Securing First Response Coordination With Dynamic Attribute-Based Encryption

Stefan G. Weber
Telecooperation Group, Department of Computer Science, Technische Universität Darmstadt
Darmstadt, Germany
sweber@tk.informatik.tu-darmstadt.de

Abstract

Every minute saved in emergency management processes can save additional lives of affected victims. Therefore, an effective coordination of the incident reactions of mobile first responders is very important, especially in the face of rapidly changing situations of large scale disasters. However, tactical communication and messaging between the headquarter and mobile first responders, initiated for coordination purposes, has to meet a strong security requirement: it must preserve confidentiality in order to prevent malicious third parties from disrupting the reactions. This paper presents concepts to support the secure coordination of mobile first responders by providing means for secure ubiquitous tactical communication. Our concept harnesses ciphertext-policy attribute-based encryption (CP-ABE) techniques. We extend current CP-ABE proposals by additionally taking into account dynamic factors: our proposed system is able to handle dynamic attributes, like current status of duty and location of mobile first responders, in a secure fashion. A flexible specification of receiver groups of tactical messages is possible, while end-to-end encryption in the messaging process is satisfied.

1. Introduction

Future first response support systems promise to improve management of large scale disasters by collecting and processing large amounts of information. Aggregated relevant information, alongside with a digital map, describing the current disaster situation, can effectively support emergency management staff members to decide on the next actions that have to be induced [1]. For example, a responsible decision maker in a head quarter might command mobile forces to evacuate citizens affected by a toxic gas cloud spreading out. In this task he needs to take care that none of the first responders is put into highest risk by the instructions given out. The decision maker may visually analyse the information he got, on the annotated digital map, in order to find an open and safe route. Consecutively, he needs a way to communicate the resulting orders to relevant first responders, out in the field. From a data security perspective, such instructions sent out are of highest sensitivity: they contain information that may be the key to the survival of victims of a disaster, may contain information about critical infrastructures, or hazardous materials. If such sensitive information is not properly secured, it may be subject to malicious tampering, manipulation or terroristic exploitation.

At this point, a failure of security has drastic consequences: inadequate handling of tactical information on the *data security* level may become a threat to *public security*. Actually, the lifes of first responders partly depend on correct information. Injured persons, that shall be immediately rescued, and physical assets the first responders are chosen to protect, may be affected by data security vulnerabilities, in the second line, as well.

Motivated by this practical setting, in this paper, we describe a concept how to actually secure the coordination of mobile first responders through a head quarter member. Especially, our approach facilliates a secure mobile attribute-based messaging system. It is based on ciphertext-policy attribute-based encryption (CP-ABE) primitives and allows to send out confidential tactical messages to dynamic groups of mobile receivers, which are implicitly adressed by composing policies of static and dynamic attributes.

In this paper, we advance the study of secure attribute-based systems, with special focus on providing confidentiality in mobile and dynamic settings. We share the opinion of various researchers [2]–[5] that such systems have an enormous potential to provide data security in distributed computing environments.

978-0-7695-3805-1/09 $26.00 © 2009 IEEE
DOI 10.1109/CONGRESS.2009.11

The remainder of this paper is structured as follows. Section 2 introduces to the field of first response support systems and details important workflows. Then, section 3 defines the concept of attribute-based messaging. An introduction to attribute-based encryption is presented in section 4. The details of our own approach, extending current CP-ABE techniques, are given in section 5. It is theoretically analyzed in section 7. Section 8 describes related work. Finally, the paper is concluded in section 9.

2. A Vision of Future First Response

First response support systems strive to improve emergency management work, to allow for a quick and coordinated response. A vision of future first response support systems is currently designed and developed in the SoKNOS project [6]. It conceptualizes the SoKNOS plattform, which follows an approach to integrate both service-oriented and event-based architecture styles. Its goal is to provide support for flexibly crafting adequate first response and incident reaction, in the face of rapidly changing situations in cases of large scale national-wide emergencies and natural catastrophes.

Allowing to integrate a wide range of relevant information sources and providing almost natural interfaces and interaction concepts, the SoKNOS plattform not only supports collaboration between persons working in an emergency management headquarter, but also faciliates collaboration between different involved agencies, and collaboration between decision makers in the headquarter and mobile first responders in the field, as well. The latter issue is addressed by this work. This is what we call *first response coordination*.

2.1. First Response Coordination Workflow

In this section, we introduce a workflow for the coordination of mobile first responders by a decision maker in an emergency headquarter (cp. Figure 2). It basically consists of three main steps, that we call *See*, *Think* and *Act*.

1) *See:* First, the decision maker visualizes the current situation of the disaster, by displaying relevant information on a large display along with a digital map (cp. Figure 1).
2) *Think:* Second, harnessing the available integrated view on the disaster situation, the actual decision making processes take place. Thus, the person with decision making responsibilities analyzes positions of first responders, takes into account the actual spatial distribution of damages caused by the catastrophe, and further factors. Also, he may display an additional layer of dynamic information, e.g. the spread of toxins in cases of chemical emergencies. All these tools support the decision maker to derive the need for action and to plan incident reactions.
3) *Act:* Third, in order to coordinate the mission of mobile first responders, the decision maker composes a message by textually authoring instructions and specifying the group of receivers, in an intuitive way. In order to so, he selects a physical area, directly on the digital map, he wants the message to be send to. Moreover, he can additionally specify the group of receivers by selecting job and skill descriptions and status information as well.

