating his record. As described in Stage 2, the protocol proceeds with these updated values.

5.3.2 Asset inheritance protocol



organization (O)
xi(a). verification of commitment
x. return T/F
ix. submit (DC_U, RC_N^U)
xi(b). return $asset_U$
viii. submit $(DC_U, RC_N^U, id_U, nonce_O, acc_U)$
i. death confirmation of user $(id_U, id_N, name_N, rp_N^U)$
ii(a). issue (DC_U, RC_N^U)
v. submit (DC_U, RC_N^U, c_1)
vi. return k_1
CAP
nominee (N)
iii. submit $(DC_U, RC_N^U, id_U, id_N)$
iv. return A_{result}
NDP Portal (NDP)
iv. verification of (DC_U, RC_N^U)
vii(a). calculate $N_{receipt}$
vii(b). calculate c^N
vii(c). CLE decryption, obtain k_2^O
vii(d). calculate (k^O, N_{cert})
ii(b). submit (id_U, DC_U)
ii(c). disclose mapping $(\widetilde{id}_U, id_U)$
IBS

Figure 5-4: Pictorial description of Asset Inheritance protocol Π_2.

The *Asset Inheritance* protocol, denoted Π_2, allows a nominee to inherit the asset after the user's death. To design the protocol, we define the following functions:

- $\mathsf{DCert}(id_U) \rightarrow DC_U$: On input id_U, CAP performs multi-level verification process. If ver-

$\Pi_2[N, O, \text{IBS}, \text{CAP}, \text{NDP}]$

Stage A: Death Confirmation by CAP

i. Nominee N sends $(id_U, id_N, name_N, id_N, rp_N^U)$ to CAP to issue death certificate (DC_U) and relationship certificate (RC_N^U).

ii. **if** (DC_U does not exist) **then**
   a. CAP invokes $\mathsf{DCert}(id_U) \rightarrow DC_U$ and $\mathsf{RCert}(id_U, name_N, id_N, rp_N^U) \rightarrow RC_N^U$ and returns (DC_U, RC_N^U) to the nominee.
   b. CAP updates death of (id_U, DC_U) to the IBS.
   c. Upon receiving (id_U, DC_U) from CAP, IBS discloses the mapping of $\widetilde{id}_U \rightarrow id_U$ to the NDP.
else
   CAP invokes $\mathsf{RCert}(id_U, name_N, id_n, rp_N^U) \rightarrow RC_N^U$ and returns (DC_U, RC_N^U) to the nominee.

Stage B : Search Query by Nominee

iii. After the death of the user, N sends $(id_U, id_N, DC_U, RC_N^U)$ to NDP to know if there are any assets against id_U for inheritance purpose.

iv. NDP verifies (DC_U, RC_N^U) from CAP; if verification is successful then it returns $A_{result} \leftarrow (id_U, A_{cert}, A_{info}, c_1)$ to N.

Stage C: Disclosure of AID to Nominee

v. N submits (DC_U, RC_N^U, c_1) to CAP and requests to decrypt the ciphertext c_1.

vi. CAP invokes $\mathsf{CLE.Dec}(mpk, pr_{\text{CAP}}, s_{\text{CAP}}, c_1) \rightarrow data_2'$; parse $data_2' = (a, b)$; verify whether $b \in DC_U$ and returns k_1 to N.

vii. N executes the following:

   (a) Invoke $\mathsf{SKS.Dec}(A_{info}, k_1) \rightarrow N_{receipt}$.

   (b) check his name and preference level from *nlist*; if it is successful then obtain c^N.

   (c) Invoke $\mathsf{CLE.Dec}(mpk, pr_N, s_N, c^N) \rightarrow k_2^O$.

   (d) Compute *asset key* by invoking $PRF(k_1, k_2^O) \rightarrow k^O$; and then compute *nominee asset certificate* by invoking $\mathsf{AES.Dec}(A_{cert}, k^O) \rightarrow N_{cert}$.

Figure 5-5: Algorithmic description of *Asset Inheritance* protocol Π_2.(a)

Stage D: Asset Inheritance

viii. N submits parameters $(DC_U, RC_N^U, id_U, nonce, acc_U^O)$ to the organization O.

ix. O verifies whether $id_U \in DC_U$; if verification is successful then submits verification request of (DC_U, RC_N^U) to CAP.

x. CAP verifies (DC_U, RC_N^U) and returns True or False.

xi. O executes the following:

(a) Validate the pre-image of commitment value, $\mathsf{H}(id_U \| nonce_O) == \widetilde{id}'_U$.

(b) If verification is successful then O transfers the asset $asset_U^O$ to the nominee N.

Figure 5-6: Algorithmic description of *Asset Inheritance* protocol Π_2. (b)

ification is successful, it creates a document m_U; compute signature $\Sigma.Sig(sk_{\text{CAP}}, m_U) \rightarrow \sigma_{dc}$; and returns *death certificate* $DC_U = (m_U||\sigma_{dc})$. Here, $m_U = (serial\ no., name, id_U, demise\ date, issue\ date, other\ parameters$ (if any)$)$.

- $\mathsf{RCert}(id_U, name_n, id_N, rp_N^U) \rightarrow RC_N^U$: On input (id_U, id_N), CAP performs multi-level verification process. If verification is successful, it creates a document m_N; generates signature $\Sigma.Sig \rightarrow \sigma_{rc}$; and returns *relationship certificate* $RC_N^U = (m_N||\sigma_{rc})$. Here, $m_N = (serial\ no., name, id_U, id_N, rp_N^U, issue\ date, other\ parameters$ (if any)].

From the design standpoint, the protocol Π_2 consists of 4 stages, all executed by parties N, O, IBS, CAP, and NDP. The nominee communicates with entities, NDP, CAP, and O, to know about the AID and retrieve the $asset_U$. The pictorial and algorithmic descriptions of Π_2 are given in Figs. 5-4 and 5-5, 5-6 respectively.

Stage A - Death Confirmation by CAP: The purpose of this stage is to generate *death and relationship certificates* by CAP that confirm the death of a user and the relationship between the nominee and the user.

In India, death confirmation is verified by multiple authorities before issuing a *death certificate* for a user. Because of such multiple confirmations, the chances of false reporting become negligible. Therefore, we assume that the *death certificate* is *correctly* issued by CAP.

Also, CAP generates *relationship certificate* based on the legal terms and conditions. We leave this at the implementation level and do not go into details of it. The relationship certificate also has the following advantages in our protocol: (i) It ensures that only the nominee will know about the asset, but an *eligible* nominee can only inherit it, and (ii) It prevents the user from designating a malicious person as a nominee. For example, suppose a user designates an unknown person as the nominee. In that case, the nominee will not get the asset, as CAP will not issue a relationship certificate to them.

Stage B - Search Query by Nominee: This stage allows any nominee to query the NDP to check whether any asset has been assigned against him.

NDP is a single dedicated entity that stores the user's asset-related information. It removes the problem arising from the nominee's unawareness of assets.

Stage C - Disclosure of AID to Nominee: The purpose of this stage is to disclose the asset information data AID of the user, which is stored at NDP, to the nominee.

In this stage, the nominee decrypts the ciphertexts, $(A_{cert},\ A_{info}, c_1)$, which he has retrieved from NDP in stage B. Any nominee can obtain the first partial key by submitting c_1 to CAP. Now, the nominee decrypts A_{info} using k_1. The decrypted values disclose the nominee list, the preference level, and any other information assigned to the nominee. Note that decrypting A_{info} will only reveal that the user has some asset whose information is stored at NDP; however, an eligible nominee can only claim the asset. Thus, it eradicates the problem of unawareness of the user's asset by the nominee. Finally, the eligible nominee N re-compute the asset key k^O using k_1 and k_2^O.

Stage D - Asset Inheritance: This stage aims to *inherit* the user's asset to the nominee.

In this stage, the nominee claims $asset_U$ from the organization O by submitting $(DC_U, RC_N^U, id_U, acc_U^O,\ nonce_O)$ along with his identity to the organization for asset inheritance. After verification, the organization transfers all the assets to the nominee. Before transferring the asset, the organization verifies the nominee's preference level from $nlist_O$ to ensure the order in which the nominee claims the asset. An alternative nominee can claim the asset only if the primary nominee has also died. In such a case, the alternative nominee has to present all the nominees' death certificates whose preference levels are higher than his.

5.4 Proof of Security of DAIP

We formally define the security properties using *universal composability (UC) framework* [48]. In this framework, the security of a protocol Π is analyzed by comparing the "real world" execution with the execution of the protocol in the "ideal world." In the "real world", the parties execute a real protocol among a set of parties, honest and dishonest. Here, the dishonest parties are under the control of a "special party" called an *adversary* $\mathcal{A}$, who controls their actions and the internal state. The "real world" protocol is said to be *secure* if it "closely" mimics the "ideal world." By "close," we mean that the real world view of the adversary is *indistinguishable* from its ideal world view.

We define the *view* of a party P, denoted view_P, that consists of its input, the value it received during execution, and its internal state. Further, the output of party P, denoted out_P, is a function of its *view*.

In the "ideal world", a trusted third party (TTP) is connected to all the parties – honest and dishonest – using a secure and authenticated channel, and no other communication channel exists among the parties. On behalf of all the dishonest parties, the adversary interacts with the TTP via a *simulator* S who simulates the execution. Informally, if the adversary cannot distinguish whether it is interacting in the ideal or real world, then the protocol is said to be *UC-secure*.

We now analyze the DAIP protocol $\Pi = \Pi_2 \circ \Pi_1$ for various security properties, as described in Sect. 5.1.4, under UC model. Here, U, N, O, IBS, CAP, NDP, and $\mathcal{A}$ denote the user, nominee, organization, identity-based system, certification authority portal, nominee display portal and adversary, respectively. In addition, there are three more entities: the simulator $\mathcal{S}$, the trusted third party $\mathcal{F}$ which is also the *ideal functionality*, and a set of honest parties $\mathcal{H}$. The input to $\mathcal{H}$ is the same as the input of the honest parties in the *real* world; the input of $\mathcal{A}$ consists of the input of the party it is corrupting, combined with the auxiliary input z; the input of $\mathcal{S}$ is the same as the input of $\mathcal{A}$; internal random tape of $\mathcal{A}$, if any, is accessible to the simulator $\mathcal{S}$.

5.4.1 Asset privacy

The *asset privacy* of DAI protocol Π_1, as defined in section 5.1.4, guarantees that no third party, including NDP, can reveal anything from the data stored at the NDP portal. It can be formalized as follows using the simulation of *ideal* and *real* worlds.

For DAIP Π_1, the ideal world experiment, denoted as $\mathsf{IDEAL}^{ASSET\text{-}PRV}(1^\lambda, z, acc_U^O,\ id_U, id_{CAP}, id_N, asset_U^O)$, is described in Fig. 5-7. The real world experiment, denoted as $\Pi_1(1^\lambda, z, acc_U^O, id_U, id_{CAP}, id_N, asset_U^O)$, is described in Fig. 5-1. The views of $\mathcal{A}(NDP)$ are defined as $\mathsf{view}^{\mathsf{real}}_{\mathcal{A}(NDP)}$ and $\mathsf{view}^{\mathsf{ideal}}_{\mathcal{A}(NDP)}$ in these two worlds.

Here the input to U is (acc_U^O, id_U); the input to CAP is (id_{CAP}); the input to IBS is (id_U, id_{CAP}, id_N); the input to O is $(acc_U^O, asset_U^O)$; the input to nominee N is (id_N); the input to $\mathcal{A}$ (NDP) is

$$\mathsf{IDEAL}^{ASSET\text{-}PRV}[1^\lambda, \mathcal{A}(NDP)]$$

Input: $\mathcal{H} = \{U, N, O, CAP, IBS\}$. U has (acc_U^O, id_U); CAP has (id_{CAP}); IBS has (id_U, id_{CAP}, id_N); Organization $(acc_U^O, asset_U^O)$; N has (id_N); $\mathcal{A}$ (NDP) has $(1^\lambda, z)$; $\mathcal{S}$ has $(1^\lambda, z)$.

Output: All parties in $\mathcal{H}$ outputs $out_{\mathcal{H}}$; and $\mathcal{A}(NDP)$ outputs $out^{\mathsf{ideal}}_{\mathcal{A}(NDP)}$.

1. U sends (acc_U^O, id_U) to $\mathcal{F}$; IBS sends (id_U, id_{CAP}, id_N) to $\mathcal{F}$; O sends $(acc_U^O, asset_U^O)$ to $\mathcal{F}$;
2. $\mathcal{S}$ interacts with $\mathcal{A}(NDP)$ and generates $\mathsf{view}^{\mathsf{ideal}}_{\mathcal{A}(NDP)}$ to mimic $\mathsf{view}^{\mathsf{real}}_{\mathcal{A}(NDP)}$.
   (a). $\mathcal{S}$ chooses $\widetilde{id}_U \overset{\$}{\leftarrow} \mathcal{T}_1$, where $\mathcal{T}_1$ is the domain of $\mathsf{PseudoID}(\cdot)$.
   (b). $\mathcal{S}$ chooses $c_1 \overset{\$}{\leftarrow} \mathcal{T}_2$, where $\mathcal{T}_2$ is the domain of $\mathsf{CLE.Enc}(\cdot)$.
   (c). $\mathcal{S}$ chooses $A_{cert} \overset{\$}{\leftarrow} \mathcal{T}_3$ where $\mathcal{T}_3$ is the domain of $\mathsf{SKS.Enc}(\cdot)$.
   (d). $\mathcal{S}$ chooses $A_{info} \overset{\$}{\leftarrow} \mathcal{T}_3$ where $\mathcal{T}_3$ is the domain of $\mathsf{SKS.Enc}(\cdot)$.
   (e). $\mathcal{S}$ sends $(\widetilde{id}_U, A_{cert}, A_{info}, c_1)$ to $\mathcal{A}$(NDP). Therefore, $\mathsf{view}^{\mathsf{ideal}}_{\mathcal{A}(NDP)} = (\widetilde{id}_U, A_{cert}, A_{info}, c_1)$ and $\mathsf{out}^{\mathsf{ideal}}_{\mathcal{A}(NDP)} = (\widetilde{id}_U, A_{cert}, A_{info}, c_1)$.
3. Finally, $\mathcal{F}$ sends $\mathsf{out}_{\mathcal{H}} = \perp$ to all the parties in $\mathcal{H}$.

Figure 5-7: Execution of the ideal world $\mathsf{IDEAL}^{ASSET\text{-}PRV}$ for ASSET-PRV security.

$(1^\lambda, z)$; the input to $\mathcal{S}$ is $(1^\lambda, z)$. Here, z is the information leaked by the adversary $\mathcal{A}$ or additional input actively influencing the execution.

Definition 5.6 (ASSET-PRV SECURITY)**.** *The protocol* Π_1 *– as described in Section 5.3.1 – is said to be* ASSET-PRV *secure, if for every non-uniform PPT adversary* $\mathcal{A}(NDP)$ *in the real world* Π_1*, there exists a non-uniform PPT simulator* $\mathcal{S}$ *in the ideal world* $\mathsf{IDEAL}^{ASSET\text{-}PRV}(1^\lambda, z, acc_U^O, id_U, id_{CAP}, id_N, asset_U^O)$ *such that*

$$\mathsf{view}^{\mathsf{ideal}}_{\mathcal{A}(NDP)} \overset{c}{\equiv} \mathsf{view}^{\mathsf{real}}_{\mathcal{A}(NDP)}$$

for all $\lambda \in \{0,1\}^*$.

Theorem 5.1 (ASSET-PRV SECURITY)**.** *Suppose the hash function* H *is collision-resistant; the certificateless encryption scheme* CLE *is* $\mathsf{IND\text{-}CCA}$ *secure; the function* f *is pseudo-random* PRF*, and the symmetric-key encryption scheme* SKS *is* $\mathsf{IND\text{-}CCA}$ *secure. Then the DAI protocol* Π_1 *(as described in Fig. 5-1) satisfies* $\mathsf{ASSET\text{-}PRV}$ *security (Def. 5.1).*

Proof.

To prove this, we need to show that the execution of protocol Π_1 in the *ideal* world in the presence of non-uniform PPT simulator $\mathcal{S}$ is indistinguishable from the *real* world execution even in the presence of an adversary $\mathcal{A}$. The complete proof consists of the following two cases:

CASE 1 - BOTH NDP AND IBS ARE CONTR OLLED BY $\mathcal{A}$: Given the input of IBS (id_U, id_{CAP}, id_N) and NDP $(1^\lambda, z)$, the view of IBS after step 4 of Π_1 is $(id_U, \widetilde{id}_U)$, the view of NDP after step 8(e) of Π_1 is $(\widetilde{id}_U, A_{cert}, A_{info}, c_1)$. Since both parties are controlled by adversary $\mathcal{A}$, the input, the internal states, and their views are known to $\mathcal{A}$. Thus, the adversary knows the identity of user's who have stored their DAI data at NDP. It is to be noted that the sharing of such mapping does not break the security of the encrypted values, $(A_{cert}, A_{info}, c_1)$, stored at NDP; therefore, the protocol guarantees asset privacy.

Thus, it is sufficient to prove that if the adversary corrupts only NDP and cannot break the IND-CCA security of CLE, then the protocol guarantees asset privacy.

CASE 2 - NDP IS CONTROLLED BY $\mathcal{A}$:

The NDP has stored the following values: $(\widetilde{id}_U, A_{cert}, A_{info}, c_1)$. If NDP can decrypt the ciphertext c_1, then we can design an adversary to break $\mathsf{IND\text{-}CCA}$ security of CLE scheme, which is impossible. Thus, *asset privacy* property is guaranteed.

Formally, we need to show that the views of the adversary in the ideal and real worlds are indistinguishable, i.e., $\mathsf{view}^{\mathsf{ideal}}_{\mathcal{A}(NDP)} \overset{c}{\equiv} \mathsf{view}^{\mathsf{real}}_{\mathcal{A}(NDP)}$. To prove that we have the following hybrids:

H_0 : This is $\Pi_1(1^\lambda, z, acc^O_U, id_U, id_{CAP},\ id_N, acc^O_U, asset^O_U)$ where $\mathsf{view}^{\mathsf{real}}_{\mathcal{A}(NDP)}$ is the view of $\mathcal{A}(NDP)$.

H_1 : Identical to H_0 except that we change step 0 of Π_1: replace $\widetilde{id}_U \leftarrow \mathsf{PseudoID}(\cdot)$ with $\widetilde{id}_U \overset{\$}{\leftarrow} \mathcal{T}_1$, where $\mathcal{T}_1$ is the distribution of $\mathsf{PseudoID}(\cdot)$.

Lemma 5.1. *If hash function H guarantees CRHF property then $\mathsf{view}^{H_1}_{\mathcal{A}(NDP)}$ is computationally indistinguishable from $\mathsf{view}^{H_0}_{\mathcal{A}(NDP)}$.*

Proof. Follows directly from CRHF property of hash function H. □

H_2 : Identical to H_1 except that we change step 3 of Π_1: replace $c1 \leftarrow$ with $c1 \overset{\$}{\leftarrow} \mathcal{T}_2$, where $\mathcal{T}_2$ is the distribution of $\mathsf{CLE.Enc}(\cdot)$

Lemma 5.2. *If CLE is $\mathsf{IND\text{-}CCA}$ secure then $\mathsf{view}^{H_2}_{\mathcal{A}(NDP)}$ is computationally indistinguishable from $\mathsf{view}^{H_1}_{\mathcal{A}(NDP)}$.*

Proof. Follows directly from CLE security. □

H_3 : Identical to H_2 except that we change step 8(a) of Π_1: replace $A_{cert} \leftarrow$ with $A_{cert} \overset{\$}{\leftarrow} \mathcal{T}_3$, where $\mathcal{T}_3$ is the domain of $\mathsf{SKS.Enc}(\cdot)$

Lemma 5.3. *If SKS is $\mathsf{IND\text{-}CCA}$ secure then $\mathsf{view}^{H_3}_{\mathcal{A}(NDP)}$ is computationally indistinguishable from $\mathsf{view}^{H_2}_{\mathcal{A}(NDP)}$.*

Proof. Follows directly from SKS security. □

H_4 : identical to H_3 except that we change step 8(d) of Π_1: replace $A_{info} \leftarrow$ with $A_{info} \overset{\$}{\leftarrow} \mathcal{T}_3$, where where $\mathcal{T}_3$ is the domain of $\mathsf{SKS.Enc}(\cdot)$

Lemma 5.4. *If SKS is $\mathsf{IND\text{-}CCA}$ secure then $\mathsf{view}^{H_4}_{\mathcal{A}(NDP)}$ is computationally indistinguishable from $\mathsf{view}^{H_3}_{\mathcal{A}(NDP)}$.*

Proof. Follows directly from SKS security. □

H_5 : This is $\mathsf{IDEAL}^{ASSET\text{-}PRV}(1^\lambda, z, acc^O_U, id_U, id_{CAP}, id_N, acc^O_U, asset^O_U)$ (as defined in Def. 5.1).

