A Secure e-Health Architecture based on the Appliance of Pseudonymization

Bernhard Riedl, Veronika Grascher, Thomas Neubauer
Secure Business Austria, Vienna
Email: {riedl, grascher, neubauer}@securityresearch.ac.at

Abstract— **Due to the cost pressure on the health care system an increase in the need for electronic healthcare records (EHR) could be observed in the last decade, because EHRs promise massive savings by digitizing and centrally providing medical data. As highly sensitive patient information is exchanged and stored within such systems, legitimate concerns about the privacy of the stored data occur, as confidential medical data is a promising goal for attackers. These concerns and the lack of existing approaches that provide a sufficient level of security raise the need for a system that guarantees data privacy and keeps the access to health data under strict control of the patient. This paper introduces the new architecture PIPE (Pseudonymization of Information for Privacy in e-Health) that integrates primary and secondary usage of health data. It provides an innovative concept for data sharing, authorization and data recovery that allows to restore the access to the health care records if the patients' security token is lost or stolen. The concept can be used as basis for national EHR initiatives or as an extension to EHR applications.**

Index Terms— **privacy, security, e-health, pseudonymization, electronic health record, authorization**

I. Introduction

The availability of sound information is essential for health care providers' decisions regarding the patients' care and thus for the quality of treatment and patients' health [1]. Therefore, the idea of nation-wide electronic health records (EHR) has been introduced within the past several years as a method for improving communication and collaboration between health care providers. As life-long medical information about patients is available within such systems, the EHR would also help reducing the alarming number of more than 200.000 cases of death a year in the US [2], caused by adverse drug events (ADE). Additionally, the costs for ADE, which count up to $175 billion a year in the U.S. [3], could be lowered because the health care teams [4] are provided with additional information (e.g., guidelines for drug interactions or sophisticated data produced by decision support systems). EHRs promise massive savings by digitizing medical data like diagnostic tests and images [4]. The non-profit research organization Rand Corporation conducted a study on adopting EHRs in the U.S. under the assumption that 90% of the health care providers used them and concluded that using EHRs could result in more than $81 billion saving per anno [2]. Although a centralized storage could decrease the operational costs of the medical care system, patients are concerned about their privacy. For example, a history containing substance abuse or HIV infection might result in discrimination or harassment. Different stakeholders like insurance companies or research groups demand the disclosure of anamnesis data for billing or R&D reasons. For instance, insurance companies could use the sensitive medical data to deny health coverage or to increase insurance premiums for those affected, whereas employers might refuse to employ people because of their medical history.

Since 2005, the processing and movement of personal data is regulated within the EU by the Directive 95/46/EC [5]. Furthermore, a citizen's right of privacy is recognized in the Article 8 of the European Convention for the Protection of Human Rights and Fundamental Freedoms [6]. Additionally, many domestic acts (e.g., the Austrian Data Protection Act) dictate strict regulations on the processing of personal data. In 2006, the United States Department of Health & Human Services issued the Health Insurance Portability and Accountability Act (HIPAA) [7], which demands the protection of any patient data that is shared from its original source of collection. As medical data tends to be very large (e.g., the image size of a x-ray is 6 MB, for a mammogram 24 MB or for a computer tomography scan counts up to hundreds of MB [8], [9]) and encryption is a highly time-consuming operation, encrypting all data would not be feasible. Using pseudonymization as a privacy-enhancing technology (PET) seems to be a promising approach in order to guarantee patients' privacy, because it allows the association between a certain patient and her datasets only under specified and controlled circumstances. However, although national laws demand the protection of health data stating privacy as a fundamental right of every citizen, existing approaches for pseudonymization (cf. [10]–[15]) have several drawbacks that pose major threats to the privacy and confidentiality of stored patient data, such as centralized lists containing patient-pseudonym-relations, or their dependence on the concealment of the applied algorithms.

In this paper we give an overview of PIPE (Pseudonymization of Information for Privacy in e-Health), a new system for the pseudonymization of health data that differs from existing approaches in its ability to securely integrate primary and secondary usage of health data (cf. [11], [15], [16] for a description of primary and secondary use) and thus provides a solution to security shortcomings of existing approaches. The focus of this

paper lies on introducing a new concept for data sharing, authorization and data recovery. Latter allows to recover the access to the health care records if the security tokens carrying the keys (e.g., a smart card) are lost or stolen. In contrast to existing approaches, our concept does not depend on a patient list that reflects the association between the patient's identity and medical data or a breakable algorithm. Instead, we base PIPE on a layered structure that guarantees that the patients are in full control of their data. The concept can be used as an extension to EHR applications but also as basis for national EHR initiatives.

II. Related Work

Pseudonymization is a technique where identification data is transformed into a specifier and then subsequently replaced by it. The specifier can only be associated with the identification data by means of a certain secret (cf. [17]–[19]). Privacy concerns the collection, storage, use, and disclosure of personal information [20]. As it is necessary to avoid storing any personal information with the pseudonymized dataset to assure patients' privacy, a pseudonymized database has to contain at least two tables, one where all the personal information is hold persistent, and another one which keeps the pseudonyms and the pseudonymized data. The process of identifying and separating personal from related data is called depersonalization [21]. After depersonalization and subsequent pseudonymization, a direct association between certain persons and their data cannot be established. Algorithms for calculating the pseudonym may be based on encrypting or hashing techniques [22]. The latter demands to store a list where all pseudonyms are kept in order to assure reversibility (cf. [11], [15], [23], [24]), but relying on the use of a list is not secure, as an attacker, who gains access to this list, could establish an unauthorized relation between the identification data and the medical data of a specific patient. Encryption provides a more secure alternative for building pseudonyms. For using encryption with a symmetric algorithm (e.g., AES [25]) a secret key, for the asymmetric alternative a key-pair (e.g., RSA [26], ECC [27]), is needed.

As demanded by Kerckhoffs' principle [28] only the keys have to be kept secret, whereas the applied algorithms are accessible. Hence, a major requirement for a secure system is that keys have to be shared with as few people as possible, preferable with nobody. Nowadays, it is a common practice to store keys on smart cards [29], [30]. They are equipped with a small logic chip in order to conduct cryptographic operations without the need to process data on open systems like a standard client (e.g., a personal computer). This technique in combination with a certified card reader assures confidentiality and integrity (cf. [31] for a security taxonomy) of sensitive data during encryption and decryption. In other words, after authenticating against the smart card by entering a PIN, data is transfered to the card reader and afterwards processed on the card's cryptochip. If the PIN is only accessible to the cardholder, this technique can be considered secure [29], [30].