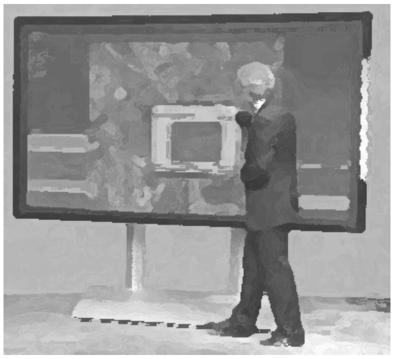

Figure 1. Vision of First Response Coordination

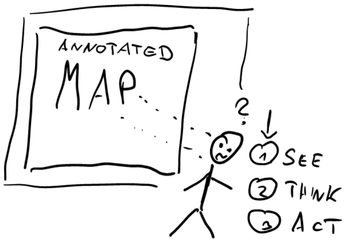

Figure 2. Coordination Workflow

2.1.1. Example. Imaging the following emergency situation: due to an accident in a large chemical plant, a fire erupts, causing containers with hazardous material to collapse. Consequently, a cloud of toxic gas is spreading towards a large city, driven by strong winds. The decision maker in an emergency HQ wants to send out the instruction that affected citizens in the city nearby shall be evacuated immediately. Thus, he sends out a tactical message targeting relevant rescue forces, i.e. high-ranked firefighters that have been active in a mission only for a short time and specialist in toxic matters that are nearby or have a lot of professional experience.

2.2. Discussion

First reponse coordination work is based on the availability of various sources of relevant information, that can technically be integrated and seamlessly be presented on a digital map. This is one of the functionalities that is supported by a first response support system. Thus, an effective coordination of the incident reactions of mobile first responders is possible. The reaction requires efficient means to communicate between emergency staff members in a headquarter and mobile first responders out in the field. The decision maker may select the group of intended receivers in an indirect fashion: in the first response application domain, it is desirable to be able to quickly send messages without knowing the exhaustive list of all receivers. Explicitly addressing all intended receivers could drain mental ressources, that are better spent within the emergency rescue planning processes itself.

3. Attribute-based Messaging

Having outlined our vision of first response coordination work, we next define the underlying concept of attribute-based messaging (ABM). Afterwards, we analyze which data security requirements have to be met in our setting.

3.1. Definition of Attribute-Based Messaging

According to [7], attribute-based messaging is the concept of dynamically specifying the group of receivers of a message in form of an attribute-based description. An attribute-based messaging application is a natural example of a one-to-many communication pattern. If certain security requirements are met, this approach enables the sender to flexibly send confidential messages, since he knows that the messages can only be read by the intended group of receivers.

3.2. Data Security Requirements

In this section, we shortly derive relevant data security requirements. First, by analyzing the representative communication network TETRA[1], second, by evaluating assumptions on possible attackers.

3.2.1. TETRA. TErrestrial Trunked RAdio is a digital trunked mobile radio standard, which is specially designed for authorities and organizations responsible for law enforcement and emergency managment services. Currently, e.g. in Germany, a new communication infrastructure based on TETRA is under development. In order to meet data security requirements of public security organisations, e.g. the police, fire brigades, or even governments, it demands for *end-to-end encryption*. In a TETRA system, mobile units are equipped with a personal device. This mobile device has special security functionalities, i.e. it contains a dedicated smart card for crypto operations and supports secure storage of data [9]. Moreover, the device may include modules that allow to analyze some form of sensor information, e.g. vital data of a mobile unit, recorded by body worn health sensors.

3.2.2. Assumptions on Attackers. We assume a passive attacker to our data security mechanism, who is able to read all submitted messages. We even allow the attacker to compromise the communication network used to transmit the messages. We believe this is an important attacker assumption, depicting real wold experiences of recent terroristic attacks. However, we do not allow the attacker to manipulate the reliabilty of data transmissions over the network. Moreover, we do not allow the attacker to infere with organizational registration processes.

3.2.3. Discussion. Based on our analysis, we conclude that the main data security requirement for a secure attribute-based messaging application is end-to-end confidentiality. Furthermore, tactical messaging requires sender authentication and non-repudiation means and integrity protection. However, in this paper we focus on the requirement of end-to-end confidentiality, based on end-to-end encryption.

3.3. Implementing Secure Attribute-based Messaging

Attribute-based messaging is usually implemented by some form of access control mechanism [7], which allows to control the information flow. A user who cannot access the content of a message is supposed not having received it. In the following, we only consider encryption-based approaches[2], to meet the end-to-end encryption requirement. Basically, encryption-based access control can be implemented based on

[1] P2P-systems alike [8] could be an alternative for providing communication services in first response scenarios. However, considering specific data security requirements of P2P-systems is beyond the scope of this paper.

[2] We are aware that messaging concepts inspired by RBAC do exist (e.g. [10]). A reconsideration and possible integration with our concepts is further work.

1) *Symmetric Encryption*
2) *Asymmetric Encryption*
3) *Identity-based Encryption*
4) *Attribute-based Encryption*

3.3.1. Discussion. In a large-scale or distributed setting, traditional cryptographic constructions suffer from key distribution problems (symmetric encryption) or encryption operation problems (asymmetric encryption). Therefore, these concepts do not scale well in our setting. Identity-based encryption, a certificateless alternative to public key encryption, allows to encrypt messages under textual strings, instead of public keys. However, this approach requires a complete list of all intended receivers. Therefore, we seek to employ a cryptographic technique that is able to express messaging policies on a higher level of abstraction than identities, which could directly translate to the idea of attribute-based messaging. Attribute-based encryption concepts seem to be a natural candidate building block for this: groups of receivers can be selected in an elegant way, by specifying combinations of descriptive attributes. We believe that ABE concepts may significantly support the flexibility secure messaging, in the first response scenario and many other applications, as well. In this paper, we aim at investigating how ABE concepts can be securely and efficiently applied on the way to realize an attribute-based messaging application which guarantees end-to-end confidentiality.

