Flexible Patient-Controlled Security for Electronic Health Records

Thomas Hupperich[1] Hans Löhr[1] Ahmad-Reza Sadeghi[2] Marcel Winandy[1]

[1]Ruhr-University Bochum, Germany
{firstname.lastname}@trust.rub.de

[2]Technical University Darmstadt, Germany
ahmad.sadeghi@trust.cased.de

ABSTRACT

Electronic health records (EHR) are a convenient method to exchange medical information of patients between different healthcare providers. In many countries privacy laws require to protect the confidentiality of these data records and let the patient control the access to them. Existing approaches to protect the privacy of EHRs are either insufficient for these strict laws or they are too restrictive in their usage. For example, smartcard-based encryption systems require the patient to be always present to authorize access to medical records. However, this does not allow a physician to access an EHR of a patient who is unable to show up in person.

In this paper, we propose a security architecture for EHR infrastructures that provides more flexibility but retains the security of patient-controlled encryption. In our proposal patients are able to authorize access to their records remotely (e.g. via phone) and time-independent for later processing by the physician. The security of our approach relies on modern cryptographic schemes and their incorporation into an EHR infrastructure. The adoption of our security architecture would allow to fulfill strict privacy laws while relaxing usage restrictions of existing security protections.

Categories and Subject Descriptors

J.3 [**Life and Medical Sciences**]: Medical Information Systems

General Terms

Security

Keywords

Electronic health records, cryptography, authorization

1. INTRODUCTION

Several countries plan to establish nation-wide or regional telematic infrastructures that include the storage and processing of medical data of patients in electronic health records (EHR), for example Austria [11], France [2], Germany [6], or Taiwan [9]. The EHRs are created, maintained, and managed by health care providers, and can be shared with other health professionals even of other health institutions.

In many European countries, especially in Germany, it is required by law to protect patients' privacy meaning that only patients themselves control who have access to their health data [1], and no one is allowed to circumvent these privacy policy. E-health systems that store patients data have to provide technical methods to support this [7].

To address security and privacy of EHRs, most approaches employ cryptographic techniques and access control based on smartcards ("chip & pin"). The smartcards are typically used to (i) authenticate health professionals and patients, (ii) sign EHR documents to provide authenticity, (iii) encrypt the EHR data before they are stored on the server and (iv) authorize the access to EHR data. Examples of these approaches are the German electronic Health Card [6], or the Taiwan Electronic Medical Record Template (TMT) [9].

Patient-controlled encryption provides the strongest security and privacy as the encryption keys are stored on the patient's smartcard. However, practical experiences from our interdisciplinary research projects show that this approach has several drawbacks regarding usability and acceptance:

- If security mechanisms do not support existing workflows of health professionals, acceptance problems might arise, in the worst case treatment is not possible.
- End-to-end encryption between health professionals is needed, but the destination is not known at encryption time. This disqualifies standard public-key encryption.
- Medical findings and diagnoses are often entered into the EHR system when the patient has already left the healthcare practice. A system that requires authorization onsite is not possible in this case.
- The patient is also not present at preparation or wrap-up time of a home visit by the doctor. Resulting data cannot be entered into an EHR for the same reason.
- Elderly and disabled people have problems remembering the PIN or using technical equipment. Hence, solutions based on chip & pin disqualify for these people.
- If a patient is too ill, a representative might go to a doctor or pharmacy. But as in some EHR projects the smartcard is also used as a means of identification, this representative should not know its PIN. Giving card and PIN to someone else might be even illegal.
- In emergency cases, patient might be unconscious or otherwise not able to authorize access to the EHR.

All these problems show that there is a conflict of goals be-

http://doi.acm.org/10.1145/2110363.2110448

tween secure patient-controlled authorization for accessing EHR and usable technical security mechanisms which are uncomplicated to use by the patients. Another problem is that the smartcard has to be connected to a local device of the health professional in existing approaches. So an authorization via Internet is not possible. For instance, in the system of the German e-health card [6] the smartcard has to be connected to an encryption/decryption device at the doctor's practice where the patient has to enter the PIN.

In this paper, we aim to solve this conflict and propose a security architecture for EHR infrastructures that provides more flexibility but retains the security of patient-controlled encryption. In particular, our proposed architecture allows patients to give an authorization secret to doctors via different communication channels (e.g. phone or as a paper code). This token allows a doctor to access the patient's EHR data while the patient does not need to be present at the time of access as he does not need to enter a PIN for authorization.