However, as smart cards may be lost, stolen, destroyed or compromised, it is a system's requirement to provide a fall-back mechanisms that allows recovering the key in order to re-establish access to the data which has been encrypted with the smart card. One approach is to keep all keys centralized within the system in a backup keystore which needs to be secured itself. Role-based access control (RBAC) models could be used for handling the authorization and authentication tasks of the backup keystore, but as role-based access control models can be by-passed or compromised [32], [33], a high level of security can only be established by encrypting the keystore itself (cf. [18], [34]–[36]). Nevertheless, persons with administrative roles have to be granted access to the backup keystore for maintenance purposes [10], [18], [34]–[36]. Therefore, this technique does not provide enough security for sensitive health data, because attacks could be performed by people working inside the system, e.g., by social engineering attacks [37]. In order to mitigate this vulnerability, threshold schemes (cf. [38]) can be used to share keys between multiple administrators.

Another shortcoming of existing systems is the patients' dependence on a single pseudonym. If a patient only holds one pseudonym, an attacker who gains access to the database could use data mining [39] for identifying relations between medical data and the patient. For example, only a certain group of patients might have had a knee surgery in a specific time slot. At the same time, only a few people of this group had been treated at a certain hospital and only one of them has seen her dentist a couple of days around the surgery. This example illustrates that the identity of persons can be discovered by combining single occurrences of their anamnesis to conclude a patients' medical history. Therefore, the usage of pseudonymization can only be considered secure if enough disjointed pseudonyms exist. Several approaches for securing EHR architectures have been proposed. The system published by Thielscher et al. (cf. [10]) is based on decentralized keys stored on smart cards. Their approach consists of two databases, one for the patient's identification and one for the anamnesis data. The relation between certain patients and their datasets can only be established by applying the secret key located on the smart card. The system allows to authorize health care providers (HCP) to access specific anamnesis datasets. The major shortcoming of this system is the dependence on a centralized patients-pseudonyms list. This list provides a fall-back mechanism for recovering the relation between patients and their datasets, if patients lose their smart cards. Thielscher et al. circumvent this security flaw by operating the patients-pseudonyms list off-line. This organizational work-around promises a higher level of security until a social-engineering attack is conducted on a system's insider [37], [40] or an attacker gets physical access to the computer which holds this list. Pommerening et al. contributed two different approaches (cf. [11], [15]),

which are both similar to the system of Thielscher et al. Their architecture, which is only applicable for the secondary use of medical data in research centers, is a combination of a hashing and an encryption technique. The encryption itself is based on a centralized secret key, which opens a vulnerability, because if an attacker knows this single key, she might gain access to all patients' related medical data. The approach of Peterson [13] is also based on a centralized table, which is used for reidentification purposes. This table is — from a security point of view — comparable to the approaches of Pommerening or Thielscher, and therefore this architecture relies on the same weak point, because a centralized list is attackable from in- or outside the system. In other words, it is a promising goal for any attacker. In 2001, another architecture was proposed by Schmidt et al. [41]. The underlying security of this system is mainly based on encryption. Consequently, the data is completely or partially encrypted, which is often too time-consuming to be applied for storing medical images.

Based on the mentioned security techniques presented in this chapter, we define the following demands for a system that allows the secure pseudonymization of health care records:

1) Use depersonalization to divide all patient related information into two different tables or databases [21], one for the personal data and the second one for the anamnesis.
2) Replace the foreign key in the anamnesis database, which is related to specific persons, with a pseudonym [19], [22] to assure the patient's privacy.
3) To avoid data mining, every dataset combination of patient, HCP and anamnesis should be defined with a unique pseudonym [18]. For the same reason it is important to hide any relation between interacting persons.
4) Secure the keys used to form pseudonyms and not the algorithm as demanded by Kerckhoff's principle [28].
5) Apply a threshold scheme to share secrets like keys [38]. Moreover, conceal the association between the patients and their responsible administrators. This demand assures that no single person is able to unveil a certain person's identity.
6) Following the previous requirement, the number of administrators, which are assigned to hold a certain person's backup key and the number of administrators, which are necessary to act together to unveil the secret, should be balanced (cf. [38]).
7) Role-based access control models should only be used if the access rights for a certain pseudonym can not be controlled by sharing encrypted secrets (e.g., keys or hidden relations).
8) Provide the patients with the possibility to decide which datasets they want to share by forming an unique pseudonym for the patient herself as well as for any patient-health care provider-anamnesis combination. In addition, hand over all rights to authorize or revoke persons, as far as possible as well as according to the legal situation [5]–[7], [42], to assure that the patient is in full control of her data.

The following section introduces a system overview of our prototype based on the demands stated in the enumeration above.

III. System Overview

The goal of our architecture PIPE [18], [34] is to gain the optimal trade-off between security on the one hand and usability and performance on the other hand. We outline the roles and components of PIPE and continue with a presentation of the design principles and applied security methods.

A. Architecture

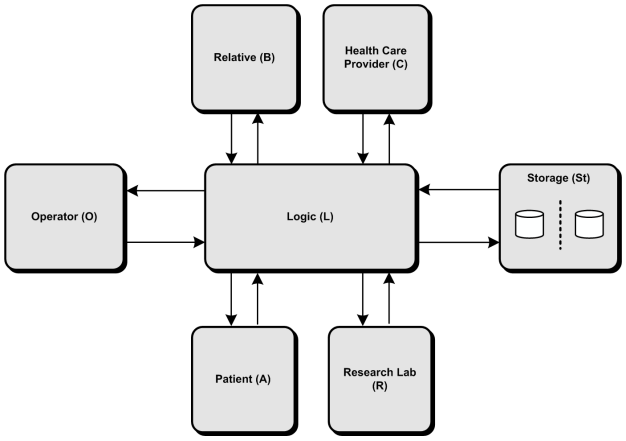

Figure 1. PIPE Architecture [18], [34]

Our architecture (cf. Figure 1) consists of the following users $\mathcal{U}$ and components:

- A central system (e.g., server, etc.) which provides access to a central storage (St) which itself is divided (e.g., logical, physical) into two separate storage systems (e.g., databases, etc.), where one is related to identification data and the other one is related to data, which should be pseudonymized as well as the associated pseudonym,
- a central logic (L) that provides an interface between the central storage and the clients for the purpose of saving and loading the data,
- the patients ($\mathcal{A}$) who have full access to their data on the central system via the central logic by using a security token (e.g., smart card with a PIN, biometric authentication, etc.),
- the relatives ($\mathcal{B}$) who might get the same rights as the patient by default, if not supervised by a role-based access control model,
- the health care providers ($\mathcal{C}$) who share one or several entries in the pseudonymized database with the patients,

