

# Title: Formal analysis of 3G authentication and key agreement protocol

**Abstract:** *This contribution presents an analysis of the 3G AKA protocol using an enhanced BAN logic. This enhanced BAN-logic is described in reference [8] of which copies are made available at the meeting for convenience of the delegates. An overview of authentication logics is given in reference [4] which can be downloaded from the web.*

## 1 Introduction

This paper is on the formal analysis of the authentication and key agreement protocol contained in [1, sect.6.3], called the SEQ-protocol in the sequel. This analysis is made using the authentication logic AUTLOG [7], [8], [5], [6], which is an enhanced BAN-logic [2]. The paper proves that the goals of the SEQ-protocol as they are stated in section 3.3. below are met by the protocol.

Authentication logics are one of the most attractive tools to analyse cryptographic protocols. Since their invention in 1989 [2] a lot of work was done on them, both theoretical research and practical applications. For an overview see [4]. Authentication logics investigate how the beliefs of the participants evolve during the exchange of the messages. A formal analysis consists of four steps:

1. The necessary prerequisites are explicitly stated.
2. The messages are formally described.
3. The protocols goals are explicitly stated.
4. The formal calculus is applied to show how the protocols goals can be derived from the messages and the prerequisites using the rules of the calculus.

Note that this method leads to positive statements like the protocol goal “ $A$  believes that  $K$  is a good key for communication with  $B$ ” can be derived. If a protocol goal cannot be derived this may have different reasons, e.g., it could be that some of needed prerequisites are missing, it could also be that there is a failure in the protocol. This method does not explicitly find the failure but it gives the protocol designer or analyser a hint where the failure might be.

In this context we should mention that *all* methods for formal analysing protocols share the feature that none of them is capable to find all mistakes of a protocol, cf. [4]. Each method provides a different view on a given protocol. This means that verifying a protocol with one formal method increases the confidence in a protocol but gives no 100%-guarantee that there is no bug.

The experience with authentication logics is that it is “easy” to apply, at least compared to other formal methods. It helps significantly to understand what a protocol really does because you have to be very precise about the prerequisites and the protocol goals. Especially, you should pay attention to the prerequisites and make sure that they really are fulfilled in the scenario where the protocol will be used.

One critique concerning BAN-logic and most of its dialects has been the fact that the calculus itself is inconsistent. Therefore AUTLOG is based on a formal semantics [8]. It is proven in [8] that the AUTLOG-calculus is correct with respect to that formal semantics. Unfortunately, most of the BAN-dialects lack this sort of justification. Another common critique

concerning BAN-logics refer to the so-called idealization step. Note that this idealization step is not used within AUTLOG [8].

## 2 The SEQ Protocol

There are three parties involved in the protocol: The user  $U$  represented by his USIM; the home environment  $HE$  represented by the authentication centre, and the serving network  $SN$  represented by the visitor location register. For the sake of simplicity we make two assumptions: Firstly, we restrict our analysis to the case that  $HE$  sends one authentication quintuplet at a time. Secondly,  $SN$  stands for the whole set of authorized serving networks. This assumption is reasonable, because the user will not get assurance which  $SN$  he is using. He will only know that  $SN$  is authorized.

There are three security relevant messages:

### Message 1

On request from  $SN$  the home environment  $HE$  generates an authentication quintuplet. This authentication quintuplet uses the following parameters:

- a random number  $r$
- a key  $K$  shared between HE and U
- a sequence number  $SEQ$  which is synchronized between HE and U
- an expected response  $RES = f2(K, r)$  using a MAC-function  $f2$ <sup>1</sup>
- a cipher key  $CK = f3(K, r)$  using a one-way function  $f3$  suitable for key derivation.
- a integrity key  $IK = f4(K, r)$  using a one-way function  $f4$  suitable for key derivation.
- an anonymity key  $Ka = f5(K, r)$  using a one-way function  $f5$  suitable for key derivation.
- authentication token  $AUTN = \{SEQ\}_{Ka}, f1(K, SEQ, r)$  using a MAC-function  $f1$

