Tatra Mt. Math. Publ. **41** (2008), 153–166

TATRA MOUNTAINS
Mathematical Publications

AN EXTENSION OF PROTOCOL VERIFICATION MODAL LOGIC TO MULTI-CHANNEL PROTOCOLS

Péter Takács—Sándor Vályi

ABSTRACT. The first purpose of this paper is to extend Coffey-Saidha-Newe modal logic to be able to deal with multi-channel protocols. Next, we apply the extended logic to verify validity of protocols in the MANA family.

1. Introduction

1.1. Security of wireless networks — Manual authentication techniques

Cryptographic applications very often use session keys in the communication processes to support secure connections. Although session keys complicate the cryptographic systems, at the same time they significantly reduce the possibility of certain attacks. For example, in ad-hoc networks — which have a growing popularity nowadays — it is necessary to apply session keys. At the same time key management infrastructure is not solved in smaller ad-hoc networks (for example, in personal area networks — PANs).

One recommended solution to build secure connections and to solve key management problems is human assisted authentication. This authentication procedure is not totally automatic, human assistance is required when the protocols run. For example, the Bluetooth technology uses short personal identification numbers to create associations between devices [3].

In these protocols, the human assistant is used as an auxiliary channel. This assistance can be, for example, key in the same information to both of the devices or comparing the outputs of the devices or key in data from one device to another device [14]. These protocols are typically multi-channel protocols. These protocols are usually called human assisted pairing protocols.

2000 Mathematics Subject Classification: 68Q60, 03B70, 03B42.
Keywords: formal verification, cryptographic protocols.

1.2. Human assisted pairing protocols

The set of human assisted pairing protocols can be divided into two subclasses.

1.2.1. Protocols based on numeric comparison

In the subclass of numeric comparison based protocols, the human assistant must compare data in the devices. Some examples for this type of protocols are the next:

- MANual Authentication Protocol - MANA I [8, 9],
- MANA II Protocol [8, 9],
- MANA IV Protocol [11],
- Three-round Mutual Authentication Protocol [10],
- Bidirectional Authentication Protocol [6].

1.2.2. Passkey-based protocols

The subclass of passkey-based protocols are based on a shared secret, between the devices. Two examples for this type of protocols are the following:

- EKE Protocol [1, 2],
- MANA III Protocol [8].

1.3. Multi-channel protocols and formal proofs

Formal methods can be used in various phases of the design of cryptographic protocols. These phases are specification, construction and verification. The verification is the most developed research area of cryptographic protocols. Based on [Buttyan], one can classify formal verification into four types — general modelling tools, expert systems, modal logics and algebraic tools.

We use modal logics tools to examine cryptographic protocols. The first momentous result was BAN logic [4] in 1989 in this area. BAN logic has been extended in many directions (GNY logic, CKT5 logic, KPL logic, etc. see [5]).

In 2005, W o n g and S t a j a n o called the public attention to the following: *" ... Finally, the last and perhaps the most important tool we need is a logic for multi-channel protocols in the spirit of BAN"* [16]. Now we extend the Coffey--Saidha-Newe (CSN) system and apply it to examine multi-channel protocols *in the spirit of BAN.*

The Coffey-Saidha-Newe (CSN) logic was presented in two papers [7, 12]. The first paper describes a modal logic which is capable to describe public key systems and the second paper gives an extension to secret key systems. In the appendix we recall the CSN system.

2. The extension of the CSN logic

We can apply the CSN system to a wide area of protocols, but it does not have enough syntactic expressive power to deal with multi-channel protocols and cannot be used in case of multi-channel protocols. We extend the original CSN system for this purpose.

2.1. The syntactical extension

We need to indicate the channels in the formalization: to extend CSN logic with channel signs.

- Let c denote the number of channels.
- Let ch_1, ch_2, $\ldots$, ch_c denote the channels.
- ENT_{ch_i} is the subset of the entities permitted to send/receive messages to/from the channel ch_i. $ENT_{ch_i} \subseteq ENT$.

We need devices to describe the channel properties in the system, too.

- Let $CH(ch_i, sec)$ denote the fact that channel ch_i is secret channel and similarly, let $CH(ch_i, pub)$ denote that ch_i is public channel. If a channel is protected, we can set the users who can use the channel: ENT_{ch_i}.

We also have to introduce a channel index to the reception predicate R and to the emission predicate S. The original R operator is $R(\Sigma, t, x)$. It means that entity Σ receives message x at time t.

- Let the new form of *receive* operator be $R(ch_i, \Sigma, t, x)$. It means that entity Σ receives message x at time t from the channel ch_i.
- $S(\Sigma, t, x)$ now abbreviates $S(ch_1, \Sigma, t, x) \vee$ $S(ch_2, \Sigma, t, x) \vee \ldots \vee S(ch_c, \Sigma, t, x)$.
- $R(\Sigma, t, x)$ now abbreviates $R(ch_1, \Sigma, t, x) \vee$ $R(ch_2, \Sigma, t, x) \vee \ldots \vee R(ch_c, \Sigma, t, x)$.

