Security and Privacy in Wireless Networks

Mohammad Hossein Manshaei

manshaei@gmail.com

Chapter 1 (secowinet.epfl.ch):

Cellular networks (GSM, UMTS), WiFi LANs, and Bluetooth

THE SECURITY OF EXISTING WIRELESS NETWORKS

Why is security more of a concern in wireless?

- **No inherent physical protection**
  - physical connections between devices are replaced by logical associations
  - sending and receiving messages do not need physical access to the network infrastructure (cables, hubs, routers, etc.)

- **Broadcast communications**
  - wireless usually means radio, which has a broadcast nature
  - transmissions can be overheard by anyone in range
  - anyone can generate transmissions,
    - which will be received by other devices in range
    - which will interfere with other nearby transmissions and may prevent their correct reception (jamming)

- eavesdropping is easy
- injecting bogus messages into the network is easy
- replaying previously recorded messages is easy
- illegitimate access to the network and its services is easy
- denial of service is easily achieved by jamming

Wireless Communication Security Requirements

- **Confidentiality**
  - messages sent over wireless links must be encrypted
- **Authenticity**
  - origin of messages received over wireless links must be verified
- **Replay detection**
  - freshness of messages received over wireless links must be checked
- **Integrity**
  - modifying messages on-the-fly (during radio transmission) is not so easy, but possible …
  - integrity of messages received over wireless links must be verified
- **Access control**
  - access to the network services should be provided only to legitimate entities
  - access control should be permanent
    - it is not enough to check the legitimacy of an entity only when it joins the network and its logical associations are established, because logical associations can be hijacked
- **Protection against jamming**

Outline

- Cellular networks
- WiFi LANs
- Bluetooth

Mobile Authentication, Confidentiality of Communications, and Privacy

GLOBAL SYSTEM FOR MOBILE COMMUNICATIONS (GSM)

GSM Security

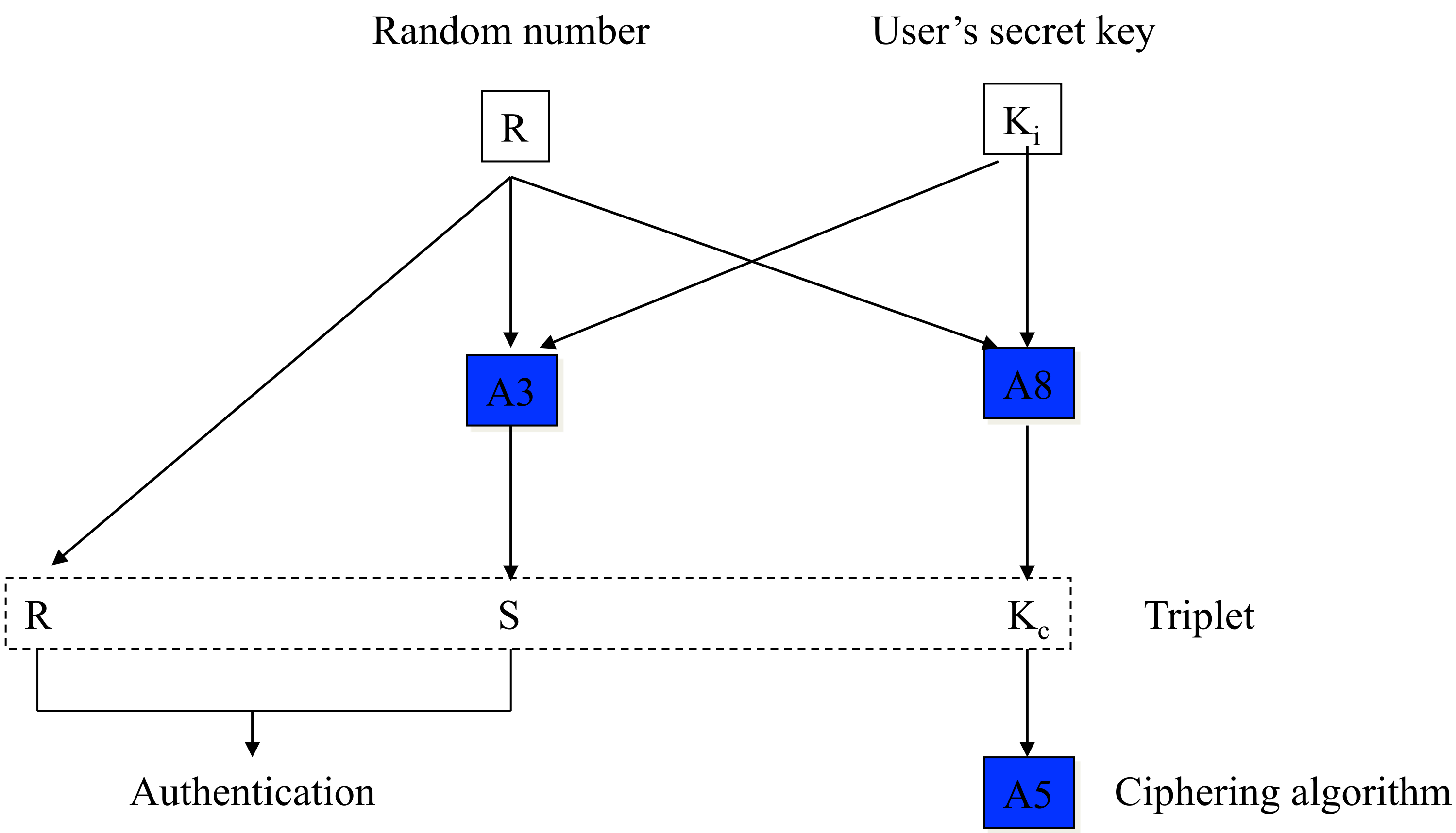
- Main security requirement
  - Subscriber authentication (for the sake of billing)
    - challenge-response protocol
    - long-term secret key shared between the subscriber and the home network operator
    - supports roaming without revealing long-term key to the visited networks

- Other security services provided by GSM
  - **Confidentiality** of communications and signaling over the wireless interface
    - encryption key shared between the subscriber and the visited network is established with the help of the home network as part of the subscriber authentication protocol
  - **Protection of the subscriber's identity** from eavesdroppers on the wireless interface
    - usage of short-term temporary identifiers

The SIM card (Subscriber Identity Module)

