A Generalized View on Pseudonyms and Domain Specific Local Identifiers

Lessons Learned from Various Use Cases

Uwe Roth
SANTEC
CRP Henri Tudor
L-1855 Luxembourg, Luxembourg
uwe.roth@tudor.lu

Abstract—**Pseudonymisation as a data privacy concept for medical data is not new. The process of pseudonymisation gets difficult in concrete use-case setups and the different variations of data flow between those who collect, who store, and who access the data. In all cases, questions have to be answered about, who has access to the demographics of a person, who has access to the pseudonym, and finally, who creates the pseudonym. Since a fundamental part of the pseudonym creation depends on the identification of a person on base of its demographics, things even get more difficult in case of unclear matching decisions, management of wrong matching or update of demographic information. In this journal article, a unified view on pseudonyms is proposed. Pseudonyms are treated as a local identifier in an identifier domain, but in a domain that has no demographics. Additionally, persistent identifiers are introduced that allow the handling of updates and internal matching reconsiderations. Finally, two concepts for pseudonymisation are shown: First, a National Pseudonymisation Service is sketched with focus on resistance against update problems and wrong matching decisions. It is designed to cover every possible variation of the exchange of local identifiers between a source of personal data and the storage destination. Second, an algorithm for the pseudonym creation from a person identifier is described. This algorithm is needed if the pseudonymisation is not performed by an external service but in-house and in case of limited number space of the pseudonyms. Both solutions are suitable to solve a huge variety of pseudonymisation setups, as it is demanded by researchers of clinical trials and studies.**

Keywords-patient privacy-enhancing technologies; secure patient data storage; pseudonymisation; local identifer; identifier domain.

I. INTRODUCTION

This article is an extended version of [1], which covers the algorithm for the generation of pseudonyms with a limited number of bits.

Pseudonymisation is a process where demographics and identifier of a person are removed out of an information record and replaced by a pseudonym. This step is demanded to protect the privacy of patients in cases of secondary usage of medical data, e.g., for research or statistical purposes. In these cases knowledge about the identity of the person is unnecessary and therefore must be protected against disclosure. In contrast to anonymisation, a pseudonym allows to link data from several sources to the same person, which helps to improve the quality of the research or statistics.

An example for the need of pseudonymization is the storage of medical data, samples, blood, and urine in biobanks. Researchers are not interested in the identity of the person behind this material. A pseudonym is needed to link all samples that have been taken from the same person at different locations and during different collection events. The pseudonym will not only allow the linkage to the same person but also allows protecting the identity of the patient behind the sensitive data.

One part of this article describes a generalized concept on how identities of patients and their pseudonyms are used and managed (including identity matching, linkage of identifiers from different domains) to securely exchange data. Despite the fact that these problems are discussed in many publications (e.g., [2] and [3]) this article gives a generalized overview of how a source-destination relation can be defined.

The main idea behind the generalization is the concept of local identifiers of identifier domains that are either bound to demographics or not. With the generalization of pseudonyms as local identifiers in a domain without demographics, transitions of identifiers between certain identifier domains become only a matter of permissions, e.g., permission to pseudonymise, permission to re-identify. So the main cases that are discussed in the article differentiate the variations of visibility of demographics, local identifiers and pseudonyms amongst the source of data and the destination storage.

All cases can be implemented by the use of a pseudonymisation service as a trusted third party. The article defines the fundamental services of the pseudonymisation service that are needed to treat all identified cases. They have been specified for the National Pseudonymisation Service of Luxembourg, which is solely responsible for the management of persons and the transition of the identifiers between the different identifier domains. The National Pseudonymisation Service will not perform pseudonymisation on medical data, nor will it have access to medical data.

With the provisioning of demographics in a certain domain (e.g., hospital, laboratory), the introduction of faulty data is likely. The update of such data might lead to a revised decision at the National Pseudonymisation Service, i.e., demographics from a certain domain now match a different

known person or it is assumed that the persons is unknown yet. This has consequences at the destination side and requires an update of the pseudonym for some of the stored data. With the introduction of persistent identifiers that are linked to the initial matching decision, update of only the pseudonyms that are concerned is possible.

Central or national pseudonymisation services run as Trusted Third Parties for example in the Netherlands (ZorgTTP [4]), and in Germany the Patient Identifier (PID) generator in combination with a pseudonymisation service of TMF (Telematikplattform für Medizinische Forschungsnetze e.V.) is well known [5]. These solutions mainly provide global person identifiers for identified persons, which can be used to create domain specific pseudonyms. Mechanisms and information to handle faulty matching decisions after the update of demographics are not foreseen.

In the TMF solution, the visibility of the demographics and the pseudonym at source and destination are restricted by passing the (encrypted) medical data, together with the global identifier (from the PID generator) through the pseudonymisation service. Such a setup on national level would require, that the National Pseudonymisation Service must be able to access services in the research domains to push the pseudonymized data to it. As a consequence, researchers need to maintain a service in their Demilitarized Zone (DMZ) that is able to receive the pseudonymized data. In the proposed solution, the pseudonymisation service acts only as a passive service that can be accessed from Intranets without the need of a DMZ. Additionally the solution does not need to bypass medical data and therefore is able to manage more requests per time.

An alternative to the use of a National Pseudonymisation Service is the implementation of a local in-house pseudonymization, which means that the pseudonym is calculated either at the data source or the storage destination out of a given person identifier without the use of an external service. In such a setup no matching decisions will take place and a person requires a stable person identifier.

In both cases (National Pseudonymisation Service or in-house pseudonymisation) the pseudonym number itself has to be calculated or determined at one point in time. There are several options to create a pseudonym with a given set of demographics. Some of these techniques base on hashing or encryption of a unique identifying number of the person. Others simply chose a random number and link this number with the identity.

Current hashing and encryption algorithms work with 128 bits minimum, which might be too much in some cases, e.g., the pseudonym must be 31 bit unsigned integer. In that case, the outcome of the process must be cropped to the desired bit-length, which leads to an unpredictable risk for pseudonym collisions.

Research that takes smaller number of bits into account is known as small-domain pseudo random permutation or small-domain cipher (e.g., [6][7][8]). Solutions that base on this research use techniques that are also used in symmetric encryption (e.g., Advanced Encryption Standard AES [9]) or hashing algorithms (e.g., Secure Hash Algorithm SHA [10]): Permutation, rotation, transformation, and diffusion of the given bit-set of data. A similar research area that uses the same tools deals is Format Preserving Encryption (FPE) (e.g., [11]). Here, more focus is made on the format of the encrypted block of data, which also includes the format on char- or word-level. The FALDUM Code [12] as another example tries to create a code with error correction properties and good readability.

For all proposals, it is difficult to estimate how secure these algorithms finally are and how difficult it is to re-compute the person identifier with a given pseudonym. Cryptanalysis on existing symmetric encryption algorithms and hashing algorithms have shown, that weaknesses can be found years after the algorithms has been proposed (e.g., [13]).

Therefore, an alternative pseudonym calculation algorithm is proposed to calculate pseudonyms from a person identifier on the base of a chosen primitive root of a fixed prime number. This calculation is more similar to asymmetric encryption techniques (e.g., the RSA algorithm [14]) or the Diffie-Hellman-Key Exchange protocol [15].

The algorithm guarantees a collision free pseudo-random distribution of the pseudonyms. The pseudonymisation algorithm acts as a one-way function if all of the calculation parameters are kept secret.

The article is structured as follows:

In *Section II – Methods*, the concept of identifier of persons and identifier domains and its relation to pseudonyms is introduced. Later, the main cases of data transmission between a source system and a destination storage are listed, including the different visibilities of local identifiers at source and destination. Persistent identifiers are introduced to solve two problematic cases that might get relevant in case of update of demographics. Then a look at the number space of identifiers and the existence of demographics in a certain identifier domain is taken. In a setup of a National Pseudonymisation Service, properties and permissions of systems and domains have to be defined. Then finally, the main identity related services of the National Pseudonymisation Service will be outlined. Since the National Pseudonymisation Service use an existing Master Patient Index for matching decisions, aspects of this relation will be discussed. The section ends with discussions about the creation of new local identifiers, especially the calculation of local identifier with small number of bits.

In *Section III – Results*, the use of the National Pseudonymisation Service and the use of an in-house pseudonymisation solution will be shown on existing use cases that have been implemented already or which are in planning.

The paper ends with *Section IV – Conclusion and Future Work*, in which the positive effects of the proposed solutions for researchers will be outlined.