4. Background on ABE

Having argumented for the use of ABE techniques, we describe most relevant basic concepts and characteristics next.

4.1. Basic Principles

Attribute-based encryption concepts generalize the idea behind identity-based encryption: the receiver is not described by a single string representing his identity, but, more generally, by a combination of several descriptive attributes. Naturally, an attribute combination, also called attribute policy in the following, may specify groups of receivers in an elegant way. Therefore, in ABE systems, users intended to receive confidential messages are associated with sets of attributes alike job description and status, and receive related private keys as well. Here, attributes represent the users credentials, i.e. components of his private key. When encrypting, ciphertexts are cryptographically associated with policies, i.e. logical statements on selected attributes. In the encryption process, the sender selects the group of valid receivers by specifying attribute policies. In current ABE implementations, encryption requires algebraic operations and bilinear mappings over certain elliptic curves [11].

ABE can be divided into two main variants: key-policy attribute-based encryption (KP-ABE) and ciphertext-policy attribute-based encryption (CP-ABE). In KP-ABE systems [3], a decryption key is associated with an attribute policy, while messages are labeled with sets of attributes. Thus, it is encoded in the decryption key which kind of messages the key can decrypt. In CP-ABE approaches [5], the control is reversed: a user whose attributes satisfy the attribute policy of a received message can actually decrypt it. Here, it is encoded in a messages, which kind of attributes are required to decrypt the ciphertext.

4.2. Characteristics

Being a generalization of identity-based encryption, ABE shares some important characteristics with IBE systems [12]:

- A sender may encrypt a message to a receiver without the need to interact with some form of Certificate Authority (CA), to obtain the receivers public key and respective certificate status. In consequence, there is no explicit certification in an ABE system. Since a sender is assured that only receivers who actually own the required attributes can decrypt, some form of implicit certification is given, however.
- On a workflow level, it is possible to encrypt a message for a group of receivers before any receiver actually owns the required decryption key.
- One trusted party called attribute authority exists, responsible for private key generation issues. Since it must be able to produce any possible private key matching an attribute combination, it possesses an inherently key escrow funtionality and represents a single point of failure.

ABE exhibits an important difference to IBE: since users are associated with sets of attributes, they might try to trade some attributes and related private key components to gain more decryption powers. However, the implicit certification assumption excludes this attack. This property is called *collusion resistance*, Bethencourt et al. state this to be ”the defining property” [5] of an ABE systems.

4.3. Challenges for applying ABE

In [3], Goyal et al. envision a kind of sophisticated broadcast encryption, where mobile users are described by sets of descriptive attributes. A sender could create a ciphertext that could be decrypted only by a receiver of that message if his attributes match an associated attribute policy. In order to realize this vision, several challenges have to be addressed:

- How can dynamic attributes alike location or status information be handled?
- How must the system be designed in order to allow for efficient application of complex cryptographic algorithms even on mobile devices?
- How may senders intuitively specify attribute policies?

5. Our Approach: Concepts

This section starts with a short introduction of the cryptographic background. Then, our modelling is introduced, next, our approach to flexible attribute policy handling is outlined.

5.1. Building Blocks

An ABE system uses several cryptographic building blocks. We next describe some relevant details. For a broader mathematical background, we refer the reader to [13].

5.1.1. Threshold Secret Sharing. This technique, introduced by Shamir [14], can be used to securely distribute a piece of data, which is considered a secret, among several parties. Especially, to share such a secret $s \in \mathbb{Z}_p$ among n parties, a random polynomial $q(x)$ over $\mathbb{Z}_p$ of degree $t-1$ is chosen, such that $q(0) = s$: $q(x) = s + \sum_{i=1}^{t-1} a_i x^i$. Each of the n parties receives a point $(x_i, q(x_i))$ on this polynomial, and $(0, q(0))$, i.e. the secret itself, is not among these points. If any t out of n parties cooperate they can reconstruct the entire polynomial $q(x)$ using Lagrange interpolation and thus they can produce $q(0) = s$.

If only $t-1$ parties cooperate they cannot reconstruct the polynomial and each constant part of the polynomial, the secret $s = q(0)$, is equally likely. In general, no party can derive information about the secret from his share. The technique described above is called a (t, n) secret sharing. t is the so called threshold, it specifies the minimum number of parties that need to cooperate to reconstruct the secret. Secret sharing is a common cryptographic technique to distribute powers, e.g. access to secret keys, in a secured information system.

In the following contruction, secret sharing is used to construct access policies by distributing a symmetric decryption key in order to create access trees. Thus, decryption capabilities are cryptographically distributed among several attributes, which represent parts of private keys. Furthermore, secret sharing is used to distribute shares of a master secret in the exponent of the user's private key components, i.e. his satisfied attributes.

5.1.2. Bilinear Maps and Pairings. Alike identity-based encryption, attribute-based encryption techniques utilize bilinear maps defined on certain elliptic curve groups. We sum up the properties of relevant maps below. For more details, we refer to [15]. Let $\mathbb{G}_0$ and $\mathbb{G}_1$ be two multiplicative cyclic groups of prime order p. Let g be a generator of $\mathbb{G}_0$ and e be a bilinear map, $e : \mathbb{G}_0 \times \mathbb{G}_0 \rightarrow \mathbb{G}_1$. The bilinear map e has the following properties:

1) Bilinearity: for all $u, v \in \mathbb{G}_0$ and $a, b \in \mathbb{Z}_p$, we have $e(u^a, v^b) = e(u, v)^{ab}$.
2) Non-degeneracy: $e(g, g) \neq 1$.