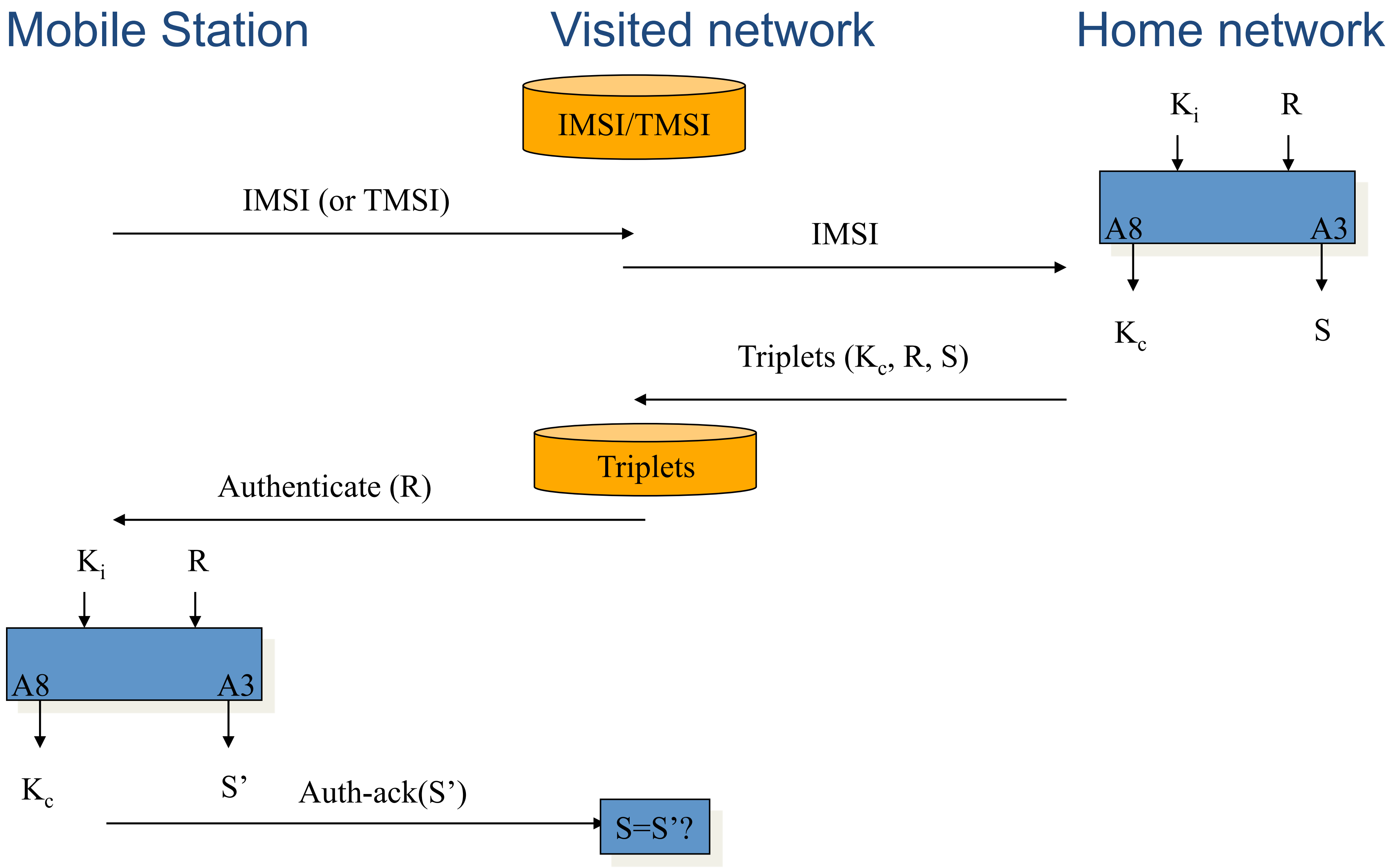
- Must be tamper-resistant
- Protected by a PIN code (checked locally by the SIM)
- Is removable from the terminal
- Contains all data specific to the end user which have to reside in the Mobile Station:
  - IMSI: International Mobile Subscriber Identity (permanent user's identity)
  - PIN
  - TMSI (Temporary Mobile Subscriber Identity)
  - K_i : User's secret key
  - K_c : Ciphering key
  - List of the last call attempts
  - List of preferred operators
  - Supplementary service data (abbreviated dialing, last short messages received,...)

Cryptographic Algorithms of GSM

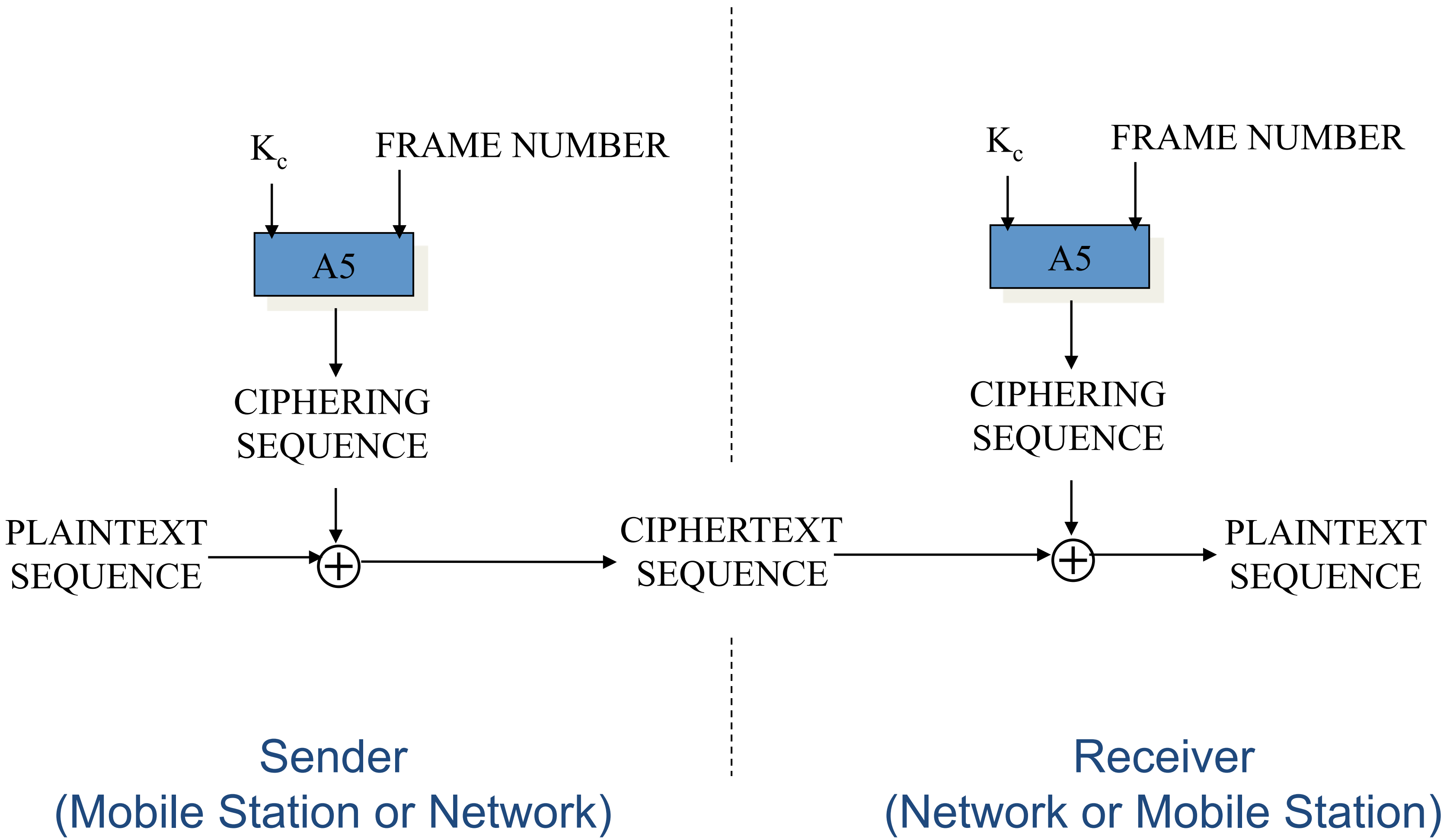


K_c: ciphering key
S : signed result
A3: subscriber authentication (operator-dependent algorithm)
A5: ciphering/deciphering (standardized algorithm)
A8: cipher generation (operator-dependent algorithm)

Authentication Principle of GSM



Ciphering in GSM



Temporary Mobile Subscriber Identifier (TMSI)

- After each successful authentication, the subscriber receives a TMSI
- TMSI is encrypted with the fresh CK
- For the following authentication, user will use TMSI (even in new networks)

Conclusion on GSM security

- Focused on the protection of the air interface
- No protection on the wired part of the network (neither for privacy nor for confidentiality)
- The visited network has access to all data (except the secret key of the end user)
- Generally robust, but a few successful attacks have been reported:
  - Faked base stations
  - Cloning of the SIM card

Integrity Protection and Network Authentication

UNIVERSAL MOBILE TELECOMMUNICATIONS SYSTEM (UMTS)

3GPP Security Principles (1/2)