The original S operator is $S(\Sigma, t, x)$. It means, Σ sends message x at time t.

- The new form of *sent* operator is $S(ch_i, \Sigma, t, x)$. It means, that entity Σ sends message x at time t to the channel ch_i.

2.2. The extended axiomatic system

We do not change the set of rules of inference. Only axioms A5, A6, A8, A12 and A15 have to be changed. The new versions are the following.

A5'(a): $S(ch_i, \Sigma, t, x) \rightarrow L_{\Sigma,t}x \wedge \exists\, i,\ i \in \{ENT_{ch_i}/\Sigma\}$ $\exists\, t',\ t' > tR(ch_i, i, t', x')$.

A6'(a): $R(ch_i, \Sigma, t, x) \rightarrow L_{\Sigma,t}x \wedge \exists\, i,\ i \in \{ENT_{ch_i}/\Sigma\}$ $\exists\, t',\ t' < tS(ch_i, i, t', x')$.

A8'(a): $\neg L_{i,t}k_\Sigma \wedge \forall t',\, t' < t \neg L_{i,t'}\big(e(x,k_\Sigma)\big) \wedge$
$\neg\big(\exists y(R(ch,i,t,y) \wedge C(y,e(x,k_\Sigma)))\big) \rightarrow \neg L_{i,t}\big(e(x,k_\Sigma)\big).$

A8'(b): $\neg L_{i,t}k_\Sigma^{-1} \wedge \forall t',\, t' < t \neg L_{i,t'}\big(d(x,k_\Sigma^{-1})\big) \wedge$
$\neg\big(\exists y(R(ch,i,t,y) \wedge C(y,d(x,k_\Sigma^{-1})))\big) \rightarrow \neg L_{i,t}\big(d(x,k_\Sigma^{-1})\big).$

A12'(a): $\big(\neg L_{i,t}ks_{(\Sigma,\Psi)} \wedge \forall t',\, t' < t \neg L_{i,t'}(E(x,ks_{(\Sigma,\Psi)})) \wedge$
$\neg(\exists y(R(ch,i,t,y) \wedge C(y,E(x,ks_{(\Sigma,\Psi)})))) \rightarrow \neg L_{i,t}(E(x,ks_{(\Sigma,\Psi)}))\big).$

A12'(b): $\big(\neg L_{i,t}ks_{(\Sigma,\Psi)} \wedge \forall t',\, t' < t \neg L_{i,t'}(D(x,ks_{(\Sigma,\Psi)})) \wedge$
$\neg(\exists y(R(ch,i,t,y) \wedge C(y,D(x,ks_{(\Sigma,\Psi)})))) \rightarrow \neg L_{i,t}(D(x,ks_{(\Sigma,\Psi)}))\big).$

We need a new axiom for secure communication.

A16: $CH(ch_b,sec) \wedge ENT_{ch_b} = \{i,j\} \wedge S(ch_b,i,t,x) \wedge R(ch_b,j,t',y)$
$\wedge \forall t''\big(t < t'' < t' \rightarrow \neg\exists u R(ch_b,j,t'',u)\big) \rightarrow x = y.$

3. Verifications for protocols in the MANA protocol family

In the previous section we have built an axiomatic system to be able to deal with multi-channel protocols. We demonstrate the use of the created system by verifying the goals of protocols in the MANA family. This family plays an important role of the SHAMAN project of leading European mobile communication firms supported by the European Commission's Information Society Technologies programme [9, 15].

These protocols constitute a basic infrastructure of mobile communications so we have chosen this protocol family as the subject of our research.

3.1. The MANA protocol family

There are four protocols (and some sub-variants) in this family at present (MANA I-IV, MA-DH etc.). Differences between the protocols are in the availability of devices (device with keypad, LED, screen, display, input button, etc.) and the steps of protocols — evidently. In the MANA protocol family the public channel is generally fast and wideband. The unpublic and secure channel is typically a lowband manual channel — the user reads or writes the channel signs [8, 9, 11, 15, 16]. Next we examine the first two protocols — MANA I, II.

The process of protocol verification starts with the formal description of the steps in the given protocol. By the initial assumptions one fixes the basic conditions. In the proof one can use the axioms, the initial assumptions and the specified steps of the protocol.

3.2. MANA I

In MANA I protocol, device A and device B try to agree on a data string D_A. For example, this could be the concatenation of the two public keys of two devices or other cryptographic initialization parameters. Device A has a display and a simple input — a binary switch. The other device B has a keypad and a simple output — a LED. They use the public (for example wireless) channel ch_1, and user U helps and supervises them. User U handles two secure channels ch_2, ch_3.[1]

3.2.1. The steps of protocol MANA I

Steps 1–2: A sends D_A to B in channel ch_1. B receives D_B in channel $ch1$. This channel is unprotected. The notation postulates the possibility $D_A \neq D_B$.

