PRESERVING PRIVACY OF HEALTH DATA RESIDING IN HL7 FHIR REPOSITORIES THROUGH DE-IDENTIFICATION

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

EZELSU ŞIMŞEK YILGIN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

JANUARY 2022

Approval of the thesis:

PRESERVING PRIVACY OF HEALTH DATA RESIDING IN HL7 FHIR REPOSITORIES THROUGH DE-IDENTIFICATION

submitted by **EZELSU ŞIMŞEK YILGIN** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Halit Oğuztüzün
Head of Department, **Computer Engineering** _____

Prof. Dr. Pınar Karagöz
Supervisor, **Computer Engineering, METU** _____

Examining Committee Members:

Prof. Dr. Nihan Kesim Çiçekli
Computer Engineering, METU _____

Prof. Dr. Pınar Karagöz
Computer Engineering, METU _____

Assist. Prof. Dr. Engin Demir
Computer Engineering, Hacettepe University _____

Date: 28.01.2022

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Ezelsu Şimşek Yılgın

Signature :

ABSTRACT

PRESERVING PRIVACY OF HEALTH DATA RESIDING IN HL7 FHIR REPOSITORIES THROUGH DE-IDENTIFICATION

Şimşek Yılgın, Ezelsu
M.S., Department of Computer Engineering
Supervisor: Prof. Dr. Pınar Karagöz

January 2022, 65 pages

Collaboration and data sharing are essential aspects of health research. Nevertheless, the number of sensitive health data breaches is increasing and there is a significant need to ensure that the privacy of patients is preserved. Health data accumulated in different repositories can be useful for statistical analysis, data mining and machine learning tasks; which results in long-term value for both healthcare professionals and patients. Preserving the privacy and ensuring the security is essential while coping with the distributed nature of health data during clinical research. Data de-identification and anonymization techniques are highly beneficial for protecting patients data against privacy risks.

In this thesis, de-identification and anonymization of health data existing in HL7 FHIR repositories has been studied to ensure the privacy protection. This work presents the development of Data Privacy Tool, which includes a novel technique for de-identification of HL7 FHIR data, as well as provides a graphical user interface. The study also includes assessment of the outcomes from a privacy point of view comparing various de-identification techniques. This study has examined the existing algorithms for de-identification of health data and proposed an efficient method-

ology to develop the privacy preservation layer. The results of this study are analyzed through several experiments. This study aims to contribute to a research project called FAIR4Health under the scope of the Horizon 2020 Research and Innovation Programme.

Keywords: Health data, FHIR, de-identification, k-anonymity, ℓ-diversity

ÖZ

HL7 FHIR KAYNAKLARINDA BULUNAN SAĞLIK VERİLERİNİN GİZLİLİĞİNİN KİMLİKSİZLEŞTİRME YOLUYLA KORUNMASI

Şimşek Yılgın, Ezelsu
Yüksek Lisans, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. Pınar Karagöz

Ocak 2022 , 65 sayfa

İşbirliği ve veri paylaşımı, sağlık alanı araştırmalarının temel unsurlarındandır. Bununla birlikte, hassas sağlık veri ihlallerinin sayısı giderek artmakta ve hastaların mahremiyetinin korunmasını sağlamak için önemli bir ihtiyaç doğmaktadır. Farklı kaynaklarda toplanan sağlık verileri istatistiksel analiz, veri madenciliği ve makine öğrenimi görevlerinde faydalı olabilirler; bu da hem sağlık uzmanları hem de hastalar için uzun vadeli değer sağlar. Klinik araştırmalar sırasında sağlık verilerinin dağıtılmış doğası ile başa çıkarken gizliliğin korunması ve güvenliğin sağlanması esastır. Veriyi kimliksizleştirme ve anonimleştirme teknikleri, hastaların verilerini gizlilik risklerinden korumak için oldukça faydalıdır.

Bu tezde, gizliliğin korunmasını sağlamak için HL7 FHIR kaynaklarında bulunan sağlık verilerinin kimliksizleştirilmesi ve anonimleştirilmesi incelenmiştir. Bu çalışma, HL7 FHIR verilerinin kimliğini gizlemek için yeni bir teknik içeren ve aynı zamanda bir grafiksel kullanıcı arayüzü sağlayan Veri Gizliliği Aracının geliştirilmesini sunmaktadır. Çalışma ayrıca, çeşitli kimlik gizleme tekniklerini karşılaştıran gizlilik açısından sonuçların değerlendirilmesini de içermektedir. Bu çalışma, sağlık

verilerinin kimliksizleştirilmesi için mevcut algoritmaları incelemiş ve gizlilik koruma katmanını geliştirmek için etkili bir metodoloji önermiştir. Bu çalışmanın sonuçları çeşitli deneylerle analiz edilmiştir. Bu çalışma, Horizon 2020 Araştırma ve İnovasyon Programı kapsamında FAIR4Health adlı bir araştırma projesine katkıda bulunmayı amaçlamaktadır.

Anahtar Kelimeler: Sağlık verisi, FHIR, kimliksizleştirme, k-anonimlik, ℓ-çeşitlilik

In memory of my father...

ACKNOWLEDGMENTS

I would like to express my sincere gratitude and appreciation to my supervisor, Prof. Dr. Pınar Karagöz for her continuous support, guidance and encouragement throughout my education. With her friendly attitude, I always feel motivated to continue writing this thesis.

I am deeply grateful to SRDC family, especially Ali Anıl Sınacı, Mert Gençtürk, Alper Teoman, and Gökçe Banu Laleci Ertürkmen for their invaluable guidance and contribution in this work. I would also like to thank FAIR4Health project partners for their collaboration and precious feedback.

I am also grateful to my husband Mert Yılgın for his love, continued motivating support and welcomed presence. I also like to thank our little Pinda, who always brings joy to our family and who helped me get away from the stress of this study.

I thank to the Scientific and Technological Research Council of Turkey (TÜBİTAK) for providing the financial means through this work. My graduate studies are supported by TÜBİTAK under 2210/A scholarship program.

Finally, my special thanks go to my friend Mehmet Utku Demirci for his help, support and cheerful presence through my university life.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

API	Application Programming Interface
CAT	Cornell Anonymization Toolkit
CDA	Clinical Document Architecture
CIA	Confidentiality, Integrity and Availability
CSV	Comma Separated Values
EHR	Electronic Health Record
EMD	Earth Mover Distance
FHIR	Fast Healthcare Interoperability Resources
GDPR	General Data Protection Regulation
GUI	Graphical User Interface
HIPAA	Health Insurance Portability and Accountability Act
HL7	Health Level Seven
HTTP	Hypertext Transfer Protocol
ICD	International Classification of Diseases
IHE	Integrating the Healthcare Enterprise
ISO	International Organization for Standardization
JSON	JavaScript Object Notation
RDBMS	Relational Database Management System
REST	Representational State Transfer
SOAP	Simple Object Access Protocol
URL	Uniform Resource Locator
UTD-AT	University of Texas at Dallas Anonymization Toolbox
WSDL	Web Services Description Language
XML	Extensible Markup Language

CHAPTER 1

INTRODUCTION

1.1 Motivation and Problem Definition

As technology continues to evolve day by day, the healthcare industry also facilitates the usage of information technology more often, in order to organize and analyze health data to improve healthcare outcomes. This results in an increase in the exchange of patient data between many different systems. For example, data can be shared between doctors to assist them in easily diagnosing and treating illnesses, at the same time allowing patients to get better services. On the other hand, such Electronic Health Records (EHR) can also be published for research purposes and they can be useful for statistical analysis, data mining, and machine learning tasks; which results in long-term value for both healthcare professionals and patients.

In spite of the numerous advantages of health data transfers and usage of them, there exists also some privacy concerns for patients who own the data. Unauthorized disclosure of personal health information may cause discrimination against patients with certain types of diseases. Moreover, they may encounter some handicaps while purchasing products, obtaining services, applying for loans or benefits, etc. [6]. Those types of situations can negatively affect an individual patient's both personal and professional life.

There are various types of precautions in different kinds of information systems against such privacy threats. Within the scope of this thesis, it is aimed to reduce the privacy risks of disclosure of electronic health records residing in HL7 FHIR format, by working on "de-identification" of such data. Moreover, those kinds of disclosure threats, methodologies to preserve privacy in the literature are reviewed, and a solu-

tion for de-identification of FHIR data is proposed. A tool called Data Privacy Tool is implemented, which de-identifies FHIR data and helps ensure the privacy of patients.

Data Privacy Tool handles de-identification of FHIR formatted data with flexible configurations. It works as a standalone desktop application running on multiple operating systems. It supports several privacy criteria including k-anonymity and ℓ-diversity, as well as various evaluation results for both data utility and privacy risks. To the best of our knowledge, currently, there exists neither such study in the literature nor a de-identification tool in the market covering all the use cases that Data Privacy Tool supports.

The reason why de-identification of FHIR data is studied in this thesis is that FHIR is an emerging data standard for healthcare, which provides a lightweight way to transfer and process data, and facilitates faster development cycles because of its simple and customizable structure. It is also interoperable between many systems like web, mobile, tablet, etc., and having many advantages, FHIR is getting more commonly used in clinical systems day by day. For this reason, it requires significant attention to protect patients' data, hence their privacy.

Finally, some experiments are conducted on the proposed solution by making use of the proposed Data Privacy Tool. Some analyses on the results are made, the outcomes are discussed and improvements that can be applied in the future are proposed.

1.2 Contributions of the Thesis Study

The contributions provided by this thesis work can be summarized as follows:

- A novel de-identification solution is proposed that can be applied to FHIR data
- A new de-identification tool, Data Privacy Tool, is implemented which is compliant with data in FHIR format
- Performance of the developed technique is examined by doing experiments in terms of data utility, privacy risks, and efficiency, using the Adult Dataset

1.3 The Outline of the Thesis

The organization of the thesis can be summarized as Chapter 1 makes an introduction by providing motivation and problem definition. Chapter 2 presents the background information that is required for understanding the study subject of this thesis. It includes the HL7 FHIR standard, data privacy concept, and de-identification topic. Chapter 3, on the other hand, is the chapter where related work in the literature is covered along with similar tools in the market. In Chapter 4, the proposed methodology to de-identify FHIR data and how it is implemented as Data Privacy Tool are explained. Chapter 5 is the section where the experiments on the data are presented and the results are discussed. In the last chapter, Chapter 6, a conclusion to the thesis is made, the final remarks are presented, and comments are made on the possible future work.

CHAPTER 2

BACKGROUND

2.1 HL7 Fast Healthcare Interoperability Resources (FHIR)

Health Level 7 (HL7) is a non-profitable standards development organization established in 1987 to constitute standards for healthcare information systems. As of today, HL7 is a universal community of healthcare informatics experts that cooperate to compose standards for the exchange of medical data and healthcare systems interoperability [7]. FHIR – Fast Healthcare Interoperability Resources is a future generation standards framework originated by HL7 [8], on top of HL7's other data standards such as HL7 V2, HL7 V3, and CDA. The biggest feature distinguishing FHIR from other HL7 standards is that it is based on RESTful principles.

Representational State Transfer (REST) [9] systems can be considered as dominant on the web since majority of the platforms use RESTful services instead of other architectural choices like SOAP and WSDL nowadays [7]. The reason can be that RESTful architectures provide a lightweight way to transfer and process data, and they facilitate faster development cycles because of their simpler structures. They are also easy to consume, fast, self-descriptive; and they support all types of data, and need fewer resources like memory [10]. Being a RESTful standard, FHIR also provides those types of advantages.

2.1.1 FHIR Resources, Elements, and Profiles

In FHIR, all the main entities involved in medical information exchange are defined as "Resources". Each Resource is a uniquely identifiable asset. There are some Re-

sources called base Resources which are defined by HL7 itself. However, it is also possible to extend the usage of the base Resources. Some example Resources include: *Patient*, *Device* and *Medication* [7]. In actual exchange, Resources can be represented in the following formats: XML, JSON, and Turtle [11]. Figure 2.1 shows an example FHIR Patient Resource in JSON format.


```
{
  "resourceType":"Patient",
  "identifier":[{
        "value":"910523-2895112"
  }],
  "name":[{
        "use":"official",
        "family":[
          "Anderson"
        ],
        "given":[
          "Rollingstone",
          "K" ] }],
  "telecom":[{
        "system":"phone",
        "value":"01057524885",
        "use":"work" }],
  "gender":"male",
  "birthDate":"1991-05-23"
}
```


Figure 2.1: Sample Patient Resource [3]

There is also a concept called "Element" in FHIR, which refers to all the components contained inside a Resource. Those Elements can either be primitive data types, which have only a value and no additional Elements as children, or complex data types, which add their own children all of which are also Elements [12]. For example, in Figure 2.1 *resourceType* is a primitive whereas *name* is a complex type of Element. All primitive types can get only a certain type of value already defined by FHIR such as *boolean*, *integer*, *string*, *dateTime*, etc. [13].