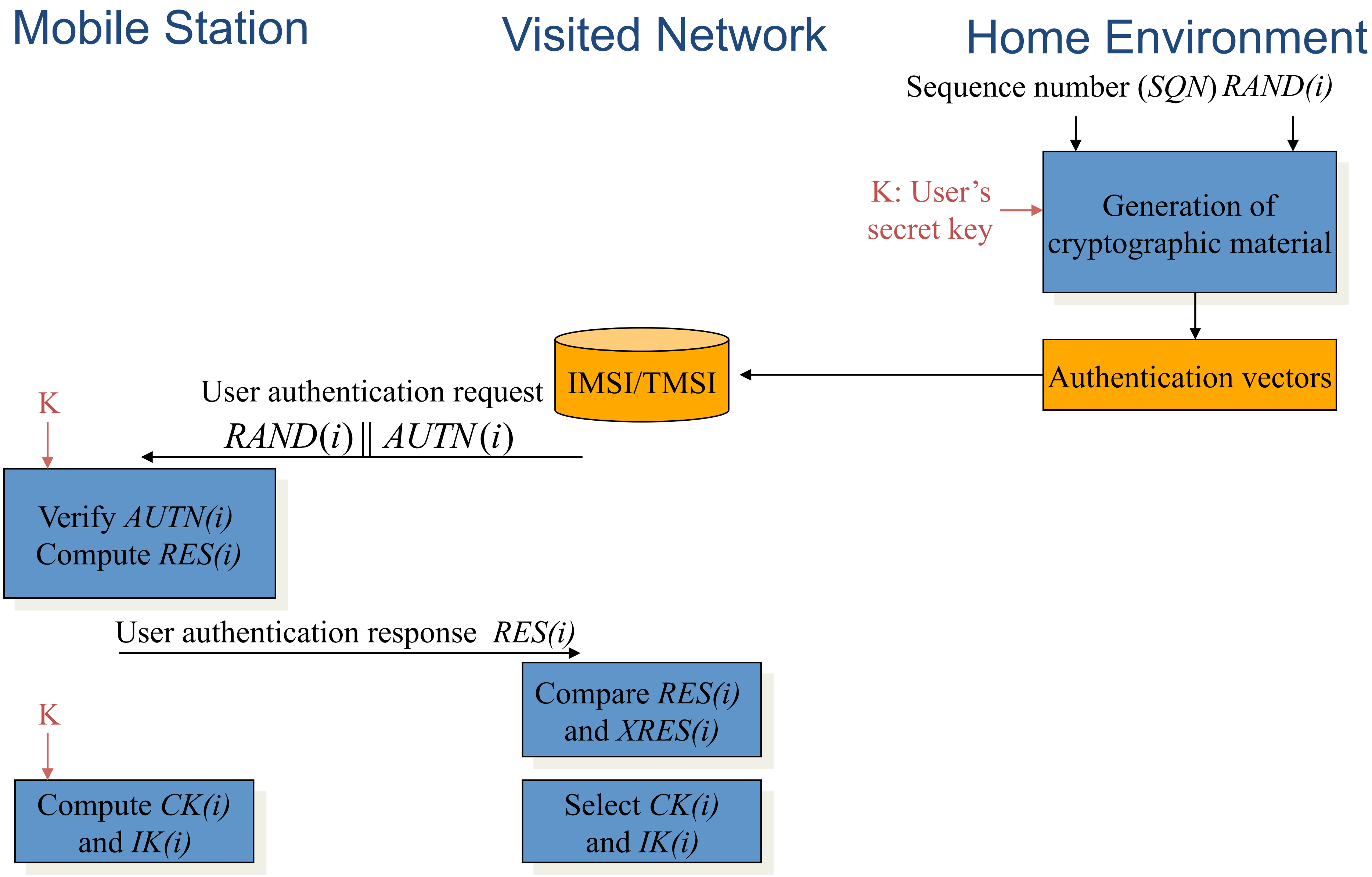
- Reuse of 2nd generation security principles (GSM):
  - Removable hardware security module
    - In GSM: SIM card
    - In 3GPP: USIM (User Services Identity Module)
  - Radio interface encryption
  - Limited trust in the Visited Network
  - Protection of the identity of the end user (especially on the radio interface)

- Correction of the following weaknesses of the previous generation:
  - Possible attacks from a faked base station
  - Cipher keys and authentication data transmitted in clear between and within networks
  - Encryption not used in some networks ➔ open to fraud
  - Data integrity not provided
  - …

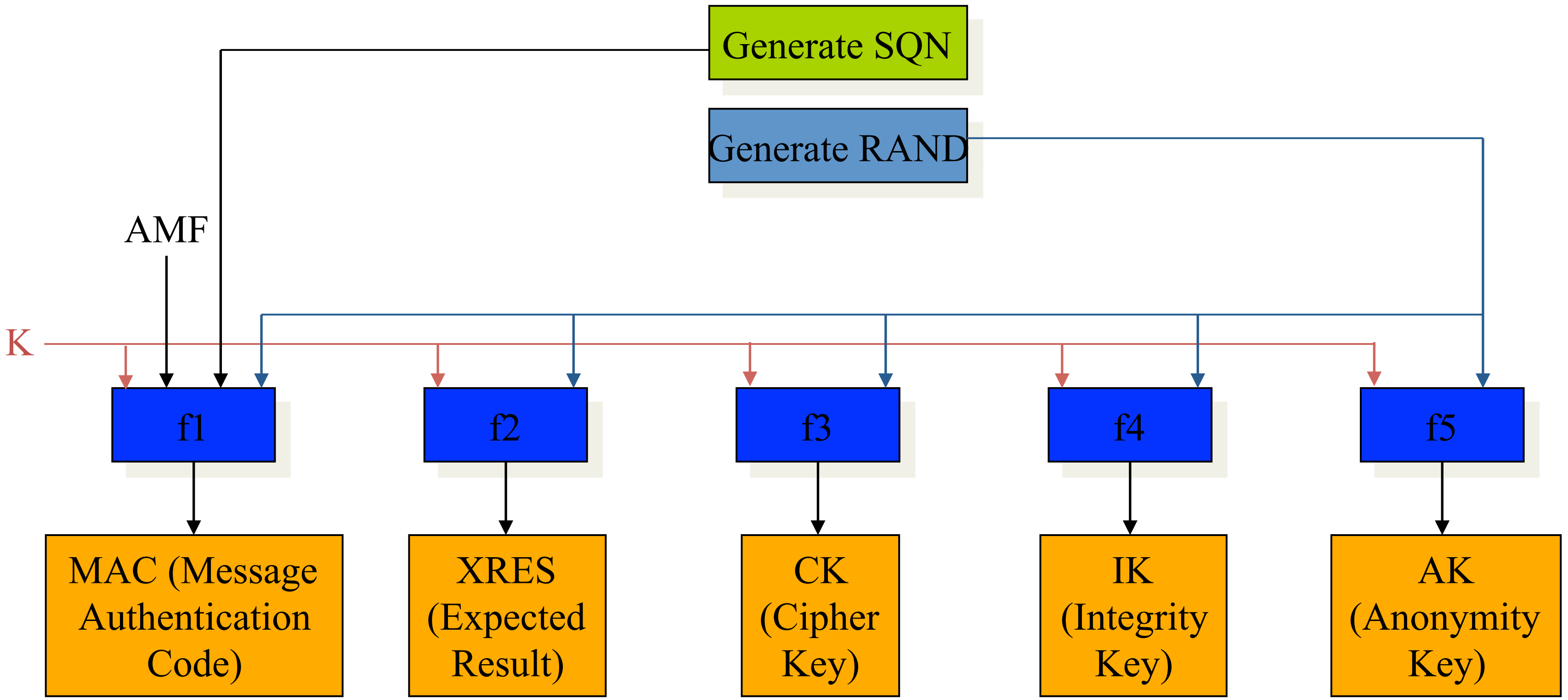
3GPP Security Principles (2/2)

- New security features
  - New kind of service providers (content providers, HLR (home location register) only service providers,…)
  - Increased control for the user over their service profile
  - Enhanced resistance to active attacks
  - Increased importance of non-voice services
  - …