This approach provides more flexibility and retains the security and privacy properties of patient-controlled EHR encryption. We believe this flexibility will make such an EHR infrastructure more likely to be accepted by end-users.

2. PROTECTION OF HEALTH RECORDS

In this section, we briefly introduce smartcard-based encryption of EHR data, which is common to many proposals for (national) telematic infrastructures (e.g., [6, 9]).

2.1 Current Proposals

Smartcard Encryption.

Using a patient-owned smartcard (SC) for encryption and decryption of electronic health records usually follows a specific workflow: During a patient's visit in a doctor's practice, the health professional composes medical data which is to be encrypted by the patient's public key stored on his smartcard chip. Therefore the patient uses his or her smartcard and PIN for authentication and authorization of this write access to his or her electronic health record. The same process is also necessary for decrypting the medical data. Another health professional downloads the data and is only able to decrypt it if the patient uses his chip & PIN to authorize access to his EHR (see Figure 1). Moreover, there may be an additional access control layer which checks for every access to the encrypted data if the accessing party is granted read or write access to this EHR.

Resulting Challenge.

Authorization via smartcard (chip & PIN) results in a practical problem: Health professionals cannot read from or write to a patient's EHR while the patient is not present. In practice, health professionals frequently need access to EHRs while the patient is absent – e.g., to furnish a medical opinion or to report results of a medical checkup. A flexible solution – where patients can authorize access to EHR data during their absence – could improve the acceptance of the system by patients and health professionals which is essential for the success of any e-health infrastructure. Providing user-controlled encryption and authorization in the user's absence is a cryptographic challenge. Moreover, the destination of encrypted EHRs is not known at the time of encryption, which precludes using standard public key encryption and requires a flexible security architecture.

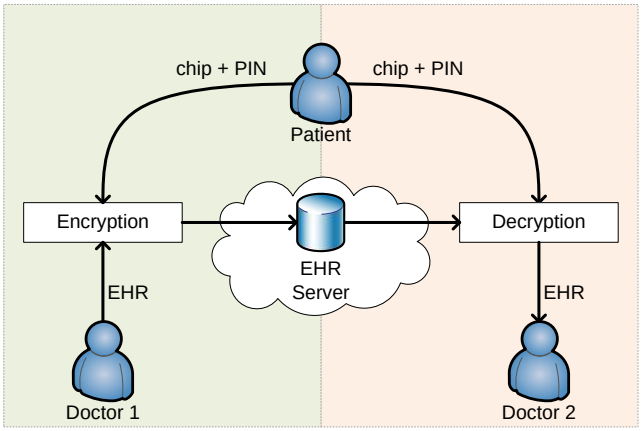

Figure 1: Common previous proposal: encryption and decryption with chip & pin.

2.2 Desired System Properties

Objectives.

Our target is to achieve a more flexible EHR infrastructure, while retaining the patient-controlled security and privacy of existing EHR systems.

Patients should be able to authorize a health professional to access their EHRs via an *asynchronous flexible* channel, i.e., via phone or by handing over a token. A health professional should be able to compose, encrypt and store medical data in a patient's EHR after the patient already left the location (cf. Figure 2). To this end, patients should be able to create and hand out an authorization secret which is used by the health professional for encryption/decryption later on.

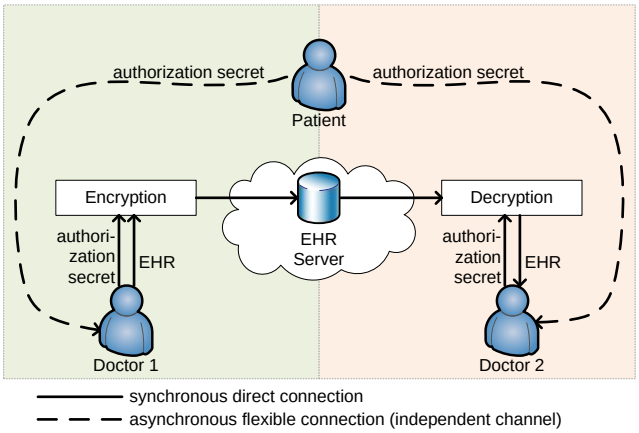

Figure 2: Our proposal: time-independent encryption and decryption with transactioncodes

This leads to the following principal objectives:

O1 **Patient-controlled *confidentiality* of EHR data**: EHR data must be encrypted and patients must be in full control of the encryption keys. Access to the EHR data must not be possible without authorization by the patient.

