



# *A Novel Decentralized Hierarchical Access Control Scheme for the Medical Scenario*

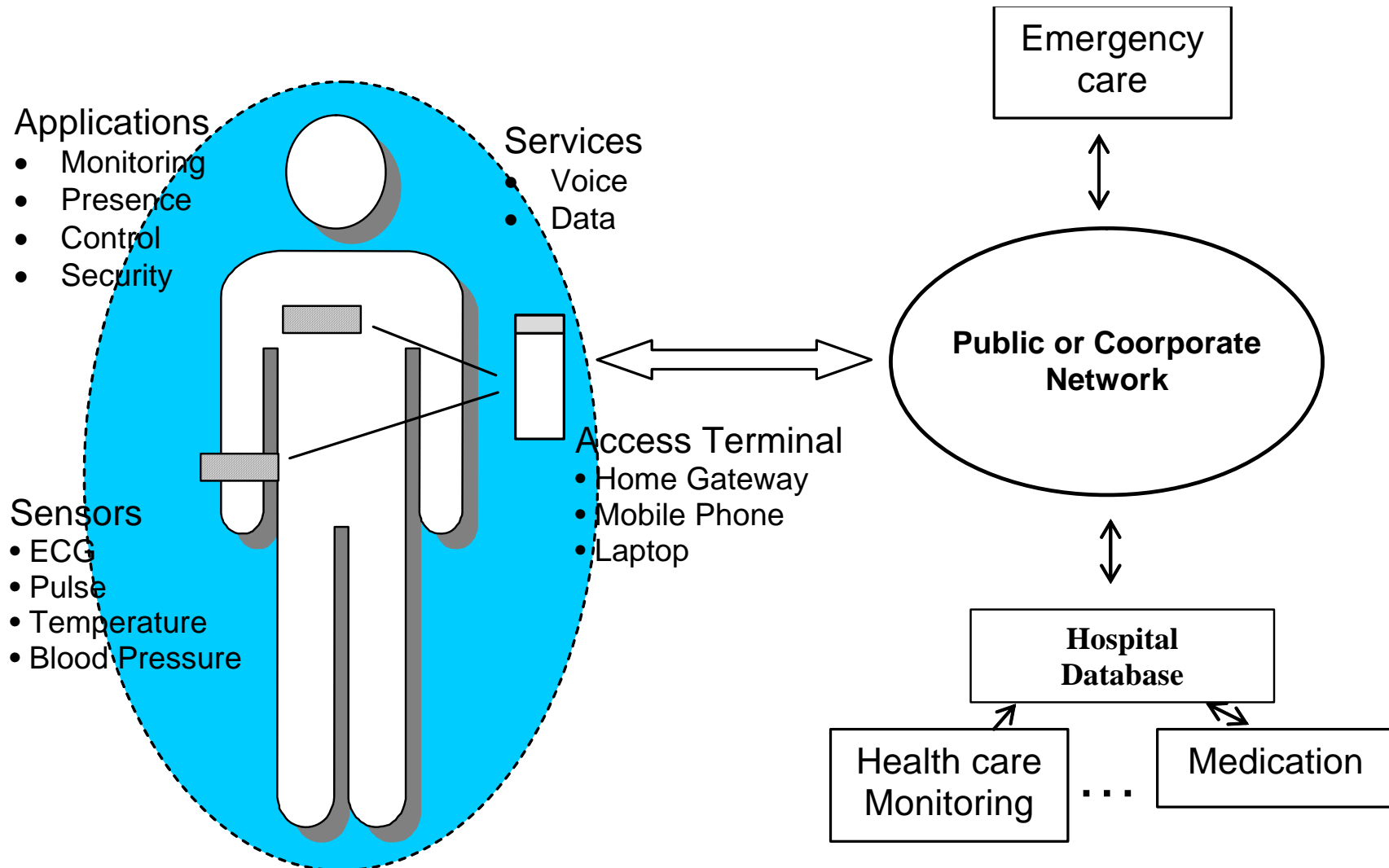
**Sigurd Eskeland  
Neeli R. Prasad**

**International Workshop on Ubiquitous Access Control  
July 17, 2006 - San Jose, California, USA**

# Outline

- Introduction
- Security Requirements
- The Proposed Scheme
  - User arrangement
  - A novel Hierarchical Conference Key Agreement (HCKA) scheme
  - Example
  - Validation and granting
- Conclusions