Authentication in 3GPP



Generation of the Authentication Vectors

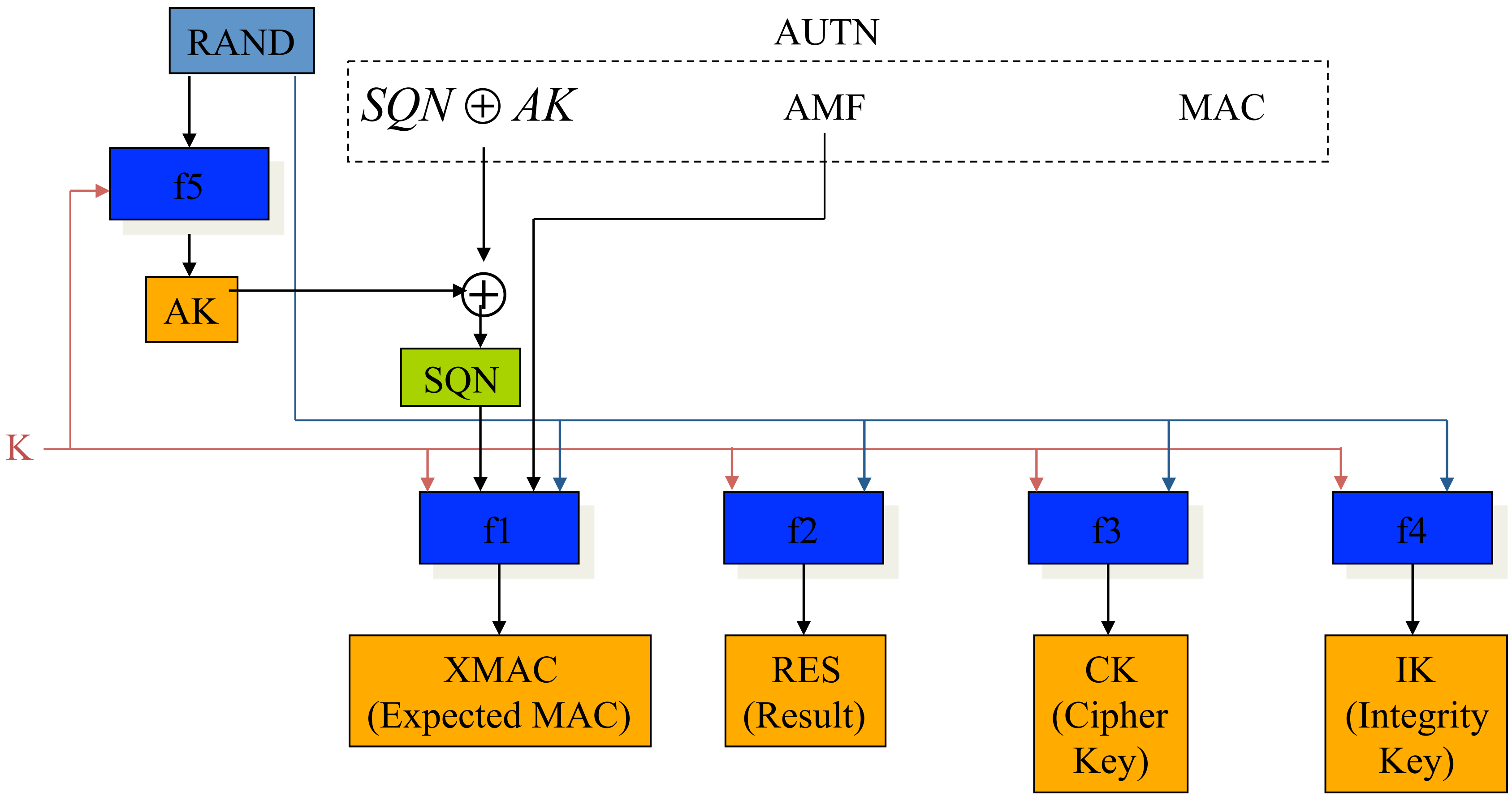


$$AUTN := (SQN \oplus AK) || AMF || MAC$$

$$AV := RAND || XRES || CK || IK || AUTN$$

AMF: Authentication and Key Management Field
AUTN: Authentication Token
AV: Authentication Vector

User Authentication Function in the USIM



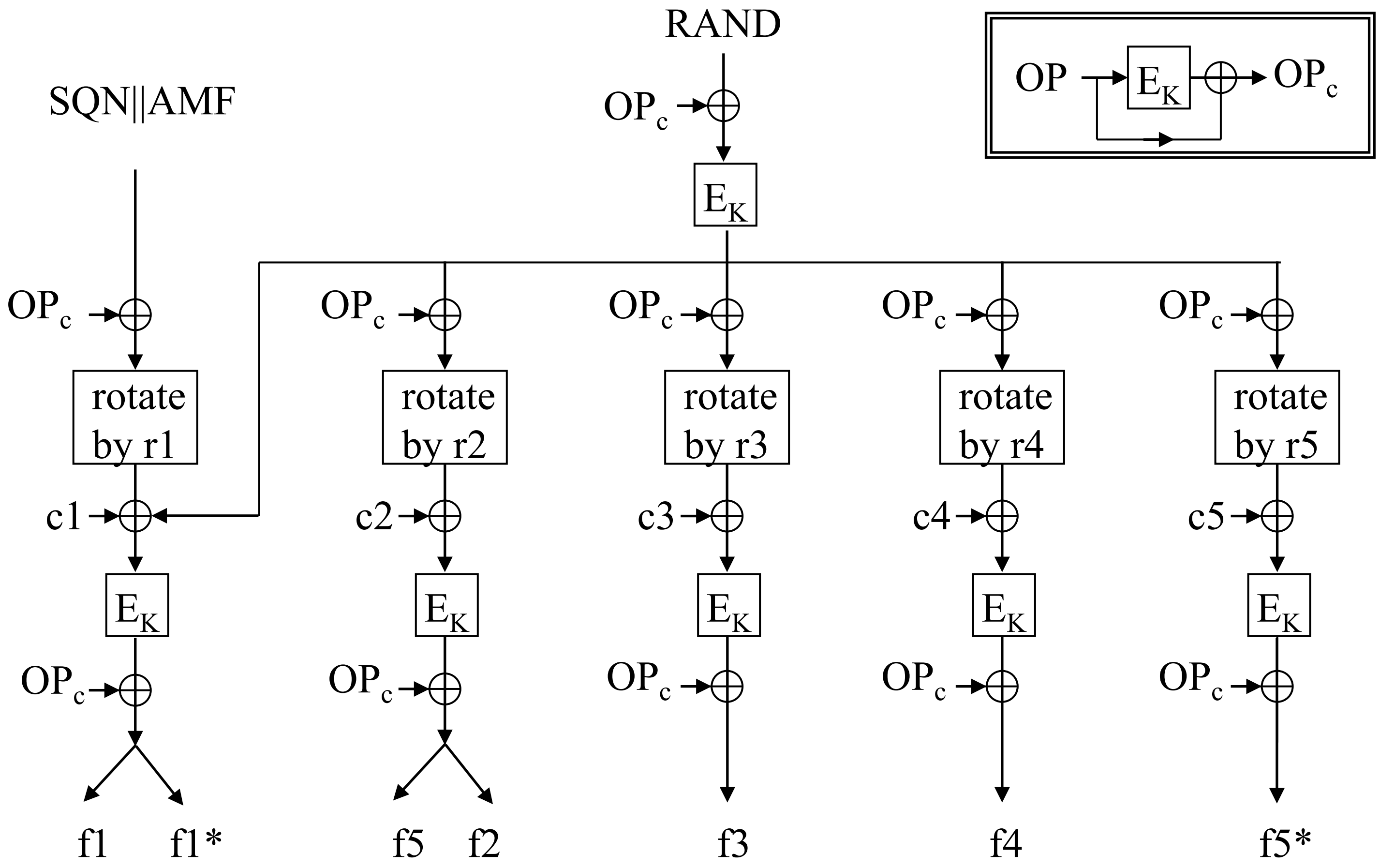
- Verify MAC = XMAC
- Verify that SQN is in the correct range

USIM: User Services Identity Module

More about the Authentication and Key Generation

- In addition to f1, f2, f3, f4 and f5, two more functions are defined: f1* and f5*, used in case the authentication procedure gets desynchronized (detected by the range of SQN).
- f1, f1*, f2, f3, f4, f5 and f5* are operator-specific
- However, 3GPP provides a detailed example of algorithm set, called *MILENAGE*
- MILENAGE is based on the *Rijndael* block cipher
- In MILENAGE, the generation of all seven functions f1…f5* is based on the Rijndael algorithm