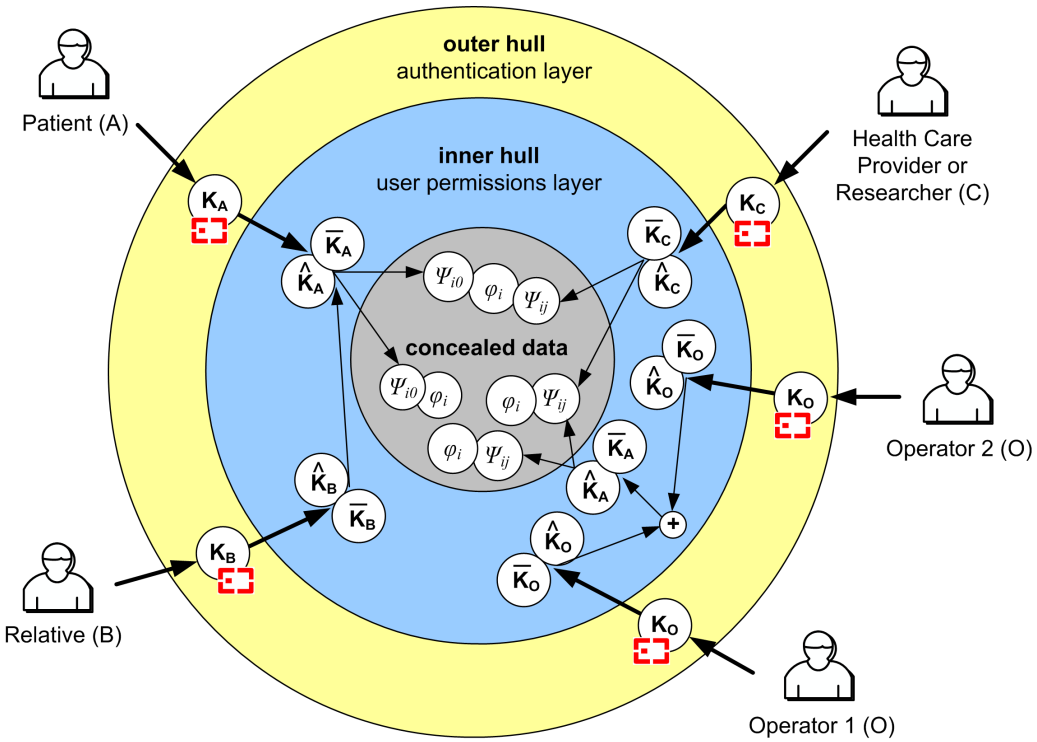

Figure 2. PIPE security hull architecture [18], [34]–[36]

- the research labs ($\mathcal{R}$) that have solely access to the anamnesis data on the server system via the central logic for the purpose of analysis needed for improving the efficiency of clinical trails, the medical treatment, or medication and
- the operator(-team) ($\mathcal{O}$) which can hold secrets on behalf of the users. This role assures that if a patient loses or destroys her smart card, the access to the system can be restored by a team of operators.

In practice the logic L, and the storage St, which might be outsourced to a data processing center, have to form a trusted instance, because smart card management is handled there.

B. Security Model

Table 1 gives an overview of the keys and abbreviations used to describe our system. Note, that all private keys (where K stands for key) are identified as K^{-1} (e.g., the patient's inner private key will be named $\widehat{K}_A^{-1}$). As shown in figure 2, PIPE is based on a hull-architecture [18], [34]–[36]. Every hull consists of one or more secrets (e.g., encrypted keys or hidden relations) which are only accessible with the unveiled secrets from the next outer hull. For instance the patient's inner private key $\widehat{K}_A^{-1}$ in the inner hull — or user permissions layer of the patient A — is encrypted with the outer public key K_A on her smart card, which represents the outer hull or authentication layer. A specific anamnesis dataset φ_i, which is associated with a list of j pseudonyms ψ_{i_j}, can only be accessed with the knowledge of the related secret, which has been encrypted with the inner symmetric key $\overline{K}_A$. As the inner symmetric key has been preliminary encrypted with the inner public key, this encryption operation has to be reversed to gain access to this key in plain-text. In other words, if a patient wants to access her data, she has to decrypt her inner private key $\widehat{K}_A^{-1}$. Latter is stored encrypted inside the system with the outer public key K_A of her smart card. Afterwards, she is able to decrypt the inner symmetric key $\overline{K}_A$ with her inner private key and can use the inner symmetric key, which is now available for her in plain-text, to access the encrypted secrets in the most inner hull – the concealed data hull – by decrypting them. Consequently, to get access to the data, every user has to 'peel the hulls'. We provide a formal example of a patient's enveloped secret in equation (1). If a key has been applied as subscripted character, the message has been encrypted with this key.

$$\left\{\left\{\left\{\{\psi_{i_0} \mapsto \varphi_i\}_{\overline{K}_A}\right\}_{\widehat{K}_A}\right\}_{K_A}\right\} \tag{1}$$

In our system, secrets can be shared between users for authorization purposes. First of all, a patient may provide a relative with her inner private key $\widehat{K}_A^{-1}$, which will then be encrypted with the relative's inner symmetric key $\overline{K}_B$. By doing this, the relative gets access to all data of the patient, until the inner private key is changed. Moreover, a health care provider can be authorized to access a subset of anamnesis datasets by sharing secrets, in our approach pseudonyms, of the concealed hull. A special case of a pseudonym, a so-called root pseudonym ψ_{i_0} exists for every dataset. This root pseudonym is only related with the patient and the anamnesis and no other user than the