Associating pairs of elements from $\mathbb{G}_0$ and $\mathbb{G}_0$ with elements in $\mathbb{G}_1$, bilinear maps are also called pairings. If the map is efficently computable, it is additionally called admissible. Such constructions have been used to implement encryption and decryption algorithms which exhibit ABE characteristics [2], [3], [5], [11].

5.2. Large Universe CP-ABE system

Our system is based on an efficient CP-ABE construction. We see it is an instantiation of Sahai and Waters' large universe ABE construction [11] and Bethencourt et al.'s [5] CP-ABE proposal, in our setting.

5.2.1. Participants. The participants of our system are:

- **Setup Authority:** This authority is responsible for generating a master key MK for the attribute authority and a set of public system parameters PK.
- **Attribute Authority:** All users' private keys are generated by a trusted third party, called attribute authority AA.
- **Senders:** They encrypt the plaintext of a message under a certain attribute policy to specify the group of receivers. They use a broadcast channel to send the encrypted message.

- **Receivers:** These parties receive ciphertexts, that are encrypted under an attribute policy. Using their mobile devices, they are able to decrypt received messages, if their attributes satisfy the encoded policy.

5.2.2. Keys, Secrets, Definitions. Several keys and key concepts are considered in our CP-ABE system. In order to avoid confusion, we list them exhaustively next:

- **Public Parameters** (PK)**:** These parameters specify algebraic parameters of the cryptographic algorithms, they can be interpreted as the public key of the whole ABE system, i.e. they are required in any cryptographic operation.
- **Master Key** (MK)**:** This is the master key of the AA. It is necessary to generate private keys.
- **Private Key** (K_{priv})**:** This is the private key of a user, it consists of several private key components, that are related to the set of attributes S satisfied by him.
- **Public Key** (K_{pub})**:** In our scheme, no explicit individual public keys exist for users[3]. Instead, an attribute policy AP plays the role of a group of public keys, for a group of users, specified by a set of attributes S.
- **Set of Attributes** (S)**:** This is a set of attributes. It specifies a group of users, i.e. each member of this group will at least posses this set of attributes.
- **Secret Key** (K_s)**:** This is a symmetric encryption key, e.g. an AES key.
- **Shared Secret** (S_i)**:** This is a share of a piece of data, computed by a secret sharing algorithm.
- **Attribute Policy** (AP)**:** This is an attribute policy, which specifies a combination of selected attributes. Internally, it is implemented as an access tree (cp. section 5.5).
- **Message** (M): A message literally contains a plaintext PT, which encodes tactical instructions.
- **Ciphertext** CT**:** A ciphertext represents an encrypted message. It inherently contains an attribute policy AP.

5.2.3. Algorithms. A CP-ABE system includes the following fundamental algorithms: *Setup*, *KeyGeneration*, *Encryption* and *Decryption*. In the following, as described in section 5.1.2, $e : \mathbb{G}_0 \times \mathbb{G}_0 \rightarrow \mathbb{G}_1$ is a bilinear map. The security parameter k determines the size of the groups. Additionally, the construction employs a hash function $H : \{0,1\}^* \rightarrow \mathbb{G}_0$. The hash function maps any attribute, represented as a bistring, to a random group element. A short description of the algorithms, according to [5], follows.

- **Setup(**k**)** The setup algorithm takes no input other than the implicit security parameter k. It generates and outputs the public parameters PK and the master key MK of the attribute authority. The setup algorithm can be seen as the key generation process for the AA.
  Especially, the cryptographic construction is as follows: after chosing a bilinear group $\mathbb{G}_0$ of prime order p, with generator g, the setup algorithm chooses two random exponents $\alpha, \beta \in \mathbb{Z}_p$. It publishes the public parameters as $PK = \mathbb{G}_0, g, h = g^\beta, e(g,g)^\alpha$. The master key (β, g^α) is kept secret.
- **Key Generation(**MK**,** S**)**: On the input of the master key MK and a set of attributes S, it generates[4] a private key K_{priv} for a user specified by S, consisting of components that are derived from each attribute. The components are cryptographically bound together by some means, in order to achieve collusion resistance.
  Especially, the cryptographic construction is as follows: first, a random value $r \in \mathbb{Z}_p$ is chosen, followed by random values $r_j \in \mathbb{Z}_p$ for each attribute $j \in S$. Then, the private key is computed as $K_{priv} = (D = g^{(\alpha+r)/\beta}, \forall j \in S : D_j = g^r \cdot H(j)^{r_j}, D_j^{'} = g^{r_j})$.
- **Encryption(**PK**,** M**,** AP**)**: On the input of the public parameters PK, a message M and an attribute policy AP, this algorithm encrypts M. Hereby, it generates a ciphertext CT such that only a user that exhibits a set of attributes S that satisfies the attribute policy is able to decrypt to recover the plaintext PT.
  Especially, the cryptographic construction is as follows: First, a polynomial q_x for each node x, even for the leaves, in the access tree representation of the attribute policy (cp. section 5.5) is chosen. Starting from root node R, the algorithm chooses polynomials, in a top-down manner, by setting for each node x the degree d_x of the polynomial q_x to be $d_x = k_x - 1$, where k_x is the threshold value of the node. Then, a secret sharing is started: beginning with the root node R, a random value $s \in \mathbb{Z}_{|}$ is chosen, $q_R(0) = s$. To define the polynomial completely, d_R other points of the polynomial q_R are randomly chosen. For any other node x, $q_x(0) = q_{\text{parent}(x)}(\text{index}(x))$

[3] However, one may target a single user by a single attribute: his identity in string representation, as in the case of IBE.