There also exists a Resource type called *StructureDefinition*, which describes the structure, Element definitions, and their usage rules. These structure definitions are

used to describe both the content defined in the FHIR specification itself, and also are used to describe how these structures are used in implementations [14]. In other words, it is possible to describe every Resource with *StructureDefinition* in FHIR. Sample *StructureDefinition* Resource for Patient Resources in XML format is shown in Figure 2.2.

```
<?xml version="1.0" encoding="utf-8"?>
<StructureDefinition xmlns="http://hl7.org/fhir">

…

<base
value="http://hl7.org/fhir/StructureDefinition/Patient" />
 <differential>
  <element>
   <path value="Patient" />
   <type>
    <code value="Patient" />
   </type>
  </element>
  <element>
   <path value="Patient.identifier" />
   <definition value="An identifier for this patient" />
  </element>
  <element>
   <path value="Patient.identifier.id" />
   <representation value="xmlAttr" />
   <definition value="unique id for the element within a
resource (for internal references)" />
  </element>

…

 </differential>
</StructureDefinition>
```

Figure 2.2: StructureDefinition for Patient Resource [3]

The base FHIR specification represents a set of base Resources that are utilized in various contexts in healthcare. Nevertheless, there is extensive diversity across the healthcare ecosystem regarding practices, requirements, regulations, and which ac-

tions are practicable or advantageous for specific use cases. For this reason, FHIR also supports defining new "Profiles" on top of the base Resources. Profiles are nothing more than rule definitions about which Resource Elements are or are not used, what additional Elements are appended that are not part of the base specification; which API features are utilized, and which terminologies are available in specific Elements [15]. These Profiles can be simply defined by updating *StructureDefinition* of the base Resources.

2.1.2 Security and Privacy in FHIR

FHIR also provides mechanisms to handle security and privacy for health data. However, it does not mandate a single technical approach; instead, its specification provides a set of building blocks that can be applied to create and maintain secure, private systems [16].

For ensuring privacy protection, FHIR supports labeling data from a security and privacy point of view. Resources can be labeled according to their security information, privacy risks, etc., with the help of security-labels. This would ease the usage of patient data without violating their privacy. Tagging can be done with different types of coding systems, which have predefined value sets. For example, using *SecurityObservationValue* coding system, Resources can be marked to indicate that they have been anonymized with the tag "ANONYED". Some useful security-tags in *SecurityObservationValue* coding system can be summarized as ANONYED, MASKED, PSEUDED, REDACTED, etc.

Besides *SecurityObservationValue*, FHIR also has a privacy metadata called *Confidentiality*, which indicates sensitivity classification, based on an analysis of applicable privacy policies and the risk of harm that could result from unauthorized disclosure. Resources can be labeled with the values from *Confidentiality* coding system as well. In Table 2.1, available *Confidentiality* labels along with their descriptions can be examined.

In order to label FHIR Resources with their security/privacy information, *meta* field in the Resource should include *security* Element. This *security* Element should be

Table 2.1: Confidentiality values for FHIR Resources [1]

Code	Display Value	Definition
L	low	The information has been de-identified, and there are mitigating circumstances that prevent re-identification, which minimize the risk of harm from unauthorized disclosure.
M	moderate	Moderately sensitive information, which presents a moderate risk of harm if disclosed without authorization.
N	normal	Typical, non-stigmatizing health information, which presents a typical risk of harm if disclosed without authorization.
R	restricted	Highly sensitive, potentially stigmatizing information, which presents a high risk to the information subject if disclosed without authorization.
U	unrestricted	Privacy metadata indicating that the information is not classified as sensitive.
V	very restricted	Extremely sensitive, likely stigmatizing information, which presents a very high risk if disclosed without authorization.

a complex data type that includes security-label along with its coding system [17]. In Figure 2.3, sample Patient Resource, tagged with "Restricted" value according to *Confidentiality* system, is shown in XML format. As can be also seen from Table 2.1, this "Restricted" value in Figure 2.3 refers to "Highly sensitive, potentially stigmatizing information, which presents a high risk to the information subject if disclosed without authorization".

```
<Patient xmlns="http://hl7.org/fhir">
  <meta>
    <security>
      <system value="http://terminology.hl7.org/CodeSystem/v3-Confidentiality"/>
      <code value="R"/>
      <display value="Restricted"/>
    </security>
  </meta>
  ...
</Patient>
```

Figure 2.3: Example Patient Resource with "Restricted" tag

2.2 Data Privacy

Although the two terms, security and privacy are interchangeably used most of the time, these can be treated as two related, but entirely separate issues:

- **Security:** Security is defined as the mechanism for the protection of data against unauthorized access. It pays particular attention to preserving data from malicious attacks and theft of data for profit. Security is typically accomplished through some operational and technical controls. The three fundamental security goals are *Confidentiality*, *Integrity* and *Availability* (CIA) [18]. Even though security is a must to protect data, it is not sufficient to address privacy issues alone.

- **Privacy:** In contrast to security, privacy is a very specific term. It is often defined as the right of an individual to keep their personally identifiable information from being disclosed. Privacy is usually accomplished through policies and procedures defined by authorities. It aims to ensure that sensitive information is being collected, shared, and used in the right ways.

It is important to ensure that privacy is accomplished in order to provide patients fundamental human rights. Because disclosure of such sensitive data may lead to serious consequences in professional and personal life. For example, an employer who does not want to invest too much on an employee who is frequently ill may want to know the probability of a new applicant being ill. There can also exist other selection scenarios in cases like buying products, obtaining services, applying for loans or benefits, etc. [6]. Some of the patients having certain diseases may also experience stigmatization[1] and discrimination from the society. According to a survey conducted in 2014, patients are willing to share their medical records under the right circumstances, even though they have some concerns about sharing it [20].

There are different types of disclosure threats that can occur when electronic health records are made publicly available:

[1] the act of treating someone unreasonably by publicly disapproving of them [19]

- **Identity Disclosure:** The case when an attacker links a specific individual to a specific record in the published table. For example, if an intruder finds a patient in a dataset and all the data linked to them, then the intruder would have all the information related to the corresponding patient in that dataset. Identity disclosure also leads to attribute disclosure.

- **Attribute Disclosure:** Type of disclosure where a piece of sensitive information can be linked to an individual. For instance, if a hospital publishes data showing that all male patients aged 65 to 70 have COVID-19, then without any need to identify the specific male individual aged 65 to 70 in that dataset, an intruder already knows that such person has COVID-19 without a doubt.

- **Membership Disclosure:** When an intruder can understand whether the data about a specific individual exists in a published dataset. While this does not disclose sensitive information directly, it can help the intruder to result in identity or attribute disclosure. For example, if the intruder knows the data of some person residing in the COVID-19 patients' dataset, then he would know the person has definitely had a COVID-19 infection.

In order to address these types of privacy threats and to minimize the disclosure risks, there are data protection regulations published by different authorities, in different countries. For example in USA, Health Insurance Portability and Accountability Act (HIPAA) forms a basis on privacy rights in such a way that states cannot administer any law that ignores HIPAA rights. In EU, all organizations are required to be compliant with General Data Protection Regulation (GDPR). More examples can also be found in other countries such as UK, Russia, India, etc. [21]. Along with data protection policies, there are also various privacy preserving approaches to mitigate the disclosure risks:

- **De-identification:** Generic term for any process of unlinking personal identity revealing attributes and the individual.

- **Anonymization:** A process to remove the relationship between the identifying dataset and the individual. It is a subcategory of de-identification.

- **Pseudonymization:** A specific type of anonymization that direct identifiers are replaced with pseudonyms[2]. This method enables the information to be linked to the same individual across multiple data records or information systems without revealing the identity of the individual, provided that all direct identifiers are systematically pseudonymized.

- **Redaction:** Straightforward removal of the information that is identifying, sensitive or confidential.

In this thesis, mostly the term *de-identification* is used instead of *anonymization* intentionally with the understanding that sometimes de-identified data can be re-identified, and sometimes it cannot.

2.3 De-identification

De-identification is a process of removing the association between personal identifying information and the individual, which aims to protect against inappropriate disclosure of sensitive data. In this section, terminology, methodology, and concepts related to de-identification are summarized.

2.3.1 Data Attribute Types

It is useful to have a distinction between types of data attributes from a privacy point of view. In this way, data attributes can be handled according to their types during de-identification steps. Those data attribute types can be summarized as below:

- **Identifiers:** Uniquely identifying attributes such as citizen ID, health insurance number, etc. Normally those attributes should not be released to the public by hiding the whole attribute or by encrypting their values. Removal of direct identifiers is a straightforward process, however, is often insufficient.

[2] a number or name with no meaning that is used instead of information relating to a particular person, for example, a name or email address, so that it is impossible to see who the information relates [22]

- **Quasi-Identifiers:** Attributes that can identify users with a combination of other quasi-identifiers. For example, attributes like date of birth, gender, and postal code can be considered as quasi-identifiers.

- **Sensitive Attributes:** Set of attributes that are not really important for determining the identity of the individual, but contain confidential person-specific information such as disease, income, religion, etc.

- **Insensitive Attributes:** Attributes that have no critical information. If these attributes are revealed somehow, that would create no problem for the privacy.

Table 2.2: Sample health data with attribute types

Identifiers		**Quasi-Identifiers**		**Sensitive Attributes**		**Insensitive Attribute**
Name	**Phone Number**	**Sex**	**Birth Year**	**Lab Test**	**Lab Result**	**Pay Delay**
Alicia Fox	12387-542	Female	1983	Creatinine	0.78	29
Joe Smith	90346-133	Male	1969	Bilirubin	Negative	21
Oscar Wolf	75183-052	Male	1990	Hemoglobin	14.8	14
Lisa Last	38107-449	Female	1978	Triglycerides	147	3
Ali Veli	56183-207	Male	1973	Hematocrit	45.3	11

In Table 2.2, hypothetical sample patient data is illustrated to give an example regarding data attributes types.

2.3.2 De-identification Techniques

Quasi-identifiers are usually the most challenging part to de-identify because they include some kind of data that would be useful for later analysis. For this reason, completely removing them would damage the data utility and they require careful consideration during de-identification.

Apart from the removal of data, there are various types of techniques for the de-identification of data elements. The key objective of these techniques is to decrease the possibility of re-identification of patients. Different techniques may be more favorable for different use cases and these techniques may have variations as well. The major techniques used in de-identification procedures are shown in Table 2.3, along

with examples having before and after values when the corresponding technique is used.

Table 2.3: Major de-identification techniques with example usages

Technique	Explanation	Before	After
Pass Through	No change on the element	"value":"Ezelsu"	"value":"Ezelsu"
Redaction	Complete removal of the element	"value":"Ezelsu"	"value":""
Substitution	Replacing value of the element with another value	"value":"Ezelsu"	"value":"******"
Recoverable Substitution	Replacing value of the element with a reversible hash value	"value":"Ezelsu"	"value":"RXplbHN1"
Fuzzing	Adding some noise to the element	"value":"1000452"	"value":"1028311"
Date Shifting	Shifting the value of the date element	"value":"2019-04-05"	"value":"2019-06-23"
Generalization	Making the element less specific	"value":"2019-04-05"	"value":"2019"

2.3.3 Evaluation

There may be some cases where information like birth date, gender, a.k.a. quasi-identifiers, should not be removed since they may lead the way during an analysis of data while doing research. Researchers may find significant outcomes varying on the ages and genders of the patients. On the other hand, having those quasi-identifiers as they are results in privacy risks for most of the patients. So there is always a trade-off between the utility of the information and the privacy of the people. As data privacy increases, data utility decreases and the same also happens for the opposite situation. For this reason, there should be a balance between utility and privacy to protect the both while not harming the other.

To assess the utility and the privacy of data, most of the time analyses are conducted

on *equivalence classes*. **Equivalence class** is a set of records having the same information on the quasi-identifiers. For example, all the patients in a dataset about 35-year-old females constitute an equivalence class. The records belonging to the same equivalence class cannot be distinguished from each other with respect to corresponding quasi-identifiers.