TABLE I.
DEFINITION OF SYSTEM ATTRIBUTES

	Patient	*Relative*	*HCP*	*Operator*	*Logic*
abbreviation	A	B	C	O	L
unique identifier	A_{id}	B_{id}	C_{id}	O_{id}	
(outer public key, private key)	(K_A, K_A^{-1})	(K_B, K_B^{-1})	(K_C, K_C^{-1})	(K_O, K_O^{-1})	
(inner public key, private key)	$(\widehat{K}_A, \widehat{K}_A^{-1})$	$(\widehat{K}_B, \widehat{K}_B^{-1})$	$(\widehat{K}_C, \widehat{K}_C^{-1})$	$(\widehat{K}_O, \widehat{K}_O^{-1})$	
inner symmetric key	$\overline{K}_A$	$\overline{K}_B$	$\overline{K}_C$	$\overline{K}_O$	K_L
key share	$\sigma_\iota(K)$				
medical data / anamnesis	φ_i				
pseudonym	ψ_{i_j}				

patient herself is able to delete this pseudonym. All other pseudonyms may be removed from the storage without authorized users' permission. For example, if two health care providers are related to see a specific anamnesis, three pseudonyms (ψ_{i_0}, ψ_{i_1}, ψ_{i_2}) exist. Both pseudonyms (ψ_{i_1}, ψ_{i_2}), which are shared between a patient and a health care provider, may be deleted without the particular health care provider's notification. The patient is the only person who can delete all pseudonyms. This assures that the patients are in full control of their data, and authorizing as well as revoking of all users is possible at any times (e.g., as defined by European Law). In case other legal acts demand that the health care provider should be the owner of the patient's data, the HCP could hold the root pseudonym on behalf of the patient. Moreover, a rule could be added to our role-based access control model, which verifies, if a patient should be able to revoke rights of certain persons or institutions for a specific anamnesis.

IV. ESTABLISHING A SECURE BACKUP KEYSTORE

As already mentioned, there is the need to assure that users still have access to their data if they lose their smart cards. In our system we provide a fall-back mechanism by sharing the user's inner private key, which grants access to the inner symmetric key and subsequently to the pseudonyms. For instance, a relative could hold an encrypted version of the user's inner private key for backup reasons and — if not controlled by a role-based access control model — would have the same rights as the patient. Users might not want to provide all their data to a certain relative. As a consequence, a backup of the necessary keys must be stored inside the system (e.g., for recovering keys and issuing new smart cards). As these keys have to be divided between more persons, we applied Shamir's threshold scheme [38] to divide the user's inner private key $\widehat{K}_A^{-1}$ into n shares $\sigma_\iota(\widehat{K}_A^{-1})$. At least k of these n shares are necessary to reconstruct the whole key. The n shares are randomly distributed amongst a set of operators, which we define as assigned operators. Any assigned operator may only hold a maximum of one share of a certain user's key. As a result, k necessary operators for every user exist, which have to act together to unveil the patient's key. We named the set of assigned operators $\mathcal{O}^n \subset \mathcal{O}$ and the subset of necessary operators $\mathcal{O}^k \subseteq \mathcal{O}^n$. The delta between the number of assigned operators and necessary operators may be seen as backup operators, in case an operator is not available.

Following Shamir [38], it is not possible to compute the key by combining $k-1$ shares, but if an attacker is able to bribe $b \geq k$ operators, she may succeed in unveiling a certain user's identity. Equation (2) states the probability of guessing at least the necessary operators for a specific user under the condition that the operators do not know for whom they are holding shares.

$$P(k \leq X \leq n) = \sum_{\iota=k}^{n} \frac{\binom{n}{\iota}\binom{|\mathcal{O}|-n}{b-\iota}}{\binom{|\mathcal{O}|}{b}} \tag{2}$$

This equation leads to the following conclusions: The larger the group of operators, the lower the probability that an attacker could bribe the assigned ones to expose a certain patient's identity. The lower the minimum of operators necessary to unveil the secret compared to the number of operators assigned to a certain patient, the higher the probability for a misuse of the system. If the operators do not know for which person they share secrets, an attacker has to compromise all operators minus the number of backup operators in the worst case. To conceal the relation between an operator and a certain patient, the system encrypts the shares $\sigma_\iota(\widehat{K}_A^{-1})$ first of all with its logic key K_L and afterwards with the inner public keys $\widehat{K}_O$ of the operators. Hence, only if an operator knows K_L it is possible for her to unveil the relation, but she needs more operators to rebuild the shared secret. Thus, the possibility for arrangements between the operators is lowered. Following these constraints, we firstly present the formal workflow for recovering a lost key and secondly we discuss the security of different examples of n assigned and k necessary operators combinations.

A. Recovering a Lost Key

For information exchange between two or more actors, we use the notation of $i^{th}\ workflow\ step : Sender \rightarrow Receiver \rightarrow \ldots \rightarrow Receiver{:}\{Message\}$. For example, the third message in a workflow between the patient and the logic, encapsulating the encrypted patient's identifier encrypted with the patient's inner symmetric key would look like 3: $A \rightarrow L{:}\Big\{\{A_{id}\}_{\overline{K}_A}\Big\}$.

A pre-condition for conducting a workflow in our prototype is the step 1, the authentication of all participating users $\mathcal{U}$ against the system.

$$f_{authenticate}(U_{id}) := \begin{cases} \left\{\widehat{K}_U^{-1}\right\}_{K_U} & U_{id} \in St \\ errorcode & U_{id} \notin St \end{cases} \quad (3)$$

1: $U \rightarrow L$:$\{U_{id}\}$, $L \rightarrow St$:$\{f_{authenticate}(U_{id}) =?\}$, $St \rightarrow L \rightarrow U$:$\left\{\left\{\widehat{K}_U^{-1}\right\}_{K_U}\right\}$

The user U authenticates against her smart card by entering her PIN. If the PIN matches, the certificate of the client software is used to sign the user's identifier U_{id}. This signed identifier is transmitted to the logic which verifies this signature. If the certificate and the signature are valid, the logic queries the storage for the encrypted inner private key $\widehat{K}_U^{-1}$ of the specific user and forwards it to the user. She decrypts her inner private key with her outer private key K_U^{-1}. Therefore, even if the keystore in the client application has been successfully compromised, an attacker who steals the encrypted inner private key is not able to use it to gain access to the pseudonymized datasets until she does not have access to the outer private key, too.

Necessary operations: mutual authentication, one SQL select statement, decrypt inner private key

In order to rebuild a lost smart card with access to the patient's inner private key, the patient identifies against an operator. This operator does not hold a part of this patient's inner private key. In fact, she just initiates the recovering process by sending a message to the logic.

Necessary operations: proof patient's identity

2: $L \rightarrow O$:$\left\{\{A_{id}\}_{K_L}\right\} \forall\ \mathcal{O}$

The central logic broadcasts a message to all operators $\mathcal{O}$ with an encrypted version of the patient's identifier A_{id} because the logic key K_L has been used to envelope the identifier first.