The following message is sent to  $SN$ :

$$HE \longrightarrow SN : \quad r, RES, CK, IK, AUTN$$

### Message 2

After receipt of this quintuplet  $SN$  forwards to  $U$  the random number  $r$  which serves as a challenge and the authentication token.

$$SN \longrightarrow U : \quad r, AUTN$$

---

<sup>1</sup>We assume that the parameters  $PAR1, \dots, PAR5$  are integrated into the definitions of the functions  $f1, \dots, f5$ , cf. [1, sect.6.3.2, note 4].

### Message 3

After receipt of the challenge  $r$  the user  $U$  computes an expected authentication token  $XAUTN$  and compares this with the received  $AUTN$ . If they are identical  $U$  computes the response  $RES$  and sends it to  $SN$ :

$$U \longrightarrow SN : \quad RES$$

## 3 Formal Analysis

### 3.1 Formalisation of the protocol prerequisites

#### 3.1.1 Prerequisites on $SN$ 's side

$SN$  believes that  $HE$  has jurisdiction concerning keys between  $U$  and  $SN$  and concerning the freshness of these keys.

$$SN \text{ believes } HE \text{ controls } (SN \stackrel{K'}{\longleftrightarrow} U \wedge \text{fresh}(K')) \quad (1)$$

$SN$  believes that if  $HE$  says  $CK$ ,  $IK$  he means that these are good key for communication with  $U$ , i.e.,  $HE$  is regarded as a sort of key server in this case:<sup>2</sup>

$$SN \text{ believes } (HE \text{ says } RES \longrightarrow HE \text{ believes } (SN \stackrel{RES}{\longleftrightarrow} U \wedge \text{fresh}(RES))) \quad (2)$$

$$SN \text{ believes } (HE \text{ says } CK \longrightarrow HE \text{ believes } (SN \stackrel{CK}{\longleftrightarrow} U \wedge \text{fresh}(CK))) \quad (3)$$

$$SN \text{ believes } (HE \text{ says } IK \longrightarrow HE \text{ believes } (SN \stackrel{IK}{\longleftrightarrow} U \wedge \text{fresh}(IK))) \quad (4)$$

It is assumed that the communication path between  $HE$  and  $SN$  is secure and that  $SN$  is able to detect that the message received from  $HE$  is a list comprising five items:

$$SN \text{ believes } HE \text{ says } (r, RES, CK, IK, AUTN) \quad (5)$$

$SN$  is able to identify  $f2(K, r)$  with the expected response  $RES$ :

$$(f2(K, r))_{SN} \equiv RES \quad (6)$$

$SN$  believes that if a user is able to say  $RES$  this user will also have the keys  $CK$  and  $IK$ . This belief is justified by the fact that  $HE$  has connected these items in his message (cf. 5) and  $SN$  trusts  $HE$  to send him correct authentication quintuplets:<sup>3</sup>

$$SN \text{ believes } (U \text{ says } RES \longrightarrow U \text{ has } (CK, IK)) \quad (7)$$

$SN$  believes that he has not generated the following message himself:

$$SN \text{ believes } \neg SN \text{ said } RES \quad (8)$$

---

<sup>2</sup>Firstly,  $SN$  does not know  $K$  and thus does not know the inner structure of  $RES=f2(K, r)$  etc; therefore we simply write  $RES$ . Secondly, in this protocol  $RES$  has the role of a secret to be shared between  $U$  and  $SN$ . The notation  $SN \stackrel{RES}{\longleftrightarrow} U$  refers both to shared key and to shared secrets.

<sup>3</sup> $HE$  can give the corresponding assurance to  $SN$  because  $U$  can only say  $RES$  if  $U$  has  $K$  and  $r$ .  $U$  can then derive  $CK$  and  $IK$ .