2.3.3.1 Data Utility and Information Loss

During de-identification, some information is lost since the quasi-identifiers are modified. This loss of information degrades the utility of the data for researchers. Therefore, it is useful to be able to measure how much information may have been lost. Even though there are no standardized metrics for this, different kinds of measures have been introduced to evaluate the performance of de-identification methodologies [23]:

- **Discernibility metric (C_{DM}):** This metric has been used commonly in various de-identification studies. This is a simple metric that assigns a penalty to each record based on how many records in the modified dataset can be distinguished from each other [24].

$$C_{DM} = \sum_{EquivClasses\ E} |E|^2$$

- **Average equivalence class size (C_{AVG}):** This metric is proposed as an alternative to C_{DM} and it measures how well partitioning approaches the optimal case. The quality of the de-identified dataset is measured by the average size of equivalence classes and it aims to decrease the normalized average equivalence class size [25].

$$C_{AVG} = (\frac{total\_records}{total\_equiv\_classes})/(k)$$

- **Non-uniform entropy metric:** This metric calculates the distance between the distribution of attribute values in the de-identified dataset and the distribution of attribute values in the original dataset. The amount of information loss is computed individually for each attribute, based on the probability of correctly guessing original attribute values of records, given their modified values [23].

2.3.3.2 Re-identification Risks

Re-identification means an attempt to recognize identities that have been removed from de-identified data. An intruder might attempt a re-identification attack because of different reasons such as harming the organization that performed the de-identification or embarrassing the individual whose sensitive information can be retrieved by re-identification [26].

Re-identification risk is the risk that the identifiers and other information about individuals in the dataset can be revealed from the de-identified data. Various types of re-identification risk scenarios can be summarized as below:

- **Prosecutor Scenario:** The risk that a particular individual in the dataset can be re-identified when the intruder knows their data exists in the dataset.
- **Journalist Scenario:** The risk when there exists at least one individual in the dataset who can be re-identified. In this scenario, the point is to harm the organization which performed the de-identification, by proving that someone can be identified.
- **Marketer Scenario:** The percentage of identifiable persons in the dataset that can be accurately re-identified.

Whenever those three types of privacy risks are measured, the following will always be true [27]:

- Journalist risk is no greater than Prosecutor risk
- Marketer risk is no greater than Journalist risk

As a consequence, the following statement will always be true as well:

$$\text{Prosecutor Risk} \geq \text{Journalist Risk} \geq \text{Marketer Risk}$$

2.3.4 Further Privacy Criteria

De-identification techniques are necessary to de-identify health data and preserve the privacy of the patients. However, in most cases, it is not sufficient for ensuring the

privacy as expected. For this reason, further privacy concepts called k-anonymity, ℓ-diversity, and t-closeness were proposed to enhance the traditional de-identification techniques.

2.3.4.1 k-anonymity

William Weld was the governor of Massachusetts in the 90s and his medical data were in some publicly shared dataset. Even though information such as name and address was not included in this medical dataset, ZIP code, birth date, and sex information were included. At that time, Cambridge Voter list was also publicly shared and it also included the governor's data. In this dataset, name and address information is included along with ZIP code, birth date, and sex information. Latanya Sweeney showed that any person that has access to those datasets was able to join the datasets and infer medical data of the voters as shown in Figure 2.4 [4]. The governor had shared his birthday with six people in the dataset; only three of those people were males; and, he was the only person in his particular ZIP code.

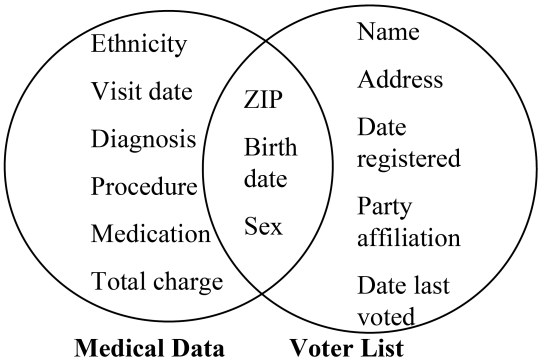


Figure 2.4: Linking two datasets to re-identify data [4]

To address these kinds of privacy vulnerabilities, Latanya Sweeney proposed the first formal data privacy criteria, called k-anonymity [4]. K-anonymity indicates that if the information for each resource in the dataset is not distinguishable from at least $k-1$ resources whose data is also in the dataset, it can be referred to as *k-anonymous*. It strengthens privacy by significantly decreasing the chance of identification. For example, if k value is 5 this means the dataset will become 5-anonymous. In this situation, any resource in the dataset will have the same quasi-identifier attributes

with at least the other 4 resources. In Table 2.4b, an example 4-anonymous dataset is provided which is derived from the dataset shown in Table 2.4a. In this example, the de-identification process is done in such a way that a set of records (at least 4 records) that have the same values on the quasi-identifiers can be grouped together.

Table 2.4: Making a dataset 4-anonymous

(a) Original patient data

	Quasi-Identifiers			Sensitive
	Zip Code	**Age**	**Nationality**	**Condition**
1	33052	27	Turkish	Arrhythmia
2	33067	26	American	Arrhythmia
3	33067	22	French	Influenza
4	33052	24	American	Influenza
5	12653	49	Chinese	AIDS
6	12653	56	Turkish	Arrhythmia
7	12650	43	American	Influenza
8	12650	47	English	Influenza
9	33052	32	American	AIDS
10	33052	34	Chinese	AIDS
11	33067	30	French	AIDS
12	33067	33	English	AIDS

(b) 4-anonymous patient data

	Quasi-Identifiers			Sensitive
	Zip Code	**Age**	**Nationality**	**Condition**
1	330**	< 30	*	Arrhythmia
2	330**	< 30	*	Arrhythmia
3	330**	< 30	*	Influenza
4	330**	< 30	*	Influenza
5	1265*	$\geq$ 40	*	AIDS
6	1265*	$\geq$ 40	*	Arrhythmia
7	1265*	$\geq$ 40	*	Influenza
8	1265*	$\geq$ 40	*	Influenza
9	330**	3*	*	AIDS
10	330**	3*	*	AIDS
11	330**	3*	*	AIDS
12	330**	3*	*	AIDS

As the value of k goes higher, the risk of re-identification will decrease. On the other hand, a higher k value may lead to more information loss, and analysis of the dataset may become harder. For this reason, this trade-off should be considered when using k-anonymity and k value should be chosen to optimize the results accordingly.

2.3.4.2 ℓ-diversity

Machanavajjhala et al. showed that even though k-anonymity is considered during the de-identification process, an attacker can still discover sensitive attributes when there is not much diversity among them. Moreover, k-anonymity cannot ensure privacy against intruders who has background knowledge about the people in the dataset. To mitigate these risks, Machanavajjhala et al. proposed an alternative model called ℓ-diversity to address the limitations of k-anonymity [28].

Table 2.5: Difference between k-anonymity and ℓ-diversity

(a) 4-anonymous patient data

	Quasi-Identifiers			Sensitive
	Zip Code	**Age**	**Nationality**	**Condition**
1	330**	< 30	*	Arrhythmia
2	330**	< 30	*	Arrhythmia
3	330**	< 30	*	Influenza
4	330**	< 30	*	Influenza
5	1265*	$\geq$ 40	*	AIDS
6	1265*	$\geq$ 40	*	Arrhythmia
7	1265*	$\geq$ 40	*	Influenza
8	1265*	$\geq$ 40	*	Influenza
9	330**	3*	*	AIDS
10	330**	3*	*	AIDS
11	330**	3*	*	AIDS
12	330**	3*	*	AIDS

(b) 3-diverse patient data

	Quasi-Identifiers			Sensitive
	Zip Code	**Age**	**Nationality**	**Condition**
1	3305*	$\leq$ 40	*	Arrhythmia
4	3305*	$\leq$ 40	*	Influenza
9	3305*	$\leq$ 40	*	AIDS
10	3305*	$\leq$ 40	*	AIDS
5	1265*	> 40	*	AIDS
6	1265*	> 40	*	Arrhythmia
7	1265*	> 40	*	Influenza
8	1265*	> 40	*	Influenza
2	3306*	$\leq$ 40	*	Arrhythmia
3	3306*	$\leq$ 40	*	Influenza
11	3306*	$\leq$ 40	*	AIDS
12	3306*	$\leq$ 40	*	AIDS

ℓ-diversity technique describes that sensitive attributes would have at most the same frequency as ℓ. In Table 2.5a and 2.5b, two different de-identified versions of the same dataset according to different metrics can be seen. In Table 2.5a, data is de-identified to satisfy 4-anonymity. However, as can be seen in the last group of records, the condition is AIDS for all rows. So if an attacker infers a patient belongs to this group, they would immediately know that the patient has AIDS. In Table 2.5b, on the other hand, data is de-identified to satisfy 4-anonymity along with 3-diversity. As can be seen, the de-identification methodology is changed and different records formed different groups at this time, having at least 3 different condition values in the same group.

2.3.4.3 t-closeness

While ℓ-diversity guarantees diversity of sensitive values in each equivalence class, it does not consider the semantical closeness of those values. To address this issue, t-closeness is proposed as a further improvement of ℓ-diversity, hence also k-anonymity. This privacy criteria basically aims that the distribution of a sensitive

attribute in any equivalence class should be close to the distribution of that attribute in the entire table. In other words, the distance between the two distributions should not be more than a threshold t. To calculate this distance between distributions, Li et al. proposed to use Earth Mover Distance (EMD) measure [5].

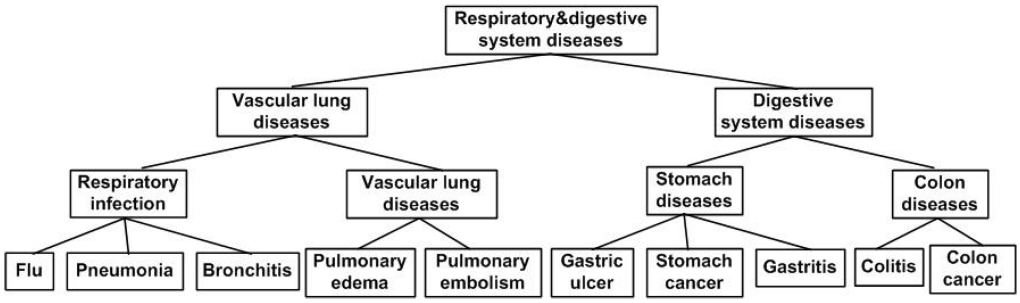


Figure 2.5: Hierarchy for categorical attributes of Condition [5]

In Figure 2.5, an example hierarchy tree for some diseases is provided. In this tree, the following distances can be given as examples:

- EMD between Flu and Bronchitis is $1/3$
- EMD between Flu and Pulmonary embolism is $2/3$
- EMD between Flu and Stomach cancer is $3/3 = 1$

Those distances are decided based on the levels where ancestors of both attributes become equal. Another example, the distance between the distribution {gastric ulcer, gastritis, stomach cancer} and the overall distribution is 0.5 while the distance between the distribution {gastric ulcer, stomach cancer, pneumonia} is 0.278.

In Table 2.6, a dataset is provided with sensitive attributes; salary and condition. All conditions in this table can also be examined in the hierarchy tree in Figure 2.5. In Table 2.7a and 2.7b, there are two different de-identified versions of that dataset according to different metrics. In Table 2.7a, data is de-identified to satisfy 3-diversity for both salary and condition fields. As can be seen, in each group there are 3 different types of salary and condition values. However, in the first group both salary and condition information are semantically similar. Salaries are close to each other, whereas conditions are all stomach diseases according to the hierarchies in Figure 2.5. So if an attacker infers a patient belongs to the first group, they would immediately know that the patient earns a salary in the range [3K, 5K] and has a stomach illness.

Table 2.6: Patient salary and condition data

	Zip Code	Age	Salary	Condition
1	17577	23	3K	Gastric ulcer
2	17502	26	4K	Gastritis
3	17578	28	5K	Stomach cancer
4	17805	41	6K	Gastritis
5	17809	50	11K	Flu
6	17806	57	8K	Bronchitis
7	17505	33	7K	Bronchitis
8	17573	31	9K	Pneumonia
9	17507	38	10K	Stomach cancer

Table 2.7: Difference between ℓ-diversity and t-closeness

(a) 3-diverse salary and condition data

	Zip Code	Age	Salary	Condition
1	175**	2*	3K	Gastric ulcer
2	175**	2*	4K	Gastritis
3	175**	2*	5K	Stomach cancer
4	1780*	≥ 40	6K	Gastritis
5	1780*	≥ 40	11K	Flu
6	1780*	≥ 40	8K	Bronchitis
7	175**	3*	7K	Bronchitis
8	175**	3*	9K	Pneumonia
9	175**	3*	10K	Stomach cancer

(b) Closeness 0.167 (salary) and 0.278 (condition)

	Zip Code	Age	Salary	Condition
1	1757*	≤ 40	3K	Gastric ulcer
3	1757*	≤ 40	5K	Stomach cancer
8	1757*	≤ 40	9K	Pneumonia
4	1780*	≥ 40	6K	Gastritis
5	1780*	≥ 40	11K	Flu
6	1780*	≥ 40	8K	Bronchitis
2	1750*	≤ 40	4K	Gastritis
7	1750*	≤ 40	7K	Bronchitis
9	1750*	≤ 40	10K	Stomach cancer

On the contrary to Table 2.7a, the semantic differences of salary and condition among people in the same group are much higher in Table 2.7b. According to Earth Mover Distance measure, this final dataset has 0.167-closeness with respect to the salary and 0.278-closeness with respect to the condition (higher the semantic difference means lower the closeness). In this example, an attacker cannot conclude that anyone has a low salary or has stomach-related conditions by knowing which equivalence class they belong to.

