Security of UMTS network and MILENAGE

Seminar Mobile Security

07.02.2011

Supervisor: Prof. Joachim von zur Gathen
Tutor: Yona Raekow, Daniel Loebenberger
Shengkun Fang
Media Informatics

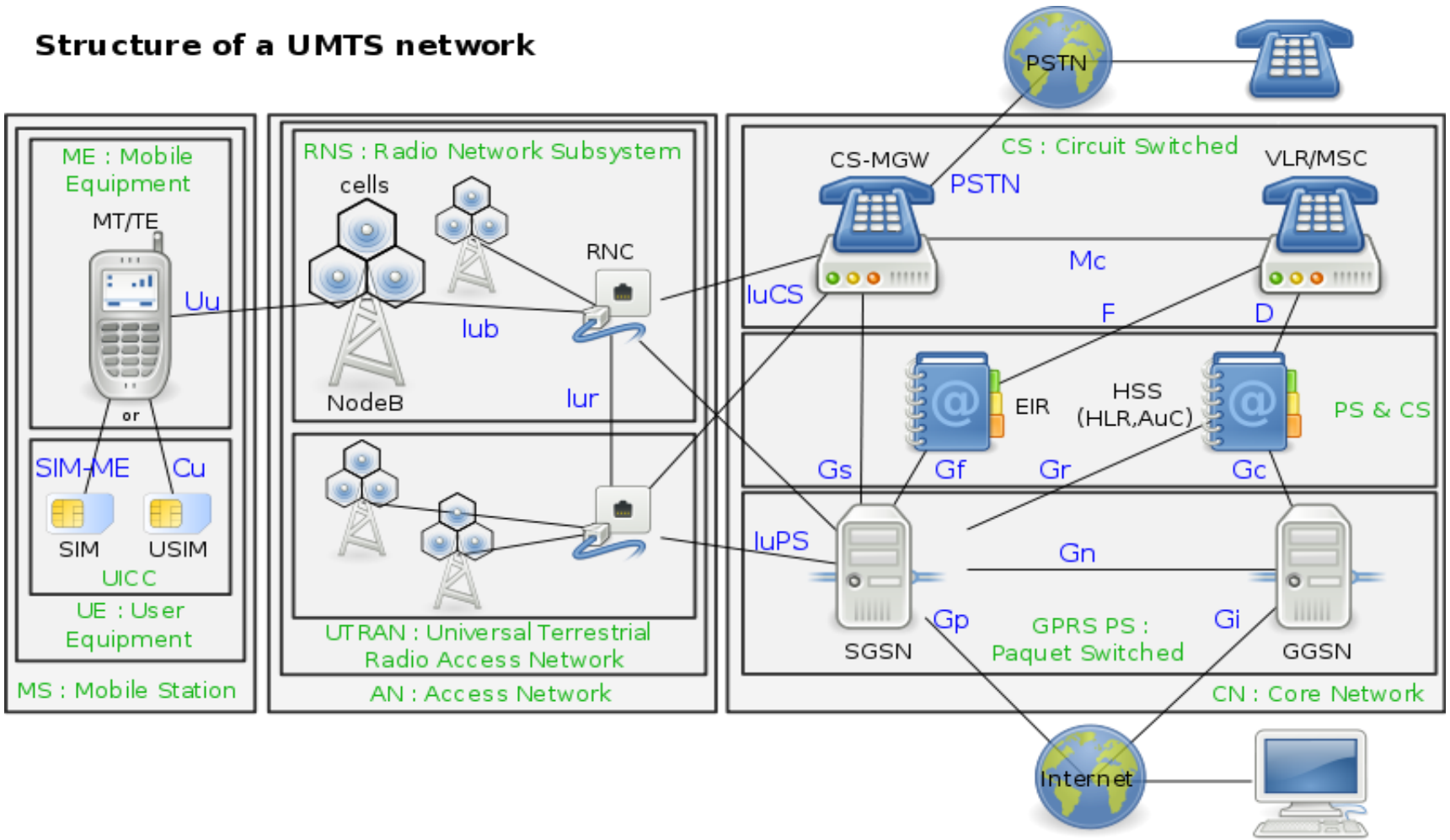
b-it

RWTH AACHEN UNIVERSITY

Agenda

- The UMTS network structure
- UMTS security overview
- KASUMI
- MILENAGE

Structure of UMTS network



What are security features in UMTS

- Features :