Steps 3–4: Device A generates a random key K and computes the check-value $m_K(D_A)$. Hereupon device A sends check-value $m_K(D_A)$ and K to user U using the protected channel ch_2. It means that $A's$ display shows K and $m_K(D_A)$ to U. U receives the message and forwards it to device B in channel ch_3 — U enters in the message keypad of B.

Step 5: Device B recomputes the value $m_K(D_B)$ with the received parameters and compares it with the value of the received $m_K(D_A)$. Let x denote the result of this comparison, so x is $'1'$ if $m_K(D_A) = m_K(D_B)$ and $'0'$ otherwise. Hereupon B sends x to the user U using the protected channel ch_3. It means B uses the LED or in other words U observes the LED of B and receives x. U forwards x to device A in channel ch_2. It means, U uses the binary switch of A.

Step 6: A receives the sent sign x. So A knows the comparison made by B.

3.2.2. Initial assumptions

In this sub-chapter we describe the channel properties and other important properties of the protocol. This is the "Specification of the initial assumptions" part of the proof.

I1. $ENT = \{A, B, U, E, M, \ldots\}$; $ENT_{ch_2} = \{A, U\}$; $ENT_{ch_3} = \{B, U\}$.

I2. $CH(ch_1, pub)$; $CH(ch_2, sec)$; $CH(ch_3, sec)$.

I3. We use the m function and $\forall\, x, y\ \big(m_K(x) = m_K(y) \rightarrow x = y\big)$.

I4. $L_{\Sigma,t}x \wedge L_{\Sigma,t}K \rightarrow L_{\Sigma,t}m_K x$. This means, Σ can use the m function.

I5. $L_{\Sigma,t}x \wedge L_{\Sigma,t}y \rightarrow K_{\Sigma,t}(x = y) \vee K_{\Sigma,t}(x \neq y)$. This means Σ can compare two data strings.

[1] This chapter is the modified version of the lecture on 7th ICAI [13].

I6. $\exists\, t\, \exists\, x\; S(ch_j, i, t, x) \rightarrow \neg\, \big(\exists\, t',\, t' > t\; S(ch_j, i, t', x)\big)$.

I7. $\forall\, t',\, t' > t_3\; R(ch_2, A, t',\, '1') \rightarrow K_{A,t'}(D_A = D_B)$.

I8. $\forall\, t\, \forall\, x_1, x_2\; \big[R(ch_j, \Sigma, t, x_1) \wedge R(ch_j, \Sigma, t, x_2)\big] \rightarrow x_1 = x_2$.

3.2.3. The formal protocol of MANA I

$t_1, \ldots t_{10}$ denote the successive time points in the protocol. We remind the reader that B computes x as we described in Step 5.

1. $S(ch_1, A, t_1, D_A)$; $R(ch_1, B, t_2, D_B)$.
2. $S\big(ch_2, A, t_3, \{K, m_K(D_A)\}\big)$; $R\big(ch_2, U, t_4, \{K, m_K(D_A)\}\big)$.
3. $S\big(ch_3, U, t_5, \{K, m_K(D_A)\}\big)$; $R\big(ch_3, B, t_6, \{K, m_K(D_A)\}\big)$.
4. $S(ch_3, B, t_7, x)$; $R(ch_3, U, t_8, x)$.
5. $S(ch_2, U, t_9, x)$; $R(ch_2, A, t_{10}, x)$.

3.2.4. Protocol goal — Theorem and proof

Now we can state and prove the following theorem for MANA I protocol.

Theorem 1. *At the end of protocol MANA I, both A and B know whether $D_A = D_B$, or not.*

$$D_A = D_B \rightarrow K_{A,t_{10}}(D_A = D_B) \wedge K_{B,t_{10}}(D_A = D_B),$$

$$D_A \neq D_B \rightarrow K_{A,t_{10}}(D_A \neq D_B) \wedge K_{B,t_{10}}(D_A \neq D_B).$$

P r o o f. Suppose $D_A = D_B$.
In step 1: $L_{A,t_1}(D_A)$, $L_{B,t_2}(D_B)$; in step 2: $L_{A,t_3}(K)$, $L_{A,t_3}\big(m_K(D_A)\big)$, $L_{U,t_4}(K)$, $L_{U,t_4}\big(m_K(D_A)\big)$; in step 3: $L_{B,t_6}(K)$, $L_{U,t_6}\big(m_K(D_A)\big)$ by axioms A5'(a) and A6'(a).