# Introduction – Medical Scenario



# Medical Database

- High number of medical professionals providing medical care to patients
- Electronic Patient Records (EPR)
  - contain highly confidential information
  - are of great privacy concern
- Only legitimate medical professionals must obtain access to EPRs
- Security Administrators may be insufficient
  - Patients have no control of EPRs
  - Possible risk of error and breach of confidentiality

# Introduction - Teams

- Medical professionals provide medical care to patients in teams
- It is in patients interest to exert control over their own EPR
- In this security model, patients can grant teams access to their own EPR

# Introduction - Hierarchies

- Each team member represents some role (or job function) like “*doctor*” or “*nurse*”
  - A medical role corresponds to a hierarchical level or user privileges
    - Doctors are to be allowed more privileges than nurses
- EPRs contain a number of data modules of various sensitivity
  - The EPR modules form a hierarchy according to sensitivity
- Higher level users should access the same EPR modules as lower level users + other EPR modules in agreement with their privileges
  - Doctors should be allowed access to more data than nurses

# Security requirements

- Key secrecy
- Hierarchical direction
- A signature scheme should be selected that provides
  - Authentication at user and privilege level
  - Forwards secrecy
    - Compromise of long-term user keys does not reveal former established keys

# The proposed scheme – Two-fold

- Hierarchical Conference Key Agreement protocol
  - The team secretly establishes by collaboration one key for each hierarchical level  
→ HCKA protocol
- Validation and granting
  - The HCKA protocol allows the EPR server and the patient to certify the composition of the team, and thus validate the team
  - Due to the team validation, the patient may grant the team access to his or her EPR

# Proposed Access Control Scheme

The proposed scheme consists of two security protocols.

**STEP 1:** The hierarchical key establishment protocol involving the team, server and patient which is executed first.

- The actual granting by means of validation of the results of the former, involving the EPR server and the patient.

**STEP 2:** The patient exerts control of authorization by certifiably obtaining the identities and ranking of the participants of the medical team.

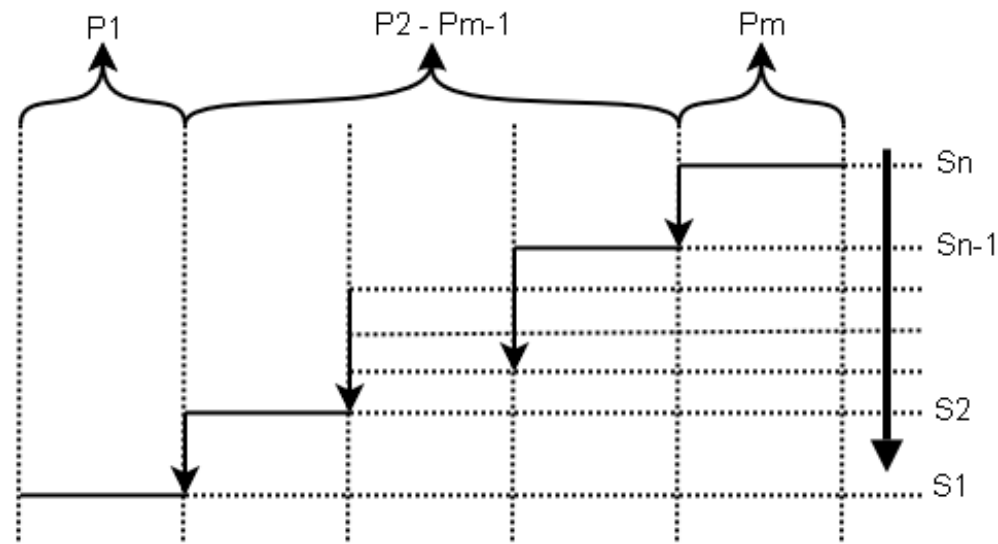
- Completion of this stage signifies that authorization to access the EPR is granted
- The medical data is encrypted according to confidentiality levels by the newly established hierarchical keys.