[4] Note that ABE systems are able to delay the generation of users keys even after the encryption under a certain set of attributes.

is set, and again d_x other points are randomly chosen. Let Y be the set of leaf nodes in the tree representation of the policy. The algorithm computes the ciphertext by giving the access tree representation of the attribute policy AP and constructs $CT = (\text{AP}, \overline{C} = Me(g,g)^{\alpha s}, C = h^s, \forall y \in Y : C_y = g^{q_y(0)}, C_y^{'} = H(\text{att}(y))^{q_y(0)})$.

- **Decryption(PK,CT,K_{priv}):** The algorithm takes as input the public parameters PK, a ciphertext CT, which contains an attribute policy AP, and a private key K_{priv}, which represents a set of attributes S. If the set of attributes S satisfies the attributes policy AP, then the algorithm decrypts the ciphertext CT and returns the resulting plaintext PT, i.e. the message M.
  Especially, the decryption algorithm makes use of a recursive function DecryptNode(CT,K_{priv},x)[5]. The input to this function consists of a ciphertext CT, a private key K_{priv}, and a node x from the tree representation of the attribute policy AP. First, it is called on the root node R of the tree representation of AP. If the set of attributes S associated with the private key K_{priv} satisfies AP, we set $A = \text{DecryptNode}(CT, K_{priv}, R) = e(g,g)^{rq_R(0)} = e(g,g)^{rs}$. To decrypt to M, the algorithm computes $M = \overline{C}/(e(C,D)/A) = \overline{C}/(e(h^s, g^{(\alpha+r)/\beta)})/e(g,g)^{rs})$.

5.3. Relevant Attributes

In our setting, attributes are an abstraction of real world facts and conditions. Analyzing the first response coordination scenario, we found the following groups of attributes that are relevant to our application and list possible values:

- job descriptions: e.g. Firefighter, Policeman, Psychologist;
- professional skills: e.g. Specialist for Toxic Matters, Member of Relief Organization;
- current state of duty: e.g. On Duty, Group Leader, Off Duty;
- current location: e.g. a GPS position, a logical description alike Frankfurt Airport;
- current state of health: e.g. affected, unaffected;
- duration of mission: the time that the person has been active so far.

5.4. Classification of Attributes

We classify the relevant attributes into two main classes: static and dynamic. This is based on the observation that the values of some kind of attributes do not change over time (e.g. Job = Firefighter), while others exhibit a frequent value change (e.g. a Location, which could be the current GPS-position, thus consisting of two coordinates representing lattitude and longitude values). Moreover, the class of dynamic attributes can be divided into groups of attributes with security constraints and without security constraints. We classify the relevant attributes in the following:

- static attribute: job descriptions and professional skills
- dynamic attribute: state of duty, location, state of health, duration of mission
  - with security constraints: state of duty (needs to be incorporated into encryption)
  - without security constraints: location, state of health, duration of mission (does not need to incorporated into encryption)

Dynamic attributes with security constraints[6] need to be directly incorporated into the cryptographic algorithms. Attributes without security constraints can be used for filtering purposes in the message transmission.

5.5. Implementing Attribute Policies

Following the approach of Goyal et al. [3] and Bethencourt et al. [5], we aim at implementing ciphertext policies through access trees. In this approach, access trees are basically built on threshold secret sharing techniques: in a CP-ABE system, a message itself is encrypted under a symmetric key K_s[7]. The symmetric key is included into an access tree. The tree encodes an attribute policy, in the following way: the roof node is associated with the symmetric key K_s, each interior node represents a k-of-n threshold gate. Note that k-of-n threshold gates allow to model logical AND and OR gates as n-of-n and 1-of-n threshold gates (Fig. 3 shows an example of an access tree, which relates to the example given in section 2.1.1). The leaves of the tree are associated with attributes. In order to generate the tree structure, a threshold secret sharing is initiated, starting from the roof node. Here, the symmetric key

[5] For the sake of simplicity, this is a simplified description of the decryption algorithm, according to [5]. Please see there for more details.

[6] We can further classify dynamic attributes with security constraints dependend on their value range: list or continuous value range. A distinct consideration of this issue is out of scope of this paper and is considered future work.

[7] However, explicitly mentioning the symmetric key, we simplify the real encryption process in this section, to make it more comprehensive. See [5], [11], [15] for more details.

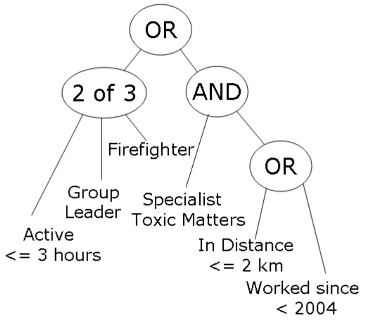

Figure 3. Example of Attribute Policy

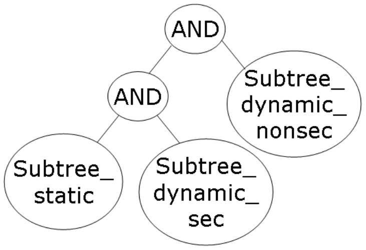

Figure 4. Subtree Construction

is shared among the child nodes according to the policy. For each child node, a threshold secret sharing is recursively executed. Finally, the leaves present a sharing of the symmetric key according to the specified attribute policy. These shares S_i are encrypted under the associated attributes, in the encryption algorithm. Thus, access trees are a way to cryptographically bind a threshold attribute policy to a message. This allows to enforce the policy specified by the sender of the message in a distributed setting. Intuitively, as long as the receiver of a message has enough attributes to satisfy the policy logic encoded in the tree, he can subsequently recover the symmetric key K_s in the roof node, by combining shares from the bottom to the top, and thus decrypt the associated message. Note that attributes can be expressed as strings and numerical values, see [5] for details.

