An Efficient Key Management Solution for Personal Network Federations

Shahab Mirzadeh, Haitham Cruickshank, Rahim Tafazolli
Centre for Communication Systems Research (CCSR), University of Surrey
{s.mirzadeh, h. Cruickshank, r. tafazolli} @ surrey.ac.uk

Abstract

Personal Network Federation (PN-F) aims to provide secure interactions between a subset of devices of different Personal Networks (PN) for achieving a common goal or providing some services in collaborative environments. Security and privacy is one of the major concerns in the development and acceptance of PN-F like collaborative networks and as any other security architecture, the key management is the corner stone of any possible solution. In this paper, we provide security mechanisms and protocols for key exchange and key management in PN Federations and specify how the established keys can be used to secure communications in different layers.

1. Introduction

The Personal Network (PN) was introduced and studied during the IST-MAGNET project [1]. PN consists of a dynamic collection of personal nodes and devices around a user (Private PAN or P-PAN), and remote personal nodes and devices in different clusters (home, office, car …) that are connected to each other through infrastructure or ad hoc networks. Figure 1 shows general concept of PN architecture.

Figure 1. PN communication architecture [1]

Figure 2. PN-F concept [2]

The concept of Personal Networks was extended into Personal Network Federations (PN-F) in IST-MAGNETBeyond project [1]. PN federation (PN-F) can be defined as a secure impromptu, situation-aware or beforehand agreed cooperation between a subset of relevant devices belonging to different PNs for the purpose of achieving a common goal or service. Within the federation, devices can communicate with each other and allow access to specific services or resources for performing a common task in collaborative working, family networks, virtual classrooms, healthcare networks, etc. Figure 2 illustrates an example of PN federation with three PNs.

Security and privacy is one of the major concerns in the development and acceptance of personal network technologies. As any other security architecture, the PN and PN-F security architecture needs to use well-established cryptography to authenticate users and devices, authorize network usages, protect the integrity and confidentiality of the data and provide a good network and service availability. In this regard, key management, with its own difficulties and limitations, is the cornerstone of any possible solution.

In MAGNET phase I, a key agreement protocol based on an authenticated Diffie-Hellman (DH) protocol as PN Formation Protocol (PFP) was developed, which fulfils the security needs of small PNs [3]. In MAGNET phase II, we introduced a new key agreement protocol called Certified PN Formation
Protocol (CPFP) for the larger PNs [4]. CPFP, based on the personal public key infrastructure (Personal PKI) and Elliptic Curve Cryptography (ECC), is scalable to larger PNs and provides an enhanced level of authentication and non-repudiation with ease of key revocation and key update.

In this paper, we discuss and specify the PN-F key management solution and cryptographic techniques. The PN federation is a challenging concept, involving open research issues related to the definition and management of the federations, service/user profiles, service/device discovery, security, context management and networking [5]. Based on how the cooperation between the devices in different PNs is realized, we can distinguish between two general type of infrastructure and ad hoc based federations. While in ad hoc based PN federation, our trust establishment is based on direct users’ involvement, in infrastructure based federations our solutions involve high level of trust relationship with a central entity which acts as the trusted third party (TTP) for all the PN participants. In this regard, in section 2 and 3 respectively, we will define our solution for key establishment and authentication between PN-F creator and PN-F participants in infrastructure and ad hoc based federations. In section 4 we discuss the security association establishment between the PN-F participants and clarify how the authentication and access control process will be done. Finally, in section 5 we conclude the paper with security evaluation of proposed protocols. In description of the protocols, we are using the table 1 notations.

<table>
<thead>
<tr>
<th>Table 1. Notation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Symbol</td>
</tr>
<tr>
<td>E(k, m)</td>
</tr>
<tr>
<td>S_X(m)</td>
</tr>
<tr>
<td>P_X(m)</td>
</tr>
<tr>
<td>Cert_X</td>
</tr>
<tr>
<td>PK_X</td>
</tr>
<tr>
<td>SK_X</td>
</tr>
<tr>
<td>X</td>
</tr>
<tr>
<td>MAC(k, X)</td>
</tr>
<tr>
<td>HASH(X)</td>
</tr>
<tr>
<td>r_X</td>
</tr>
<tr>
<td>Concatenation</td>
</tr>
<tr>
<td>Insecure</td>
</tr>
<tr>
<td>PAC</td>
</tr>
</tbody>
</table>

2. PN-F key management in infrastructure based PN federations

MAGNET Beyond project has studied the PN federation architecture in details and has specified a central entity named PN directory service (PNDS) in infrastructure based PN federations. PNDS embraces a business model where users register their PN to a PN service provider and establish their federations via it (figure 3). PNDS plays a central role in infrastructure based federations with acting as a directory service for federations.