Lemma 5.5. *If* SKS *is* $\mathsf{IND\text{-}CCA}$ *secure then* $\mathsf{view}^{H_5}_{\mathcal{A}(NDP)}$ *is computationally indistinguishable from* $\mathsf{view}^{H_4}_{\mathcal{A}(NDP)}$.

Proof. The proof follows from the design of the experiments that the interaction between $\mathcal{S}$ and $\mathcal{A}(NDP)$ in $\mathsf{IDEAL}^{ASSET\text{-}PRV}(1^\lambda, z, acc^O_U, id_U, id_{CAP}, id_N, acc^O_U, asset^O_U,)$ is same as the interaction between U and $\mathcal{A}(NDP)$ in Π_1, since SKS is $\mathsf{IND\text{-}CCA}$ secure. □

Combining Lemmas 6.1-6.5, the theorem is proved. □

5.4.2 Non-repudiation

Non-repudiation ensures that during the execution of protocol Π_2, an organization cannot deny the asset's holding when the nominee requests it (see Section 5.1.4). To prove this, consider a nominee having input id_N. During the execution of the protocol Π_2, the nominee sends $(DC_U, RC^U_n, id_U, nonce, acc_U)$ to organization for claiming the asset. If the organization is honest, he will transfer the asset to the nominee after performing the necessary verification, as shown in the steps ix-xi. If the organization is malicious, then there are two cases: if the organization denies the existence of the user's account, then in this case nominee can provide $(M^U_O, \sigma^{(U)}_O)$ which ensures that user's account does exist in that organization; if organization denies transferring the asset then, in this case, the parties need to resolve the dispute with the help of trusted legal third party.

5.4.3 Identifiability

The protocol Π is said to achieve *identifiability* if it fulfills the following conditions (as defined in Section 5.1.4): i) the user's identity is not known to the organization before death; and ii) After death, an organization knows the identity of the user. Note that the protocol Π_1 ensures the first condition while Π_2 ensures the second condition. The user U stores $\widetilde{id}'_U = H(id_U||nonce_O)$ to the organization O during the execution of protocol Π_1 in step 2. To prove the first property, we consider the following cases:

CASE 1: IF ORGANISATION IS CORRUPTED BY ADVERSARY $\mathcal{A}$: In this case, $\mathcal{A}$ will try to find out the pre-image $(id^*_U, nonce^*)$ such that $\widetilde{id}'_U == H(id^*_U||nonce^*)$. Since H is secure by the Hash function's pre-image property, the user's identity can not be known to the organization. Thus, the protocol Π_1 guarantees the first property identifiability.

CASE 2: BOTH ORGANIZATION AND IBS ARE CORRUPTED BY $\mathcal{A}$: We know that IBS has the complete list of identity of users. If $\mathcal{A}$ corrupts

both organization and IBS, then he knows the mapping $(id_U, \widetilde{id}_U)$. However, $\widetilde{id}'_U$ is computed using $(id_U, nonce_O)$; thus, due to the pre-image security of the hash function, $\mathcal{A}$ will not be able to know the mapping between id_U and $\widetilde{id}'_U$. This guarantees the first property of identifiability in protocol Π_1.
Nominee submits $(DC_U, RC^U_N, id_U, nonce_O, acc_U)$ to the organization to retrieve the asset during the execution of protocol Π_2. The organization verifies the correctness of the commitment value as $\mathsf{H}(id_U \| nonce_O) == \widetilde{id}'_U$, which reveals the identity of the user to the organization. Thus, it ensures the second identifiability property in protocol Π_2.

5.4.4 Key inheritance privacy

The asset key inheritance privacy guarantees that the key is revealed to the designated nominee only in the protocol. During the execution of protocol Π_2, nominee sends $(id_U, id_N, name_N, id_N, rp^U_N)$ to obtain (DC_U, RC^U_N). Using this, the nominee can get the first partial key k^O from CAP. He can recover the list of nominees $nlist$ from A_{info} designated against the asset A_{cert}. If n does not belong to the $nlist$, then he can not decrypt c^{N_i} because of the CLE security definition. Thus, protocol Π_2 guarantees the key inheritance privacy property.

5.4.5 User's privacy

The DAI protocol Π guarantees the user's privacy if the data stored at the NDP portal does not reveal the user's identity. The user U stores asset inheritance data using $\widetilde{id}_U$. If both NDP and IBS do not collude, then the protocol Π achieves the user's privacy due to the pre-image security of the hash function, $\mathcal{A}$ will not be able to know the mapping. However, if adversary $\mathcal{A}$ corrupts and controls both, then their input, internal state, and views will be known to $\mathcal{A}$. Since $\mathcal{A}$ knows the identity of users' who have stored DAI data at NDP, user privacy is not guaranteed.

5.4.6 Correctness

The protocol Π guarantees the correctness properties if both user and nominee can be identified and verified correctly by the organization before the asset transfer (as defined in Section 5.1.4). We can prove the correctness of both properties using the following cases:

CORRECTNESS 1- USER'S IDENTIFICATION: The organization can verify the user correctly before asset transfer. During the execution of the protocol Π_2, nominee submits $(DC_U, RC^U_N,$

$id_U, nonce_O,\ acc_U)$. The organization can calculate commitment value as $\mathsf{H}(id_U \| nonce_O) == \widetilde{id}'_U$, which ensures the correctness of the user's identity. An adversary $\mathcal{A}$ will not be able to generate a value id^*_u such that $\widetilde{id}'_U == H(id^*_U || nonce_O)$ due to the security of hash function.

CORRECTNESS 2- NOMINEE VERIFICATION: The nominee n obtains $(id_U, A_{cert}, A_{info}, c_1)$ from step iv in the execution of protocol Π_2. Using this, only the designated nominee can retireve the value of $nonce_O$. Nominee send $(DC_U, RC^U_N, id_U, nonce_O,\ acc_U)$ to the organization. The organization can verify $(DC_U, RC^U_N$ from the CAP to ensure the correctness. In the case of adversarial nominee $\mathcal{A}$, he can obtain the value $(DC_U, id_U, nonce_O, acc_U)$ by colluding with another nominee. However, $\mathcal{A}$ will not be able to generate RC^U_N, so it would not be able to claim for the asset.

5.4.7 Robustness

A DAI protocol Π is robust if the nominee can retrieve the asset or asset information without loss. The execution of protocol Π_1 and Π_2 clearly shows that the nominee can obtain data from IBS, CAP, and NDP and know DAI. He can communicate with the organization to retrieve the asset successfully. However, asset retrieval will not work if any entity, such as IBS or NDP, refuses to hold data.

5.5 Comparison Between Various Asset Inheritance Models

In this section, we compare our DAIP with the four existing *digital asset management* models: DigiPulse [166], PassOn [168], SafeHaven [169], and TrustVerse [170]. All these models are proposed as white papers without any detailed technical construction, well-defined definitions, descriptions of properties, or security proofs. Therefore, we have mentioned only those properties specified. Otherwise, we have specified it as "unknown" for this comparison. We have compared the models in terms of design and security properties.

5.5.1 Design comparison

We now compare the existing protocols of [166, 168, 169, 170] with our DAIP in terms of design. We have summarized the comparison in Table 5.1.

Currently, all the existing models are more focused on type 3 category assets and do not discuss handling other categories of assets whose inheritance is equally essential (see Sect. 5.1.1 for details on the type of assets). The DAIP supports the inheritance of all four categories of assets. The asset inheritance model should require *minimum participation of nominee*. However, all four models require the high involvement of nominees during the execution of the protocol (see Sect. 2.3 for details). The DAIP requires nominee participation only at the time of key generation. The protocol will not fail even if any nominee dies or denies the protocol's execution. In the worst case, if only one nominee survives, he will receive the assets.

Table 5.1: Design Properties Comparison

Properties	PassOn [168]	DigiPulse [166]	SafeHaven [169]	TrustVerse [170]	**DAIP**
Category of asset	Type 3	Type 3	Type 3	Type 3	**Type 1, 2, 3, 4**
Participation of nominee	Full	Full	Full	Full	**Partial**
Death confirmation methods	Inactivity period, Weighted voting, Notary	Inactivity period	Notary	All Nominee Consent	**Death certificate by CAP**
Storage Mechanism	Blockchain	Blockchain	Blockchain	Blockchain	**NDP**
Verification of nominee	No	No	No	No	**Yes**

The successful implementation of the asset inheritance model relies on the *correct confirmation of death*. Each model has different approaches to confirm and prove the death, such as "inactivity period, voting, weighted voting, or through a notary.". All these methods have inherent limitations (see Sect. 5.1.5 for details). Our model uses a robust method of death confirmation – by a legal entity (internally verified and approved by multiple entities) – that maintains the record of birth and death within a territory. Therefore, it will neither deny the issuance of a death certificate, nor it will confirm an incorrect death. The property 4 compares the storage mechanism. As discussed earlier in Section 5.1.2, asset storage mechanisms can be realized using decentralized architecture (such as blockchain) or a centralized system. All four models use blockchain as a storage mechanism to store assets or asset information, while the DAIP model uses a centralized entity, NDP. Blockchain-based systems are generally designed to enable transparency and availability but are complex to construct and manage. The goal of DAI is to store asset information securely and deliver it correctly to the right nominee. If a non-blockchain model can have such functionality, it may provide ease of implementation. The DAIP model emphasizes this and affirms that a non-blockchain IBS-based efficient system can be designed and integrated with the existing infrastructure for asset inheritance.

The nominee is a legal custodian of the asset. The asset inheritance protocol should allow asset transfer to the *legal nominee only*. All the above four models authorize anyone to become nominee and claim the asset. This kind of declaration may fail for type 1 and type 4 categories of assets because the organization can deny the transfer of the asset to a nominee who is not legally bound to inherit it. Our model uses the concept of *death certificate* and *relationship certificate*, which bounds a user to declare only a legal person as a nominee to ensure the correct delivery of the assets.

Table 5.2: Security Properties Comparison

Security Goals	**PassOn** [168]	**DigiPulse** [166]	**SafeHaven** [169]	**TrustVerse** [170]	**DAIP**
Asset Privacy	No	Yes	No	No	**Yes**
Identifiability	No	No	No	No	**Yes**
Key inheritance privacy	No	No	No	No	**Yes**
Non-repudiation	Not applicable	Not applicable	Not applicable	Not applicable	**Yes**
User Privacy	No	No	No	No	**Yes**
Robustness	Unknown	No	No	No	**Yes**

5.5.2 Security properties comparison

We now compare the existing protocols of [166, 168, 169, 170] with our DAIP in terms of security properties. We have summarized the comparison in Table 5.2.

We found that our DAIP, as well as [166], achieves the *asset privacy*. Our protocol achieves it by storing the asset information at NDP in encrypted form. All four existing models handle only the type 3 category of assets, thus, identifiability is not an issue. Since DAIP is applicable for all four categories of assets, it solves the *identifiability* issue for type 1 and type 4 categories of assets, enabling the pseudonym users to transfer their assets correctly. We critically analyze the *key inheritance privacy* property in [166, 169, 170, 168] but did not find enough details to support their claim; thus, we assume that these protocols do not guarantee this property. Our protocol achieves this property using CLE scheme that hides the secret key from the nominee. Thus, only designated nominee can retrieve it. The *non-repudiation* property does not have any significance for type 3 category assets. Hence, it is not applicable to all the existing models. Our DAIP ensures the non-repudiation property. Unlike all the existing models, our protocol ensures *user's privacy* under the assumption that IBS and NDP do not collude and reveal the pseudo-identity, $\widetilde{id}'_U$. All the existing models do not guarantee *robustness* property due to incorrect death confirmation methods or failure to recover the key. Our protocol achieves robustness properties under a semi-honest setting.

5.6 Application and Implementation of DAIP

5.6.1 DAIP Application

The importance of personal data has increased after the recent bills such as GDPR [74], which impels data fiduciary from transferring significant user's assets to descendants posthumously. This necessity enables stakeholders such as social media platforms, cloud service providers, software developers, system designers, and other intermediaries to design and implement an asset inheritance model. Until now, the stakeholders could not comply with the implementation due to its design challenges (as described in Sect. 2). In this work, we have analyzed the challenges thoroughly, and an efficient model DAIP is proposed. The model stores distinct attributes of users securely, reveals the user's identity correctly, and transfers assets to the right nominee. For stakeholders, implementing the DAIP model would enable the transfer of inheritable data to the correct nominee and allow the deactivation and closing of the user's account after demise. It directs the service provider on how the protocol can be implemented as a concrete design with the coordination of unique IBS and CAP. It also works as a guideline for authorities such as IBS to enhance their implementation as per the protocol to empower digital asset inheritance.

The DAIP can be easily integrated with the existing systems. We discuss here how the entities of DAIP can be realized and mapped with the current system, along with some expected challenges while incorporating the protocol Π.

For instance, Aadhar, a unique biometric-based authentication system of users of India, can be used as IBS for authentication and identity storage [2]. However, a minor modification is required in the existing system. Firstly, it has to provide a service to the users so that they can store pseudo-identity against their actual identity. This service can be integrated easily into the existing system. Secondly, it has to provide additional functionality for generating/storing the master key and the user's partial private keys. The challenging part is the storage of other attributes for large-scale users, as it requires a regulatory effort.

Next, in India, every state has a death certificate issuing portal that is responsible for confirming the demise, issuance, and verification of the death certificate. The same entity can be realized as

CAP by integrating additional attributes as discussed in the protocol Π_1 and Π_2. Since every state uses independent architecture to issue certificates, integration of a common other attribute among them will require effort. For instance, one such attribute is the incorporation of decryption modules. Similarly, the entity must incorporate appropriate authorization and access control to verify the nominee and transfer the decrypted data.

DigiLocker [165] facilitates Indian citizens to store personal documents through an account associated with Aadhar. NDP services can be conveniently realized using DigiLocker with some modifications. DigiLocker may implement the storage and management functionality of asset or asset information. It will enable users to link and store the inheritable data with their accounts directly. DigiLocker account also has to introduce the ability to include nominees. This would make it convenient for nominees to coordinate with Digilocker (as NDP) to request and download the inheritable data and keys.

Further, every organization will also need to implement some cryptographic primitives. Generally, organizations are equipped with digital certificates. Therefore, they can conveniently implement the signature schemes and store nominee declaration information as per the protocol Π_1 and Π_2.

Finally, a communication mechanism with IBS and CAP can be established to verify users' identity and death confirmation, respectively. Integrating the DAIP model with existing infrastructure can work as a single umbrella to store and convey the asset posthumously.

5.6.2 Implementation

We now describe the implementation results of our DAI protocol Π (as described in Sect. 5.3). We have analyzed the protocol regarding storage overhead and the computation cost for various entities involved in executing Π (see Sect. 5.1.2). First, we describe the experimental setup for the simulation.

Experimental Setup. The experiments were performed on a machine running 64 bits GNU Linux kernel version 5.8.0-43, Ubuntu 20.04.2 LTS, with an Intel® $Core^{TM}$ i5-8250U CPU @ 1.60GHz × 8 and 12GB memory. The prototype simulation is implemented. in C/C++ GCC compiler version 9.3.0-17 and Python 3.8.10. We have used the PBC Library version PBC-0.5.14 for the implementation of pairing functions in certificateless encryption [18], and PyCryptodome-3.12 for the implementation of other low-level cryptographic primitives [24].

We have initialized the pairing parameters for CLE encryption and decryption based on Type

Table 5.3: Performance analysis: storage and computation overhead for type 1 and type 4 category assets (No. of nominee: $l = 10$).

Entity	Components used	Storage overhead	Computation cost (in ms)
IBS	IBS system + CLE+ KeyGen	32 bytes/user+384 bytes	10.516
NDP	Any storage system	3708 bytes/user	None
Organization	Signature Scheme Σ + Hash	1032 bytes/user	64.678
CAP	Signature Scheme Σ+ DCert+ RCert + KeyGen+ CLE	2486 bytes (an user + l nominee)	103.070
User	KeyGen+ PseudoID + SKS+ CLE	448 bytes	41.056
Nominee	CLE +KeyGen	448 bytes	16.672

1 symmetric pairing $e : G_1 \times G_1 \rightarrow G_2$. The PBC library defines the hash function H_1, which maps an arbitrary string to a group element G_1. The hash function H_2 is constructed based on the SHA256 algorithm. This hash function can be easily replaced with any other cryptographic hash function based on the requirements. The message (M) size for CLE encryption is 32 bytes (256 bits) to enable elementary XOR operations. In every stage of the protocol Π, cryptographic operations were called, which are simulated using the following parameters: The function KeyGen() is realized using the function Crypto.Random, and the functions PseudoID() and PRF() are realized using SHA256. The number of nominees selected for asset inheritance creation varies from $2 \leq l \leq 10$. We have used AES 256-bit encryption in CFB mode for SKS.Enc(), SKS.DEC(), and RSA 1024-bit signature algorithm for implementing signature schemes Σ.KeyGen(), Σ.Sig(), Σ.Verify().

Next, based on the experimental setup, the approximate size for different parameters used in Π is defined as follows:

- PseudoID(), KeyGen(), Identity (id_u, id_n), Serial no., Account no., Organization name, Name, rp_n^o: 32 bytes
- CLE.Enc() (U,V): (128, 32) bytes, pk:(X,Y)=(128, 128) bytes, s_k, s'_k:128 bytes
- Sign(): 128 bytes
- Demise date, Issue date: 8 bytes
- Preference Level: 4 bytes

Table 5.4: Size of parameters used for AID creation (in bytes)

No. of Nominee	2	4	6	8	10
N_{CERT} **Data Size**	702	885	1074	1270	1467
$N_{Receipt}$ **Data Size**	336	643	1240	1555	1689
$data_2$ **size**	64	64	64	64	64

Computation overhead. We evaluate the computation overhead of execution of protocol Π. We have implemented each computational stage. All the intermediate step outputs are stored in a buffer and processed in the next step as per the protocol. Firstly, we have computed the execution time for each entity. Every entity takes input data, executes its algorithm(s), and computes the output. The execution time is measured and summed up to calculate the final computation time for each entity. For instance, the computation time of user U is the total time to compute KeyGen(), Key.Extract(), PseudoID(), SKS.Enc(), and CLE.Enc() operation is performed in stage 0 to stage 3 of protocol Π_1. The results are shown in Table 5.3. The results show that each entity must exercise minimal computation to manage and process the data during asset inheritance modeling.

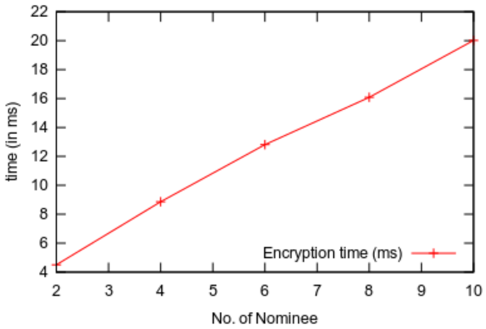


Figure 5-8: Run time of certificateless encryption (CLE) for input data size of 32 bytes vs. no. of nominees.