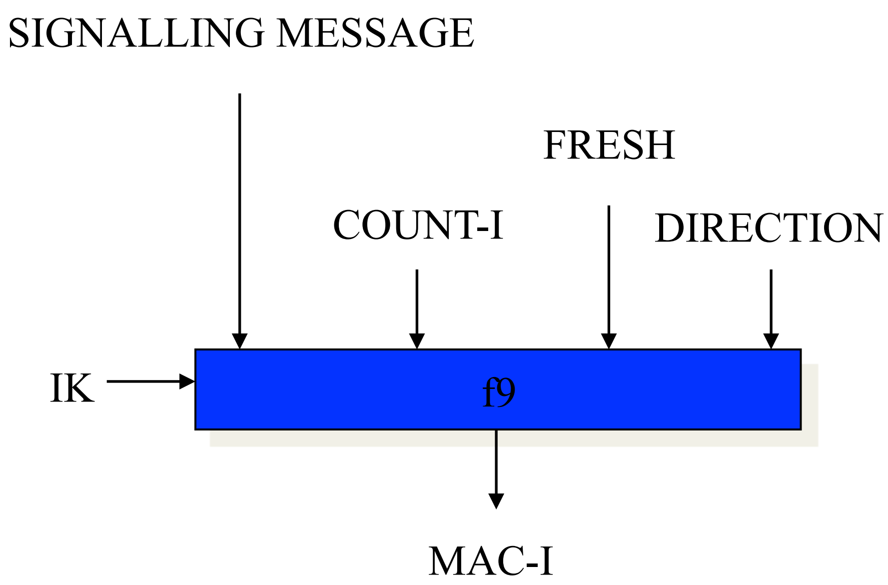
Authentication and Key Generation Functions f1…f5*



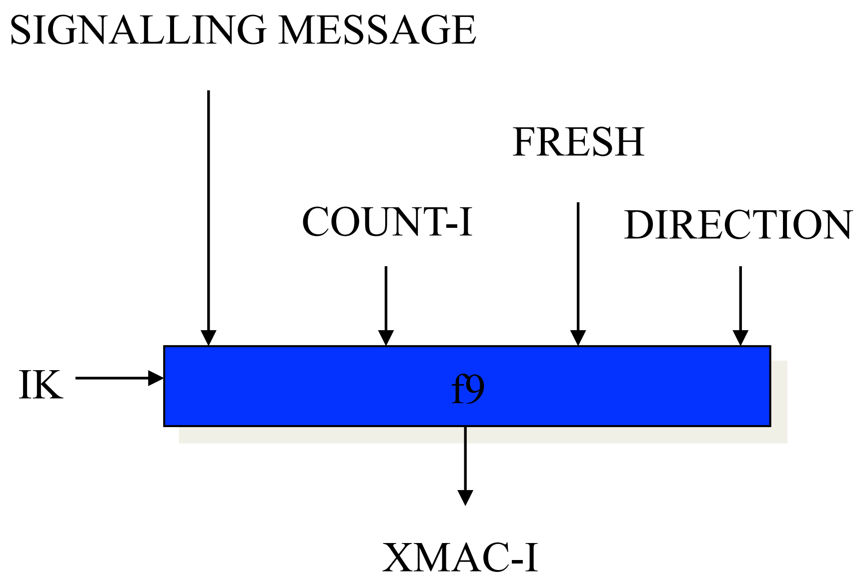
OP: operator-specific parameter
r1,…, r5: fixed rotation constants
c1,…, c5: fixed addition constants

E_K : Rijndael block cipher with 128 bits text input and 128 bits key

Signalling Integrity Protection Method



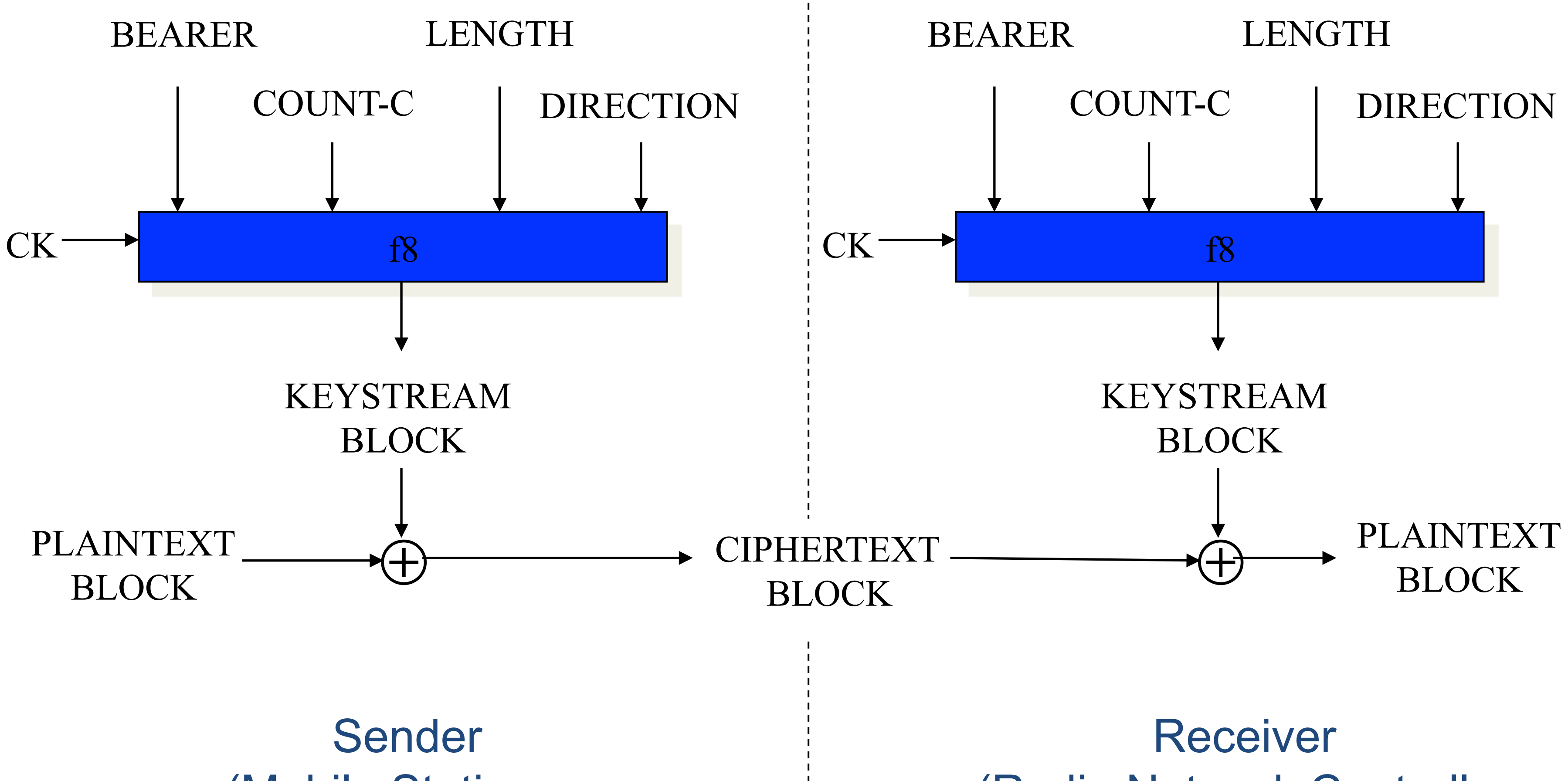
Sender
(Mobile Station or
Radio Network Controller)



Receiver
(Radio Network Controller
or Mobile Station)