CHAPTER 3

RELATED WORK

3.1 Literature Survey

There are various efforts on de-identification models, algorithms, and tools in the literature. Neto et al. proposed an anonymization methodology for FHIR Resources by using encryption on sensitive patient information, as part of their security architecture proposal within a disease surveillance middleware platform [29]. Kim et al. introduced a medical data collection platform that stores and transfers foot pressure data of people with Parkinson's disease using FHIR, in which data is de-identified by simple anonymization via removal of the quasi-identifiers [3].

Croft et al. offered a new system of geographic-based de-identification for healthcare data by using a Voronoi diagram when the protection of geographic information is a high priority [23]. Prasser et al. structured a new and efficient anonymization process and developed a plugin for a common ETL platform that facilitates the integration of data anonymization and re-identification risk analyses straight into ETL workflows [30].

Ye and Chen introduced a utility-based anonymization method based on k-anonymity that aims to address the various needs of different researchers while preserving privacy along with the utility of the dataset [24]. Elabd et al. proposed an ℓ-diversity-based semantic anonymization approach which consists of enhanced privacy preserving techniques to prevent similarity attacks and increase the privacy level [31].

Benitez and Malin proposed a set of methods for calculating the probability that de-identified information can be re-identified as part of data sharing principles engaged

with the HIPAA Privacy Rule [32]. El Emam et al. offered a novel model to assess re-identification risk that considers the intruder's ability to validate matching, which is relatively a common model of how re-identification attacks are actively performed in real life [33].

Moreover, there exist several survey papers analyzing and comparing different existing works in the literature [34, 35, 36, 37, 38]. The De-identification technique used in Data Privacy Tool differs from the works in the literature in terms of different aspects. First of all, most of the de-identification attempts focus on tabular data, whereas Data Privacy Tool handles FHIR formatted data. Furthermore, our tool works standalone rather than depending on or being integrated into some other systems. It also provides flexible de-identification configurations, several privacy criteria including k-anonymity and ℓ-diversity, as well as various evaluation results for both data utility and privacy risks. On the other hand, there exists no such study in the literature covering all the use cases that Data Privacy Tool supports as of our knowledge.

3.2 Similar Tools

There exist different de-identification tools having different features. In this section, some commonly used standalone applications that have similarities with Data Privacy Tool are described and comparisons are made between them. Along with a short description of each tool, Table 3.1 summarizes the related tools and their properties.

The Cornell Anonymization Toolkit (CAT)

The Cornell Anonymization Toolkit (CAT) is proposed by researchers from Cornell University. The tool is implemented in C++ and it enforces input data to be in a tool-specific format, hence it does not support FHIR. It supports ℓ-diversity, t-closeness along with the evaluation of disclosure risk. However, CAT was developed for only demonstration purposes [42].

The UTD Anonymization Toolbox (UTD-AT)

This Anonymization Toolbox is implemented in Java by UT Dallas Data Security and Privacy Lab. It does not provide any graphical user interface and configurations

Table 3.1: The comparison between our Data Privacy Tool and the other tools

Tool	**Data Formats**	**Privacy Criteria**	**Evaluation**	**GUI**	**Open Source**
CAT [39]	Proprietary	ℓ-diversity, t-closeness	privacy risk	Yes	Yes
UTD-AT [40]	CSV	k-anonymity, ℓ-diversity, t-closeness	-	No	Yes
Amnesia [41]	CSV	k-anonymity, k^m-anonymity	-	Yes	Yes
ARX [42]	CSV, Excel, RDBMS	k-anonymity, ℓ-diversity, t-closeness	privacy risk, information loss	Yes	Yes
Microsoft FHIR Tools for Anonymization [43]	FHIR	-	-	No	Yes
Data Privacy Tool (Our solution)	FHIR	k-anonymity, ℓ-diversity	privacy risk, information loss	Yes	Yes

are done via an XML file. UTD-AT only supports unstructured text files where each record stands in a single line and attribute values of a record are split by some characters like comma, semi-column, etc. [40]. So it cannot de-identify any FHIR data. Although it supports k-anonymity, ℓ-diversity, and t-closeness; it does not assess any privacy risks.

Amnesia

Amnesia is a data anonymization tool developed by Research Center Athena. It is built with Java and Javascript and it offers graphical tools for both anonymization and analysis of the results. It supports k-anonymity and k^m-anonymity, which is proposed by the authors, where any combination of m items does not appear less than k times in equivalence classes [44].

Amnesia has both an online web application and a desktop application. The web application is used for demonstration and testing purposes mostly, whereas the purpose of the desktop application is to anonymize sensitive data locally. This tool only supports tabular data hence it cannot be considered for FHIR data anonymization.

ARX

ARX is an open-source data anonymization tool that is widely used in the literature to evaluate different anonymization methodologies/algorithms. It has a comprehensive user interface and it provides flexible configurations to the user. It enables users to anonymize their data according to k-anonymity, ℓ-diversity, t-closeness metrics they provide. ARX also evaluates the de-identification results in information loss and privacy risk aspects. Overall, ARX supports a more comprehensive methodology than Data Privacy Tool for both de-identification and evaluation of it. However, currently, it only supports importing data from CSV files, MS Excel spreadsheets, and relational database management systems (RDBMSs), such as PostgreSQL or MySQL [45]. Although it is such an extensive tool with various metrics, it has no value for FHIR Resources since they are not supported in ARX.

Microsoft FHIR Tools for Anonymization

FHIR Tools for Anonymization is an open-source tool developed by Microsoft to de-identify FHIR data. Among all the tools examined, this one is the most familiar to Data Privacy Tool since its aim is to also protect the privacy of FHIR Resources. This tool also uses methodologies like *redact*, *keep*, *dateShift*, *generalize*, etc. that are similar to our de-identification techniques.

In this tool, users need to edit a JSON file to provide the configurations, instead of making it through an application as in Data Privacy Tool. This is a big disadvantage for the users since it is error-prone and hard to follow every configuration in such a huge file. Other than that, users also need to run the tool through the command line, which is also not user-friendly and not easy to use for people without any technical background. Another option is to use Microsoft Azure PowerShell to create a Data Factory and a pipeline to anonymize FHIR data, which again requires technical background and also some prerequisites like setting up Azure Data Factory [43].

Another drawback of this tool is that it does not support any privacy criteria like k-anonymity and ℓ-diversity. Moreover, it does not provide any metrics after the de-identification like information loss, privacy risks, etc. Those are also disadvantages of this tool compared to our Data Privacy Tool.

CHAPTER 4

PROPOSED DATA PRIVACY TOOL

4.1 Methodology

Data Privacy Tool is targeted at a wide variety of end-users who are guided through the tool with some recommendations and can also make flexible configurations. The aim of the tool is to enable people to de-identify FHIR data, evaluate the results and save the final data to the FHIR repository. Data Privacy Tool is designed to operate on an HL7 FHIR API so that it can be utilized on top of any standard FHIR Repository.

There are four main steps that users must follow in Data Privacy Tool. Those steps and their explanations are shown to the user when the tool is first opened. Steps can be summarized as below:

- **Verify FHIR Repository:** In this step, the user should provide the FHIR repository endpoint in which de-identification will be done. Then, Data Privacy Tool pings the FHIR URL and if it is verified, the user can continue to the next steps.

- **Select Attribute Types:** In this step, for every available Resource/Profile in the repository, the user maps types of primitive attributes coming from FHIR Resources to the values *Identifier*, *Quasi-identifier*, *Sensitive*, *Insensitive*. Initially, some common attributes are already assigned to recommended values and the user can examine the properties of each attribute by clicking on them. Moreover, if there exists any configuration made and saved before, the user can simply select the corresponding configuration and load it to Data Privacy Tool.

  For example, if the user starts configuring attribute types of Patient Resource,

gender and *birthDate* fields will be Quasi-identifiers by default, but the user will be able to change them. Moreover, the user also has a choice to reset all recommended values and make all the mappings from scratch. In Appendix A, default attribute types for Patient Resource are provided.

- **Configure Algorithms:** In this step, de-identification algorithms and their parameters are configured by the user according to the attribute types defined in the previous step. Identifiers are directly removed; hence they do not require any configuration. Insensitive attributes remain as they are, so they do not require any configuration as well. Therefore, in this step, the user makes de-identification algorithm configurations for Quasi-identifiers and Sensitive attributes, which are explained in a detailed way in Section 4.1.1 and 4.1.2, respectively.

- **De-identify Data:** In this step, first of all, the user sees the summary of the Resources they configured. They can also save the configurations they made to use them again in the future. When the user starts the de-identification process, Resources are de-identified according to the configurations made and different types of re-identification risks are calculated along with some other evaluation metrics. The details of the de-identification methodology are explained in Section 4.1.3, whereas evaluation metrics are detailed in Section 4.1.4.

  Other than evaluation of the de-identified Resources from the point of information loss and privacy risks, those final Resources can also be viewed in JSON format in this step. Moreover, Data Privacy Tool validates the de-identified Resources according to their StructureDefinitions, then shows the validation outcomes to the user. When the user finishes examination, they can save validated Resources either to a new FHIR repository or to the current one where Resources are fetched for de-identification. If they select to save de-identified Resources to the current repository, existing data will be overwritten; which means the Resources residing in the current repository will be updated and versioned accordingly.

4.1.1 Quasi-Identifier Configuration

Quasi-identifiers can be de-identified by various techniques depending on their primitive type. For example, *date* type can support Date Shifting, whereas a *string* cannot. Different techniques are more favorable for different use cases and some techniques even have different configurations in themselves. In Section 4.1.3, de-identification techniques that are used in Data Privacy Tool and their configurations are explained in a detailed way.

In Quasi-Identifier Configuration step, the user needs to decide on which techniques with which parameters will be used for each quasi-identifier. Techniques that cannot be applied to certain primitive types are disabled so that the user cannot make any mistake. In Appendix B, supported techniques in the tool for each primitive type available in FHIR are provided. Other than these, there is also another restriction for required attributes. If a primitive type is defined as required in StructureDefinition of its Resource, then Redaction cannot be applied to it and this technique will be disabled as well. For example, if *birthDate* is a required attribute by definition, it must have some value and it cannot be redacted.

Table 4.1: Recommended de-identification techniques for some specific attributes

Attribute	**Recommended Technique (if attribute not required)**	**Recommended Technique (if attribute required)**
display	Redaction	Recoverable Substitution
text	Redaction	Recoverable Substitution
value	Redaction	Pass Through

Apart from those restrictions, the user is also guided with default parameters applied to the quasi-identifiers. For some specific attributes common in different fields in FHIR, the techniques shown in Table 4.1 are recommended. The reason why we do not recommend Pass Through (if attribute required) for *display* and *text* here is that both are usually a human display for the text and they are not intended for computation. On the other hand, *value* usually keeps data that can be meaningful for data utility.

Table 4.2: Recommended de-identification techniques for primitive types

Attribute	**Recommended Technique (if attribute not required)**	**Recommended Technique (if attribute required)**
boolean	Redaction	Pass Through
integer	Fuzzing	Fuzzing
string	Substitution	Substitution
decimal	Fuzzing	Fuzzing
uri	Recoverable Substitution	Recoverable Substitution
url	Recoverable Substitution	Recoverable Substitution
canonical	Recoverable Substitution	Recoverable Substitution
base64Binary	Redaction	Substitution
instant	Generalization	Generalization
date	Generalization	Generalization
dateTime	Generalization	Generalization
time	Generalization	Generalization
code	Redaction	Substitution
oid	Recoverable Substitution	Recoverable Substitution
id	Pass Through	Pass Through
markdown	Redaction	Substitution
unsignedInt	Fuzzing	Fuzzing
positiveInt	Fuzzing	Fuzzing
uuid	Recoverable Substitution	Recoverable Substitution

If the attribute is not one of the specific types in Table 4.1, then its recommendation is provided based on its primitive type such as being *boolean*, *string*, etc. In Table 4.2, these recommendations are shown. For specific types, these recommendations are not taken into account but the ones in Table 4.1. For all the default values and recommendations in the tool, IHE's IT Infrastructure Handbook for De-Identification and its supplement Algorithm Mapping Spreadsheet are followed [46].