O2 ***Flexible authorization* of access to EHR data**: Patients must be able to authorize access to EHRs in

a flexible way, i.e., via different communication channels (e.g., phone, paper-based authorization), without physical presence, asynchronously for later use, and selectively for individual records.

O3 ***Emergency access***:
Emergency data must be accessible to physicians without prior authorization. Each emergency access must be logged / audited by the infrastructure so that its legitimacy can be checked after the fact ("break-the-glass principle").

Requirements.

An EHR system must meet the following requirements:

R1 ***End-to-end encryption***:
To achieve confidentiality (objective O1), EHRs must be stored and transmitted encrypted, without the keys being known to storage providers and communication infrastructure providers. Patients need to have individual secrets (different for each patient) for encryption of EHR data.

R2 ***Record-dependent encryption***:
For patients to be able to grant selective access to individual EHRs (objective O2), they must be able to create a secret, which can be used for encryption and decryption of a *specific* record. This implies that different secrets must be used to encrypt different EHRs.

R3 ***Transferability* of authorization secrets**:
The authorization secret should be transferrable via different channels to authorize encryption and decryption of EHR data by other parties, such as physicians, patient representatives, etc. (objective O2). This implies that such secrets must be manageable for a "normal person" to read and understand[1]. Yet they must be secure enough to preclude brute-force attacks.

R4 ***Asynchronous authorization***:
Patients must be able to authorize *later* access to EHRs (objective O2). Note that this is not the same as decrypting the EHR immediately and keeping a copy, because (i) other parties could modify the EHR between authorization and intended decryption time, and (ii) authorized parties must also be able to store EHRs.

R5 ***Access to emergency data***:
It must be possible to encrypt medical data specifically as emergency data, which can be read by any authorized physician or clinic, without the need of prior authorization by the patient (objective O3).

R6 ***Accountability of emergency access***:
Each emergency access must be logged and audited, such that the parties accessing emergency data can be held responsible (objective O3).

3. OUR SECURITY ARCHITECTURE

The key idea of our approach is to avoid the use of smartcards as a direct input for encrypting and decrypting EHRs. Before medical data is to be stored on an EHR server, the patient provides his smartcard only to generate a transaction code (TAC) which will be used as authorization secret. The encryption key is only based on the TAC and the patient's identity. When the EHR is to be read again, the patient gives the TAC to the health professional who needs to access the EHR. The novelty in this approach is that patients do not need to be present with their smartcards for decryption, but can provide the TAC via, e.g., phone.

For realization, we use attribute-based encryption (ABE), an asymmetric encryption scheme [5, 10]. ABE allows data to be encrypted by specific attributes and be decrypted if the decrypting user matches these attributes [3]. In ABE, a trusted third party is necessary for decryption: The private key generator (PKG) creates individual secret keys corresponding to the attributes, in our case the patient's identity and a TAC. The PKG also generates a master key M for each user during the setup phase. To generate private keys, the PKG uses a function called *extract*. Encryption is denoted by the function *enc*; decryption by the function *dec*.

Using ABE in e-health scenarios is already proposed by Narayan et al. [8]. We extend this approach with an explicit authorization parameter as an attribute to ensure that EHR accesses are actively authorized by the patient. Active authorization is required, because the PKG should not be able to generate the private key for an EHR without the patient's active involvement. Hence, we use two attributes for the ABE scheme: (i) the patient's identity (PID) – probably represented by his insurance number – and (ii) a record-specific transaction code (TAC) – which is created by a trusted third party (TAC service). Thus, our architecture includes a TAC service and a PKG (see Figure 3). The TAC service generates transaction codes using a function $genTAC$ on the basis of the patient's chipcard resp. its secret key $chipcard_{SK}$ and another input, e.g., a timestamp t.