FRESH: random input

Ciphering Method

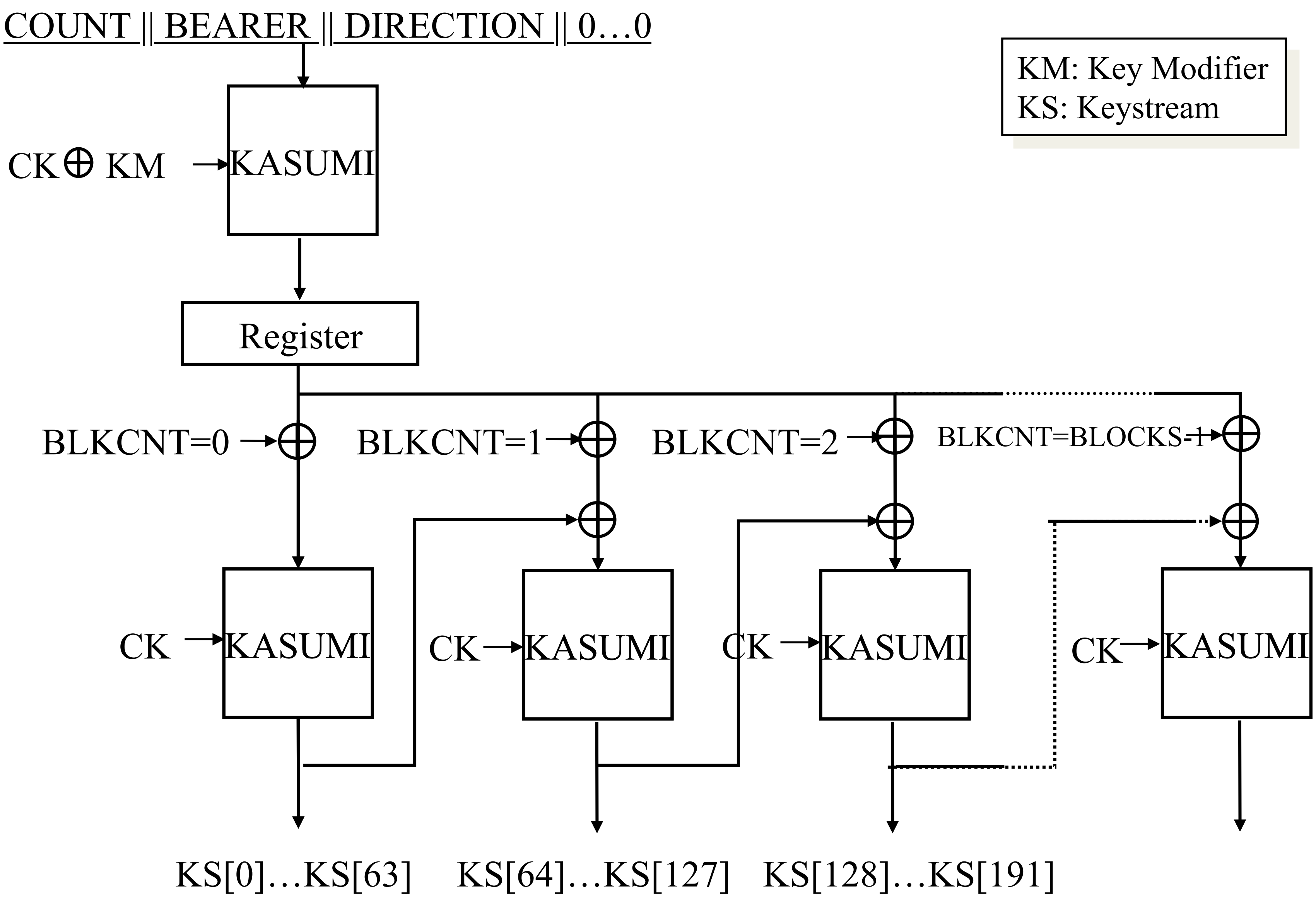


Sender
(Mobile Station or
Radio Network Controller)

Receiver
(Radio Network Controller
or Mobile Station)

BEARER: radio bearer identifier
COUNT-C: ciphering sequence counter

The Keystream Generator f8



Details of Kasumi

KL_i, KO_i , KI_i : subkeys used at ith round
S7, S9: S-boxes

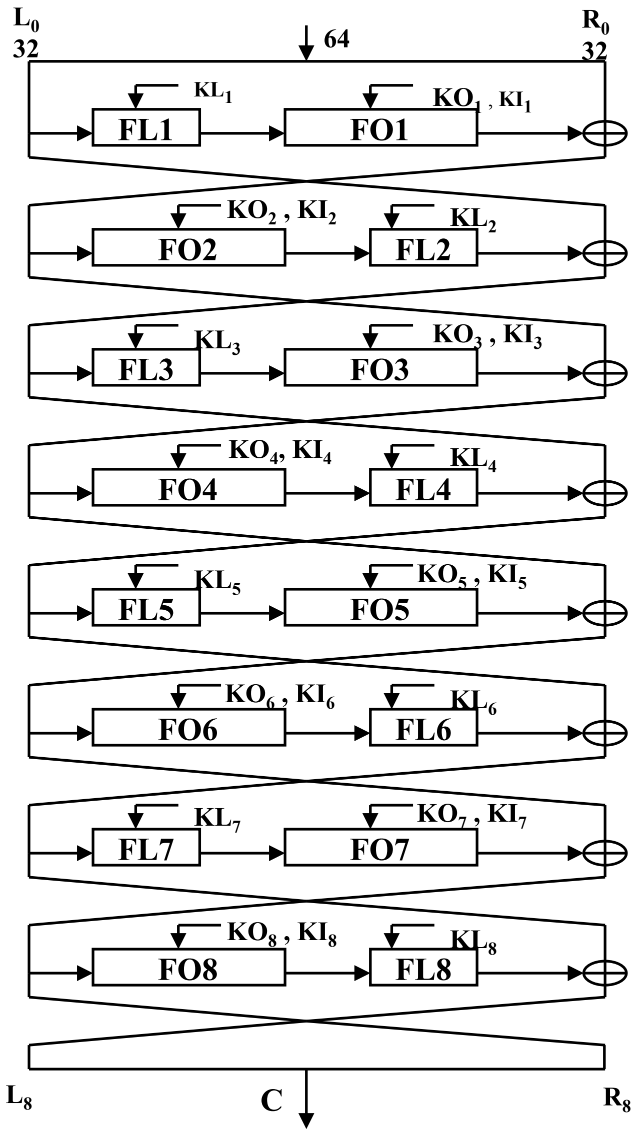


Fig. 1 : **KASUMI**

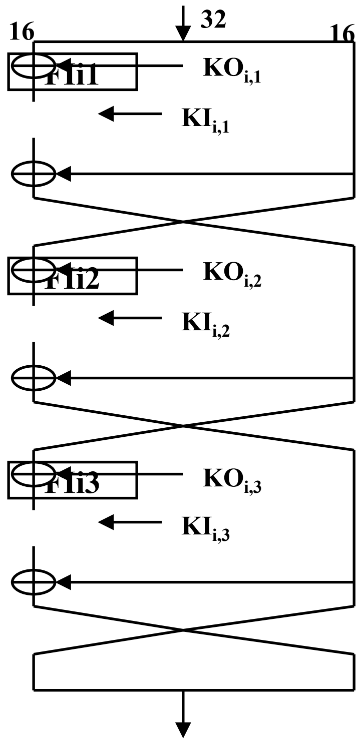


Fig. 2 : **FO Function**

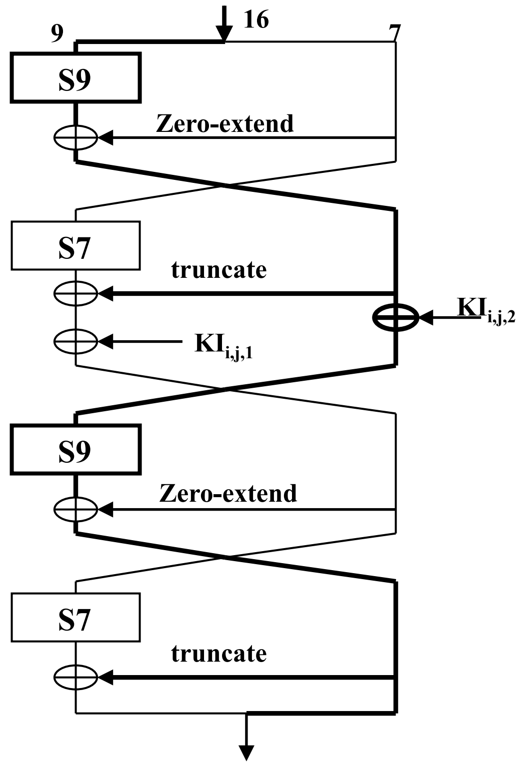


Fig. 3 : **FI Function**

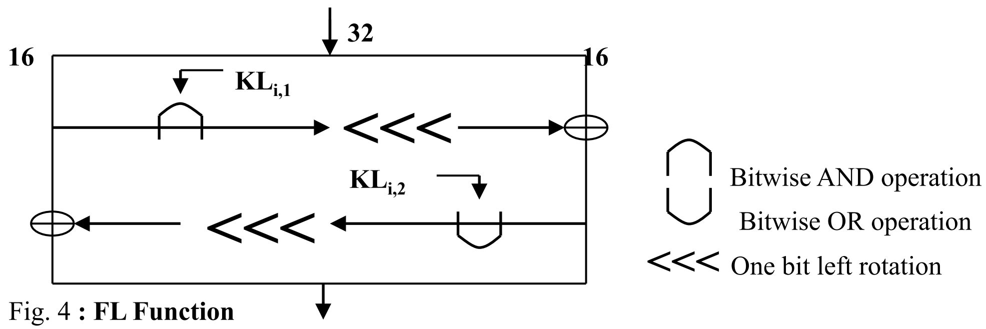


Fig. 4 : **FL Function**

Weakness in the UMTS

- The visited network is not authenticated to the subscriber.

