Detlef Hühnlein et al. (Eds.): Open Identity Summit 2015,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2015 71

Using Proxy Re-Encryption for Secure Data Management in an Ambient Assisted Living Application

Hannes Zach[1,2] Philip Peinsold[2] Johannes Winter[3] Peter Danner[2,3] Jakob Hatzl[2]

Abstract: Whenever applications process sensitive user data, secure storage and distribution plays a key role. This paper points out the security demands of an Ambient Assisted Living (AAL) application and demonstrates the usage of proxy re-encryption in order to fulfil its security requirements for storage and distribution of sensitive data. Because AAL systems often exhibit the same security needs as the application developed in the presented project, the described implementation can serve as a point of reference for similar projects.

Keywords: IT Security, Proxy re-encryption, Ambient Assisted Living, secure data storage and distribution

1 Introduction

In the near future, we will have to face a dramatic change in the age structure of our population. Society gets older, while at the same time birth rates are decreasing. As a result of the aging population, the amount of care-dependent people is rising constantly while there are less people to provide care. Ambient Assisted Living (AAL) enables elderly people to live a more convenient and self-determined life, and counteracts the increasing demand for care and an approaching financial crisis due to the increasing costs for professional care. Most AAL systems have to process personal sensitive health data. Hence, careful consideration of security and privacy concerns is required and protection of this data should have highest priority.

DALIA (**DA**ily **LI**fe **A**ctivities at Home) is a research project co-funded by the European AAL joint programme. DALIA works on an integrated home system to support older adults with their daily life activities. One key goal is to provide easy to use, privacy and security aware mechanisms to exchange sensitive data, for example medication data, between older adults and their carers. This paper discusses a work-in-progress approach being developed in DALIA to bring security to an AAL environment.

In order to guarantee data protection, several protective goals have been established and include, among others, confidentiality, integrity and availability [ST07], which has to be

[1] FH Joanneum - University of Applied Sciences, Department of Applied Computer Sciences, Werk-VI-Strasse 46, 8605 Kapfenberg, Austria
[2] exthex GmbH – explore the excellence, Göstinger Straße 213, 8051 Graz, Austria
[3] Graz University of Technology, Institute for Applied Information Processing and Communications (IAIK), Inffeldgasse 16a, 8010 Graz, Austria

guaranteed by the provider of an AAL system. In order to fulfil these needs, many factors have to be considered, especially security and privacy, not only regarding the monitoring of AAL users in their home environment [Mo07] but also in terms of data security.

AAL systems often connect elderly people with their carers. To ensure proper protection of the user's data it is crucial that secure AAL systems employ adequate cryptographic measures to secure network connections and to provide secure data storage capabilities. Transport security can be addressed by using appropriate transport layer protocols, such as TLS/SSL, to realize secure channels that guarantee data-confidentiality, data-integrity and origin-integrity properties between any two network-connected components of an AAL system. Within the scope of this paper, we assume that transport layer security can be solved adequately with secure channels. Secure channels only address security requirements related to *sensitive data in transit* between any two AAL system components. They do not provide any help with storing and sharing data securely.

Secure data storage capabilities are essential for AAL system components to ensure that confidentiality and integrity of sensitive user data is maintained at all times. Encryption of sensitive data stored on servers is one feasible approach to reduce the problem of protecting huge amounts of sensitive data. If proper algorithms and key sizes are used, encryption makes it virtually impossible for an attacker to break confidentiality of *sensitive data at rest*, without knowing the encryption keys.

Secure data sharing capabilities are required to facilitate data exchange between older adults and their carers. Within an AAL system, the data exchange is commonly done indirectly over a cloud-like infrastructure provided by the AAL provider. To protect the privacy and informational self-determination of the end-user, it is essential to have secure data sharing capabilities in the AAL system, which allows them to control sharing of sensitive data.

In order to enable users to store and synchronize their sensitive data in a secure way, it is the providers responsibility to implement an appropriate solution. A relatively new approach to realize a secure data storage is proxy re-encryption. Proxy re-encryption schemes are cryptographic schemes, which allow transformation of ciphertexts, encrypted with one secret key, to ciphertexts that can be decrypted with a different secret key. The ciphertext transformation requires a re-encryption key, which can (only) be derived by the owner of the original secret key.

