The personal CA – PKI for a Personal Area Network[*]

Christian Gehrmann[1], Kaisa Nyberg[2] and Chris J. Mitchell[3]

[1] Research Dept, Technology Strategies
Ericsson Mobile Platforms AB
Schelevägen 15
SE-221 83 Lund
Sweden
Christian.Gehrmann@emp.ericsson.se

[2] Nokia Research Center
P.O. Box 407
FIN-00045
NOKIA GROUP
Finland
Kaisa.Nyberg@nokia.com

[3] Information Security Group
Royal Holloway
University of London
Egham, Surrey TW20 0EX
UK
C.Mitchell@rhul.ac.uk

ABSTRACT

Initialising and managing security parameters for personal mobile devices within a Personal Area Network (PAN) presents a variety of security problems. In this paper we consider one possible paradigm for mobile device security management, namely the use of a 'personal CA' to provide certificates within a PAN. The security requirements for such a personal CA are analysed, and a novel protocol for secure public key registration and certification is described and discussed. The ongoing management issues for a personal CA are also considered.

I. INTRODUCTION

The next generation of mobile communications is expected to be different from current systems. We foresee changes both for the type of *accesses* to the networks and the *terminals* used to access the networks. We expect future multi-function mobile terminals to consist of several different configurable components, which may be worn about the body and are connected through local wireless communication. The IST SHAMAN (Security for Heterogeneous Access in Mobile Applications and Networks) project (see `http://www.ist-shaman.org`) addresses security problems for *distributed dynamically configurable* terminals. A distributed terminal consists of several *components* in physical proximity to each other and the user or users. They are interconnected with local communication links such as short-range wireless connections, e.g. Bluetooth. This type of personal local network is often called a *Personal Area Network* (PAN). A PAN is a collection of fixed, portable, or moving components close to one person (typically within 10 meters). In order for the user to be confident in using a PAN, the communication between the PAN components must be secured. Secure communication can be achieved by proper security protocols and suitable security associations between the PAN components. Security associations can be created in several different ways. In this paper we introduce a new concept that provides the means for user friendly and fast set up of security associations between PAN components. We call our concept: "The personal CA".

In a 'conventional' Public Key Infrastructure (PKI) model, a Certification Authority (CA) issues a public key certificate. The CA is responsible for checking that the public key in an issued certificate corresponds to a private key that the holder (with the ID given in the certificate) of the certificate possesses. This is necessary in order to maintain the security of a global or very large PKI. The drawbacks of a central CA include:

- it must issue all certificates used by the communication units, and all units must share trusted public root keys; this can be a tedious process that the user of a communication unit would like to avoid;
- it is very costly to maintain a well-controlled highly secure certification process that can handle thousands of users;
- a user that wants to manage his/her own local environment, such as a PAN, will gain few benefits within the PAN from employing a centralised CA;
- the user might not want, for privacy reasons, to delegate the CA operation to a centralised entity outside his personal environment.

Nevertheless, a PKI would be a convenient solution for creating security associations in a PAN. We seek a solution that is adapted to the local PAN environment and that minimises the necessary user interaction. Furthermore, we would like to maintain a reasonable security level. Our personal CA concept provides this.

II. REQUIREMENTS FOR A PERSONAL CA

A personal CA is different from large scale or global CA functions. The personal CA is used by an ordinary user for home or small office deployment. As with any other PKI, we would like all units in a communication network to share common root public keys and use

[*] The work described in this paper has been supported by the European Commission through the IST Programme under Contract IST-2000-25350. The information in this document is provided as is, and no guarantee or warranty is given or implied that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

certificates issued by a trusted CA corresponding to the public root key. In order to use PKI technology in such an environment we need to reconsider the CA policies. One of the *personal* components must act as a "personal CA". Such a component is able to issue certificates to all other personal components. Hence, since all the personal components can be equipped with certificates issued by the same CA, i.e., the personal CA, they will all share a common root public key. Consequently, the public keys in the certificates can be used to exchange session keys or authenticate personal components in a PAN.

