Recommendations on shaping technology according to GDPR provisions

An overview on data pseudonymisation

NOVEMBER 2018

www.enisa.europa.eu European Union Agency For Network and Information Security

About ENISA

The European Union Agency for Network and Information Security (ENISA) is a centre of network and information security expertise for the EU, its member states, the private sector and EU citizens. ENISA works with these groups to develop advice and recommendations on good practice in information security. It assists member states in implementing relevant EU legislation and works to improve the resilience of Europe ’s critical information infrastructure and networks. ENISA seeks to enhance existing expertise in member states by supporting the development of cross-border communities committed to improving network and information security throughout the EU. More information about ENISA and its work can be found at www.enisa.europa.eu.

Contact

For queries in relation to this paper, please use isdp@enisa.europa.eu.
For media enquires about this paper, please use press@enisa.europa.eu.

Contributors

Konstantinos Limniotis (Hellenic DPA), Marit Hansen (DPA Schleswig-Holstein)

Editors

Athena Bourka (ENISA), Prokopios Drogkaris (ENISA)

Acknowledgements

We would like to thank the following experts for reviewing this report and providing valuable comments:

Giuseppe D'Acquisto (Garante), Soerin Bipat (SIG), Mathieu Cunche (INRIA), Meiko Jensen (Kiel University)

ISBN 978-92-9204-281-3, DOI 10.2824/74954

Table of Contents

Executive Summary

Pseudonymisation is an established and accepted de-identification process that has gained additional attention following the adoption of the General Data Protection Regulation (GDPR) [1], where it is referenced as both a security and data protection by design mechanism. As a result, in the GDPR context, pseudonymisation can motivate the relaxation to a certain degree of data controllers' legal obligations if properly applied.

In this report, we present an overview of the notion and main techniques of pseudonymisation in correlation with its new role under GDPR.

In particular, starting from the definition of pseudonymisation (as well as its differences from other key techniques, such as anonymization and encryption), the report first discusses its core data protection benefits. Following this analysis, the report then addresses some techniques that may be utilised for pseudonymisation, such as hashing, hashing with key or salt, encryption and other cryptographic mechanisms, tokenization, as well as other relevant approaches. Last, certain pseudonymisation use cases and best practices are discussed, focusing especially on the area of mobile apps and revisiting some of the earlier discussed techniques.

Although the report does not seek to conduct a detailed analysis of the different aspects related to specific pseudonymisation methods and implementations, it touches upon some of the key issues in this regard. However, further research is needed, as well as practical experience, involving all stakeholders in the field.

To this end, the main conclusions and recommendations of the report are presented below.

Pseudonymisation as a core data protection by design strategy

Pseudonymisation can clearly contribute towards data protection by design, especially by technically supporting a broader interpretation of the notion of data minimisation in the digital world. This approach, however, is highly relevant to the adoption by data controllers of appropriate data protection by design frameworks, where data minimisation, also by means of pseudonymisation, is a core strategy.

Data controllers, as well as producers of products, services and applications, should adopt data protection as a key design approach in their processes; doing so, they should reassess their possibilities of implementing data minimisation by applying proper data pseudonymisation techniques.

Regulators (e.g. Data Protection Authorities and the European Data Protection Board) should promote the use of pseudonymisation as a core data protection by design strategy by further elaborating on its role under GDPR and providing relevant guidance to controllers.

[1] Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation), https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679&from=EN

Defining the state-of-the-art

The technical implementation of pseudonymisation is highly dependent on the state-of-the-art and the way that this is known and/or available to controllers. The combination of pseudonymisation with other privacy enhancing technologies is also critical to enhance overall efficiency.

The research community should continue working on privacy and security engineering, including state-of-the-art pseudonymisation (and anonymisation) techniques and their possible implementations, with the support of the European Union (EU) institutions in terms of policy guidance and research funding.

Pseudonymisation best practices in the context of GDPR

Clearly, pseudonymisation is not a prerequisite for all cases of personal data processing; hence, evaluating the relevant data protection risks (for each specific data processing case) is inherent to the decision on whether and how pseudonymisation can be implemented.

Regulators (e.g. Data Protection Authorities and the European Data Protection Board) should discuss and promote good practices across the EU in relation to state-of-the-art solutions of pseudonymisation under GDPR. EU Institutions could promote such good practices.

The research community should work out best practices out of the pooled experience on pseudonymisation (and anonymisation) at DPAs level.

Transparency and well established procedures

GDPR provides a certain relaxation of some controllers' obligations when pseudonymisation is applied. As this is a significant aspect of the GDPR's implementation, further guidance (on the regulators side) and good management (on the controllers side) is essential.

Regulators (e.g. Data Protection Authorities and the European Data Protection Board) should provide guidance and best practices on the interpretation and practical implementation of the aforementioned provisions.

Data controllers should establish well-determined procedures to this end, as well as share information regarding pseudonymisation methods applied (and their overall data processing activities).

1. Introduction

1.1 Background

In the digital world, there is a continuous increase of information flow, including also high volumes of personal data, often presented as the 'oil' of the new digital economy. The protection of this data is a key requirement in order to build trust in the online ecosystem and support fundamental values, such as privacy, freedom of expression, equity and non-discrimination. At the same time, technological advances and innovative ways of analytics, accelerate the online processing of personal data in several unpreceded and unexpected ways, for example by enabling the correlation of different types of data that may link to the same individual. It is, therefore, essential for the entities processing personal data (data controllers) on the one hand to collect and further process only those data that are necessary for their purpose, and on the other to employ proper organisational and technical measures for the protection of these data. Pseudonymisation is one well-known practice that can contribute to this end.

Broadly speaking, pseudonymisation aims at protecting personal data by hiding the identity of individuals in a dataset, e.g. by replacing one or more personal data identifiers with the so-called pseudonyms (and appropriately protecting the link between the pseudonyms and the initial identifiers). An identifier is a specific piece of information, holding a privileged and close relationship with an individual, which allows for the identification, direct or indirect, of this individual[2]. This process is not at all new in information systems design but gained special attention after the adoption of the General Data Protection Regulation (GDPR)[3], where pseudonymisation is explicitly referenced as a technique which can both promote data protection by design (article 25 GDPR), as well as security of personal data processing (article 32 GDPR).

Pseudonymisation can indeed greatly support data protection in different ways. It can hide the identity of the individuals in the context of a specific dataset, so that it is not trivially possible to connect the data with specific persons. It may also reduce the risk of linkage of personal data for a specific individual across different data processing domains. In this way, for example, in case of a personal data breach, pseudonymisation increases the level of difficulty for a third party (i.e. other than the data controller) to correlate the breached data with certain individuals without the use of additional information. This can be of utmost importance for both data controllers, as well as the individuals whose data are being processed (data subjects). Recognizing the aforementioned properties of pseudonymisation, GDPR provides a certain relaxation of the data protection rules if the data controllers have provably applied pseudonymisation techniques to the personal data.

Having said that, however, not all pseudonymisation techniques are equally effective and possible practices vary from simple scrambling of identifiers to sophisticated techniques based on advanced cryptographic mechanisms. Although many of these techniques would fall under the broad definition of pseudonymisation, they would not offer the same level of protection for the personal data. In fact, in certain cases, poor pseudonymisation techniques might even increase the risks for the rights and freedoms

[2] Examples of identifiers are the name, email address, picture of an individual, as well as specific device identifiers (e.g. a MAC address) that can be used to single out an individual in the digital world.
[3] Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation), https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679&from=EN

of data subjects, giving a false sense of protection. It is essential therefore, to explore the availability of existing solutions, together with their strengths, as well as their limitations.

The above discussion becomes even more demanding in the area of mobile applications (apps), where multiple identifiers of the mobile users are being processed by several distinct parties (e.g. app developers, app libraries, operating system –OS- providers, etc.), often without the users (data subjects) being aware of it. A recent ENISA study [ENISA, 2017] in the field highlighted the need for scalable methodologies and best practices on how to implement specific data protection measures by design in the mobile ecosystem.

Against this background and following previous ENISA work in the field [ENISA, 2014a], [ENISA, 2015], the Agency elaborated under its 2018 work-programme[4] on the concept and possible techniques of data pseudonymisation.

1.2 Scope and objectives

The scope of this report is to explore the concept of pseudonymisation alongside different pseusonymisation techniques and their possible implementation. In particular, the report has the following objectives:

- Examine the notion of pseudonymisation and its data protection goals.
- Describe different techniques that could be employed for data pseudonymisation.
- Discuss possible pseudonymisation best practices particularly for the mobile app ecosystem.

The target audience consists of data controllers, producers of products, services and applications, Data Protection Authorities (DPAs), as well as any other party interested in the notion of data pseudonymisation.

It should be noted that this report does not aim to serve as a handbook on when and how to use specific pseudonymisation techniques, but rather to provide an overview on the concept and possible practices of data pseudonymisation. The discussion and examples presented in the report are only focused on technical solutions that could promote privacy and data protection; they should by no means be interpreted as a legal opinion on the relevant cases.

1.3 Outline

The outline of this report is as follows:

- Chapter 2 presents the overall notion of pseudonymisation, including its use in GDPR and its relation to other key data protection and security techniques.
- Chapter 3 describes possible pseudonymisation techniques, including their advantages and limitations.
- Chapter 4 further focuses on pseudonymisation examples in the mobile ecosystem.
- Chapter 5, summarizing the previous discussions, provides the main conclusions and recommendations for all related stakeholders.

[4] ENISA programming document 2018-2020, https://www.enisa.europa.eu/publications/corporate-documents/enisa-programming-document-2018-2020.

This report is part of the work of ENISA in the area of privacy and data protection[5], which focuses on analysing technical solutions for the implementation of GDPR, privacy by design and security of personal data processing.

[5] https://www.enisa.europa.eu/topics/data-protection

2. The notion of pseudonymisation

This Chapter provides an analysis on the notion of pseudonymisation and its overall role in the protection of personal data. In particular, Section 2.1 starts with a definition of pseudonymisation, including both its technical description, as well as its definition under GDPR. Based on this analysis, Section 2.2 discusses the difference between pseudonymisation and anonymisation. Section 2.3 elaborates on the core data protection benefits of pseudonymisation, while Section 2.4 examines its role in GDPR.

For the discussions in this Chapter, we use the following terminology, derived from the relevant GDPR definitions:

- **Data controller** is the entity that determines the purposes and means of the processing of personal data (article 4(7) GDPR). The data controller is responsible for the data processing and may employ pseudonymisation as a technical measure for the protection of personal data.
- **Data processor** is the entity that processes personal data on behalf of the controller (article 4(8) GDPR). The processor may apply pseudonymisation techniques to the personal data, following relevant instructions from the controller.
- **Data subject** is a natural person whose personal data are processed and may be subject to pseudonymisation. The term **individual** is also used in the text to refer to a data subject. Moreover, the term **user** is utilised in the same sense, especially when discussing online/mobile systems and services.
- **Third party** is any entity other than the data subject, controller or processor (article 4(10) GDPR).

Any examples presented in the text are only to support the underlying technical description and are not meant to present a legal interpretation on the relevant cases.

2.1 Definition of pseudonymisation

2.1.1 Technical description

In broad terms, pseudonymisation refers to the process of *de-associating* a data subject's identity from the personal data being processed for that data subject. Typically, such a process may be performed by replacing one or more *personal identifiers*, i.e. pieces of information that can allow identification (such as e.g. name, email address, social security number, etc.), relating to a data subject with the so-called *pseudonyms*, such as a randomly generated values.

To this end, the ISO/TS 25237:2017 standard defines pseudonymisation as a '*particular type of de-identification that both removes the association with a data subject and adds an association between a particular set of characteristics relating to the data subject and one or more pseudonyms*' [6][ISO, 2017]. De-identification, according to the same standard is a '*general term for any process of reducing the association between a set of identifying data and the data subject*'. A pseudonym is also defined as '*a personal identifier that is different from the normally used personal identifier and is used with pseudonymized data to provide dataset coherence linking all the information about a data subject, without disclosing the real world person identity'.* As a note to the latter definition, it is stated in ISO/TS 25237:2017 that pseudonyms are usually restricted to mean an identifier that does not allow the direct derivation of the normal personal

[6] A similar definition of pseudonymisation has also been adopted by the US National Institute of Standards and Technlogy (NIST), see in: https://csrc.nist.gov/glossary/term/pseudonymization

identifier. They can either be derived from the normally used personal identifier in a reversible or irreversible way or be totally unrelated.