5.6. Handling Dynamic Attributes

In this section, we describe our approach to dealing with dynamic attributes. Especially, we need to consider two aspects: both the generation and the revocation of dynamic attributes.

5.6.1. Basic Construction. Thus far, dynamic aspects of attributes in ABE systems have only been considered from an attribute revocation perspective, by introducing a validity period. This period represents the maximum tolerable vulnerability window, in case the attribute is compromised. It directly extends the value of any attribute, e.g. ”Job = Firefighter - 31stAugust”. This revocation approach requires the attribute authority to regularly issue new keys, respective attributes, to the users of the system[8]. However, in our application, this approach is not applicable since there is no a priori known key validity period concerning dynamic attributes like location and health status. Moreover, a starting point of the validity period is not known before, either. To overcome these issues, we introduce an approach that is able to handle both characteristics. Note that, we only need to consider dynamic attributes with security constraints, i.e. such attributes that have a organisational access control function, in the cryptographic constructions. Therefore, we implement access trees in the following way (cp. Fig. 4): attribute policies are hierarchically divided into three subtrees. The subtree for static attribute is an access tree containing static attribute (job descriptions and professional skills). It is bound to the subtree of dynamic attributes with security constraints (state of duty), which is also an access tree, by an AND gate, i.e. a 2-of-2 threshold gate. The third subtree represents dynamical attributes without security constraints (location, state of health, duration of mission). Therefore, this subtree requires no threshold secret sharing based implementation, however, it follows the same logical construction principle. It can be directly evaluated on a mobile device, once a message has been received, without starting decryption processes. It is logically ANDed with the roof node of the compound access tree made up by the other two subtrees.

5.6.2. Generation of Dynamic Attributes. In our scenario analysis, we identified one dynamic attribute with a security constraint: state of duty. We model this attribute as a list[9] L of possible values v_i, e.g. ”on duty”, ”off duty”, ”group leader”. In case we are able to express the values of dynamic attributes in the form of a list, we are able to apply the following approach: In the key generation, additionally to the static attributes, each attribute of the list is uniquely mapped to a group element of the underlying algebraic

[8] However, we note that other comparable revocation solutions in public key infrastructures (PKIs), e.g. certificate revocation lists (CRLs) are also typically of periodic nature.

[9] The treatment of continuous attributes respective attributes with a very large range of values is out of the scope of this paper. It is considered further work.

group, using a predefined hash function. Then, each group element is translated into a component of the private key, according to the key generation algorithm of the CP-ABE system. These components are transfered to the users device, by the trusted attribute authority. However, components relating to dynamic attributes with security constraints are only stored in a secured compartment of the device, that a user is unable to access. In the key generation process, all attributes, static and dynamic ones with security constraints, are bound together by using a common random factor, which blinds each key component. Jointly using these attributes in an encryption also blinds the ciphertext, in some way. However, in the decryption process, the common blinding factor is used to algebraically unblind the ciphertext, such that the plaintext can be discovered. Note that, if malicous users try to combine components of different keys, the unblinding process in the decryption algorithm does not work [11]. This results in the collusion resistance property of an ABE system.

5.6.3. Activation and Revocation of Dynamic Attributes. Since the solution of restricted validity periods does no apply to our setting, we propose the following concept: after having received an encrypted message, and after filtering on dynamic attributes without security constraints has taken place, the device initiates a decryption process. Therefore, it verifies if the values of dynamic attributes are satisfied, according to the policy associated with the ciphertext. If the verification succeeds, the devices activates necessary attributes. Therefore, it transfers them to a second secure storage compartment that may be read by the decryption algorithm. Together with all other private key components associated with static attributes, they make up the set of attributes S that is feeded to the decryption algorithm. After this algorithm has terminated, the content of the second secure storage compartment is erased, revocating the dynamic attributes.

6. Our Approach: First Response Coordination Scheme

Having described our concepts so far, in the next sections we return to an application scenario point of view. First, we describe the participants and outline the scheme that implements our concepts.

6.1. Participants

We distinguish the following entities in our system:

- a **decision maker**, sender of tactical messages, i.e. messages that contain intructions for first responders,
- a **first responder**, receiver of tactical messages. She is equipped with a personal trusted device that enables her to receive messages from a decision maker, that are send out via a communication channel. The device is able to detect her current state of health by analyzing sensor data from body worn health sensors.
- a **network provider**, which represents a communication channel between decision makers and first responders. We assume that it guarantees for a reliable transport, once a tactical message has been sent. It can be interpreted to implement a broadcast channel.
- a **malicious party**, tries to intercept tactical messages to exploit their content for disrupting the incident response. We model this attacker to have the power to be able to even compromise the network provider, but he is unable to affect the reliability of the communication channel. Moreover, he can't interfere with the registration phase.