This paper contains six major sections: Section 1 discusses the motivation for this work and introduces the overall security requirements in the area of AAL. Afterwards, section 2 briefly describes related work in the area of proxy re-encryption, on which this paper builds on. Next, the DALIA security concept and its underlying principles are introduced in section 3. Section 4 contains the main contributions of this paper, which consist of the implementation of the DALIA security framework and its proxy re-encryption library. Section 5 discusses critical aspects of the described implementation. Finally, Section 6 concludes the paper by summing up the advantages of proxy re-encryption in the presented scenario.

2 Related work and our contribution

The first concept of proxy re-encryption was introduced by Mambo and Okamoto in 1997 [MO97]. The authors described a cryptosystem that allows an original decryptor to transform its ciphertext into a ciphertext for a designated proxy decryptor, which is then able to compute a plaintext in place of the original decryptor.

In 1998, Blaze et al. [Bl98] described atomic proxy re-encryption as the currently used scheme for proxy re-encryption. A proxy is able to re-encrypt a ciphertext, produced by the public key of Alice into a ciphertext, Bob is able to decrypt with his own secret key and without actually knowing the secret key of Alice. Therefore, Alice has to create a re-encryption key, which consists of her private key and Bob's public key.

With this basic concept of proxy re-encryption, several possible applications have been explored. Kallahalla et al. [Ka03] examined re-encryption methods for realizing secure file sharing on an untrusted storage. This idea is carried on in DALIA, which is also used as trusted storage for its users, but is, for security reasons, treated as an untrusted entity anyway.

Chow et al. [Sh10] as well as Green and Ateniese [GA07] set the background for realizing a secure distributed storage with proxy re-encryption by examining unidirectional proxy re-encryption. In this concept, which is also applied in DALIA, the data owner does not have to share the private key with the proxy in order to make the data accessible to another user.

Meingast et al. [Me06] studied security risks in healthcare applications and defined relevant questions and requirements that have to be considered by data holders in order to guarantee appropriate data protection in healthcare settings. The key aspects of AAL, that were identified as security relevant, include data ownership, data storage, and data access.

The primary contribution of this paper is twofold: First, we discuss security issues in the area of AAL and show how DALIA addresses these issues. As second part of our contribution, we demonstrate how proxy re-encryption can be used to construct an AAL system that provides a good balance between security requirements, complexity of deployment and usability. Our system architecture enables the user to exercise full control over data sharing, while reducing the computational effort required at the users device to a minimum.

3 DALIA's security concept

One of the key objectives of the DALIA project is to develop an integrated home system supporting older adults as primary end-users in their daily life. To achieve this goal, DALIA incorporates a data storage and distribution framework that allows carers, as secondary end-users of the system, to securely access data produced by the older adults as primary end-users.

The primary DALIA end-users possess smartphones and smart TVs through which they are connected with their relatives or carers. Elderly people can easily stay in touch with their carers, who benefit from a simplified caring process. With DALIA, it is for example possible to automatically disseminate information about changes in the prescriptions of medication for older adults to their carers and relatives. This feature is only possible if all involved participants are able to share sensitive data with each other, in a manner that protects the confidentiality and integrity of the data and the privacy interests of its users.

In DALIA, an older adult can share data or groups of data – called *modules* – like agenda items or emergency contacts, but also sensitive health data like their medications with carers. Sharing data is not unique to DALIA. AAL systems often base on several users who are connected with each other allowing them to share sensitive health data. This is the reason why DALIA relies on proxy re-encryption to make dynamic sharing of data possible. However, unique to DALIA is, that high server-side security mechanisms guarantee for the safety of user data while at the same time ensuring strong cryptographically enforcement of user control over the data distribution. While DALIA enables users to precisely specify where their data may go, it does not demand an unreasonably high level of trust in the intermediate components responsible for the actual data transfer. The remainder of this section discusses the roles and data-flows within DALIA and the usage of proxy re-encryption.