Another technical definition of pseudonymisation is provided by the ISO/IEC 20889:2018 standard as a '*de-identification technique that replaces an identifier (or identifiers) for a data principal with a pseudonym in order to hide the identity of that data principal*[7]' [ISO, 2018]. A pseudonym is subsequently defined as a '*unique identifier created for a data principal to replace the commonly used identifier or identifiers for that data principal*'. Relevant definitions can also be found in [Pfitzmann, 2010], where a pseudonym is considered as *'an identifier of a data subject other than one of the subject's real names*' and the notion of *pseudonymity* is defined as '*the use of pseudonyms as identifiers*'.

Despite the different terminology used, in all the aforementioned definitions, it is clear that pseudonymisation is expected to take out of sight, or 'hide' the identifying information (i.e. personal identifiers) relating to data subjects by replacing them with pseudonyms, while, however maintaining an association between the two (personal identifiers, pseudonyms) that allows for the re-identification when needed. Clearly, towards providing a high level of protection of data subjects' identities, such an association should be, somehow, secured and not obvious to anyone having access only to the pseudonymised data to render pseudonymisation a realistic choice. This association falls under the concept of '*additional information*' introduced by GDPR and will be discussed in Section 2.1.2.

Note that for the remainder of the document we use interchangeably the terms **personal identifier** or **initial identifier** or **identifier** to refer to any piece of information that can be used to identify a data subject (see also Section 2.1.3). The term **pseudonym** is used to refer to a piece of information that replaces a personal identifier as the result of a pseudonymisation process.

2.1.2 Pseudonymisation in GDPR

Pseudonymisation is defined in article 4(5) of the GDPR as: '*the processing of personal data in such a manner that the personal data can no longer be attributed to a specific data subject without the use of additional information, provided that such additional information is kept separately and is subject to technical and organisational measures to ensure that the personal data are not attributed to an identified or identifiable natural person'*.

In GDPR, the data controller is the entity responsible to decide whether and how pseudonymisation will be implemented. Data processors, acting under the instructions of controllers, may also have an important role in implementing pseudonymisation. Recital (29) GDPR states that the application of pseudonymisation to personal data can reduce the risks to the data subjects concerned and help data controllers and processors to meet their data protection obligations. Moreover, recital (30) states that measures of pseudonymisation should, whilst allowing general analysis, be possible within the same controller, when that controller has taken appropriate technical and organisational measures and that additional information for attributing the personal data to a specific data subject is kept separately.

The GDPR definition of pseudonymisation, while in accordance with the aforementioned technical descriptions, provides a stricter framework for implementation as it states that '*personal data can no longer be attributed to a specific data subject without the use of additional information*'. Following the definition of personal data in article 4(1) GDPR, i.e. any information relating to an identified or identifiable person, this would mean in practice that pseudonymised data should not allow for any direct or indirect

[7] The term 'data principal' has similar meaning with 'data subject'.

identification of data subjects (without the use of additional information). Therefore, pseudonymisation under GDPR goes beyond the protection of *'the real world person identity'* to also cover the protection of indirect identifiers relating to a data subject (e.g. online unique identifiers – see also Section 2.1.3). Moreover, the reversal of pseudonymisation should not be trivial for any third parties that do not have access to the '*additional information'*. This is clearly relevant to the pseudonymisation technique applied, which should also be in accordance with the GDPR data protection principles (article 5 GDPR).

Moreover, the GDPR definition of pseudonymisation puts a lot of emphasis on the protection of the *additional information*, which, taking into account the technical meaning of pseudonymisaton, would in practice refer to the association between the initial identifiers of the data subjects and the pseudonyms. According to GDPR, this association needs to be secured and separated from the pseudonymised data (by the data controller). Indeed, if anyone with access to pseudonymised data had also access to the additional information, then he/she would be trivially able to reverse the pseudonymisation, i.e. to identify the individuals whose data are processed. There is no specific reference in the GDPR with regard to whether such a data separation should be logical or physical. In any case, if a data controller performs pseudonymisation, it is evident that appropriate measures need to be implemented to prevent access to associations between pseudonyms and initial identifiers (e.g. by putting them into a different database or into a trusted third party). Cleary, destroying such associations, in cases where preserving is not required by the controller, may add an additional layer of protection.

It should be pointed out though that there might be cases in which a third party (i.e. other than the controller or processor) could possibly be able to re-identify an individual from pseudonymised data, even without access to the additional information being kept by the data controller. For instance, this may occur in cases where the pseudonymisation technique is not strong enough, for example due to the fact that the pseudonyms are "trivially" generated from personal data that are publicly available (note, however, that such a technique would probably not fall under the strict definition of GDPR in the first place). In addition, there is always the risk that the post-pseudonymized dataset still contains fields (e.g. a street address) or combination of fields that, when correlated with other information, could allow for the re-identification of the individuals (see also relevant discussion on anonymisation in Section 2.2). For example, free text fields with a message and a greeting line could potentially allow linking to a specific individual even when the data are pseudonymised (i.e. personal identifiers have been removed). The characteristics of the dataset could play an important role to this end, as they could potentially facilitate inference of individuals' identifies from the post-pseudonymised data (e.g. if a dataset relates to a small/specialised group of persons, certain attributes may immediately infer the identifiers of specific individuals within this group, even when personal identifiers have been removed). This risk is further accentuated by the fact that even if re-identification is not possible at a certain point in time, accumulating additional data that are associated with a pseudonym could possibly allow for re-identification in the future.

To this end, data controllers should have a clear understanding of the scope of data pseudonymisation and select the appropriate technique that could suffice for this particular scope. As mentioned earlier, an inadequate level of pseudonymisation would probably not meet the requirements laid down by the data protection principles of GDPR (article 5), even if they fall under the broader technical meaning of pseudonymisation.

2.1.3 The notion of identifiers

We have already referred in several instances to the notion of identifiers (or personal identifiers or initial identifiers) and their central role in pseudonymisation. In this Section, the report elaborates further on this important matter.

According to the Article 29 Working Party [WP29, 2007], identifiers are pieces of information, holding a particularly privileged and close relationship with an individual, which allows for identification, whilst the extent to which certain identifiers are sufficient to achieve identification is dependent on the context of the specific personal data processing. Hence, identifiers may be single pieces of information (e.g. name, email address, social security number, etc.) but also more complex data. For instance, although the name of the person is one of the most common identifiers, complex-type data (e.g. photos, templates of fingerprints etc.) or combination of data (e.g. combination of street address, date of birth and sex) may also play the role of identifiers. Moreover, the possibility of an identifier to lead to the identification of a specific data subject is highly relevant to the particular context in which it is applied, which in practice means that the same identifier might provide for different levels of identifiability of the same data subjects in different contexts. For example, even the simple case of an individual's name may not always suffice to uniquely identify the individual, unless additional information is being considered – for instance, a very common family name will not be sufficient to identify someone (i.e. to single someone out) from the whole of a country's population [WP29, 2007]; however, this might become possible if the name is combined with other data, such as for example telephone number or email address.

Another important aspect to this end is that, when considering whether a piece of information could qualify as a personal identifier, the possibility of both direct, as well as indirect identification of the data subject by the data controller needs to be taken into account, which broadens the overall notion of identifiers. This aspect is especially relevant to the use of online and mobile services, where a multitude of device and application identifiers are utilised (by the device/service/application providers) to single out specific individuals (i.e. the users of the relevant devices or applications). For instance, in the mobile ecosystem, the usage of unique device identifiers may have a significant impact on the private lives of the users [WP29, 2013], allowing for extensive tracking and profiling.

Therefore, pseudonymisation should not necessarily be interpreted as a technique applying to a single simple attribute/identifier, since there may be cases which necessitate application of pseudonymisation techniques to a bundle of multiple attributes of an individual (e.g. name, location, timestamp) to bring data protection benefits. Hence, depending on the context, different requirements with respect to pseudonymisation may occur. In this document, we use the term identifier (or personal identifier or initial identifier) to refer to all the possible cases (complex or not).

2.1.4 Pseudonymisation and self-chosen pseudonyms

It is important to stress that the notion of data pseudonymisation should not be confused with the practice of self-chosen pseudonyms, i.e. pseudonyms that individuals might select and apply themselves, such as for example nicknames of users in online blogs or forums[8]. The latter is a well-known practice that can contribute to 'hiding' an individual's real name but it is based on the choice of the individuals themselves and does not rely on a process applied by a data controller[9].

Although self-chosen pseudonyms might contribute to reducing the exposure of an individual's identity in specific contexts, collection and storage of merely such type of data by a data controller (e.g. provider of an online blog or forum) does not constitute pseudonymised data processing in the meaning of the GDPR. In fact, self-chosen pseudonyms actually play the role of identifiers and can be used to single out specific

[8] Note also the general notion of pseudonym as a fictious name or alias, e.g. in literature.
[9] Although a data controller, e.g. the provider of an online blog or forum, might allow or even opt for such type of 'pseudonymous' use of its services, i.e. without asking users to register or otherwise provide proof of their real identity.

individuals, especially in correlation with other relevant data (e.g. posts in a blog) or even from the chosen pseudonyms themselves.

Note, however, that the aforementioned concept of self-chosen pseudonyms, should not be confused with cases of pseudonymisation where the pseudonym (as a part of the whole pseudonymisation process performed by a data controller) is generated locally in the data subject's environment (e.g. user's device via a cryptographic technique). Such cases do fall under the definition of pseudonymisation and can even constitute best practices in the field, as described in Chapters 2 and 3.

2.2 Pseudonymisation and anonymisation

There is often some confusion between the notion of pseudonymisation and that of anonymisation and their application in practice. However, as discussed in this Section, these two notions are clearly different and attention should be paid so as not to perceive pseydonymised data as anonymised.

ISO/TS 25237:2017 defines anonymisation as a '*process by which personal data is irreversibly altered in such a way that a data subject can no longer be identified directly or indirectly, either by the data controller alone or in collaboration with any other party*' [ISO, 2017]. Similarly, NIST refers to anonymisation[10] as a '*process that removes the association between the identifying dataset and the data subject*'. In simple words, an anonymised dataset does not allow for identifying any individual, for either the controller or third party. Therefore, anonymised data do not qualify as personal data.

Clearly, removing any user's identifier is prerequisite for anonymisation, but in general this does not suffice to ensure anonymity and more sophisticated approaches need to be adopted. Several known anonymisation techniques exist [WP29 - 2014], whereas two main approaches are the so-called randomisation and generalisation techniques: the first one aims to "randomly" alter the data (e.g. via noise addition) in order to remove the link between the dataset and the individual, whilst the latter aims to appropriately modify, for specific attributes, the respective scale or order of magnitude (e.g. an exact age can be replaced by an age range etc.) in order to prevent singling out. In general though, there is no anonymisation technique that should be considered as panacea[11].

As already mentioned, a common misleading mistake is that pseudonymised data are perceived as anonymous data[12]. However, this is not the case; recalling the relevant definitions, pseudonymisation is related to the existence of an association between personal identifiers and pseudonyms, whilst in anonymisation such an association should not be available by any means. Hence, re-identification is possible (and even required for the data controller) in pseudonymisation whereas in anonymisation this is in principle not the case. In other words, pseudonymised data are still personal data, while anonymised data are not.

The GDPR also explicitly clarifies this misinterpretation. More precisely, as stated in its Recital (26), anonymous information refers to information which does not relate to an identified or identifiable natural person - and, thus, anonymous data are not considered as personal data (in such a case, the legal

[10] See in: https://csrc.nist.gov/glossary/term/anonymization

[11] Different privacy enhancing techniques may also contribute to this end. A practical example of a relevant user application is 'I reveal my attributes' (IRMA), which allows users to disclose properties (attributes), such as 'I am over 18 years old', without disclosing other data. For more information see: https://privacybydesign.foundation/en/

[12] This was also the case of the famous AOL incident in 2006, when a database containing twenty million search keywords for over 650,000 pseudonymous users over a 3-month period was released in public, which in turn resulted into the identification of several users – see. e.g. the case of the user with the pseudonym 4417749 in https://www.nytimes.com/2006/08/09/technology/09aol.html (last accessed: July 20th, 2018).

framework on personal data protection does not apply)[13]. On the contrary, pseudonymised data, which can be attributed to a natural person by the data controller with the use of additional information, are personal data and all relevant data protection principles apply to them.

Still, the term "anonymous" is often used in common language to describe cases where the identities of the data subjects are only hidden (but the data are not truly anonymised). For example, there exist several so-called "anonymous" social network applications that typically do not require their users to create profiles and collect very limited information about them. By these means, it is supposed that users may express their beliefs and opinions freely without exposing their identities. However, many of these applications process an identifier of the user's device – for instance, to send notifications to users whenever other "anonymous" users like their posts or to provide information on nearby "anonymous" users of the same network. However, as also stated earlier, device identifiers should in principle be considered as personal data since they are associated with the device's users. This is especially the case for permanent identifiers. Still, even if a non-permanent device ID is being used by such "anonymous" applications, there might still exist an association between this identifier and the device, which in turn poses risks for user's privacy [Chatzistefanou, 2017], e.g. potentially facilitating the process of device fingerprinting[14].