By B's computation method of x, and by $D_A = D_B$, B gets $x = {}'1'$ and by I3, $K_{B,t_6}(D_A = D_B)$. By axiom A3(b), $K_{B,t_{10}}(D_A = D_B)$. By repeated application of A16, finally we can conclude $R(ch_2, A, t_{10}, '1')$, so by $t_{10} > t_7 > t_3$ and I7 we have $K_{A,t_{10}}(D_A = D_B)$.

If we assume $D_A \neq D_B$, then B gets $x = {}'0'$ so $K_{B,t_6}(D_A \neq D_B)$ and by analog reasoning, finally we have $K_{B,t_{10}}(D_A \neq D_B)$. □

3.3. MANA II

This protocol is a simple variant of MANA I. Both devices (A and B) have a display and simple input switch. The main security step is the fourth step in channel ch_3 as we can see it in the analysis in detail.

3.3.1. The steps of protocol MANA II

1. **steps:** A sends D_A to B in channel ch_1. B receives D_B in channel ch_1 (ch_1 is unprotected channel).
2. **steps:** A generates key K and computes $m_K(D_A)$. Hereupon device A sends $\{K, m_K(D_A)\}$ to user U in protected channel ch_2.
3. **steps:** A sends K to B in channel ch_1. B receives K'.
4. **steps:** B computes $m_{K'}(D_B)$ and sends $\{K', m_{K'}(D_B)\}$ to U in protected channel ch_3.
5. **steps:** U compares K and K' and also $m_K(D_A)$ and $m_{K'}(D_B)$. Let x denote the result of this conjunction. Let $x = '1'$ denote the case $K = K'$ and $m_K(D_A) = m_{K'}(D_B)$ and $x = '0'$, otherwise. U sends x to A in channel ch_2.
6. **step:** U sends x to B in channel ch_3.

3.3.2. Initial assumptions

The initial assumptions of protocol MANA II are the following.

I21. $ENT = \{A, B, U, \ldots\}; ENT_{ch_2} = \{A, U\}; ENT_{ch_3} = \{B, U\}$.

I22. $CH(ch_1, pub)$; $CH(ch_2, sec)$; $CH(ch_3, sec)$.

I23. We use the m function and $\forall x, y \, \big(m_K(x) = m_K(y) \rightarrow x = y\big)$.

I24. $L_{\Sigma,t}x \wedge L_{\Sigma,t}K \rightarrow L_{\Sigma,t}m_K x$.

I25. $L_{\Sigma,t}x \wedge L_{\Sigma,t}y \rightarrow K_{\Sigma,t}(x = y) \vee K_{\Sigma,t}(x \neq y)$.

I26. $\exists t \, \exists x \, S(ch_j, i, t, x) \rightarrow \neg \big(\exists t', \, t' > t \, S(ch_j, i, t', x)\big)$.

I27. $\forall t', \, t' > t_5 R(ch_2, A, t', '1') \rightarrow K_{A,t'}(D_A = D_B)$,
$\forall t'', \, t'' > t_7 R(ch_3, B, t'', '1') \rightarrow K_{B,t''}(D_A = D_B)$.

I28. $\forall t \, \forall x_1, x_2 \big[R(ch_j, \Sigma, t, x_1) \wedge R(ch_j, \Sigma, t, x_2)\big] \rightarrow x_1 = x_2$.

3.3.3. The formal protocol of MANA II

$t_1, \ldots t_{12}$ denote the successive time points in the protocol. We remind the reader that B computes x as we described in Step 5.

1. $S(ch_1, A, t_1, D_A)$; $R(ch_1, B, t_2, D_B)$.
2. $S\big(ch_2, A, t_3, \{K, m_K(D_A)\}\big)$; $R\big(ch_2, U, t_4, \{K, m_K(D_A)\}\big)$.
3. $S(ch_1, A, t_5, K)$; $R(ch_1, B, t_6, K')$.
4. $S\big(ch_3, B, t_7, \{K', m_{K'}(D_B)\}\big)$; $R\big(ch_3, U, t_8, \{K', m_{K'}(D_B)\}\big)$.
5. $S(ch_2, U, t_9, x)$; $R(ch_2, A, t_{10}, x)$.
6. $S(ch_3, U, t_{11}, x)$; $R(ch_3, B, t_{12}, x)$.