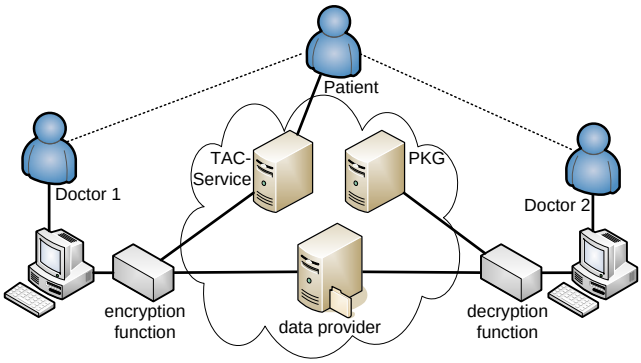

Figure 3: system architecture

Encryption.

As shown in figure 4, a patient first generate a TAC using their smartcard. Handing out the TAC to health professional 1 is an act of authorization, as now the health professional needs to enter an TAC to the encrypt medical data. The Encryption/Decryption function (EF/DF)[2] generates the public key for encryption as a hash value of the patient's identity (PID) and an TAC. Due to this procedure, there is

[1]Note that – in contrast to passwords – it is *not* necessary that authorization secrets can be easily *remembered*.

[2]The EF/DF can be either realized as software running on the doctor's computer (cf. [2]) or as a special hardware device as part of the EHR infrastructure (cf. [6]).

a specific TAC for every EHR which results in specific public keys for every entry. The encrypted EHR is sent to an external data provider to store it persistently. To avoid reuse of TAC the validity of a transaction code has to be checked by the TAC service during the encryption procedure.

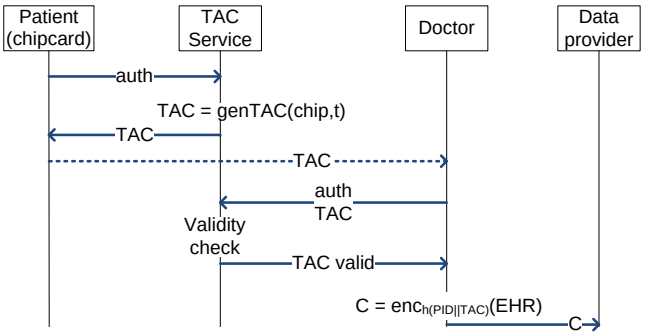

Figure 4: encryption protocol

1. The patient authenticates to the TAC service by chip & pin and creates a valid TAC.
2. A TAC is generated as: $TAC = genTAC(chipcard_{SK}, timestamp)$
3. The TAC service sends the code to the patient.
4. The patient passes the TAC to the health professional 1 – e.g. personally or via telephone.
5. Health professional 1 creates an EHR using his primary system (PVS) and enters the received TAC after authenticating to the EF.
6. The EF sends the TAC to the TAC service, which checks its validity.
7. If the TAC's validity is verified, the EF encrypts the EHR using a public key based on the attributes as $K_U^{pub} = h(P_{ID}||TAC)$, where h is a cryptographic hash-function, P_{ID} the patient's identity and TAC the record-specific transaction code. The ciphertext is created as $C = enc_{K_U^{pub}}(EHR)$.
8. Health professional 1 sends C to a storage provider.

Decryption.

If another health professional wants to decrypt the EHR to read the medical data, he first needs to obtain the corresponding TAC. Hence the patient is able to authorize health professional 2 by transferring the corresponding transaction code. Health professional 2 then authenticates to the private key generator (PKG) using his personal smartcard and PIN. Then he transfers PID and TAC to the PKG, which will generate a corresponding private key as a result of the ABE scheme's *extract*-method. This key is then used by the DF to decrypt the EHR. Health professional 2 can read the EHR only if he knows the corresponding TAC (see Figure 5).

1. Health professional 2 receives the transaction code (TAC) corresponding to the relevant EHR.
2. Health professional 2 authenticates to the PKG and transfers the patient's identity P_{ID} and TAC.
3. The PKG generates a private user key with attributes P_{ID}, TAC: $K_U^{priv} = extract(M, K_U^{pub}, TAC)$,
4. Health professional 2 obtains the ciphertext C and decrypts it as $EHR = dec_{K_U^{priv}}(C)$.

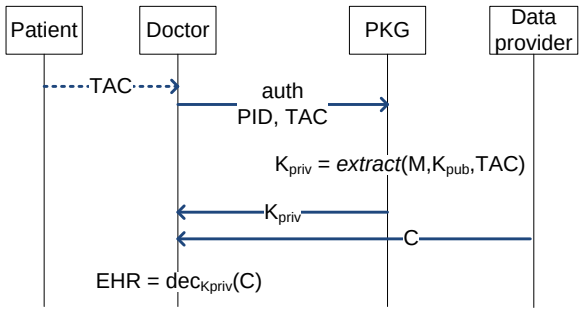

Figure 5: decryption protocol

Emergency Records.