II. METHODS

The generalized concept of a National Pseudonymisation Service (NPS) and of an in-house pseudonymisation solution bases on use cases that have been identifier by questioning various researchers in the field of clinical and population

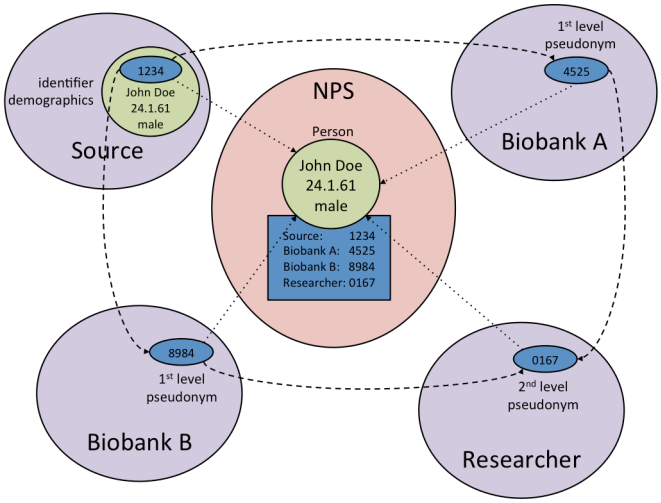

Figure 1. Identities and Domains

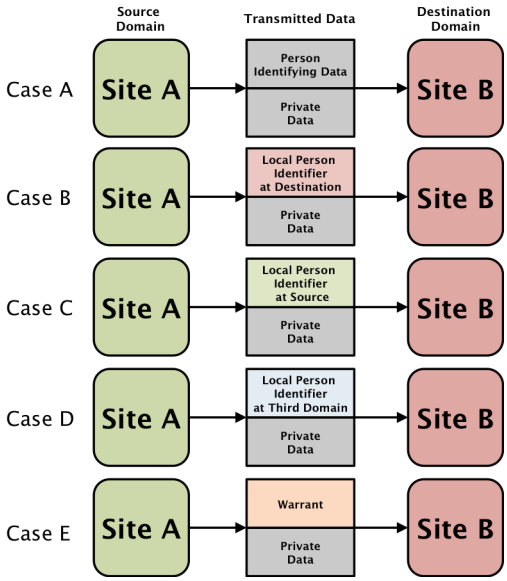

Figure 2. Different cases of transmitting data

based studies. First, some terms must be clarified; later, these cases will be discussed.

A. *Identifier of persons and identifier domains*

In the digital world, the use of identifier of persons is quite common. It simplifies the linkage of data of the same persons, if unique identifiers are used. This linkage is quite complicated if only demographics (e.g., name, address, birthday) are given.

1) *Local identifier and identifier domain*

The concept of (local) identifiers for persons that are only valid in a certain local context is one of the basic concepts of the IHE Patient Identifier Cross Referencing (PIX) Integration Profile [16][17], as it is implemented inside hospitals or laboratories. Usually different systems (e.g., storage systems, imaging systems) use different identifiers inside the same institution to identify the same person. The Patient Identifier Cross-reference Manager enables the systems to communicate with each other, even if they use different identifier for the same person. This is solved by so called identifier domains for the different systems. Usually, the same person should only have one identifier inside an identifier domain. This concept cannot only be used for the exchange of data inside an institution but also between different institutions (different domains), for which a person has different patient identifiers (local identifier).

The local identifier of a person in one domain is different from the local identifier of the same person in another domain. Without help of the Patient Identifier Cross-reference Manager it is difficult to translate the link of persons between the two domains.

The concept of local identifier and identifier domains is used in the concept of the National Pseudonymisation Service. Identifier domains not only describe institutions but also might identify applications or application contexts, e.g., national laboratory-application, clinical study about cancer. The identifier domain usually is identified by a unique OID (Object Identifier) [18].

2) *Pseudonym*

In the proposed concept, a pseudonym is seen as a local identifier inside an identifier domain where no demographics are available.

Pseudonyms from different domains must be different. Having a local identifier from one domain must not allow calculating the pseudonym from another domain, except the domain is responsible for the creation of the pseudonym. This statement ensures that it is not possible to break the pseudonymisation on known identifiers.

3) *2nd-level pseudonym*

As for pseudonyms, a 2nd-level pseudonym is also only a local identifier in a certain identifier domain where no demographics are available. In this case, the source of data is a domain that identifies persons by pseudonyms and not by demographics.

2nd-level pseudonyms in an identifier domain can be linked to the same person, even if the 1st-level pseudonym was from different domains.

Example (Figure 1): Medical data of a person are sent to Biobank A that works with 1st-level pseudonyms. Medical data of the same person are sent to Biobank B that works with different 1st-level pseudonyms. Data of both biobanks are sent to a researcher who works with 2nd-level pseudonyms. The researcher is able to link data of both biobanks to the same person, in case of the same 2nd-level pseudonym.

It is clear that such a scenario in reality requires approval by ethics commissions or data protection authorities.

B. *Main cases of data transmission*

After being familiar with the terms *local identifier* and *identifier domains*, it is possible to describe the main cases of transmitting data between a source and a destination. The described use cases cover cases that include the use of a

National Pseudonymisation Service and the use of an in-house pseudonymisation, with a stronger focus on the design of the National Pseudonymisation Service.

In the proposed setup the communication between source and destination systems is direct, so no system is involved during the transmission of medical data between source and destination that modifies the transmitted data. This is not only true for the in-house pseudonymisation but also in case of the use of the National Pseudonymisation Service. The National Pseudonymisation Service is defined solely as a passive service that is used to identify persons and request or to manage local identifies. It will not allow the bypass of medical data from source to destination. Also, it will not perform a pseudonymisation of medical data on the fly (i.e., replace demographics in the medical data by pseudonyms).

The question that results from these restrictions is: What information is send from source to destination (apart from the medical data) that allows the mapping of the medical data to a certain person at the destination?

There are several options to answer this question. The five possible cases that describe these options are shown in Figure 2:

A. Demographics of the person are exchanged.
B. The local identifier from destination domain is exchanged.
C. The local identifier from the source domain is exchanged.
D. The local identifier form a third domain is exchanged.
E. A warrant is exchanged that can be used by the destination to request the local identifier from its domain by showing the warrant.

All cases do not make an assumption on how the data and information is transmitted between source and destination. It does not have to be electronically only. Alternatively, this data could be send by the use of physical objects (e.g., as barcode on paper or box).

1) Case A: Demographics of the person are exchanged

In this case private data of a person is exchanged between source and destination together with the demographics of the person. So the destination is forced to link data from the same identity on base of the given demographics. This could be done by the use of a local Mater Patient Index (MPI) or by the use of a National Pseudonymisation Service. Anyway, it is clear that this local identifier is not a pseudonym, as the identity of the person is known.

2) Case B: The local identifier from the destination domain is exchanged

In this case private data of a person is exchanged between source and destination together with the local identifier of the person of the destination domain attached to it. So the sources need to calculate, determinate or know the local identifier of the destination domain on base of its own local identifier or the known demographics of the given person. Alternatively, it needs to ask the National Pseudonymisation Service to provide this identifier of the destination.

As a consequence, all source systems from all source domains will know the local identifiers or pseudonyms from the destination domain but not vice versa. In case of the use of a National Pseudonymisation Service, the sources systems needs permissions to request the local identifier of the destination domain on base of its own local identifier or demographics.

3) Case C: The local identifier from source domain is exchanged

In this case private data of a person is exchanged between source and destination together with the local identifier of the person at the source domain attached to it.

As a consequence the destination system of the destination domain will know the local identifiers from the source domain but not vice versa, so the local identifier or pseudonym that is used at the destination is hidden to all sources.

In case of in-house pseudonymisation, this case only makes sense in case of one source only, otherwise it will be impossible to link identifier from different sources to the same person. This limitation does not exist in a setup with the use of a National Pseudonymisation Service, for which the destination needs permission to translate the local identifiers of the sources to its local domain identifier.

4) Case D: The local identifier from a third domain is exchanged

This case introduces a third identifier domain. This case makes sense if such a third domain is created especially for the exchange between source and destination and nowhere else. In such a setup local identifiers from a source will not be disclosed at the destination and vice versa. Source and destination systems must only use the identifier of the third domain during the exchange of the private data and not for the storage of the private data.

This case allows different variations by using in-house pseudonymisation or the National Pseudonymisation Service during the transition of the identifier between source to the third domain, and between third to the destination domain. As for Case C, an in-house pseudonymisation between sources and third domain is only useful in case of only one source domain, because it is impossible to define a calculation or determination process that would allow the transition of local identifiers of the same person from different sources that result in the same identifier in the third domain.

The translation between the identifier of the third domain and destination domain can be performed in-house or at the National Pseudonymisation Service.

5) Case E: A warrant is exchanged

In this use case private data of a person is exchanged between source and destination together with a warrant attached to it. This use case requires the use of the National Pseudonymisation Service and does not work with in-house pseudonymisation.

The warrant is created and/or managed by the National Pseudonymisation Service on base on information, provided by the source (e.g., local identifier or demographics). The

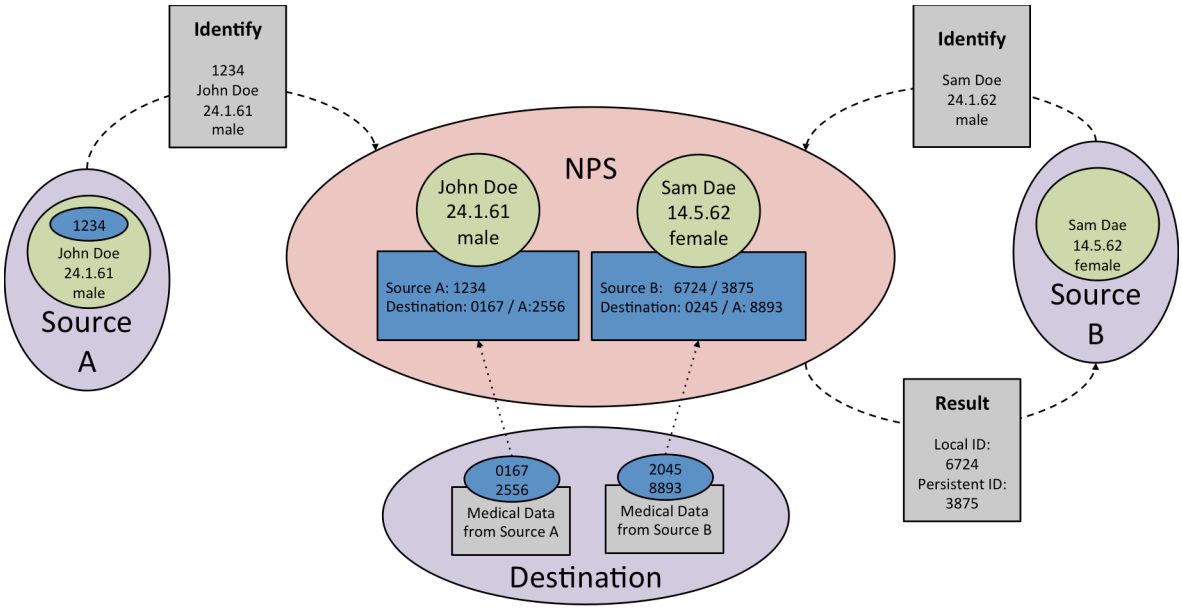

Figure 3. Persistent identifier and initial identification

destination will then be able to retrieve the identifier belonging to the destination domain on base of the warrant.