Necessary operations: encrypt patient's identifier

3: $O \rightarrow L \rightarrow St$:$\left\{\left\{\{A_{id}\}_{K_L}\right\}_{\overline{K}_O}\right\} \forall\ \mathcal{O}$

Upon receipt, all operators query their backup keystore via the central logic by encrypting these received ciphertexts with the particular operator's inner symmetric key $\overline{K}_O$. With this message the logic is able to find out which operator possesses a patient's key share.

Necessary operations: encrypt shares by $|\mathcal{O}|$ operators

4: $St \rightarrow L \rightarrow O$:$\left\{\left\{\left\{\sigma_\iota(\widehat{K}_A^{-1})\right\}_{K_L}\right\}_{\overline{K}_O}\right\} \forall\ \mathcal{O}^n$

After querying the double encrypted ciphertexts against the storage, the logic receives the associated double encrypted key shares and forwards them to the assigned operators.

Necessary operations: $|\mathcal{O}^n|$ SQL select statements

5: $O \rightarrow L$:$\left\{\left\{\sigma_\iota(\widehat{K}_A^{-1})\right\}_{K_L}\right\} \forall\ \mathcal{O}^n$

The next step is that all assigned operators decrypt their particular shared secrets with their inner symmetric key $\overline{K}_O$ and transmit them to the logic. The logic is now able to decrypt these shares with its key K_L and consequently to combine the parts. As soon as the logic receives the shares from a minimum number of k necessary operators, the patient's inner private key can be re-calculated with appliance of Shamir's threshold scheme [38].

Necessary operations: decrypt a maximum of $|\mathcal{O}^n|$ key shares, apply threshold scheme

6: $L \rightarrow St$:$\left\{\left\{\widehat{K}_A^{-1}\right\}_{K_{A'}}\right\}$

Afterwards, the logic retrieves a new outer key pair $(K_{A'}, K_{A'}^{-1})$ from the storage which will replace the outer keys (K_A, K_A^{-1}) of the lost smart card. The logic uses the new outer public key to encrypt the patient's inner private key. The logic saves this ciphertext in the storage and initates the smart card production. To avoid replay-attacks the storage moreover deletes the operator shares and their relations to the patient.

Necessary operations: generate new asymmetric key pair, encrypt patient's inner private key, $|\mathcal{O}^n|$ SQL delete statements

7: $L \rightarrow O$:$\left\{\left\{\left\{\sigma_\iota(\widehat{K}_A^{-1}), A_{id}\right\}_{K_L}\right\}_{\widehat{K}_O}\right\} \forall\ \mathcal{O}^n$

Subsequently, the logic randomly chooses $\mathcal{O}^n$ assigned operators and uses the threshold scheme to divide the patient's inner private key into n shares. Once more, all shares will be double-enveloped. Firstly, the logic applies its key K_L and secondly, encrypts the gained ciphertexts with the certain inner public keys $\widehat{K}_O$ of the selected operators. These encrypted secret shares will then be transmitted to the operators. Moreover, the logic applies the same encryption procedures to the patient's ID A_{id} and transfers this ciphertext to the operators, too.

Necessary operations: apply threshold scheme, encrypt shares and patient's identifier twice for $\mathcal{O}^n$ operators

8: $O \rightarrow L \rightarrow St$:$\left\{\left\{\left\{\sigma_\iota(\widehat{K}_A^{-1}), A_{id}\right\}_{K_L}\right\}_{\overline{K}_O}\right\} \forall\ \mathcal{O}^n$

Upon receipt, the assigned operators decrypt their particular shares and the patient's identifier with their inner private keys $\widehat{K}_O^{-1}$. Then they encrypt both attributes again with their inner symmetric keys $\overline{K}_O$ and return these ciphertexts to the logic which saves them in the storage.

Necessary operations: decrypt and encrypt the key shares and the patient's identifier for $\mathcal{O}^n$ operators; $|\mathcal{O}^n|$ SQL insert statements to store the ciphertexts in the database

B. Security Investigations on the Backup Keystore

The trade-off between the probability that an attacker is able to bribe enough operators to compromise the privacy of a specific patient and the difference between the assigned and necessary operators, representing the fall-back mechanism, is a vital information for granting security, subsequently confidentiality and availability obligations

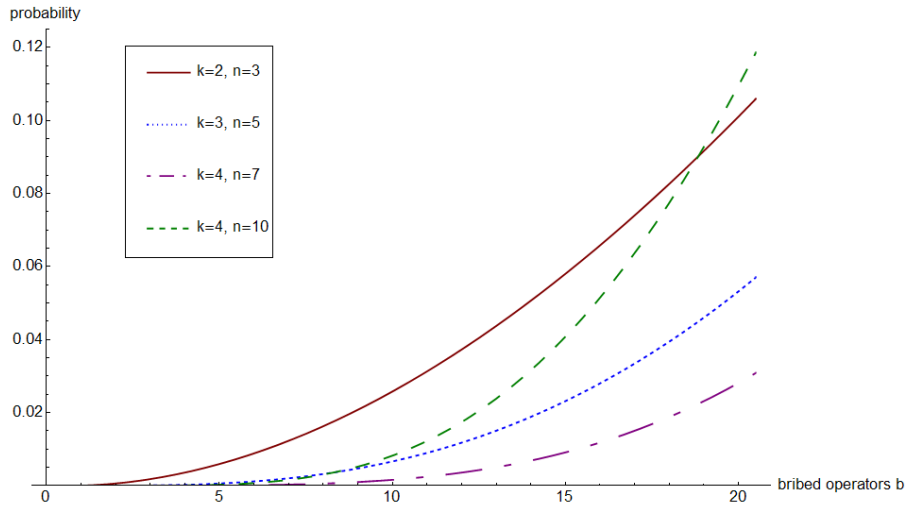

Figure 3. Different combinations of assigned and necessary operators for a sample of ≤ 20 bribed operators

(cf. [31] for a security taxonomy). We define a successful attack by bribing $b \geq k$ operators of all assigned operators for a certain patient in a system under the constraint that the operators are randomly assigned to the patients, and the operators do not know for which patients they hold shares. In other words an attacker does not gain more knowledge about the relation between operators and patients by successfully bribing one except the certain operator's share of the secret.