  Entity authentication

  User and UE authentication

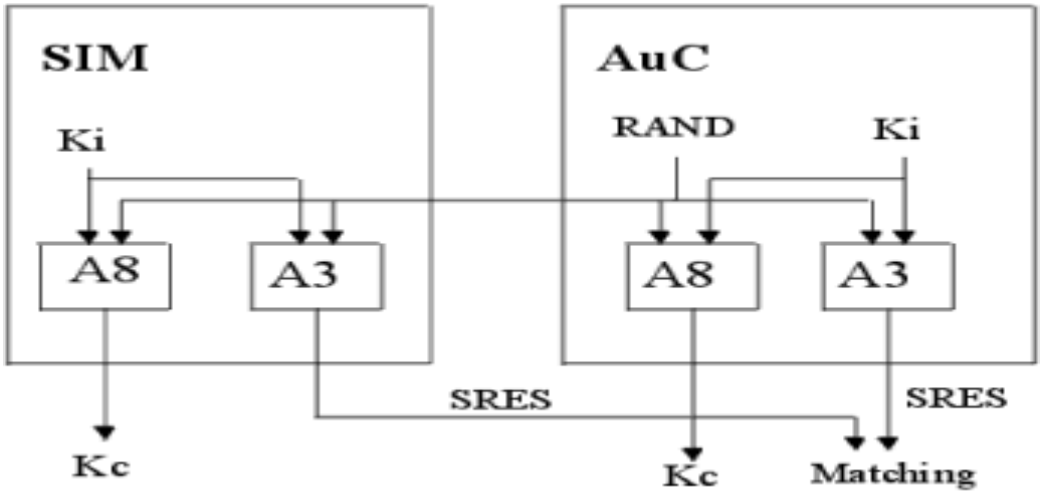
  Traffic confidentiality

  Data integrity

Mobile security

- GSM security

  The GSM adopted the Comp128-1/2/3 algorithm, which is also known as A3 A8 algorithm. The algorithm A5 is used to encrypt the phone call.