3.1 Roles within the DALIA security framework

The three key roles in DALIA are data subjects, data holders, and third parties with access to the actual data. Data subjects are the producers of sensitive data, respectively the persons who are cared for, also called the “older adult”. The original data subject has full control over what happens to the data. Data holders are professional entities who take care of intermediate and long-term storage and processing of data and should not know the real content of the data. In particular, data holders must not be able to share sensitive data with any unauthorized third parties that were not explicitly approved by the data subject. However, it is possible for data subjects to allow data holders to perform a restricted form of analysis or processing of the data.

The third key role are trusted third parties, typically professional and informal carers or medical services, whom the data subjects trust and explicitly grant access to the data. Those trusted third parties should be able to acquire the relevant data for which they have proper authorization from data holders.

Practical realization of this setting depends on suitable cryptographic methods that allow expressions of trust relationships between the stakeholders by cryptographic means. Therefore, we identified unidirectional proxy re-encryption as such method to realize the DALIA security framework.

3.2 Unidirectional proxy re-encryption

With a re-encryption key, DALIA is able to make data, which is originally encrypted for the older adult, accessible to a carer. In addition, with proxy re-encryption it is possible to share the same ciphertext created by the older adult with different individuals only through re-encrypting the ciphertext with different re-encryption keys.

Proxy re-encryption can be implemented via unidirectional and bidirectional schemes. DALIA uses unidirectional proxy re-encryption because one older adult can have multiple carers to whom the data has to be shared. Bidirectional proxy re-encryption schemes would additionally allow the proxy to re-encrypt the carer's ciphertexts for the older adult. The following figure shows the concept of proxy re-encryption in DALIA.

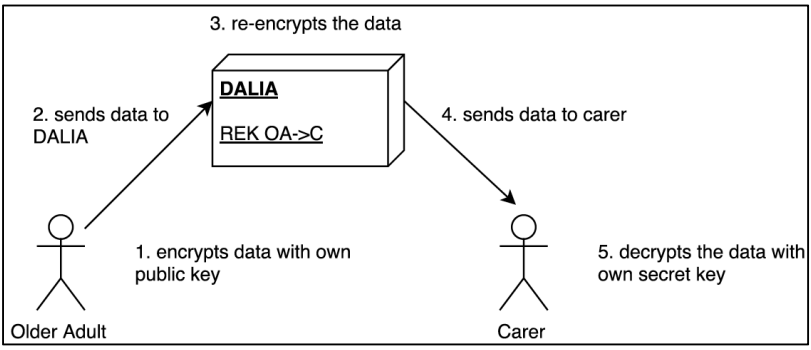

Fig. 1: Proxy re-encryption concept

DALIA uses the re-encryption key derived from the older adults private key and the carers public key (REK AO→C) in order to make the data of the older adult accessible to the carer.

3.3 Encryption and storage of data via proxy re-encryption

In DALIA, the older adult can share several modules with a specific carer. In order to illustrate the proxy re-encryption process in DALIA, the following example explains a typical scenario:

- The older adult creates a new record (e.g. a new medication to be taken).
- The older adult's device creates a key to encrypt the record symmetrically.
- The symmetric key is encrypted with the older adult's public key.
- The encrypted record is put on the DALIA server together with the encrypted symmetric key.

The client encrypts the data symmetrically because of performance reasons. In some cases,

the data could contain binary content like pictures or videos. As a result, the data exhibits a large size and would take much longer to encrypt, decrypt and re-encrypt asymmetrically. The re-encryption of a relatively small key, used for symmetric encryption and decryption, is much more performant than the re-encryption of the actual data.

3.4 Data sharing

An older adult can decide to share a specific module with a carer. To continue the above-mentioned example, an older adult could decide to share all medications with a specific carer and therefore grants access to the medication module. DALIA then allows the carer to receive all medications in an encrypted way. Initially, if a user wants to share data with a carer, the client of the older adult generates a re-encryption key. Only with this re-encryption key, the client of the carer can decrypt the encrypted symmetric key to decrypt the data.

The following picture visualizes the process of sharing data of a specific module of an older adult with a carer.