Finally, in Quasi-Identifier Configuration step, the user should choose whether they will utilize k-anonymity for quasi-identifiers or not. If they want to make use of it, then they will need to provide k value as well. Currently, the default k value in Data Privacy Tool is 3.

4.1.2 Sensitive Attribute Configuration

In this step, the user will see the sensitive attributes and configure them. If in the previous step k-anonymity is enabled, the user will be able to select ℓ values for each sensitive attribute to enable ℓ-diversity for them. Moreover, by the definition, ℓ value cannot be more than k value, so ℓ values that cannot be applied are disabled as well in this step. Other than that, the user can also indicate some possible rare values that may exist in sensitive attributes. Those rare values also require a particular configuration.

Handling Sensitive Rare Values

Sensitive attributes may have some rare values. They should be indicated because they may cause the identification of patients easily. For instance, if a patient has an illness that is encountered one in a million, and this information is kept in the database as it is, it can be used for identifying the patient. For this reason, possible rare values should be selected and de-identification techniques such as Redaction, Replace or Substitution should be configured for them. Details about these techniques are provided in Section 4.1.3.

4.1.3 De-identification Methodology

De-identification workflow of Resources in Data Privacy Tool can be seen as a simplified pseudo-code in Algorithm 1. Here in line 2, the corresponding de-identification technique is executed with its parameters. How these techniques work is explained ahead in this section. At lines 10, 12, and 16; *security* labels of Resources are updated according to their privacy risks. Moreover, for lines 5 and 6 here the Resources are processed block by block, which are basically sets of Records. As Prasser et al. proposed, this blocking technique is guaranteed to be correct in terms of privacy protection [30]. For other types of effects this technique may cause, some experiments are conducted and the outcomes are discussed in Chapter 5.

Furthermore, at line 6 in Algorithm 1, Algorithm 2 is called to make Resources k-anonymous and ℓ-diverse as expected, in case of current Resources required to be compliant. As can be examined from the pseudo-code, the main logic in Algorithm 2 is to anonymize Resources as much as possible to satisfy k-anonymity and ℓ-diversity.

Algorithm 1 De-identify Resources

```
for all resource in resources do
    update attributes of resource according to the configurations
end for
if k-anonymity is enabled then
    for all blocks in resources do
        Try making blocks k-anonymous and ℓ-diverse
    end for
    for all resource in resources do
        if k-anonymity and ℓ-diversity is satisfied then
            mark resource as low-risk
        else
            mark resource as restricted
        end if
    end for
else
    mark all resources as low-risk
end if
return resources
```

If any Resource cannot be anonymized more and does not satisfy required privacy criteria, then they will be marked as *restricted* when this function returns to the main workflow Algorithm 1.

By the term "anonymizing more" we mean making data less specific by updating their de-identification parameters. In the tool, this process is done in a rule-based manner. Rules are defined for each technique covering their alternative options. For example, if Pass Through is not enough for the desired privacy criteria, then Generalization technique is used for the supported types (if not supported, then Substitution is used). If still needed, Generalization parameters are narrowed until privacy criteria is satisfied. For instance, instead of generalizing birth date to include year and month information like 1978-03, keeping only the year 1978 can be an example. If the resource is still not anonymized enough, then the value can be completely removed if the attribute is not a required field by the definition. In case of a required field, the re-

source is tagged as *restricted* afterward. The configuration details of de-identification techniques used in Data Privacy Tool are explained hereinafter.

Algorithm 2 Make Resources k-anonymous and ℓ-diverse

```
find equivalence classes in resources
while resources can be anonymized more do
    for all equivalence classes do
        if equivalence class size < k or sensitive attributes diversity < ℓ in equivalence class then
            if anonymization parameters can be updated to anonymize more then
                update parameters to anonymize more
                anonymize resources in equivalence class with updated parameters
            else
                resources cannot be anonymized more
            end if
        end if
    end for
    find new equivalence classes in updated resources
    if all equivalence classes satisfy k-anonymity and ℓ-diversity as expected then
        break while loop
    end if
end while
return resources
```

If there exists any field with missing value, Data Privacy Tool does not make anything to such fields and they stay as they are. However, during evaluation steps, all the fields with missing values constitute the same equivalence classes as if they have the same value. This results in consistent evaluation results.

On the other hand, if the data is not structured well and there exist many valuable fields in free-text format, then the tool may not be sufficient enough to de-identify such data effectively. In our methodology, it is assumed that the data should be well-fitting to the purpose of the FHIR format and should have well-structured fields, in order for Data Privacy Tool to work as expected.

Pass Through

In this technique, the attribute is saved with no change and no parameter configuration is needed. Pass Through can be applied to every type of attribute.

Redaction

In this technique, the attribute is completely removed and no parameter configuration is needed. Redaction can be applied to every type of attribute given the condition that the attribute is not a required field in its Resource.

Substitution

In this technique, the attribute is substituted with a new value. If the attribute requires regex[3], the attribute value is replaced with a randomly generated value that fits the attribute's regular expression. Primitive types that have regex constraints can be examined in Appendix B.

For the primitive types without regex, the attribute is replaced with the substitution character that the user provides. The attribute can stay in its original length or the user can provide a fixed length to be followed. For example, if the user provides substitution character "*" with fixed length 5, then the attribute will be converted to "*****". This technique can be applied to the following primitive types in FHIR: *string*, *uri*, *url*, *canonical*, *base64Binary*, *code*, *oid*, *id*, *markdown*, *uuid* [13].

Recoverable Substitution

In this technique, the new value of the attribute is generated automatically with a hash function by using the present value of the attribute. Then the current value is replaced with this new value and no parameter configuration is needed. When Recoverable Substitution is used, all the same attributes are mapped to the same value, so this technique can be useful for clustering cases. Recoverable Substitution can be applied to the following primitive types in FHIR: *string*, *uri*, *url*, *canonical*, *uuid* [13].

Fuzzing

In this technique, noise is added to the attribute within the range of the percentage user provides. For example, if the user wants to apply Fuzzing with 3% noise, an

[3] a regular expression (regex) is a sequence of characters defining a search pattern

attribute having value 100 will be mapped to a random value between 97-103. This technique can be applied to the following primitive types in FHIR: *integer*, *decimal*, *unsignedInt*, *positiveInt* [13].

Date Shifting

This technique can be considered as Fuzzing for primitive types in date or time format. The date is shifted randomly within a range that is provided by the user. The user can provide a range of years, months, days, hours, minutes, or seconds. As an example, if the user wants to shift the date 2019-06-23 for 2 months, then the attribute will get a random date between 2019-04-23 and 2019-08-23. This technique can be applied to the following primitive types in FHIR: *instant*, *date*, *dateTime*, *time* [13].

Generalization

In this technique, the attribute is generalized according to its type. For numbers, the value is rounded to make it more generic and the user can select rounding to floor or ceiling value along with the rounding precision. Generalization can also be applied to values in date or time format. This time, date information is generalized according to the information on the date unit that is provided by the user. For instance, if the user wants to generalize the date 2020-11-16 to year, then the new value will be only 2020. This technique can be applied to the following primitive types in FHIR: *integer*, *decimal*, *instant*, *date*, *dateTime*, *time*, *unsignedInt*, *positiveInt* [13].

Replace

This technique can only be used for sensitive rare values. Attributes are replaced with new values provided by the user. As an example, a patient having HIV can be considered. ICD10 classification coding levels of HIV can be observed as follows [47]:

- **(A00-B99)** Certain infectious and parasitic diseases
  - **(B20-B24)** Human immunodeficiency virus [HIV] disease
    - **(B20)** Human immunodeficiency virus [HIV] disease resulting in infectious and parasitic diseases
    - **(B21)** Human immunodeficiency virus [HIV] disease resulting in malignant neoplasms

For instance, if the patient has **(B21)** as a Condition code, by using Replace technique, the user can update that value as **(A00-B99)** in Data Privacy Tool. This way, inferring that the patient has HIV will be way much harder. Replace technique can be applied to the following primitive types in FHIR: *string*, *uri*, *url*, *canonical*, *code*, *uuid* [13].

4.1.4 Information Loss and Privacy Risks

Data Privacy Tool evaluates the de-identification process by calculating Information Loss and other metrics related to Prosecutor Risk. Information Loss is estimated according to the Expected Equivalence Class Size metric. Basically, Average Equivalence Class Size [48] metric is used in Data Privacy Tool, but it is slightly modified to also take into consideration the weights of equivalence classes. The reason why the weights are also included is to increase sensitivity against various sizes among equivalence classes as pointed out in [49]. Restricted Resources are excluded when Information Loss is calculated.

$$Information\ Loss = \frac{\sum_{EquivClasses\ E} |E|^2}{total\_records^2}$$

Re-identification risks are calculated according to the Prosecutor Risk model which is the risk that a particular individual in the dataset can be re-identified when the intruder knows they are in the dataset. In Data Privacy Tool, Prosecutor Risks are calculated separately for each equivalence class. Then the lowest, highest risks, and average risks of all equivalence classes are also computed by the tool. By using the risk outcomes, Records Affected By Lowest/Highest Risks are also calculated according to the percentage of identities in the dataset that has risk more than lowest/highest Prosecutor Risk. We are inspired by the supplementary material of [7], for the calculation of these metrics [30]. The reason why Prosecutor Risk is utilized in Data Privacy Tool is because of the fact that it is always the highest risk scenario when compared to Journalist and Marketer Risks.

$$Average\ Prosecutor\ Risk = \frac{total\_equiv\_classes}{total\_records}$$

4.2 Implementation

Data Privacy Tool is implemented as a standalone desktop application running on multiple operating systems (Windows, macOS, Linux). The main reason why it is not a web application is to prevent data sharing across the Web while sending requests. Hence, the desktop application is a safer option for privacy protection. On the other hand, compliance with various operating systems is another advantage.

The tool is developed with the Electron framework which enables building cross-platform desktop apps and combines the Chromium rendering engine with the Node.js run-time. The programming languages, frameworks, and libraries used in the development of Data Privacy Tool are as follows:

- **Programming Languages:**
  - *JavaScript*, scripting language that allows you to implement complex features on web pages
  - *TypeScript*, basically a JavaScript with syntax for types
  - *ES6*, significant update to the JavaScript programming language
- **Frameworks:**
  - *Electron*, runtime GUI framework that allows the user to create desktop applications
  - *Vue.js*, JavaScript framework for building user interfaces
  - *Quasar*, front-end framework with Vue.js components
- **Libraries:**
  - *Vuex*, state management library for Vue.js
  - *Axios*, promise-based HTTP Client for Node.js

In Figure 4.1, the user interface of Select Attribute Types step is provided. In this example, attributes in **Patient** Resources and **Patient-eu-f4h** Profiles are configured in a privacy point of view. As can be seen, *gender* and *birthDate* are selected as Quasi-identifiers whereas the remaining attributes are Insensitive. The user also clicked

birthDate here, so explanations of *Patient.Patient-eu-f4h.birthDate* are prompted in the right. Progress in the overall workflow is also tracked on the left side.

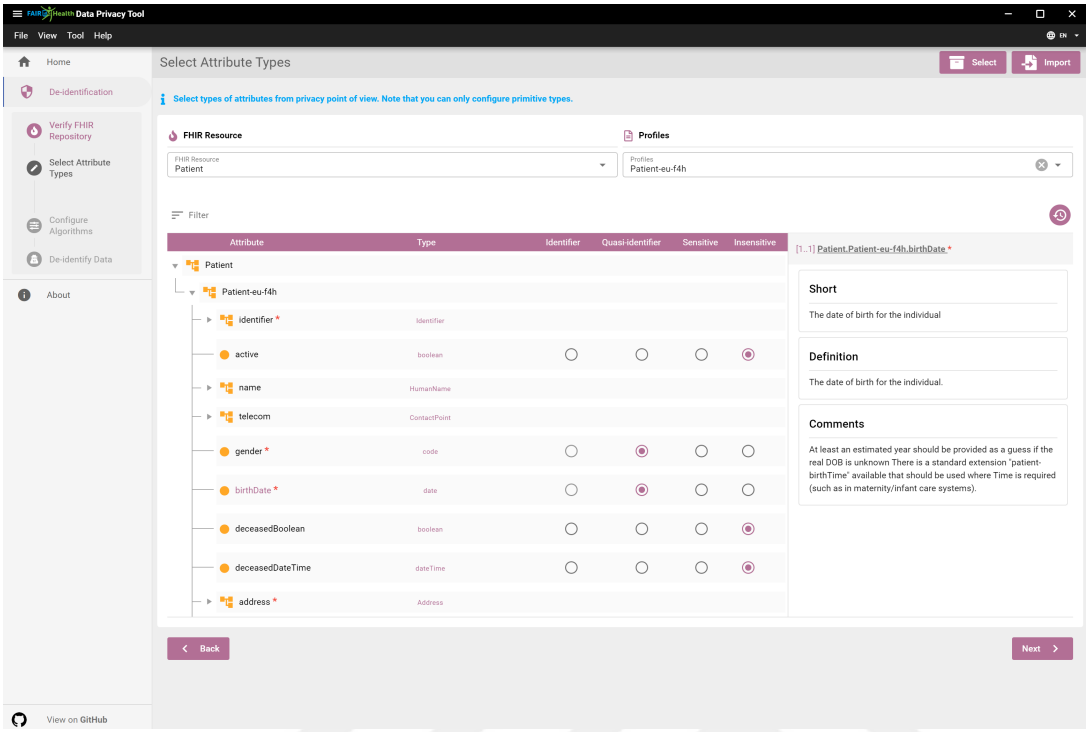


Figure 4.1: Attribute selection screen

In order to test Data Privacy Tool during implementation, unit tests are utilized with the help of two testing frameworks, Mocha and Spectron. Usage of these frameworks can be summarized as below:

- **Mocha:** Feature-rich test framework running on Node.js
  - Testing the connection of the FHIR Repository and its functionality to be used in the tool, such as querying and writing Resources.
  - Testing the de-identification steps, which include de-identification of Resources, generation of equivalence classes, and validation of modified Resources.
- **Spectron:** Framework for writing automated integrations and end-to-end tests for Electron apps
  - Verifying the behavior of Data Privacy Tool window and the tool-specific processes on different operating systems.