Shamir stated, that a minimum of $n = 2k - 1$ users, in our case operators, required to unveil a certain secret, makes a "very robust key management scheme" [38]. We therefore define all four operator combinations with this constraint. From an economical point of view, the combination of 2 $necessary$ and 3 $assigned$ operators results in the minimal costs for applying the threshold scheme with the minimum fall-back of one backup operator. Note, that the higher we set the number of necessary operators, the lower the quantity of users they are able to serve, because it consumes working time to handle recovering key requests. Following Shamir, a security model with 3 $necessary$ and 5 $assigned$ as well as 4 $necessary$ and 7 $assigned$ operators are the next larger possible variations. Moreover, we state a system with higher costs but significant fall-out rate of 6 operators, consisting of 4 $necessary$ and 10 $assigned$ operators, which would also raise the processing time of recovering key requests. Consequently, if we want to compare the results of all combinations we have to start the investigation with a minimum of 4 bribed operators.

Figure 3, based on the results of table II presents the behavior of a system with 100 operators and the four defined settings of assigned and necessary operators for the range of $\leq$ 20 bribed operators. We see, that the probability for bribing 4 operators and a combination of $k = 2, n = 3$ tends towards a value of 0.004, in other words 0.4 percent, whereas the percentages of all

TABLE II.
DIFFERENT COMBINATIONS OF ASSIGNED AND NECESSARY OPERATORS FOR A SAMPLE OF $\leq$ 20 BRIBED OPERATORS

	k=2, n=3	*k=3, n=5*	*k=4, n=7*	*k=4, n=10*
b=4	**0,0036**	**0,0002**	$<$ **0,0001**	**0,0001**
b=5	0,0059	0,0006	$<$ 0,0001	0,0003
b=6	0,0088	0,0012	0,0001	0,0007
b=7	0,0123	0,0020	0,0003	0,0016
b=8	0,0163	0,0032	0,0006	0,0030
b=9	0,0208	0,0047	0,0010	0,0052
b=10	**0,0258**	**0,0066**	**0,0016**	**0,0082**
b=11	0,0313	0,0090	0,0025	0,0123
b=12	0,0373	0,0118	0,0036	0,0174
b=13	0,0437	0,0151	0,0050	0,0238
b=14	0,0506	0,0188	0,0069	0,0316
b=15	0,0580	0,0232	0,0091	0,0408
b=16	0,0658	0,0280	0,0118	0,0515
b=17	0,0740	0,0334	0,0150	0,0637
b=18	0,0826	0,0394	0,0188	0,0775
b=19	0,0917	0,0460	0,0232	0,0928
b=20	**0,1011**	**0,0532**	**0,0281**	**0,1096**

other constellations are still nearly zero. If an attacker is able to bribe $b = 10$ operators, the probability of compromising the privacy of a certain user in a system with $k = 2, n = 3$ operators is somewhat 2.6 percent, for $k = 3, n = 5$ as well as $k = 4, n = 10$ slightly more than 0.5 percent. The combination of 4 necessary and 7 operators still tends towards zero. If in average 20 out of 100 operators are acting corruptly, it is the first time the approach representing the largest number of backup operators ($k = 4, n = 10$) gains the lowest security. Again, there is a significant difference between the previously mentioned approaches, which reach up to somewhat 10 percent, compared to 5 percent for $k = 3, n = 5$ and slightly less than 3 percent for $k = 4, n = 7$. The best security level for a maximum of 20 percent of bribable operators can be achieved with 4 necessary and 7 assigned operators (less than 3 percent). In the following we provide a comparison of different numbers

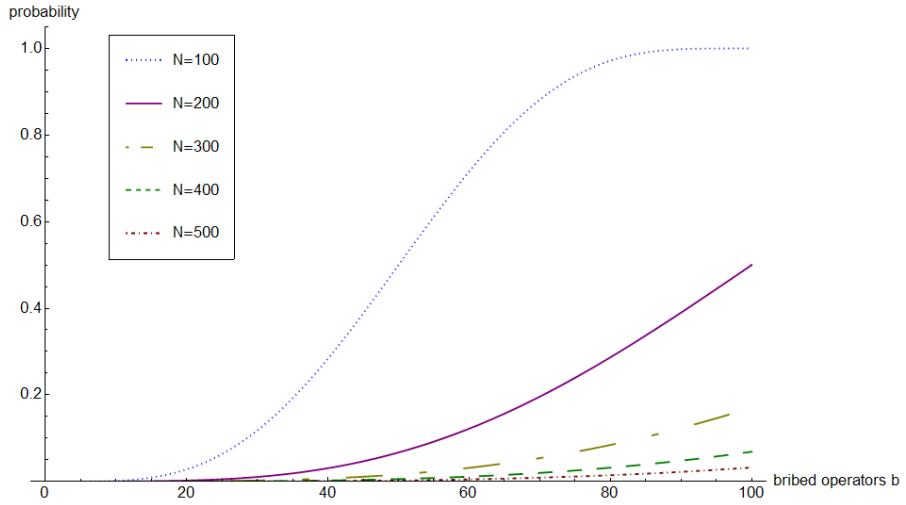

Figure 4. The combination of 4 necessary and 7 assigned operators compared for varying numbers of operators

of total operators for the combination of 4 necessary and 7 assigned operators.

TABLE III.
THE COMBINATION OF 4 NECESSARY AND 7 ASSIGNED OPERATORS COMPARED FOR VARYING NUMBERS OF OPERATORS IN THE SYSTEM

	$\lvert\mathcal{O}\rvert = 100$	200	300	400	500
b=10	0,0016	0,0001	0,0001	$<$ 0,0001	$<$ 0,0001
b=20	**0,0281**	**0,0021**	**0,0004**	**0,0001**	**0,0001**
b=30	0,1179	0,0106	0,0023	0,0008	0,0003
b=40	0,2837	**0,0307**	0,0071	0,0024	0,0010
b=50	0,5000	**0,0670**	**0,0164**	**0,0057**	**0,0025**
b=60	0,7163	0,1221	**0,0316**	0,0113	0,0050
b=70	0,8821	0,1960	0,0540	0,0199	0,0089
b=80	0,9719	0,2868	0,0845	**0,0320**	0,0146
b=90	0,9984	0,3901	0,1234	0,0482	0,0223
b=100	1,0000	0,5000	0,1707	0,0688	**0,0323**

Figure 4, which is based on the results of table III, shows, that the probability of bribing 20 operators, is converging to zero for all constellations equal or greater 200 operators. If 50 operators are acting corruptly, the probabilities for systems with 300, 400 and 500 operators are about 1 percent, whereas for a total of 200 operators it is slightly less than 7 percent. Regarding a number of 100 bribed operators, which represents one fifth of a system with 500 operators, the probability is still significantly lower than 3.5 percent. This result can be compared to a system of 100 operators with 20, 200 operators with 40, 300 operators with 60 and 400 operators with 80 bribed operators. Hence, the security of the combination of 4 necessary and 7 assigned operators is also reliable for different numbers of operators. We conclude that the maximum level of security respectively the lowest probability to bribe enough operators to find out a certain user's key can be gained with 7 assigned and 4 necessary operators.