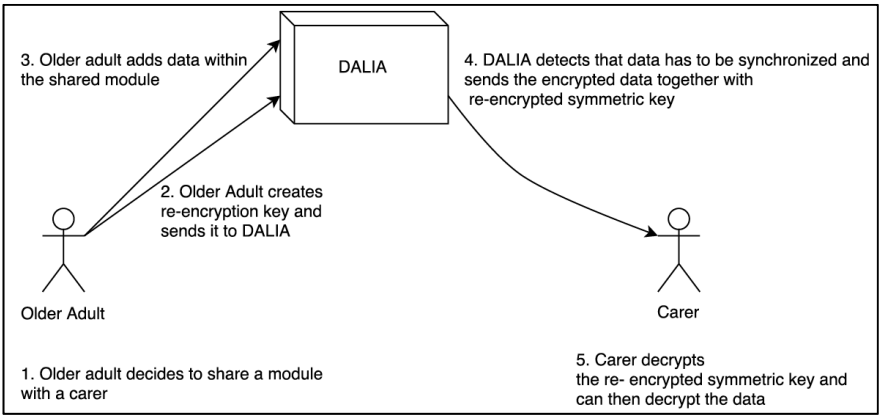

Fig. 2: Encrypted data transfer

Access revocation

DALIA maintains a list that includes a reference between the older adult's modules and the information with whom they are shared.

If an older adult decides that a specific module should no longer be shared with a carer, this reference is simply deleted, which causes DALIA to remove all related data from the carers device. Furthermore, no data is synchronized any longer to the carer.

4 DALIA's security framework

4.1 Proxy re-encryption implementation

In order to generate a proxy re-encryption key, an authentic copy of the carer's public key and the older adult's private key is needed. In the DALIA setting, the owner of the private key (the older adult) always derives proxy re-encryption keys. Therefore, authenticity of private keys is guaranteed implicitly. In order to guarantee that a given public key really belongs to the intended recipient (the carer) explicit authentication is needed. This setting equals other asymmetric encryption schemes, such as RSA, and the solutions to the public key-distribution problem (e.g. PKI).

Proxy Re-Encryption Java Library (reproxy)

We have implemented a standalone Java library (reproxy) to provide the cryptographic primitives for proxy re-encryption in the prototype of the DALIA security framework. This library is designed to abstract the specifics of unidirectional proxy re-encryption schemes behind a scheme-independent simple application programming interface (API).

The reference implementation of our library implements a variant of the unidirectional proxy re-encryption found in Ateniese et al. [At06]. This scheme uses an elliptic curve variant of the El-Gamal encryption scheme in combination with bilinear maps. For elliptic curve and pairing support, our reference implementation uses the open-source jPBC library[4]. No additional external dependencies (apart from the Java Runtime Environment) are used.

The scheme-independent API is the primary interface to the proxy re-encryption library. It contains Java interfaces modelling domain parameters, proxy re-encryption schemes, and the different types of keys (public key, private key and proxy re-encryption key). Operations like key generation, encryption, re-encryption and decryption are modelled as methods on these interfaces.

One challenge that is addressed in the scheme-independent API are the differences between encoding of plaintexts and ciphertexts for different schemes. Depending on the internals of a particular proxy re-encryption scheme the actual message may be mapped to different mathematical objects like integers in some group, points on an elliptic curve, or elements of some finite (extension) field. Even for the same type of mathematical object (like an elliptic curve point) there is often more than one possible way of mapping an arbitrary plaintext byte array to a point. To deal with these differences our scheme-independent API encapsulates plaintext and ciphertext messages that are encoded for a particular proxy re-encryption scheme as Java objects of special type. Encoding and decoding facilities that allow mapping between arbitrary Java byte arrays and these objects are provided as part of the generic proxy re-encryption scheme interface. This design enables library users to work with the library, without noticing the low-level details of the

[4] http://gas.dia.unisa.it/projects/jpbc/

proxy re-encryption scheme in use.

The message encoding facilities of the library also partly address the issue of message length limitations, for example due to the bit-length of moduli. When instantiating a proxy re-encryption scheme, the library user can select a message "codec". That defines how message byte arrays are encapsulated and padded. The reference implementation of our library currently implements four different codec types:

The **"raw"** codec literally maps the message byte array to an unsigned integer value without any further padding. The maximum message length is subject to bit-length limitations (e.g. moduli, group orders) of the scheme, and cannot (automatically) recover the length of the message byte array on decryption.