Based on how the PN-F creator or PN-F participants publish their federation(s) or announce their willingness to participate in federations, there is two mode of publish based and invitation based PN federation. In publish mode, the creators uses the PNDS to advert their PN-F and candidate participants use it to look-up for the adverted PN-Fs. In the Invitation based, in contrast, the candidate PN-F participants announce their willingness of participation to PNDS and PN-F creators browse and inquire it to find and invite the interested participants to their PN federations [2].

Our key management solution in infrastructure based federations is based on existence of a high level trust relationship with the PNCA as the common trusted third party (TTP) for all the PN-F participants. Without losing any generality, the proposed solutions can be also easily applied to more general case of hierarchical certificate authorities (CAs) of a common public key infrastructure (PKI). In our solution, all the PN-F participants, before participating in any federation, authenticate themselves with the PNDS and get certificates binding their identities to public keys suitable for both encryption and signature verification.

From the security point of view, both cases of publish and invitation mode impose the same requirements on trust establishment and key management solutions and so we exemplify our solution in publish mode. As a result of publish and
Figure 4. Infrastructure based PN federation

discovery stage, PN-F candidate participant knows PN-F creator and can send it a PN-F joining request. PN-F candidate participant and PN-F creator authenticate each other based on their PNDS certificates and establish a secure channel, over which the participant sends its PN-F Participation Profile, mainly consisting on the resources that it makes available to this federation. The Creator then checks whether the Candidate fulfills the federation policies and if this is the case, the private part of the PN-F Profile, consisting on the complete list of PN-F members and a group key is securely forwarded to the new PN-F member. The secure channel should ensure that no adversary can get the group key or replay previously captured messages.

The shared group key allows a light authentication and trust establishment between the PN-F participants, but it does not provide an individual member authentication by its own. If PN-F application mandates the individual member authentication, in extra to group key, the PNDS certificate and PN-F profile (including the members list), can be used to provide mutual authentication. The PN-F creator can also optionally issues PN-F certificates for all the PN-F participants, which can be used in mutual authentication and security association. In this case, the PN-F members have a certificate issued by PN-F creator as the trusted CA of their common PN-F and use it to prove their membership and establish security association with other PN-F participants.

Figure 4 depicts a typical example of authentication and security establishment in publish mode of infrastructure based PN federation, the details is as follow:

- **PN-F creator and participant authenticate with PNDS and get certificates on their public keys. It can be done through the normal and complicated PKI methods or other simple and more usable ways (it has been shown by secure channel in figure). In the current implementation of PNDS in project, PN-F creator and participants authenticate the PNDS based on its certificate (which is preloaded in their terminals), and the PNDS authenticates them through a kind of two factor authentication based on their User ID and Password. The User ID is set by user and includes a mobile phone number which is used to receive the generated Password by PNDS in SMS (Short Message Service) format [6].**

- **Creator uses the PNDS public key to establish a symmetric session key with it and publishes its PN-F in secure way (encrypted by established symmetric key and signed by its private key). In publishing the PN-F, creator can limit its visibility to authorized participants by specifying the credentials that participants should have.**

- **Candidate participant uses the PNDS public key to establish a symmetric session key with it and enquires the PNDS for the available federations. The PNDS controls the discoverable federations based on the participants’ credentials and replies with primary information, including PN-F creator names and their point of contacts and certificates, for the visible registered federations. All the communications are encrypted by established symmetric key and signed by the private keys.**

- **PN-F creator and participant authenticate each others based on their PNDS certificates and proceed to establish a secure connection between them over the main wireless link. To this end, they can use any established public-key-based key exchange protocol which requires them to prove possession of a particular private key such as Secure Socket Layer (SSL) or Transport Layer Security (TLS).**

As an example, depicted in figure, they can establish a pair-wise symmetric key through a 3-pass challenge-response protocol as follow [7]:

- **Candidate participant sends the PN-F creator a signed PN-F joining request, its certificate and a fresh random number \( r_A \).**