It should be pointed out that even in the absence of personal identifiers, data are not necessarily anonymous. For example, in the previous case of the anonymous social networks, a user of such a network might be identified by, e.g., his/her posts and/or other activities, without the use of any device identifier. Similarly, as it has been shown in [Su, 2017], simple browsing histories could be linked to some social network profiles such as Twitter or Facebook accounts, owing to the fact that users are more likely to click on links posted by accounts that they follow. In the same context, it was shown in [Kurtz, 2016] that users of iOS devices could be singled out through their personalised device configurations, despite the fact that there was no access from third-party apps to any device hardware identifiers. Moreover, as stated in [Zhou, 2013], personal data could be inferred from publicly available information in earlier versions of the Android system. This shows the difficulty of yielding data anonymous, while widening the notion of pseudonymised data[15].

However, it should be noted that, despite the distinction between pseudonymisation and anonymisation, the former often relies on techniques of the latter in order to enhance its efficiency. For instance, in some cases it might be a good practice to involve certain anonymisation techniques (e.g. attributes generalisation) in the pseudonymisation process, so as to reduce the possibility of third parties to infer personal data.

2.3 Data protection benefits of pseudonymisation

[13] However, as also stated in the same Recital, to determine whether a natural person is identifiable, account should be taken of all the means reasonably likely to be used by anyone to identify the natural person directly or indirectly, taking into account all objective factors. This is extremely important since it requires, before characterising data as anonymous, to cautiously answer the question whether it is really impossible for any party – including the data controller - to identify from these data any individuals. In the big data era, such privacy risks are further increased [ENISA, 2015].

[14] The term *device fingerprinting* is used to describe any technique for identifying a computing *device* (e.g. laptop, tablet or smartphone) based on its unique configurations. This term was first used in [Eckersley, 2010], where it is shown that the web browser of a device usually leaves a unique digital "trace" such that a website may uniquely distinguish the user (i.e. device) from all the other visitors to that site (this special case is also being called as browser fingerprinting)..

[15] An illustrative guide of general de-identification steps is shown in https://fpf.org/2016/04/25/a-visual-guide-to-practical-data-de-identification/ (last access: October 21st, 2018).

Following the GDPR definition of pseudonymisation, one important remark is that pseudonymisation starts with a *single input* (original dataset) and results in a *pair of outputs* (pseudonymised dataset, additional information) that together can reconstruct the original input. The same logic also resides behind the technical definitions of pseudonymisation, as the pseudonymised dataset is actually a modified version of the original dataset where data subjects' identifiers have been replaced by pseudonyms, whereas the additional information provides the link between the pseudonyms and the identifiers. Therefore, pseudonymisation in fact separates the original dataset in two parts, where each of the parts has a meaning with regard to specific individuals only in combination with the other. This decoupling is essential in understanding the notion of pseudonymisation and the benefits that it brings with regard to data protection.

To start with, the first and obvious benefit of pseudonymisation, actually directly derived from its definition, is to hide the identity of the data subjects for any third party (i.e. other than the controller or processor) in the context of a specific data processing operation, thus enhancing their security and privacy protection. Indeed, if by means of security (e.g. access control, chain of custody), the data controller can keep the two distinct outputs of pseudonymisaton separate, then any recipient or other third party having access to pseudonymised data cannot trivially derive the original dataset and, thus, the identity of the data subjects.

To this end, pseudonymisation can actually go beyond the hiding of real identities in a specific data processing context into supporting the data protection goal of unlinkability[16] [ENISA, 2014a], i.e. reducing the risk that privacy-relevant data can be linked across different data processing domains. Indeed, when data are pseudonymised, it is more difficult for a third party to link them to other personal data that might be relating to the same data subject (again without the use of additional information)[17]. Unlinkability is closely relevant to the fundamental data protection principles of necessity and data minimisation.

Furthermore, it is important to consider that there might be cases where the controller does not need to have access to the real identities of data subjects in the context of its specific processing, for example, it might be sufficient for the controller only to trace/track the data subjects without storing their initial identifiers[18]. Certain pseudonymisation techniques, on the basis of the decoupling mentioned above, can facilitate this goal, thus supporting the overall concept of data protection by design (e.g. by technically using the least possible personal data for a given purpose of processing).

Last, recalling the role of decoupling in pseudonymisation, another important benefit of this process that should not be underestimated is that of data accuracy. Indeed, if a data controller has in its possession the two outputs of pseudonymisation, the integrity of the original dataset (which can only be reconstructed on the basis of both these outputs) cannot be contested. This can be a useful tool for the data controller, contributing to the data protection principle of accuracy.

Following the above elements, pseudonymisation, if properly applied, can be beneficial for the data controller as a useful tool that not only can enhance the security of personal data but also support its overall compliance with the GDPR data protection principles. At the same time, pseudonymisation is also beneficial for the data subjects, whose personal data protection is enhanced, thus further contributing to building trust between controllers and data subjects, which is an essential element for digital services.

[16] Note that anonymisation also has the same goal but in a broader sense as the data controller would also qualify as third party.
[17] This in fact refers to the possibility for a third party to sufficiently distinguish whether two or more entities are related, i.e. the case where different pseudonyms can be linked to each other.
[18] Note that tracking stills enables singling out the specific individuals by the controller, even if the initial identifiers are not stored.

2.4 The role of pseudonymisation in GDPR

Recognizing the possible benefits of pseudonymisation, the GDPR refers approximately fifteen (15) times to it in several forms, including the following:

- According to the Article 25(1) GDPR, pseudonymisation may be an appropriate technical and organisational measure towards implementing data protection principles in an effective manner and integrating the necessary safeguards into the processing (data protection by design);
- According to the Article 32(1) GDPR, pseudonymisation – as well as encryption - may be an appropriate technical and organisational measure towards ensuring an appropriate level of security (security of processing).

For both the above cases, the GDPR explicitly mentions that the choice of the pseudonymisation as an appropriate measure is contingent on the cost of implementation and the nature, scope, context and purposes of processing as well as the relevant risks for the rights and freedoms of natural persons. Therefore, a decision on whether pseudonymisation should take place or not rests with the associated data protection risks, which means that there are cases where pseudonymisation stands as a prerequisite (e.g. whenever it is needed to ensure that the processing is proportionate to the purpose it is meant to address), but there are also cases where pseudonymisation may not be necessary. Even though in cases where pseudonymisation needs to take place, the data controller should proceed one step further to examine which is the optimal approach, taking into account all the aforementioned relevant factors. This is a direct consequence of the fact that neither all pseudonymisation techniques are effective up to the same extent nor do they have the same requirements in terms of implementation. Therefore, it is probable that even if a specific pseudonymisation approach suffices to address the data protection risks in one case of data processing, it may not be appropriate for a different data processing.

Moreover, it should be pointed out that pseudonymisation, according to the GDPR provisions, serves as the vehicle to somehow "relax" some of the data controller's obligations. For instance, personal data may be further processed, in accordance with the principle of purpose limitation, for archiving purposes in the public interest, scientific or historical research purposes or statistical purposes, if these data are subject to specific safeguards, including pseudonymisation (articles 5(1)(b) and 89(1) GDPR). In the same line, article 6(4) GDPR states that towards deciding whether any new purpose (other than that for which the data have been collected) is compatible with the initial one, several factors should be considered – including the possible existence of pseudonymisation as an appropriate safeguard. In addition, appropriately-implemented pseudonymisation can reduce the likelihood of individuals being identified in the event of a data breach, which is a factor to be considered by the data controller towards assessing the risks of the breach and deciding whether data subjects should be informed [WP29-2018] (although, as it is also explicitly stated in [WP29-2018], pseudonymisation techniques alone cannot be regarded as making the data unintelligible).

Finally, in cases that the data controllers process the personal data in a way that they cannot identify the individuals (e.g. in processes where the additional information allowing for re-identification has been deleted by the controller[19]), additional exemptions might apply for the controller, on the basis of articles 12(2) and 11 GDPR. More precisely, articles 15 to 20 (i.e. the data subjects rights concerning access to the data, rectification and erasure of data, restrictions on processing and data portability) do not apply

[19] Note that in such cases and depending on the technique used, this way of processing might actually lead effectively to data anonymisation (see also Section 2.2).

whenever the controller is provably unable to identify the users[20] (unless the data subjects provide additional information themselves enabling their identification[21]). However, this "relaxation" on the controller's obligations necessitates that the controller should establish well-determined procedures for proving that such re-identification is indeed impossible based on the current data that are being processed. To achieve this goal, a prerequisite is the transparency of the overall pseudonymisation process, which clearly raises another important aspect of pseudonymisation that should be considered by the controllers.

Apparently, the aforementioned "relaxation" of data controllers' obligations holds only if a proper pseudonymisation approach has been adopted, so as to ensure that the risks to the rights and freedoms of data subjects are indeed reduced. Therefore, the above discussion further illustrates the importance of choosing appropriate pseudonymisation techniques.

2.4.1 Pseudonymisation and encryption

It should be noted that there is often confusion among data controllers around the notions of encryption and pseudonymisation, both referenced in GDPR as security measures (article 32). However, despite some common elements, the main goals of these techniques are actually different. We will briefly discuss this difference in the next paragraphs.

With regard to pseudonymisation, it is evident from the previous discussion that it mainly focuses on protecting the *identities* of individuals (for anyone without access to additional information). Yet, pseudonymised data do provide some legible information and, thus, a third party (i.e. other than the controller or processor) may still understand the semantic (structure) of the data[22], despite the fact that these data cannot be associated with an individual.

On the other hand, encryption aims at ensuring – via appropriately utilizing mathematical techniques - that the *whole dataset that is being encrypted*[23] *is unintelligible* to anyone but specifically authorised users, who are allowed to reverse this unintelligibility (i.e. to decrypt)[24]. To this end, encryption is a main instrument to achieve confidentiality of personal data by hiding the whole dataset and making it unintelligible to any unauthorised party (as long as state-of-the-art algorithms and key lengths are used and the encryption key is appropriately protected).

Moreover, as mentioned earlier, pseudonymisation rests on decoupling, i.e. from one single input (initial dataset) to a dual output (pseudonymised data, additional information). Reversal of pseudonymisation is possible for anyone who can retrieve the additional information or who can link pseudonymised data to the initial data with the use of any other information. On the contrary, encryption, from a single input

[20] Note though that the controller has to respond to data subject requests, only with the appropriate information that the data are not identifiable, whereas prior to that, the controller has to have informed the data subjects (if possible).
[21] See also [Enisa, 2008], stating "*The right to access personal data requires some kind of identity proof so that the data are not disclosed to an unauthorised person. If a user has disclosed data under a certain pseudonym, a proof has to be given that the requesting user really is the holder of this pseudonym. This requires appropriate – data minimising – authentication mechanisms*".
[22] For instance, a statistical analysis can be performed on pseudonymous data.
[23] Note that encryption may be applied to the full dataset or specific parts of a dataset (e.g. certain fields in a database), depending on the specific protection goals.
[24] Note that there are specific cryptographic techniques allowing a third party, without knowledge of the encryption key, to perform operations on encrypted values (homomorphic ciphers) or to get some kind of information (e.g. the order-preserving encryption preserves numerical ordering of the initial plaintexts). However, the encrypted data are still unintelligible in the sense that no one can reveal the initial data.

(initial data), generates again a single output (encrypted data) and its reversal mainly lies with any unauthorised access to the decryption key[25].

However, despite the aforementioned distinction, it is important to state that encryption may also be used as a pseudonymisation technique (whereas the opposite is impossible). Cryptographic primitives can in general be used in pseudonymisation techniques to generate pseudonyms with desired properties.

[25] As long as state-of-the-art algorithms and key lengths are used.

3. Pseudonymisation techniques

This Chapter addresses some techniques that may be utilised for pseudonymisation, describing their main characteristics, advantages and/or limitations. In order to do so, Section 3.1 sets some basic design goals and discusses the topic of data separation, which is essential to the notion of pseudonymisation. The different techniques are then described, in particular hashing without key (Section 3.2), hashing with key or salt (Section 3.3), encryption as pseudonymisation technique (Section 3.4), other cryptographic techniques (Section 3.5), tokenization (Section 3.6) and other approaches (Section 3.7).

It should be noted that the various pseudonymisation techniques are presented in the context of deriving pseudonyms from initial identifiers that are associated to individuals (see also discussion on identifiers in Section 2.1.3). Although for reasons of simplicity the focus is on cases that a unique identifier associated to an individual (e.g. a device identifier) is being transformed into a pseudonym, the presented techniques can be also applied to more complex cases (e.g. where there is a combination of identifiers).