### 3.1.2 Prerequisites on $U$ 's side

$U$  has key  $K$ , recognizes it, and believes that it is good key for communication with  $HE$ :

$$U \text{ has } K \quad (9)$$

$$U \text{ recognizes } K \quad (10)$$

$$U \text{ believes } HE \xleftrightarrow{K} U \quad (11)$$

$$U \text{ believes } HE \text{ has } K \quad (12)$$

$U$  believes that the sequence number  $SEQ$  is fresh, i.e. not used in an earlier protocol run. This belief is justified because  $U$  is able to check whether  $SEQ$  is fresh.

$$U \text{ believes fresh}(SEQ) \quad (13)$$

$U$  regards the following messages as atomic messages

$$(X)_U \equiv X \quad \forall X \in \{SEQ, r, K\} \quad (14)$$

$U$  believes that he has not generated the following messages himself:

$$U \text{ believes } \neg U \text{ said } f1(K, r) \quad (15)$$

$U$  believes that  $HE$  has jurisdiction concerning keys between  $U$  and  $SN$  and  $U$  believes that if  $HE$  says  $r$  he means that the derived keys  $CK = f3(K, r)$  and  $IK = f4(K, r)$  are good keys for communication with  $SN$ :

$$U \text{ believes } HE \text{ controls } SN \xleftrightarrow{K'} U \quad (16)$$

$$U \text{ believes } (HE \text{ says } r \longrightarrow HE \text{ believes } SN \xleftrightarrow{f_i(K, r)} U) \quad \text{for } i = 3, 4 \quad (17)$$

$U$  believes that  $HE$  has jurisdiction about the freshness of random numbers, and  $U$  believes that if  $HE$  says  $r$  he means that this number is fresh:

$$U \text{ believes } HE \text{ controls fresh}(r) \quad (18)$$

$$U \text{ believes } (HE \text{ says } r \longrightarrow HE \text{ believes fresh}(r)) \quad (19)$$

### 3.1.3 Prerequisites on $HE$ 's side

$HE$  has a key  $K$  and believes that  $K$  is a good key for communication with  $U$ :

$$HE \text{ has } K \quad (20)$$

$$HE \text{ believes } HE \xleftrightarrow{K} U \quad (21)$$

$HE$  believes that the random number  $r$  is good for deriving of shared keys (resp. shared secrets):

$$HE \text{ believes good}_{f_i}(r) \quad \text{for } i = 2, \dots, 5 \quad (22)$$

### 3.2 Formal Descriptions of the Transactions

$$SN \text{ sees } r, f2(K, r), f3(K, r), f4(K, r), \{SEQ\}_{f5(K, r)}, f1(K, SEQ, r) \quad (23)$$

$$U \text{ sees } r, \{SEQ\}_{f5(K, r)}, f1(K, SEQ, r) \quad (24)$$

$$SN \text{ sees } f2(K, r) \quad (25)$$

### 3.3 Security goals

The following goals should be achieved after a successful run of the protocol:

1. Entity authentication of  $U$  to  $SN$  and of  $HE$  to  $U$ :

- a.  $SN \text{ believes } U \text{ says } RES$
- b.  $U \text{ believes } HE \text{ says } (SEQ, r)$

2. Key agreement between  $U$  and  $SN$ :

- a.  $SN \text{ has } CK, \quad SN \text{ has } IK$
- b.  $U \text{ has } f3(K, r), \quad U \text{ has } f4(K, r)$

3. Implicit key authentication between  $U$  and  $SN$ :

- a.  $SN \text{ believes } SN \xleftrightarrow{CK} U, \quad SN \text{ believes } SN \xleftrightarrow{IK} U$
- b.  $U \text{ believes } SN \xleftrightarrow{f3(K, r)} U, \quad U \text{ believes } SN \xleftrightarrow{f4(K, r)} U$