- Problem:
  - Allows a malicious network *operator X* to masquerade as *network Y* to the subscriber
  - Charging!

Conclusion on 3GPP Security

- Some improvement with respect to 2nd generation
  - Cryptographic algorithms are published
  - Integrity of the signalling messages is protected
- Quite conservative solution
- Privacy/anonymity of the user not completely protected
- 2nd/3rd generation interoperation will be complicated and might open security breaches

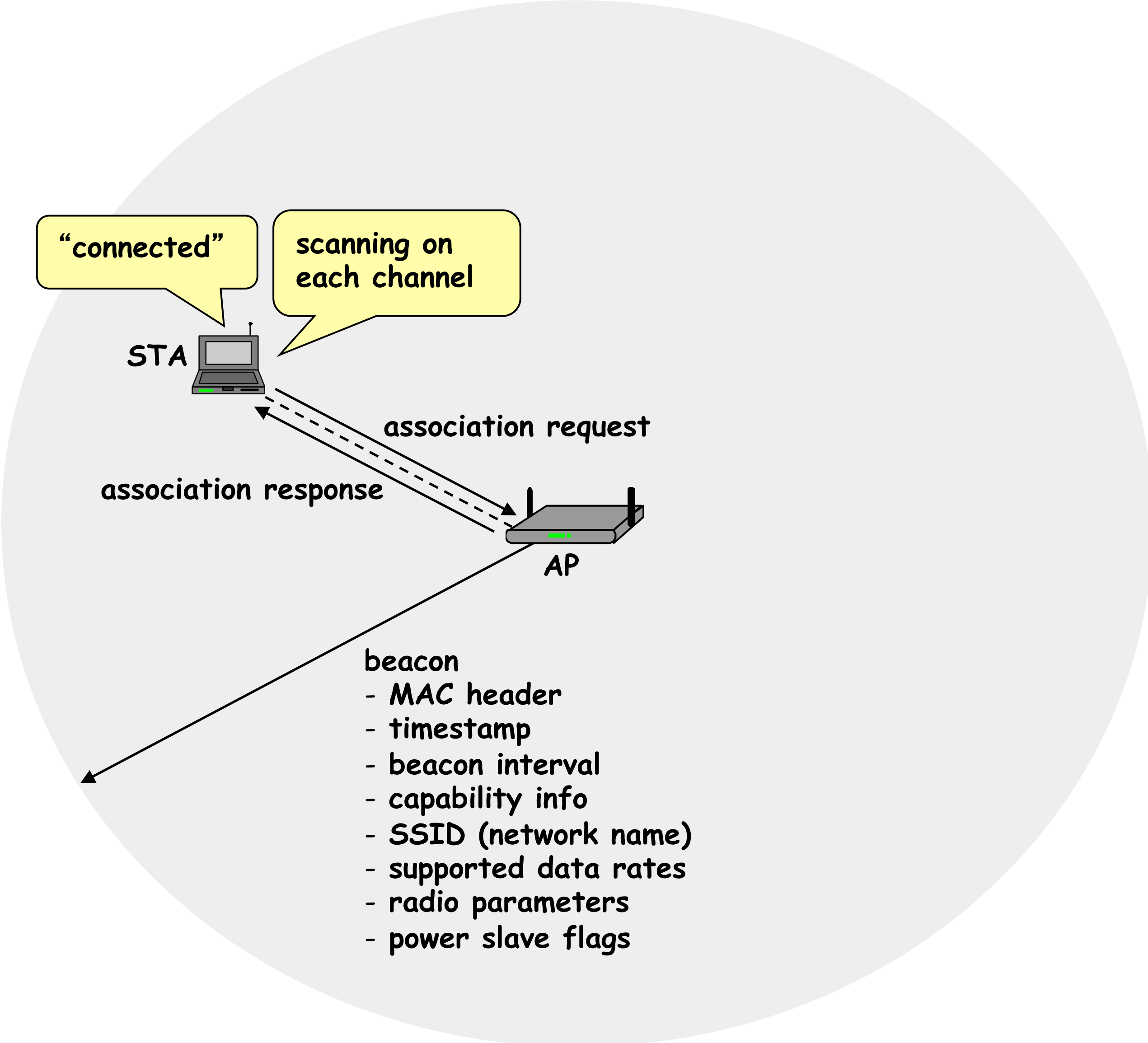
Outline

- Cellular networks
- WiFi LANs
- Bluetooth

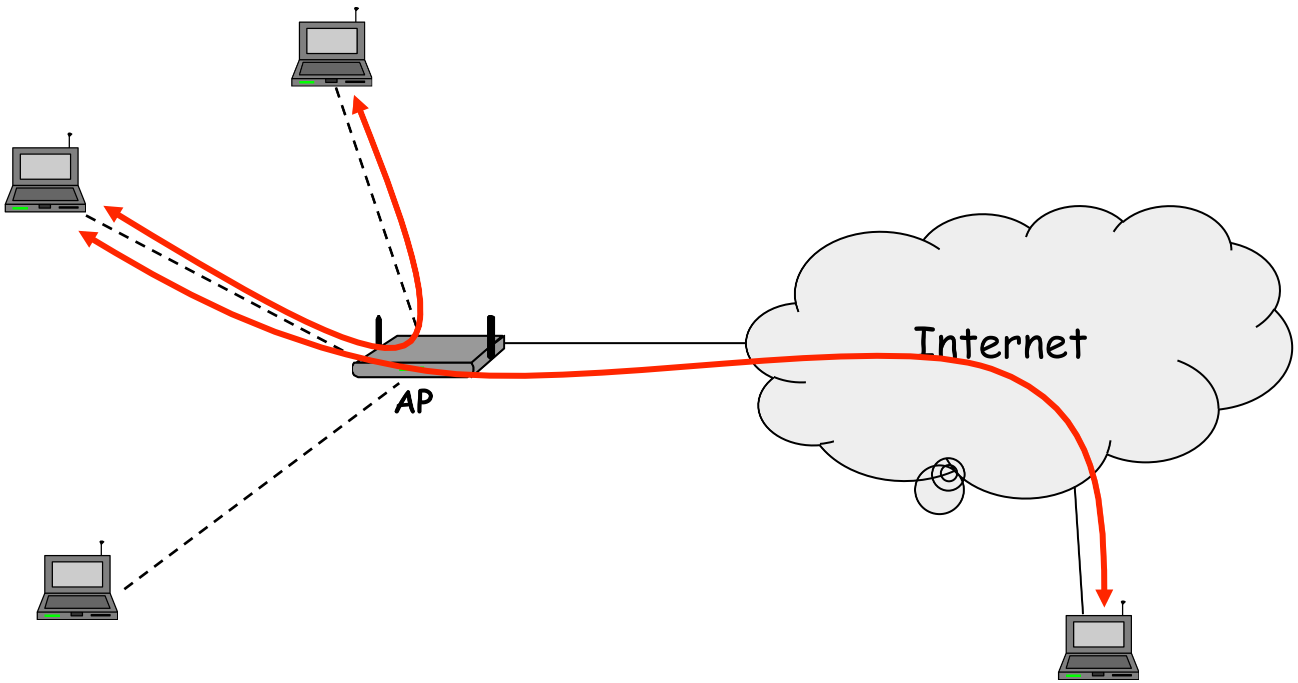
WEP and 802.11i

WIFI/802.11

Introduction to WiFi



Introduction to WiFi



WEP – Wired Equivalent Privacy

- **Part of the IEEE 802.11 specification**

- **Goal**
  - Make the WiFi network *at least as secure as a wired LAN* (that has no particular protection mechanisms)
  - WEP was never intended to achieve strong security