The **"simple"** and **"oaep"** codecs are based on the PKCS#1 specified padding schemes for RSA encryption. Both of these codecs pad the message byte array, before mapping them to an unsigned integer. The maximum message length is subject to bit-length limitations (e.g. moduli, group orders) of the scheme. For both codecs the padding method allows unambiguous recovery of the length of the message byte array. Decrypted messages will have exactly the same length as the original plaintext byte array used during encryption.

All of the three codecs discussed so far are only suitable for encoding messages, which are shorter than a size limit implied by the choice of domain parameters. To overcome these limitations, our library provides an **"ephemeral"** codec implementing a simple key wrapping scheme: The **"ephemeral"** codec generates a random (ephemeral) key that is encrypted using the proxy re-encryption scheme. The actual message byte array provided by the user is symmetrically encrypted using the ephemeral key. This codec does not have any message length limitations, since the actual payload is encrypted with a normal block cipher. Message integrity can be provided by using a block cipher that supports authenticated encryption (AE), such as AES-GCM.

4.2 Key management

In order to guarantee confidentiality of the stored sensitive data via our proxy re-encryption concept, all participants need to possess a key pair, consisting of a secret key and a public key. Therefore, it is important to define a key distribution concept, which ensures the correct key management in DALIA. The following section describes our approach in more detail.

An older adult usually owns two devices: a smartphone and a smart TV. Both devices have to encrypt their sensitive data via the same secret key, so it is necessary to distribute the keys between them. While there are several concepts available to distribute the asymmetric keys between a system and its users, DALIA generates the key pair for the user. This approach offers a maximum of convenience for DALIA users, while at the same time it demands a certain amount of trust in DALIA. Alternatively, it is possible to create the keys on the client-side or to include a third party into the whole process, as described in

the discussion in section 5. One of the most important aspects of the concept is the recovery password, which DALIA generates randomly. The secret key of a user is encrypted symmetrically with this password to enable DALIA to back up the secret key without having access to it. The user receives the recovery password via mail. With this recovery password, a user can retrieve the encrypted key pair from DALIA and decrypt it. If an older adult receives DALIA for the first time and sets up the devices, each client tries to receive its encrypted key pair. After the setup, the client can use the secret key to decrypt the data and to create re-encryption keys for data sharing.

The actual re-encryption is done on the cloud-server for performance and availability reasons. The client devices used in DALIA, like smartphones and smart TVs, have limited resources and the devices cannot always rely on an available internet connection. Therefore, shared data may not be available for authenticated receivers all the time. This is why the server holds the re-encryption keys and takes care of all re-encryption and data distribution tasks.

4.2.1 Key recovery and backup

If a user loses a device, for example the smartphone, there would normally be no way to obtain the encrypted data, which is stored on DALIA, back on a new smartphone. With the recovery password, the user can easily regain access to the encrypted data with a new smartphone.

The DALIA server has access to the database server, which stores all relevant information about the users like username, hash of the password as well as their encrypted secret key and public key but not the recovery password. After the key pair was initially created and encrypted with a randomly generated recovery password, the recovery password is send to the user but is not stored on the server. Having access to the recovery password would enable unauthorized persons to gain access to the secret key, which then gives access to the encrypted sensitive data. Therefore, only the user has access to the recovery password, which is distributed to the users via a QR-Code that is included in the welcome letter.

4.3 Practical Proxy re-encryption scenarios in DALIA

In DALIA, four different data exchange scenarios can occur. In the first scenario, an older adult creates data within a certain module, either via the smartphone or via the smart TV. This data has to be synchronized to the carer who has access to the module. In this case, DALIA re-encrypts the data in order to make it accessible for the carer. Figure 3 visualizes this case in scenario 1.

Another scenario would be a carer, entering data for a specific older adult. In this case, no re-encryption has to take place because the data created by the carer has a clear destination, which is a specific older adult. Therefore, the data is encrypted with the public key of the older adult and can be decrypted with the older adult's secret key (see scenario 2 in Figure 3). This scenario works similar, regardless whether the data is synchronized to the older

adult's smartphone or the smart TV.