In this case, the source does not know the local identifiers or pseudonyms in the destination domain and the destination does not know the demographics at the source domain.

In contrast to Case D and the use of identifiers from a third domain, the warrant can be managed by the source and might be defined with at time-to-live. The warrant should not be used as a replacement of a local identifier because they are not unique in case of the same person. Additionally, the National Pseudonymisation Service might delete the warrant out of its systems after use.

The warrant-based approach might be used in cases of re-identification of patients. In that case, a warrant is requested by the destination on base of the pseudonym and the source is able to re-identify the patient on base of the warrant.

C. Identifier management

Usually, hospital information systems or equivalent systems manage local identifier for patients themselves. In this case, the local identifier is created inside the identifier domain of the data sources. The identifier domain guarantees that the person behind the local identifier never changes, even if the demographics of that person change significant. This is an important requirement. In the future, it might be possible that two identifiers are merged because they have been identified as doublets of the same person. But an identifier never changes the link to the individual person.

Not all identifier domains manage identifiers themselves. As an example, collection sites of a clinical study might be located at hospitals, but have no access to the hospital information systems and therefore not to the local identifier of that hospital. In that case a new local identifier has to be created for the collection site domain. The National Pseudonymisation Service can overtake this task on the base of given demographics.

In the National Pseudonymisation Service, it must be configured for each data source, if it creates and manages identifiers themself or if the National Pseudonymisation Service has to take responsibility for this.

D. Persistent local identifier

The National Pseudonymisation Service decides with given demographics, if the demographics match with the demographics of a known person or not. If demographics of a person are updated at a source, this might lead to a different matching decision at the National Pseudonymisation Service, so the demographics are linked to a different person.

Sources who manage local identifiers in their domain are not affected by this decision because the local identifier of the person at the source will not change. For sources and destinations with local identifiers management by the National Pseudonymisation Service, things are different: some data sets with an associated local identifier might need to be changed in a way that reflects the new matching decision, i.e., local identifiers of some datasets need to be updated too. The identification of these datasets on base of the current local identifier is not sufficient, because for some datasets from different sources, the change must not be performed.

To solve this issue, an additional persistent local identifier is introduced. The persistent local identifier will never change, regardless of updates of demographics. It can be used to provide update information for exactly those entities that are affected by the update decision.

The persistent identifier is an addition to the local identifier inside an identifier domain and is linked to the demographics that were used during the first identification step of the demographics from a source at the National Pseudonymisation Service.

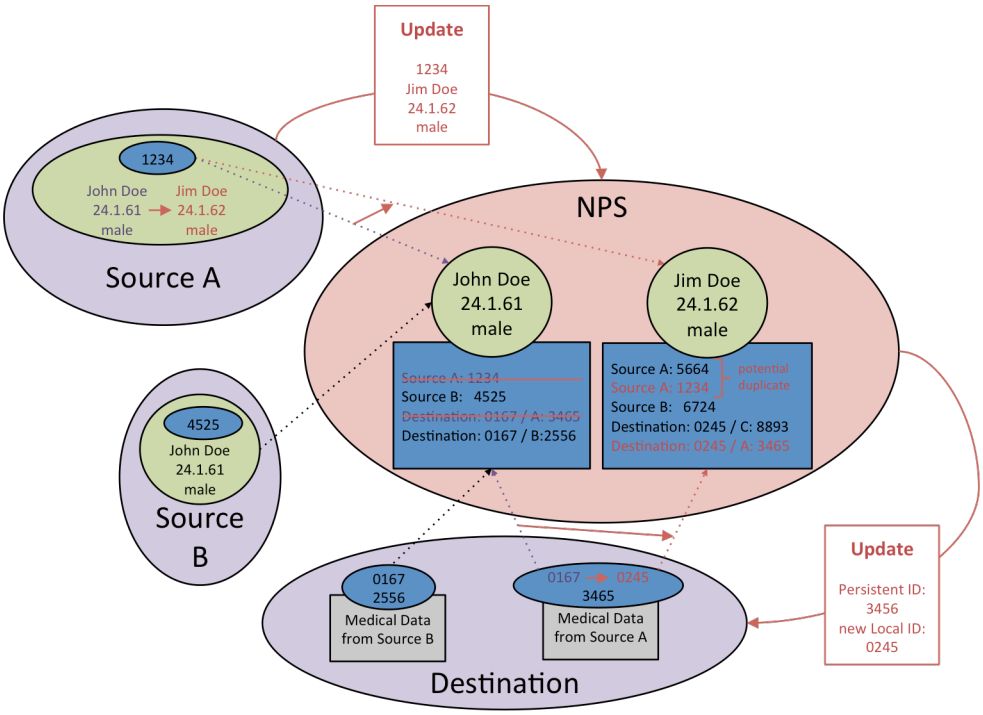

Figure 4. Problematic case: Update affects destination

At one point in time, a data source needs to identify demographics of a person at the National Pseudonymisation Service, either to make it aware of the local identifier in the source domain or to request a local identifier on base of the demographics. The persistent identifier is bound to that identification process.

In the example of Figure 3, Source A is identifying John Doe together with its local identifier 1234. Medical data that is sent from Source A to the destination is linked to that person at the destination via the local identifier 0167 and the persistent identifier 2556. In the same example, Source B identifies demographics of Sam Dae without a locally managed local identifier. This source will receive the local identifier 6724 from the National Pseudonymisation Service plus a persistent identifier 3865. If Source B identifies a person with the same demographics in future, it will receive the same local identifier 6724 but always with a different persistent identifier.

E. Problematic cases

The persistent identifier can be used to solve two problematic cases:

- Update affects destination
- Update affects source

1) Update affects destination

Two sources from different domains (Example Figure 4: Source A, B) provide demographics that lead to the same local identifier/pseudonym at the destination (0167). Then one of the sources (Source A) updates the demographics and the National Pseudonymisation Service decides that the previous matching decision was wrong and that this new demographics belongs to a different person. So the local identifier (1234 of Source A) is re-linked in the National Pseudonymisation Service to a different or new person. On base of the persistent identifier (3465), the destination can be informed to update the local identifier (0245). This affects only the medical data that has its origin in Source A.

One could argue that a persistent identifier could be avoided, if the destination would store information about the source domain together with the local identifier. In the example an update then would be: Update data from Source A with local identifier 0167 to the new local identifier 0245.

This argument is true, but there are good data protection arguments to hide the origin of the data at the destination. The persistent identifier in that case acts as a pseudonymisation of the source.

2) Update affects source

The National Pseudonymisation Service manages the local identifiers of a source domain (Example Figure 5: Source A), so the National Pseudonymisation Service provides the local identifiers plus a persistent identifier after identification of demographics.

During two independent events, demographics are identified at the National Pseudonymisation Service that lead to the same local identifier at the source (1234) but with different additional persistent identifiers (2347, 5678). Then later, one set of demographics (identified by local identifier plus persistent identifier: 1234 / 2347) is updated and the National Pseudonymisation Service decides that the demographics belong to a different person as previously suggested (Figure 6). So the data that was provided in one event at Source A has the wrong local identifier and needs to be changed to the new local identifier (5667). This change does not affect the local identifier from the second event.