# User arrangement

- All participants  $\{P_1, \dots, P_m\}$  are sequentially arranged, forming a logical line:
  - $P_1$  denotes the patient
  - $P_m$  denotes the EPR server
  - $\{P_2, \dots, P_{m-1}\}$  denotes the members of a team
- The participants are arranged into levels:
  - The patient  $P_1$  is located at the bottom level  $S_1$
  - The EPR server  $P_m$  is located at the top level  $S_n$
  - The team members  $\{P_2, \dots, P_{m-1}\}$  are located in between according to their respective roles

# User arrangement (cont'd)

- All participants of each security level must be arranged in one sequence
- The security levels must be increasingly arranged, i.e.  $(S_1, S_2, \dots, S_{n-1}, S_n)$



# A novel Hierarchical Conference Key Agreement scheme

- Hierarchical key agreement scheme for medical teams
- Preparations
  - A Trusted Party (TP) selects two large two large primes  $p$  and  $q$  where
$$p = 2 \cdot q + 1$$
- All computations are done *modulo*  $p$

## Round 1

- Each user  $P_j$ ,  $1 \leq j \leq m$ , generates a random secret number  $r_j$ , and computes and broadcasts  $z_j = \alpha^{r_j} \pmod{p}$

## Round 2

$$k_{j-1,j} = z_{j-1}^{r_j} \pmod{p} \quad \text{and} \quad k_{j,j+1} = z_{j+1}^{r_j} \pmod{p}$$

- Each user  $P_j$ ,  $2 \leq j \leq m-1$ , computes

$$v_j = k_{j-1,j}^* - f(k_{j,j+1})$$

The sequentially first positioned user of each level computes

$$v_j = k_{j-1,j} - k_{j,j+1}$$

where  $f$  is a secure one-way function

- Other users compute

## Round 2 (cont'd)

- The patient  $P_1$  and EPR server  $P_m$  who do not have two adjacent users set:  $v_1 = v_m = c$
- Each user  $P_j$ ,  $1 \leq j \leq m$ , broadcasts  
 $(ID_j, v_j, Sig(v_j, c))$

where  $Sig_j(v_j, c)$  is a signature binding a user with his identity, user level/role and the current session:  $c = f(z_1, \dots, z_m)$

# Controlling entrance keys

- It is essential that *key direction* is preserved, i.e. that lower level users cannot deduce higher level keys
- To verify this, for each level, compute a candidate “entrance” key:

$$k'_{j-1,j} = v_j + f(k_{j,j+1})$$

If the following verification holds, key direction is preserved:

$$v_j^2 \stackrel{?}{\equiv} k_{j-1,j}'^2 - 2k_{j-1,j}' f(k_{j,j+1}) + f(k_{j,j+1})^2$$

# Establishment of key of same level

- For a given security level  $S_\ell$ ,  $1 \leq \ell \leq n$ , all participants in  $S_\ell$  contributes to the key  $Z_\ell$
- Each user  $\{P_i, \dots, P_k\} \subseteq S_\ell$  where  $P_i$  is sequentially first and  $P_k$  is sequentially last, can compute the key as:

$$\begin{aligned} Z_\ell &= (k - j) \cdot z_{j-1}^{r_j} + \sum_{l=i+1}^{j-1} (l - j) \cdot v_l - \sum_{l=k-1}^j (k - l) \cdot v_l \pmod{p} \\ &= k_{i,i+1} + k_{i+1,i+2} + \dots + k_{k-2,k-1} + k_{k-1,k} \end{aligned}$$