- **PN-F creator verifies the authenticity of participant’ certificate; extracts its public key and verifies its signature on joining request. Creator then generates a fresh random number \( r_A \) and a symmetric key \( k_1 \); encrypts \( k_1 \) with participant’ public key and send the result along with \( r_A \) and its signature on \( r_A, r_B, \) participant identity and encrypted \( k_1 \) to the participant.**

- **Participant decrypts key \( k_1 \) and verifies creator’ signature. Participant then generates symmetric key \( k_2 \); encrypts \( k_2 \) with creator public key and sends the result along with its signature on \( r_A, r_B \), creator identity and encrypted \( k_2 \) to the creator. Participant uses a known key**
derivation function (KDF) on both key k1 and k2 to drive the shared symmetric key K_{AB}.

- PN-F creator decrypts key k2, verifies participant’s signature and uses the same KDF on both key k1 and k2 to drive the shared symmetric key K_{AB}.

- Creator sends the PN-F profile, encrypted by the shared symmetric key, to participant. PN-F profile includes an updated list of current joined members and a group key (PN-F key).

3. PN-F key management in ad hoc based federations

In contrary to infrastructure based federations, the dynamic nature of ad hoc based federations does not guarantee that a trusted third party will always be available for the trust establishment and authentication between the PN-F members. In the other hand, key pre-distribution schemes are not also generally applicable, since all the participants within the PN-F may not be known a priori. Based on these facts, our key management solution in ad hoc based federations is based on direct users’ involvement and using extra proximity authenticated channel (PAC) in authenticating the exchanged keys.

A proximity authenticated channel is a communication interface between two devices, which is authenticated by physical means of users. We distinguish between two types of PAC channels, private and public PAC channels, with respect to the level of security the PAC channel can provide. A private PAC channel provides authenticity, integrity and confidentiality, while a public PAC channels provide authenticity and integrity only.

A typical example of a private PAC channel is realized by a user, who reads an alphanumeric string from the display of one device and then enters it to the other device using the keypad. Clearly such a channel sets some limits to the length of the string that can be transferred from one device to another, e.g., typically 32-40 bits, which is feasible to be transferred using devices’ user interfaces by the user. Typical realizations of public PAC are RFID tags, Infrared communication, and public displays on the devices (such as an overhead display over a cashier, printer, or network access point [8]). If the PAC is public, the protocol requires that at least 160 bits of information can be transferred over it.

Similar to infrastructure based federations, there is also two kind of publish and invitation modes in ad hoc based federations. While in invitation mode, the PN-F creator sends its invitation with the public part of PN-F profile to the known participants (known from their adverts for participation or known by neighbor discovery mechanisms), in publish mode, PN-F creators publish their federation(s) by broadcasting their adverts including public part of PN-F profile.

The key management and trust establishment in both cases is similar and we concentrate our discussion on invitation based federations which is depicted by figure 5 and includes the following stages:

- Creator invites the candidate participants by sending them signed invitation messages including the public part of PN-F profile and its public key.
- Candidate participants study the received PN-F profile, verify the signature and send back a signed Join_Request with their public keys (if they are interested in invited federation).
- PN-F Creator and participant authenticate the exchanged public keys over the private or public PAC as follow:
  - Private PAC (figure 5.a): Creator generates a key K and computes a keyed hash on both public keys using the key K, and shows the key and MAC result in truncated form to its user. Candidate participant’s user enters the result in its federation manager which uses the key to compute the similar keyed hash and verifies authenticity of the received
public keys and updates the creator about the outcome.

- Public PAC (figure 5.b): Creator generates a hash of both public keys and sends it to participant over the public PAC. The participant calculates the similar hash, compares the result and updates the creator about the outcome.

- After authentication, creator sends an encrypted copy of the PN-F profile to each participant which includes the PN-F key or PN-F certificate issued by creator. PN-F participants use the PN-F key as the group key or the PN-F certificates for authentication and secure communication in PN-F.

### 4. Security association between the PN-F members

The security association between the PN-F members can be established using shared PN-F key, PN-F certificates and PNDS certificate plus creator-signed PN-F member list. In the first case, all the PN-F members share PN-F key as the group key and use it to prove their PN-F membership. In the second case, all PN-F members have a PN-F certificate on their public key, issued by creator as the PN-F common root certificate authority (CA) and use their private keys to prove they are owners of such that certificates. In the last one, PN-F participants use their PNDS certificate and creator-signed member list (part of PN-F profile) as a proof for their PN-F membership. In this section we investigate these mechanisms in establishing security association between the PN-F members and discuss pros and cons of each method.