V. CONCLUSIONS

Electronic health records not only promise a significant reduction of the costs for managing medical information, they also achieve a higher level of service quality for patients [2]. As highly sensitive data is stored and handled in nation-wide medical systems, there is an increasing demand for assuring the patients' privacy in order to avoid misuse. Although national laws and regulations have been set-up in order to assure patients' privacy, the security of existing approaches presented in this article is often too weak to assure confidentiality of life-long medical data storage. This especially holds for their dependence on a centralized patient-pseudonyms list, a life-long pseudonym or the concealment of an algorithm. Based on these shortcomings we introduced the secure architecture PIPE for the combined primary and secondary usage of health-related data. Our system assures that the patients are in full control of their data with the maximum of gainable security, achieved by applying authorization on encryption, in- and outside the system as well as for all communication. In other words, even if all communication between the actors is transmitted over unsecure channels like the Internet, the confidentiality is granted because all attributes in the database are already secured by encryption with different keys. For integrity purposes, we additionally propose the usage of Transport Layer Security (TLS), signed messages or the creation of hash-values for database tuples.

As users in our system possess smart cards as security tokens, we applied a secure fall-back mechanism, in case a smart card is lost, stolen, compromised or just worn out. Therefore, we introduced the administrative role operator who holds a backup of the user keys. We applied Shamir's threshold scheme to securely divide the backup keys between the operators in order to assure inner system's security. We gave a recommendation for the distribution of operators, which are assigned to hold a certain person's

secret, and the number of operators, which are necessary to unveil that secret, regarding different numbers of operators. Although this backup-mechanism assures a very high level of security, an increasing number of operators would result in high operational costs, especially if all operators are human beings. For reducing the costs we propose a combination of humans equipped with smart cards as well as hardware security modules (cf. [43]) which could act on behalf of human operators. Further research will especially focus on conducting experiments with our business partners. Beside that, we will introduce a mechanism for ad-hoc access to a subset of the anamnesis data, the patient's emergency data.

ACKNOWLEDGMENT

We want to thank Karl Grill and Erich Neuwirth for their support on our statistical model, our master thesis students Mathias Kolb and Markus Pehaim as well as the members of our business partner Braincon Technologies, Oswald Boehm, Alexander Krumboeck, and Gert Reinauer for their support, further Stefan Jakoubi for his review.

This work was performed at Secure Business Austria, a competence center that is funded by the Austrian Federal Ministry of Economics and Labor (BMWA) as well as by the provincial government of Vienna.

REFERENCES

[1] S. Maerkle, K. Koechy, R. Tschirley, and H. U. Lemke, "The PREPaRe system – Patient Oriented Access to the Personal Electronic Medical Record," in *Proceedings of Computer Assisted Radiology and Surgery, Netherlands*, 2001, pp. 849–854.

[2] F. R. Ernst and A. J. Grizzle, "Drug-related morbidity and mortality: Updating the cost-of-illness model," University of Arizona, Tech. Rep., 2001.

[3] ——, "Drug-related morbidity and mortality: Updating the cost-of-illness model," University of Arizona, Tech. Rep., 1995.

[4] J. Pope, "Implementing EHRs requires a shift in thinking. PHRs–the building blocks of EHRs–may be the quickest path to the fulfillment of disease management." *Health Management Technology*, vol. 27(6), p. 24, 2006.

[5] European Union, "Directive 95/46/ec of the european parliament and of the council of 24 october 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data." *Official Journal of the European Communities*, vol. L 281, pp. 31–50, 1995, http://europa.eu/scadplus/leg/en/lvb/l14012.htm.

[6] Council of Europe, *European Convention on Human Rights*. Martinus Nijhoff Publishers, 1987.

[7] United States Department of Health & Human Service, "Hipaa administrative simplification: Enforcement; final rule," *Federal Register / Rules and Regulations*, vol. Vol. 71, No. 32, 2006.

[8] M. Ackerman, R. Craft, F. Ferrante, M. Kratz, S. Mandil, and H. Sapci, "Telemedicine technology," *Telemedicine Journal and e-Health*, vol. 8, No. 1, pp. 71 –78, 2002.

[9] J. Montagnat, F. Bellet, H. Benoit-Cattin, V. Breton, L. Brunie, H. Duque, Y. Legr, I. E. Magnin, L. Maigne, S. Miguet, J. M. Pierson, L. Seitz, and T. Tweed, "Medical images simulation, storage, and processing on the european datagrid testbed," *Journal of Grid Computing*, vol. 2, Number 4, pp. 387–400, 2004.

[10] C. Thielscher, M. Gottfried, S. Umbreit, F. Boegner, J. Haack, and N. Schroeders, "Patent: Data processing system for patient data," *Int. Patent, WO 03/034294 A2*, 2005.

[11] K. Pommerening and M. Reng, *Medical And Care Compunetics 1*. IOS Press, 2004, ch. Secondary use of the Electronic Health Record via pseudonymisation, pp. 441–446.

[12] G. de Moor, B. Claerhout, and F. de Meyer, "Privacy enhancing technologies: the key to secure communication and management of clinical and genomic data," *Methods of information in medicine*, vol. 42, pp. 148–153, 2003.

[13] R. L. Peterson, "Patent: Encryption system for allowing immediate universal access to medical records while maintaining complete patient control over privacy," *US Patent US 2003/0074564 A1*, 2003.

[14] J. Gulcher, K. Kristjansson, H. Gudbjartsson, K., and Stefanson, "Protection of privacy by third-party encryption in genetic research," *European journal of human genetics*, vol. 8, pp. 739–742, 2000.

[15] K. Pommerening, "Medical Requirements for Data Protection," in *Proceedings of IFIP Congress, Vol. 2*, 1994, pp. 533–540. [Online]. Available: citeseer.ist.psu.edu/330589.html

[16] D. Lobach and D. Detmer, "Research challenges for electronic health records," *American Journal of Preventive Medicine*, vol. 32, Issue 5, pp. 104–111, 2007.

[17] A. Pfitzmann and M. Koehntopp., "Anonymity, Unlinkability, Unobservability, Pseudonymity, and Identity Management A Consolidated Proposal for Terminology," in *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 2005.