Source code of Data Privacy Tool can be accessed from GitHub and directly used [50]. The tool is validated by the tests implemented with Mocha and Spectron frameworks during development process. Furthermore, the tool is tested with different data and different configurations manually. In Chapter 5, some experiments are examined that are conducted by Data Privacy Tool.

CHAPTER 5

EXPERIMENTS AND RESULTS

In this chapter, it is described how an experimental evaluation on Data Privacy Tool is conducted by making use of different de-identification configurations. All experiments are run on Windows on a machine with a 2.40 GHz Intel Core i7-5500U processor and 12 GB RAM. The Adult Dataset from UCI Machine Learning Repository is used for the experiments [2]. This dataset includes 48842 records from the 1994 US Census database and it has been widely used in de-identification research areas [51].

The original dataset has the attributes with the possible values shown in Table 5.1. Among those attributes, *age*, *sex*, *marital-status*, *native-country* are used in the experiments since possibility of having those information in healthcare systems is higher than the others. They can be part of patient data as quasi-identifiers, which may cause some privacy concerns. In Figure 5.1 and Figure 5.2, distributions of the available values for *age*, *sex* and *marital-status* can be examined. On the other hand, 89.74% of the values for *native-country* attribute are United-States, 1.75% are empty and the remaining 8.51% are the other 40 countries.

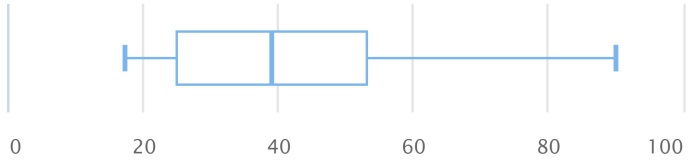
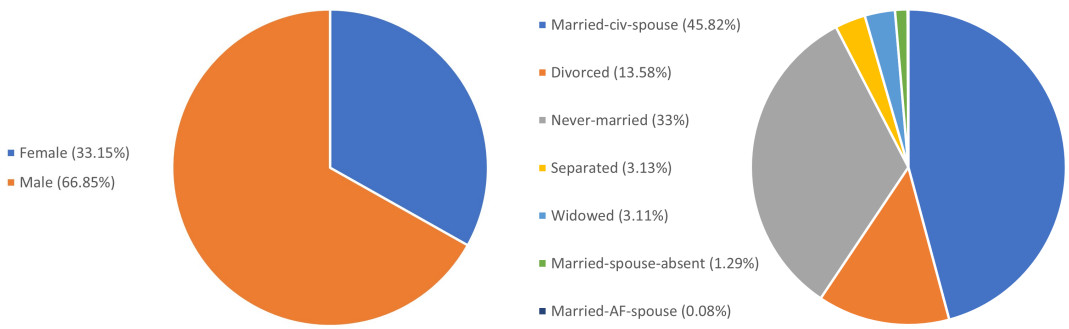


Figure 5.1: Distribution of *age* values in the original Adult Dataset

Table 5.1: Attributes included in the original Adult Dataset [2]

Attribute	**Values**
age	continuous
workclass	Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked
fnlwgt	continuous
education	Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool
education-num	continuous
marital-status	Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse
occupation	Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces
relationship	Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried
race	White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black
sex	Female, Male
capital-gain	continuous
capital-loss	continuous
hours-per-week	continuous
native-country	United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinadad&Tobago, Peru, Hong, Holand-Netherlands
class	>50K, <=50K



(a) Distribution of *sex* values

(b) Distribution of *marital-status* values

Figure 5.2: Distribution of *sex* and *marital-status* values in the original Adult Dataset

5.1 Data Preparation

In order to run Data Privacy Tool, some data in FHIR format is needed. However, there exists no common FHIR dataset which is used for research purposes. Hence, the Adult Dataset which is in CSV format is made used. For this reason, the dataset is needed to be modified slightly and converted into FHIR format.

First of all, a FHIR repository is set up to the testing machine. An open-source FHIR compliant secure health data repository called *onFHIR* is used for this purpose [52]. This tool simply enables you to have a FHIR database running on your machine. Then, it is needed to modify the dataset to make it compliant with the FHIR format. Initially, the dataset had attributes in both categorical and continuous format related to personal information including civil status, job/income information, etc. By considering the fact that the needed data should be more related to healthcare and since FHIR only supports certain types of attributes [53], from the original dataset only the following attributes are used: *age*, *sex*, *marital-status*, *native-country*.

To get a data compliant with FHIR format, the original values of *age*, *marital-status*, *sex*, *native-country* are needed to be mapped into FHIR compliant values. Other than that, some randomly generated attributes are also added to the dataset in order to examine the de-identification process in a more proper way. Overall, the following changes are done to the original dataset:

- **ID** field is added to each record in a consecutive way
- **first-name** field is added to each record with a random name value
- **last-name** field is added to each record with a random surname value
- for *sex* field, "Female" changed to "female" and "Male" changed to "male", to be compliant with FHIR
- *age* field is mapped to **birth-year** (assumed the ages are in 2021), since only birth dates are supported in FHIR
- *marital-status* field is mapped to **maritalStatus** in FHIR according to the coding system "v3-MaritalStatus" [54]
- *native-country* field is mapped to **country-code** with the ISO-3166 3 letter codes of the related countries, since it is the supported country format in FHIR addresses.

In Table 5.2, five sample records can be seen after the modifications have been done. In this step, records that have no country information are also removed and the number of total records is decreased to 47985. Now that there is a tabular dataset that includes attributes that are compliant with the FHIR standard, these data should be put into the *onFHIR* repository running on the testing environment. For this purpose, an open-source *FAIR4Health Data Curation & Validation Tool* which can migrate data from CSV files into an HL7 FHIR Repository is used [55]. This tool simply enables you to map tabular data into FHIR Resources and put them into the repository you provide. As an example of what is done in this step, sample Patient Resource generated by *FAIR4Health Data Curation & Validation Tool* from the first record in Table 5.2 can be examined in Appendix C.

5.2 Data Utility and Risks

After the data is prepared, it is de-identified with different configurations to compare the results. As data attribute types and de-identification techniques, de facto standards are used which are supported by Data Privacy Tool. In Table 5.3, the setup

Table 5.2: Some records after the modification of Adult Dataset

ID	**first-name**	**last-name**	**sex**	**birth-year**	**maritalStatus code**	**maritalStatus display**	**country-code**
1	Jordan	West	male	1996	S	Never Married	USA
129	Edgar	Morgan	male	1994	M	Married	IRL
1578	Samantha	Russell	female	1988	D	Divorced	MEX
12703	Lana	Scott	female	1965	W	Widowed	CAN
44514	Lucas	Allen	male	1957	M	Married	ITA

for each field is shown. By using these configurations, the de-identified version of the Patient Resource in Appendix C can be examined in Appendix D. As can be seen, during de-identification; some fields are removed, some others are changed and, `meta.security` field is added to the de-identified version.

Table 5.3: De-identification attribute and technique setups

Field in modified Adult Dataset	**Field in FHIR**	**Attribute Type**	**De-identification Technique**
ID	identifier.value	Identifier	Redaction
first-name	name.given	Quasi-identifier	Redaction
last-name	name.family	Quasi-identifier	Redaction
sex	gender	Quasi-identifier	Pass Through
birth-year	birthDate	Quasi-identifier	Generalization
maritalStatus code	maritalStatus.coding.code	Quasi-identifier	Pass Through
maritalStatus display	maritalStatus.coding.display	Quasi-identifier	Pass Through
country-code	address.country	Quasi-identifier	Recoverable Substitution

Having these configurations, experiments are conducted by changing the k value in k-anonymity and block sizes in row blocking. Information Loss, Average Prosecutor Risk, and Number of Restricted Records are calculated for different k values in the range [1, 5]. The de-identification process is started with block size *100* and then this number is increased logarithmically as Prasser et al. proposed [30].

Figure 5.3 provides Information Loss changes as k increases according to different

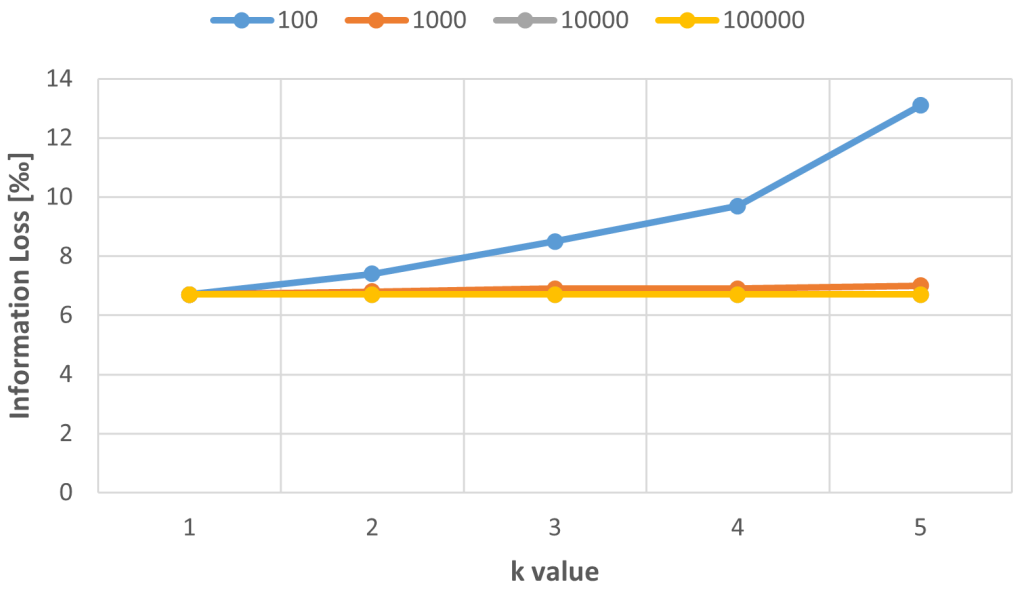


Figure 5.3: Information Losses

block sizes. As can be seen, for block size *100*, obviously Information Loss increases as k increases; whereas for the other block sizes, this cannot be seen easily. The reason is the fact that satisfying k-anonymity in smaller sets is more difficult as the possibility of having a unique value in a smaller set is higher. So, Resources should be more de-identified in smaller sets which also leads to more Information Loss. Ye et al. also had similar results as us, where more information loss happens with increasing k [24]. Moreover, it has been proven that deciding on the optimal k-anonymity solution with minimal information loss is NP-hard [56].

In Figure 5.4, Average Prosecutor Risks are provided for different k and block size values. Here the risks are decreasing as k is increasing for block sizes *1000*, *10000*, *100000*, which is the expected behaviour. On the other hand, for block size *100*, the risk seems to be increasing as k increases. The reason for that is during risk calculation restricted Resources are not taken into account in Data Privacy Tool. Almost half of the Resources de-identified with block sizes *100* are restricted to satisfy k-anonymity when $k = 5$ as can be seen in Figure 5.5a. Since such a huge number of Resources are restricted, the number of Resources that are being evaluated decreased and this led to an increase in privacy risk.