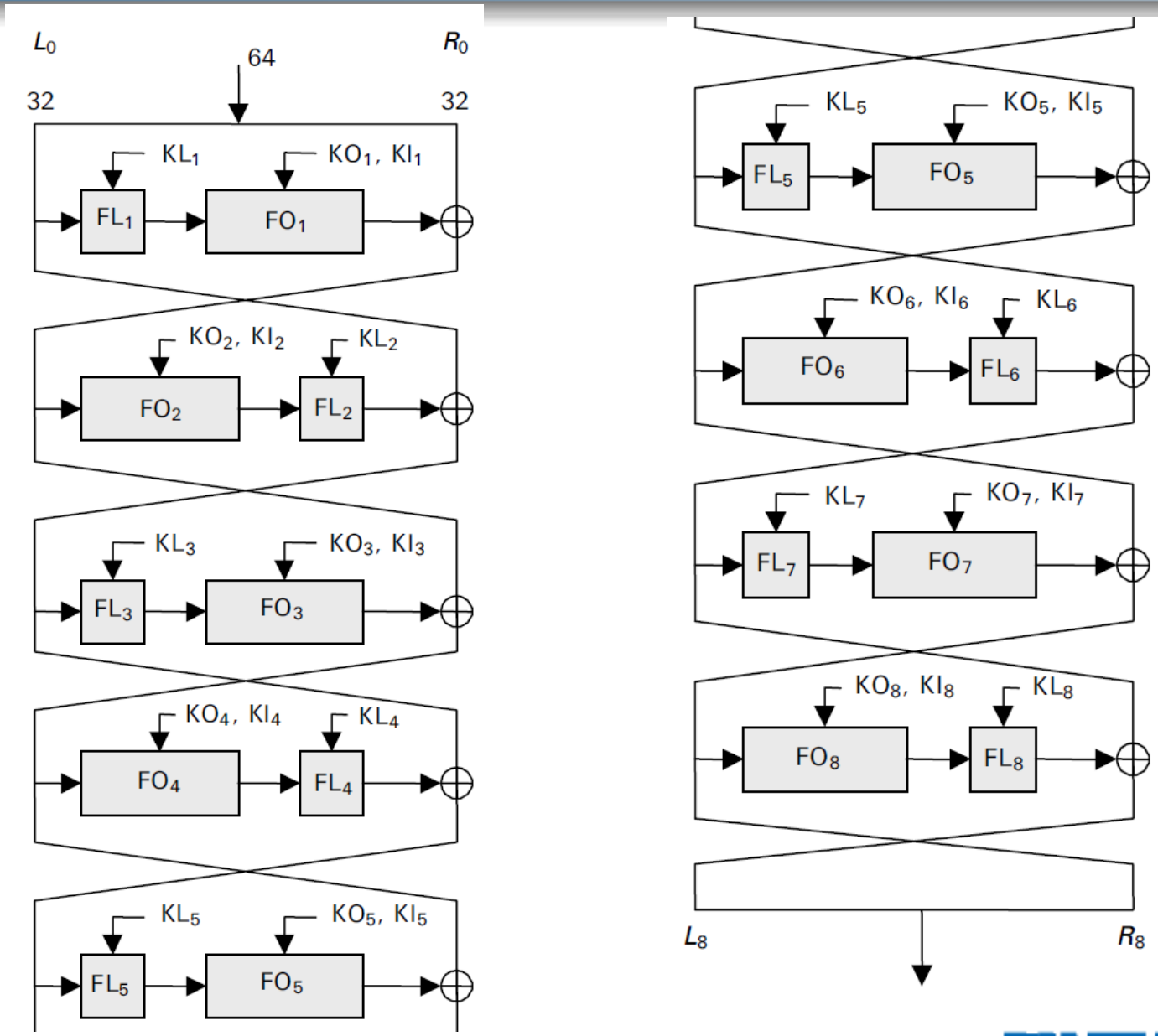


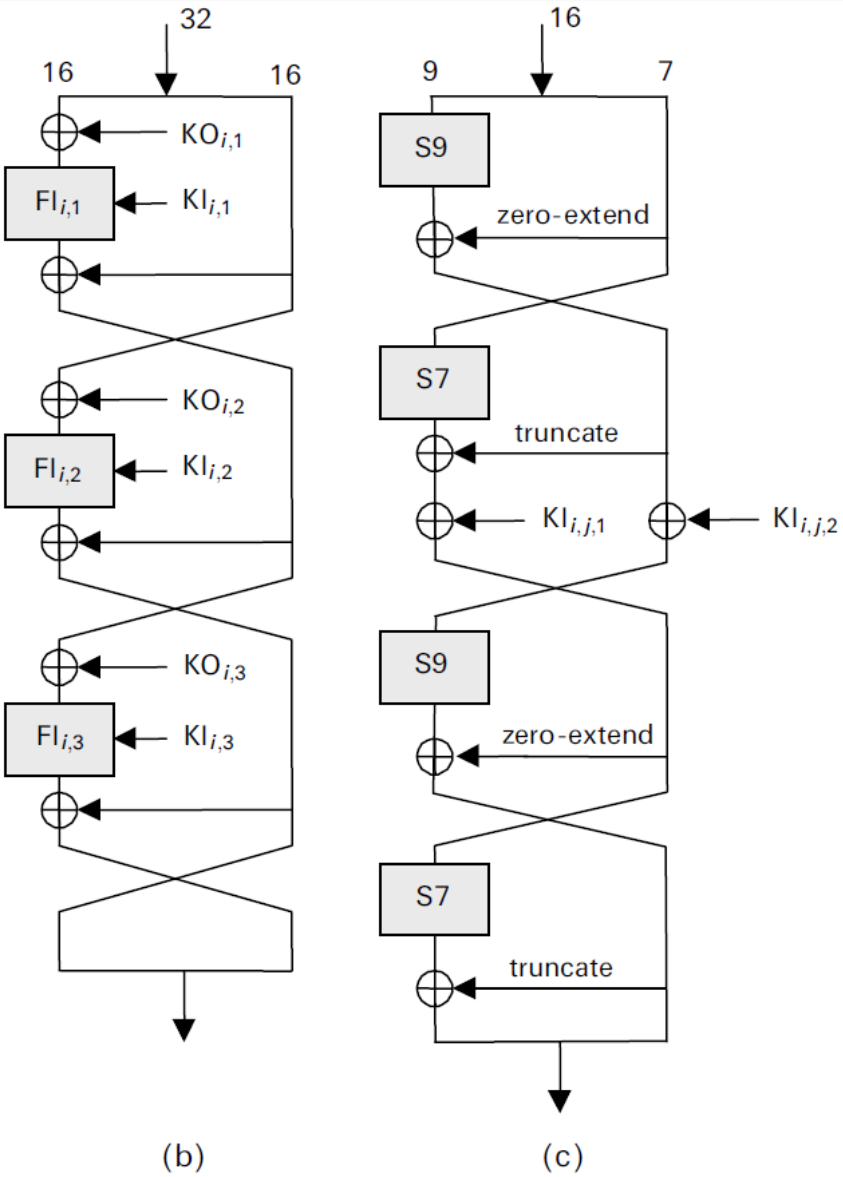
UMTS security

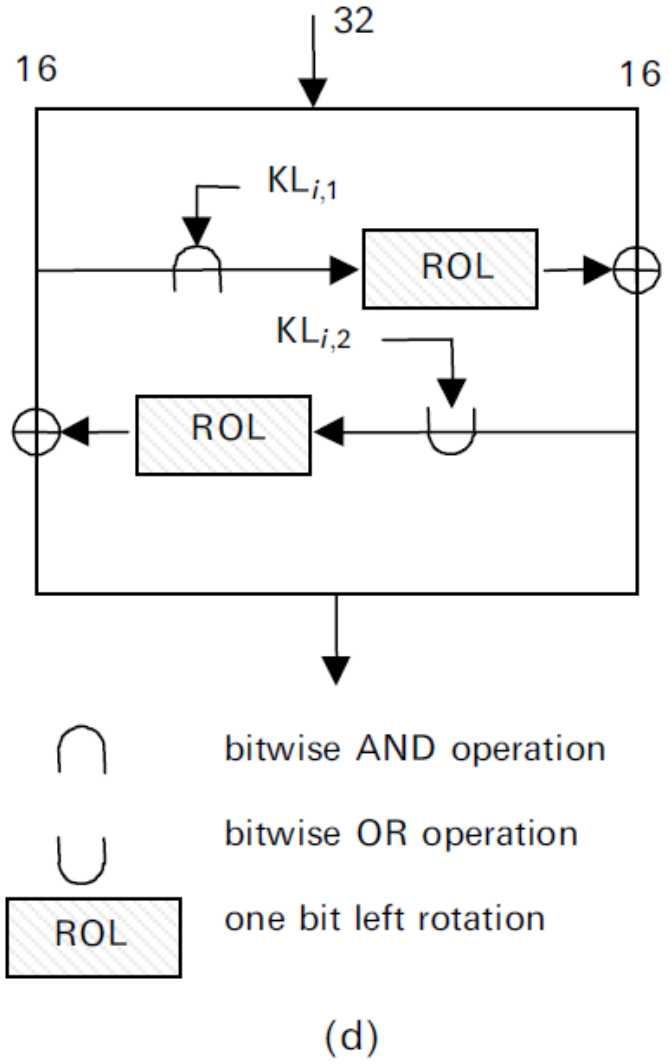
- KASUMI for confidentiality and integrity protection.
- MILENAGE for authentication and key agreement.

KASUMI

- KASUMI applies a 64-bit block with a & 128-bit key.
- The process of KASUMI has eight rounds of Feistel diphers. Each round require 32-bit input corresponding with 32-bit output.