3.3.4. Protocol goals — Theorems and proofs

Theorem 2. *Suppose that the parameters (D_A, D_B) are not equal. Then at the end of the protocol MANA II both A and B know that $D_A \neq D_B$. Formally,*

$$D_A \neq D_B \to K_{A,t_{12}}(D_A \neq D_B) \wedge K_{B,t_{12}}(D_A \neq D_B).$$

P r o o f. If $D_A \neq D_B$, then in Step 1: $L_{A,t_1}(D_A)$, $L_{B,t_2}(D_B)$; in Step 2: $L_{A,t_3}(K)$, $L_{A,t_3}\big(m_K(D_A)\big)$, $L_{U,t_4}(K)$, $L_{U,t_4}\big(m_K(D_A)\big)$; in Step 3: $L_{B,t_6}(K')$; in Step 4: $L_{B,t_7}\big(m_{K'}(D_B)\big)$, $L_{U,t_8}(K')$, $L_{U,t_8}\big(m_{K'}(D_B)\big)$ by axioms A5'(a) and A6'(a). By U's computation method of x, x gets value $'0'$. So by axiom A16, $R(ch_2, A, t_{10}, '0')$ and $R(ch_3, B, t_{12}, '0')$ and by A3(b) and I27 we have $K_{A,t_{12}}(D_A \neq D_B)$ and $K_{B,t_{12}}(D_A \neq D_B)$. □

Theorem 3. *$D_A = D_B$ does not guarantee that at the end of the protocol MANA II A and B know that $D_A = D_B$.*

P r o o f. If $D_A = D_B$ but $K \neq K'$, then by U's computation method of x, x gets value $'0'$. □

So we stress that protocol MANA II satisfies its goals only partially. If $K = K'$ can be guaranteed, then the missing direction can be verified similarly the previous verifications. But the condition $K = K'$ oversteps the possibilities of protocol. It contains dangers from the point of view of cryptography.

4. Suggestions

We have examined protocols MANA I and II with modal logics tools. We have extended the original CSN system with channel signs, and we have applied the created system.

We have established that protocol MANA I is correct, but protocol MANA II is only partially correct in the sense that the satisfaction of one of its goals is not guaranteed after the execution of the protocol. We suggest the development of the protocol in this direction.

These examined protocols are used in many areas of communication. The number of personal area networking systems grows and these systems expand all over the world. Important application areas are health information systems, business information systems — among others.

We suggest to involve new protocols in this research and, if needed, to extend the axiomatic system in the appropriate way. For example, an interesting question is how to examine the role of the concrete time restrictions (say "wait 10 seconds for the answer").

Another possible research is to build a semantics for the original axiomatic system of Coffey, Saidha and Newe or for the extended version and prove completeness theorems for these semantics.

Acknowledgements. We thank T. Mihálydeák for his valuable advices.

REFERENCES

[1] BELLOVIN, S. M.—MERRITT, M.: *Encrypted key exchange: password-based protocols secure against dictionary attacks,* in: Proc. IEEE Computer Society Symposium, Oakland, USA, May 1992, pp. 72–84.

[2] BELLOVIN, S. M.—MERRITT, M.: *Augmented encrypted key exchange: a password-based protocol secure against dictionary attacks and password file compromise,* in: Proc. of the First ACM Conference on Computer and Communications Security, November, 1993, pp. 244–250.

[3] *Bluetooth core specification,* Version 2.1., 2007.

[4] BURROWS, M.—ABADI, M.—NEEDHAM, R.: *A logic of authentication,* Research Report 39., Digital System Research Center, 1989.

[5] BUTTYÁN, L.: *Formal methods in the design of cyptographyprotocols (State of the Art),* EPFL SSC Technical Report, No. SSC/1999/038, 1999.

[6] ČAGALJ, M.—ČAPKUN, S.—HUBAUX, J-P.: *Key agreement in peer-to-peer wireless networks,* Proc. of the IEEE **94** (2006), 467–478.

[7] COFFEY, T.—SAIDHA, P.: *Logic for verifying public-key cryptographic protocols,* IEE Proc. Comput. Digit. Tech. **144** (1997), 28–32.

[8] GEHRMANN, C.—MITCHELL, C. J.—NYBERG, K.: *Manual authentication for wireless devices,* Cryptobytes **7** (2004), 29–37.

[9] GOEMAN, S. (ED.),: *Specification of prototypes–D11, IST–2000–25350–SHAMAN.* Public Report, 2002, 26–29.

[10] LAUR, S.—ASOKAN, N.—NYBERG, K.: *Efficient mutual data authentication using manually authenticated strings.* Cryptology ePrint Archive: Report 2005/424, 2005.

[11] LAUR, S.—NYBERG, K.: *Efficient mutual data authentication using manually authenticated string: extended version.* Cryptology ePrint Archive, Report 2005/424, 2006. A shorter more compact version was published at CANS 2006.

[12] NEWE, T.—COFFEY, T.: *Formal verification logic for hybrid security protocols,* Comput. Syst. Sci. and Eng. **18** (2003), 17–25.

[13] TAKÁCS, P.: *The extension of CNS-logic for multi-channel protocols,* in: 7th International Conference on Applied Informatics, Eger, Hungary, 2007.

[14] VALKONEN, J.: *Ad-Hoc Security Associations for Wireless Devices.* Masters Thesis, Department of Computer Science and Engineering, Helsinki University of Technology, 2005.