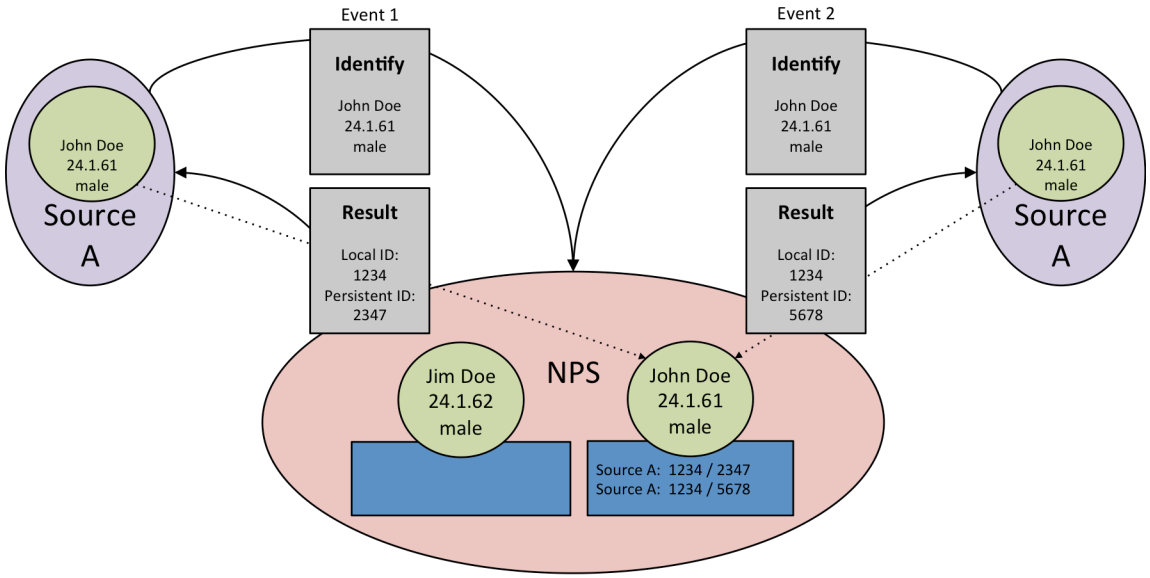

Figure 5. Problematic case: Update affects source (initial state)

One might argue that such a use case is not likely, especially in case of information systems inside the source domain. In case of clinical studies, sometimes the management of identities depends on papers, Excel sheets or other unreliable tools. So it is not an unrealistic scenario that nurses collect samples at different collection events and use the National Pseudonymisation Service to retrieve a local identifier on base of re-typed erroneous demographics, which later needs to be updated.

F. Avoidance of persistent local identifiers

In the case of local identifiers of a source domain that is managed by the National Pseudonymisation Service, the use of persistent local identifiers is one way to manage updates. An alternative approach can avoid the use of persistent local identifiers. It foresees that each matching request at the National Pseudonymisation Service that is performed without the use of a local identifier will lead to a new local identifier, regardless whether the demographics match a known person or not.

As for local identifiers that are managed by the sources, this identifier will never change even after update of demographics. The National Pseudonymisation Service can be asked, for which local identifiers it assumes that they belong to the same identity. This list might change after demographics are updated for a given local identifier.

So there are two options to treat potential update problems at domains that do not create or manage local identifiers: Either a persistent identifier is provided together with the local identifier, or always new local identifier are created even if the National Pseudonymisation Service assumes that the demographics belong to a known person.

G. Identifier domain and identifier number space

In case of local identifier created and managed by the National Pseudonymisation Service, it is suggested, that the number is a purely (pseudo-)random integer number from the range zero to a maximal number. It does not include any information that is linked to the demographics of the person. The maximum has to be defined per identifier domain at the National Pseudonymisation Service.

It is up to the users of the local identifier if they encode the number into a character representation or if they add error correction or error detection codes, e.g., to make it human readable. During communication with the National Pseudonymisation Service, only the integer representation must be used.

H. Identifier domain and availability of demographics

One can distinguish between domains where demographics are available and domains where demographics are not available.

Usually domains with no demographics are these where the identifier is seen as the pseudonym. But this is not always the case. There are cases where an identifier is linked to a person and at the same time demographics of that person are not available. An example for such a case is the domain of health professionals. In that case, the eHealth ID of a health professional is not a pseudonym, but access to the demographics of the health professional is not necessary available at the source.

This case is important, as a pseudonymisation of identifiers from a domain without demographics is generally possible (e.g., to pseudonymise the eHealth ID of health professionals). Since such identifiers are registered at the National Pseudonymisation Service without any demographics, a link to an existing person is never possible.

To stay in the example: Pseudonyms of health professionals are never linkable to pseudonyms of patients, even if the health professional and the patient are the same person.

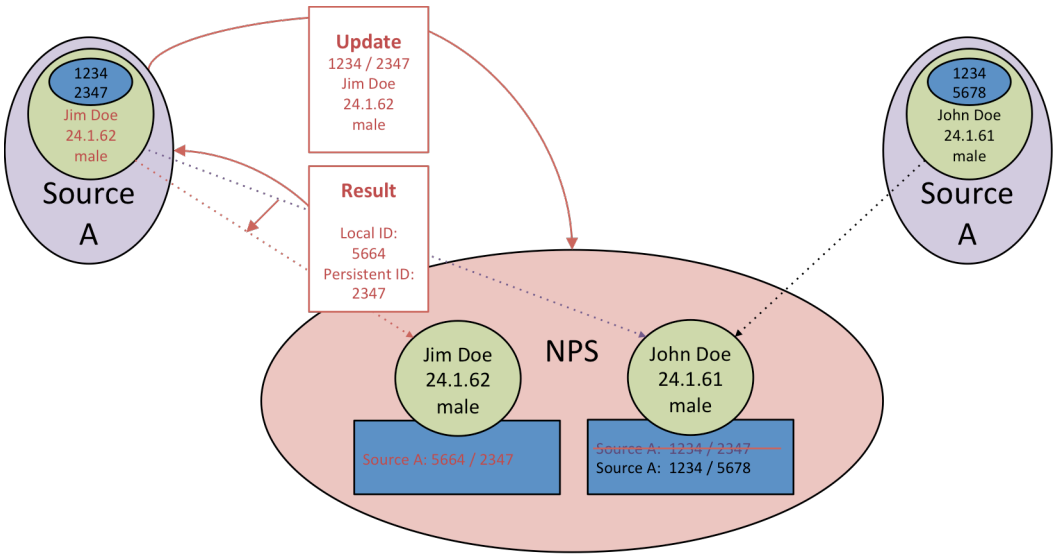

Figure 6. Problematic case: Update affects source (after update)

I. Identity Linking

If a source figures out that two local identifiers belong to the same person (even if the demographics are different), the source can perform a linkage-request to tell the National Pseudonymisation Service that one local identifier will never be used anymore and that all data that is linked to the obsolete identifier will belong to the surviving local identifier. Properties and permissions

Figure 7 gives an overview about the relationship of properties and rights concerning systems and identifier: A system, e.g., server, client application or user, belongs to one or more identifier domains. For a specific domain it is defined whether a system has certain permissions or not:

Provide demographics:

Not all systems inside a domain should be allowed to provide demographics at the National Pseudonymisation Service. Some systems are only allowed to use the local identifier inside that domain.

Update demographics:

In case of first contact, some systems must be allowed to provide demographics to the National Pseudonymisation Service. Update of demographics is a critical task that only should be permitted to some selected systems.

Link identifier:

Similar to update of demographics, the linking of identifier is a rare case that only should be done after the identification of doublets in the local system is beyond question.

Retrieve demographics:

This is the most critical task in the whole concept of the National Pseudonymisation Service. Retrieval of demographics on base of a given local identifier should only be possible in rare cases, e.g., re-identification of persons in case of important notifications.

For reasons of data protection, the National Pseudonymisation Service will only provide the latest version of demographics that has been provided by a system in that domain. Demographic details from other domains will not be accessible. Also this permission will only provide data, if the domain itself manages demographics. Since domains that only have access to pseudonyms never provide demographics to the National Pseudonymisation Service, the retrieval of demographics in that domain is excluded.

For a specific domain, properties define whether it is a source domain with demographics or a destination domain with pseudonyms:

Demographics available:

In domains with demographics available, a source domain is given. Usually, in domains without demographics, this is not the case (except in a relation 1st level pseudonym, 2nd level pseudonym).

Identifier managed by source:

For source domains, it has to be defined, whether a local identifier is managed inside the domain, or if it has to be provided by the National Pseudonymisation Service. In the second case, it must be defined, whether a persistent identifier is used to manage update conflicts or if always a new local identifier will be used in that case.

For destination domains without demographics, the National Pseudonymisation Service will always manage the local identifier. It is not possible that the domain itself manages pseudonyms.

Number range of local identifier:

In *Section G.* Identifier domain and identifier number space it is explained, why the National Pseudonymisation Service only manages numbers as local identifiers. This property defines the range of the number space.

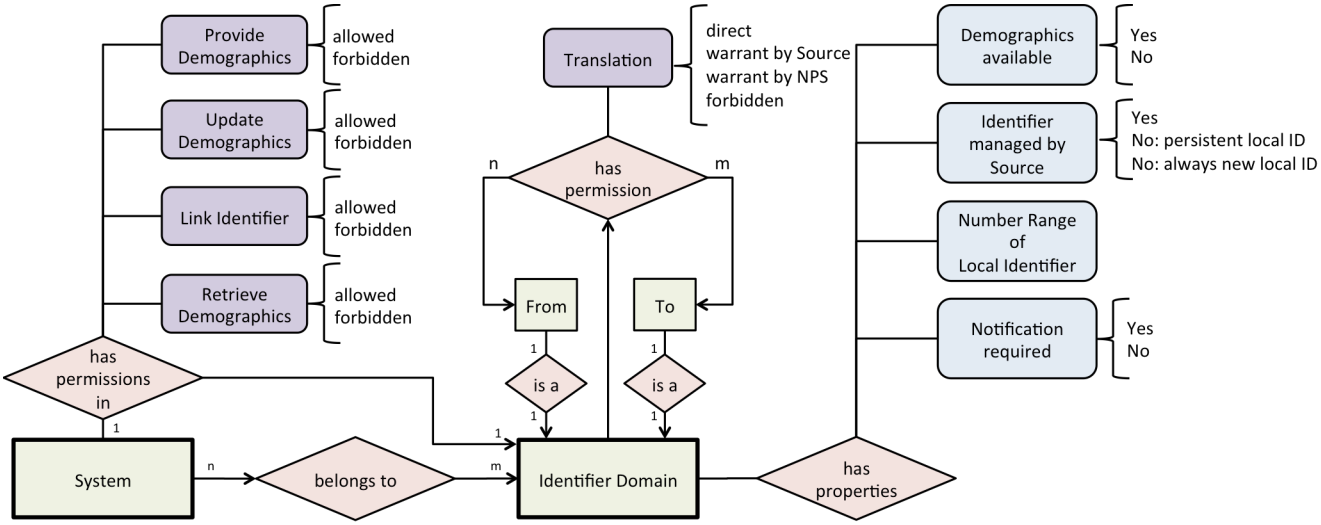

Figure 7. Properties and permissions

Notification required:

Systems might fail or crash in the wrong moment. Some tasks might require notification to ensure that the involved systems have stored the result of a request in their databases. If a system notifies a certain result, the responsibility for the use of the information is moved from the National Pseudonymisation Service to the notifying system.