L_0
64
R_0
32
32
KL_1
KO_1, KI_1
FL_1
FO_1
KO_2, KI_2
KL_2
FO_2
FL_2
KL_3
KO_3, KI_3
FL_3
FO_3
KO_4, KI_4
KL_4
FO_4
FL_4
KL_5
KO_5, KI_5
FL_5
FO_5
KO_6, KI_6
KL_6
FO_6
FL_6
KL_7
KO_7, KI_7
FL_7
FO_7
KO_8, KI_8
KL_8
FO_8
FL_8
L_8
R_8


32
16
16
$KO_{i,1}$
$FI_{i,1}$
$KI_{i,1}$
$KO_{i,2}$
$FI_{i,2}$
$KI_{i,2}$
$KO_{i,3}$
$FI_{i,3}$
$KI_{i,3}$
(b)
16
9
7
S9
zero-extend
S7
truncate
$KI_{i,j,1}$
$KI_{i,j,2}$
S9
zero-extend
S7
truncate
(c)


32
16
16
$KL_{i,1}$
ROL
$KL_{i,2}$
ROL
bitwise AND operation
bitwise OR operation
ROL
one bit left rotation
(d)

- Key schedule: 16-bit Subkey Ki is derived by subdivision of 128-bit key K

- Encryption function f8 is based on KASUMI
- Security consideration:

  1, Key only attack

  2, Malleability

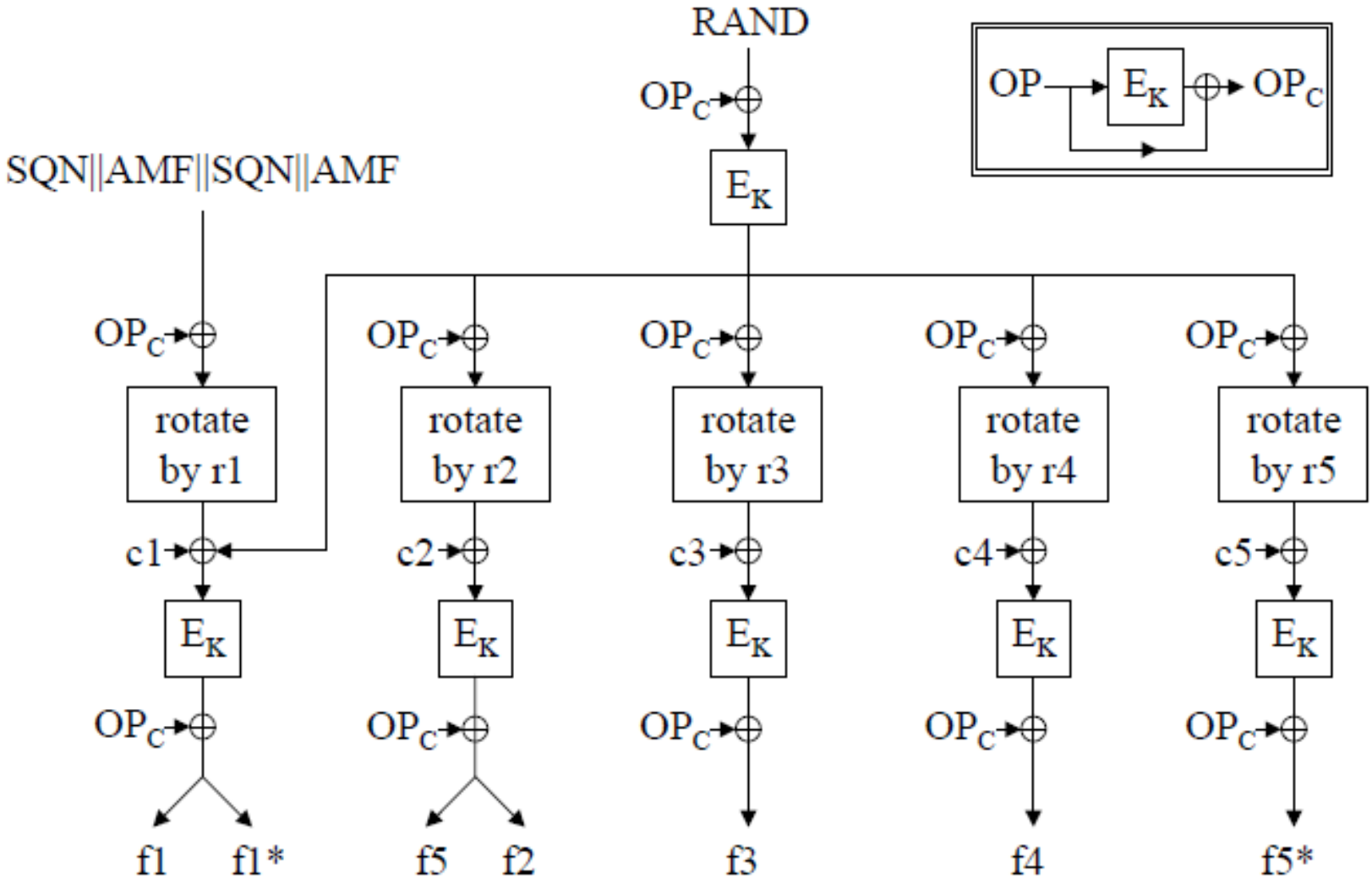
  3, Distinguishability

MILENAGE

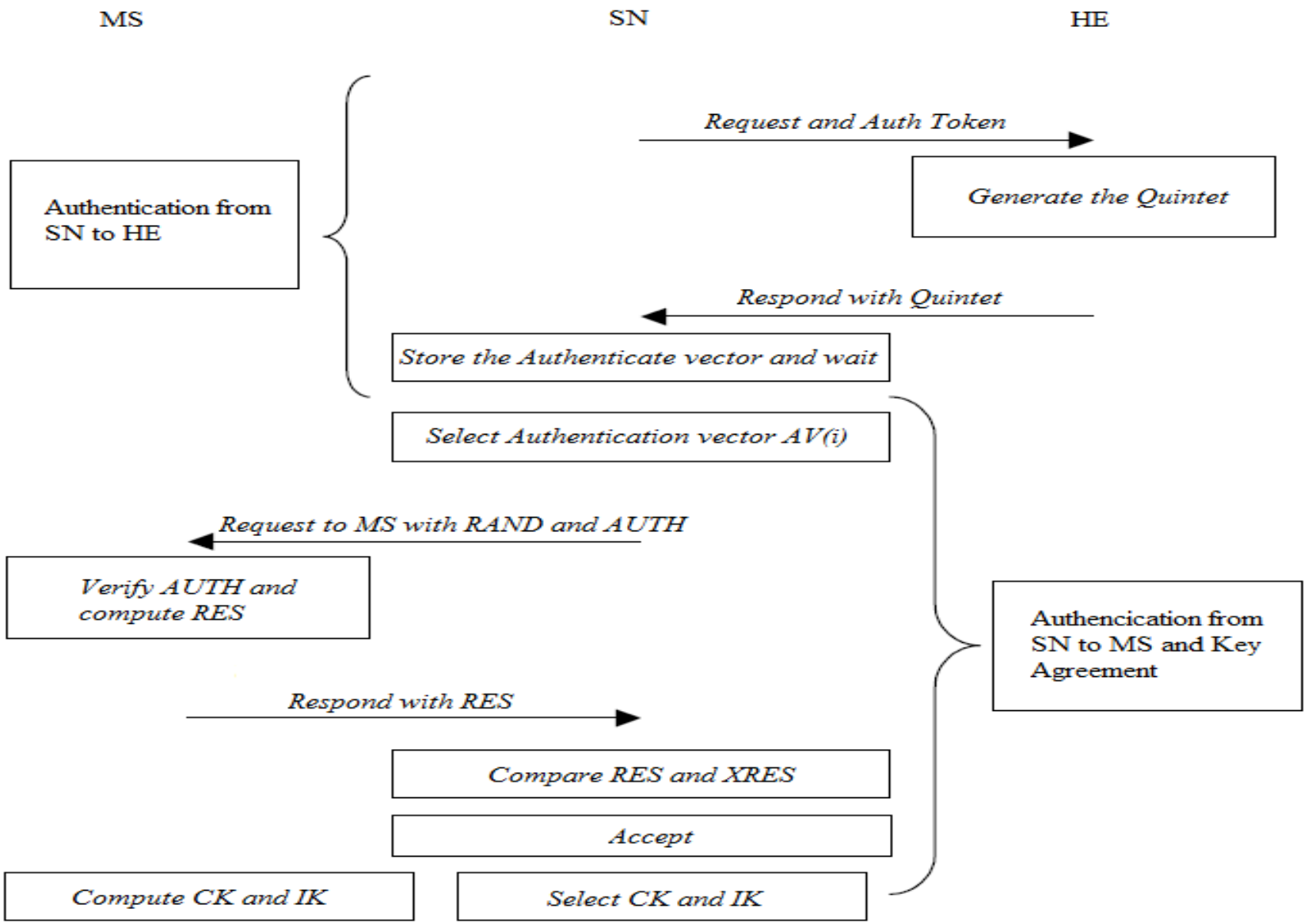
The definition of the seven functions

- f1: computes MAC.
- f1*: computes MAC-S.
- f2: computes RES and XRES.
- f3: generates key CK.
- f4: generates key IK.
- f5: generates key AK.
- f5*: computes AK in re-synchronization procedure.