[15] WINDIRSCH, P. (ED.) : *Security for mobile systems beyond 3G,* Presentations and Posters of the IST - 2000 - 25350 - SHAMAN WorShop, 2002.

[16] WONG, F-L.—STAJANO, F.: *Multi-channel protocols,* in: Proc. of Security Protocols, 13th International Workshop, Cambridge, UK, April 20–22, 2005, Lecture Notes in Comput. Sci., Vol. 4631, Springer-Verlag, Berlin, 2007.

APPENDIX

The language

CSN logic is a first-order modal logic having the following syntactic resources and notations.

- **a, b, c,** ... – general propositional variables;
- Φ – an arbitrary statement;
- Σ, Ψ – arbitrary entities; **ENT** the set of all possible entities in the system;
- **i, j** – range over entities;
- **K** – is Hintikka's propositional knowledge operator,
  $K_{\Sigma,t}\Phi$ means: Σ knows statement Φ at time t,
- **L** – knowledge predicate,
  $L_{\Sigma,t}x$ means: Σ knows and can reproduce object x at time t,
- **B** – belief operator,
  $B_{\Sigma,t}\Phi$ means: entity Σ believes at time t that statement Φ is true;
- k – a cryptographic (public) key, k_Σ is the public key of entity Σ;
- k^{-1} – a cryptographic (secret, private) key, k_Σ^{-1} is the private key of entity Σ;
- $t_1, t_2, \ldots$ – notation of time;
- **e()** – encryption function,
  $e(x, k_\Sigma)$ means: encryption of x using public-key k_Σ;
- **d()** – decryption function,
  $d(x, k_\Sigma^{-1})$ means: decryption of x using the corresponding private-key k_Σ^{-1}, and this function still means: signing of x;
- **S** – emission operator, $S(\Sigma, t, x)$ means: Σ sends message x at time t;
- **R** – reception operator, $R(\Sigma, t, x)$ means: Σ receives message x at time t;
- **C** – 'contains' operator, $C(x, y)$ means: object x contains the object y
- **ks** – shared secret key, $ks_{(\Sigma,\Psi)}$ shared secret key for entities Σ and Ψ;
- **KS** – set of good keys, $KS_{(\Sigma,\Psi)}$ set of good shared keys for entities Σ and Ψ;
- **ss** – shared secret, $ss_{(\Sigma,\Psi)}$ shared secret for entities Σ and Ψ (it can be fresh);
- **SS** – set of good shared secrets,
  $SS_{(\Sigma,\Psi)}$ set of good shared secrets for entities Σ and Ψ;
- **E** – encryption function, $E(x, ks_{\Sigma,\Psi})$ encryption of plaintext message x using the shared secret key of entities Σ and Ψ;
- **D** – decryption function, $D(x, ks_{\Sigma,\Psi})$ decryption of ciphertext message x using the shared secret key of entities Σ and Ψ;
- **A** – authentication operator,
  $A(\Sigma, t, \Psi)$ means: Σ authenticates Ψ at time t.

We follow [7, 12] in the usage of the following logical signs.

Standard logical signs: $\wedge$ – conjunction; $\vee$ – disjunction; $\neg$ – complementation; $\rightarrow$ – implication; $\exists$ – existential quantification; $\forall$ – universal quantification; ϵ – membership of a set; $/$ – set exclusion; $\vdash$ – logical theorem; $\neg$ – negation. Free variables are implicitly quantified with universal quantifiers in the CSN axioms and inference rules.

Inference rules

We recall the following inference rules from [7, 12].

R1 from $\vdash p$ and $\vdash (p \rightarrow q)$ infer $\vdash q$ (Modus ponens).
R2(a) from $\vdash p$ infer $K_{\Sigma,t}p$ (Generalisation rule I).
R2(b) from $\vdash p$ infer $B_{\Sigma,t}p$ (Generalisation rule II).
R3 from $(p \wedge q)$ infer p.
R4 from p and q infer $(p \wedge q)$.
R5 from p infer $(p \vee q)$.
R6 from $\neg(\neg p)$ infer p.
R7 from (from p infer q) infer $(p \rightarrow q)$.

Axioms

Papers [7, 12] fix the following axioms. We call it the CSN axiomatic system. We repeat that free variables are meant as universally quantified.

A1(a) $K_{\Sigma,t}p \wedge K_{\Sigma,t}(p \rightarrow q) \rightarrow K_{\Sigma,t}q$,
application of the modus ponens to the knowledge operator.

A1(b) $B_{\Sigma,t}p \wedge B_{\Sigma,t}(p \rightarrow q) \rightarrow B_{\Sigma,t}q$,
application of the modus ponens to the belief operator.

A2(a) *knowledge axiom*
$K_{\Sigma,t}p \rightarrow p$,
if something is known, then it is true;
this property distinguishes between the K operator from the B operator.