Secondly, we have computed the execution time for certificateless encryption (CLE) for a fixed input data (the second partial key) of 32 bytes by varying the number of nominees from 2 to 10, refer Fig. 5-8. The simulation result affirms that the encryption time grows with the number of nominees. However, the encryption computation time is minimal and can be efficiently integrated with the existing infrastructure.

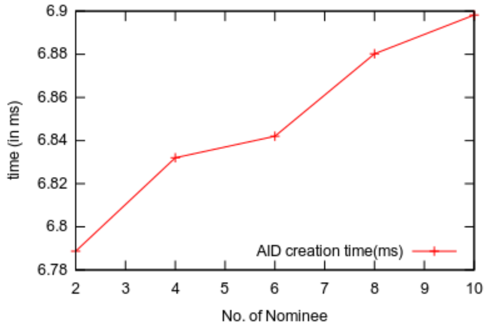


Figure 5-9: Run time of creating asset inheritance data AID by a user vs. the number of nominees.

Finally, we have analyzed a user's computation time for creating asset inheritance data (AID). The AID data consists of $\{A_{cert}$,A_{info}, c_1 $\}$ and uses both symmetric key encryption for (A_{cert},A_{info}) and CLE encryption for c_1. The input size for constructing all three parameters is shown in Table 5.4. The results of Fig. 5-9 illustrate that the AID creation cost is minimal (in ms), and hence, it would be efficient in practice. The variation of encryption time is also nominal due to the following reasons: i) AID creation uses symmetric key encryption, ii) c_1 is independent of the number of nominees, and iii) the size of $(N_{cert}, N_{receipt})$ are small.

Storage overhead. The various components used by the entities to implement Π are shown in Table 5.3. We can use an existing Aadhar-based system (in India) or Social Security Number (SSN) system (in the USA) as an IBS to design DAIP. The Aadhar-based IBS has to integrate the CLE component, which has the functionality to generate a master public/private key pair and a partial private key for a user. The NDP can be implemented using a storage system to store the asset inheritance data. For example, in India, Digilocker [165] is a centralized cloud-based storage service for an individual to store their data. CAP is a legal entity that confirms death within the state or territory. Many countries are already using some digital methods to confirm the death and generation of death certificates for a user. The same entity can be amplified to generate a relationship certificate as per the protocol Π and execute CLE decryption to disclose the first partial key. We want to point out that such a system would need a small overhead for storing (DC_U, RC_N^O).

The organization stores the list of nominees and their preference levels. It should support the signature scheme Σ to sign the message M_U^O. The remaining two entities, the user and nominee, do not need any separate storage space. We observe that the storage overhead in all the above-discussed entities is modest and can be easily implemented with the existing technologies.

In summary, the efficient results highlight that DAIP can be easily and effectively implemented in real environments and integrated with the current infrastructure by incorporating a small modification in the existing components.

5.7 Other discussion and Challenges

This section provides further discussion on comparison and challenges while designing DAIP:

Comparison between blockchain (decentralized) and centralized approaches: All four models claim they would simulate and implement digital asset storage and inheritance based on blockchain (decentralized) techniques, while NDP is modelled using centralized approaches. Both approaches may appear incomparable on certain parameters. However, the comparison mentioned in Table 5.1 and Table 5.2 concerning the design functionalities and security requirements is quite relevant and pertinent. It demonstrates the similarities, differences, challenges and applicability of storing and transferring various categories of digital assets in both settings. And, with the same indented objective comparisons are described in Section 5.5.1 and Section 5.5.2. For instance, all four models primarily target only Type 3 category assets e.g. Cryptocurrency Keys, NFTs and Tokens, but do not consider Type 1 category assets. It might be feasible that Type 1 category assets could be integrated in the future decentralised-based model with certain settings. However, the proposed system would become highly complex and inefficient in handling data due to the participation of many heterogeneous organizations.

Other differences include; i) A blockchain-based model would have the functionality of data integrity, tamper-proof, and transparency. The malicious processing of cryptographic data may be captured efficiently. ii) The integration of many entities over the blockchain platform would be complicated. iii) DAIP using a centralized approach is more scalable. iv) The implementation of a blockchain-based approach would be complex for Type 1 and Type 2 category assets.

Comparison with models following centralized techniques: Any design model in state of art or any service provider implementing centralized techniques to offer digital asset inheritance services could not be located. However, many organizations started allowing the transfer of data or handover of ownership of an account using centralized approaches. In this scenario, the successors are allowed either to download the inheritable personal data or they may be allowed to perform certain restricted activities.

For instance, Google provides the option of Inactive Account Manager service[1]. Any user

[1] Google's Inactive Account Manager https://support.google.com/accounts/answer/3036546

can set the *legacy contact (or trusted contact)*. A notification email is sent to a trusted contact to download the inheritable data if inactivity in the account is observed for a specified amount of time. The inactivity period is determined with the combination of several parameters such as the user's declared time limits or not using Google services (e.g. Gmail, Drive, Map) for a specified duration etc.

Facebook implements the option of a *Legacy account* [2]. The successor can perform certain limited activity on the deceased's account. For instance, he can change the main profile picture, post a final message, set the account as a memorialized account, download certain data, or request to remove the account. However, the legacy account holders are not allowed to read old posts or write new posts.

Similarly, Instagram [3] allows to *report a deceased person's account*. A successor may report the death of users along with legal documents such as birth and death certificates of the deceased and Proof of authority under local law. The successors are allowed to request to convert the account into a memorialized account or may request to remove the account.

The proposed DAIP model designs and simulates the inheritance functionalities, differently. For instance, i) DAIP does not restrict that nominees should have an account on the same platform (e.g. in Type 1 category assets) as happens in Facebook and Instagram. ii) The communication overhead for the nominee is minimal to know and claim the asset; ii) DAIP has the advantage that a nominee may receive the details of all assets in a single place.

Other Challenges in centralized NDP: A few other challenges may be considered. For instance, to efficiently implement DAIP; i) organizations should allow the option to declare the nominee and to store metadata as per the protocol. ii) DAIP is designed in semi-honest settings. Needs to analyze the security properties in dishonest settings also. ii) would need to integrate the DAIP functionalities of death certificate authority, IBS and NDP iii) Assurance of secure data storage because if the encrypted data is deleted, the details may be lost forever.

[2] Facebook Legacy Account https://www.facebook.com/help/1568013990080948/

[3] Instagram: reporting a deceased account https://help.instagram.com/264154560391256

Chapter 6

Data Breach Assessment

Data breach incidents are growing as threat actors have several adversarial benefits. The proposed Indian Digital Personal Data Protection Bill (DPDPA-2023) defines multiple constituents on data breaches and imposes high penalties in case of a breach. The service providers should design and implement systems with adequate security mechanisms to prevent data breaches. However, prevention may not always be guaranteed, and a breach may happen; therefore, it requires a suitable evaluation and assessment.

In the existing system, the correct assessment is not possible without the availability of the necessary details and a breach analysis model. In this work, we propose the system model of a Data Breach Incident Assessor (DBIA) aiming for breach evaluation that can assess and respond to data breach incidents (if they happen) within the organization. DBIA helps validate a threat actor's claim, understand the root cause of a breach, and analyze the scope of the compromise for mitigation of security gaps and robust compliance under DPDPA. The design of DBIA is simulated as a security information event management system. The simulation results and discussions show the necessity and efficacy of the model.

Recalling the background, a *data breach* is defined as an unauthorized access of system resources or stolen confidential information of victims without their awareness. As digital assets are growing over the internet, there is also a rise in data breach cases. Threat actors (TA) have a growing interest in assets because they can collect data with minimum effort (due to the cyber security aperture within the organization) and may leverage it for distinct adversarial and monetary benefits based on the nature and category of the asset. TA announces this critical information for sale on the dark web or social media platforms like Telegram. These activities are also indexed in some websites (e.g. *BreachForum*) or shared by threat intelligence communities (e.g. Figure 6-1[1]). The category of resources may include network access, admin account access, database dumps, list of user records, etc. Table 1 6.1 provides a detailed list of popular data breach categories, descriptions, and impacts. Further, TA sometimes has a double extortion impact on the organization by extracting data and then performing ransomware attacks. The Indian government's report describes a 53%

[1] By a Threat Intelligence Platform *FalconFeedsio* related to a *BianLian Ransomware Group* https://twitter.com/FalconFeedsio/status/1677191915324801024 (Last accessed July 2023)

From an organization's view, one of the objectives is: *to design a system to prevent data breaches*. Many design standards, best practices, directions, and guidelines exist and are issued from time to time by organizations, cyber security communities, and regulatory authorities (e.g., [20, 25]) to enforce stronger cyber security within the organization. However, The possibility of a data breach can not be ruled out and is occurring (*BreachForum*). The breaches may even occur within a strong network infrastructure system with advanced security systems [15]. We study how we should assess data breaches if they occur.

The motivation of this work is: *if a data breach happens or is announced, then how can we efficiently respond against it?*. A right response is essential because data breaches create panic. The panic is proportional to the amount of data loss, its social, business, and financial impacts, and the severity of the exposed information.

An adversarial event having cyber security impact is defined as *incident* such as malware, APTs, etc. The *incident handling* includes procedures that may be followed in case an incident happens. Its life cycle [52] includes activities to prevent, detect, contain, eradicate, and post-analysis an incident.

The current incident-handling activities are more aligned with the response of APTs, ransomware malware, etc. The analysis goal may consist of identifying initial access, root cause, analysis of the persistence and lateral movement mechanisms, etc. However, for data breach incidents, the analysis goals are not well formalized.

The right analysis goals are expected due to the following reasons: **i)** ***Regulatory compliance*** the latest data protection frameworks such as GDPR [74] and proposed India digital data protection bill (DPDPB-2022) [6] emphasize that it is the service provider's responsibility to apply the appropriate security on the data to prevent misuse. He has to pay a penalty in case a data breach happens. The penalty varies based on the type and volume of data an organization processes. Therefore, organizations must prevent data breaches and comply with the framework. **ii)** ***Claims validation*** publicly announced breaches may have a severe impact, so it is essential to evaluate claims thoroughly. **iii)** ***Impact assessment*** for instance, if TA announces a breach of 1 lac records and exposes the 100 records as sample data. The appropriate analysis is necessary to evaluate the impact. **iv)** ***Security enhancement*** formal analysis will lead to identifying the root cause of the data breach and, after that, enhancing the organization's security posture. **v)** ***Effort reduction*** the analysis will reduce the effort in repeated disclosure of the same data or a portion of data.

To achieve the above goals, data breach analysis has the following challenges: i) ***Disclosure challenges*** For instance, sometimes, the TA does not provide enough information to validate the claim, or the data is too generic, making analysis difficult. A detailed discussion is provided in

Section 6.1.3. ii) ***Analysis challenge*** The organization may not maintain suitable logs, security events, or shreds of evidence that can help in analysis. Details are given in Section 6.1.3.

The above analysis goals and challenges motivate integrating a framework that can help in data breach incident analysis and response. The aim is Preparing things that should exist within an organization to evaluate and assess the incident. At a high level, the analysis framework objectives are:

1. What are the things that can be integrated for formal analysis of data breaches?

2. What information will be stored and processed in the model?

3. How will the information and activities of item 1 and item 2 help analyze the root cause and respond to regulatory queries, and will fulfill the expectations of data breach analysis goals?

With the above goal, we propose the model of ***Data Breach Incident Assessor (DBIA)*** for evaluating and assessing data breach incident analysis.

- The proposed DBIA model consists of tools and algorithms that collect and process the significant security events and security logs essential for a data breach analysis compliance from systems of the organization's network.

- The collected data are processed in a dedicated system accountable for the collection, storage, and processing of data regularly to provide the answer to queries of item 3.

- The simulation of the DBIA model is done as a *DBIA SIEM*. The analysis results, use cases, and efficacy are discussed.

We urge that storing, logging, and processing the above information, aiming at data breach analysis, will help evaluate and assess data breaches efficiently.

Another aim of this work is to determine whether security analysis solutions may be costly. Small organizations run with limited budgets and do not emphasize security events. The purpose is to determine how these organizations can be equipped with the capabilities to capture important security events with limited resources.

The rest of the sections in the chapter are organized as follows: Sec. 6.1 describes the problem formulation, Sec. 6.3 provides preventive approaches to review system, Sec. 6.4 DBIA model, Sec. 6.5 and Sec. 6.6 have simulation and further discussions.

6.1 Problem formulation

DPDPA-2023 [6] defines regulatory expectations and constituents related to data processing. A data breach is a significant clause in the bill and has received the utmost attention. It is the service provider's responsibility to process the data lawfully and comply with the regulations. As per Section 4, *"DF shall implement appropriate technical and organizational measures.."* to adhere processing with the framework. Service providers shall protect the data by applying *appropriate security and safeguard measures to prevent personal data breach .. (Section 8(5))*. In case a data breach is observed, service provider *shall notify the board and each affected data principal's..*, as per Section 8(6). The provision of penalty is also proposed by the framework, which includes penalty in case of *failure of providing reasonable security and safeguards causing data breach or failure to notify the board or affected data principal, in the event of data breach ..*, as per Section 27. The penalty depends on various factors.

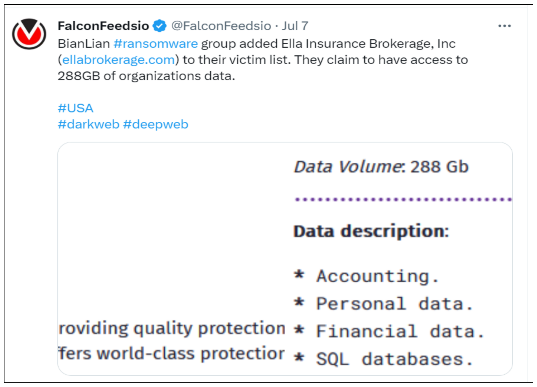
If a data breach happens, the organization is expected to perform a data breach analysis and identify why this breach happened. The list of entities and systems affected. Based on analysis, they should implement possible mitigation measures to prevent future recurrences.

6.1.1 Data breaches announcement

Table 6.1 the list of a few popular categories of assets announced as data breaches, possible reasons, and their impacts. We enlist a few reasons why data breach prevention and analysis should get the utmost attention:

i) ***Diverse category:*** Table 6.1 confirms that a diversified category of data can be sold as a data breach. **ii)** ***Severity*** The data breach may have a severe impact (e.g., disclosure of critical defence data) or a less severe impact (e.g., leakages of credentials of ordinary users of a website). **iii)** ***data breach beyond data protection*** DPDPB defines data from the angle of personal data. However, the announced data breach categories are more devastating. For instance, it can include design data of the manufacturing industry or may have access to the entire network after compromise. **iv)** ***immortality of data:*** Once data is breached, the owner has no control over it.

Therefore, the severity of data breaches depends on the nature and category of the information announced. The last column of Table 6.1 discusses the security expectation if any such data breaches happen.



FalconFeedsio @FalconFeedsio · Jul 7

BianLian #ransomware group added Ella Insurance Brokerage, Inc (ellabrokerage.com) to their victim list. They claim to have access to 288GB of organizations data.

#USA
#darkweb #deepweb

Data Volume: 288 Gb

Data description:

roviding quality protection
fers world-class protectior

```
* Accounting.
* Personal data.
* Financial data.
* SQL databases.
```

Figure 6-1: An Example of a claimed ransomware and data breach posted at Twitter

6.1.2 Incident life cycle and data breach incident

A security event that can have an adversarial impact within the system is defined as an incident consisting of low-severity incidents, e.g., network scanning to high severity, e.g., ransomware. The incident life cycle [52] consists of a set of activities an organization does if an incident happens. The activities for popular categories of incidents (e.g., malware) are well-defined. However, the activity and goal of analyzing data breach incidents are not well defined. The following sections discuss the challenges and expectations in data breach analysis.

6.1.3 Organization challenges in data breach handling

The organization may observe many inherent challenges when a data breach is announced, Ref Fig. 6-2. We divide it into two categories:

Disclosure challenge It consists of a list of challenges resulting from disclosed information evidence by the threat actor. For instance: i). *sample size is not enough:* to attribute and locate the victim's systems. ii) *data is very generic:* announced with victim's name but disclosed widespread data as a sample, e.g., name, email ID. iii) *unclear victim's name:* e.g., TA announces data breach as "banking sector" which does not specify victim name. iv) *false claims:* based on random data v) *Lack of evidence :* no sample data provided by threat actor vi) *old data:* data sets may be old, repeated disclosure, or a portion of an earlier data breach. vii) *exaggerated claims:* e.g., TA announcing data breach as "hacking of website" but showing sample data consisting of screenshots of successful user login of ordinary users indicating user compromise.

Analysis challenge It consists of a list of challenges that occur during data breach analysis. Generally, organizations do not pay much attention to designing the system considering data

Table 6.1: Popular categories of data breaches announcement

Data Breach Type	**Categories**	**Possible causes**	**Security impacts**	**Analysis requirements**
Access	Admin access, system access, network access web access, web admin panel access, exploits to enable access	Vulnerabilities (e.g. IDOR, open AWS instances), weak or default credentials, poor security configuration or misconfiguration, exposed services, malware (e.g. APTs)	Website, system, device compromise, data exfiltration, further attack.	Root cause, disinfection, security fixes
Compromise	Email compromise, credential compromise (single, multiple, or entire user's set)	Phishing, malware (e.g. Redline data stealer), vulnerabilities (e.g. SQL injection), weak or default credentials	Loss of sensitive information, email data, further attacks (e.g. sending spear phishing emails)	Root cause, disinfection of system and users
Data	Defense (sensitive information), government (e.g. Aadhar, confidential data), health, banking, financial, organizational (e.g. manufacturing)	Phishing, malware (e.g. Redline data stealer), vulnerabilities (e.g. sql injection), weak or default credentials	Financial impacts, sensitive information disclosure, loss of researches, patents, designs, copyright data, sensitive data exfilteration (e.g. PAN, Aadhar, accounts), loss of data related to state security, reputation damage etc, further attacks	Root cause, disinfection of system and users

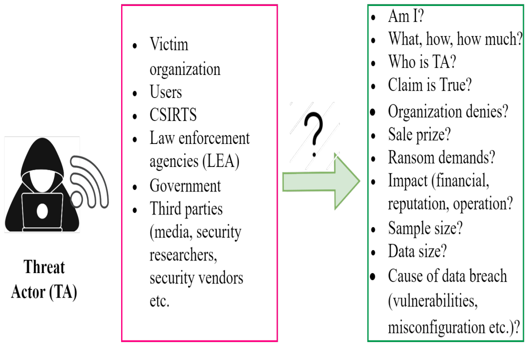

Threat
Actor (TA)
Victim organization
Users
CSIRTS
Law enforcement agencies (LEA)
Government
Third parties (media, security researchers, security vendors etc.
?
Am I?
What, how, how much?
Who is TA?
Claim is True?
Organization denies?
Sale prize?
Ransom demands?
Impact (financial, reputation, operation?
Sample size?
Data size?
Cause of data breach (vulnerabilities, misconfiguration etc.)?

Figure 6-2: Challenges of different entities in data breach analysis

breaches. Therefore, when a breach happens, its analysis does not proceed well due to several obstacles such as i) *no proper monitoring:* organization does not concern about data exposure; therefore sensitive data may be left unmonitored ii) *no proper logging:* security event logs play a vital role in analysis but devices may not configure with suitable logging mechanism iii) *data gathered from multiple sources:* e.g., data stole from two sources and mixed iv) *breach of unmonitored data* e.g., A small office storing PAN and Aadhar get breached, and threat actor is selling data in the name of actual owner (e.g., Aadhar). This makes analysis difficult. v)*distributed nature:* organization has offices at different geographical locations and lag with central monitoring. vi) *end user notifications:* whether end users should be notified? vii)*third party breach* e.g. data breach from a vendor viii) *impossible tracking* what data is disclosed from a computer source is unknown. ix) *Severity estimation* Each organization evaluating data breach may determine its severity differently.