Note that the same terminology as presented in Chapter 2 is also used in this Chapter, including the notion of identifiers and pseudonyms, which are core in the descriptions that will follow.

3.1 Design goals

As mentioned earlier, pseudonymisation may contribute towards hiding an individual's real identity, as well as supporting unlinkability across different data processing domains. When examining different pseudonymisation techniques, it is important to assess whether the aforementioned purposes can be met and to what extent. In this regard, note should be taken of the fact that, as mentioned in Section 2.2, not all pseudonymisation techniques would fall under the stricter definition of GDPR, which requires that pseudonymised data can no longer be attributed to a specific data subject without the use of additional information. Therefore, the choice of the pseudonymisation technique is essential for controllers.

To this end, the following design goals can be set by the data controllers towards adopting an optimal technique, taking into account the risks of the specific data processing operation to the rights and freedoms of individuals:

- D1) the pseudonyms should not allow an "easy" re-identification by any third party (i.e. any other than the controller or processor) within a specific data processing context (so as to "hide" the initial identifiers in a specific context).
- D2) it should not be trivial for any third party (i.e. any other than the controller or processor) to reproduce the pseudonyms (so as to avoid the usage of the same pseudonyms across different data processing domains – unlinkability across domains).

The aforementioned goals are based on the assumption that a data controller will be able to re-identify the data subjects after a pseudonymisation process (as it has access to the additional information). A data processor might also have this possibility under the instructions of the controller. This is clearly not the case for third parties, against whom the data are actually protected.

There are also cases in which there is no need for the controller to associate the pseudonymised data with specific initial identifiers. For instance, a controller may only need to perform *tracking* of individuals, i.e. to be able to distinguish any individual from others within a specific processing context, without actually

having knowledge of the individual's real identity or, more generally, his or her initial identifiers[26]. Again, pseudonymisation may also be the vehicle for fulfilling such a requirement, via appropriately employing a technique to ensure that the same pseudonym will always be assigned to the same individual. As it will be discussed next, the choice of a proper pseudonymisation technique is strongly contingent on whether there is a possibility, in the context of the data processing, for the controller to refrain from storing the initial identifiers and only track the data subjects on the basis of pseudonyms.

It should also be pointed out that, as mentioned earlier, a pseudonymisation approach may also yield additional data protection gains in terms of data accuracy. This adds a third design goal, which should also be considered by data controllers. For instance, there exist pseudonymisation techniques generating pseudonyms that are mathematically bound to the initial identifiers and, thus, these pseudonyms may suffice to allow verification of data subjects' identities under specific frameworks.

Furthermore, together with the aforementioned design goals, another important aspect for controllers to consider is that of *data separation*, i.e. separation of the pseusonymised data from the additional information (the two distinct outputs of pseudonymisation). Indeed, since the notion of pseudonymisation implies an association between pseudonyms and the initial identifiers (the additional information), a mapping table or other relevant structure that allows for this association (e.g. a key, as discussed next) would probably need to be in place.

Depending on the data processing operation, the data controller may employ different security measures for the protection of additional information, like physical separation of identity/pseudonym mappings in conjunction with strict access control mechanisms and/or other security techniques. In some cases, the data processor may undertake the storage of the additional information under the instructions of the controller (although this approach could in certain implementations raise privacy concerns, especially if the security of the additional information is not under the direct control of the data controller). For special cases, a trusted third party could also be employed for the storage of the additional information (e.g. an authority providing guarantees for such a role). Finally, as also stated above, there are also cases in which the data controller needs only to track the users or where the need for re-identification occurs in special cases (i.e. for a subset of the pseudonymised data). In such cases, more sophisticated approaches may be employed for de-centralised storage of the additional information, e.g. the generation of the pseudonyms can be mounted on the user's environments, without necessitating a central point for storing the identifiers/pseudonyms mappings.

All the aforementioned aspects need to be carefully considered by the data controller before selecting a specific pseudonymisation technique. In the next Sections, we shall present some relevant techniques, assess them with regard to the above design goals (D1 and D2) and describe their relative advantages and disadvantages. Although in the descriptions and relevant examples we focus explicitly on cases that a unique identifier associated to an individual is being transformed into a pseudonym, the same techniques can be also applied to more complex cases. For example, such a case could be that of a combination of identifiers, where the initial identifiers are concatenated to form a new "generalised" identifier, which in turn will be the basis to generate the corresponding pseudonym (recall also the relevant discussion in Section 2.1.3).

3.2 Hashing without key

[26] Note that tracking stills enables singling out the specific individuals by the controller, even if the initial identifiers are not stored.

Hashing is a technique that can be used to derive pseudonyms, but, as will be shown later in this Section, has some serious drawbacks with regard to the design goals set in Section 3.1. Still, it is a starting point for understanding other stronger techniques in the field and this is why we present it first. Moreover, hashing can be a useful tool to support data accuracy.

A cryptographic hash function *h* is a function with specific properties (as described next) which transforms any input message *m* of arbitrary length to a fixed-size output *h(m)* (e.g. of size 256 bits, that is 32 characters), being called *hash value* or *message digest*.

The message digest satisfies the following properties [Menezes, 1996]: i) given h(m), it is computationally infeasible[27] to compute the unknown m, and this holds for any output h(m) - i.e. the function h is mathematically irreversible (pre-image resistance), ii) for any given m, it is computationally infeasible to find another m'≠ m such that h(m')=h(m) (2nd pre-image resistance), iii) it is computationally infeasible to find any two distinct inputs m, m' (free choice) such that h(m')=h(m) (collision resistance). Clearly, if a function is collision-resistant, then it is 2nd pre-image resistant too[28].

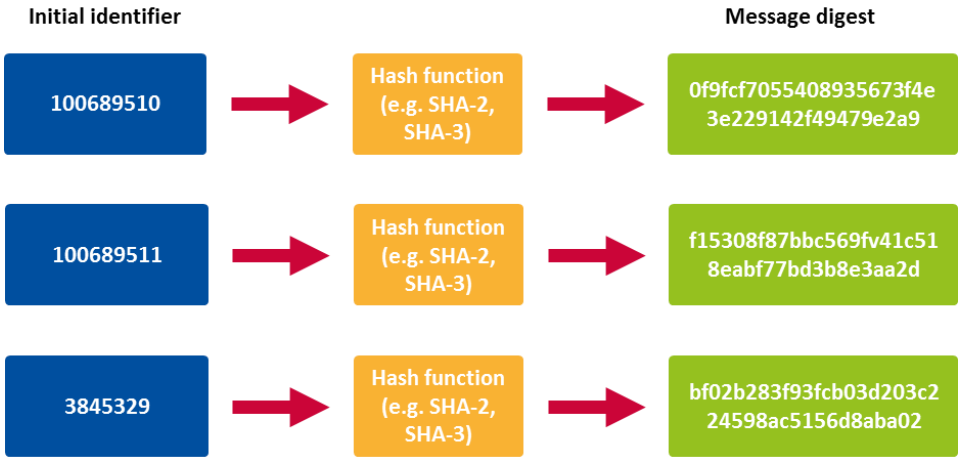

Figure 1: Operation of a cryptographic hash function

In other words, a cryptographic hash algorithm is one that generates a unique digest[29] (which is also usually called fingerprint) of a fixed size for any single block of data of arbitrary size (e.g. an initial identifier of any kind). Note that for any given hash function, the same unique digest is always produced for the same input (same block of data).

It is important to point out that state-of-the art hash functions should be chosen; therefore, commonly used hash functions such as MD5 and SHA-1 [Menezes,1996] with known vulnerabilities – with respect to the probability of finding collisions - should be avoided (see [Wang, 2005], [Dougherty,2008], [Stevens, 2017a], [Stevens,2017b]). Instead, cryptographically resistant hash functions should be preferable, e.g. SHA-2 and SHA-3 are currently considered as state-of-the-art [FIPS, 2012], [FIPS,2015].

[27] This means that the required effort exceeds by far the understood resources [Menezes,1996].
[28] Note that, since the inputs of the hash functions may be or arbitrary length whilst the outputs are of fixed length, there exist different inputs m, m' such that h(m)=h(m') – i.e. collisions do exist. However, despite their existence, a state-of-the art cryptographic hash function should not allow finding out collisions in practice (i.e. it should be computationally infeasible).
[29] Of course, if the digest is truncated, the uniqueness is not ensured.

The above properties of hash functions allow them to be used in several applications, including data integrity[30] and entity authentication[31]. For instance, once an app market has a hash server storing hash values of app source codes, any user can verify whether the source code has been modified or not via a simple validation of its hash value - since any modification of the code would lead to a different hash value (see, e.g., [Jeun, 2011]). Similarly, recalling the discussion in Section 3.1 on data accuracy, a pseudonym that is generated via hashing user's identifiers may be a convenient way for a data controller to verify a user's identity.

However, when it comes to pseudonymisation, despite the aforementioned properties of a cryptographic hash function, simple hashing of data subjects' identifiers to provide pseudonyms has major drawbacks.

More precisely, with regard to the aforementioned D1 and D2 design goals, we have the following:

- The D2 property does not hold, since any third party that applies the same hash function to the same identifier gets the same pseudonym[32].
- In relation to the above observation, the D1 property also does not necessarily hold, since it is trivial for any third party to check, for a given identifier, whether a pseudonym corresponds to this identifier (i.e. though hashing the identifier[33]).

Therefore, a reversal of pseudonymisation is possible whenever such an approach is adopted, as having a list of the (possible) initial identifiers is adequate for any third party to associate these identifiers with the corresponding pseudonyms, with no any other association being in place[34]. In fact, following the GDPR definition of pseudonymisation, one could argue that hashing is a weak pseudonymisation technique as it can be reversed without the use of additional information. Relevant examples are provided in [Demir, 2018] (and in references therein), where the researchers refer to the Gravatar service[35] and describe how users' email addresses can be derived through their hash value, which is shown in the URL that corresponds to the gravatar of the user, without any additional information.

Hence, hash functions are generally not recommended for pseudonymisation of personal data, although they can still contribute to enhancing security in specific contexts with negligible privacy risks and when the initial identifiers cannot be guessed or easily inferred by a third party. For the vast majority of the cases, such pseudonymisation technique does not seem to be sufficient as a data protection mechanism [Demir, 2018]. However, a simple hashing procedure may still have its own importance in terms of data accuracy, as stated previously.

3.3 Hashing with key or salt

[30] Data integrity is the property whereby data has not been altered in an unauthorised manner since the time it was created, transmitted, or stored by an authorised source [Menezes, 1996].
[31] Entity (or data origin) authentication is a type of authentication whereby a party is corroborated as the (original) source of specified data created at some (typically unspecified) time in the past [Menezes, 1996].
[32] This is relevant to the so-called "dictionary attacks", which are a type of brute force attacks where the attacker is trying a large number of likely possibilities (such as words in a dictionary) to gain access to data (e.g. by discovering a passphrase).
[33] This procedure is the same with the one that is being used for password cracking, since generally passwords are being stored in a hashed form.
[34] This idea is similar to the well-known rainbow attacks employing the so-called rainbow tables, which are tables containing precomputed hash values, most commonly used by hackers to recover users' passwords from their hash values (see, e.g. [Kumar, 2013]).
[35] See in: https://en.gravatar.com

A robust approach to generate pseudonyms is based on the use of keyed hash functions – i.e. hash functions whose output depends not only on the input but on a secret key too; in cryptography, such primitives are being called message authentication codes (see, e.g., [Menezes, 1996]).

The main difference from the conventional hash functions is that, for the same input (a data subject's identifier), several different pseudonyms can be produced, according to the choice of the specific key – and, thus, the D2 property is ensured. Moreover, the D1 property also holds, as long as any third party, i.e. other than the controller or the processor, (e.g. an adversary) does not have knowledge of the key and, thus, is not in the position to verify whether a pseudonym corresponds to a specific known identifier. Apparently, if the data controller needs to assign the same pseudonym to the same individual, then the same secret key should be used.

To ensure the aforementioned properties, a secure keyed-hash function, with properly chosen parameters, is needed. A known such standard is the HMAC [FIPS, 2008], whose strength is contingent on the strength of the underlying simple hash function (and, thus, incorporating SHA-2 or SHA-3 in HMAC is currently a right option). Moreover, the secret key needs to be unpredictable and of sufficient length, e.g. 256 bits, which could be considered as adequate even for the post-quantum era[36]. If the secret key is disclosed to a third party, then the keyed hash function actually becomes a conventional hash function in terms of evaluating its pseudonymisation strength. Hence, recalling the definition of pseudonymisation in the GDPR, the data controller should keep the secret key securely stored separately from other data, as it constitutes the additional information, i.e. it provides the means for associating the individuals – i.e. the original identifiers – with the derived pseudonyms.