6.2. Phases

Our scheme consists of the following consecutive phase:

- **Setup:** In this phase, the setup algorithm of the CP-ABE system is executed, in order to generate necessary system parameters and the attribute authority's master key.
- **Registration:** In this phase, each user of our scheme, i.e. any relevant first responder, receives a distinct mobile device. She interacts with the attribute authority in order to receive a private key according to the attributes she actually satisfies. Especially, private key components that relate to dynamic attributes with security constraints are stored in a secure storage compartment of the devices. In the registration phase, the attribute authority executes the key generation algorithm for each first responder.
- **Messaging:** This phase relates to the *act* step of the first response coordination workflow (cp. section 2.1). A decision maker initiates an attribute-based messaging, basically by applying the encryption algorithm. Therefore, he needs to specify the attribute policy, which may consist of static and dynamic attributes. The computer device of the decision maker encodes the attribute policy according to the construction in section 5.5. Additionally, the decision maker textually author the

content of the message, i.e. tactical instructions for first responders. Afterwards, he sends the message over a broadcast channel, i.e. he hands it out to the network provider in an encrypted form.
- **Receiving:** In this phase, a first responder receives an encrypted message. First, the mobile device evaluates the attribute policy subtree which contains dynamic attributes without security constraints. If the filter conditions specified in this subtree do not hold, the device discards the message. If the encrypted message is not discarded, the device initiates the decryption process. Herein, dynamic attributes are activated according to section 5.6.3. If the resulting set S of satisfied attributes satifies the attribute policy of the encrypted message, the device decrypts the ciphertext. Afterwards, dynamic attributes are revoked. If the mobile device succeeded to decrypt, the device alarms the mobile responder, e.g. by vibrating or by sounding an acoustic signal. Now the first responder can draw his attention to this mobile device and read the plaintext of the received message. Afterwards, he can conduct the proposed real world incident reactions.

7. Analysis

In this section, we theoretically analyse our concepts and the scheme.

7.1. Construction Issues

- Since we employ the large universe construction of [11] and [5], arbitrary bitstrings can be used as attributes. Internally, the construction is based on the use of a collusion-resistant hash-function for incorporating any verifiable attribute into the cryptographic algorithms.
- The subtree approach allows for efficient handling of dynamic attributes without security constraints. We propose to filter first, before starting to decrypt, to yield an efficient construction.

7.2. Security Issues

- Generally, ABE systems exhibit an inherent key escrow functionality, due to the power of the attribute authority. However, this is no drawback in our application. The AA can generate any private key, matching any attribute combination. This even allows to implement a multilateral secure logging system for the messages sent out by the decision maker: during the actual emergency management, any message send out can be recorded in the encrypted form. Afterwards, in dispute cases, a judge may mandate to open a message, again, by requesting the attribute authority to reproduce the necessary keys.
- In some applications, the attribute authority can be a single point of failure. However, we do only require an offline attribute authority. Attackers can not actively mount attacks on the key distribution, as it would be the case in using an online AA. Therefore, this point of failure is removed during the normal operation. However, we require the registration phase to be trustworthy. Since this phase can be organizationally controlled by the emergency management parties, we believe that is a realistic assumption.
- Especially, excluding online attribute authorities, we also avoid to use a network providers for distributing dynamic attributes. However, this could even be in conflict with our end-to-end confidentiality attacker model. Instead, we move the trust into the personal device of the first responders.

7.3. Runtime Prospects

This paper focuses on theoretical considerations. However, our research aims at providing prototypes, as well. Recent research results indicate that it is possible to implement the necessary underlying cryptographic mechanisms, i.e. pairings over elliptic curves, on devices with constraint resources [16]–[19]. An analysis concerning the complexity and average performance of attribute-based encryption on server plattforms can be found in [2]. We are currently working on JAVA and C implementations for both the Android [20] and the Windows Mobile [21] plattforms. Due to this, a runtime evaluation is no available, yet. However, preindicators for the feasibility of our approach are given, without having completed the prototype implementation, so far.

8. Related Work

The concept of attribute-based encryption (ABE) was first proposed in the work of Sahai and Waters [11]. It is an generalization of identity-based cryptography [15], [22]. In [11], both ciphertexts and private key components are associated with descriptive attributes. A user may decrypt a received ciphertext if the set of attributes she owns overlaps with the set of attributes bound to the ciphertext by at least a fixed threshold value, i.e. if they satisfy the attribute policy.

Pirretti et al. [2] and Bethencourt et al. [5] described a variant of this concept, called ciphertext-policy attribute-based encryption (CP-ABE). Here, attributes are associated with components of private keys, attribute policies with ciphertexts. The complementary variant of key-policy attribute-based encryption (KP-ABE) was introduced by Goyal et al. [3]. In this variant, a ciphertext is associated with a set of attributes. A users private key includes an attribute policy. Thus, a user can decrypt, if the ciphertext attributes satisfy this access structure. Both [5] and [3] proposed access trees based on threshold secret sharing to implement more complex policies. We are aware of a growing attention in secure attribute-based systems throughout the data security research community. However, to the best of our knowledge, we believe to be the first ones to propose an ABE-based messaging and coordination application in the first response domain.

9. Conclusion and Future Work

In this paper, we argued that ciphertext-policy attribute-based encryption is a valuable technique to implement secure attribute-based messaging applications with respects to end-to-to confidentiality. Especially,

1) we sketched a workflow to the secure coordination of first responders from a central decision maker in an emergency management headquarter;
2) we described an application of CP-ABE techniques in a first response application scenario;
3) we extended current CP-ABE proposals by additionally taking into account dynamic attributes;
4) we describe and evaluated a scheme based on our dynamic CP-ABE for secure first reponse coordination purposes.

The main features of our proposed concepts are

1) attributes and arbitrary respective private keys can be generated by the sender on the fly, due to the use of a large universe ABE;
2) the scheme is able to handle dynamic attributes, like location - without security constraints - or the status of mobile first responders, in an efficient and secure fashion;
3) while guaranteeing end-to-end encryption, which is one of the major data security requirements for emergency communication;
4) our offline attribute authority construction removes the single point of failure, but allows for handling dispute cases after an emergency has succesfully been dealt with.