#### 4.1. PN-F key based security association

In this method, PN-F creator sends all the PN-F participants a shared PN-F key as a part of PN-F profile which will be used in authentication and security association establishment between the PN-F participants. PN-F participants authenticate each others by showing their knowledge of shared PN-F key and use the PN-F key for secure communication within the PN-F. Figure 6 shows a typical challenge-response protocol that can be used in PN-F participants’ authentication as follow:

- Participant B generates a random number $r_B$ and send it to participant A.
- Participant A generates a random number $r_A$ and sends back a shared PN-F key encrypted version of both random number and participant B identity to participant B.

![Figure 6. PN-F key based security association](image)

- Participant B decrypts the packet, verifies its random number and sends back a shared PN-F key encrypted version of both random number to participant A.
- Participant A decrypts the packet and verifies its random number.

The shared group key does not need any asymmetric cryptography or storage for different key or certificates and allows a light authentication and trust establishment between the PN-F participants, but it does not provide an individual member authentication.

When there is a need to expel a participant, a new group key should be used; in this case, the creator shall update the group key of all remaining members and remove the revoked PN from the member list in the PN-F database.

#### 4.2. PN-F certificate based security association

In this solution, PN-F creator acts as a certificate authority (CA) for its PN-F and issues PN-F certificate for all its participants. It means that every PN-F member get a PN-F certificate on their authenticated public key (authenticated via proximity authenticated channels in ad hoc based federation or via PNCA certificate in infrastructure based federation) which is valid for that PN-F and include PN-F’ ID, member’s identity, issue date, validity and also shows whether the member can invite new member to that federation or no (creator’s right delegation).

The PN-F certificates are used by PN-F participants as proof of membership in PN-F by proving that certificates stem from the same root and participants poses the respective pair keys. PN-F members also use the certificate in authentication and security association with each other by using any established public-key-based key exchange protocol such as Secure Socket Layer (SSL) or Transport Layer Security (TLS).

PN-F certificate allows individual authentication in price of higher processing. It shall be also possible to
revoke a PN from the federation. In case of PN-F certificates, a certification revocation list (CRL) will be used, which will contain all revoked certificates that are still within their validity period. Members will make sure that they are aware of the revoked certificates. The PN-F profile shall define how this CRL will be made known to the members. An obvious option is to retrieve the CRL at the federation manager of the PN-F creator.

4.3. PNDS Certificate based security association

In infrastructure based federations, when each member has a PNCA certificate, PNCA certificates and creator-signed PN-F member list which is part of PN-F profile can be used to prove PN-F membership. In this case, PN-F participants’ mutual authentication and security association establishment is based on PNCA certificate and will be done through the standard public-key-based key exchange protocol such as SSL or TLS.

5. Security evaluation of protocols

Our PN-F key management solution in infrastructure based federation uses the established cryptographic algorithms which according to the current state of knowledge are secured. In ad hoc based federations, our protocols based on using the PAC in authenticating the exchanged public keys, are such designed to be usable (e.g. just entering 4 digits which is easy even for non-technical people) and secure against any man-in-the-middle (MITM) attack.

In private PAC scenario, as the key K is chosen randomly each time and the private PAC provides confidentiality, an attacker gets no knowledge on the key K or on the MAC from the protocol runs. Hence, the only possible attacks are to block the messages over the Private PAC to prevent that the imprinting stage from finishing or to replace PN device’ key with its own key for impersonation and to hope that the MAC value remains valid by coincidence. Assuming a message size of 8 digits, the probability of success in that case is less than $2^{-16}$ which is an accepted target for our protocol.

In public PAC scenario, as the public PAC provides integrity and authenticity, an attacker can either again block the messages to prevent the completion of the imprinting stage or replace either of public keys with another key to achieve impersonation. The replacement remains only undetected if the hash value would be the same. However, as our hash function is collision resistant and its output is at least 160 bits, this is possible only with a negligible probability.

6. Acknowledgement

The authors are grateful of all the partners involved in the MAGNET and MAGNET-Beyond projects, especially security work-packages (WP4), for their fruitful discussions and collaborations.

7. References