6.1.4 Expectation in data breach analysis

The ideal expectation is that the purpose of data breach incident response should be well-defined. This was known for other cases (e.g., malware) of incident analysis but was not well-defined for data breach incidents. The analysis starts once the organization learns about the data breach incident. The first challenge during analysis is to verify if the data breach happened. If so, what is the data that was stolen? When does the threat actor get initial access? Given the exposed data and claim of the threat actor, the answer to the following queries may help in stronger compliance and analysis:

- How can the analysis be performed and the claims be validated?

- When did the data breach happen in the system?
- Which systems are compromised?
- What is the impact of exposed exfiltrated data or compromised resources?
- What is the amount of data exfiltrated?
- What is the impact on business, services, and organization?
- How the root cause and the existing gaps could be identified?

Responding to the above queries will help strengthen the evaluation and assessment of data breaches. The following section defines how the above goal can be achieved through the DBIA framework.

To simulate the design of the proposed model, we take the scenario of a small enterprise or organization consisting of hundreds of users. The reason for using this setting is small organizations usually have limited resources. The proposed frameworks can be integrated into these for stronger compliance.

6.2 Common causes of data breach

The data breach may occur through *”insider threat”* where someone from within the organization is disclosing data, or through *”external threat actor”* where the threat actor is an external figure. Another term *”data leak”* is used in parallel to the data breach, an instance of accidental data exposure due to system misconfiguration where no threat actor is suspected to have gained access to the data. We are studying the causes of data breaches primarily related to external threat actors. Refer Fig. 6-3, we describe the common causes as follows:

6.2.0.1 Credentials

Several data breaches are due to stolen passwords. Threat actors collect credentials through various mechanisms such as phishing, credential stealing malware’s [30], VPN compromise [1] or previously disclosed breaches. Exposed leaked credentials are published over the internet, which may be used by threat actors to perform ransomware and data breach attacks. For instance, one of the TTPs of BlackCat ransomware is to use VPN compromise credentials to infiltrate the network

[14]. Using default or weak credentials in the infrastructure is a strong attack vector to gain access to the system. Ex. user name *admin*, and password *admin*.

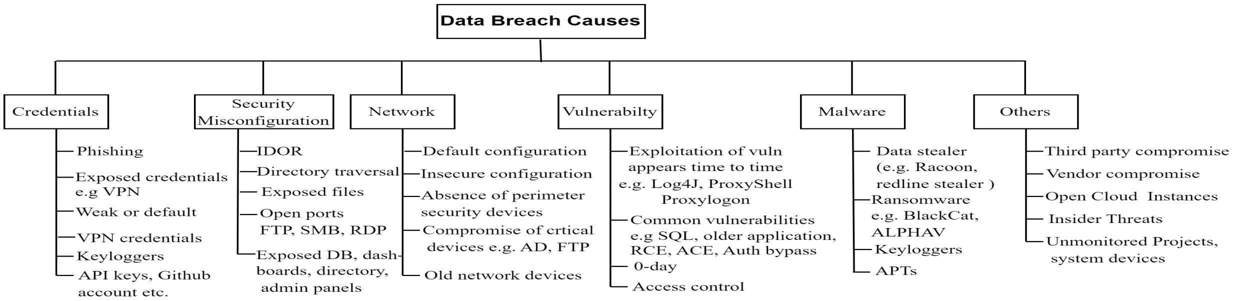

Data Breach Causes
Credentials
Phishing
Exposed credentials e.g VPN
Weak or default
VPN credentials
Keyloggers
API keys, Github account etc.
Security Misconfiguration
IDOR
Directory traversal
Exposed files
Open ports FTP, SMB, RDP
Exposed DB, dashboards, directory, admin panels
Network
Default configuration
Insecure configuration
Absence of perimeter security devices
Compromise of crtical devices e.g. AD, FTP
Old network devices
Vulnerabilty
Exploitation of vuln appears time to time e.g. Log4J, ProxyShell Proxylogon
Common vulnerabilities e.g SQL, older application, RCE, ACE, Auth bypass
0-day
Access control
Malware
Data stealer (e.g. Racoon, redline stealer)
Ransomware e.g. BlackCat, ALPHAV
Keyloggers
APTs
Others
Third party compromise
Vendor compromise
Open Cloud Instances
Insider Threats
Unmonitored Projects, system devices

Figure 6-3: Data Breach Causes

6.2.0.2 Security misconfiguration

The websites may have security misconfigurations such as directory listing, information disclosure (version of a product), exposed API keys, and git hub secrets parameters or codes.

Mis-configuration also includes common errors in settings that may lead to data exposure. For instance: hosting public folders, hosting public shared directories, exposing common ports (e.g., FTP, RDP, SMB), exposed database, exposed dashboard (e.g., kibana dashboard displaying data), inappropriate indexing of file, exposed admin interface, exposed passwords, etc.

6.2.0.3 Network

Many organizations use a flat network where users and critical systems are in the same network. These network does not implement segmentation's [124] such as constructions of the militarized zone (MZ) and demilitarized zone (DMZ), segregation of users and critical services, etc., leveraging threat actors to gain a strong foothold within a network after initial access. Further, many organizations are not equipped with perimeter devices such as firewall SIEM, WAF, etc., which makes them more vulnerable. If the networks have all these, they suffer from challenges such as older, unpatched, vulnerable devices, misconfiguration of parameters, etc. The compromise of perimeter devices (e.g., compromise of firewall, then the compromise of Domain controller/Active Directory) is a very popular technique to exfiltrate data from an infrastructure [14].

6.2.0.4 Vulnerabilities

Many vulnerabilities (both in applications and products) exist in the system where security is not duly taken care of while designing. The necessary security is either not implemented in the system, or it is implemented with errors. For instance, 1).*Common vulnerabilities*: the websites may have common vulnerabilities such as SQL injection, cross-site scripting (CSS), improper file upload/download, and Indirect object reference (IDOR) 2). 3) *incorrect validation*: the websites do not have appropriate verification mechanisms such as No Rate Limit (NRL) in parameters (e.g., allowing submission of OTP multiple times), sensitive information disclosure (e.g., admin pages, library, source code), allowing brute-force attempts. 4) *authentication*: The websites do not have any user authentication methods or an error-based authentication mechanism. 5) *services* The websites expose unnecessary services such as running insecure RDP.

Vigilant hackers are constantly searching for weaknesses in computer systems and exploiting them whenever they come across them. E.g., high severe vulnerabilities of products appear from time to time and are targeted by threat actors. For instance, *Apache server log4j* vulnerability targeting remote code execution, *Maggie backdoor* targeting MS SQL servers, etc. Many security organizations publish the list of most exploited vulnerabilities that require immediate attention. For instance, CISA maintains the list of known exploited vulnerabilities [11].

6.2.0.5 Malware

The malware steals data and later uses it for a data breach. One popular category is Ransomware, which was primarily launched for monetary benefits and other purposes. Multiple ransomware families are active in cyberspace and are continuously on the rise [10]. The threat actor uses advanced techniques to avoid detection [14]. They exfiltrate data, then perform encryption on systems (to make them unusable, unavailable for the organization), and later demand ransom. Non-payment of ransom may lead to a data breach, as warned by the threat actors.

Many APT-based threat actors also steal data, for instance, APT36 [158]. These data may be sold as data breaches to gain adversarial advantages.

Similarly, the data stealer malware family, for instance, *Redline Stealer*, continuously collects user's credentials [30]. The data is regularly sold on the dark web.

6.2.0.6 Others

Data breaches occur when third parties, vendors, etc., do not implement adequate security mechanisms. The data breach from third parties will also impact the organization. *Cloud Mis-configuration:* Mis-configured cloud instances, such as open AWS buckets and AZURE instances, may lead to a data breach. *Insider Threats* The threat actors within the organization, such as disgruntled employees and third-party vendors, may disclose the data.

6.3 System Review

Initially, assess and harden the existing system and devices of the organization before capturing the necessary details for analysis. The following things should be assessed:

6.3.1 Network assessment and assets review

Let's take the scenario of a small enterprise/organization with hundreds of users. The network comprises perimeter devices(e.g., Firewall/UTM), Windows-based servers, application servers, DB servers, and optional VPN servers. The Fig. 6-4 denotes the list of systems and devices. The detailed design architecture of the network and communication mechanism is not discussed but may be designed following the best practices [20].

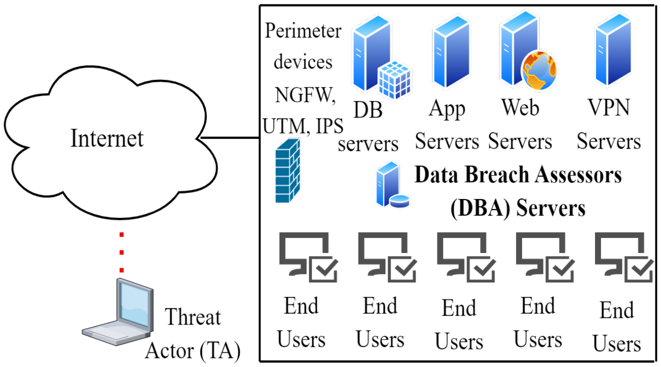


Figure 6-4: A DBIA network scenario

Constructing a secure network infrastructure can provide a stronger level of security. It includes the following: A). *structured network*: contains segmentation of devices, users, and applications based on their criticality. For instance: 1) each department should be in a different network segment 2) identify the department and nature of connectivity required with each department; the communication between departments should be allowed only if it is essential. 3) Critical assets should have further segmentation based on needs, role, and necessity, e.g., public-facing server in a separate segment, database server in a different segment, internal server in a separate segment, and other internal resources in a separate segment. B) *firewall configuration* All firewall rules should be correct, verified, and include conditions only as per the business requirement. C) *security monitoring devices* inclusion of security devices such as SIEM, IPS, IDS, etc. may provide stronger threat detection D) *system hardening* the systems should be hardened with best security policies.

Review assets and review of potentially exposed information with stress on the following: i) ***Asset lists:*** Discover all physical systems/devices, including endpoints, servers (including the list of both physical and virtual servers), perimeter systems, and other devices (e.g., wi-fi access points). These are necessary because any unmonitored device may get exposed to a data breach. Also, it makes data breach analysis difficult. ii) ***asset categorization*** ensure network assessment is done with proper configuration and a secure architecture is implemented. Maintain the information on network segmentation, private virtual LAN, public IP pools, and a list of critical and non-critical assets. iii) ***Hardening list*** Maintain the list of App version, OS version, and hardening status iv) ***exposed data*** Maintain the list of places that may expose data along with their criticality). It is required because sometimes data may get exposed unintentionally. *Example*: a test bed network, unmonitored URLs, or URLs running with default credentials.

The above check is useful in identifying and locating devices, systems, and resources in data breach response.

6.3.2 Vulnerabilities and threats review

Threats and vulnerabilities are the major attack vectors for a data breach. Many organizations run with older, unpatched applications that make an easy target for threat actors. The security audit of networks, applications, and websites will enable the disclosure of the existing vulnerabilities in the system [171].

Maintenance of a list of vulnerabilities and threats (e.g., as shown in Fig. 6-3) may be prepared, consisting of a detailed list of possible threats causing data breaches. Mitigate these to minimize the probability of data breach. We stress to ensure enlisting of relevant details that can be helpful in data breach analysis. For instance, maintain the hardening status of the vulnerability and configuration. *Example 1:* knowledge of how many systems are running and hosting data belongs to project *A*. *Example 2:* the data hosted in the cloud. *Example 3:* Listing if any security misconfiguration (e.g., IDOR, directory listing, etc.). These will help to identify the root cause.

Enlisting the above details in the system will help in data breach incident response.

6.4 Designing of a system to evaluate and assess the impact of DB

The aim is to design a system/model to process significant information useful in analyzing data breaches. The correct analysis will help verify the threat actors' claim, identify the root cause, and collect stronger evidence to perform incident analysis. This will also help in compliance with the requirements of regulatory authority.

6.4.1 Data Breach Incident Assessors (DBIA)

We propose the model of a ***Data Breach Incident Assessor (DBIA)***. The DBIA collects and stores the significant event from multiple systems and processes it for analysis. The design of DBIA is based on the SIEM. A ***Security Information Event Management (SIEM)*** is an application that can manage and process security-related events for better visualization, analysis, and threat alert purposes.

The architecture of SIEM is open to interpretation, and its design model can be implemented based on the nature and category of logs to be handled. Here, we design the SIEM to collect and process the logs aiming for data breach response.

We use "ELK", an open-source project, to implement the SIEM application for DBIA purposes. ***ELK*** is abbreviation of three open source project; Elasticsearch, Logstash, and Kibana. The ELK can collect the logs from distinct sources and process them for visualization and analysis.

We define data to be logged, aiming for the data breach analysis. The DBIA model consists of the following set of activities: 1) A set of algorithms and tools to enable log and logging information 2) Algorithms to collect the logged data 3) Preprocessing 4) Storage of data 5) Post-analysis. A pictorial description of steps is given in Fig 6-5.

6.4.1.1 Logging of events

A system generates many security events during run time. Considering the data breach analysis aim, enabling the appropriate event logs may be helpful.

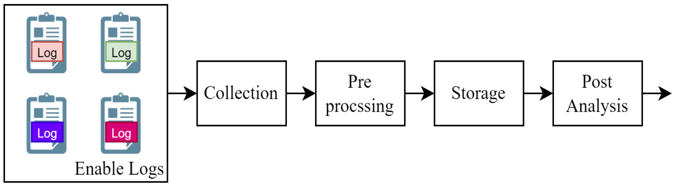


Figure 6-5: Event logs processing steps

Table 6.2: Examples of threats and helpful logs to identify

S. No.	Threat examples	Supported logs
1.	Network logon, RDP etc	Windows event log
2.	Security Misconfiguration (IDOR, auth bypass)	Web Server Access log
3.	SQL injection	Web server access logs, DB access logs
4	Malicious VPN login	VPN authentication logs
5	API exploitation	API access logs

We take one example to show the importance of enabling logs: *Suppose in a data breach a threat actor announces a list of files belonging to an organization*. A threat actor might log in by exploiting RDP services and exfiltrate data in this scenario. Windows system logs the network login activity in Windows event logs. If these event logs are enabled, they can provide evidence of threat activity.

Table 6.2 shows examples of threats and the relevant logs that can provide supporting evidence during analysis if threats are executed successfully. Log category may differ from system to system, and by default, it may be enabled or disabled. Table 6.3 shows the examples of log categories some applications provide. The necessary settings may need to be enabled for collecting and processing the relevant logs.

6.4.1.2 Collection

The collection operation collects the relevant details from various devices. The relevant logs may be enabled based on the nature and category of the devices. Generally, the server and application provide options to configure logs. Once the correct gets enabled, the system starts collecting the logs.

For endpoints, we design algorithms or tools to capture the required details. For example, integration of sysmon [29] tools to capture critical information.

Table 6.3: List of logs help in identifying data breach cause

S.No.	**Device**	**Log**
1.	Web server	Web server access logs, WAF, error logs etc. [3]
2.	DB server	Application and DB access logs. [27]
3.	End points	Autorun, active connection, windows event logs, windows registry, Files modifications, sysmon, other details. [29]
4.	VPN Server	Authentication, user access, session logs. [16]
5.	Firewall	Traffic low, events (e.g. system and administrative events) , UTM logs (e.g. protocol violation, network intrusion, SSH, SSL, DNS, WAF, IPS, AV events etc.). [8]
6.	Application Server	Application access logs, API access logs, session logs. [9]

The collection process runs regularly. The following things will be defined during log collection: i) *Duration* the time logs will be captured. ii) *Storage* The amount of storage to be provided for log collection. Define the location (local, remote, network, or cloud) for storage. Further discussion on storage is given in Sec. 6.6. iii)*Configuration* The logs can be collected in default settings or manual configurations. The changes in configuration impact storage and processing time. Therefore, it should not be disproportionate iv) *Backup policy* A secure backup procedure as the malware may delete/encrypt the captured logs.

6.4.1.3 Preprocessing

The collected data are heterogeneous, for instance, sysmon logs (.evtx), windows event logs (.evtx), windows autorun (.csv), web server logs (csv), etc. The preprocessing steps run algorithms for the preprocessing of data to make it in a processable format. Further, many logs are in their proprietary format, which needs appropriate conversion before processing. We must define a common format to aggregate and process all the logs.

6.4.1.4 Storage

The processed data is stored on a centralized storage system. It receives data from various devices. The storage depends on the following parameters: i) the number of devices pushing logs into the storage system, ii) the amount of data generated by each system, iii) the duration to which logs are retained, and iv) the data processing capability of the storage system.

6.4.1.5 Post analysis

The stored data is analyzed for two cases: i) The data is processed using a log analyzer. The result is used to perform threat hunting and proactive threat alert generation, filter-based alerts, and any further analysis and correlation. The alerts could be like: “A threat actor is trying to exploit some vulnerabilities.”. ii) The data is processed using a log analyzer during data breach analysis to extract the relevant evidence. Sec. 6.5 discusses the simulation of post-analysis steps.

With the help of algorithms, processing methods, and the above steps, we can process the data for data breach analysis and response.

6.5 Simulation and use cases discussion

We discuss the model simulation, examples of scenarios, and use cases of how DBIA-SIEM can satisfy the data breach analysis goals.

The simulation was performed on a Windows machine with configuration as Intel(R) Xeon(R) CPU ES-1660 V3@3.00GHz process, 64 GB RAM, and Windows 10 Pro OS. A total of 5 virtual machines are created as end users in a virtual box, each having Windows 10 OS and 4 GB of RAM. The ELK instance is constructed on a virtual machine running Ubuntu 20.04 LTS with 16 GB of RAM. We enable the following logs for analysis and to discuss the implementation use cases:

End points we enable the following logs in endpoints: i) *windows autorun* [2] ii) *netstat* command. A script is created and executed on every machine to execute both. The script runs at a specified interval (say 3 Hours), and the output is stored in local storage and sent to the ELK server. iii) *sysmon*[3] A sysmon is a Windows application, one installed in the system, remains persistent and used for monitoring and logging activities of Windows event logs. Its configuration has multiple capabilities (such as file changes, network, connection, inclusion and exclusion of certain files, hashes, process GUID, etc.) to log. The storage and computation are proportional to the configuration. Enabling many logs may have high resource utilization, while less configuration can leave important details. We use a configuration file “sysmon modular”[4] with a default/balanced configuration option. Sysmon is enabled in every system with the above configuration, and the log is pushed to ELK using the “win-log-beat” application provided by the ElasticSearch community.

Web server We have exported the web server access log of two Apache web servers of one-

[2] https://learn.microsoft.com/en-us/sysinternals/downloads/autoruns

[3] https://learn.microsoft.com/en-us/sysinternals/downloads/sysmon

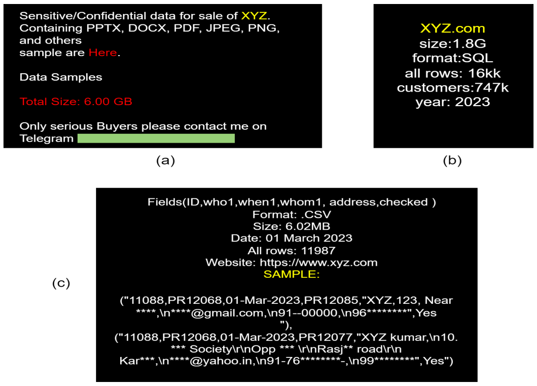
[4] https://github.com/olafhartong/sysmon-modular

month duration and pushed it to ELK using the “filebeat” client of ELK. The total size of logs is 80 GBs and 13 GBs (in tar.gz format). Logs are stored and processed for analysis. The data is anonymized to show results.

We discuss the following scenario of data breaches and how the above logs are helping in the analysis.