Algorithm Architecture



Workflow in authentication and key agreement



Implementation consideration

- **OP or OPc in USIM?**
- What is OP?

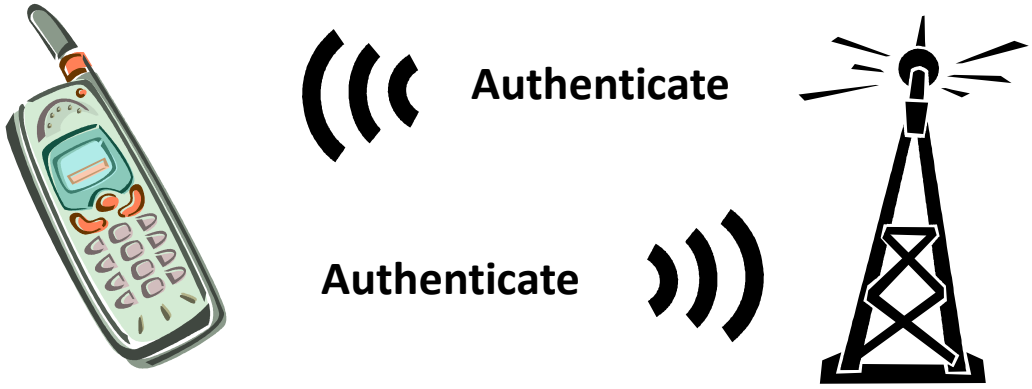
  OP is a 128-bit operator value (Operator Variant Algorithm Configuration Field).

  $OPc = OP \oplus EK(OP)$.
- Better choice:

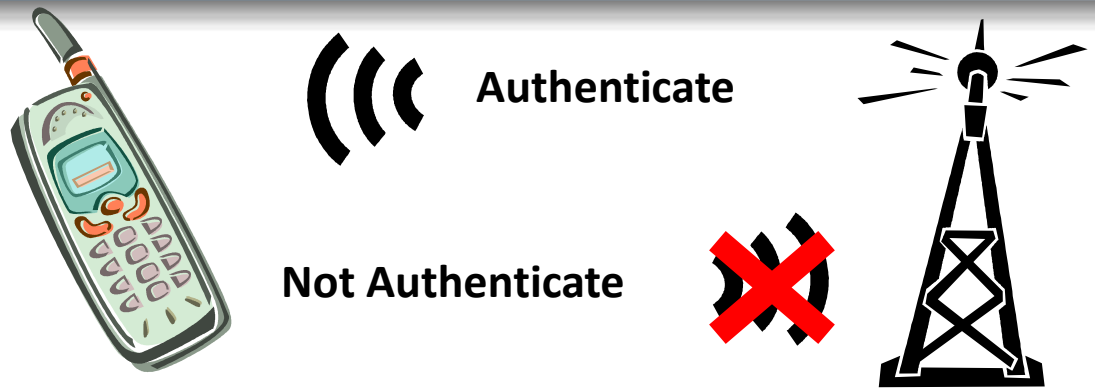
  Store OPc in USIM

Authentication in GSM and UMTS

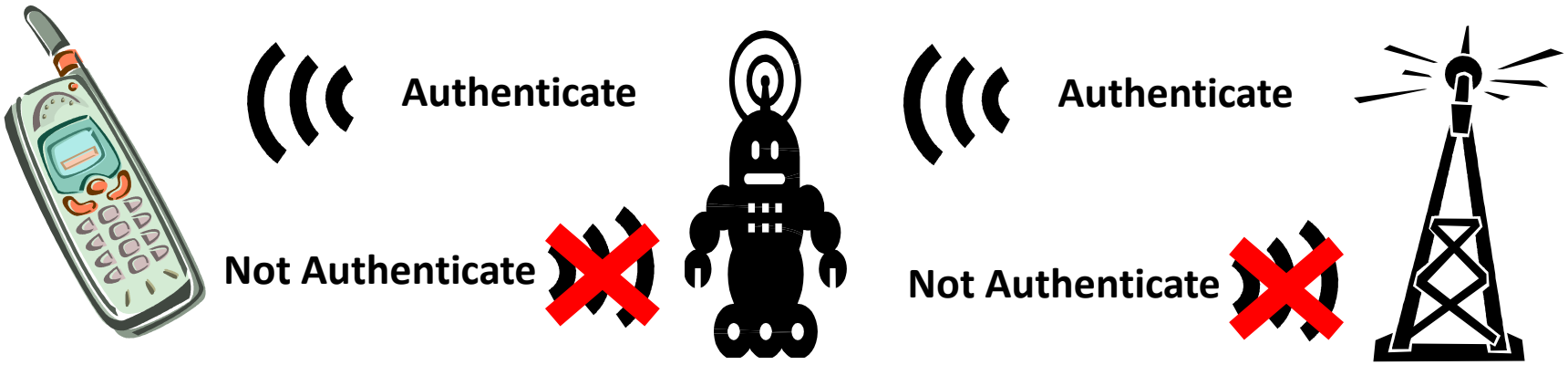
- Enhance the GSM AKA, avoid the Middle Man Attack.



UMTS Authentication Case



GSM Authentication Case



Middle Man (Forgery Radio Station)

Middle Man Attack in GSM

Investigation of forgery

- Assumption: the kernel function is a strong block chipper function.

- **Internal collision attack against f1/f1* :**

  $t' \oplus c1 \oplus rot(z' \oplus OPc, r1) = t'' \oplus c1 \oplus rot(z'' \oplus OPc, r1)$

  if a value s is xored on the equation:

  $w1(x', z' \oplus sz) = w2(x'', z'' \oplus sz)$

- This collision on f1/f1* is similar to that of the standard CBC-MAC which theoretically exits.

Attack against combination of f2-f5 :

Two distinct inputs x' and x", let equation:

$wi(x')wj(x')=wj(x'')wi(x'') <=>$

$rot(t',ri) \oplus rot(t'',rj)=ci \oplus cj \oplus rot(OPc,ri) \oplus rot(OPc,rj)$

There are four cases:

- **Case 1**: $ri=rj$.
- **If** $wi(x')=wj(x'')$,

  implies: $wi(x')=wj(x'')$.

Attack against combination of f2-f5 :

- **Case 2:** $ri\text{-}rj = 64 \bmod 128$, given a value v that $ci \oplus cj = rot(v,ri) \oplus rot(v,rj)$.
- Input x => output $wi(x)$ and $wj(x)$,

  at least one x input: $wi(x) = wj(x)$.
- The restrict condition: if and only if

  $rot(y \oplus OPc,ri) \oplus rot(y \oplus OPc,rj) = ci \oplus cj$ with possible $y \oplus OPc$.

Attack against combination of f2-f5 :

- **Case 3:** $ri=0$.
- Finding pair(x', z) and x'' under the condition:

  $t' \oplus rot(t'',ri)=ci \oplus ci \oplus rot(z,r1) \oplus rot(OPc,r1) \oplus rot(OPc, ri) <=> w1(x',u)=wi(x)$.

  It is highly probable to produce $w1(x',u)=wi(x'')$.
- As $ri=0$, also $w1(x'',u)=wi(x')$.

Attack against combination of f2-f5 :

- **Case 4:** $r1=0$ *or* 64 and $ri=rj$.
- given pairs (x,z) as input with corresponding $w1(x,z)$ and $wi(x)$ where there exists x', zi and x'' which produces the equation $w1(x',zi)=wi(x'')$
- Let $zj=zi \oplus ci \oplus cj$, we can forgery:

  $w1(x',zj)=wj(x'')$

Prevention of Attack

- 1st, the proper constants are selected
- 2nd, a large number of random challenges are required.
- 3rd, compute output through two independent permutation.

Thank You!