# Establishment of keys of lower levels

- For a user  $P_k \in S_\ell$ ,  $2 \leq \ell \leq n$ , to obtain a lower level key  $Z_\gamma$ ,  $1 \leq \gamma < \ell$ , the entrance key for each intermediate level  $\delta$  must be computed:

$$k_{i-1,i}^* = v_i + f\left(\sum_{j=i+1}^{k-1} v_j + k_{k-1,k} \pmod{p}\right) \pmod{p}$$

where

- $k_{k-1,k}$  is the entrance key of  $S_\delta$
- $k_{i-1,i}$  is the entrance key of  $S_{\delta-1}$

# Establishment of keys of lower levels (cont'd)

- The target level key is computed as:

$$\begin{aligned} Z_\gamma &= (k-1) \cdot k_{k-1,k}^* + \sum_{l=i+1}^{k-1} (k-l) \cdot v_l \pmod{p} \\ &= k_{i,i+1} + k_{i+1,i+2} + \dots + k_{k-2,k-1} + k_{k-1,k} \end{aligned}$$

# Example

- There are eight users are located in two security levels,  $S_1 = \{P_1, P_2, P_3, P_4\}$  and  $S_2 = \{P_5, P_6, P_7, P_8\}$  where  $S_1 < S_2$ .
- $P_8$  can compute  $Z_2$  as:

$$Z_2 = v_6 + 2 \cdot v_7 + 3 \cdot k_{7,8} = k_{5,6} + k_{6,7} + k_{7,8} \pmod{p}$$

# Example (cont'd)

- The entrance key is deduced as:

$$\begin{aligned}k_{4,5}^* &= v_5 + f(v_6 + v_7 + k_{7,8}(\text{mod } p)) \pmod{p} \\ &= (k_{4,5} - f(k_{5,6})) + f((k_{5,6} - k_{6,7}) + (k_{6,7} - k_{7,8}) + k_{7,8}(\text{mod } p)) \pmod{p}\end{aligned}$$

- $Z_1$  is then:

$$Z_1 = v_2 + 2 \cdot v_3 + 3 \cdot (v_4 + k_{4,5}^*) = k_{1,2} + k_{2,3} + k_{3,4} \pmod{p}$$

# Validation and granting

- By means of the presented HCKA protocol, the EPR server  $P_m$  obtains the composition of the team, i.e. a list  $ID$  of the identities of the team members
  - $P_m$  signs a nonce  $N_m$ , the list  $ID$  and  $k_{1,2}$ , and sends this to  $P_1$ :

$$P_m \rightarrow P_1 : N_m, Sig_m((ID), f(k_{1,2}), N_m)$$

- The patient  $P_1$  validates the composition of the team according to the list  $ID$ . If he wants to grant the team EPR access, the patient:
  - verifies the signature, and  $k_{1,2}$
  - signs the nonce  $N_m$  and  $k_{1,2}$ , and sends the signature to the server:

$$P_1 \rightarrow P_m : Sig_1(N_m, f(k_{1,2}))$$

- otherwise, stop
- $P_m$  verifies the signature, and if valid, encrypts the EPR according to the established key hierarchy

# Conclusions

- Cryptographic hierarchical key agreement scheme for teams
- The patient can grant or deny a candidate medical team access to his or her EPR
- Due to hierarchy, more fine-granular EPR access can be obtained according to the individual user privileges
- **Adaptive**

# Typos in the paper

- In page 3 in "Round 2".  $k_{j-1,j}$  etc. NOT  $k_{i-1,i}$ .
- There are two errors on page 4
  - line 6, col 1:  
 $k_{j-1} = v_j + f(k_{j,j+1})$ ,  
NOT  $k_{j-1} = v_j + f(k_{j-1,j})$
  - end of page 4:  
 $Z_2 = v_6 + 2 v_7 + 3 k_{7,8}$  ,  
NOT  $Z_2 = v_6 + 2 v_7 + 4 k_{7,8}$ .



# Thank you!