Scenario 1 *Suppose a threat actor announces a data breach of important files (e.g., pptx, pdf, images, docx, etc.) belonging to an organization, for instance, refer Fig.* 6-6*(a).*

The analysis starts based on the sample data provided by the threat actor. The system containing the above files may be probably infected. A system is identified and isolated. The following steps are performed for the analysis: i) *Locating the presence of any malware* during the analysis, a few suspicious sample files have been identified, created in the directory “C://ProgramData/ -LSBController”. The malicious files LSBController.exe (7f915ffd0d57f177ce -a88f15cd74be0f) dotsqueeze.dll(ca2c477f1632c05e481b985cd9e05e17) oraclenotepad45.dll(8e868999bb513c496 - a8e0ab82da436c0) are associated with RAT. The presence of the malicious files, along with additional payloads, are shown in Fig. 6-7. ii) *Identifying the root cause* the analysis is performed with the logs shipped to DBIA. It indicates that malware was first executed on 14th October 2022. The execution entry of the executable LSBController.exe is shown in Fig. 6-8.



Sensitive/Confidential data for sale of XYZ.
Containing PPTX, DOCX, PDF, JPEG, PNG,
and others
sample are Here.

Data Samples

Total Size: 6.00 GB

Only serious Buyers please contact me on
Telegram [redacted]

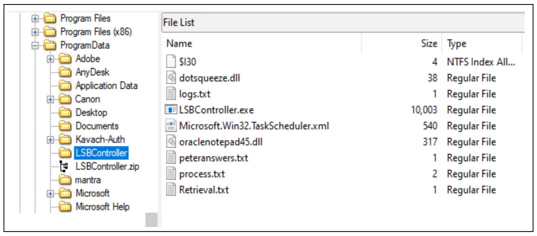
XYZ.com
size:1.8G
format:SQL
all rows: 16kk
customers:747k
year: 2023

```
Fields(ID,who1,when1,whom1, address,checked )
Format: .CSV
Size: 6.02MB
Date: 01 March 2023
All rows: 11987
Website: https://www.xyz.com
SAMPLE:

("11088,PR12068,01-Mar-2023,PR12085,"XYZ,123, Near
****,\n****@gmail.com,\n91--00000,\n96********",Yes
"),
("11088,PR12068,01-Mar-2023,PR12077,"XYZ kumar,\n10.
*** Society\r\nOpp *** \r\nRasj** road\r\n
Kar***,\n****@yahoo.in,\n91-76********-,\n99********",Yes")
```

Figure 6-6: The example of threat actor announcement of data breaches, (a). List of files (b). The SQL database dump (c). The list of user’s records of the organization

Further analysis of samples reveals it has data exfiltration capabilities. The analysis requires forensics, identification of suspicious activities, malware analysis, and timeline correlation with the stored logs.

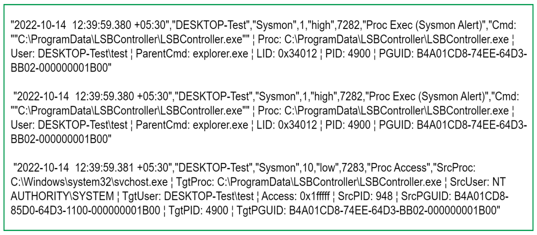
Scenario 2: *A threat actor announces a data breach showing a sample screenshot of the schema of database dump of an organization for instance, refer Fig.* 6-6*(b).* The threat actor



File List

Name	Size	Type
$I30	4	NTFS Index All...
dotsqueeze.dll	38	Regular File
logs.txt	1	Regular File
LSBController.exe	10,003	Regular File
Microsoft.Win32.TaskScheduler.xml	540	Regular File
oraclenotepad45.dll	317	Regular File
peteranswers.txt	1	Regular File
process.txt	2	Regular File
Retrieval.txt	1	Regular File

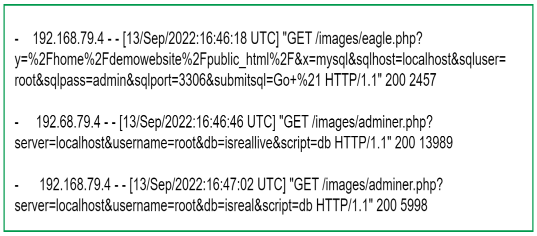
Figure 6-7: The snapshot of the presence of malicious files in the infected system, probably leading to data breach



```
"2022-10-14  12:39:59.380 +05:30","DESKTOP-Test","Sysmon",1,"high",7282,"Proc Exec (Sysmon Alert)","Cmd:
""C:\ProgramData\LSBController\LSBController.exe"" ¦ Proc: C:\ProgramData\LSBController\LSBController.exe ¦
User: DESKTOP-Test\test ¦ ParentCmd: explorer.exe ¦ LID: 0x34012 ¦ PID: 4900 ¦ PGUID: B4A01CD8-74EE-64D3-
BB02-000000001B00"

 "2022-10-14  12:39:59.380 +05:30","DESKTOP-Test","Sysmon",1,"high",7282,"Proc Exec (Sysmon Alert)","Cmd:
""C:\ProgramData\LSBController\LSBController.exe"" ¦ Proc: C:\ProgramData\LSBController\LSBController.exe ¦
User: DESKTOP-Test\test ¦ ParentCmd: explorer.exe ¦ LID: 0x34012 ¦ PID: 4900 ¦ PGUID: B4A01CD8-74EE-64D3-
BB02-000000001B00"

 "2022-10-14  12:39:59.381 +05:30","DESKTOP-Test","Sysmon",10,"low",7283,"Proc Access","SrcProc:
C:\Windows\system32\svchost.exe ¦ TgtProc: C:\ProgramData\LSBController\LSBController.exe ¦ SrcUser: NT
AUTHORITY\SYSTEM ¦ TgtUser: DESKTOP-Test\test ¦ Access: 0x1fffff ¦ SrcPID: 948 ¦ SrcPGUID: B4A01CD8-
85D0-64D3-1100-000000001B00 ¦ TgtPID: 4900 ¦ TgtPGUID: B4A01CD8-74EE-64D3-BB02-000000001B00"
```

Figure 6-8: The capturing of events in sysmon logs analyzed using DBIA for the execution of the application "LSBController.exe"

shares screenshots of the database schema. In this scenario, one possibility is that the threat actor has exploited SQL injection vulnerability or has unauthorized access to the database.



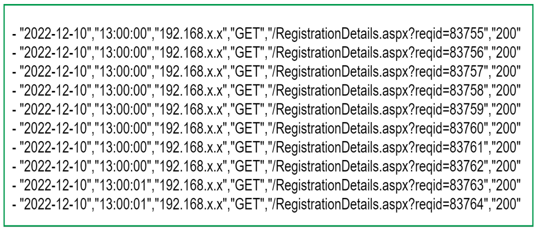
```
-   192.168.79.4 - - [13/Sep/2022:16:46:18 UTC] "GET /images/eagle.php?
y=%2Fhome%2Fdemowebsite%2Fpublic_html%2F&x=mysql&sqlhost=localhost&sqluser=
root&sqlpass=admin&sqlport=3306&submitsql=Go+%21 HTTP/1.1" 200 2457

-    192.68.79.4 - - [13/Sep/2022:16:46:46 UTC] "GET /images/adminer.php?
server=localhost&username=root&db=isreallive&script=db HTTP/1.1" 200 13989

-     192.168.79.4 - - [13/Sep/2022:16:47:02 UTC] "GET /images/adminer.php?
server=localhost&username=root&db=isreal&script=db HTTP/1.1" 200 5998
```

Figure 6-9: The sample access logs showing the login activities of TA using a web shell "eagle.php", user "root" and with a default password

The analysis of logs shows that the attacker(s) has logged into the MYSQL database with the default credentials as root database user using IP address 192.168.79.4, refer Fig. 6-9. Later, the threat actors accessed the database using the "adminer.php" web shell present in /images/ directory to download the data from the database. This web shell is usually used to access the web application database remotely.

Scenario 3 *A threat actor announces a data breach consisting of a list of records of several users with attributes like user ID, name, email ID, mobile number, address, etc. The TA also publishes some sample records; for instance, refer to Fig. 6-6(c).*

In this scenario, one possibility is that threat actors might have exploited some vulnerability (e.g., SQL injection, IDOR) to exfiltrate the data. The IDOR (InDirect Object Reference) is a security misconfiguration vulnerability that may allow data enumeration. As it looks like a legitimate query, it may not be detected by a threat monitoring solution.



```
- "2022-12-10","13:00:00","192.168.x.x","GET","/RegistrationDetails.aspx?reqid=83755","200"
- "2022-12-10","13:00:00","192.168.x.x","GET","/RegistrationDetails.aspx?reqid=83756","200"
- "2022-12-10","13:00:00","192.168.x.x","GET","/RegistrationDetails.aspx?reqid=83757","200"
- "2022-12-10","13:00:00","192.168.x.x","GET","/RegistrationDetails.aspx?reqid=83758","200"
- "2022-12-10","13:00:00","192.168.x.x","GET","/RegistrationDetails.aspx?reqid=83759","200"
- "2022-12-10","13:00:00","192.168.x.x","GET","/RegistrationDetails.aspx?reqid=83760","200"
- "2022-12-10","13:00:00","192.168.x.x","GET","/RegistrationDetails.aspx?reqid=83761","200"
- "2022-12-10","13:00:00","192.168.x.x","GET","/RegistrationDetails.aspx?reqid=83762","200"
- "2022-12-10","13:00:01","192.168.x.x","GET","/RegistrationDetails.aspx?reqid=83763","200"
- "2022-12-10","13:00:01","192.168.x.x","GET","/RegistrationDetails.aspx?reqid=83764","200"
```

Figure 6-10: The sample access logs showing the IDOR enumeration of user's records

The analysis of logs using DBIA shows that the TA has enumerated a list of records from the website. It concludes the suspected URL has the presence of IDOR vulnerability, leading to the enumeration of data. Fig. 6-10 shows sample entries from the logs.

The above examples show data breach response was possible only if the relevant logs correlating data breach incidents were enabled and collected safely.

6.6 Further discussion

We discuss a few more constituents: i) The proposed model differs from traditional SIEM as it urges the collection of logs aiming for data breaches and has more rigorous logging. As a similarity, the collected data may also be used for threat alerts and proactive monitoring, similar to traditional SIEM.

ii) It may be difficult to determine which log and time duration to look for to validate the threat actors' claim if the timeline of the breach is not clear iii) It is difficult to determine the duration for which the logs should stored. Indian cyber security guideline [7] advised to store the logs for at least six months, which may be a safer approach. iv) Storage and computational performance of DBIA. The logs may be huge depending on the number of systems and configuration. As the work emphasizes how the data breach can be evaluated and assessed, the computation performance is not discussed here. However, we can use the references of ElasticSearch[5] for further details.

[5] https://www.elastic.co/blog/benchmarking-and-sizing-your-elasticsearch-cluster-for-logs-and-metrics

6.6.1 Data breach minimization techniques

We also list the approaches to reduce the possibility of data breaches and minimize their impact through the example of data processing in the education sector.

Data breach incidents are impacting education sectors. For instance, the ransomware incident on India's prominent institution AIIMS [163], or an alleged leak of students' data listing at a breach forum [17], urge institutions for stronger data management.

The education sector includes data of varied degrees, with less importance, such as school data to high severity, such as medical institutes, research laboratories, and patent design data requiring different protection degrees. We enlist the few possible reasons for data breaches in this sector: i) It is unregulated [72], institutions are not bound; therefore, securities are not correctly monitored. ii) Many college institution websites may be running on older technologies. The software and applications run with the older version and are not updated. iii) Security vulnerabilities are not monitored. The common vulnerabilities, such as XSS, SQL injection, etc., are not updated. iv) Sometimes, websites disclose more data than necessary. For instance, disclosure of results and personal information such as mobile number. v) websites may not have proper security devices in place such as WAF, SIEM, etc. vi) Institutions may have poor network infrastructure without any structured network vii) Many of the websites are created without applying appropriate security by design principles.

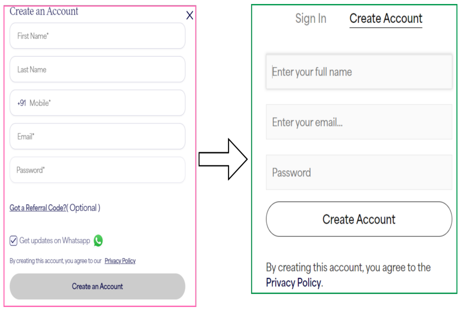
The education sector is diversified in nature. It includes medical, atomic, engineering, science, and many other fields. Therefore, the nature, category, and sensitivity of the data it holds are equally important and require significant protection. The organization must assess the data's sensitivity and criticality and emphasize its security. They need to understand and estimate the impact of the data breach (if it happens to them). Then, they must redesign the infrastructure to minimize the impact of the breach. The analysis should include impact aspects such as storing, processing, sharing, and disclosing data and preventing data breaches.

We describe the incorporation of various constituents as a preventive step to reduce the possibility of data breaches and minimization of data breach impact impacts:

6.6.1.1 Minimum data collection

Minimum data collection will help service providers process less data, reducing the probability of more data loss in case of a breach. Take an example of user account creation, Fig 6-11. The first part takes multiple fields (e.g., name, email ID, mobile no, and enabling WhatsApp services) while the second part collects lesser data (e.g., name and email) and achieves the same goal of ”account creation” of a user. If a data breach happens, the first one will reveal more data about users than the second one. Therefore, minimum data collection can minimize the impact of a data breach.

Data minimization techniques enable service providers to collect minimum data. Several techniques have been discussed in [157].



By creating this account, you agree to our Privacy Policy

By creating this account, you agree to the Privacy Policy.

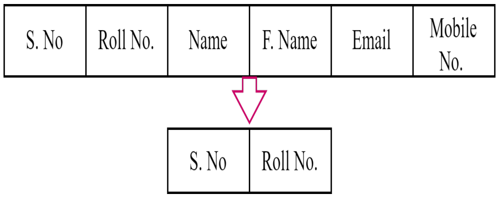
Figure 6-11: Minimum data collection

6.6.1.2 Minimum data disclosure

When users perform a query, many organizations disclose more data than essential, Ref 6-12. For example, an organization discloses the results of all students who passed an exam. The disclosure includes the details of all students (including name, roll number, F.name, email ID, and phone number). This disclosure publishes more than the requirement. The same goal is to ”disclose the result ” by hiding some data. The second part of Fig. 6-12 shows the same thing and discloses only the ”serial number and roll number” of all passed students.

Similarly, many service providers host the data in different file formats, such as MS Excel, PDFs, or Word documents. These files consist of user details and are accessible to all, leading to information disclosure to threat actors.

Strong authentication and access control can be implemented to achieve the goal of minimum information disclosure [157].



S. No	Roll No.	Name	F. Name	Email	Mobile No.

S. No	Roll No.

Figure 6-12: Minimum data disclosure

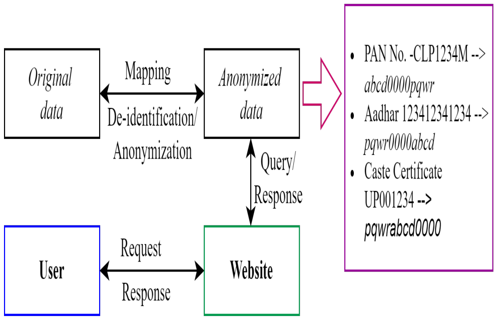

Original data
Mapping
De-identification/ Anonymization
Anonymized data
PAN No. -CLP1234M --> abcd0000pqwr
Aadhar 123412341234 --> pqwr0000abcd
Caste Certificate UP001234 --> pqwrabcd0000
Query/ Response
User
Request
Response
Website

Figure 6-13: De-identification and anonymization of data

6.6.1.3 Redesigning of system with consideration of data breach

The design of current systems is based on business requirements. Since the data breach (as defined in the protection framework) imposes a significant penalty, the organization must start designing systems by considering the impact of the data breach. This includes assessing the data collection, sharing, and processing based on the consequences of a data breach. For example, if an organization is collecting an Aadhar card, they should evaluate the impact of the breach of the Aadhar card from their system. Then, the system should be redesigned to minimize the consequences of an Aadhar card breach.

6.6.1.4 De-identification and anonymization of data

This means reshaping/conversion of data to a non-identifiable form. This Processing has an advantage as if a data breach happens; the threat actor will have anonymized data, which is less useful. One example is given in Fig. 6-13. Suppose an institute collects users' PAN card and Aadhar card information. If we assume that this data is collected only for KYC purposes, then such data may be stored in encrypted format at a secure place. A de-identified, anonymized mapping can be stored for regular processing. Every time, the anonymized data will be accessed. If a data breach happens, the threat actor will access only anonymized data at the regular server, having the minimum impact. A few methods of anonymization techniques are discussed in [157].

Designing a framework by integrating the above constituents can provide stronger protection from breaches.

Chapter 7

ESUL Analyzer

Cyber threats are growing, and almost everyone is being impacted. In cyberspace, many systems run as standalone or as a small network segment with limited users. The systems are not equipped with adequate network-level security monitoring solutions. The implementation of these may also be costly. Further, as a standalone system, the security is enabled with endpoint security such as antivirus that usually works on signature-based detection methods. As the capabilities of threat actors increase, they bypass detection, lure victims, and persist for an extended period.

The latest Indian cyber security regulation emphasizes that enabling different logs may play a vital role in defending cyber security and incident response. We propose the model of an end-system URL log (ESUL) analyzer for URL-based threats present in standalone systems. The model continuously analyzes the user's browser history logs of the End system (ES) and announces the list of malicious URLs, if visited previously, based on a received adversarial list. This early threat identification from log data will help end users learn about threats, perform incident response, and minimize their impact. It also assists users with relevant advisory and best practices. The model is simulated using a phishing database library, and the results describe its efficacy.

Revisiting the background, threats having an adversarial impact within the system are defined as incidents, for instance, phishing, malware, APTs, ransomware, etc. As a primary objective, the incident should not happen; however, if it occurs, it should be identified and mitigated. The incident response model describes high-level procedures for threat identification, analysis, and removal of infections from the system [52]. The rise in cyber incidents [10, 143] urge victims to detect incident at the earliest, estimate associated risk, and take appropriate action to minimize the impact.

Phishing and malicious URLs are the most common categories of threats. Phishing allures targeted individuals to steal sensitive information while malicious URLs tempt victims to visit it for adversarial intent, e.g., downloading malicious executables. Many APT-based attacks use such techniques as reconnaissance steps on an organization to make initial footholds [158]. If a user falls victim, the threat actor may have a whip hand over them. He may perform additional threats such as email compromise, malware outspread, privilege escalation, or network propagation.

In this work, we study phishing and malicious URL threats from the point of inclusion of incident response to end users. Generally, end users do not understand and analyze the impact of threats if any of this has infected them, consequentially leading to improper compliance and response. As cyber threats are growing, our goal is for all end users to understand the threats, their impact, and the scope of compromise and be able to apply suitable responses efficiently. We emphasize that including these can enable them to take robust corrective actions.

We discuss the existing approaches users generally follow to handle adversarial URL incidents. We identify the existing gaps and propose designing our model ESUL analyzer to help end users perform effective incident response. Currently, the following approaches are used: i) *user awareness* - the user is aware of the threats and does not fall victim. ii) *endpoint protection system (EPS)* - e.g. antivirus detect threats and issue warning/alerts. iii) *web browser warnings*- using the behavioral analysis of the visited page or through prior knowledge regarding the URLs. iv) *network level*: e.g., using network level perimeter devices such as intrusion prevention system, SIEM, firewall, or using the behavioral analysis of threat information shared by collaborative teams.