4. Assurance of key freshness between  $U$  and  $SN$ :

- a.  $SN \text{ believes } fresh(CK), \quad SN \text{ believe } fresh(IK)$
- b.  $U \text{ believes } fresh(f3(K, r)), \quad U \text{ believes } fresh(f4(K, r))$

5. Key confirmation between  $U$  and the network:

- a.  $SN \text{ believes } U \text{ has } (CK, IK)$
- b.  $U \text{ believes } HE \text{ has } (f3(K, r), f4(K, r))$

6. Confidentiality of the user identity related information (i.e. sequence number  $SEQ$ ) on the air interface: Note that the achievement of confidentiality goals cannot be proven by authentication logics in an explicit manner. The confidentiality follows from the following fact:

$$HE \text{ believes } HE \xleftrightarrow{f5(K, r)} U$$

### 3.4 Proving the security goals

#### 3.4.1 Security goals refering to $SN$

Concerning the notation: The number in front of the implication arrow shows which formulae lead to the implication, the symbols above the arrow refer to the rule of the calculus [8, sect. 4] which has been used for this implication. Since the rules  $MP$  (modus ponens) and  $K$  (rationality rule) are used very often we do not always mention it.

Receiving the first message 23 it follows directly by rule H1:

$$23 \xrightarrow{H1} \boxed{SN \text{ has } (CK, IK)} \text{ (goal 2.a)} \quad (26)$$

From 5 and prerequisite 2 it follows

$$5, 2 \xrightarrow{MP} SN \text{ believes } HE \text{ believes } (SN \xleftrightarrow{RES} U \wedge \text{fresh}(RES)) \quad (27)$$

$$1, 27 \xrightarrow{J} SN \text{ believes } (SN \xleftrightarrow{RES} U \wedge \text{fresh}(RES)) \quad (28)$$

Analogously we can prove from 5 and the prerequisites 1, 3, and 4 the goals key agreement, implicit key authentication and assurance on freshness on  $SN$ 's side:

$$\boxed{SN \text{ believes } SN \xleftrightarrow{CK} U, \quad SN \text{ believes } SN \xleftrightarrow{IK} U} \text{ (goal 3.a)} \quad (29)$$

$$\boxed{SN \text{ believes } \text{fresh}(CK), \quad SN \text{ believes } \text{fresh}(IK)} \text{ (goal 4.a)} \quad (30)$$

From the third message  $SN$  derives:

$$25, 6 \xrightarrow{C} SN \text{ believes } SN \text{ sees } RES \quad (31)$$

$$31, 28, 8 \xrightarrow{A1, K} SN \text{ believes } U \text{ said } RES \quad (32)$$

$$28, 32 \xrightarrow{NV, K} \boxed{SN \text{ believes } U \text{ says } RES} \text{ (goal 1.a)} \quad (33)$$

$$7, 33 \xrightarrow{MP, K} \boxed{SN \text{ believes } U \text{ has } (CK, IK)} \text{ (goal 5.a)} \quad (34)$$

#### 3.4.2 Security goals refering to $U$

$$24 \xrightarrow{H1} U \text{ has } r \quad (35)$$

$$9, 35 \xrightarrow{H2} U \text{ has } (K, r) \quad (36)$$

$$36 \xrightarrow{H3} \boxed{U \text{ has } fi(K, r) \text{ for } i = 2, \dots, 5} \text{ (goal 2.b)} \quad (37)$$

$$24, 37 \xrightarrow{H1, H3} U \text{ has } SEQ \quad (38)$$

$$36, 38 \xrightarrow{H2, C3} (f1(K, SEQ, r))_U \equiv f1((K, SEQ, r)_U) \quad (39)$$

$$10 \xrightarrow{C1} (K, SEQ, r)_U \equiv (K_U, SEQ_U, r_U) \quad (40)$$

$$14, 39, 40 \xrightarrow{E2, E3} (f1(K, SEQ, r))_U \equiv f1(K, SEQ, r) \quad (41)$$