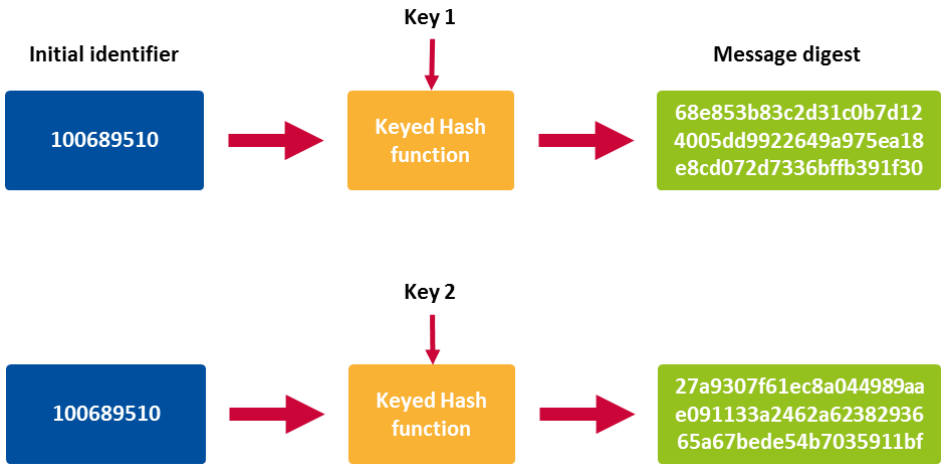

Figure 2: Operation of a keyed hash function

Keyed hash functions are especially applicable as pseudonymisation techniques in cases that a data controller needs - in specific data processing context - to track the individuals without, however, storing their initial identifiers (see also [Digital Summit, 2017]). Indeed, if the data controller applies - always with the same secret key - a keyed hash function on a data subject's identifier to produce a pseudonym, without though storing the initial user's identifier, then we have the following outcomes:

[36] Post-quantum cryptography is cryptography under the assumption that the attacker has a large quantum computer. It is currently known that a key length of 256 bits for symmetric cryptographic primitive – such as a keyed hash function – is a secure key size for post-quantum cryptography [Bernstein,2017]).

- The same pseudonym will always be computed for each data subject (i.e. allowing tracking of the data subject).
- Associating a pseudonym to the initial identifier is practically not feasible (provided that the controller does not have knowledge of the initial identifiers).

Therefore, if only tracking of data subjects is required, the controller needs to have access to the key but does not need to have access to the initial identifiers, after pseudonymisation has been performed. This is an important consideration that adheres to the principle of data minimization and should be considered by the controller as a data protection by design aspect.

Moreover, a keyed hash function has also the following property: if the secret key is securely destroyed and the hash function is cryptographically strong, it is computationally hard, even for the data controller, to reverse the pseudonym to the initial identifier, even if the controller has knowledge of the initial identifiers. Therefore, the usage of a keyed hash function may allow for subsequent anonymisation of data, if necessary, since deleting the secret key actually deletes any association between the pseudonyms and the initial identifiers. More generally, using a keyed hash function to generate a pseudonym and subsequently deleting the secret key is somehow equivalent to generate random pseudonyms, without any connection with the initial identifiers.

Another approach that is often presented as an alternative to the keyed hash function is the usage of an unkeyed (i.e. conventional) hash function with a so-called "salt" – that is the input to the hash function is being augmented via adding auxiliary random-looking data that are being called "salt". Again, if such a technique is appropriately applied, for the same identifier, several different pseudonyms can be produced, according to the choice of the salt – and, thus, the D2 property is ensured, whilst the D1 property also holds with regard to third parties provided that they do not have knowledge of the salt. Of course, this conclusion is valid only as long as the salt is appropriately secured and separated from the hash. Note that, as in the case of keyed hash, the same salt should be used by the controller in cases that there is need to assign always the same pseudonym to the same individual[37]. Moreover, salted hash functions can be utilized in cases where the controller does need to store the initial identifiers, while still being able to track the data subjects. Last, if the salt is securely destroyed by the controller, it is not trivial to restore the association between pseudonyms and identifiers.

However, it should be stressed that in several typical cases employing salts for protecting hashes has some serious drawbacks:

- On one hand, the salt does not share the same unpredictability properties as secret keys (e.g. a salt may consist of 8 characters, i.e. 64 bits, as in the cases of protecting users' passwords in some Linux systems). More generally, from a cryptographic point of view, a keyed hash function is considered as more powerful approach than a salted hash function[38]. There exist though several cryptographically strong techniques for generating salted hashes, which in turn could be considered as appropriate candidates for generating pseudonyms – a notable example being the bcrypt [Provos, 1999].

[37] Note that this is different from the case of using salts to provide hash vales of users passwords; in such a case, there is a need to generate different hash values even for users having the same passwords and, thus, different salts are being used for each user's password.

[38] There is a known attack, called message extension (or length extension) attack, which allows an attacker, once he knows a pair message-hash value (where the hash value is salted) to generate the valid salted hash value of another message without having knowledge of the secret salt (see, e.g. [Oppliger, 2015]). Such an attack is strongly contingent on the hash function that is being used – e.g. SHA-2 is vulnerable, but SHA-3 (i.e. the so-called Keccak hash function) does not have this weakness (see, e.g. https://keccak.team/keccak_strengths.html).

- Moreover, salts in most common scenarios are generally stored together with corresponding hash values, thus seriously weakening protection. The alternative use of the so-called *peppers*, which are hidden protected salts and are separately stored from hashes, can provide an enhanced alternative. A pseudonymisation approach that is based on salted/peppered hash values (namely, the case of Entertain TV) is described in [Digital Summit, 2017].

It is, therefore, recommended that salted hashes are used with caution for pseudonymisation and in accordance with available best-practices in the field.

3.4 Encryption as a pseudonymisation technique

Symmetric encryption of data subjects' identifiers is also considered as an efficient method to obtain pseudonyms. In a typical case, the original identifier of a data subject can be encrypted through a symmetric encryption algorithm (e.g. the AES, being the encryption standard [FIPS, 2001]), thus providing a ciphertext that is to be used as a pseudonym; the same secret key is needed for the decryption.

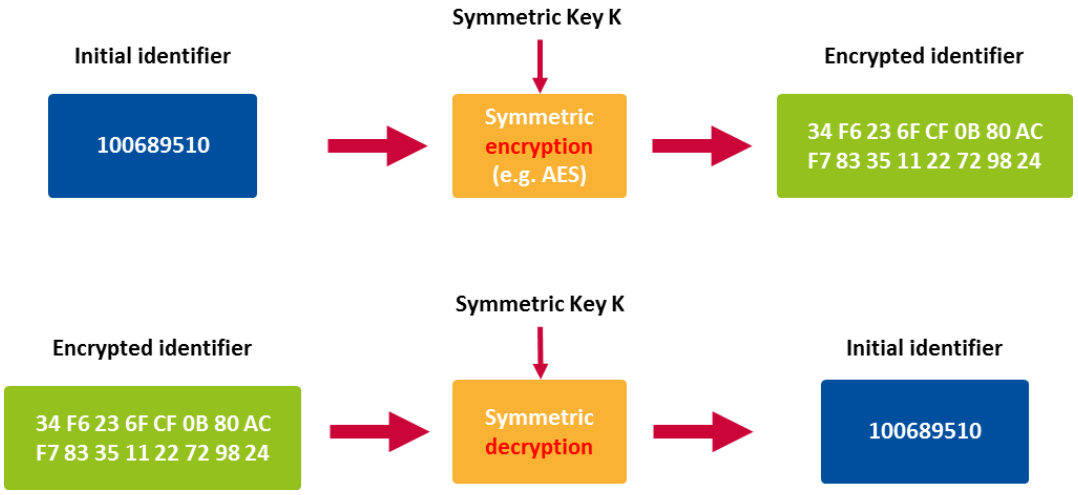

Figure 3: Operation of symmetric encryption

Such a pseudonym satisfies the D2 property, as well as the D1 property as long as no third party, i.e. any other than the controller or processor, has access to the encryption key and under the assumption that state-of-the art algorithms and sufficient lengths are used (see also in [ENISA, 2014b]). For symmetric encryption algorithms, similarly to the keyed hash functions, a key size of 256 bits is currently being considered as adequate for security, even for the post-quantum era as has been also stated above [Bernstein, 2017].

The main difference of encryption with respect to the keyed hash functions – in terms of pseudonymisation - is that the secret key owner (i.e. the data controller) may always obtain the data subjects' initial identifiers, through a simple decryption process[39]. On the contrary, as explained earlier, keyed hash functions provide the possibility to the data controllers for tracking the individuals, without

[39] Despite the fact that in some modes of operation of specific symmetric ciphers the input message needs to be appropriately modified prior to encryption (i.e. via adding padding bits), knowledge of the encrypted message and the key suffices to recover the initial message.

having knowledge (storing) of the initial identifiers. This is not the case with encryption (as pseudonymisation method), where the initial identifiers may always be known to the controller.

Aside this aspect, symmetric encryption has other similar properties – in terms of pseudonymisation - with keyed hash functions, namely: i) the same secret key should be used to provide the same pseudonym for the same identifier, ii) if the key is destroyed, it is not trivial to associate a pseudonym with the initial identifier, even if the initial identifiers are being stored by the data controller.

Hence, symmetric encryption can generally be employed (as a pseudonymisation technique) in cases that a data controller needs not only to track the data subjects but also to know their initial identifiers (see also [Digital Summit, 2017]). Traceability is grounded on a deterministic nature of the encryption method, i.e. encrypting the same identifier with the same key always yields the same pseudonym. Re-identifiability (of initial identifiers) rests, as explained above, with the very nature of symmetric encryption.

Apart from symmetric encryption algorithms, asymmetric (i.e. public key) encryption algorithms may be also used in specific cases for pseudonymisation purposes. The main characteristic of public key encryption is that each entity participating in the scheme has a pair of keys, i.e. the public and the private key. The public key of an entity can be used by anyone to encrypt data but only the specific entity can decrypt these data with the use of its private key. Although the two keys are necessarily mathematically related, knowledge of the public key does not allow determining the private key. To provide the so-called *ciphertext indistinguishability* property, the public key algorithms may be appropriately implemented in a *probabilistic form* by introducing randomness in the encryption process. This means that randomly chosen values are being used at each encryption cycle. In this way, if the same message is encrypted twice with the same public key each time, the corresponding two ciphertexts will be different, without affecting the decryption capability for the holder of the decryption key.

Due to its aforementioned properties, public key encryption may serve as an instrument for pseudonymisation in some specific contexts. For example, it might be desirable for the data controller that the entity (e.g. role or team) authorized to perform the pseudonymisation (within the same controller) is not the same with the one that is authorized to perform the re-identification. The usage of asymmetric encryption can facilitate this (i.e. by using the public key of the entity that is authorized to perform re-identification to generate the pseudonyms[40]), thus allowing for separation of duties, especially in complex or high-risk environments [Elger, 2010]. There have been relevant applications of this approach in the health sector (see, e.g., [Aamot, 2013], [Verheul, 2016]).

Moreover, as mentioned earlier, asymmetric encryption in probabilistic form allows for generating different pseudonyms for the same individual (with the same public key)[41]. Hence, it may also find application in cases where a data controller needs to assign each time a different pseudonym for the same identifier (data subject), especially when there is no need to track the data subjects (still being able to re-identify them). Note, however, that in such cases, both the public and the private key rest with the data controller (as there is no need for the public key to be accessible to other parties). Moreover, it should be stressed that appropriate implementations in symmetric ciphers may also yield probabilistic encryption (see also [Digital Summit, 2017]).

---

[40] For example in a health care environment, pseudonymisation could be performed at a front desk upon patient's registration by using the public key of the treating doctor.

[41] Probabilistic asymmetric encryption is a prerequisite towards achieving the property of chiphertext indistinguishability. For instance, the well-known RSA public key algorithm [Rivest, 1978] is deterministic and not probabilistic by definition – since encrypting the same message m with the same public key always yields the same ciphertext c – but the RSA Standard PKCS #1 v2.2 [RSA Labs, 2012] defines a scheme that renders the algorithm probabilistic.