Due to the above methods, the following three categories of reason may be bypassed, and threats may occur: ***A. Advanced attack vectors***- the threat actor's capabilities are changing, which sets a strenuous challenge to users such as: i) sometimes, the signatures of newly created malicious URLs are unknown and, therefore, may not be detected by EPS, and users may fall victim. For instance, the malicious domain ”kavach-app[.]com” and ”kavach-app[.]in” is recognized by only 8 out of 96 antivirus engine at virus total (as visited on March 2023) [31] ii) threat actors are using advanced techniques to bypass detection such as encoding of URLs in password protected pdf or zip files, constructing emails looks more legitimate (e.g. received from senior authority), or impersonating legitimate urls. iii) sharing known threats among other teams may be delayed, providing a sufficient timeline to perform attacks. ***B. Absence of security monitoring solutions***- a large set of systems runs as a standalone system or as a part of a small network. Implementing security devices performing proactive network monitoring may be costly and, hence, not feasible in such a scenario. ***C. Threat existence for short duration*** Many malicious domains are created on a demand basis and exist only briefly. Once the URLs get offline, security solutions may not integrate their signature. Therefore, detection and alerts may not generated. The short-term domain has enough potential to target victims.

The above-identified reasons urge us to include additional mechanisms to detect threats in the system. For this work, we aim at threats arising from URL-based activities. We have analyzed and solved the following problems:

- How the notable threats can be easily communicated and shared with users?

- What are the advantages of regular analysis of logs in ES?
- How may the log analysis be efficient for early incident detection?
- How can every end user apply the incident response method effectively to minimize the impact?

With the above goal, we have proposed the model of an ESUL analyzer for inceptive threat identification and detection. The model runs in two sections. Both sections are circular and run concurrently. The first one performs synchronization of notable threats. The second section consists of an analyzer that collects and aggregates the history from distinct sources, e.g., multiple browsers. It analyzes it at regular intervals based on received threat lists. The user may review all threats and may take necessary action. In particular, we have proposed the following:

- The model of a concurrent, synchronized list of threats is proposed, particularly for threats that target a mass number of people or hundreds of users of an organization.
- A description is provided urging the necessity of integrating such a model and how it can be incorporated into the existing system.
- The model of the ESUL analyzer is proposed to analyze threats.
- The simulation is done using a *phishing database* library, results are shown, and possible applications of ESUL are discussed.

The rest of the chapter is organized as follows: Section 7.1 formulates the problem, Section 7.2 discusses the model, and Section 7.3 covers simulation analysis and result discussion.

7.1 Problem formulation

The primary motivation is how we can provide a better understanding of threats to end users and enable them to perform incident response activity for precise mitigation of threats. It is necessary, as the number of incidents is growing [10] and the user cannot take appropriate action against it. We explain three antecedents to demonstrate the current severity of threats and the existing gaps in incident handling that urge the integration of a suitable incident response process by end users.

First, Table 7.1 consists of domains associated with different malicious activities. We have the following observations for these domains/URLs: i) The detection rate by popular search engines on virus total (VT) is much lower. ii) These are created regularly and live only for a short duration. For example, the domain ”jeevanpraman.online” was registered in April 2022, updated in December 2022, and became offline within a week. ii) targets a large audience. For instance, the domain email-gov[.]in impersonate the email web page of the Indian government and have the potential to target their users. iii) The cyber security skills of users may vary, hence varying the probability of becoming victims.

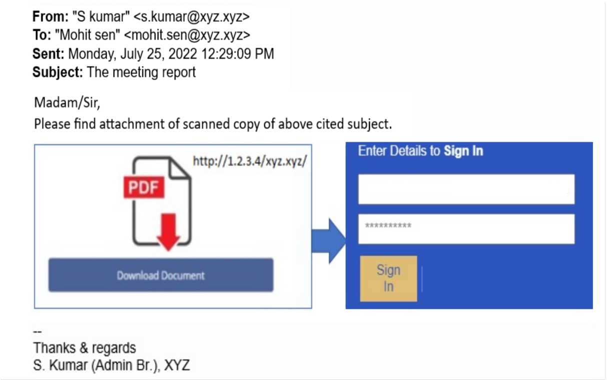
Table 7.1: Examples of detection rate of malicious urls by popular AV engines

Domain	**Target sector**	**Domain creation date**	**Activities**	**detection, status**
jeevanpraman[.] online	Public	14 April 2022	Fraud	0, offline
email-qov[.]in, email-gov.in	Government	19 Sept 2022, 30 July 2020	Spear phishing	0,1, offline
sbiekycs[.]com	Banking	27 Jan 2022	Phishing, Fraud	1,1, offline
kavach-app[.]com, kavach-app[.]in	Government	20 Jan 2022, 13 Dec 2021	Malicious Apps	8,8, offline
armaanapp[.]in	Defense	02 Sep 2021	Malicious Android Apps	4, offline

Second, the severity of the threat may be low to high. For example, in Fig.7-1, a user receives a spear phishing email from a compromised email ID consisting of a PDF file. The PDF file has a phishing URL (say, ”https://1.2.3.4/xyz.xyz/”) requesting user credentials. The situation becomes more severe if the email consists of a malicious executable, therefore, early detection is expected.

Third, large enterprises create a structured network and apply appropriate monitoring at perimeter levels. They also use preventing approaches such as threat sharing (e.g., using MISP [177]) or SIEM alerts [178] to detect threats within their cyberspace. However, the threat may be undetected considering the situation where users work as an individual system and are not part of any monitored infrastructure (Refer, Table 7.1). These individual systems should be appropriately analyzed to detect the incidents as they may also be critical. The criticality increased after COVID-19 when a large number of people were doing work from home.

The above severity of threats motivates us to design a model that can perform stronger URL-based threat detection and incident response. We propose the inclusion of four ingredients:



From: "S kumar" <s.kumar@xyz.xyz>
To: "Mohit sen" <mohit.sen@xyz.xyz>
Sent: Monday, July 25, 2022 12:29:09 PM
Subject: The meeting report

Madam/Sir,
Please find attachment of scanned copy of above cited subject.

--
Thanks & regards
S. Kumar (Admin Br.), XYZ

Figure 7-1: An example of a spear phishing email sent from a compromised email id ”s.kumar@xyz.xyz” impersonating the web page of ”xyz.xyz” hosted at IP address ”1.2.3.4”

7.1.1 Inclusion of information

AVs do not include many common threats (Table 7.1). The same can be included, synchronized, and integrated with users. Such a threat list can be publicly available, verified by a trusted verifier, and updated occasionally. It can include specific details such as *”indicator of compromise(IoCs), possible impact, and an advisory/best practices”* (Table 7.2). The advisory can be a multilingual explanation helpful for ordinary users to understand threats and apply suitable mitigation measures. The synchronization of information will help easily deliver verified known threats and reduce the communication delay in threat sharing.

7.1.2 Leveraging logs

Logs may play a crucial role in incident response proactively and reactively. The efficient use of logs will help in a better understanding of threats. The recent bill of GDPR [74] directs data controllers to protect users’ data. The latest guidelines issued by CERT-In as, *”Directions under sub-section (6) of section 70B of the Information Technology Act, 2000”* [7] also emphasizes storage and analysis of logs for advanced incident handling. Analyzing distinct logs in the system may provide early threat identification and detection. Continuous analysis can help to evaluate the historical data and identify previous mistakes.

7.1.3 Inclusion of self governing incident response

An incident response is a four-stage process [52]. In a qualitative incident handling urge, the victim must take appropriate mitigation measures. At the end of the system, AVs usually generate threat alerts but do not assist users with necessary actions against an incident. For instance, a user downloads a keylogger from a phishing link. If detected, AV may quarantine the malware. However, it does not advise other actions, such as sanitizing the credentials of compromised user's accounts.

We emphasize that if an incident alert is generated, the end user could understand the impact and should be able to perform the requisite response. The advisory discussed in Sec. 7.1.1 should be in such a way that can help in requisite response with available information, even with minimal support.

7.1.4 Assessment mechanism

We include two assessment attributes. First is a *self-review* - where a user can review its alerts to evaluate the mistakes, e.g., *"how many suspicious URLs visited in a specified timeline"*. The second is *feedback*. It is needed if end system users also affiliate with some organizations, such as users working at home but belonging to an organization. In this case, he can share the findings through a defined channel. The organization may evaluate feedback to review the security awareness of a user security posture of the organization or may use this information for further investigation, coordination, and threat sharing.

7.2 ESUL analyzer

Based on the above problem definition, we propose the model of *End System URLs Analyzer (ESUL)*. We assume that many threats are available in the public domain (e.g., over social media, Twitter, web portals, etc.) and are typically not included, recognized/detected by Endpoint detection response (EDR)/AVs. These threats need verification, confirmation, aggregation, and analysis. The model will work to identify these threats, accumulate them, and help minimize their impact. The model has two sections: a) Synchronization of threats and b) Inclusion of ESUL analyzer. The pictorial description is provided in Figure 7-2, and the algorithmic description is given in Figure 7-3.

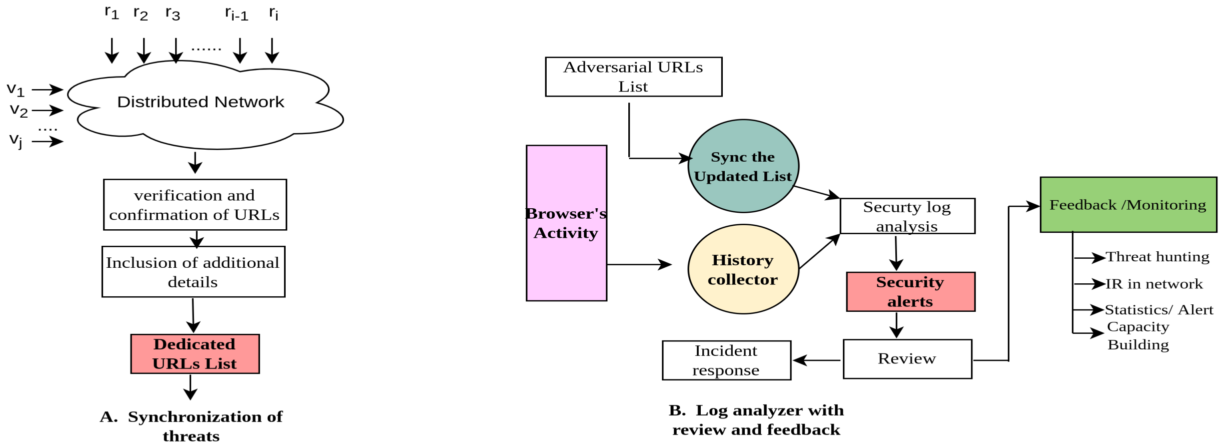


Figure 7-2: Processing steps of ESUL analyzer security threat identification and mitigation

7.2.1 A. Synchronization of threats

In the current system, URL-based threats are not synchronized. Threats are reported to entities such as national level CERTs, ISPs, domain, registrar, and website owners as respective concerned agencies and are taken offline. Once it becomes offline, these URL threats are omitted by EDRs and are not included for detection and analysis purposes. It is possible that many entities are not even aware of these threats (because they are taken offline by one entity and not disclosed to another). Therefore, the list is not synchronized and updated. The second concern is many individuals observe threats; they report them to concerned entities, publishing over social media and blogs. However, these are scattered, unverified, and available only to some entities. For instance, Table 7.2 URLs exist over different portals as partial views. If we accumulate such threats, they may be used to improve incident response for individuals. We propose the model of *synchronization of threats*, refer to the Part A of algorithm 7-3.

Human intelligence and cyber awareness are increasing. People are detecting threats and notifying them. We need a model for aggregating these threats to maximize the benefit. Suppose, for instance, i; multiple reporters $(r_1, r_2, r_3, ..., r_i)$ know some threats and are willing to share. A distributed network N consisting of a list of verified reviewers $(v_1, v_2, v_3, ..., v_j)$ receives new threats. These reviewers are responsible for verifying the URL threats. The reporter r_k selects a URL as url_k appends the relevant indicator of compromise (IoCs) as ioc_k, and their proof of concepts (PoCs) poc_k and shares to the network as threat $threat_k \leftarrow (r_k, url_k, ioc_k, poc_k)$. The IoCs may be additional details such as port number, file hashes, file path, etc. The PoCs may be the screenshots, active status, videos, fraudulent transactions, etc, to support the claim. The threat is shared in the

$$\Pi[List, R, N, V, U, history_U]$$

Part A: Synchronization of threats

i. At instance, i , $(r_1, r_2, r_3, ..., r_i)$ reporters. r_k collects IoCs and corresponding PoCs. r_k submits findings to a distributed networks N. N includes j verifiers $(v_1, v_2, v_3, ..., v_j)$.

ii. N receives threat $threat_k \leftarrow (r_k, url_k, ioc_k, poc_k)$ shared by r_k, and allocate to a verifier v_s. v_s is selected based on work allocation policy.

iii. v_s collects the $threat_k$ and verify it.

iv. if verification successful, v_s performs $temp \leftarrow addition(url_k, ioc_k, description_k, impact_k, mitigation - steps_k)$ operation to add additional parameters with the url.

v. v_s calls aggregate operation and update the $list \leftarrow aggregate(list, temp)$.

Part B : ESUL Analyzer

1. The user u obtains the current processed feeds $list$ from the update channel.
2. Prepare the data structure of the data using $list1 \leftarrow BloomFilter(list)$.
3. a) user u Calls aggregate operation to add the history created after the previous run. $history_u \leftarrow aggregate(history_u, chrome_u, firefox_u, edge_u)$ b)[optional] store $history_u$ at remote location for backup. .
4. user u perform matching operation as: $matching(list1, history_u)$.
5. if matching result identify the detection, then it discloses the result.
6. user u perform the followings: a) perform the review() operation and follow the best practices. b) [optional] user u submits feedback.

Figure 7-3: Algorithmic description of adversarial URLs processing and ESUL analyzer

network. Integration of additional levels of information will be helpful for users to understand the incident effectively and useful to identify the scope of compromise.

When a reviewer v_s receives the list, it verifies the $threat_k$. The reviewer are trusted, and their review may considered correct. This reviewer may be designated CSIRT, CERT, or LEA, etc. They are already working on disabling threats. But here, the additional thing is that, along with mitigation, the teams will work in a coordinated fashion to disclose the URLs and threats in a synchronized format. Another advantage is that they will be designated points of contact to speed up threat sharing.

When an end-user, even with minimum cyber-security knowledge, detects such a threat, several concerns arise, such as i) What are the consequences of threats? ii) What is the impact? iii) what actions are required iv) What are the best practices that need to be implemented to prevent future recurrences of the threat? To enable maximum help, the proposed models include additional parameters with the threat as shown in step iv. The v_s briefly describes the threat, impact, best practices, and mitigation steps. Table 7.2 shows one example of how it can be included. This is a significant step because it will enable end-users to understand the threat and incident. It will help them evaluate logs and appropriate measures. For additional ease, the explanation may also be included in the user's native language.

After including the additional parameters, the $list$ is updated, and new threats are appended. It is a dedicated list, verified, reviewed, with no false positives, and updated occasionally. This was previously not available, especially for short-term-based malicious URLs. The feeds may be shared to all. Individuals may receive such feeds and integrate them with the system.

Table 7.2: Inclusion of description, impact and best practices in threat list alerts

Domain	**Category**	**Description**	**Impact**	**Best Practices**
SBI-kyc.com	Banking, Phishing	The page targets users of SBI and collects debit card information.	The victim's may loose money due to fraud	Don't click or urls and submit details. Refer advisory xxxx for mitigation and best practices.

7.2.2 B. ESUL analyzer

The *ESUL analyzer* is created and runs on the user side. The user u receives the updated list of threats from the synchronized feeds.

The user performs history collection. Since the analyzer runs continuously, it collects the latest history and appends it to the previous list. The history could be from multiple sources such as Google Chrome, Firefox, Microsoft Edge, etc. All the history is appended and created as a single list. As an optional step, the history can be stored at a remote location for backup purposes. This will be helpful in cases where users download and execute malware, and malware can delete the history.

7.2.2.1 Pre-processing dataset using Bloom filter

A bloom filter [75] is a probabilistic data structure used to test whether an element is a set member. The users may have computation constraints. Therefore, it may provide space-efficient searching capability. The received list is processed and stored using the bloom set data structure.

After aggregation, the analyzer matches the $history_u$ with the $list$ and generates the alert if any matching is found along the description, impact, and mitigation steps. The users may review the generated alert, and necessary steps may be taken. As an optional measure, if the review is used as feedback, then the same can be escalated to the concerned entity such as SOC Teams to identify the cyber awareness level of users, how many people are clicking on the URLs, and the type of people who are not able to distinguish the malicious links.

7.3 Implementation

To simulate the results, we have taken a phishing database [19] consisting of a unique set of known phishing URLs and domains. The list does not include active periods and other parameters per our model, but we can use it for simulation. For logs, the history of common known browsers is selected and aggregated to make a single list. The code is simulated on Python ”3.10.10” and executed on the Windows 10 CPU Intel(R) Core(TM) i7-1165G7 @ 2.80GHz 16 GB RAM.

The initial dataset consists of 894028 URLs and 496169 unique domains (as of March 2023). The initial history file consists of 5000 URLs after aggregation. The simulation runs in two modes. 1) when users visit a URL, the list is compared, and an alert is generated. 2) on a specific time (say every 24 hours) with aggregated history. The simulator analyzes the updated history with the latest adversarial list and displays results.

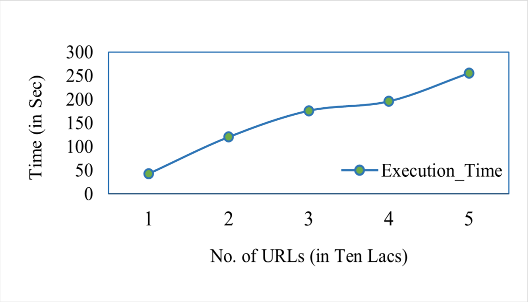


Figure 7-4: Preparation time of phishing URLs data set using bloom filter

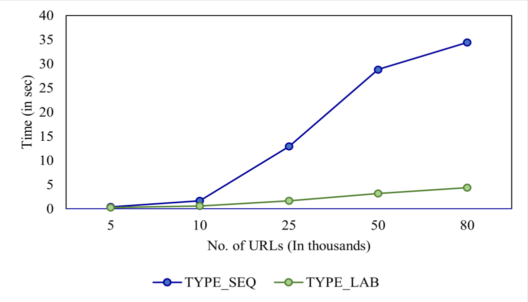


Figure 7-5: Runtime of ESUL analyzer for both Type_Seq and Type_LABF history lists

Fig. 7-4 shows the execution time of preprocessing the data set using the Bloom filter. As phishing URLs increase over time, the URLs are varied from 10 lac to 50 lac. This is done by replicating the initial list of phishing databases and appending indexes to make each URL unique. The execution time of data structure preparation is a few seconds, which is manageable.

Fig. 7-2 shows the analyzer execution time. The analyzer is run on the initial dataset. The history list is varied from 5 thousand to 80 thousand. The variation is used as history will always grow over time. To show the computation time, a comparison is made for a) searching the history list sequentially in the URL list (Type_Seq) and b). they search history lists in processed data structures using bloom filter(Type_LABF). A double verification is implemented for false positives by verifying the alerts in both data sets. The result shows both model executes in a few seconds; however, as urls increase, Type_LABF provide better performance.

Chapter 8

Conclusion and Future Work

Personal data protection is a pivotal responsibility. Organizations strive to fulfil this commitment by implementing robust security and safeguard measures. However, many data fiduciaries are increasingly entangled in the contentious compromising of users' privacy. The aim of data protection could not accomplished correctly due to a lack of regulations and guidelines. The Data Protection Act (DPDPA) and other regulations have been introduced to ensure a privacy-preserving protected data processing. We have studied all draft versions of the bill and the DPDPA act-2023.

We have done a detailed analysis of various obligations such as consent, data collection, data processing, security by design, transparency, and data audit. We have described how cryptographic (such as encryption, signature schemes, zero-knowledge proof, etc.) and other solutions (such as anonymization, de-identification, access control, etc.) can be used for enhanced data processing complying the act. Then, we have further analyzed four obligations: consent, right to nominee, data breach, and storage/logging.

Acquiring user consent poses significant challenges especially in establishing a transparent consent collection mechanism. Following the user's consent obligation of DPDPA, a study on consent, its formalization and standardization was analyzed with a three layer architecture and Proofs of Consent (PoC). The model of Shielded Consent Manager (SCM) is proposed to grant permissions to access android resources. SCM includes parameters as per the framework, satisfies the security properties such as integrity of consent, non-deniability by users, auditability of logs in data processing, and provides finer visualization of user's consents. We conclude that consent management is still at the initial stage and requires more future research, particularly related to security properties.