Below we discuss the functional requirements for the personal CA and the security requirements on a PKI for a PAN.

A. The Personal CA component

We assume that one of the PAN components is defined as the personal CA. Preferably the component should have a display and a keypad. Examples of possible personal CA components are mobile phones, PDAs or PCs. A personal CA component might be pre-configured (at the manufacturer) with a private/public key pair or might be able to generate such a key pair. The personal CA is used to *initialise* other PAN components. Here we slightly extend the "resurrecting duckling" model of Stajano and Anderson, [4].

At manufacture a component does not have an owner or users. Instead, the component is made first party by an *initialisation* or (as in [4]) at an *imprinting* phase (see section III). At the imprinting phase the necessary keys and certificates are *securely* transferred to the component that can be used to identify the component to all other first party components of the same owner.

The personal CA key pair should be securely generated within the device, or securely generated and transferred to the device at manufacture, and (in both cases) the private key securely stored when on the device. PAN components should be able to verify certificates issued by the personal CA, and check certificate validity and revocation status when appropriate.

Additional and optional functional requirements on the personal CA are:

- the security-critical personal CA functionality (including key generation and storage functions) should preferably be removable, personal and transferable;
- the security-critical personal CA functionality can be directly verified and readily enabled/disabled from a single gateway and/or master user.

B. Security requirements

It is the owner that is responsible for the primary imprinting of his/her devices. At the initialisation phase it should be possible for the user of the personal CA component to confirm the initialisation of the new component. This can be done, for example, by a special key on the keypad of the initialisation component or by entering a check value into the device (see section III).

The general security requirements applying to methods used in the personal PKI are:

- no third party passive interceptor of communications can learn any secret information;
- no third party active interceptor of communications can manipulate the exchanges between mobile device and personal CA so that a public key certificate is created for the incorrect device or that contains incorrect data (e.g. a public key other than that created by the mobile device);
- the interaction between a mobile device and personal CA necessary to transfer the personal CA root certificate to the mobile device, shall use at least a 'weak' shared secret, e.g. a shared password or PIN, and the method used should be capable of resisting 'brute force' attacks on the shared secret.

III. IMPRINTING A MOBILE DEVICE

A. A protocol for device initialisation

The security requirements for the device initialisation process have been listed above. In this section, a protocol for the device initialisation is sketched – see also [1]. Before giving this protocol we observe that, in order to operate successfully, the mobile device and CA must meet certain minimum requirements.

- The personal CA must be equipped with a display and a simple input device for giving it commands.
- The mobile device must possess a moderately sophisticated user interface – that is it must possess both the means for a user to input a sequence of digits (e.g. a numeric keypad or at least two buttons to insert a sequence of zeros and ones), and a simple output device, e.g. an audio output, to indicate success or failure of the initialisation process.

The protocol can be modified for devices with just a display. How the initialisation process might be performed for mobile devices which do not possess a numeric keypad or a display is an issue currently being considered by the SHAMAN project.

Finally note that we also assume that the mobile device and personal CA can communicate via a wireless interface.

The protocol operates as follows.

1. The Personal CA must be reliably informed of the identifier for the mobile device. This could, for example, be achieved by the user typing the identifier for the mobile device into the keyboard of the Personal CA. However, it could also be achieved as part of the protocol itself (see below).
2. The Personal CA sends its public key P_{CA} to the mobile device, and the mobile device sends its

public key P_M to the personal CA. This transfer is assumed to take place via the wireless interface. Along with P_M, the mobile device can send any other information it wishes to have included in the public key certificate which the personal CA will generate (again via the wireless interface). This could, for example, include the identifier for the mobile device.

3. The Personal CA now generates a random key K, where K is suitable for use with a MAC function shared by the Personal CA and the mobile device. Using this key K, the Personal CA computes a MAC as a function of P_{CA}, P_M and any other data supplied by the mobile device. The MAC and the key K are then output by the personal CA (e.g. via a display attached to the personal CA).