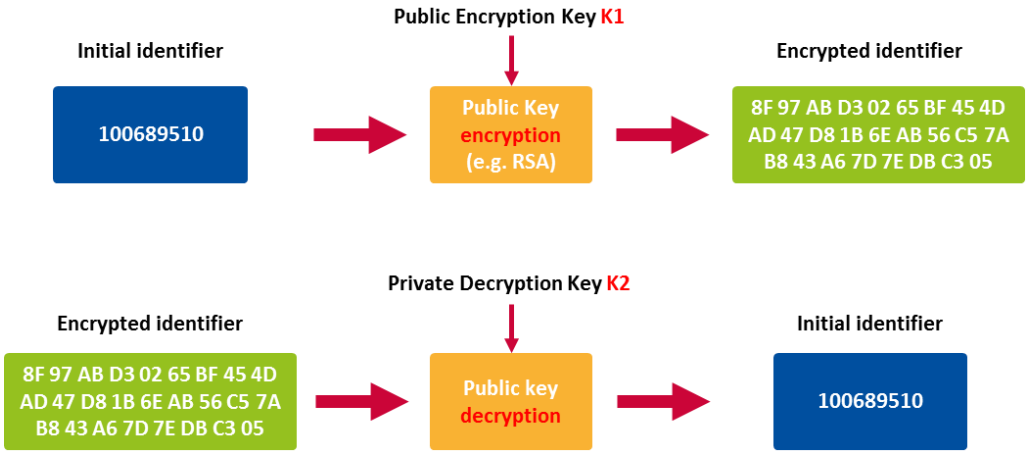

Figure 4: Operation of public key (asymmetric) encryption

The properties of asymmetric encryption may also be used in several other contexts that are related to obscuring the individuals' identities. For instance, in distributed ledger technologies[42] in which the users do not reveal their real identities, their corresponding unique addresses may be obtained through their public keys [ENISA,2016]; such a user's address allows the other users to verify his or her digital signature – that is to verify that the data have been indeed signed by the user with the claimed address.

Having said that, it should be stressed that asymmetric key algorithms necessitate the usage of very large keys, which in turn may give rise to implementation restrictions – e.g. 3072 key bits are needed in RSA (see, e.g. [Bernstein, 2017]). Even if the elliptic curve cryptography is considered, which offers much smaller key sizes than the RSA as well as faster computation (see, e.g., [Gura, 2004]), it is still less efficient than symmetric key algorithms. Moreover, one should keep in mind that the currently most known and widely used public key algorithms – including RSA and elliptic curve cryptographic algorithms - will not be strong in the post-quantum era [Bernstein, 2017][43].

3.5 Other cryptography-based techniques

Appropriate combination of several cryptographic schemes may also provide robust pseudonymisation approaches, for example by the use of techniques such as secure multi-party computation and homomorphic encryption (see [ENISA, 2014a] and the references therein). Several advanced cryptography-based pseudonymisation solutions have been proposed to alleviate data protection issues, especially in cases of personal data processing that present very high risks – e.g. in wide scale e-health systems. As a recent characteristic example, we refer to the polymorphic encryption and pseudonymisation technique proposed in [Verheul, 2016]. In this method, each user (i.e. patient in case of an e-health system) has a cryptographically generated different pseudonym at different parties. For instance, the patient has different pseudonyms at doctors X, Y, Z, and at medical research groups U, V, W – that is domain-specific

[42] A distributed ledger is a type of database that is spread across multiple sites, countries or institutions, and is typically public. Records are stored one after the other in a continuous ledger, rather than sorted into blocks, but they can only be added when the participants reach a quorum [UK Government, 2016]. Blockchain is a type of distributed ledger.

[43] As described in [Bernstein, 2017], known public key algorithms are not post-quantum resistant due to the existence of a fast quantum algorithm (the Shor 's algorithm). NIST has already initiated a process to evaluate and standardise one or more quantum-resistant public-key cryptographic algorithms (https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Post-Quantum-Cryptography-Standardization).

pseudonyms are being produced[44]. Based on these so-called polymorphic techniques, a pilot example in health sector is already in place[45].

Another important cryptographic approach for deriving pseudonyms rests with appropriately realising a decentralized solution, to allow the participating users to generate their own pseudonyms and subsequently allow them keep pseudonyms under in their own custody [Lehnhardt, 2011]. Such a design goal is not a trivial task since several crucial issues need to be resolved – e.g. the pseudonym generation process should avoid duplicates, whereas each user should be able to unambiguously prove, whenever he or she wants, that is the owner of a specific pseudonym. All these approaches necessitate the appropriate use of several cryptographic primitives (see, e.g., [Schartner, 2005], [Lehnhardt, 2011] – the latter one has been applied by a large producer of healthcare information systems located in Germany, as is stated therein). For example, the approach in [Lehnhardt, 2011] rests with the usage of public key cryptography – and, more precisely, of elliptic curve cryptography – in a way that each user computes his or her own pseudonym based on a secret that he or she acquires. These ideas introduce a fundamental property in the overall concept of pseudonymisation, since the additional information that is needed to re-identify each user is solely under the control of the user himself or herself and not of the data controller, whose role is to provide such a decentralized pseudonymisation technique. Therefore, these approaches – although costly - seem to be the best options in cases that the data protection by design principle necessitates to ensure that the data controller should not have a priori knowledge of the data subject's identity, unless the data subject chooses to prove his or her identity at any time.

As a final point, it should be noted that a common challenge for most cryptographic techniques is key management, which is usually not trivial, depending also on the overall scale of application, as well as the specific technique chosen.

3.6 Tokenisation

Tokenisation refers to the process that the data subjects' identifiers are replaced by randomly-generated values, known as tokens, without having any mathematical relationship with the original identifiers. Hence, knowledge of a token has no usefulness for a third party, i.e. any other than the controller or processor (see, e.g., [WP29, 2014]). Tokenisation is commonly used to protect financial transactions[46], but it is not limited to such applications[47].

Clearly, the tokenization system should be appropriately designed to ensure that indeed there is no mathematical relationship between pseudonyms and the original identifiers. Moreover, other restrictions should also be taken into account, depending on the context of the overall processing – e.g. if tokenisation is being used to pseudonymise credit card number in payments systems, the randomly generated tokens should not have any possibility of matching real card numbers (such a risk could possibly exist in cases of format-preserving tokenisations, i.e. in cases that the tokens have the same format with the initial data). Due to the random hidden mapping from original data to a token, it becomes evident that tokenisation satisfies both the D1 and D2 pseudonymisation properties (see Section 3.1). Since there is an entity which stores this hidden mapping (i.e. a token server in the tokenisation system), re-identification of data

[44] A somehow relevant idea is being discussed in the context of multilevel relational databases, as described in [Jajodia, 1990].
[45] See https://pep.cs.ru.nl
[46] E.g. tokenisation is mainly used to ensure compliance with several security requirements stemming from the Payment Card Industry Data Security Standard.
[47] For example it is used also in medical research enironments (see, e.g. [Elger, 2010]).

subjects by the data controller will be possible in all cases. This also includes tracking, as long as there is only one mapping for each identifier.

However, it should be noted that, despite the efficiency of tokenization, its deployment may be, depending on the context, quite challenging, e.g. synchronization of tokens across several systems may be needed in several applications. Therefore, previously mentioned approaches that employ keyed hash functions or encryption algorithms could be preferable with regard to reducing complexity and storage.

3.7 Other approaches

Several other well-known techniques, such as masking, scrambling and blurring, can also be considered in the context of pseudonymisation, having though restrictions with regard to their possible applications, whilst all of them mainly focus on pseudonymising data being at rest (i.e. data that are being stored in a file/database).

Masking refers to the process of hiding part of an individual's identifier with random characters or other data. For example, a masked credit card number may be transformed in the following way:

4678 3412 5100 5239 -> XXXX XXXX XXXX 5239

Clearly, such an approach cannot ensure that the D1 and D2 pseudonymisation properties are always satisfied. For example, masking the IP addresses of computers lying in the same LAN may allow for re-identifying the original IP address (once a third party is able to find out the entire space of the available IP addresses in this LAN). Moreover, there are also risks, if masking is not carefully designed, to assign the same pseudonym to different users, therefore potentially leading to collisions.

Scrambling refers to general techniques for mixing or obfuscating the characters. The process can be reversible, according to the chosen technique. For example, a simple permutation of characters may be such a scrambling, e.g. a credit card number may be transformed as follows:

4678 3412 5100 5239 -> 0831 6955 0734 4122

Apparently, scrambling can be considered as a simple form of symmetric encryption, which would not satisfy either the D1 or the D2 property – e.g. a simple permutation of characters may allow re-identification in specific cases (see again the previous example with the IP addresses of a LAN).

Generally, both masking and scrambling are in fact weak pseudonymisation techniques and their use is generally not recommended as a good practice in personal data processing. However, despite their limitations, they may be utilized to provide a level of protection in specific contexts (for instance, masked telephone numbers can be used for displaying, for billing purposes, the telephone calls made from business premises).

Blurring is another technique, which aims to use an approximation of data values, so as to reduce the precision of the data, reducing the possibility of identification of individuals. For instance, appropriate round functions can be used to transform numerical values into new ones. Blurring can be also applied to individuals' pictures (i.e. image obfuscation) as a part of a pseudonymisation process; recent research though illustrates that image recognition techniques based on artificial neural networks may recover the hidden information from such blurred images [McPherson, 2016].

Other known techniques that can be referenced in this context are those of barcodes, QR codes or similar methods, which, however, aim mainly towards supporting data accuracy, rather than providing a data pseudonymisation solution.

4. Pseudonymisation in the mobile ecosystem

In this Chapter we discuss some pseudonymisation best practices and examples, focusing especially in the area of mobile apps, where a large number of identifiers that can be linked to specific individuals (e.g. device or app identifiers) may be processed by several different entities (e.g. app developers, OS providers, library providers, etc.), often without the individuals' being aware of it.

For instance, in a mobile device, the following device identifiers are present [Son, 2016] and can be linked to its user:

- The International Mobile Subscriber Identity (IMSI), which is an up to 15-digit decimal identifier representing the mobile subscriber identity.
- The International Mobile Equipment Identity (IMEI), which is a 15-digit decimal identifier associated with the mobile phone.
- The Media Access Control (MAC) address, which is a 48-bit number assigned to the device 's network interface, e.g. Wi-Fi or Bluetooth.

Moreover, there are also several other identifiers owing to the corresponding operating system used in the mobile device. For example, in Android systems there is the Android ID, which is a 64-bit randomly generated number, as well as the Google Advertising ID (GAID), which is 32-digit alphanumeric identifier that is available on devices that have the Google Play service installed. Similarly, in iOS devices, there is the Unique Device IDentifier (UDID), which is a 40-character string composed from various hardware identifiers; more precisely, as stated in [Agarwal, 2013], it is based on the serial number of the device, the IMEI, the Wi-Fi MAC and the Bluetooth MAC.

Note that most of these identifiers are generally considered as permanent - an exception being the GAID, which can be reset by the user at any time[48].

As already mentioned, in the mobile ecosystem, there are several actors, which may qualify as data controllers [ENISA, 2017], as they process the individuals' (i.e. mobile app users) personal data. Even in cases, however, that these actors are not data controllers or processors, they are encouraged – according to the recital (78) GDPR - to make sure that controllers and processors are able to fulfil their data protection obligations. In all cases, pseudonymisation is an approach that can support the protection of personal data, especially taking into account the special characteristics of the mobile environment.

In the next Sections, we explore some use cases where pseudonymisation could be employed to enhance data protection in the field of mobile apps, especially by app developers/providers, library providers, as well as OS providers.

It should be stressed that our aim is not to provide a detailed implementation guide, but rather to revisit with some simple examples the earlier presented pseudonymisation techniques; by no means these solutions should be interpreted as a legal opinion on the corresponding use cases.

[48] MAC addresses may also be changed in some cases with the use of specific software; however average users are generally not expected to perform such a change.

4.1 App developers/providers

In the mobile ecosystem, the app developers are the actors that are responsible for the development of the app itself, i.e., for coding the app functionalities and requirements. They provide the app to the app providers or the end users, depending on the business model [ENISA, 2017].

In the next paragraphs, we discuss, through four pseudonymisation use cases, four relevant best practices, which could be utilised by app providers in order to enhance data protection by design. Although clearly these examples are not exhaustive, we tried to cover some typical cases, which controllers could meet in practice. Note that for simplicity we consider in all examples that the app developer is also the app provider, i.e. the data controller, which processes individuals (users) personal data in the context of the app.

Use case 1 – Tracking without storing the initial identifiers

In a social network app the users may simply observe posts from other users and comment on them or create new posts, without necessitating a login procedure (a typical case of a so-called "anonymous" social network). However, as described also in Section 2.2, despite the fact there is no login procedure, the app provider still needs to keep track of a user's device (e.g. on the basis of a device identifier), so as to send him or her notifications whenever somebody likes and/or comments to his/her posts. However, although tracking is needed, the app provider does not actually need to know the specific device identifier (as long as this can be singled out from all other identifiers). Note also that there is no need for this app to share the same user/device identifier with other apps[49].