We explored the right-to-nominee obligations of PDDPA and formalized the DAI system. We have defined its functionalities and security goals to construct DAIP. We proposed a new digital asset inheritance protocol (DAIP) using certificateless encryption. Our newly designed protocol allows a user to create asset information that can be conveyed to the nominee for inheritance. We have designed the protocol with different cryptographic primitives to ensure the correct delivery and inheritance of the asset. With security proofs and performance analysis of the proposed model, we have shown that DAIP can be efficiently integrated with the existing system with a small modification.

We have explored data breach obligations. We have described common causes that may lead to breaches, challenges in their prevention, and issues in data breach incident response. Further, prevention may not always be guaranteed, and a breach may happen. The DPDPA will urge organizations for the right assessment to avoid penalties and to mitigate the existing security gaps. We have explored that a correct breach assessment is not possible without the availability of the necessary details and a breach analysis model. Following this, we have proposed the model of a Data Breach Incident Assessor (DBIA) that help service provider to identify the root cause and enable obligations expected breach assessment. The data breach evaluation systems have a progressive future.

The study also explores how log analyses can contribute to early threat detection within end systems. We have proposed the model of an End System URLs Log (ESUL) analyzer for URL-based threats present in standalone systems. The proposed model with continuous log analysis based on the latest feed may disclose the historical adversity, which some tools might have bypassed. Such modules can integrated independently or incorporated with other modules such as antivirus.

Many times organizations violate users' privacy and are not very concerned about it, but they will be accountable under data protection regulations. The compliance will urge organizations for the right implementation and assessment. Non-compliance not only exposes organizations to punitive measures but also inflicts reputational damage that can be irreparable. The urgency to align with these regulations is a catalyst, prompting organizations to enhance their processing methodologies. The studied work is one step towards solving the above aim. This work will provide a direction to different entities to design a system by implementing DPDPA complying approaches at different levels.

In future work, we aim to explore more advanced methods to solve the existing limitations. A new set of techniques will be analyzed for more effective implementation of major tenets like data consent, privacy by design, data breaches, data audit, etc.

The enhanced methods for data protection frameworks aligned data management and processing have a progressive future and require new researches.

Bibliography

[1] 500,000 Fortinet VPN credentials exposed: Turn off, patch, reset passwords. Available at "https://www.malwarebytes.com/blog/news/2021/09/500000-fortinet-vpn-credentials-exposed-turn-off-patch-reset-passwords".

[2] Aadhar. Available at https://uidai.gov.in/. (Last accessed April 2021).

[3] Apache server logs. Available at "https://httpd.apache.org/docs/2.4/logs.html" (Last accessed August 2023).

[4] Blockchain. Available at "https://www.blockchain.com/" (Last accessed April 2020).

[5] California Consumer Privacy Act (CCPA). Available at https://oag.ca.gov/privacy/ccpa (Last accessed April 2020).

[6] Digital Data Protection Bill (DPDPB-2022). Available at https://www.meity.gov.in/content/digital-personal-data-protection-bill-2022.

[7] Directions under sub-section (6) of section 70B of the Information Technology Act, 2000. Available at "https://cert-in.org.in/PDF/CERT-In_Directions_70B_28.04.2022.pdf".

[8] Fortinet firewall logs. Available at "https://docs.fortinet.com/document/fortigate/7.4.0/administration-guide/738890/log-and-report" (Last accessed August 2023).

[9] HTTP API logging. Available at "https://docs.aws.amazon.com/apigateway/latest/developerguide/http-api-logging.html" (Last accessed August 2023).

[10] India ransomware report. Available at "https://www.csk.gov.in/documents/RANSOMWARE_Report_Final.pdf".

[11] Known exploited vulnerabilities catalog. Available at "https://www.cisa.gov/known-exploited-vulnerabilities-catalog".

[12] Law 78-17 of January 6, 1978, on Information Technologies, Data Files and Civil Liberties (consolidated version as of Aug. 27, 2011). Available at http://www.cnil.fr/fileadmin/documents/en/Act78-17VA.pdf. (Last accessed December 2019).

[13] Malicious Armaan app. Available at "https://blog.cyble.com/2022/01/28/indian-army-personnel-face-remote-access-trojan-attacks/", (Last accessed March 2023).

[14] The many lives of BlackCat ransomware. Available at "https://www.microsoft.com/en-us/security/blog/2022/06/13/the-many-lives-of-blackcat-ransomware/".

[15] Mitigation for china-based threat actor activity. Available at "https://blogs.microsoft.com/on-the-issues/2023/07/11/mitigation-china-based-threat-actor/" (Last accessed August 2023).

[16] Open VPN server logs. Available at "https://openvpn.net/access-server-manual/status-log-reports/" (Last accessed August 2023).

[17] Outrage as personal data of Kashmir University students, teachers 'hacked, put on sale' on internet. Available at "https://www.hindustantimes.com/cities/chandigarh-news/outrage-as-personal-data-of-kashmir-university-students-teachers-hacked-put-on-sale-on-internet-101660157852936.html".

[18] PBC Library: The Pairing-Based Cryptography Library. Available at https://crypto.stanford.edu/pbc/, (Last Accessed December 2021).

[19] Phishing database repository. Available at "https://github.com/mitchellkrogza/Phishing.Database", (Last accessed March 2023).

[20] Preventing data breach/data leak. Available at "https://www.cert-in.org.in/s2cMainServlet?pageid=PUBVLNOTES02&VLCODE=CIAD-2021-0004"(Last accessed August 2023).

[21] The privacy act (1983), Canada. Available at https://www.priv.gc.ca/en/privacy-topics/privacy-laws-in-canada/the-privacy-act/ (Last accessed April 2020).

[22] The privacy act (1988), Australia. Available at https://www.oaic.gov.au/privacy/the-privacy-act/ (Last accessed April 2020).

[23] The privacy act (1993), Newzealand. Available at https://www.privacy.org.nz/the-privacy-act-and-codes/the-privacy-act/ (Last accessed April 2020).

[24] PyCryptodome. Available at https://pycryptodome.readthedocs.io, (Last Accessed December 2021).

[25] Responding to ransomware attacks. Available at "https://www.cert-in.org.in/s2cMainServlet?pageid=PUBVLNOTES02&VLCODE=CIAD-2022-0023" (Last accessed August 2023).

[26] Right to privacy. Available at https://www.sci.gov.in/pdf/LU/ALL_WP(C)_No.494_of_2012_Right_to_Privacy.pdf (Last accessed April 2020).

[27] Sql server error logs. Available at "https://learn.microsoft.com/en-us/sql/database-engine/configure-windows/scm-services-configure-sql-server-error-logs" (Last accessed August 2023).

[28] Steppy kavach attack. Available at "https://www.securonix.com/blog/new-steppykavach-attack-campaign/".

[29] Sysmon. Available at "https://learn.microsoft.com/en-us/sysinternals/downloads/sysmon" (Last accessed August 2023).

[30] Technical analysis of the RedLine Stealer. Available at "https://cloudsek.com/blog/technical-analysis-of-the-redline-stealer".

[31] VT results of malicious Kavach authentication app. Available at "https://www.virustotal.com/gui/domain/kavach-app.in", (Last accessed March 2023).

[32] Hassan Adamu, Abdullahi Adamu Ahmad, A Hassan, and SB Gambasha. Web browser forensic tools: Autopsy, BHE and net analysis. *Int. J. Res. Innov. Appl. Sci*, 6(5):103–107, 2021.

[33] Rachit Agarwal, Tarek Elsaleh, and Elias Tragos. GDPR-inspired IoT ontology enabling semantic interoperability, federation of deployments and privacy-preserving applications. *arXiv preprint arXiv:2012.10314*, 2020.

[34] Sattam S. Al-Riyami and Kenneth G. Paterson. Certificateless public key cryptography. In Chi-Sung Laih, editor, *Advances in Cryptology - ASIACRYPT 2003, 9th International Conference on the Theory and Application of Cryptology and Information Security*, volume 2894 of *Lecture Notes in Computer Science*, pages 452–473. Springer Berlin Heidelberg, 2003.

[35] Mahathir Almashor, Ejaz Ahmed, Benjamin Pick, Sharif Abuadbba, Raj Gaire, Seyit Camtepe, and Surya Nepal. Characterizing malicious url campaigns. *arXiv preprint arXiv:2108.12726*, 2021.

[36] Jane Andrew, Max Baker, and Casey Huang. Data breaches in the age of surveillance capitalism: do disclosures have a new role to play? *Critical Perspectives on Accounting*, page 102396, 2021.

[37] Natalie M Banta. Inherit the cloud: The role of private contracts in distributing or deleting digital assets at death. *Fordham Law Review*, 83:799–854, 2014.

[38] Mehdi Barati and Benjamin Yankson. Predicting the occurrence of a data breach. *International Journal of Information Management Data Insights*, 2(2):100128, 2022.

[39] D. Basin and H. Thomas. On purpose and by necessity: Compliance under the GDPR. In *Financial Cryptography and Data Security*, pages 20–37. Springer, 2018.

[40] Anna Berlee. Digital inheritance in the Netherlands. *Journal of European Consumer and Market Law (EuCML)*, pages 256–260, 2017.

[41] T. Bertram, E. Bursztein, S. Caro, H. Chao, R.C. Feman, P. Fleischer, A. Gustafsson, J. Hemerly, C. Hibbert, L. Invernizzi, L.K. Donnelly, J. Ketover, J. Laefer, P. Nicholas, Y. Niu, H. Obhi, D. Price, K. Thomas A. Strait, and A. Verney. Five years of the right to be forgotten. In *ACM SIGSAC Conference on Computer and Communications Security (CCS'19)*, 2019.

[42] Marzieh Bitaab, Haehyun Cho, Adam Oest, Penghui Zhang, Zhibo Sun, Rana Pourmohamad, Doowon Kim, Tiffany Bao, Ruoyu Wang, Yan Shoshitaishvili, Adam Doupé, and Gail-Joon Ahn. Scam pandemic: How attackers exploit public fear through phishing. In *2020 APWG Symposium on Electronic Crime Research (eCrime)*, pages 1–10, 2020.

[43] Michael Bitzer, Björn Häckel, Daniel Leuthe, Joshua Ott, Bastian Stahl, and Jacqueline Strobel. Managing the Inevitable – A maturity model to establish incident response management capabilities. *Computers & Security, ELSEVIER*, 125:103050, 2023.

[44] Dino Bollinger, Karel Kubicek, Carlos Cotrini, and David Basin. Automating cookie consent and {GDPR} violation detection. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 2893–2910, 2022.

[45] Kerstin Bongard-Blanchy, Arianna Rossi, Salvador Rivas, Sophie Doublet, Vincent Koenig, and Gabriele Lenzini. " i am definitely manipulated, even when i am aware of it. it's ridiculous!"-dark patterns from the end-user perspective. In *Designing Interactive Systems Conference 2021*, pages 763–776, 2021.

[46] Stephen Breen, Karim Ouazzane, and Preeti Patel. GDPR: Is your consent valid? *Business Information Review*, 37(1):19–24, 2020.

[47] Jed R Brubaker and Vanessa Callison-Burch. Legacy contact: Designing and implementing post-mortem stewardship at Facebook. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 2908–2919. ACM, 2016.

[48] Ran Canetti. Security and composition of multiparty cryptographic protocols. *Journal of Cryptology*, 13(1):143–202, 2000.

[49] Claude Castelluccia, Mathieu Cunche, Daniel Le Metayer, and Victor Morel. Enhancing transparency and consent in the iot. In *2018 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, pages 116–119. IEEE, 2018.

[50] A. Cavoukian. Privacy by design: The 7 foundational principles. *Information and Privacy Commissioner of Ontario, Canada*, 2009.

[51] Rahul Chatterjee, Joseph Bonneau, Ari Juels, and Thomas Ristenpart. Cracking-resistant password vaults using natural language encoders. In *2015 IEEE Symposium on Security and Privacy*, pages 481–498. IEEE, 2015.

[52] Paul Cichonski, Tom Millar, Tim Grance, Karen Scarfone, et al. Computer security incident handling guide. *NIST Special Publication*, 800(61):1–147, 2012.

[53] John Conner. Digital life after death: The issue of planning for a person's digital assets after death. *Estate Planning & Community Property Law Journal*, 3:301–324, 2010.

[54] SET Consortium. Secure electronic transaction (SET). Available at http://www.maithean.com/docs/set_bk1.pdf (Last accessed April 2020).

[55] Ivan Damgård. A design principle for hash functions. In Gilles Brassard, editor, *Advances in Cryptology - CRYPTO '89, 9th Annual International Cryptology Conference*, volume 435 of *Lecture Notes in Computer Science*, pages 416–427. Springer, 1989.

[56] Laurens Debackere, Pieter Colpaert, Ruben Taelman, and Ruben Verborgh. A policy-oriented architecture for enforcing consent in solid. In *Companion Proceedings of the Web Conference 2022*, pages 516–524, 2022.

[57] Christophe Debruyne, Harshvardhan J Pandit, Dave Lewis, and Declan O'Sullivan. "just-in-time" generation of datasets by considering structured representations of given consent for gdpr compliance. *Knowledge and Information Systems*, 62(9):3615–3640, 2020.

[58] M. Degeling, C. Utz, C. Lentzsch, H. Hosseini, F. Schaub, and T. Holz. We value your privacy ... now take some cookies: Measuring the GDPR's impact on web privacy. In *Network and Distributed Systems Security (NDSS) Symposium*, 2019.

[59] Chandramohan Dhasarathan, Mohammad Kamrul Hasan, Shayla Islam, Salwani Abdullah, Umi Asma Mokhtar, Abdul Rehman Javed, and Sam Goundar. Covid-19 health data analysis and personal data preserving: A homomorphic privacy enforcement approach. *Computer communications*, 199:87–97, 2023.

[60] Whitfield Diffie and Martin Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.

[61] Yana Dimova, Gertjan Franken, Victor Le Pochat, Wouter Joosen, and Lieven Desmet. Tracking the evolution of cookie-based tracking on facebook. In *Proceedings of the 21st Workshop on Privacy in the Electronic Society*, pages 181–196, 2022.

[62] DPDPA. The Digital Personal Data Protection Act, 2023. Available at https://www.meity.gov.in/writereaddata/files/Digital%20Personal%20Data%20Protection%20Act%202023.pdf. (Last accessed August 2023).

[63] Olha Drozd and Sabrina Kirrane. Privacy cure: Consent comprehension made easy. In *IFIP International Conference on ICT Systems Security and Privacy Protection*, pages 124–139. Springer, 2020.

[64] Stefan Dziembowski, Sebastian Faust, and Kristina Hostáková. General state channel networks. CCS '18, page 949–966, 2018.

[65] P.F. Edemekong and M.J. Haydel. Health insurance portability and accountability act (HIPAA). In *StatPearls [Internet]*. StatPearls Publishing, 2019.

[66] A. Esteve. The business of personal data: Google, Facebook, and privacy issues in the EU and the USA. *International Data Privacy Law*, 7(1):36–47, 2017.

[67] Ethereum. Smart contract. Available at.https://www.ethereum.org/. Last accessed April 2020.

[68] Zijian Fang, Maochao Xu, Shouhuai Xu, and Taizhong Hu. A framework for predicting data breach risk: Leveraging dependence to cope with sparsity. *IEEE Transactions on Information Forensics and Security*, 16:2186–2201, 2021.

[69] Florian M Farke, David G Balash, Maximilian Golla, Markus Dürmuth, and Adam J Aviv. Are privacy dashboards good for end users? Evaluating user perceptions and reactions to Google's my activity. In *USENIX Security Symposium*, pages 483–500, 2021.

[70] Kaniz Fatema, Ensar Hadziselimovic, Harshvardhan J Pandit, Christophe Debruyne, Dave Lewis, and Declan O'Sullivan. Compliance through informed consent: Semantic based consent permission and data management model. In *PrivOn@ ISWC*, 2017.

[71] Flipkart. Saved card details. Available at https://www.flipkart.com/account/carddetails (Last accessed April 2020).

[72] Noran Shafik Fouad. Securing higher education against cyberthreats: from an institutional risk to a national policy challenge. *Journal of Cyber Policy*, 6(2):137–154, 2021.

[73] C.S. Fuller. Is the market for digital privacy a failure? *Public Choice, Springer*, 180(3):353–381, 2019.

[74] GDPR-EU. General Data Protection Regulation-European Union. Available at https://eugdpr.org/.

[75] Shahabeddin Geravand and Mahmood Ahmadi. Bloom filter applications in network security: A state-of-the-art survey. *Computer Networks*, 57(18):4047–4064, 2013.

[76] Alexandra Giannopoulou. Algorithmic systems: the consent is in the detail? *Internet Policy Review*, 9(1):1–19, 2020.

[77] Andrew Gilden. The social afterlife. *Harvard Journal of Law & Technology, Forthcoming*, 2019.

[78] O. Gkotsopoulou, E. Charalambous, K. Limniotis, P. Quinn, D. Kavallieros, and G. Sargsyan. Data protection by design for cybersecurity systems in a smart home environment. In *IEEE Conference on Network Softwarization (NetSoft)*, pages 101–109. IEEE, 2019.

[79] Oded Goldreich. *Foundations of cryptography: Volume 1, basic tools*. Cambridge university press, 2007.

[80] Shafi Goldwasser, Silvio Micali, and Ronald L Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal on Computing*, 17(2):281–308, 1988.

[81] PAJ Graßl, HK Schraffenberger, FJ Zuiderveen Borgesius, and MA Buijzen. Dark and bright patterns in cookie consent requests. 2021.

[82] G. Greenleaf. Data protection: A necessary part of india's fundamental inalienable right of privacy – submission on the white paper of the committee of experts on a data protection framework for india. *UNSW Law Research*, (18-6), 2018.

[83] N. Gruschka, V. Mavroeidis, K. Vishi, and M. Jensen. Privacy issues and data protection in big data: A case study analysis under GDPR. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 5027–5033. IEEE, 2018.

[84] Hana Habib, Megan Li, Ellie Young, and Lorrie Cranor. "okay, whatever": An evaluation of cookie consent interfaces. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pages 1–27, 2022.

[85] Hana Habib, Yixin Zou, Yaxing Yao, Alessandro Acquisti, Lorrie Cranor, Joel Reidenberg, Norman Sadeh, and Florian Schaub. Toggles, dollar signs, and triangles: How to (in) effectively convey privacy choices with icons and link texts. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–25, 2021.

[86] Hicham Hammouchi, Othmane Cherqi, Ghita Mezzour, Mounir Ghogho, and Mohammed El Koutbi. Digging deeper into data breaches: An exploratory data analysis of hacking breaches over time. *Procedia Computer Science*, 151:1004–1009, 2019.

[87] Edina Harbinja. Digital inheritance in the United Kingdom. *The Journal of European Consumer and Market Law (EuCML)*, 2017.

[88] Edina Harbinja. Post-mortem privacy 2.0: Theory, law, and technology. *International Review of Law, Computers & Technology*, 31(1):26–42, 2017.

[89] Yasasvi Hari, Rohit Singh, Kizito Nyuytiymbiy, and David Butera. Consenting to internet of things across different social settings. *arXiv preprint arXiv:2102.09499*, 2021.

[90] Ying He, Efpraxia D Zamani, Stefan Lloyd, and Cunjin Luo. Agile incident response (AIR): Improving the incident response process in healthcare. *International Journal of Information Management*, 62:102435, 2022.

[91] Maximilian Hils, Daniel W Woods, and Rainer Böhme. Measuring the emergence of consent management on the web. In *Proceedings of the ACM Internet Measurement Conference*, pages 317–332, 2020.

[92] K. Hjerppe, J. Ruohonen, and V. Leppanen. The general data protection regulation: Requirements, architectures, and constraints. Available at https://arxiv.org/abs/1907.07498, 2019.