4. The user now types the MAC and key K into the mobile device, which uses the key K to recompute the MAC value (using its stored versions of the public keys and associated data). If the two values agree then the mobile device gives a success signal to the user. Otherwise it gives a failure signal.

5. If (and only if) the mobile device emits a success indication, the user instructs the personal CA to generate an appropriate public key certificate. This certificate generation must only take place **after** the mobile device has given the required positive indication. This certificate can then be sent (unprotected) to the mobile device via the wireless interface.

6. The mobile device now performs two checks before accepting the certificate. Firstly the mobile device checks the signature using the personal CA's public key (P_{CA}). Secondly the mobile device verifies that the data fields within the certificate (including the public key P_M and the identifier for the mobile device) are all as expected. The protocol is now complete.

B. Implementation considerations

Apart from meeting the security objectives of the initialisation process, a further primary objective for the design process is to minimise the length of the data strings that the user has to type into the mobile device. This is important for several reasons.

- Firstly, the user will wish the initialisation process to be as quick and simple as possible, arguing in favour of the minimum number of required keystrokes. This is accentuated by the fact that the keypad on the mobile device may be rather small and awkward to use for large strings of data (notwithstanding the ability of many users of existing mobile devices to send text messages using small numeric-only keypads).

- Secondly, the initialisation process should have a high probability of successful completion. This will clearly not be the case if the user is required to enter a large number of digits, especially using a small keypad and/or with a small or non-existent display to give feedback.

- Thirdly, if typing in long data strings is necessitated by the scheme, then it might be just as simple to type in the respective public keys, thus avoiding the threats that arise from use of the wireless interface.

In the protocol described in subsection *A* this minimisation of data entry can be achieved by using a very short key K and a very short MAC. For example, if the key and MAC both contain 4 decimal digits, then the probability that an attacker can successfully manipulate any of the information protected by the MAC is very small (see subsection *D* below).

C. Proof of possession requirements

In some circumstances, before generating a certificate, it is necessary for a CA to ensure that the requester of a public key certificate knows the private key corresponding to the submitted public key. To provide this service, the mobile device could supply a 'proof of possession' of the private key in step 2 of the protocol specified in subsection *A* above.

The nature of this proof of possession will vary depending on the 'type' of the mobile device's public/private key pair. For example, if it is a signature key pair, then the private key can be used to create a 'self-signed certificate', i.e. a signature generated using the mobile device's private key on a string containing the mobile device public key and the mobile device's identifier.

The nature of a proof of possession for a private decryption key or a private key agreement key is not so simple. Preliminary analysis reveals that for decryption and key agreement keys giving proof of possession may require interactions between the personal CA and the client device. The design of a single imprinting protocol providing both secure key exchange and proof of possession for these cases is the subject of further research within SHAMAN.

D. Analysis of the protocol

The purpose of the protocol described in subsection *A* is to transfer the public keys and other data needed for production of the certificate. All data to be transferred is assumed to be public. Therefore the security goal is to protect the integrity of the data, not the confidentiality. The necessary integrity protection is performed using the MAC-based checking procedure in steps 3 and 4 of the protocol.

The security threat against the protocol is an active adversary who by any possible means tries to modify the data exchanged between the CA and the mobile device in step 2. If such a modification, insertion of new data or deletion of data takes place on the wireless communication between the devices, then the data sent by one party will be different from the data received by the other party.

The adversary is successful, if the integrity protection method fails to detect modification of data. In what follows the probability of failure is determined.

For the security analysis of the protocol it is essential to observe that the communication channel used for the checking procedure in steps 3 and 4 is completely independent of the wireless communication channel used for other exchanges of data in the protocol.

Also, different instances of the protocol are independent. This is due to the fact that for each protocol instance the key *K* is randomly generated. The key is generated independently for each protocol instance and for each MAC computation. This means, in particular, that even if the data between two protocol instances are strongly related, the respective MAC values computed using different keys are independent. To achieve this randomisation property of the MAC the length of the key should be larger than or equal to the length of the MAC value.