In the use cases that are described in *Section B.* Main cases of data transmission, all cases have a from-to relation in regards to the translation of local identifiers or the creation of warrants. These relations need to be defined as permissions for direct or warrant-based translations in the National Pseudonymisation Service.

Case B:

A source system from a source domain has permission to translate its local identifier (From-Domain) directly to the destination domain (To-Domain).

Case C:

A system from a destination domain has permission to translate the local identifier from a source domain (From-Domain) directly to its local identifier (To-Domain).

Case D:

A source system from a source domain has permission to translate its local identifier (From-Domain) directly to a third domain (To-Domain), and a system from a destination domain has permission to translate the local identifier from a third domain (From-Domain) directly to its local identifier (To-Domain).

Case E:

A direct translation of local identifiers is not permitted, so a translation requires the use of a warrant. A system of the source domain (From-Domain) has permission to create a warrant (Warrant by Source) or retrieve a warrant (Warrant by NPS) for the destination domain (To-Domain).

The data model of Figure 7 allows the definition of permissions that are not useful: In the direct translations the system must either belong to the from-domain or to the to-domain. In the warrant-based translation, the permission, the system must belong to the from-domain.

J. Identity related services

As a result from the previous sections, the following services are required at the National Pseudonymisation Services. Services that are needed in the in-house setup are explicitly named. For simplification reasons, persistent local identifiers are mentioned in most of the description, but it depends on the definition of the domain, whether a persistent identifier has to be used or if it will be returned or not.

Services for the notification of the reception of identifiers and warrants are not listed.

All functions require the "Identifier Domain" parameter. This parameter is needed to identify the current domain of the calling system, since systems might belong to several domains.

1) Register a person by Demographics

```
Register Person
     Identifier Domain
     Demographics
→    Local/Persistent Identifier
```

Returns a local identifier on base of demographics.

2) Register a person by demographics and local identifier

```
Register Identified Person
     Identifier Domain
     Local Identifier
     Demographics
```

In domains that manage the local identifiers by themselves, the service makes the National Pseudonymisation Service aware of the local identifier and its associated demographics in that identifier domain. No persistent identifiers are provided in self-managed domains. This function does not return any result.

3) Update of demographics of a person

```
Update Person
     Identifier Domain
     Local/Persistent Identifier
     Demographics
→    Local Identifier
```

If a local identifier has already been registered at the National Pseudonymisation Service, or the National Pseudonymisation Service has returned a local/persistent identifier, this function is used to update the demographics. This might lead to an update of the local identifier (see *E.2)*.

4) Translate identifer at the source domain

```
Translate Identifier
     Local Identifier Domain
     Foreign Identifier Domain
     Local/Persistent Identifier
→    Foreign/Persistent Identifier
```

This is a simple translation of identifiers between the local (source) and the foreign (destination) domain.

This is mainly the function that is needed in the in-house setup, so the local identifier of the foreign domain is calculated or determined on base of the local identifier only.

5) Trananslate identifier at the destination domain

```
Retrieve Identifier
     Local Identifier Domain
     Foreign Identifier Domain
     Foreign/Persistent Identifier
→    Local/Persistent Identifier
```

This service is similar to 4) but in this case, the destination domain is calling the service. This leads to a change of the focus of the local-foreign relation: the destination is requesting its local identifier on base of the foreign identifier of the source.

6) Register a warrant, associated to a local identifier

```
Register Warrant
     Local Identifier Domain
     Foreign Identifier Domain
     Local/Persistent Identifier
     Warrant
```

This function registers a warrant for a foreign domain with a given local identifier. In this case the warrant is managed (provided) by the source

The warrant is only valid in the foreign domain to retrieve the local identifier of that domain.

7) Request a warrant, associated to a local identifier

```
Request Warrant
     Local Identifier Domain
     Foreign Identifier Domain
     Local/Persistent Identifier
→    Warrant
```

This function requests a warrant for a foreign domain with a given local identifier. In this, case the warrant is managed (provided) by the National Pseudonymisation Service.

The warrant is only valid in the foreign domain to retrieve the local identifier of that domain.

8) Retrieval of the local identifier at the foreign domain on base of a warrant

```
Redeem Warrant
     Local Identifier Domain
     Warrant
→    Local/Persistent Identifier
```

Having a warrant of the correct domain, this service will allow the retrieval of the identifier in that domain.

9) Re-identification of demographics on base of a local identifier

```
Re-Identify Person
     Local Identifier Domain
     Local/Persistent Identifier
→    Demographics
```

In case of re-identification requests, this service will only provide the latest version of demographics that have been registered in that domain. Demographics from different domains related to the same persons are not accessible.

This service might also be useful in the in-house setup in case of re-identification requests.

10) Linking of local identifiers, in case of identified doublets

```
Link Local Identifier
     Local Identifier Domain
     Obsolete Local Identifier
     Surviving Local Identifier
```

If a source that manages the local identifiers itself identifies doublets, should use this function to inform the National Pseudonymisation Service about the merge in the local (in-house) system.

11) Get updates of identifiers in the domain

```
Get Updates
     Local Identifier Domain
→    List of
     [Persistent Identifier: New Local Identifier]
```

In case of updates at the National Pseudonymisation Service, local identifiers might change for some data (see *E.* Problematic cases). The National Pseudonymisation Service is a passive service so it only responses to requests. Identifier domains must use this service regularly to get notified about the latest updates, since the last request.

12) Identification of potential duplicates

```
Vigilance Request
     Local Identifier Domain
     Local/Persistent Identifier 1
     Local/Persistent Identifier 2
```

On base of the medical data, a destination domain might come to the conclusion that the given local identifiers are potential duplicates and belong to the same person. An alert will be triggered at the identity vigilance of the National Pseudonymisation Service to check the case.

13) Identification of potential splits

```
Vigilance Request
     Local Identifier Domain
     Local Identifier
     Persistent Identifier 1
     Persistent Identifier 2
```

On base of the medical data, a destination domain might come to the conclusion that the given local and persistent identifiers are potential splits and should belong to different persons. An alert will be triggered at the identity vigilance of the National Pseudonymisation Service to check the case.

K. Matching of identities

The National Pseudonymisation Service uses an underlying Master Patient Index to figure out, whether the given demographics of a person are known (match), or if they identify an unknown person (no-match). The matching algorithm depends on mandatory demographics (first name, last name, gender, and birthday) and optional demographics (national social security number, zip-code of the birthplace).

Depending on the degree of agreement, the algorithm will distinguish, true matches (the person is known with high probability), true non-matches (the person is not known with high probability), and ambiguous matches (there is more than one potential candidate or it is not clear whether the person is known or not).

If the decision is not clear (ambiguous match), a new person will be created in the system, and the identity vigilance will be informed to solve the problem by requesting additional information from the involved domains.

Since the National Pseudonymisation Service acts as a shell around an existing Master Patient Index, the Master Patient Index service could be replaceable at any time in case without affecting the pseudonymisation service.

L. Calculation of local identifiers

An important part of the entire process of identification of persons is the creation of the local identifiers of a domain. This calculation has to be done at the National Pseudonymisation Service or locally at the in-house solution for new persons or for persons that are accessed for the first time by an identifier domain. Domains that provide their own local identifier are not affected by this question.

Each person that is managed by the National Pseudonymisation Service (or internally by its Master Patient Index) is represented by a person-object. This object consist of the single best record of the demographics of the persons plus an internal identifier of the object. Each local identifier of an identifier domain is linked to that object via the internal object identifier. The link will be established during the registration step of the person or the translation of identifiers between different domains. If an identifier does not exist at that time, it must be created.

In the in-house solution, usually the managed persons are stored inside a database with a person identifier associated to it. This might be an attribute of the database table or it is a given identifier that was inscribed together with the demographics (e.g., social security number) or it was already a pseudonym that was given with the data. If personal data needs to be delivered to a certain domain, the domain specific identifier needs to be created, if this has not been done already.

In both cases there are several options to create the identifier out of the person identifier (object identifier, person identifier, pseudonym, social security number etc.):

Take the next free available number: last used number plus 1:

In this case all created numbers build a continuous running number. This must be avoided, if the identifier is used as a pseudonym. If the original identifiers are already continuous numbers, a link could be established between identifier and time of creation of the person inside the system.

Chose a random number:

The use of random numbers should be the preferred choice, but require the management of mapping tables (local identifier → person identifier).

Such mapping tables could be used in case of selective anonymisation of individuals: If the entry (local identifier → person identifier) is replaced with (local identifier → NULL) Every data that is stored with the local identifier can never be linked to the person again.