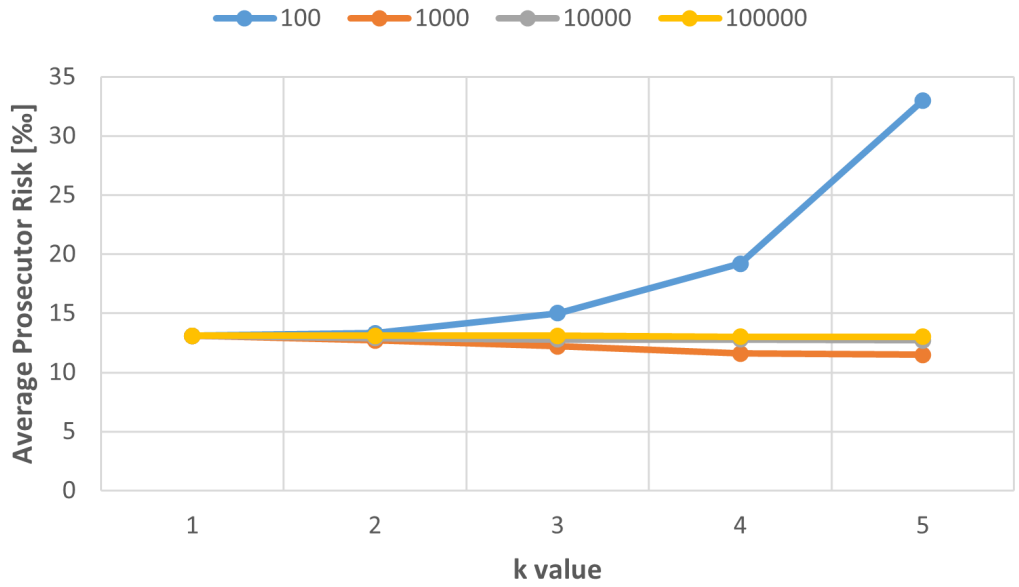
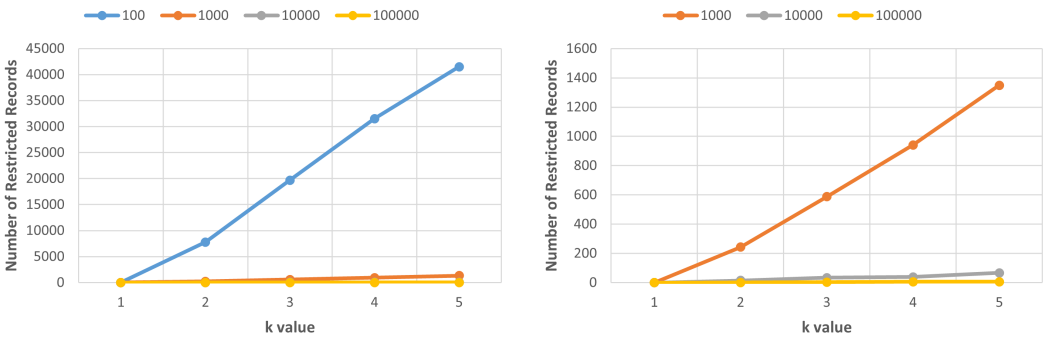


Figure 5.4: Average Prosecutor Risks



(a) With row block size 100

(b) Without row block size 100

Figure 5.5: Number of Restricted Records in two scales

In Figure 5.5, the number of restricted records is provided in two different scales where Figure 5.5a includes block size *100* whereas Figure 5.5b does not. The scale for the Number of Restricted Records residing in the y-axis drastically changes between the two graphs, because of the frequent restrictions that happened when the block size is *100*.

Considering the outcomes in both Figure 5.4 and Figure 5.5, it can be concluded

that small block sizes like *100* may cause de-identification of Resources more than needed. In the end, this unnecessary de-identification may result in more restricted Resources and more privacy risks among unrestricted Resources.

5.3 Efficiency and Scalability

The same experiments are also conducted to assess the execution times of the runs. Average Execution Times were calculated for different k values in range [1, 5]. Again, the de-identification process is started with block size *100* and then this number is increased logarithmically as Prasser et al. proposed [30]. For better precision, the same experiments are run twice and the average of two are taken as an outcome.

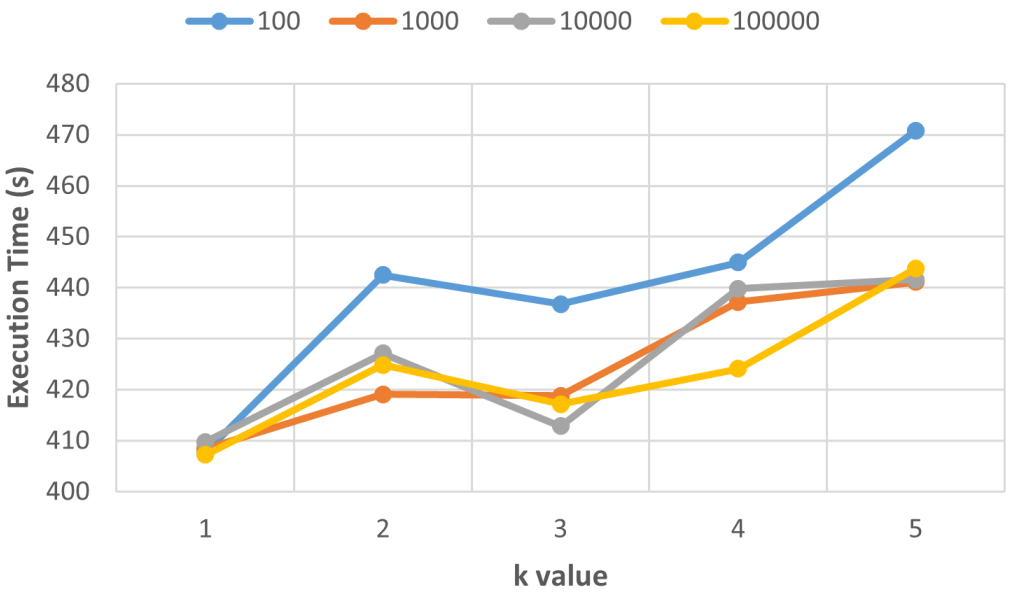


Figure 5.6: Execution Times grouped by block size

Figure 5.6 provides execution times taken when the tool is run with different k and block size values. Here it can be seen from the graph that execution times are increasing when k increases regardless of the block size. This direct proportion can be interpreted by the fact that Data Privacy Tool doing more de-identification to satisfy k-anonymity as explained in Chapter 4. As the value k increases, the tool may require to process data more for this reason.

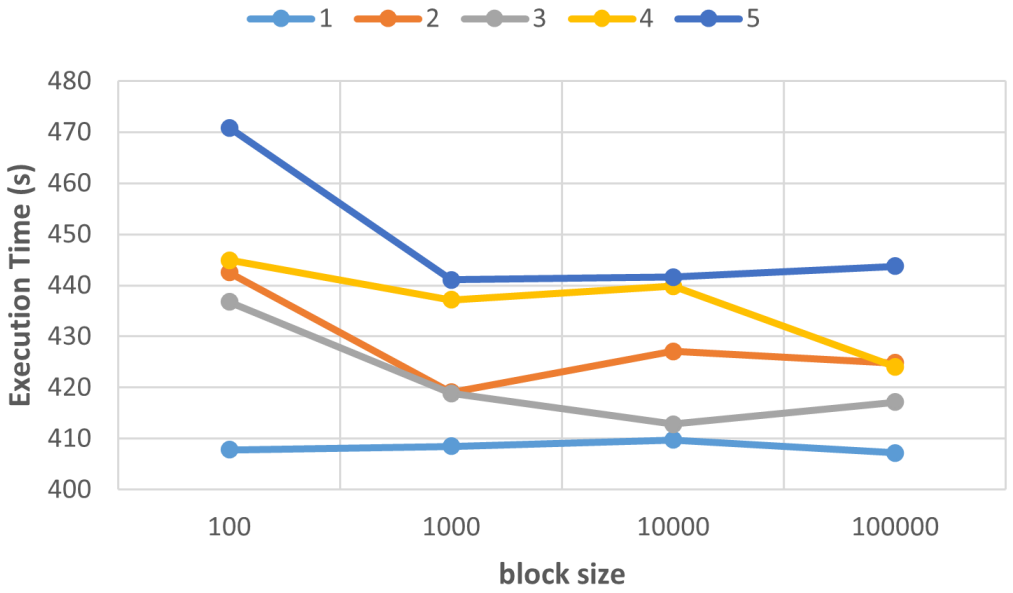


Figure 5.7: Execution Times grouped by k value

In Figure 5.7, we have the same data as in Figure 5.6 but with a different view. In this graph, execution times are grouped according to their k values rather than block sizes. Here it can be seen that when k-anonymity is not enabled (which also means $k = 1$), runs take almost the same time for each block size. Another obvious reading is that execution times are always the highest with block size 100, regardless of the k value when k-anonymity is valid. The reason for that is de-identifying fewer data tends to be computationally more costly, as good solutions are more difficult to reach [57].

On the other hand, for the other block sizes, there seems to be not much of a correlation between block size and execution time in the results. When Prasser et al. conducted a similar experiment by changing block sizes logarithmically, execution times decreased with raising block sizes up to a block size of approximately 100000, from where on they gradually raised again. They explained this rise by the fact that bigger data volumes needed to be handled in each de-identification operation. The reason why we do not have the same pattern for larger block sizes can be the difference between our datasets and quasi-identifier configurations used in the experiments. Nevertheless, we both showed that execution times are higher when block sizes are

so small. This is also in line with the fact that de-identifying fewer data tends to be computationally more costly [57].

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

Today, usage of HL7 FHIR standard in healthcare repositories is an important rising trend, along with data analysis researches getting more common for FHIR repositories. As a result of this trend, preserving the privacy of patients is a must while their data is being processed.

In this thesis, the design and implementation details of Data Privacy Tool are presented, which is developed to de-identify patient data residing in FHIR repositories. The main aim of the tool is to decrease privacy risks of patients by de-identifying their data, meanwhile keeping the data utility as much as possible for analysis needs. Users can make flexible de-identification configurations for their use cases and they can see the evaluation results afterward in the tool.

To validate our methodology and do a comparison, some experiments are conducted on the tool by de-identifying patient data with different configurations. The major findings of the experiments are that de-identification of larger datasets is an easier problem compared to the smaller sets. If FHIR Resources are de-identified block by block, block sizes should be at least *1000* for better privacy, data utility, and efficiency.

In future work, some extensions can be added to Data Privacy Tool in order to support more use cases. The possible future extensions/use cases can be summarized as below:

- Initially only attributes that have some available data can be shown to the user for configuration. Moreover, Resources that are already marked as *restricted* or *low* can be differentiated. For example, the user can filter those Resources and they are not de-identified accordingly.

- Custom hierarchies can be supported by the tool and Generalization can be applied to more primitive types with those hierarchies. t-closeness can be also enabled by using the hierarchies.
- Multiple threads can run de-identification on different blocks at the same time. This parallelization can increase efficiency and scalability significantly.
- Different types of evaluation metrics can be added to the tool.

REFERENCES

[1] "HL7 Healthcare Privacy and Security Classification System (HCS), Release 1," standard, HL7, June 2019. https://www.hl7.org/implement/standards/product_brief.cfm?product_id=345. Last accessed on November 2021.

[2] "UCI Machine Learning Repository Adult Data Set." http://archive.ics.uci.edu/ml/datasets/Adult. Last accessed on December 2021.

[3] D.-Y. Kim, S. ho Hwang, M.-G. Kim, J. H. Song, S.-W. Lee, and I. K. Kim, "Development of Parkinson Patient Generated Data Collection Platform Using FHIR and IoT Devices," *Studies in health technology and informatics*, vol. 245, pp. 141–145, 2017.

[4] L. Sweeney, "k-Anonymity: A Model for Protecting Privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.

[5] N. Li, T. Li, and S. Venkatasubramanian, "t-Closeness: Privacy Beyond k-Anonymity and ℓ-Diversity," in *2007 IEEE 23rd International Conference on Data Engineering*, pp. 106–115, 2007.

[6] B. H. M. Custers, "The Power of Knowledge. Ethical, Legal, and Technological Aspects of Data Mining and Group Profiling in Epidemiology," 2004.

[7] D. Bender and K. Sartipi, "HL7 FHIR: An Agile and RESTful approach to healthcare information exchange," in *Proceedings of the 26th IEEE International Symposium on Computer-Based Medical Systems*, pp. 326–331, 2013.

[8] "Introducing HL7 FHIR." http://www.hl7.org/FHIR/summary.html. Last accessed on November 2021.

[9] R. T. Fielding, "Architectural Styles and the Design of Network-based Software Architectures; Doctoral dissertation," 2000.

[10] S. Hussein, "Review of Web Service Technologies: REST over SOAP," p. 2020, 01 2021.

[11] "Resource Formats." https://www.hl7.org/fhir/formats.html. Last accessed on November 2021.