Let *m* be the bit length of the MAC and *k* the bit length of the key. Then the adversary is successful either if he guesses the key *K* correctly, or if the guess for the key is not correct, but the MAC values for the different data happen to be the same. Hence the probability of success is

$$\frac{1}{2^k} + \left(1 - \frac{1}{2^k}\right) \times \frac{1}{2^m} = \frac{2^m + 2^k - 1}{2^{m+k}}.$$

For a fixed total length of the bit string to be entered to the mobile device, this probability is minimised if the lengths of the MAC and the key *K* are equal, that is, if $m = k$, in which case the success probability for an adversary is approximately equal to 2^{1-k}.

IV. CERTIFICATE MANAGEMENT

Once a mobile device has been imprinted and provided with a public key certificate by the personal CA, there is a need for ongoing management of key pairs and certificates. There are three main issues which need to be resolved within the PAN:

- *Certificate and key pair update*, i.e. methods to be used when a device wishes to use a new key pair or when the certificate for a current key pair has expired,
- *Key status management*, i.e. disseminating information regarding revoked public keys across the PAN, and
- *Trust management*, i.e. managing the relationship between the mobile device and the personal CA, including CA (root) key update and the possible replacement of personal CA devices (especially in the event of lost or stolen personal CA devices).

We now consider each of these issues in more detail.

A. Certificate and key pair update

If the mobile device merely wishes to obtain a new certificate for an existing public key, then because of the scale of the personal PKI a simple solution is possible. Given that the total number of personal devices will be small it is likely to be possible for the personal CA to securely retain a copy of all public keys for which it generates certificates. It could even routinely check the certificates to see if any of them have expired. Once the need for a new certificate has been determined, the personal CA device simply asks the user if the existing key pair should be renewed. Once the user has agreed, a new certificate can be generated and passed to the device concerned across the wireless interface at the next opportunity.

Even if storing all public keys at the personal CA is not feasible, in certain cases it may be possible to use a relatively simple certificate renewal process. The mobile device requiring a new certificate could pass the expired certificate to the personal CA which would then pass the relevant information to the user for a decision. If the user agrees a new certificate can be generated.

If a new key pair is to be assigned to the mobile device, then the renewal process becomes more difficult. In some cases it may be possible to use the old key pair to establish a secure exchange between personal CA and mobile device – however, if the key pair is still trusted to secure this process then it is not clear why it would need to be changed. Indeed, the default for many inexpensive mobile devices may simply be to use the same key pair indefinitely.

However, if a new key pair is definitely required, and if the old key pair cannot be used to secure the interactions between personal CA and mobile device, then a new imprinting process will probably be necessary. Given that this will involve relatively few user keystrokes, and given also that this will probably be a rare event, this should not present a huge practical problem for the user.

B. Key status management

We consider two different ways in which certificate status information can be disseminated to a mobile device. The choice between the two approaches depends on the online availability of the personal CA.

The first approach we call *online status dissemination*. This is designed for use in the case where the personal CA is available online to every mobile device either permanently or at least at frequent intervals. In the case where the personal CA is permanently online then an online status query protocol could be used, e.g. a protocol along the lines of the Online Certificate Status Protocol (OCSP) – see, for example, [2]. However, because of the small scale and relatively closed nature of the personal PKI it may be possible to use a simplified version of OCSP.

In the case where the personal CA is not always online, but is nevertheless online at frequent regular intervals, the use of routinely distributed Certificate Revocation Lists (CRLs) – see, for example, X.509, [3] – would appear to be appropriate. In this approach the personal CA generates new CRLs at regular intervals and distributes them automatically to all mobile devices.

Whilst the personal CA is not online permanently, and neither are all mobile devices, this approach will be appropriate in cases where the personal CA is online sufficiently often that the chances of every mobile device having the latest CRL are very high.