Calculate the identifier from the person identifier:

If the management of mapping tables must be avoided (especially in the in-house setup) and a selective anonymisation is not required, the calculation of the local identifier on base of the person identifier together with a certain secret is a good alternative to the random number.

Good strategies are the use of salted hashes (`Hash(Salt + person identifier)`) or encryption (`Enc(Key, person identifier)`). In both cases, the salt or the key is the secret that is linked to the identifier domain.

This strategy is problematic, if the calculated local identifier has limitations related to the data type. Example: The person identifier at the source is of data type *4 byte unsigned integer* (=32 bit), and the resulting local identifier must be from the same data type.

Current hashing or encryption algorithms usually work with 128 bit minimum, so are not suitable in the described case. Cropping of the result to 32 bit is not a way to go because this introduces a risk of collisions, which means that for some person identifier the calculated local identifier will be the same. This behavior cannot be tolerated. For this special case, a new calculation algorithm is proposed.

M. Calculation of local identifier with small number of bits

The mathematics behind the local identifier calculation of a person identifier is based on selected primitive roots of fixed prime numbers as it is used in the Diffie-Hellman protocol to ensure a secure key exchange [15]. First we need to introduce some fundamental mathematics.

1) Discrete logarithm

Having the equation:

$$b = a^i \bmod p, \text{ with } p \text{ prime}, i \in \{1..p\text{-}1\} \qquad (1)$$

Then i is called the discrete logarithm. This is equivalent to

$$i = \log_a b \bmod p, i \in \{1..p\text{-}1\} \qquad (2)$$

The calculation of b is easy but currently there exists no efficient way to find the discrete logarithm i with given a, b and p.

This statement is only true if p is big enough to make the use of pre-calculated solution tables impossible and if no pre-knowledge about i exists that allows reducing the search space.

2) Primitive roots

The property of a being a primitive root of prime p means that

$$a^i \bmod p, \text{ with } i = 1..p\text{-}1 \quad (3)$$

results in all values of $1..p$-1, with no value double or missing. This property is relevant to create collision free local identifiers.

Primitive roots have been used already a long time ago to build good random number generators [19]. The proposed algorithm uses this knowledge to introduce pseudo-randomness into the series of pseudonyms.

3) Adaption for the calculation of the local identifier

With k bits that are reserved for the local identifier, a prime number p should be chosen that in best case is the highest prime number lower than 2^k. With the given p, the interval of possible person and local identifiers is $1..p$-1. The numbers that are invalid in the k-bit number space are 0 and $p..2^k$-1. As an example: For k=31, the highest prime lower than 2^{31} is 2^{31}-1. In this case, only 0 and 2^{31}-1 cannot be used as person and local identifier.

The difficulty to find the discrete logarithm i of the equation $a^i \bmod p$ is based on the assumption that i is randomly distributed and that no information can be used to reduce the number of possible values. This may not be the case if the persons person identifier is used as exponent i.

Two examples might help to demonstrate the problem. In both cases, i equals the person identifier id.

In the first example the exponent i is a continuous number starting with 1, so the n^{th} local identifier belongs to the person identifier n. If an attacker is able to estimate the number of already managed persons, the number of potential i is heavily reduced.

In the second case, the person identifier is created out of the birthday and a running number (e.g., 1985032312 for the 12th person born in March 23 of 1985). In the example, knowing that a person was born at a certain day, this limits the number of potential i to 100.

To avoid the reduction of potential i with prior knowledge about the person identifier id, two processing-steps are performed, including one non-linear step:

1. XOR (non-linear exclusive or):
   The person identifier will be XORed with a constant $c \neq 0$ of k bits
2. EXPAND:
   The intermediate result is multiplied with an expansion factor q mod p, $(1<q<p)$

Step 1 might lead to an invalid results that is out of the range of the allowed values (0, $p..2^k$ -1). If this happens the XOR must be reversed. In case of p be close to 2^k, the number of invalid values (p..2^k-1) can be minimized, which lowers the risk to revers the XOR step.

p being prime guarantees that the result of step 2 is still in the range of $1..p$-1, avoiding any doubles.

At that point, even with pre-knowledge about the person identifier, no conclusions about the exponent i of the calculation $a^i \bmod p$ can be made, which would allow to reduce the search space. Finally, the main calculation step $a^i \bmod p$ can be performed.

Unfortunately, if the prime number p is small, it is possible to calculate all possible $b=a^i \bmod p$ to set up a solution table $b \mapsto i$. For a prime smaller than 2^{31}, maximal 8GiB are needed to setup such a table (1GiB = 2^{30} Byte). Even for prime smaller than 2^{40}, a solution table with maximal 5TiB needs to be pre-calculated (1TiB = 2^{40} Byte). Tables with that size fit in currently used RAM or hard disks and are no burden for potential attackers. A solution to overcome this problem is to also keep the primitive root a secret. In that case, with given b and p, for each a a different i exists that fulfills the equation.

The entropy of the secrets a, q and c that have been used so far might be insufficient to avoid brute force attacks. So a final round of confusion is performed:

3. XOR (non-linear exclusive or):
   The intermediate result will be XORed with a constant $d \neq 0$ of k bits
4. ROL (shift rotate left):
   The intermediate result will be shift-rotated s bits left ($|s|>0$)

As with step 1, step 3 must be reversed, if the result is invalid. If the intermediate result of step 4 leads to an invalid value, it must be repeated until the intermediate result is in the allowed range. Both strategies do never introduce duplicates.

The calculated local identifier finally is the outcome of step 4. Figure 8 lists the entire algorithm as pseudo code.

The complexity of an attacker to re-identify the person ID is based on the secrets a, c, d, q and s and requires knowledge about some person and local identifier pairs to proof if the secrets are correctly identified.

4) Example

All calculation steps of the local identifier for the person identifier id= 300568 are shown in Figure 9.

- Let k=31 and prime $p=2^{31}$-1=2147483647.
- The initial value of id will be XORed with c=1656294509.
- The expansion factor is defined as q=41795.
- a=572574047 is a primitive root from p.
- The intermediate result will be XORed with d=913413943.
- Finally, an intermediate result will be shift-rotated left with s=11 bits.
- The pseudonym that has been calculated from this identifier is 353489627.

5) Finding a primitive root

For a given prime number p it is unnecessary to find all primitive roots to select the secret a; only one primitive root

```
FUNCTION calculateLocalIdentifer (id, k, a, p, c, d, q, s)
BEGIN
   t1 := id XOR c                  // XOR person identifier
                                   // with secret c
   IF (t1 ∉ {1 .. p-1}) THEN       // if out of range
      t2 := id                     // reverse if necessary
   END IF
   t2 := (t1 * q) mod p            // expand with secret p
   i := t2                         // this is the exponent

   b := a^i mod p                  // the main calculation

   t3 := b XOR d                   // XOR with secret d
   IF (t3 ∉ {1 .. p-1}) THEN       // if out of range
      t3 := b                      // reverse if necessary
   END IF
   t4 := t3 ROL s                  // shift-rotate-left s bits
   WHILE (t4 ∉ {1 .. p-1}) DO      // if out of range
      t4 := t4 ROL s               // repeat if necessary
   END WHILE
   lid := t4                       // the local identifier
   RETURN lid
END
```

Figure 8. Pseudocode of the algorithm

is needed. The density of primitive roots is quite high so it requires approximately four random tries in case of $p=2^{31}-1$ until a primitive root is found. To proof if a selected a is a primitive root, the series of $a^i \bmod p$ (i=1..p-1) has to be checked. If $a^i \bmod p = 1$ *with* $i \neq p-1$, the series can be stopped and a is not a primitive root. In that case two exponents are found resulting in the same value: $a^{i+1} \bmod p = a = a^1 \bmod p$.

The series can easily be calculated with

$$a^0 \bmod p = 1 \tag{4}$$

$$a^i \bmod p = a(a^{i-1} \bmod p) \bmod p \text{ for } i=1..p-1 \tag{5}$$

This is a quite time consuming process. A faster way to go is this:

First all prime factors of p-1 have to be identified. In case of $p=2^{31}-1$, the prime factors of $2^{31}-2 = 2147483646$ are 2, 3, 7, 11, 31, 151, and 331. The time to identify the prime factors has only to be spent once and does not affect the time to test the primitive root candidates.

For each prime factor f from p-1 the values $a^i \bmod p$ with $i=(p-1)/f$ need to be calculated. a is a primitive root of p if none of the results equals *1*. In the example the series of $a^{2147483646/2} \bmod p$, $a^{2147483646/3} \bmod p$, $a^{2147483646/7} \bmod p$, ..., $a^{2147483646/331} \bmod p$ needs to be calculated. These are maximal seven calculations.