Clearly, in this case, simply using a permanent device identifier to track the user, may potentially lead to the identification of the user through the identification of his/her device. The situation is slightly improved if a non-permanent identifier is used (e.g. GAID in Android devices), but again identification is possible within certain time limits. Simple hashing of such identifier would not offer significant protection, as anyone with knowledge of the device identifier will be trivially able to re-identify the device (and, hence, possibly the user). Moreover, in all cases, the app provider would need to store the aforementioned device identifiers, although this is not needed for the purpose of the specific processing operation.

To this end, pseudonymisation can greatly support data protection in this scenario if properly implemented by design. Indeed, a possible approach would be to use a keyed-hash function on a non-permanent identifier for creating pseudonyms that can be used in the place of the initial identifiers. In this way, the app provider would also not need to store the initial identifiers, whilst the corresponding secret key for the hashing should be securely kept in a different database from the one that the pseudonyms are being stored. Moreover, the transmission of the identifier to the app server should be done over a secure channel – e.g. via the Transport Layer Security (TLS) protocol - so as to ensure that network eavesdroppers cannot capture the identifiers in transit and, hence, cannot by any means associate them with the corresponding pseudonyms. Yet, the TLS protocol also ensures[50] that the device is actually connected to the legitimate app server, which is necessary for both privacy and security purposes.

Use case 2: Protecting credentials in a database

[49] It should be stressed, however, that this is also dependant on the OS platform used.
[50] This is an intrinsic property of the TLS protocol, based on the usage of digital certificates. There exist though attacks aiming to compromise the TLS certificate infrastructure; to thwart such attacks, apps should use certificate pinning (or preinstalled keys) [ENISA, 2017].

Let us consider a mobile app that monitors user's footsteps and stores this information (measurement data) in the app's server, so that the user is able to access it through Internet from any device. For simplicity, we assume that the app simply counts the number of user's steps, without combining these data with any other data about the user (e.g. from other apps) or sending the data to any other recipient. Still, even in this simple case, the app provider builds a profile of the user with regard to his or her daily walking habits. The user is authenticated to the app server, for accessing his/her data, with a combination of an e-mail address and a password. Thus, the app provider can clearly identify the user, since each registered user should be able to access explicitly his/her specific user profile.

In this use case, we will explore the possibility to use pseudonymisation in order to protect the users' credentials in the app's server (database). A simple hash function on the user's name or email address is clearly again not a proper pseudonymisation approach. On the contrary, a keyed or salted hash function could be used. The corresponding key/salt, as well the original identifiers, should be securely stored and separated from the database with the pseudonymised data, e.g. in trusted authentication server. Alternatively, the pseudonyms may be produced by applying a deterministic symmetric cipher, such as the AES; again, the encryption key – which coincides with the decryption key in this case – should be securely kept separately.

Note that, after the application of such a pseudonymisation process and depending on the scale and specific characteristics of the database of pseudonymised data (lifestyle data), such database could be used for statistical purposes, even from a third party.[51] Indeed, as long as this party does not have access to the secret key/salt, it is not trivially possible to identify the users. In the same line, if a breach occurs in this pseudonymised database, re-identification will be computationally hard.

In any case, the pseudonymised database should not be correlated with any other device identifier that the app developer possibly processes – e.g. for providing personalised app configurations that the user chooses; actually, another pseudonymisation process may occur for pseudonymising any such identifier (see Use case 1).

Use case 3: Multiple pseudonyms for the same data

A smart meter is an electrical meter that records consumption traces of a household and sends them to the corresponding electricity supplier. Such traces are being used for billing purposes by the supplier (data controller). The users (electricity consumers) are able, via a relevant mobile app, to check information on their energy usage in real time.

To alleviate privacy risks with regard to the profiling of household 's habits (that can be derived through the smart meter's operation)[52], one possible option could be that the supplier stores consumption traces in pseudonymised form, in a way that different pseudonyms are being assigned to each different measurement stemming from the same household (consumer). Hence, for a given consumer, his or her traces are stored under the pseudonym A in one time interval, under a different pseudonym B in the next time interval and so on. To satisfy such a property, a probabilistic encryption scheme (e.g. as described in Section 3.4) could be a possible pseudonymisation approach.

Use case 4: Local generation of pseudonyms

A smart app provides monitoring of a driver's behavior. In particular, whenever the driver (user of the app) keeps the application active, a profiling of his/her driving habits is being built (and stored by the

[51] The overall context is important to determine this option, as especially in small/specialised datasets the possibilities of inference of personal data increase (see also relevant discussion in Section 2.1.2).
[52] Note that pseudonymisation in this case is relevant to the overall smart meter's implementation but is also applicable to the operation of the user's app.

app provider). By default, the app provider does not associate the data with any other data of the user's device and does not automatically send the data to any recipient. An optional function of the app, allows the user to authorize transfer of data by the app provider to affiliated insurance companies (e.g. in order for the user to get a discount rate). In the general case, although the app provider needs to be able to track the user (driver), so as to deliver the data that are relevant to him or her in his/her specific device, there is no need for the provider to know the real identity of the user. Such type of identification will only be needed whenever the user explicitly authorizes the provider to send his or her data to an insurance company.

Pseudonymisation can clearly support this scenario too. A possible data protection by design solution rests with allowing the user to generate a pseudonym in his/her device, in a way that nobody else can re-identify him/her, unless the user allows it – e.g. through appropriately encrypting user identifiers in a way that only the user has access to the decryption key (i.e. a passphrase). Of course, appropriate security mechanisms should be put in place in this approach; for instance, the secret key/passphrase should not be shipped in the app. Moreover, the pseudonym generated by the app in the users ' device should be transmitted encrypted to the app server – e.g. via the TLS protocol – and uncorrelated from any other device identifier. Note also that there exist specialized cryptographic techniques (see Section 3.5) that allow a user to generate a pseudonym locally in his/her environment, without necessitating exchange of information with issuing parties, such that he/she can prove at any time that he/she is the owner of the pseudonym (see, e.g., [Schartner, 2005]).

4.2 Library providers

The usage of third-party libraries by mobile apps raises several privacy concerns [Grace, 2012], owing to the fact that library providers (e.g. ad providers) are able to execute code on users' devices with the same permissions as the host applications; this in turn results in collecting personal data [ENISA, 2017]. Hence, the owners of the libraries may build detailed users profiles by combining the data they collect from different mobile apps that are using the same app. Such a threat is also known as *intra-library collusion* and rests with processing globally unique identifiers through different apps with (possibly) different permissions installed in the same device [ENISA, 2017] [Taylor, 2017].

Pseudonymisation, in combination with other privacy enhancing mechanisms, could possibly be used to limit the above-mentioned issue. In this direction, it is essential that each library provider associates a different identifier per application, even for the same device. To this end, such a unique identifier may be obtained though, e.g., the following calculation [Stevens, 2012]:

hash(library provider || app identifier || device ID)

Such a computation allows for deriving a different identifier, for the same library provider and the same device, across different applications (and, of course, a different app identifier, for the same device, across different library providers). Moreover, a non-permanent device identifier (i.e. a user-resettable identifier) can be also used in cases that the OS supports such an option, which further enhances the privacy of the user (see Section 4.3).

Another issue that is associated with the processing of unique identifiers by library providers is that any unauthorised party (e.g. an adversary) who simply monitors the network may be able to build user profiles via associating such unique identifiers. This is especially relevant when libraries (e.g. ad-providers) APIs, embedded in mobile apps, send user information over the Internet in clear text [Chen, 2014]. An approach to alleviate such a concern is to secure – i.e. to encrypt – all communications between the user's device and the library provider; by these means, the adversary will not be able to correlate network traffic

corresponding to the same device [Stevens, 2012] [Chen, 2014][53]. Note that simply hashing a device identifier, without encryption, does not solve this issue; for example, even if an ad provider hashes the Android ID, other ad providers may still transmit it in plaintext and, thus, a correlation between the Android ID and its hash value is trivial.

4.3 Operating system providers

Operating system (OS) providers play a central role in mobile apps users' privacy, as several aspects with regard to the processing of personal data (e.g. permissions model) are platform dependent. To this end, an OS provider, towards supporting the data protection by design principle, could adopt specific approaches to facilitate pseudonymisation techniques, e.g. whenever this can promote data minimisation and in combination with other privacy-enhancing measures. As stated in previous Sections, a major source of privacy risks is the usage of permanent device identifiers by mobile apps developers/providers and/or library providers. Therefore, the OS providers should put effort to impede such a processing.

In this direction, the OS providers can restrict applications and third parties from accessing the permanent unique device identifiers via providing non-permanent software-based identifiers (it should be pointed out that the most recent versions of the popular operating systems follow this approach with respect to the tracking purposes from third parties[54]). Such identifiers are suitable for user tracking only to a limited extent. Ideally, it is essential to differentiate the identifier per app and per user. By these means, pseudonymisation of such identifiers (e.g. by app providers) can lead to stronger protection of personal data, provided that the knowledge of the non-permanent identifiers does not allow a computation of a permanent device ID. This in turn means that special emphasis should be given on how to appropriately generate these non-permanent IDs. For instance, it has been recently shown that commonly used MAC randomisation techniques may be inappropriate in case that specific, well-determined, best practices are not adopted[55] [Vanhoef, 2016], [Martin, 2017]. In the same line, the OS providers should make reasonable efforts that apps will be rejected during the review process (i.e. if the OS provider runs an app store [ENISA, 2017]) in case that they misuse the device identifiers[56].

In addition, there are also further options for the OS providers to facilitate the development of efficient pseudonymisation techniques. Recalling the issues discussed in Section 4.2, a decoupling of application and advertising permissions seems to be a proper design principle, which is unfortunately not the case for all famous OS; for instance, certain versions of the Android security model do not support the separation of privileges between apps and their embedded libraries [Spensky, 2016]. As also stated in [Stevens, 2012], third-party code should not be allowed to access application-specific data unless the user provides his/her explicit informed consent (in the app containing the third-party code).

[53] In a recent research work [Taylor,2018], it is shown that smartphone apps can be fingerprinted and later identified by analysing the encrypted network traffic coming from them via exploiting side-channel information such as packet size and direction.
[54] See for example in [Chen, 2014], [Kurtz, 2016].
[55] Note, however, that in typical scenarios an app does not have access to MAC address.
[56] To the extent that the OS provider is able to determine such misuse by the app.

5. Conclusions and recommendations

Pseudonymisation is an established and accepted de-identification process that has gained additional attention following the adoption of the GDPR, where it is referenced as both a security and data protection by design mechanism. As a result, in the GDPR context, pseudonymisation can motivate the relaxation to certain degree of data controllers' legal obligations if properly applied. In this report, we presented an overview on the notion and main techniques of pseudonymisation in correlation with its new role under GDPR.

Although the report does not seek to conduct a detailed analysis of the different aspects related to specific pseudonymisation methods and implementations, it touches upon some of the key issues in this regard. However, further research is needed, as well as practical experience, involving all stakeholders in the field. In this way, our work does not aim to conclude but rather to *initiate* a broader discussion on pseudonymisation under GDPR and its potential application in different scenarios, especially concerning best-practice techniques, use cases and practical examples.

In the following, we present our main conclusions to this end, together with specific recommendations for relevant stakeholders.

Pseudonymisation as a core data protection by design strategy

Pseudonymisation is clearly a process that can contribute towards data protection by design, especially by technically supporting a broader interpretation of the notion of data minimisation in the digital world[57]. As an example, the potential use of pseudonymisation has been discussed, in cases where the data controller (while still being able to deliver a specific service) does not need to store the initial user identifiers. Such interpretation can greatly contribute towards the privacy-friendly operation of online systems and services, not only in the private, but also in the public sector (e.g. e-voting or e-petition systems). This approach, however, is highly relevant to the adoption by controllers of appropriate data protection by design frameworks, where data minimisation, also by means of pseudonymisation, is a core strategy.

Data controllers, as well as producers of products, services and applications, should adopt data protection as a key design approach in their processes; doing so, they should reassess their possibilities of implementing data minimisation by applying proper data pseudonymisation techniques.

Regulators (e.g. Data Protection Authorities and the European Data Protection Board) should promote the use of pseudonymisation as a core data protection by design strategy by further elaborating on its role under GDPR and providing relevant guidance to controllers.

Defining the state-of-the-art

Yet, the technical implementation of pseudonymisation is highly dependent on the state-of-the-art and the way that this is known and/or available to controllers. While not all pseudonymisation techniques are equally effective, there might be certain implementation challenges or limitations with regard to each technique. This is not only relevant to the choice of the technique itself, but also to the overall design of the pseudonymisation process, including especially the protection of the additional information (i.e. the information that allows for the association between pseudonyms and initial identifiers). The combination

[57] See also relevant analysis in [Gurses, 2015].

of pseudonymisation with other privacy enhancing technologies is also critical to enhance overall efficiency.