[12] "Element." https://www.hl7.org/fhir/element.html. Last accessed on November 2021.

[13] "Primitive Types." https://www.hl7.org/fhir/datatypes.html#primitive. Last accessed on November 2021.

[14] "Resource StructureDefinition - Content." https://www.hl7.org/fhir/structuredefinition.html. Last accessed on November 2021.

[15] "Profiling FHIR." https://www.hl7.org/fhir/profiling.html. Last accessed on November 2021.

[16] "Security and Privacy Module." http://hl7.org/implement/standards/fhir/secpriv-module.html. Last accessed on November 2021.

[17] "Security Labels." http://hl7.org/implement/standards/fhir/security-labels.html. Last accessed on November 2021.

[18] S. Haas, S. Wohlgemuth, I. Echizen, N. Sonehara, and G. Müller, "Aspects of privacy for electronic health records," *International Journal of Medical Informatics*, vol. 80, no. 2, pp. e26–e31, 2011. Special Issue: Security in Health Information Systems.

[19] "Stigmatization." https://dictionary.cambridge.org/dictionary/english/stigmatization. Last accessed on November 2021.

[20] K. T. Pickard and M. Swan, "Big Desire to Share Big Health Data: A Shift in Consumer Attitudes toward Personal Health Information," in *AAAI Spring Symposia*, 2014.

[21] K. Abouelmehdi, A. Beni-Hessane, and H. Khaloufi, "Big healthcare data: preserving security and privacy," *Journal of Big Data*, vol. 5, 01 2018.

[22] "Pseudonym." https://dictionary.cambridge.org/dictionary/english/pseudonym. Last accessed on November 2021.

[23] W. L. Croft, W. Shi, J.-R. Sack, and J.-P. Corriveau, "Geographic Partitioning Techniques for the Anonymization of Health Care Data," 2015.

[24] H. Ye and E. Chen, "Attribute Utility Motivated k-anonymization of Datasets to Support the Heterogeneous Needs of Biomedical Researchers," *AMIA ... Annual Symposium proceedings / AMIA Symposium. AMIA Symposium*, vol. 2011, pp. 1573–82, 01 2011.

[25] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan, "Mondrian Multidimensional K-Anonymity," *22nd International Conference on Data Engineering (ICDE'06)*, pp. 25–25, 2006.

[26] S. L. Garfinkel, *De-Identification of Personal Information*. Oct 2015.

[27] L. Kniola, "Plausible Adversaries in Re-Identification Risk Assessment," 2017.

[28] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "ℓ-Diversity: Privacy beyond k-Anonymity," *ACM Trans. Knowl. Discov. Data*, vol. 1, p. 3–es, mar 2007.

[29] S. Neto, F. S. Ferraz, and C. A. G. Ferraz, "Towards Identity Management in Healthcare Systems," 2016.

[30] F. Prasser, H. Spengler, R. Bild, J. Eicher, and K. A. Kuhn, "Privacy-enhancing ETL-processes for biomedical data," *International Journal of Medical Informatics*, vol. 126, pp. 72–81, 2019.

[31] E. Elabd, H. S. Abdul-Kader, and A. A. Mubark, "L-Diversity-Based Semantic Anonymaztion for Data Publishing," *International Journal of Information Technology and Computer Science*, vol. 7, pp. 1–7, 2015.

[32] K. Benitez and B. Malin, "Evaluating re-identification risks with respect to the HIPAA privacy rule," *Journal of the American Medical Informatics Association*, vol. 17, pp. 169–177, 03 2010.

[33] K. E. Emam, F. K. Dankar, A. Neisa, and E. Jonker, "Evaluating the risk of patient re-identification from adverse drug event reports," *BMC Medical Informatics and Decision Making*, vol. 13, pp. 114 – 114, 2013.

[34] V. Ayala-Rivera, P. Mcdonagh, T. Cerqueus, and L. Murphy, "A Systematic Comparison and Evaluation of k-Anonymization Algorithms for Practitioners," *Transactions on Data Privacy*, vol. 7, pp. 337–370, 12 2014.

[35] M. S. Simi, K. S. Nayaki, and M. S. Elayidom, "An Extensive Study on Data Anonymization Algorithms Based on K-Anonymity," *IOP Conference Series: Materials Science and Engineering*, vol. 225, p. 012279, aug 2017.

[36] C. Kushida, D. Nichols, R. Jadrnicek, R. Miller, J. Walsh, and K. Griffin, "Strategies for De-identification and Anonymization of Electronic Health Record Data for Use in Multicenter Research Studies," *Medical care*, vol. 50 Suppl, pp. S82–101, 07 2012.

[37] G. S and V. P, "A Survey on Privacy Preserving Data Publishing," Feb 2014.

[38] B. Eze and L. Peyton, "Systematic Literature Review on the Anonymization of High Dimensional Streaming Datasets for Health Data Sharing," *Procedia Computer Science*, vol. 63, pp. 348–355, 2015.

[39] X. Xiao, G. Wang, and J. Gehrke, "Interactive Anonymization of Sensitive Data," (New York, NY, USA), p. 1051–1054, Association for Computing Machinery, 2009.

[40] "UTD Anonymization ToolBox." `http://cs.utdallas.edu/dspl/cgi-bin/toolbox/index.php`. Last accessed on December 2021.

[41] "Amnesia - High accuracy Data Anonymization." `https://amnesia.openaire.eu`. Last accessed on December 2021.

[42] F. Prasser, F. Kohlmayer, R. Lautenschlaeger, and K. Kuhn, "ARX–A Comprehensive Tool for Anonymizing Biomedical Data," *AMIA ... Annual Symposium proceedings / AMIA Symposium. AMIA Symposium*, vol. 2014, pp. 984–93, 11 2014.

[43] "Tools for Health Data Anonymization." https://github.com/microsoft/Tools-for-Health-Data-Anonymization. Last accessed on December 2021.

[44] O. Gkountouna, S. Angeli, A. Zigomitros, M. Terrovitis, and Y. Vassiliou, "k^m-anonymity for continuous data using dynamic hierarchies," in *Privacy in Statistical Databases* (J. Domingo-Ferrer, ed.), (Cham), pp. 156–169, Springer International Publishing, 2014.

[45] F. Prasser and F. Kohlmayer, *Putting Statistical Disclosure Control into Practice: The ARX Data Anonymization Tool*, pp. 111–148. Cham: Springer International Publishing, 2015.

[46] "IT Infrastructure User Handbooks - De-Identification." https://www.ihe.net/resources/technical_frameworks. Last accessed on December 2021.

[47] "ICD-10 Version:2019." https://icd.who.int/browse10/2019/en. Last accessed on December 2021.

[48] F. Kohlmayer, F. Prasser, and K. A. Kuhn, "The cost of quality: Implementing generalization and suppression for anonymizing biomedical data with minimal information loss," *Journal of Biomedical Informatics*, vol. 58, pp. 37–48, 2015.

[49] G. T. M. Altmann, ed., *Cognitive models of speech processing: Psycholinguistic and computational perspectives*. Cambridge, MA: Bradford Books, 1991.

[50] "FAIR4Health Data Privacy Tool." https://github.com/fair4health/data-privacy-tool. Last accessed on December 2021.

[51] J. Li, R. C.-W. Wong, A. W.-C. Fu, and J. Pei, "Anonymization by Local Recoding in Data with Attribute Hierarchical Taxonomies," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 9, pp. 1181–1194, 2008.

[52] "onFHIR FHIR Repository." https://github.com/srdc/onfhir. Last accessed on December 2021.

[53] "Resource Patient - Content." http://www.hl7.org/fhir/patient.html. Last accessed on December 2021.

[54] "CodeSystem: MaritalStatus." http://terminology.hl7.org/CodeSystem/v3-MaritalStatus. Last accessed on December 2021.

[55] "FAIR4Health Data Curation & Validation Tool." https://github.com/fair4health/data-curation-tool. Last accessed on December 2021.

[56] A. Meyerson and R. Williams, "On the Complexity of Optimal K-Anonymity," in *Proceedings of the Twenty-Third ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, PODS '04, (New York, NY, USA), p. 223–228, Association for Computing Machinery, 2004.

[57] F. Prasser, F. Kohlmayer, and K. Kuhn, "Efficient and effective pruning strategies for health data de-identification," *BMC Medical Informatics and Decision Making*, vol. 16, 04 2016.

Appendix A

DEFAULT ATTRIBUTE TYPES FOR PATIENT RESOURCE

Primitive Attribute	**Type**
gender	Quasi-identifier
birthDate	Quasi-identifier
deceasedBoolean	Quasi-identifier
deceasedDateTime	Quasi-identifier
multipleBirthBoolean	Quasi-identifier
multipleBirthInteger	Quasi-identifier
name.text	Identifier
name.family	Quasi-identifier
name.given	Quasi-identifier
telecom.value	Identifier
address.text	Identifier
address.line	Quasi-identifier
address.city	Quasi-identifier
address.district	Quasi-identifier
address.state	Quasi-identifier
address.postalCode	Quasi-identifier
address.country	Quasi-identifier
maritalStatus.text	Quasi-identifier
maritalStatus.coding.code	Quasi-identifier
maritalStatus.coding.display	Quasi-identifier

Appendix B

SUPPORTED DE-IDENTIFICATION TECHNIQUES FOR PRIMITIVE TYPES IN FHIR

Primitive Type	**Regex**	**Supported Techniques**
boolean	No	Pass Through, Redaction
integer	No	Pass Through, Redaction, Fuzzing, Generalization
string	No	Pass Through, Redaction, Substitution, Recoverable Substitution, Replace
decimal	No	Pass Through, Redaction, Fuzzing, Generalization
uri	No	Pass Through, Redaction, Substitution, Recoverable Substitution, Replace
url	No	Pass Through, Redaction, Substitution, Recoverable Substitution, Replace
canonical	No	Pass Through, Redaction, Substitution, Recoverable Substitution, Replace
unsignedInt	No	Pass Through, Redaction, Fuzzing, Generalization
positiveInt	No	Pass Through, Redaction, Fuzzing, Generalization
uuid	No	Pass Through, Redaction, Substitution, Recoverable Substitution, Replace
base64Binary	Yes [13]	Pass Through, Redaction, Substitution
instant	Yes [13]	Pass Through, Redaction, Generalization, Date Shifting
date	Yes [13]	Pass Through, Redaction, Generalization, Date Shifting
dateTime	Yes [13]	Pass Through, Redaction, Generalization, Date Shifting
time	Yes [13]	Pass Through, Redaction, Generalization, Date Shifting
code	Yes [13]	Pass Through, Redaction, Substitution, Replace
oid	Yes [13]	Pass Through, Redaction, Substitution
id	Yes [13]	Pass Through, Redaction, Substitution
markdown	Yes [13]	Pass Through, Redaction, Substitution

Appendix C

SAMPLE PATIENT RESOURCE IN THE FHIR REPOSITORY


```
{
  "fullUrl": "http://127.0.0.1:8282/fhir/Patient/90fd033e84e10bed8f6b2225c0d02c62",
  "resource": {
    "resourceType": "Patient",
    "id": "90fd033e84e10bed8f6b2225c0d02c62",
    "meta": {
      "versionId": "1",
      "lastUpdated": "2021-12-04T15:47:48.327Z"
    },
    "identifier": [
      { "value": "1" }
    ],
    "gender": "male",
    "maritalStatus": {
      "coding": [
        { "system": "http://terminology.hl7.org/CodeSystem/v3-MaritalStatus",
          "code": "S",
          "display": "Never Married" }
      ]
    },
    "birthDate": "1996-01-01",
    "name": [
      { "family": "West",
        "given": [ "Jordan" ] }
    ],
    "address": [
      { "country": "USA" }
    ]
  },
  "search": { "mode": "match" }
}
```

Appendix D

DE-IDENTIFIED PATIENT RESOURCE IN THE FHIR REPOSITORY


```
{
  "fullUrl": "http://127.0.0.1:8282/fhir/Patient/90fd033e84e10bed8f6b2225c0d02c62",
  "resource": {
    "resourceType": "Patient",
    "id": "90fd033e84e10bed8f6b2225c0d02c62",
    "meta": {
      "versionId": "1",
      "lastUpdated": "2021-12-04T15:47:48.327Z",
      "security": [
        { "system": "http://terminology.hl7.org/CodeSystem/v3-Confidentiality",
          "code": "L",
          "display": "low" }
      ]
    },
    "gender": "male",
    "maritalStatus": {
      "coding": [
        { "system": "http://terminology.hl7.org/CodeSystem/v3-MaritalStatus",
          "code": "S",
          "display": "Never Married" }
      ]
    },
    "birthDate": "1996",
    "address": [
      { "country": "VVNB" }
    ]
  },
  "search": { "mode": "match" }
}
```