The second case we call *ad hoc status dissemination*. This is designed for use when the personal CA is only online intermittently or rarely. In such a case, a mobile device may not be online at the same time as the personal CA very often, in which case a directly distributed CRL no longer seems appropriate. Thus an alternative means for distributing CRLs is necessary.

As in the previous case we assume that the personal CA generates CRLs at regular intervals. We also suppose that the personal CA is online sufficiently often that it can distribute the latest CRL to at least one mobile device (if not then there is clearly no way of distributing timely status information). Subsequent distribution of CRLs is then assumed to occur in an ad hoc fashion between mobile devices. That is, whenever mobile devices communicate, they exchange the serial number of the CRLs they possess. If one device has a higher serial number than the other then it passes the latest CRL to the other device. By this means the latest CRL should disseminate across the PAN very rapidly, without requiring active support from the personal CA.

Such an approach may even be appropriate in other networks. Note, however, that in networks with a central or critical node through which all traffic passes, e.g. networks with a star topology, problems may arise if the certificate of the critical component is about to revoked and the component does not co-operate.

C. Trust management

We first consider the routine updating of root keys, i.e. when an existing personal CA wishes to update its key pair. If the old root public key has not been revoked, then this could be achieved by distributing a certificate for the new root public key signed using the old CA private key. Whilst this approach has dangers, it may be sufficiently secure for use in a PAN environment. The only alternative would appear to be to engage in a new imprinting process with all mobile devices, which could be a rather onerous process for the user.

The case of a compromised or stolen personal CA is rather more difficult. In such a case there is a need to inform all mobile devices of this in a timely way. Of course, once the root key has been revoked, then secure communications between devices will become impossible unless another root key (and a certificate signed using this key) is available. There would appear to be two main approaches to dealing with this issue.

The first approach is to use multiple personal CAs. In this case every device will have multiple root keys and multiple certificates for their public key(s). If two or more Personal CAs are available at the time a mobile device is imprinted, then it should be possible to devise a special version of the imprinting protocol given in section III.*A* to enable simultaneous registration and certificate generation. When one CA root public key is to be revoked, then the mobile devices can be informed by the remaining personal CAs, using the same mechanism as is used to disseminate revocation information for other mobile devices.

The second approach is to re-imprint every device with a replacement personal CA as soon as possible after the loss of the old personal CA. Such a process can be designed to simultaneously revoke the old CA and register with the new CA. An appropriately modified version of the imprinting protocol described in section III.*A* above will need to be used.

V. SUMMARY AND CONCLUSIONS

We have introduced the “Personal CA” concept, designed to simplify security management within a PAN. We have analysed the security requirements for the personal CA, and we have proposed a protocol for device imprinting, which combines security with minimal intervention by the user. Finally we have considered ongoing security management issues.

Although the proposed imprinting protocol appears to meet all the identified requirements, certain issues still remain, and these will be the subject of future research within the IST SHAMAN project. These issues include the design of imprinting protocols for mobile devices without any keypad or similar user input capability, and the integration of proof of possession techniques with imprinting protocols for the cases of decryption and key agreement keys.

The authors would like to acknowledge the helpful advice and encouragement of all their colleagues in the SHAMAN project, and in particular those of András Méhes and Scarlet Schwiderski-Grosche.

REFERENCES

[1] C. Gehrmann and K. Nyberg, “Enhancements to Bluetooth baseband security”, in *Proceedings of Nordsec 2001, Copenhagen*, November 2001.

[2] ISO/IEC 15945: 2002, *Information technology – Security techniques – Specification of TTP services to support the application of digital signatures*. [=ITU-T X.843].

[3] ITU-T X.509 (03/2000), *Information technology – Open systems interconnection – The directory: Public-key and attribute certificate frameworks*. [=ISO/IEC 9594-8: 2001].

[4] F. Stajano and R. Anderson, “The resurrecting duckling: Security for ad-hoc wireless networks”, in B. Christianson, B. Crispo, and M. Roe (Eds.), *Security Protocols, 7th International Workshop Proceedings*, LNCS, vol. 1796, Springer, 1999, pp. 172-194.