6) Calculating $a^i \bmod p$

For the calculation of $a^i \bmod p$ in the described algorithm, the pre-calculation of $a^{i-1} \bmod p$ is not available; so, the recursion as mentioned in the equations (4) and (5) is not applicable. Alternatively, the calculation can be quickened if i is split into its binary representation of k bits:

$$i = (i_{k-1}, i_{k-2}, \dots, i_2, i_1, i_0) \text{ with } i_j \in \{0,1\} \tag{6}$$

```
t1   = id XOR c
     = 300568 XOR 1656294509 =
     = 1656593013

t2   = (t1 · q) mod p
     = (1656593013 · 41795) mod 2147483647
     = 284715408

b    = a^t2 mod p
     = 572574047^284715408 mod 2147483647
     = 465777933

t3   = b XOR d
     = 465777933 XOR 913413943
     = 766681658

t4   = t3 ROL s
     = 766681658 ROL 11
     = 353489627

lid = t4
     = 353489627
```

Figure 9. Example calculation

$$i = \sum_{j=0}^{k-1} 2^j \cdot i_j \quad with\ i_j \in \{0,1\} \tag{7}$$

Then

$$a^i \bmod p = \tag{8}$$

$$a^{\sum_{j=0}^{k-1} 2^j \cdot i_j} \bmod p = \tag{9}$$

$$\left(\prod_{j=0}^{k-1} a^{2^j \cdot i_j} \right) \bmod p \tag{10}$$

This calculation is very fast in case of pre-calculated $a^{2^j} \bmod p$ using

$$a^{2^0} \bmod p = a \tag{11}$$

$$a^{2^j} \bmod p = (a^{2^{j-1}} \bmod p)^2 \bmod p \quad \text{for } j=1..k-1. \tag{12}$$

As an example, let $i = 25 = 11001_2$. Then

$$a^{25} \bmod p = \tag{13}$$

$$\left(a^{2^4 \cdot 1} \cdot a^{2^3 \cdot 1} \cdot a^{2^2 \cdot 0} \cdot a^{2^1 \cdot 0} \cdot a^{2^0 \cdot 1}\right) \bmod p = \tag{14}$$

$$\left(a^{2^4} \cdot a^{2^3} \cdot a^0 \cdot a^0 \cdot a^{2^0}\right) \bmod p = \tag{15}$$

$$\left(a^{2^4} \cdot a^{2^3} \cdot 1 \cdot 1 \cdot a^{2^0}\right) \bmod p = \tag{16}$$

$$\left(a^{2^4} \bmod p \cdot a^{2^3} \bmod p \cdot a^{2^0} \bmod p\right) \bmod p \tag{17}$$

7) Bit-depth of the secrets

The algorithm for the calculation of the local identifiers is useless, if the used secrets allow a brute-force attack. This is not the case, if the entropy of the used secretes is big enough. Furthermore, the effort to calculate the pseudonym must allow the calculation of a high number of pseudonyms per time.

Several secrets to calculate the pseudonym are used:

TABLE I. FACTS

	4-byte signed integer	5-char base64 6-char base32	2-byte signed short integer
Bits	32	30	16
maximal positive value	2^{31}-1	2^{30}-1	2^{15}-1
highest possible prime	2^{31}-1	2^{30}-35	2^{15}-19
highest possible person identifer	2 147 483 646	1 073 741 789	32 748
number of invalid values	2	36	20
number of possible primitive roots of the prime	534 600 000	459 950 400	10 912

The number of possible primitive roots can be calculated with Eulers φ-function and is $\varphi(\varphi(p)) = \varphi(p-1)$.

- The random number *c* that was used to XOR the exponent.
- The factor *q* that was used to expand the exponent.
- The primitive root *a.*
- The random number *d* that was used to XOR the intermediate result.
- The number of ROLs (left-shift-rotate) of the intermediate result *s*.

As an example, the bit-depth of the secrets are calculated in case of data types that are usually used to store person identifiers

- 4-Byte signed integer:
  The number space is sufficient for a third of the entire living population on earth or four times the number of the living population of the European Union.
- 2-byte signed short integer:
  The number space is only useful for a small set of persons, e.g., for persons of a clinical study.
- 5 chars of base64-encoded numbers or 6 chars of base32-encoded numbers
  (in case of efficient human readability):
  The number space is sufficient for two times of the living population of the European Union but insufficient for the living population the People's Republic of China.

With the information of Table I, the entropy of the secrets can be calculated that are used during the calculation (Table II).

For integer and the encoded char-values, the secret with entropy of ≈124 bits is sufficient to avoid effective brute force attacks. This is void for short integer. Here the entropy of the secrets is only ≈64 bits. In that case, the calculation of the pseudonym must be performed in two rounds with different primitive roots, expansion factors, XORs and shift values. This does not fully double the entropy of the secrets because the final steps XOR and ROL are directly followed by another XOR step of the next round. All three steps can be simplified to only one XOR plus ROL. However, the entropy of the secret (≈111 bits) is sufficient today.

TABLE II. ENTROPY OF THE SECRET

Secret	4-byte signed integer	5-char base64 6-char base32	2-byte signed short integer
a: primitive roots	≈ 29 bit	≈ 29 bit	≈ 13 bit
q: expansion factor	≈ 31 bit	≈ 30 bit	≈ 16 bit
c: XOR exponent	31 bit	30 bit	15 bit
d: XOR result	31 bit	30 bit	15 bit
s: ROL result	≈ 5 bit	≈ 5 bit	≈ 4 bit
total	*≈127 bit*	*≈ 124 bit*	*≈ 63 bit*

8) Calculation speed

There are only a few steps involved in the calculation of the pseudonym. The calculation of $a^i \bmod p$ is identified as the most time consuming calculation. The calculation is straightforward and avoids several rounds until the final result is available. Multiplications are always more time consuming than XOR or shift operations so it is assumed that the pseudonym calculation is slower that the competitive approaches. In the known scenarios, the number of pseudonymisation calculations per time is sufficient: Tests have shown that on average hardware (Intel Core 2 Duo, 2.66 GHz) 132.5-thousand pseudonyms per second can be calculated.

9) Attacks

Important for the evaluation of the algorithm is the resistance against attacks and the possibility for re-identification.

It is known that for $b = a^i \bmod p$ (p prime, a primitive root of p) it is difficult to calculate the discrete logarithm i, if b, a, and p are known and p being big enough to avoid solution tables. In our case, also the primitive root a is unknown. On the other hand, there might be pre-knowledge about i. With the non-linear diffusion steps that base on the use of non-trivial secrets (e.g., $q \neq 1$, $c \neq 0$), the exponent is complex enough to make the information of the initial series useless.

Brute force attacks will only be possible if an attacker is able to validate the set of parameters with a given set of person identifiers and their associated local identifiers. An attacker will in worst case only get both sets, not knowing what person identifier and local identifier is finally linked. Depending on the size of the set, it is likely that several secret sets lead to the same transformation of the set of person IDs to the set of pseudonyms. In case of leaked pairs of person plus local identifier, this information can only be used to perform a brute force attack. A recalculation of the used parameters is not possible.

10) Re-Identification

A fast re-calculation of the person identifier is possible if all secrets are known. In case of small p and a given a, the solution table for $b=a^i \bmod p$ is made fast and every step of the entire calculation process can be reversed.

Only if the solution table cannot be pre-calculated, it is quicker to pseudonymise all known person identifiers again to find the correct local identifier.

III. RESULTS

A National Pseudonymisation Service on base of the described concept has been specified and is in the final phase of implementation in Luxembourg. The concept was developed after an intensive study of the demands has been carried out in Luxembourg. The National Pseudonymisation Service creates a shell around the National Master Patient Index that will be used in the National eHealth Platform of Luxembourg. This ensures, that the National Pseudonymisation Service will cover all persons working or living in Luxembourg and that all persons are managed with high quality demographics. Matching difficulties of identities should therefore be an exception.

Identity vigilance in case of uncertainty will be covered on a national level and no double structures have to be created. The use of the National Master Patient Index by the National Pseudonymisation Service does not affect the productivity of the used system. Both systems can be enhanced independently and update paths do not affect each other.

The described functions and the possibility to adapt the properties of an identifier domain for several needs, allows the use of the National Pseudonymisation Service in all Cases from B to E as described in Section *B*.

A. Case B: Cancer Register using National Pseudonymisation Service

The use of the National Pseudonymisation Service as described in Case B is planned for a cancer register.

In the described use case, the sources have access to the clinical data of the patients and will send pseudonymized extracts of this data to the cancer register. Sources can be divided into sources that manage their local identifier, and those who do not manage a local identifier.

The process of sending data from the sources to the cancer register can be described as follows:

1a) Sources with managed local identifer register local identifer and demographcis at the National Pseudonymisation Service

```
Register Identified Person
     Source Domain (Managed)
     Local Identifier
     Demographics of Patient
```

1b) Sources with unmanaged local identifer request local identifer on base of demographics from the National Pseudonymisation Service

```
Register Person
     Source Domain (Unmanaged)
     Demographics of Patient
→    Local/Persistent Identifier
```

2) Request the pseudonym of the cancer register from the National Pseudonymisation Service

```
Translate Identifier
     Source Domain (Managed/Unmanaged)
     Cancer Register Domain
     Local/Persistent Identifier

→    Pseudonym/Persistent Identifier
         of Cancer Register
```

3) Source sends medical data and pseudonym to the cancer register

```
Send Medical Data
     Pseudonym/Persistent Identifier
         of Cancer Register
     Medical Data
```

4) National Cancer Register stores medical data and pseudonym

```
Store Medical Data
     Pseudonym/Persistent Identifier
         of Cancer Register
     Medical Data
```

B. Case E: Biobank using National Pseudonymisation Service

A Luxembourgish biobank currently uses the principles of Case E with the use of a Trusted Third Party as pseudonymisation service. The migration of that service to the National Pseudonymisation Service is planned as soon as the service is available. The specialty of this concept is the uses of the warrant. In the biobank case, cryo-boxes are sent by the biobank to the collection sites. If samples are collected from donors (specimen, blood, urine) the samples are put into the cryo-box that is sent back to the biobank. The kit-ID of the cry-box acts as the warrant in the process of person identification and pseudonym retrieval.