A3(a) *monotonicity of knowledge*
$L_{i,t}x \rightarrow \forall\, t',\ t' \geq t\ \ L_{i,t'}x$,
knowledge once gained cannot be lost.

A3(b) *monotonicity of knowledge*
$K_{i,t}x \rightarrow \forall\, t',\ t' \geq t\ \ K_{i,t'}x$,
knowledge once gained cannot be lost.

A3(c) *monotonicity of belief*
$B_{i,t}x \to \forall\, t',\ t' \geq t\ \ B_{i,t'}x$,
belief once gained cannot be lost.

A4(a) $L_{i,t}y \wedge C(y,x) \to \exists\, j,\ j \in \{ENT\}\ L_{j,t}x$,
if piece of data is constructed from other pieces of data, then each piece of data involved in the construction must be known to some entity.

A5(a) *emission axiom*
$S(\Sigma,t,x) \to L_{\Sigma,t}x \wedge \exists\, i, i \in \{ENT/\Sigma\}\ \exists\, t',\ t' > t\ \ R(i,t',x)$,
if Σ sends a message x at time t, then Σ knows x at time t and some entity i other than Σ will receive x at time t' subsequent to t.

A6(a) *reception axiom*
$R(\Sigma,t,x) \to L_{\Sigma,t}x \wedge \exists\, i,\ i \in \{ENT/\Sigma\}\ \exists\, t',\ t' < t\ \ S(i,t',x)$,
if Σ receives a message x at time t, then Σ knows x at time t and some entity i other than Σ has sent x at time t' prior to t.

A7(a) $L_{i,t}x \wedge L_{i,t}k_\Sigma \to L_{i,t}\big(e(x,k_\Sigma)\big)$,
the ability of an entity to *encrypt a message* when it has knowledge of a public cryptographic key.

A7(b) $L_{i,t}x \wedge L_{i,t}k_\Sigma^{-1} \to L_{i,t}\big(d(x,k_\Sigma^{-1})\big)$,
the ability of an entity to *decrypt a message* when it has knowledge of a private cryptographic key.

A8(a) $\neg\, L_{i,t}k_\Sigma \wedge \forall\, t', t' < t\ \ \neg\, L_{i,t'}\big(e(x,k_\Sigma)\big) \wedge$
$\neg\, \big(\exists\, y(R(i,t,y) \wedge C(y,e(x,k_\Sigma)))\big) \to \neg\, L_{i,t}\big(e(x,k_\Sigma)\big)$,
the impossibility of encrypting a message without knowledge of the correct key; if an entity does not know k_Σ at t and does not know, prior to t the encryption $e(x,k_\Sigma)$ and also does not receive $e(x,k_\Sigma)$ at t in a message, then the entity cannot know $e(x,k_\Sigma)$ at time t.

A8(b) $\neg\, L_{i,t}k_\Sigma^{-1} \wedge \forall\, t', t' < t\ \neg\, L_{i,t'}\big(d(x,k_\Sigma^{-1})\big) \wedge \neg\, \big(\exists\, y(R(i,t,y)$
$\wedge\, C(y,d(x,k_\Sigma^{-1})))\big) \to \neg\, L_{i,t}\big(d(x,k_\Sigma^{-1})\big)$,
the impossibility of decrypting a message without knowledge of the correct key; if an entity does not know k_Σ^{-1} at t and does not know, prior to t the decryption $d(x,k_\Sigma^{-1})$ and also does not receive $d(x,k_\Sigma^{-1})$ at t in a message, then the entity cannot know $d(x,k_\Sigma^{-1})$ at time t.

A9(a) *key secrecy axiom*
$\forall\, t \forall\, i\big(\ L_{i,t}k_i^{-1} \wedge \forall\, j,\ j \in \{ENT/i\}\ \ \neg\, L_{j,t}k_i^{-1}\big)$,
the private keys used by the system are known only to their rightful owners.

A10(a) $L_{i,t}\big(d(x, k_{\Sigma}^{-1})\big) \rightarrow L_{\Sigma,t}x$,
a private key owner must know any data which have been decrypted using that private key.

A11(a) $L_{i,t}x \wedge L_{i,t}ks_{(\Sigma,\Psi)} \rightarrow L_{i,t}\big(E(x, ks_{(\Sigma,\Psi)})\big)$,
the ability an entity has to encrypt a message using a symmetric system, when it has knowledge of a secret key; if some entity i knows and can reproduce x at time t and i knows and can reproduce the shared secret key of entities Σ and Ψ at time t, then i can encrypt x using the shared secret key of Σ and Ψ at time t.

A11(b) $L_{i,t}x \wedge L_{i,t}ks_{(\Sigma,\Psi)} \rightarrow L_{i,t}\big(D(x, ks_{(\Sigma,\Psi)})\big)$,
the ability an entity has to decrypt a message using a symmetric system, when it has knowledge of a secret key.