In addition to usual EHRs, our architecture is able to handle emergency EHRs. These contain relevant data for medical emergencies, such as chronic diseases. To provide this feature, emergency EHRs have to be encrypted in a different way: Instead of using an EHR-specific TAC as an attribute, these EHRs are tagged as "emergency".

Hence, steps 1-6 of the encryption protocol are not necessary for emergency EHRs. The public key for encryption is based on the patient's identity and the tag "emergency": $K_{U,E}^{pub} = h(P_{ID}||\text{"emergency"})$ (Step 7). The EHR is then encrypted as usual (Step 8): $C = enc_{K_U^{pub}}(EHR)$.

The PKG accepts the patient's identity and the tag "emergency" instead of a TAC for decryption. In Step 3 of the decryption protocol, the PKG then generates an emergency private key $K_{U,E}^{priv} = extract(M, K_U^{pub}, \text{"emergency"})$ which enables the decryption of all emergency EHRs (Step 4): $EHR = dec_{K_U^{priv}}(C)$. Using the tag "emergency" instead of a TAC allows doctors to access the relevant data directly.

The PKG logs all emergency requests to enable audit and accountability. It can also send an automatic alarm to an appropriate authority in order to ensure that no accesses to emergency data go undetected. Authorization has to happen retrospectively in such cases, as for a medical assistance in emergencies, the patient's authorization is implied.

4. DISCUSSION

Workflow & Encryption. Implementing ABE and TAC as two components expands the usual security architecture for electronic health record systems. Whenever the doctor needs access to an EHR, the patient discloses the corresponding transaction code, for instance via telephone. This is an active authorization and allows the doctor to decrypt the medical data while the patient does not need to provide his chip & PIN or to be physically present.

To implement our proposal, only few changes to the existing infrastructure are necessary. The workflow of chip & PIN authorization can even be simulated completely. A patient is still able to authorize access to his EHR while he is at a doctor's office using common chip & PIN procedures. If a new EHR should be encrypted the patient provides his smartcard and PIN for encryption. A TAC is automatically generated and assigned to the new EHR. As the authorization has already taken place by entering the PIN, there is no need to enter the TAC manually.

Our security architecture provides end-to-end encryption (thus fulfilling requirement R1) using individual secrets for every EHR (therefore fulfilling requirement R2). These in-

dividual secrets are represented by transaction codes which are necessary for decrypting medical data.

Authorization in Absence. Handing out the TAC to a health professional authorizes access to the EHR. However, it is not necessary for the patient to be present for authorization: The TAC can be passed via phone, paper, or left at the doctor's practice for later use. Hence, our security architecture fulfills requirements R3 (transferability of authorization secrets) and R4 (asynchronous authorization).

Scalability. As the length and complexity (e.g., characters that can be used) of the TACs is scalable, the combination of ABE and TAC provides a high level of flexibility. So different EHRs can be categorized into security levels with TACs of different complexity. It is possible to use additional attributes for highly confidential EHR.

Trust Issues. Our approach relies on two partially trusted parties: the TAC generator and the private key generator PKG. As these components are essential for enforcing the confidentiality of medical records, they are assumed to work correctly and need to be trusted. However, the PKG can be implemented by an external service provider because it cannot decrypt EHRs without the corresponding TACs. Since the TAC generator can be under the control of the patient, the overall scheme is patient-controlled as required.

Structure of data elements. A limitation to our approach is determining the interweaving of EHRs. It is the health professional's task to define the structure of an EHR and its linkability as the structure of data elements must be adjustable at runtime so that EHR data elements may be merged or split for a medical treatment.

Parameterized Flexibility. The flexibility of our architecture can be parameterized by the number of PKGs for the ABE scheme rangeing from only one (centralized service), to one for each patient (on each smartcard itself), or anything in between. Therefore, chip&pin-based EHR infrastructures can be implemented as a special case of our approach, where the PKG resides in the smartcard of each patient.