The process can be described as follows:

1) The biobank sends cryo-boxes with unique kit-IDs to the collection sites

```
Send Cryo-Box
     Kit-ID
```

2) Collection sites with unmanaged local identifer request local identifer on base of demographics from the National Pseudonymisation Service

```
Register Person
     Collection Site Domain
     Demographics of Donor
→    Local Identifier
```

3) Collection Sites take samples of donors and stores it into cryo-boxes

```
Collect Samples
     Cryo-box with Kit ID
     Samples of a Donor
```

4) Collection Sites send cryo-boxes to biobank

```
Send Cryo-Box
     Cryo-box with Kit ID
     Samples of a Donor
```

5) Collection Site registers Kit-ID of the cryo-box as warrant at the National Pseudonymisation Service

```
Register Warrant
     Collection Site Domain
     Biobank Domain
     Local Identifier
     Kit-ID
```

6) Biobank request pseudonym at the National Pseudonymisation Service on base of the Kit-ID of the received cryo-box

```
Redeem Warrant
     Biobank Domain
     Kit-ID
→    Pseudonym/Persistent Identifier
```

7) Biobank stores samples in its repository and links it in its Laboratory Information Management System (LIMS) with the pseudonym

```
Store and Manage
     Sample of Donor
     Pseudonym/Persistent Identifier
```

C. Case B: HIV Register using In-House Pseudonymisation

A local HIV register performs long-term studies on HIV. It was created several years ago and recently introduced the concept of in-house pseudonymisation to improve data privacy and data security. Since all tools and mechanisms had been implemented around an existing database structure, it was decided to keep the original database with all the medical data plus the demographics of the patient untouched. Persons who have direct contact to the patients fill this database.

A tool is used to create pseudonymized copies of the original database that contain only research and study specific subsets of the original data. Therefore, the database model is only a subset of the original database model, but it is ensured that none of the used tools have to be adapted. Such extraction, transform and load tools are called ETL tools. The ETL tool will handle all mappings between both database models and finally will create the pseudonym out of a given person identifier.

In the described case, the keeping of mapping tables (person identifier-to-random pseudonym) was not wanted, and the described techniques of hashing or encryption have also not be suitable, since the data type that the person identifier and pseudonym is 4 byte signed integer with 1 as the smallest, and 2^{31}-1 as the highest possible values.

With the algorithm that has been described in *II.M.* Calculation of local identifier with small number of bits, study specific pseudonyms are calculated out of the given person identifier. For each study, a different set of secrets (as listed in Table II) is used as the calculation parameters.

Since the identification of primitive roots is not an easy task, a tool was provided to identify primitive roots.

D. Use cases in discussion

Other Luxembourgish institutes are highly interested in using a National Pseudonymisation Service in the near future, either to secure their existing databases or for newly planned databases.

For some of the analyzed use cases, the use of a National Pseudonymisation Service seems to be far too much and an in-house pseudonymisation is demanded.

IV. Conclusion and Future Work

The use of a National Pseudonymisation Service solves several problems of researchers. It divides infrastructure and personal costs among all users of the national service. It ensures the quality of the underlying demographics that is ensured by the existence of a centralized identity vigilance that already exists for the underlying National Master Patient Index of the National Pseudonymisation Service. The team that performs identity vigilance on national level has permission to solve unclear matching decisions of the Master Patient Index by questioning all sources of demographics. Update of demographics and reconsiderations at the National Pseudonymisation Service about the matching of persons are manageable. With the use of the persistent identifiers, only selective data needs to be updated in case of change of identifiers.

In case of new studies or trials that have to be approved by ethics commissions, questions about data protection will be asked. The use of mechanisms that already have been accepted on a national level will simplify the answering of these questions.

If approved by the ethics commission and with given consent by patients, a National Pseudonymisation Service enables the exchange of data between different studies or trials and link data from the different sources to the same person, even if the sources only use pseudonyms.

The given set of services and the various properties that can be configured for an identifier domain allows the implementation of all described cases A to E. There are always arguments pro and contra the implementation of a certain case, depending on risks to disclose sensitive information. Each designer of a clinical study or trial setup can decide, which of the cases suits most his requirements and data privacy demands.

Even with an up and running National Pseudonymisation Service, the use of in-house pseudonymisation might be the first choice, especially in case of limited participants in the setup. In that case the use of a national service might be far too much, and the costs for the service might be too high. In that case the described algorithm for the creation of pseudonyms out of person identifiers provides a collision free one-way pseudonymisation technique for small bit-depth that still fulfills the requirements of a one-way function, if the secrets behind the calculation are kept secret.

The past has shown that an up-and-running National Pseudonymisation Service improves the willingness to include pseudonymisation solutions already during the design phase of new research databases. This is good, since privacy-by-design strategies are more durable than security patches that are introduced in a later phase of development.

It is expected that with the establishing of the National Pseudonymisation Service, local companies will link their software solution to the national service. Alternatively, consultant companies will offer help in the planning of the integration of the National Pseudonymisation Service into future applications and to find the correct setup (case A to E) that suites most the demands of the customer on data protection and disclosure risks.

ACKNOWLEDGMENT

This material is based upon work supported by the Agence eSanté of Luxembourg.

REFERENCES

[1] U. Roth, "Protecting the privacy with human-readable pseudonyms: One-way pseudonym calculation on base of primitive roots," *Proceedings of the Sixt International Conference on eHealth, Telemedicine, and Social Medicine, eTelemed 2014*, pp. 111–115, 2014.

[2] B. Alhaqbani and C. Fidge, "Privacy-preserving electronic health record linkage using pseudonym identifiers," *10th International Conference on e-health Networking, Applications and Services, HealthCom 2008*, pp. 108–117, 2008.

[3] B. Riedl, V. Grascher, S. Fenz, and T. Neubauer, "Pseudonymization for improving the privacy in e-health applications," *Proceedings of the 41st Annual Hawaii International Conference on System Sciences, HICSS 2008*, p. 255, 2008.

[4] ZorgTTP, "Transparency builds trust," 2012. [Online]. Available from: https://www.zorgttp.nl/userfiles/Downloads/ZorgTTP-englishbrochure-2012.pdf 2014.11.30

[5] K. Pommerening and M. Reng, "Secondary use of the EHR via pseudonymisation," In: *L. Bos, S. Laxminarayan, A. Marsh (eds.): Medical Care Compunetics 1*, pp. 441–446, IOS Press, 2004.

[6] B. Morris, P. Rogaway, and T. Stegers, "How to encipher messages on a small domain," *29th Annual International Cryptology Conference, CRYPTO 2009, Santa Barbara, CA, USA, August 16-20, 2009. Proceedings*, pp. 286–302, LNCS 5677, Springer 2009.

[7] E. Stefanov and E. Shi, "FastPRP: Fast pseudo-random permutations for small domains," *IACR Cryptology ePrint Archive, Report 2012/254*, 2012. [Online]. Available from: http://eprint.iacr.org/2012/254 2014.11.30

[8] S. Dara and S. Fluhrer, "FNR: Arbitrary length small domain block cipher proposal," *Security, Privacy, and Applied Cryptography Engineering, 4th International Conference, SPACE 2014, Pune, India, October 18-22, 2014. Proceedings*, pp. 146–154, LNCS 8804, Springer 2014.

[9] J. Daemen and V. Rijmen, "The design of Rijndael," Springer-Verlag New York, Inc., 2002.

[10] D. Eastlake 3rd and T. Hansen, "US secure hash algorithms (SHA and SHA-based HMAC and HKDF)," Request for Comments 6234, RFC 6234 (Informational), 2011.

[11] T. Ristenpart and S. Yilek, "The mix-and-cut shuffle: Small-domain encryption secure against N queries," *33th Annual International Cryptology Conference, CRYPTO 2009, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I*, pp. 392–409, LNCS 8042, Springer 2013.

[12] A. Faldum and K. Pommerening, "An optimal code for patient identifiers," *Computer Methods and Programs in Biomedicine,* vol. 79, no. 1, pp. 81–88, 2005.

[13] T. Xie, F. Liu, and D. Feng, "Fast collision attack on MD5," *IACR Cryptology ePrint Archive, Report 2013/170*, 2013. [Online]. Available from: http://eprint.iacr.org/2013/170 2014.11.30

[14] L. R. Rivest, A. Shamir, and L. Adleman,"A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM,* Vol 21 (2), pp. 120–126, ACM, 1978.

[15] W. Diffie and M. E. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory,* vol. 22, no. 6, pp. 644–654, IEEE Press, November 1976.

[16] IHE ITI Technical Committee, "IHE IT infrastructure technical framework supplement: Patient identifier cross-reference HL7 V3 (PIXV3) and patient demographic query HL7 V3," IHE, August 10, 2010.

[17] IHE Wiki. *Patient identifier cross-referencing*. [Online]. Available from: http://wiki.ihe.net/index.php?title=Patient_Identifier_Cross-Referencing 2014.11.30

[18] ITU T-HDB-LNG.4-2010. *Object identifiers (OIDs) and their registration authorities.* [Online]. Available from: http://www.itu.int/pub/T-HDB-LNG.4-2010 2014.11.30

[19] S. K. Park and K. W. Miller, "Random number generators: good ones are hard to find," *Commun. ACM,* vol. 31, no. 10, pp. 1192–1201, 1988.