In addition, data which is available on the older adults smart TV has also to be synchronized to the older adult's smartphone. Therefore the older adult's smart TV encrypts the data with the own public key and sends it to DALIA. The smartphone checks regularly whether there exists new data, which has to be synchronized and finds the new entry. DALIA recognizes, that the data comes from the older adult and has to be send to another device of the same older adult. Therefore, no re-encryption is necessary but the data is directly sent to the older adult's smartphone. The smartphone decrypts the data with the secret key and stores it. Figure 3 visualizes this case in scenario 3.

The same synchronization is necessary if a carer adds data on behalf of the older adult via the hosted service and then has to be synchronized to the carer's mobile devices. In this case, the carer encrypts the data with the older adult's public key and stores it on DALIA. In order to access the data on a different device of the carer, the data is again re-encrypted to make a decryption with the carer's secret key possible. Figure 3 visualizes this case in scenario 4.

In all scenarios, DALIA takes care of a possible necessary re-encryption which is always necessary if data of an older adult has to be synchronized to a carer or if data of carer's device 1 has to be synchronized to carer's device 2. The devices of both, the carer and the older adult, always receive the data in a way that it can be decrypted with their own secret key, which is done in the "crypto service". The crypto service is also responsible to encrypt outgoing data with the older adult's public key. Hence, DALIA always encrypts user data with the older adult's public key.

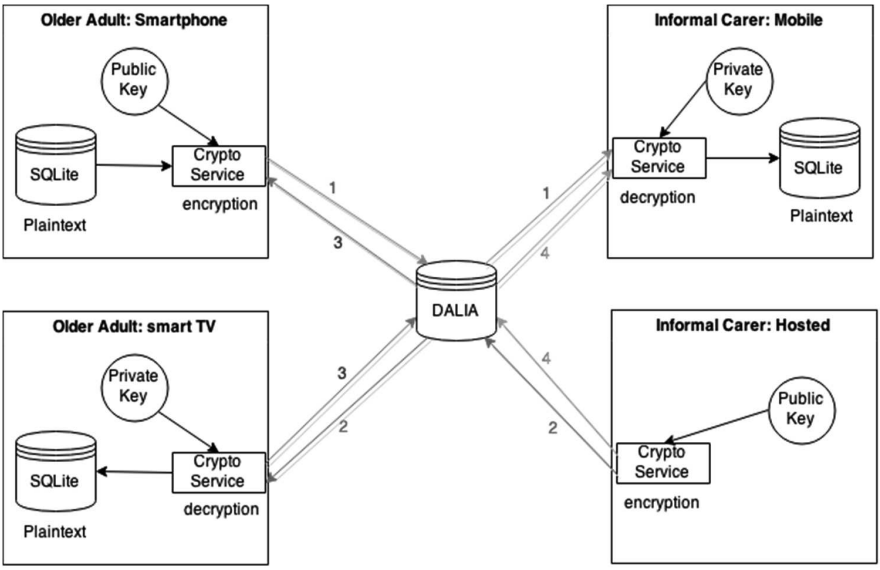

Fig. 3: Data exchange scenarios

5 Discussion

5.1 Alternative key management approaches

Usually, it is common to generate the key pair on the client device. In DALIA, we decided to generate the key pair on the server and to store it in an encrypted way. This approach offers a minimum of user interaction and at the same time ensures that the key pair can never be lost. If the client generates the key pair and the user does not correctly backup the secret key, the loss of the smartphone and hence the secret key has severe consequences. The data, which is stored on DALIA, is only accessible for the older adult with the correct secret key. Because DALIA never has access to the plain data, an unauthorized recovery would not be possible. Relying on the server in terms of key management requires a specific amount of trust in DALIA. This is why a third party should monitor and approve this process to guarantee trustworthiness of DALIA.

Another possibility would be the inclusion of a third, unbiased and trusted party (like a notary) into the key management process. This third party would undertake the task of creating, storing and sending the key pair to the users instead of DALIA. This approach combines both targets, namely usability and security as the user receives again a letter with a QR-Code, which contains the recovery password. The user then only has to scan the QR-Code in order to set up a new device. In terms of security, the whole concept

becomes more reliable because of the exclusion of DALIA from the critical process of key generation.