Emergency Access and "Break-the-Glass". Specific EHRs can be encrypted as emergency records using the attribute "emergency", such that no record-specific TAC is required for decryption (cf. Section 3). Hence, requirement R5 (access to emergency data) is fulfilled. Logging and audit facilities of the EHR infrastructure must keep track of emergency accesses for retroactive authorization (see R6).

5. RELATED WORK

Benaloh et al. [4] point out the need of encryption of medical data in addition to access control. In their approaches, patients generate and store encryption keys on their own. While this lets patients control the encryption of their medical data, the keys are not disclosed. Hence, patients still have to be present to authorize access to EHRs.

Akinyele et al. [3] introduce a more flexible approach: They use attribute-based encryption to provide a secure encryption of electronic medical health records. Their focus is concentrated on availability of medical data and therefore especially on Personal Health Records (PHRs). In contrast to EHRs which are administered by health care professionals, PHRs are administered by the patients themselves. PHRs make no claim to be a consistent collection of medical history and are meant to be used by the patients themselves. Therefore, an explicit authorization parameter is not required in PHR systems, but is essential for EHR systems.

Narayan et al. [8] show how to provide privacy in e-health systems with attribute-based encryption. However, their approach is based on a specific structure for health records and in their model patients administer their own health records as PHRs. The authors do not discuss transferrable authorization secrets, but let patients define encryption policies. In contrast to our approach, they assume that patients know whom to authorize when they create (or re-encrypt) EHRs.

6. CONCLUSION

In this paper, we propose a flexible security architecture for EHR infrastructures to address a drawback of existing proposals: Authorizing access to EHRs with smartcards (using chip & PIN) is only possible if the patient is present to enter the PIN. Our approach leverages attribute-based encryption and scalable authorization secrets for more flexibility in accessing EHRs, while providing full patient-controlled security and privacy. This enables authorization to access EHRs via telephone or even in advance while the encryption does not depend on the private key of the person who is to be authorized. Our solution can be implemented without much effort, as existing telematics and EHR infrastructure only needs to be extended by transaction code generators and public key generators. The major workflows remain the same, hence we expect a high acceptance among end-users, leading to a realistic chance for deployment in practice.

7. REFERENCES

[1] Sozialgesetzbuch V, §291a Elektronische Gesundheitskarte, July 2004. German Federal Law.

[2] Agence des systèmes d'information partagés de santé. Dossier de spécifications fonctionnelles et techniques des interfaces DMP des logiciels de professionnels de santé. Version 1.0.0, 2010. http://www.asipsante.fr.

[3] J. A. Akinyele, C. U. Lehmann, M. D. Green, M. W. Pagano, Z. N. J. Peterson, and A. D. Rubin. Self-protecting electronic medical records using attribute-based encryption. Cryptology ePrint Archive, Report 2010/565, 2010.

[4] J. Benaloh, M. Chase, E. Horvitz, and K. Lauter. Patient Controlled Encryption: Ensuring Privacy of Electronic Medical Records. In *The ACM Cloud Computing Security Workshop, CCSW '09*, pages 103–114. ACM, 2009.

[5] J. Bethencourt, A. Sahai, and B. Waters. Ciphertext-policy attribute-based encryption. In *Proceedings of the 2007 IEEE Symposium on Security and Privacy*, S&P '07, pages 321–334. IEEE Computer Society, 2007.

[6] Gematik - Gesellschaft für Telematikanwendungen der Gesundheitskarte. Die elektronische Gesundheitskarte. http://www.gematik.de.

[7] German Federal Ministry of Health. Entscheidungsvorlage - Festlegung der Authentisierungs-, Autorisierungs- und Auditmechanismen der Telematikinfrastruktur für die Fachanwendungen, Version 0.9.0, March 2006.

[8] S. Narayan, M. Gagné, and R. Safavi-Naini. Privacy preserving EHR system using attribute-based infrastructure. In *The ACM Cloud Computing Security Workshop, CCSW '10*, pages 47–52. ACM, 2010.

[9] H.-H. Rau, C.-Y. Hsu, Y.-L. Lee, W. Chen, and W.-S. Jian. Developing electronic health records in Taiwan. *IT Professional*, 12:17–25, 2010.

[10] A. Sahai and B. Waters. Fuzzy identity-based encryption. In *Advances in Cryptology - EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 457 – 473. Springer, 2005.

[11] SVC - Sozialversicherungs-Chipkarten Betriebs- und Errichtungsgesellschaft. e-card. Website. http://www.chipkarte.at.