[93] Jamie P Hopkins. Afterlife in the cloud: Managing a digital estate. *Hastings Science & Technology Law Journal*, 5:209, 2013.

[94] David Horton. Tomorrow's inheritance: The frontiers of estate planning formalism. *BCL Review*, 58:539, 2017.

[95] Xuehui Hu, Nishanth Sastry, and Mainack Mondal. Cccc: Corralling cookies into categories with cookiemonster. In *13th ACM Web Science Conference 2021*, pages 234–242, 2021.

[96] Soheil Human, Harshvardhan J Pandit, Victor Morel, Cristiana Santos, Martin Degeling, Arianna Rossi, Wilhelmina Botes, Vitor Jesus, and Irene Kamara. Data protection and consenting communication mechanisms: Current open proposals and challenges. In *2022 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, pages 231–239. IEEE, 2022.

[97] Jane Im, Jill Dimond, Melody Berton, Una Lee, Katherine Mustelier, Mark S Ackerman, and Eric Gilbert. Yes: Affirmative consent as a theoretical framework for understanding and imagining social platforms. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–18, 2021.

[98] InfoSecurity. GoDaddy suffers data breach. Available at https://www.infosecurity-magazine.com/news/godaddy-suffers-data-breach/ (Last accessed May 2020).

[99] Vitor Jesus and Harshvardhan J Pandit. Consent receipts for a usable and auditable web of personal data. *IEEE Access*, 10:28545–28563, 2022.

[100] Freeha Khan, Jung Hwan Kim, Lars Mathiassen, and Robin Moore. Data breach management: An integrated risk model. *Information & Management*, 58(1):103392, 2021.

[101] Rishabh Khandelwal, Asmit Nayak, Hamza Harkous, and Kassem Fawaz. Cookieenforcer: Automated cookie notice analysis and enforcement. *arXiv preprint arXiv:2204.04221*, 2022.

[102] Atika Khurana, Amy Bleakley, Amy B Jordan, and Daniel Romer. The protective effects of parental monitoring and internet restriction on adolescents' risk of online harassment. *Journal of youth and Adolescence*, 44(5):1039–1047, 2015.

[103] Dubravka Klasiček. What happens to your Gmail and Facebook account after you die? *Proceeding of Economics of Digital Transformation Conference*, pages 37–56, 2018.

[104] Helen Knapman. What happens to your digital assets when you die? Available at https://www.moneywise.co.uk/work/everyday-life/what-happens-your-digital-assets-when-you-die (Last accessed April 2021), 2017.

[105] Kravitz and W. David. Digital signature algorithm, 1993. US Patent 5,231,668.

[106] Michael Kretschmer, Jan Pennekamp, and Klaus Wehrle. Cookie banners and privacy policies: Measuring the impact of the gdpr on the web. *ACM Transactions on the Web (TWEB)*, 15(4):1–42, 2021.

[107] Karel Kubicek, Jakob Merane, Carlos Cotrini, Alexander Stremitzer, Stefan Bechtold, and David Basin. Checking websites' gdpr consent compliance for marketing emails. *Proceedings on Privacy Enhancing Technologies*, 2022(2):282–303, 2022.

[108] Noam Kutler. Protecting your online you: A new approach to handling your online persona after death. *Berkeley Technology Law Journal*, 26:1641, 2011.

[109] Goo Yeon Lee, Kyung Jin Cha, and Hwa Jong Kim. Designing the GDPR compliant consent procedure for personal information collection in the IoT environment. In *2019 IEEE International Congress on Internet of Things (ICIOT)*, pages 79–81. IEEE, 2019.

[110] Liyuan Liu, Meng Han, Yan Wang, and Yiyun Zhou. Understanding data breach: A visualization aspect. In *Wireless Algorithms, Systems, and Applications: 13th International Conference, WASA 2018, Tianjin, China, June 20-22, 2018, Proceedings 13*, pages 883–892. Springer, 2018.

[111] Juan Miguel López Velásquez, Sergio Mauricio Martínez Monterrubio, Luis Enrique Sánchez Crespo, and David Garcia Rosado. Systematic review of SIEM technology: SIEM-SC birth. *International Journal of Information Security*, 22(3):691–711, 2023.

[112] Chaoyi Lu, Baojun Liu, Yiming Zhang, Zhou Li, Fenglu Zhang, Haixin Duan, Ying Liu, Joann Qiongna Chen, Jinjin Liang, Zaifeng Zhang, et al. From WHOIS to WHOWAS: A large-scale measurement study of domain registration privacy under the GDPR. In *NDSS*, 2021.

[113] May O Lwin, Andrea JS Stanaland, and Anthony D Miyazaki. Protecting children's privacy online: How parental mediation strategies affect website safeguard effectiveness. *Journal of Retailing*, 84(2):205–217, 2008.

[114] Eryn Ma and Eleanor Birrell. Prospective consent: The effect of framing on cookie consent decisions. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts*, pages 1–6, 2022.

[115] Dominique Machuletz and Rainer Böhme. Multiple purposes, multiple problems: A user study of consent dialogs after GDPR. In *Proceedings on Privacy Enhancing Technologies*, pages 481–498, 2019.

[116] A. Madhukalya. Data theft in delhi man's PAN card details stolen. Available at https://www.businesstoday.in/current/economy-politics/delhi-man-pan-number-misused-made-director-of-13-firms/story/281665.html" (Last accessed July 2019).

[117] Stefan Mager and Johann Kranz. Consent notices and the willingness-to-sell observational data: Evidence from user reactions in the field. In *ECIS*, 2021.

[118] Sweta Mahaju and Travis Atkison. Evaluation of firefox browser forensics tools. In *Proceedings of the SouthEast Conference*, pages 5–12, 2017.

[119] Nathan Manworren, Joshua Letwat, and Olivia Daily. Why you should care about the target data breach. *Business Horizons*, 59(3):257–266, 2016.

[120] Damien McCallig. Facebook after death: An evolving policy in a social network. *International Journal of Law and Information Technology*, 22(2):107–140, 2014.

[121] J. Mccarthy. Privacy is fundamental right. Available at https://www.npr.org/sections/thetwo-way/2017/08/24/545963181/indian-supreme-court-declares-privacy-a-fundamental-right (Last accessed April 2020).

[122] Laura McCarthy. Digital assets and intestacy. *BUJ Science & Technology Law Journal*, 21:384, 2015.

[123] Maryam Mehrnezhad. A cross-platform evaluation of privacy notices and tracking practices. In *2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, pages 97–106. IEEE, 2020.

[124] Neerja Mhaskar, Mohammed Alabbad, and Ridha Khedri. A formal approach to network segmentation. *Computers & Security*, 103:102162, 2021.

[125] Tiina Mikk and Karin Sein. Digital inheritance: Heirs' right to claim access to online accounts under Estonian law. *Juridica International*, 27:117, 2018.

[126] Abbas Mirshekari, Ramin Ghasemi, and Ali Abedi. Inheritance of digital accounts. *National Journal of Cyber Security Law*, 3(1), 2020.

[127] Montrel D Morgan, Md Minhaz Chowdhury, and Shadman Latif. Protecting business from data breach. In *2021 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*, pages 1–5. IEEE, 2021.

[128] Apurva Nalawade, Smita Bharne, and Vanita Mane. Forensic analysis and evidence collection for web browser activity. In *2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT)*, pages 518–522, 2016.

[129] F. Navarro. Apps are selling your location data without your knowledge. Available at https://www.komando.com/happening-now/519770/popular-smartphone-apps-are-selling-your-location-data-without-your-knowledge (Last accessed April 2020).

[130] Midas Nouwens, Ilaria Liccardi, Michael Veale, David Karger, and Lalana Kagal. Dark patterns after the GDPR: Scraping consent pop-ups and demonstrating their influence. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–13, 2020.

[131] Junhyoung Oh, Jinhyoung Hong, Changsoo Lee, Jemin Justin Lee, Simon S Woo, and Kyungho Lee. Will eu's gdpr act as an effective enforcer to gain consent? *IEEE Access*, 9:79477–79490, 2021.

[132] Carl Öhman and Luciano Floridi. The political economy of death in the age of information: A critical approach to the digital afterlife industry. *Minds and Machines*, 27(4):639–662, 2017.

[133] Yvonne O'Connor, Wendy Rowan, Laura Lynch, and Ciara Heavin. Privacy by design: informed consent and internet of things for smart health. *Procedia computer science*, 113:653–658, 2017.

[134] Barbara O'Neill. Document your digital assets. Available at https://njaes.rutgers.edu/sshw/message/message.php?p=Finance&m=338 (Last accessed April 2021), 2016.

[135] Harshvardhan J Pandit. Proposals for resolving consenting issues with signals and user-side dialogues. *arXiv preprint arXiv:2208.05786*, 2022.

[136] Harshvardhan J Pandit, Christophe Debruyne, Declan O'Sullivan, and Dave Lewis. GConsent-a consent ontology based on the gdpr. In *European Semantic Web Conference*, pages 270–282. Springer, 2019.

[137] Emmanouil Papadogiannakis, Panagiotis Papadopoulos, Nicolas Kourtellis, and Evangelos P Markatos. User tracking in the post-cookie era: How websites bypass GDPR consent to track users. In *Proceedings of the 2019 The Web Conference 2021 (WWW 2021)*, 2021.

[138] Yong Jin Park, Yoonmo Sang, Hoon Lee, and S Mo Jones-Jang. The ontology of digital asset after death: Policy complexities, suggestions and critique of digital platforms. *Digital Policy, Regulation and Governance*, 2019.

[139] V.T. Patil and R.K. Shyamasundar. Efficacy of GDPR's right-to-be-forgotten on Facebook. In *Information Systems Security*, pages 364–385. Springer, 2018.

[140] Francesco Paolo Patti and Francesca Bartolini. Digital inheritance and post mortem data protection: The Italian reform. *European Review of Private Law (ERPL), Forthcoming*, 2019.

[141] PDPB. The Personal Data Protection Bill, 2019. Available at http://164.100.47.4/BillsTexts/LSBillTexts/Asintroduced/373_2019_LS_Eng.pdf. (Last accessed April 2020).

[142] Paulina Jo Pesch. Drivers and obstacles for the adoption of consent management solutions by ad-tech providers. In *2021 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, pages 269–277. IEEE, 2021.

[143] Check Point. Cyber attack trends: 2022 mid-year report. Available at "https://pages.checkpoint.com/cyber-attack-2022-trends.html" , (Last accessed March 2023).

[144] E. Politou, F. Casino, E. Alepis, and C. Patsakis. Blockchain mutability: Challenges and proposed solutions. Available at https://arxiv.org/abs/1907.07099, 2019.

[145] Konstantinos Rantos, George Drosatos, Konstantinos Demertzis, Christos Ilioudis, and Alexandros Papanikolaou. Blockchain-based consents management for personal data processing in the iot ecosystem. In *ICETE (2)*, pages 738–743, 2018.

[146] Sven Carsten Rasmusen, Manuel Penz, Stephanie Widauer, Petraq Nako, Anelia Kurteva, Antonio Roa-Valverde, and Anna Fensel. Raising consent awareness with gamification and knowledge graphs: an automotive use case. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 18(1):1–21, 2022.

[147] Aamir Rasool and Zunera Jalil. A review of web browser forensic analysis tools and techniques. *Researchpedia Journal of Computing*, 1(1):15–21, 2020.

[148] Giorgio Resta. Personal data and digital assets after death: A comparative law perspective on the BGH Facebook ruling. *Journal of European Consumer and Market Law*, 7(5), 2018.

[149] Joseph Ronderos. Is access enough: Addressing inheritability of digital assets using the three-tier system under the revised uniform fiduciary access to digital assets act. *Transactions: The Tennesse Journal of Business Law*, 18:1031, 2016.

[150] Cristiana Santos, Midas Nouwens, Michael Toth, Nataliia Bielova, and Vincent Roca. Consent management platforms under the GDPR: processors and/or controllers? In *Privacy Technologies and Policy: 9th Annual Privacy Forum, APF 2021, Oslo, Norway, June 17–18, 2021, Proceedings*, pages 47–69. Springer, 2021.

[151] Yogesh Sapkale. What you should know about digital inheritance. https://www.moneylife.in/article/what-you-should-know-about-digital-inheritance/56329.html, (Last accessed October 2020), 2020.

[152] Frederic Schlackl, Nico Link, and Hartmut Hoehle. Antecedents and consequences of data breaches: A systematic review. *Information & Management*, 59(4):103638, 2022.

[153] Patrick Scolyer-Gray, Arash Shaghaghi, and Debi Ashenden. Digging your own digital grave: How should you manage the data you leave behind? Available at `https://theconversation.com/digging-your-own-digital-grave-how-should-you-manage-the-data-you-leave-behind-143755`, 2020. (Last accessed April 2021).

[154] H. Shafaghand L. Burkhalter, A. Hithnawi, and S. Duquennoy. Towards blockchain-based auditable storage and sharing of IoT data. In *Cloud Computing Security Workshop*, CCSW '17, pages 45–50. ACM, 2017.

[155] Faheem Ahmed Shaikh and Mikko Siponen. Information security risk assessments following cybersecurity breaches: The mediating role of top management attention to cybersecurity. *Computers & Security*, 124:102974, 2023.

[156] S. Shastri, M. Wasserman, and V. Chidambaram. How design, architecture, and operation of modern systems conflict with GDPR. *USENIX HotCloud*, 2019.

[157] Ram Govind Singh and Sushmita Ruj. A technical look at the Indian personal data protection bill. *arXiv*, page 2005.13812, 2020.

[158] Sudeep Singh. APT-36 uses new TTPs and new tools to target Indian governmental organizations. Available at `"www.zscaler.com/blogs/security-research/apt-36-uses-new-ttps-and-new-tools-target-indian-governmental-organizations"`.

[159] L. Sion, P. Dewitte, D. Van Landuyt, K. Wuyts, I. Emanuilov, P. Valcke, and W. Joosen. An architectural view for data protection by design. In *IEEE International Conference on Software Architecture (ICSA)*, pages 11–20. IEEE, March 2019.

[160] O. Solon. Facebook says cambridge analytica may have gained 37M more users' data (In the Guardian). Available at `https://www.theguardian.com/technology/2018/apr/04/facebook-cambridge-analytica-user-data-latest-more-thanthought` (Last accessed April 2020).

[161] J. Sorensen and S. Kosta. Before and after GDPR: The changes in third party presence at public and private european websites. In *The World Wide Web Conference (WWW'19)*, pages 1590–1600. ACM, 2019.

[162] Hong Sun, Maochao Xu, and Peng Zhao. Modeling malicious hacking data breach risks. *North American Actuarial Journal*, 25(4):484–502, 2021.

[163] Aihik Sur. AIIMS-like ransomware attacks will continue unless there is proper cyber hygiene: Google. Available at `"https://www.moneycontrol.com/news/business/aiims-like-ransomware-attacks-will-continue-unless-there-is-proper-cyber-hygiene-google-9730411.html"`.

[164] Jan Svacina, Jackson Raffety, Connor Woodahl, Brooklynn Stone, Tomas Cerny, Miroslav Bures, Dongwan Shin, Karel Frajtak, and Pavel Tisnovsky. On vulnerability and security log analysis: A systematic literature review on recent trends. In *Proceedings of the International Conference on Research in Adaptive and Convergent Systems*, RACS '20, page 175–180, 2020.

[165] DigiLocker Team. DigiLocker. Available at https://digilocker.gov.in/, (Last accessed April 2021), 2015.

[166] DigiPulse Team. DigiPulse: Digital inheritance service. Available at https://cryptorating.eu/whitepapers/DigiPulse/whitepaper.pdf, (Last Accessed April 2021), 2018.

[167] MoneyLife Digital Team. Unclaimed assets with financial regulators. Available at https://www.moneylife.in/article/a-whopping-rs36000-crore-of-peoples-unclaimed- money-lying-with-just-three-financial-regulators/57587.html, (Last Accessed April 2021), 2019.

[168] PassOn Team. PassOn:Inventing digital inheritance. Available at https://passon.com/content/home/modules/1-intro-1af8iji/passon-white-paper.pdf, (Last accessed April 2021), 2019.

[169] SafeHaven Team. Safe Haven: The Solution to digital inheritance. Available at https://safehaven.io/files/SafeHaven_WhitePaper.pdf, (Last access April 2021), 2019.

[170] TrustVerse Team. TrustVerse:AI-wealth management & digital estate planning protocol backed by blockchain. Available at https://icosbull.com/eng/ico/trustverse/whitepaper, (Last accessed April 2021), 2018.

[171] Nguyen Duc Thai and Nguyen Huu Hieu. A framework for website security assessment. In *Proceedings of the 7th International Conference on Computer and Communications Management*, pages 153–157, 2019.

[172] Manisekaran Thangavelu, Venkataraghavan Krishnaswamy, and Mayank Sharma. Impact of comprehensive information security awareness and cognitive characteristics on security incident management – an empirical study. *Computers & Security,ELSEVIER*, 109:102401, 2021.

[173] Business Today. PAN card must for Rs 50K cash payments to settle hotel bills. Available at https://www.businesstoday.in/money/tax/pan-must-for-rs-50k-cash-to-settle-hotel-bills-foreign-airfare/story/227176.html (Last accessed April 2020).

[174] Nguyen Binh Truong, Kai Sun, Gyu Myoung Lee, and Yike Guo. GDPR-compliant personal data management: A blockchain-based solution. *IEEE Transactions on Information Forensics and Security*, 15:1746–1761, 2019.

[175] C. Utz, M. Degeling, S. Fahl, F. Schaub, and T. Holz. (Un)informed consent: Studying GDPR consent notices in the field. In *SIGSAC Conference on Computer and Communications Security (CCS'19)*. ACM, 2019.

[176] Juan Camilo Vargas. Blockchain-based consent manager for gdpr compliance. page 165, 2019.

[177] Cynthia Wagner, Alexandre Dulaunoy, Gérard Wagener, and Andras Iklody. MISP: The design and implementation of a collaborative threat intelligence sharing platform. In *Proceedings of the 2016 ACM on Workshop on Information Sharing and Collaborative Security*, pages 49–56, 2016.

[178] Thomas D Wagner, Khaled Mahbub, Esther Palomar, and Ali E Abdallah. Cyber threat intelligence sharing: Survey and research directions. *Computers & Security*, 87:101589, 2019.

[179] Michael D Walker. The new uniform digital assets law: estate planning and administration in the information age. *Real Property, Trust and Estate Law Journal*, 52(1):51–78, 2017.

[180] C. Wang, S.M. Chow, Q. Wang, K. Ren, and W.Lou. Privacy-preserving public auditing for secure cloud storage. *IEEE Transactions on Computers*, 62(2):362–375, Feb 2013.

[181] Christian Wirth and Michael Kolain. Privacy by blockchain design: a blockchain-enabled GDPR-compliant approach for handling personal data. In *Proceedings of 1st ERCIM Blockchain Workshop 2018*. European Society for Socially Embedded Technologies (EUSSET), 2018.

[182] Daniel W Woods and Rainer Böhme. The commodification of consent. *Computers & Security*, 115:102605, 2022.

[183] Maochao Xu and Quynh Nhu Nguyen. Statistical modeling of data breach risks: Time to identification and notification. *arXiv preprint arXiv:2209.07306*, 2022.

[184] Min Yang, Liuyan Tan, Xingshu Chen, Yonggang Luo, Zhenwu Xu, and Xiao Lan. Laws and regulations tell how to classify your data: A case study on higher education. *Information Processing & Management*, 60(3):103240, 2023.

[185] X. Yue, H. Wang, M. Li D. Jin, and W.Jiang. Healthcare data gateways: Found healthcare intelligence on blockchain with novel privacy risk control. *Journal of Medical Systems, ACM*, 40:1–8, 2016.

[186] G. Zyskind, O. Nathan, and A.Pentland. Decentralizing privacy: Using blockchain to protect personal data. In *IEEE Security and Privacy Workshops*, pages 180–184. IEEE, 2015.