Even though a prototypical implementation of our proposed concepts is not finished, yet, we gave arguments that an implementation on standard hardware is possible. A prototype implementation is currently under development. We directly integrate our concepts into the next generation of our first response testbed [23], which will allow us to simulate and evaluate realistic emergency scenarios and respective tactical messaging on a large scale.

We see the need to conduct research in the following areas, in order to develop a full-featured secure attribute-based messaging system:

1) Efficency improvements: in several application scenarios, it might be necessary to associate dynamic attributes with a large set of values with security constraints, e.g. location. We will investigate how to efficently incorporate such continuous dynamic attributes into the cryptographic algorithms. Decryption and attribute generation algorithms will be revisited in order to find optimizations. Moreover, we will practically experiment on resulting design choices for attribute-based systems: which classes of dynamic attributes should be used for message filtering purposes, and which incorporated into the cryptographic algorithms?
2) Authentication and integrity: beyond confidentiality, tactical messaging requires sender authentication and non-repudiation means and integrity protection. We are working on an integration of these issues as extensions to our basic scheme.
3) User interfaces: the specification of messaging-respective attribute policies can be supported in order to speed up the process and to prevent errors. As sketched in our first response coordination vision, we are working on the design of user-friendly interfaces that allow to specify policies in an intuitive way.

We believe that emergency staff members with decision making responsibilites may use our approach to target groups of receivers in a secure way, in order to efficiently coordinate the incident reactions, in the face of rapidly changing situations of large scale disasters. More generally, we are convinced that attribute-based systems have a large potential to realizing secure ubiquitous communication applications, if ressource constraints of mobile and embedded devices can be met.

Acknowledgments. This work was supported by the German Federal Ministry of Education and Research in the context of the project SoKNOS and by CASED (www.cased.de). The authors are responsible for the content of this publication.

References

[1] C. on Planning for Catastrophe, Ed., *Successful Response Starts With A Map: Improving Geospatial Support for Disaster Management*. National Academy Press, 2007.

[2] M. Pirretti, P. Traynor, P. McDaniel, and B. Waters, "Secure attribute-based systems," in *CCS '06: Proceedings of the 13th ACM conference on Computer and communications security*. New York, NY, USA: ACM, 2006, pp. 99–112.

[3] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *CCS '06: Proceedings of the 13th ACM conference on Computer and communications security*. New York, NY, USA: ACM, 2006, pp. 89–98.

[4] P. Traynor, K. Butler, W. Enck, and P. McDaniel, "Realizing massive-scale conditional access systems through attribute-based cryptosystems," in *Proceedings of 15th Annual Network and Distributed System Security Symposium (NDSS 2008)*, 2008.

[5] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *SP '07: Proceedings of the 2007 IEEE Symposium on Security and Privacy*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 321–334.

[6] www.soknos.de, 2008.

[7] R. Bobba, O. Fatemieh, F. Khan, C. A. Gunter, and H. Khurana, "Using attribute-based access control to enable attribute-based messaging," in *ACSAC '06: Proceedings of the 22nd Annual Computer Security Applications Conference on Annual Computer Security Applications Conference*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 403–413.

[8] D. Bradler, E. Aitenbichler, N. Liebau, and B. Schiller, "Towards a Distributed Crisis Response Communication System," *ISCRAM*, vol. 9, p. to appear, 2009.

[9] C. Linde, *Aufbau und Technik des digitalen BOS-Funks*. Franzis Verlag, 2008.

[10] D. Chadwick, G. Lunt, and G. Zhao, "Secure role based messaging," in *IFIP TC-6 TC-11 Conference on Communications and Multimedia Security (CMS 2004*, 2004, pp. 303–316.

[11] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Advances in Cryptology EUROCRYPT 2005*, 2005, pp. 457–473.

[12] C. Gentry, "IBE (Identity-Based Encryption)," in *Handbook of Information Security - Volume 2*, H. Bidgoli, Ed. John Wiley and Sons, 2006, pp. 575–592.

[13] J. Hoffstein, J. Pipher, and J. H. Silverman, *An Introduction to Mathematical Cryptography*. Springer Verlag, 2008.

[14] A. Shamir, "How to Share a Secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[15] D. Boneh and M. K. Franklin, "Identity-based encryption from the weil pairing," in *CRYPTO*, 2001, pp. 213–229.

[16] Y. Kawahara, T. Takagi, and E. Okamoto, "Efficient implementation of tate pairing on a mobile phone using java," in *CIS*, 2006, pp. 396–405.

[17] M. Yoshitomi, T. Takagi, S. Kiyomoto, and T. Tanaka, "Efficient implementation of the pairing on mobilephones using brew," *IEICE Transactions*, vol. 91-D, no. 5, pp. 1330–1337, 2008.

[18] G. M. Bertoni, L. Breveglieri, L. Chen, P. Fragneto, K. A. Harrison, and G. Pelosi, "A pairing sw implementation for smart-cards," *J. Syst. Softw.*, vol. 81, no. 7, pp. 1240–1247, 2008.

[19] M. Scott, N. Costigan, and W. Abdulwahab, "Implementing cryptographic pairings on smartcards," in *CHES*, 2006, pp. 134–147.

[20] www.android.com, 2009.

[21] www.microsoft.com/windowsmobile, 2009.

[22] A. Shamir, "Identity-based cryptosystems and signature schemes," in *CRYPTO*, 1984, pp. 47–53.

[23] D. Bradler, K. Panitzek, I. Schweizer, and M. Muehlhaeuser, "First response communication sandbox," *11th Communications and Networking Simulation Symposium*, pp. 115–122, 2008.