[18] B. Riedl, T. Neubauer, G. Goluch, O. Boehm, G. Reinauer, and A. Krumboeck, "A secure architecture for the pseudonymization of medical data," in *Proceedings of the Second International Conference on Availability, Reliability and Security*, 2007, pp. 318–324.

[19] K. Taipale, "Technology, Security and Privacy: The Fear of Frankenstein, the Mythology of Privacy and the Lessons of King Ludd," *International Journal of Communications Law & Policy*, vol. 9, 2004.

[20] J. C. Cannon, *Privacy: What Developers and IT Professionals Should Know*. Addison-Wesley, 2004.

[21] A. Rector, J. Rogers, A. Taweel, D. Ingram, D. Kalra, J. Milan, P. Singleton, R. Gaizauskas, M. Hepple, D. Scott, and R. Power, "Clef - joining up healthcare with clinical and post-genomic research," in *Proceedings of UK e-Science All Hands Meeting*, 2003, pp. 203–211. [Online]. Available: citeseer.ist.psu.edu/rector03clef.html

[22] A. Lysyanskaya, R. L. Rivest, A. Sahai, and S. Wolf, "Pseudonym systems," in *Proceedings of the Sixth Annual Workshop on Selected Areas in Cryptography (SAC '99)*. [Online]. Available: citeseer.ist.psu.edu/lysyanskaya99pseudonym.html

[23] J. Biskup and U. Flegel, "Transaction-based pseudonyms in audit data for privacy respecting intrusion detection," in *Proceedings of RAID '00: Proceedings of the Third International Workshop on Recent Advances in Intrusion Detection*. London, UK: Springer-Verlag, 2000, pp. 28–48.

[24] U. Flegel, "Pseudonymizing unix log files," in *Proceedings of the International Conference on Infrastructure Security*. London, UK: Springer-Verlag, 2002, pp. 162–179.

[25] J. Daemen and V. Rijmen, *The Design of Rijndael: AES - The Advanced Encryption Standard*. Springer; 1 Edition, 2002.

[26] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[27] V. S. Miller, "Use of elliptic curves in cryptography," *Lecture notes in computer sciences*, vol. 218 on Advances in cryptology—CRYPTO 85, pp. 417–426, 1986.
[28] B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*. Wiley; 2 edition, 1995.
[29] M. Hendry, *Smart Card Security and Applications, Second Edition*. Norwood, MA, USA: Artech House, Inc., 2001.
[30] W. Rankl and W. Effing, *Smart Card Handbook*. New York, NY, USA: John Wiley & Sons, Inc., 1997.
[31] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," *IEEE Transactions on Dependable and Secure Computing*, vol. 1, no. 1, pp. 11–33, 2004.
[32] R. Russell, D. Kaminsky, R. F. Puppy, J. Grand, D. Ahmad, H. Flynn, I. Dubrawsky, S. W. Manzuik, and R. Permeh, *Hack Proofing Your Network (Second Edition)*. Syngress Publishing, 2002.
[33] T. Westran, M. Mack, and R. Enbody, "The last line of defense: a host-based, real-time, kernel-level intrusion detection system," in *submitted to IEEE Symposium on Security and Privacy*, 2003.
[34] B. Riedl, T. Neubauer, and O. Boehm, "Patent: Datenverarbeitungssystem zur Verarbeitung von Objektdaten," *Austrian-Patent, No. A 503 291 B1*, 2007.
[35] B. Riedl, V. Grascher, and T. Neubauer, "Applying a threshold scheme to the pseudonymization of health data," in *to appear in the proceedings of the 13th IEEE Pacific Rim International Symposium on Dependable Computing (PRDC07)*, 2007.
[36] B. Riedl, V. Grascher, S. Fenz, and T. Neubauer, "Pseudonymization for improving the privacy in e-health applications," in *to appear on Proceedings of the Forty-First Hawai'i International Conference on System Sciences*, 2008.
[37] T. Thornburgh, "Social engineering: the "Dark Art"," in *Proceedings of the 1st annual conference on Information security curriculum development*. New York, NY, USA: ACM Press, 2004, pp. 133–135.
[38] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, 1979.
[39] J. Han and M. Kamber, *Data mining: concepts and techniques*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2000.
[40] K. Maris, "The Human Factor," in *Proceedings of Hack.lu, Luxembourg*, 2005.
[41] V. Schmidt, W. Striebel, H. Prihoda, M. Becker, and G. D. Lijzer, "Patent: Verfahren zum be- oder verarbeiten von daten," *German Patent, DE 199 25 910 A1*, 2001.
[42] Republic of Austria, "Datenschutzgesetz 2000 (DSG 2000), BGBl. I Nr. 165/1999," 1999.
[43] D. C. Wherry, "Secure your public key infrastructure with hardware security modules," SANS Institute, Tech. Rep., 2003.

Bernhard Riedl is currently a Ph.D. candidate at the Technical University of Vienna. He graduated with a Master in Economics and Computer Science and a Master in Software Engineering and Internet Computing (Computer Sciences) at the Technical University of Vienna as well as a Master in Informatics Management at the University of Vienna. He has been self-employed in Software Development for about ten years. Moreover he taught several university courses in the field of Quality Assurance as well as Software Project Management and participated in research at the Institute for Software Technology and Interactive Systems at the Technical University of Vienna. Furthermore, he initiated and participates in the development of a couple of open-source projects. Mr. Riedl is a project manager and researcher of Secure Business Austria in Vienna. His research focuses on concepts of privacy in information systems.

Veronika Grascher is currently a MS candidate at the Alps-Adriatic University of Klagenfurt. She participated at the Institute of Mathematics of the Alps-Adriatic University of Klagenfurt, research group of Linear Algebra. Furthermore she lead several Analysis and Algebra-workshops. Ms. Grascher is a project member at Secure Business Austria. Her research focuses on algebraic applications in IT-Security.

Thomas Neubauer is senior researcher and project manager at Secure Business Austria and lecturer at the Institute of Software Technology and Interactive Systems (IFS) at the Vienna University of Technology. His research focuses on the integration of security concepts to business process management with a special focus on e-health. He further applies approaches from multiobjective decision support to security to provide support in allocating optimal portfolios of security safeguards. He received a Master in Business Informatics from the University of Vienna, a Master in Computer Science from the Vienna University of Technology and finished his PhD thesis at the Institute of Software Technology and Interactive Systems at the Vienna University of Technology. Dr. Neubauer worked for two years in the financial sector. He was consultant for The Austrian Federal Chancellery (CIO Office) and Austrian Social Security Institutions.