5.2 Granularity of shared modules

An important aspect of the whole DALIA security framework is the granularity of modules. If an older adult shares a specific module with a carer, all data, which falls into a specific module, is accessible for the carer. At the current state of DALIA, there is no need to make the sharing concept more fine-grained but in some cases, it makes sense to limit the access to the data even within a specific module. An example would be to share just a limited list of medications to a specific carer if the older adult wants to hide specific information like a specific medication, which could indicate a certain medical condition.

6 Conclusion

Because AAL applications often process sensitive health data, secure storage and data protection is essential. Protecting the stored data in an encrypted way and at the same time allowing users to share their data with selected individuals is a highly complex task. In this paper, we investigated the use of proxy re-encryption as one possible solution that can fulfil these requirements. With the DALIA security framework introduced in this paper, users can share specific data sets with selected individuals, while at the same time the data store never has access to the plaintext data. In addition, the presented solution allows users to revoke the granted access to the data easily. Furthermore, it is possible to create a re-encryption key at a later point in time, even for users, which did not exist at the time of encryption. This means, an older adult can store data in a secure way and can share it later with another user like a physician. This approach works because data is always encrypted in the same way, namely with the older adults public key. The server then performs the re-encryption if necessary to make the data accessible for different individuals. This proxy re-encryption approach in combination with the usage of transport layer security (TLS) in the background results in a strongly protected data storage and distribution system. All of the described proxy re-encryption features make the presented concept highly dynamical and easy to use because all of its complexity is hidden for the users, which makes it ideal for Ambient Assisted Living applications.

Acknowledgments: This work has been supported in part by the European Commission through the AAL Joint Programme under contract AAL-2012-5-249 DALIA.

References

[ST07] Sattarova Feruza Y. and Prof. Tao-hoon Kim. IT Security Review: Privacy, Protection, Access Control, Assurance and System Security. International Journal of Multimedia and Ubiquitous Engineering, Vol. 2, No. 2, April 2007.

[Mo07] Moncrieff, S., Venkatesh, S. and West, G. 2007. Privacy and the Access of Information in a Smart House Environment, in Wang, J.Z. et al (ed), Proceedings of the 9th ACM SIGMM International Workshop on Multimedia Information Retrieval (MIR 2007), Sep 24-29 2007, pp. 671-680. Augsburg, Germany: Association for Computing Machinery (ACM).

[MO97] M. Mambo and E. Okamoto. Proxy Cryptosystems: Delegation of the Power to Decrypt Ciphertexts. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, 80(1):54-63, 1997.

[Bl98] M. Blaze, G. Bleumer, and M. Strauss. Divertible protocols and atomic proxy cryptography. In K. Nyberg, editor, Advances in Cryptology - EUROCRYPT '98, International Conference on the Theory and Application of Cryptographic Techniques, volume 1403 of Lecture Notes in Computer Science, pages 127-144. Springer, 1998.

[Ka03] Mahesh Kallahalla, Erik Riedel, Ram Swaminathan, Qian Wang, and Kevin Fu. 2003. Plutus: Scalable Secure File Sharing on Untrusted Storage. In *Proceedings of the 2nd USENIX Conference on File and Storage Technologies* (FAST '03). USENIX Association, Berkeley, CA, USA, 29-42.

[Sh10] Sherman S. M. Chow, Jian Weng, Yanjiang Yang, and Robert H. Deng. Efficient Unidirectional Proxy Re-Encryption. In Progress in Cryptology -AFRICACRYPT 2010, volume 6055 of LNCS, pages 316–332. Springer, 2010.

[GA07] Matthew Green and Giuseppe Ateniese. Identity-Based Proxy Re-encryption. In ACNS 2007, volume 4521 of LNCS, pages 288–306. Springer, 2007.

[Me06] Meingast M, Roosta T, Sastry S. Security and privacy issues with health care information technology. Proceedings of the 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS '06); September 2006; pp. 5453–5458

[At06] Giuseppe Ateniese, Kevin Fu, Matthew Green, and Susan Hohenberger. Improved proxy re-encryption schemes with applications to secure distributed storage. *ACM Trans. Inf. Syst. Secur.* 9, 1 (February 2006), 1-30. DOI=10.1145/1127345.1127346 http://doi.acm.org/10.1145/1127345.1127346