The research community should continue working on privacy and security engineering, including state-of-the-art pseudonymisation (and anonymisation) techniques and their possible implementations, with the support of the EU institutions in terms of policy guidance and research funding.

Pseudonymisation best practices in the context of GDPR

Clearly pseudonymisation is not a prerequisite for all cases of personal data processing; hence, evaluating the relevant data protection risks (for each specific data processing case) is inherent to the decision on whether and how pseudonymisation can be implemented. Defining the goals and objectives of pseudonymisation in each particular case is central in this process. To this end, relevant best practices and examples of pseudonymisation in the context of GDPR can be of great value to controllers (as well as to producers of products, services and applications). For instance, it would be beneficial to point out any successful implementation, in the private or public sector, analyzing its key attributes, as well as the possibilities of data controllers to utilize the same model in the future.

Regulators (e.g. Data Protection Authorities and the European Data Protection Board) should discuss and promote good practices across the EU in relation to state-of-the-art solutions of pseudonymisation under GDPR. EU Institutions could promote such good practices.

The research community should work out best practices out of the pooled experience on pseudonymisation (and anonymisation) at DPAs level.

Transparency and well established procedures

As already mentioned, GDPR provides certain relaxation of some controllers' obligations when pseudonymisation is applied. Moreover, the controllers are exempted from their obligations with regard to certain data subjects rights (articles 15-20 GDPR) when they are provably not in position to identify the data subjects. As this is a significant aspect of the GDPR's implementation, further guidance (on the regulators side) and good management (on the controllers) side is essential.

Regulators (e.g. Data Protection Authorities and the European Data Protection Board) should provide guidance and best practices on the interpretation and practical implementation of the aforementioned provisions.

Data controllers should establish well-determined procedures to this end, as well as share information regarding pseudonymisation methods applied (and their overall data processing activities).

6. References

H. Aamot, C. D. Kohl, D. Richter, P. Knaup-Gregori, "*Pseudonymization of patient identifiers for translational research*", BMC Medical Informatics and Decision Making, vol. 13, pp. 1-15, 2013.

Article 29 Working Party, "Opinion 04/2007 on the concept of personal data", 2007. (WP29, 2007)

Article 29 Working Party, "Opinion 02/2013 on apps on smart devices", 2013. (WP29, 2013)

Article 29 Working Party, "Opinion 05/2014 on anonymisation techniques", 2014. (WP29, 2014)

Article 29 Working Party, "Guidelines on Personal data breach notification under Regulation 2016/679", 2018. (WP29, 2018)

Y. Agarwal and M. Hall, "*Protectmyprivacy: detecting and mitigating privacy leaks on ios devices using crowdsourcing,*" in Proceeding of the 11th annual international conference on Mobile systems, applications, and services, pp. 97–110, ACM, 2013.

D. J. Bernstein and T. Lange, «*Post-quantum cryptography – dealing with the fallout of physics success*», Cryptology ePrint Archive, 2017.

V. Chatzistefanou and K. Limniotis, "*On the (non-)anonymity of anonymous social networks*", E-Democracy – Privacy-Preserving, Secure, Intelligent E-Government Services, Communications in Computer and Information Science, Springer, vol. 792, pp. 153-168, 2017.

T. Chen, I. Ullah, M. A. K âafar, R. Boreli, "*Information leakage through mobile analytics services*". HotMobile 2014, pp. 15:1-15:6, 2014.

L. Demir, A. Kumar, M. Cunche and C. Lauradoux, "*The pitfalls of hashing for privacy*", IEEE Communications Surveys and Tutorials, vol. 20, no. 1. pp. 551-565, 2018.

Digital Summit 's Data Protection Focus Group, "White Paper on Pseudonymization", 2017.

C. R. Dougherty, *"Vulnerability Note VU#836068 - MD5 vulnerable to collision attacks",* Vulnerability notes database, CERT Carnegie Mellon University Software Engineering Institute, 2008.

P. Eckersley, "How Unique Is Your Web Browser?", PETS 2010, pp. 1-18, 2010.

B. S. Elger, J. Iavindrasana, L. L. Iacono, H. Müller, N. Roduit, P. Summers and J. Wright, "Strategies for health data exchange for secondary, cross-institutional clinical research", Computer Methods and Programs in Biomedicine, Elsevier, vol. 99, pp. 230-251, 2010.

ENISA, "Algorithms, key sizes and parameters", 2014. (Enisa, 2014b)

ENISA, "Distributed Ledger Technology and Cybersecurity", 2016.

ENISA, “Privacy and data protection in mobile applications - A study on the app development ecosystem and the technical implementation of GDPR”, 2017.

ENISA, “Privacy and Data Protection by Design – from policy to engineering”, 2014. (Enisa, 2014a)

ENISA, “Privacy by design in big data”, 2015.

ENISA, “Technology-induced challenges in privacy and data protection in Europe”, 2008.

FIPS, Federal Information Processing Standards Publication 197, “*Advanced Encryption Standard ”*, 2001.

FIPS, Federal Information Processing Standards Publication 198-1, “*The Keyed-Hash Message Authentication Code (HMAC) ”,* 2008.

FIPS, Federal Information Processing Standards Publication 180-4, “*Secure Hash Standard*”, 2012.

FIPS, Federal Information Processing Standards Publication 202, “*SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions*”, 2015.

M. C. Grace, W. Zhou, X. Jiang, A.-R. Sadeghi, “*Unsafe exposure analysis of mobile in-app advertisements*”, WISEC 2012, pp. 101-112, 2012.

S. Gurses, C. Troncoso and C. Diaz, “Engineering privacy by design reloaded” Amsterdam Privacy Conference, 2015.

N. Gura, A. Patel, A. Wander, H. Eberle and S. C. Shantz, “*Comparing elliptic curve cryptography and RSA on 8-bit CPUs*”, In International workshop on Cryptographic Hardware and Embedded Systems, Springer, pp. 119-132, 2004.

ISO, ISO/IEC 20889:2018, Privacy enhancing data de-identification terminology and classification of techniques, ISO, Geneva, ISO, Geneva, Switzerland, 2018.

ISO, ISO/TS 25237:2017, Health Informatics — Pseudonymization. ISO, Geneva, Switzerland, 2017.

S. Jajodia and R. Sandhu, “*Polyinstantiation integrity in multilevel relations*”. In Proc. Of IEEE Computer Society Symposium on Research in Security and Privacy, pp. 104-115, 1990.

I. Jeun, K. Lee and D. Won, “*Enhanced Code-Signing Scheme for Smartphone Applications*”, In: Kim T. et al. (eds.), Future Generation Information Technology, FGIT 2011, Lecture Notes in Computer Science, vol. 7105. Springer, Berlin, Heidelberg, pp. 353-360, 2011.

H. Kumar, S. Kumar, R. Joseph, D. Kumar, S. K. S. Singh and P. Kumar, “*Rainbow table to crack password using MD5 hashing algorithm*”, In IEEE Conference on Information & Communication Technologies (ICT), pp. 433-439, 2013.

A. Kurtz, H. Gascon, T. Becker, K. Rieck and F. C. Freiling, “*Fingerprinting Mobile Devices Using Personalized Configurations*”, PoPETs 2016 (1), pp. 4-19, 2016.

J. Lehnhardt and A. Spalka, "*Decentralized Generation of Multiple, Uncorrelatable Pseudonyms without Trusted Third Parties*", In: Furnell S., Lambrinoudakis C., Pernul G. (eds.), Trust, Privacy and Security in Digital Business (TrustBus) 2011, Lecture Notes in Computer Science, vol. 6863, pp. 113-124, Springer, Berlin, Heidelberg, 2011.

J. Martin, T. Mayberry, C. Donahue, L. Foppe, L. Brown, C. Riggins, E. C. Rye and D. Brown, "*A Study of MAC Address Randomization in Mobile Devices and When it Fails*", PoPETs 2017 (4), pp. 365-383, 2017.

Richard McPherson, R. Shokri, V. Shmatikov, "*Defeating Image Obfuscation with Deep Learning*", CoRR abs/1609.00408, 2016.

A. J. Menezes, S. A. Vanstone, and P. C. V. Oorschot, Handbook of Applied Cryptography, CRC Press, 1996.

NIST, "Guide to Protecting the Confidentiality of Personally Identifiable Information (PII)", Special Publication (NIST SP) - 800-122, 2010.

N. Provos and D. Mazieres, "*A future-adaptable password scheme*", In Proceedings of USENIX annual technical conference, Monterey, 1999.

R.L. Rivest, A. Shamir, and L. Adleman, "*A Method for Obtaining Digital Signatures and Public-Key Cryptosystems*", Communications of the ACM, vol. 21, no. 2, pp. 120-126, 1978.

R. Oppliger, "Contemporary Cryptography", Artech House Publishers, 2005.

A. Pfitzmann and M. Hansen, "A terminology for talking about privacy by data minimization: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management", 2010.

RSA Laboratories, "*PKCS#1 v2.2:RSA Cryptography Standard ",* 2012.

P. Schartner and M. Schaffer, "*Unique User-Generated Digital Pseudonyms*", Computer Network Security, MMM-ACNS 2005, Lecture Notes in Computer Science, vol. 3685, pp. 194-205, Springer, Berlin, Heidelberg, 2005.

C. Spensky, J., Stewart, A. Yerukhimovich, R. Shay, A. Trachtenberg, R. Housley, and R. K. Cunningham, "*SoK: Privacy on Mobile Devices – It's Complicated*", Proceedings on Privacy Enhancing Technologies ; 2016 (3):96–116, 2016.

R. Stevens, C. Gibler, J. Crussell, J. Erickson and H. Chen, "*Investigating User Privacy in Android Ad Libraries*", In Workshop on Mobile Security Technologies (MoST), page 10, 2012.

M. Stevens, E. Bursztein, P. Karpman, A. Albertini, and Y. Markov, "*The First Collision for Full SHA-1*", Crypto 2017, Lecture Notes on Computer Science, Springer, vol. 10401, pp. 570-596, 2017. (Stevens, 2017a)

M. Stevens and D. Shumow, "*Speeding up detection of SHA-1 collision attacks using unavoidable attack conditions*", USENIX Security Symposium, pp. 881-897, 2017. (Stevens, 2017b)

S. Son, D. Kim, and V. Shmatikov, "*What Mobile Ads Know About Mobile Users*", Network and Distributed System Security Symposium, 2016.

Su, J., Shukla, A., Goel, S. and Narayanan, A. (2017) '*De-anonymizing web browsing data with social networks*', WWW '17, pp.1261–1269, 2017.

V. F. Taylor, A. R. Beresford and I. Martinovic, "*Intra-Library Collusion: A Potential Privacy Nightmare on Smartphones*", CoRR abs/1708.03520, 2017.

V. F. Taylor, R. Spolaor, M. Conti, and I. Martinovic, "*Robust Smartphone App Identification via Encrypted Network Traffic Analysis*". IEEE Trans. Information Forensics and Security, vol. 13, no. 1. pp. 63-78, 2018.

UK Government, Office for Science, "*Distributed Ledger Technology: beyond block chain",* 2016.

M. Vanhoef, C. Matte, M. Cunche, L. S. Cardoso and F. Piessens, "*Why MAC Address Randomization is Not Enough: An Analysis of Wi-Fi Network Discovery Mechanisms*," in Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security, pp. 413–424, 2016.

E. Verheul, B. Jacobs, C. Meijer, M. Hildebrandt, J. Ruiter, "*Polymorphic Encryption and Pseudonymisation for Personalised Healthcare – A Whitepaper*", Cryptology ePrint Archive, Report 2016/411, 2016.

X. Wang and H. Yu, "How to Break MD5 and Other Hash Functions", EUROCRYPT 2005, Lecture Notes in Computer Science, Springer, vol. 3494, pp. 19–35, 2005.

X. Zhou, S. Demetriou, D. He, M. Naveed, X. Pan, X. Wang, C. A. Gunter, and K. Nahrstedt, "*Identity, location, disease and more: Inferring your secrets from Android public resources*", in ACM CCS 2013.

ENISA

European Union Agency for Network
and Information Security
1 Vasilissis Sofias
Marousi 151 24, Attiki, Greece

Heraklion Office

Science and Technology Park of Crete (ITE)
Vassilika Vouton, 700 13, Heraklion, Greece

Catalogue Number TP-06-18-398-EN-N

1 Vasilissis Sofias Str, Maroussi 151 24, Attiki, Greece
Tel: +30 28 14 40 9710
info@enisa.europa.eu
www.enisa.europa.eu

ISBN: 978-92-9204-281-3
DOI: 10.2824/74954