$$24, 41 \xrightarrow{C} U \text{ believes } U \text{ sees } f1(K, SEQ, r) \quad (42)$$

$$42, 11, 15 \xrightarrow{A1} U \text{ believes } HE \text{ said } (SEQ, r) \quad (43)$$

$$13 \xrightarrow{F1} U \text{ believes } fresh(SEQ, r) \quad (44)$$

$$43, 44 \xrightarrow{NV} \boxed{U \text{ believes } HE \text{ says } (SEQ, r)} \text{ (goal 1.b)} \quad (45)$$

$$12, 45 \xrightarrow{H1, H2} U \text{ believes } HE \text{ has } (K, r) \quad (46)$$

$$46 \xrightarrow{H3} \boxed{U \text{ believes } HE \text{ has } fi(K, r) \text{ for } i = 3, 4} \text{ (goal 5.b)} \quad (47)$$

$$17, 45 \xrightarrow{K} U \text{ believes } (HE \text{ believes } SN \stackrel{fi(K, r)}{\leftrightarrow} U) \text{ for } i = 3, 4 \quad (48)$$

$$16, 48 \xrightarrow{J} \boxed{U \text{ believes } SN \stackrel{fi(K, r)}{\leftrightarrow} U \text{ for } i = 3, 4} \text{ (goal 3.b)} \quad (49)$$

$$19 \xrightarrow{MP} U \text{ believes } HE \text{ believes } fresh(r) \quad (50)$$

$$18, 50 \xrightarrow{J} U \text{ believes } fresh(r) \quad (51)$$

$$51 \xrightarrow{F2} \boxed{U \text{ believes } fresh(fi(K, r)) \text{ for } i = 3, 4} \text{ (goal 4.b)} \quad (52)$$

### 3.4.3 Security goals referring to HE

$$21, 22 \xrightarrow{KD} \boxed{HE \text{ believes } HE \stackrel{f5(K, r)}{\leftrightarrow} U} \text{ (goal 6)} \quad (53)$$

## 4 Comments

Assurance of key's freshness on the user's side: The user  $U$  trusts that  $HE$  has chosen a fresh random number  $r$  together with a fresh  $SEQ$  (formula 18). If one does not want to make this reasonable assumption (cf. [3]) then  $SEQ$  could be included in the computation of  $CK = f3(K, SEQ, r)$  and  $IK = f4(K, SEQ, r)$ . Then  $U$  could convince himself that the derived keys are fresh because he can control whether  $SEQ$  is fresh.

## References

- [1] 3GPP S3.03 VO.1.2 (1999-03) *3G Security Architecture*.
- [2] M. Burrows, M. Abadi, R. Needham. *A logic for authentication*. DEC System Research Technical Report No 39, Feb 1989.
- [3] ETSI TDoc SMG 10 99C013, "Mutual authentication and key establishment for UMTS Phase 1 based on random challenges and sequence numbers".
- [4] Stefanos Gritzalis, Diomidis Spinellis, Paagiots Georgiadis. *Security Protocols over Open Networks and Distributed Systems: Formal Methods for their Analysis, Design, and Verification*. Computer Communications 1999, <http://kerkis.math.aegean.gr/~dspin/pubs/jrnl/1997-CompComm-Formal/html/formal.htm>.

- [5] Volker Kessler, Heike Neumann. *A Sound Logic for Analysing Electronic Commerce Protocols*. Computer Security - ESORICS 98, Springer LNCS 1485, 345-360.
- [6] Heike Neumann, Volker Kessler. *Formale Analyse von kryptographischen Protokollen mit BAN-Logik*. Datenschutz und Datensicherheit 2/1999, 90-93.
- [7] Gabriele Wedel. *Formale Semantik für Authentifikationslogiken*. Diplomarbeit, RWTH Aachen (1995).
- [8] Gabriele Wedel, Volker Kessler. *Formal Semantics for Authentication Logics*. Computer Security - ESORICS 96, Springer LNCS 1146, 218-241.