A12(a) $\big(\neg L_{i,t}ks_{(\Sigma,\Psi)} \wedge \forall t',\, t' < t \neg L_{i,t'}(E(x, ks_{(\Sigma,\Psi)})) \wedge$
$\neg\big(\exists y(R(i,t,y) \wedge C(y, E(x, ks_{(\Sigma,\Psi)})))\big) \rightarrow \neg L_{i,t}\big(E(x, ks_{(\Sigma,\Psi)})\big)\big)$,
if entity i does not know $ks_{(\Sigma,\Psi)}$ at t and does not know prior to t the encryption $E(x, ks_{(\Sigma,\Psi)})$ and also does not receive a message containing $E(x, ks_{(\Sigma,\Psi)})$ at t, then i does not know $E(x, ks_{(\Sigma,\Psi)})$ at t.

A12(b) $\big(\neg L_{i,t}ks_{(\Sigma,\Psi)} \wedge \forall t',\, t' < t \neg\, L_{i,t'}(D(x, ks_{(\Sigma,\Psi)})) \wedge$
$\neg\,(\exists\, y(R(i,t,y) \wedge C(y, D(x, ks_{(\Sigma,\Psi)})))) \rightarrow \neg\, L_{i,t}(D(x, ks_{(\Sigma,\Psi)}))\big)$,
if entity i does not know $ks_{(\Sigma,\Psi)}$ at t and does not know prior to t the decryption $D\big(x, ks_{(\Sigma,\Psi)}\big)$ and also does not receive a message containing $D\big(x, ks_{(\Sigma,\Psi)}\big)$ at t, then i does not know $D(x, ks_{(\Sigma,\Psi)})$ at t.

A13(a) $\forall t, \big((\forall i, i \in \{ENT/\Sigma, \Psi\} \neg L_{i,t}ks_{(\Sigma,\Psi)} \wedge$
$\exists j, j \in \{\Sigma, \Psi\}\ L_{j,t}ks_{(\Sigma,\Psi)}) \rightarrow ks_{(\Sigma,\Psi)} \in \{KS_{(\Sigma,\Psi)}\}\big)$,
that only the rightful owners of a shared secret key know this key, this implies that this key is a good key.

A14(a) $\forall t\big((\forall i, i \in \{ENT/\Sigma, \Psi\} \neg L_{i,t}ss_{(\Sigma,\Psi)} \wedge \exists j, j \in \{\Sigma, \Psi\} L_{j,t}ss_{(\Sigma,\Psi)})$
$\rightarrow ss_{(\Sigma,\Psi)} \in \{SS_{\{\Sigma,\Psi\}}\}\big)$,
only the rightful owners of a shared secret know this secret, this implies that this is a good secret.

A15(a) *authentication axiom — symmetric form*
$\big(A(\Sigma, t, \Psi) \rightarrow (L_{\Sigma,t}ss_{(\Sigma,\Psi)} \wedge ss_{(\Sigma,\Psi)} \in \{SS_{\{\Sigma,\Psi\}}\} \wedge R(\Sigma, t, x))$
$\wedge C(x, ss_{(\Sigma,\Psi)}) \wedge \forall t',\, t' < t \neg S(\Sigma, t', x)\big) \rightarrow K_{\Sigma,t}\big(S(\Psi, t', x)\big)$,
if Σ knows a secret $ss_{(\Sigma,\Psi)}$ that it shares with Ψ (the secret can be fresh), and this secret is a good secret, and Σ receives a message containing $ss_{(\Sigma,\Psi)}$ at t that it did not send, then Σ knows that Ψ sent this message prior to t.

A15(b) *authentication axiom — asymmetric form*
$\big(A(\Sigma,t,\Psi) \to (L_{\Sigma,t}k_\Psi \wedge L_{\Sigma,t}x \wedge R(\Sigma,t,y) \wedge C(y,e(x,k_{\Psi^{-1}})))$
$\to (\forall\, t',\, t' < t, K_{\Sigma,t}(S(\Sigma,t',y)))\big)$,
if Σ knows the public key of $\Psi_{(k_\Psi)}$ and message x, and if Σ receives a message y containing $e(x,k_{\Psi^{-1}})$, then Σ knows that Ψ sent message y prior to t.

Received October 1, 2007

Péter Takács
Department of Health Informatics
Faculty of Health
University of Debrecen
Sóstói út. 2–4
H–4400 Nyíregyháza,
HUNGARY
E-mail: vtp@de-efk.hu

Sándor Vályi
Department of Economics and
Agricultural Informatics
Faculty of Agribusiness and Rural Development
University of Debrecen
Böszörményi út. 138.
H-4032 Debrecen
HUNGARY
E-mail: valyis@thor.agr.unideb.hu