- **Services**
  - Access control to the network
  - Message confidentiality
  - Message integrity

WEP – Access Control

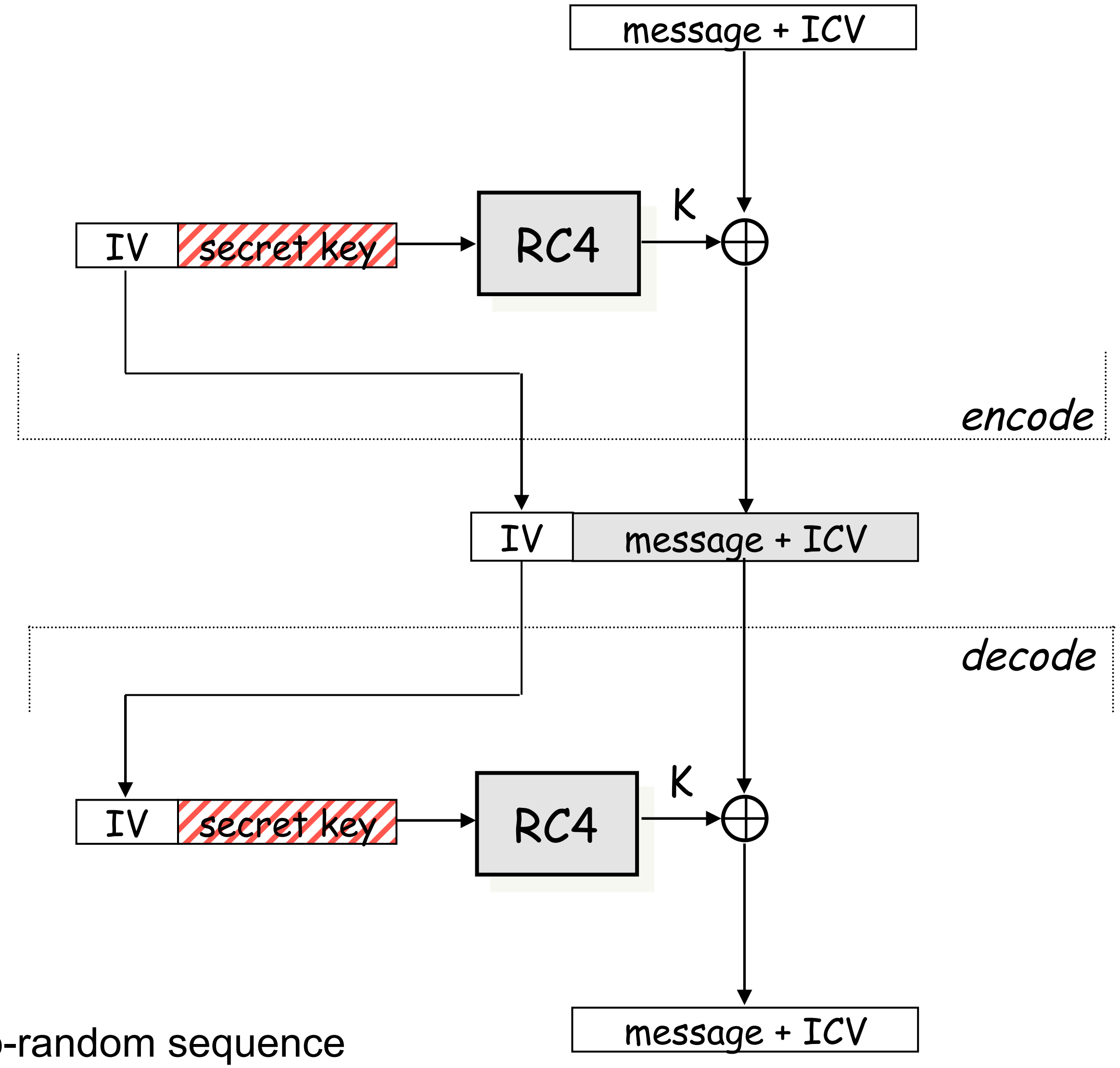
- Before association, the STA needs to authenticate itself to the AP
- Authentication is based on a simple challenge-response protocol:
  - STA → AP: authenticate request
  - AP → STA: authenticate challenge (r) // r is 128 bits long
  - STA → AP: authenticate response ($e_K(r)$)
  - AP → STA: authenticate success/failure
- Once authenticated, the STA can send an association request, and the AP will respond with an association response
- If authentication fails, no association is possible

WEP – Message Confidentiality and Integrity

- WEP encryption is **based on RC4** (a stream cipher developed in 1987 by Ron Rivest for RSA Data Security, Inc.)
  - **operation:**
    - for each message to be sent:
      - RC4 is **initialized** with the shared secret (between STA and AP)
      - RC4 produces a **pseudo-random byte sequence** (key stream)
      - this pseudo-random byte sequence is **XORed** to the message
    - reception is analogous
  - It is essential that each message is encrypted with a different key stream
    - the RC4 generator is initialized with the shared secret and an **IV (initial value) together**
      - shared secret is the same for each message
      - **24-bit IV changes for every message**

- WEP integrity protection is based on an encrypted CRC value
  - operation:
    - **ICV (integrity check value)** is computed and appended to the message
    - The message and the ICV are **encrypted together**

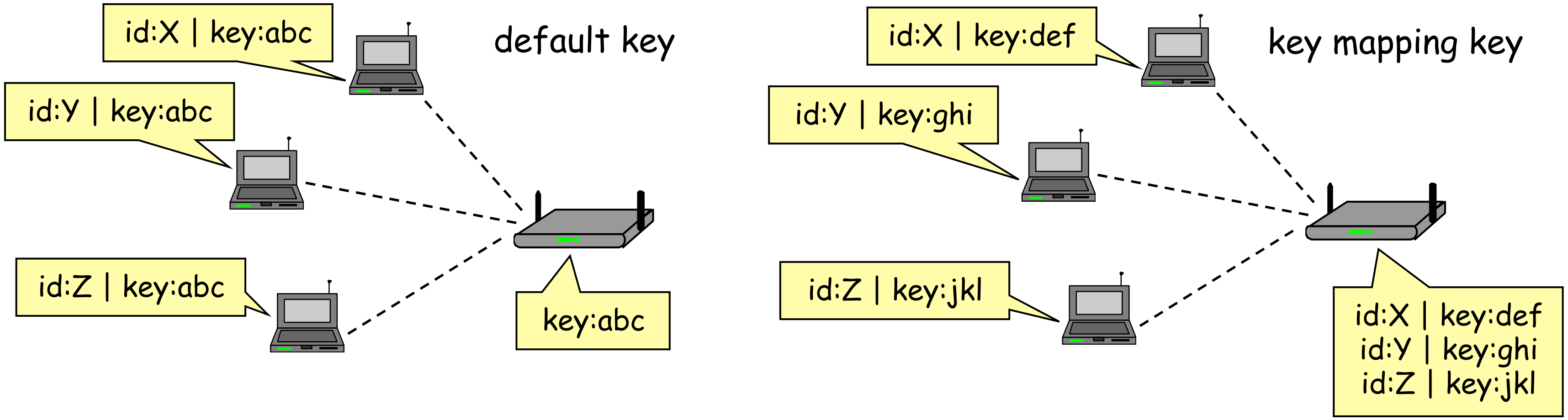
WEP – Message Confidentiality and Integrity



K: pseudo-random sequence

WEP – Keys

- Two kinds of keys are allowed by the standard
  - **default key** (also called shared key, group key, multicast key, broadcast key, key)
  - **key mapping keys** (also called individual key, per-station key, unique key)

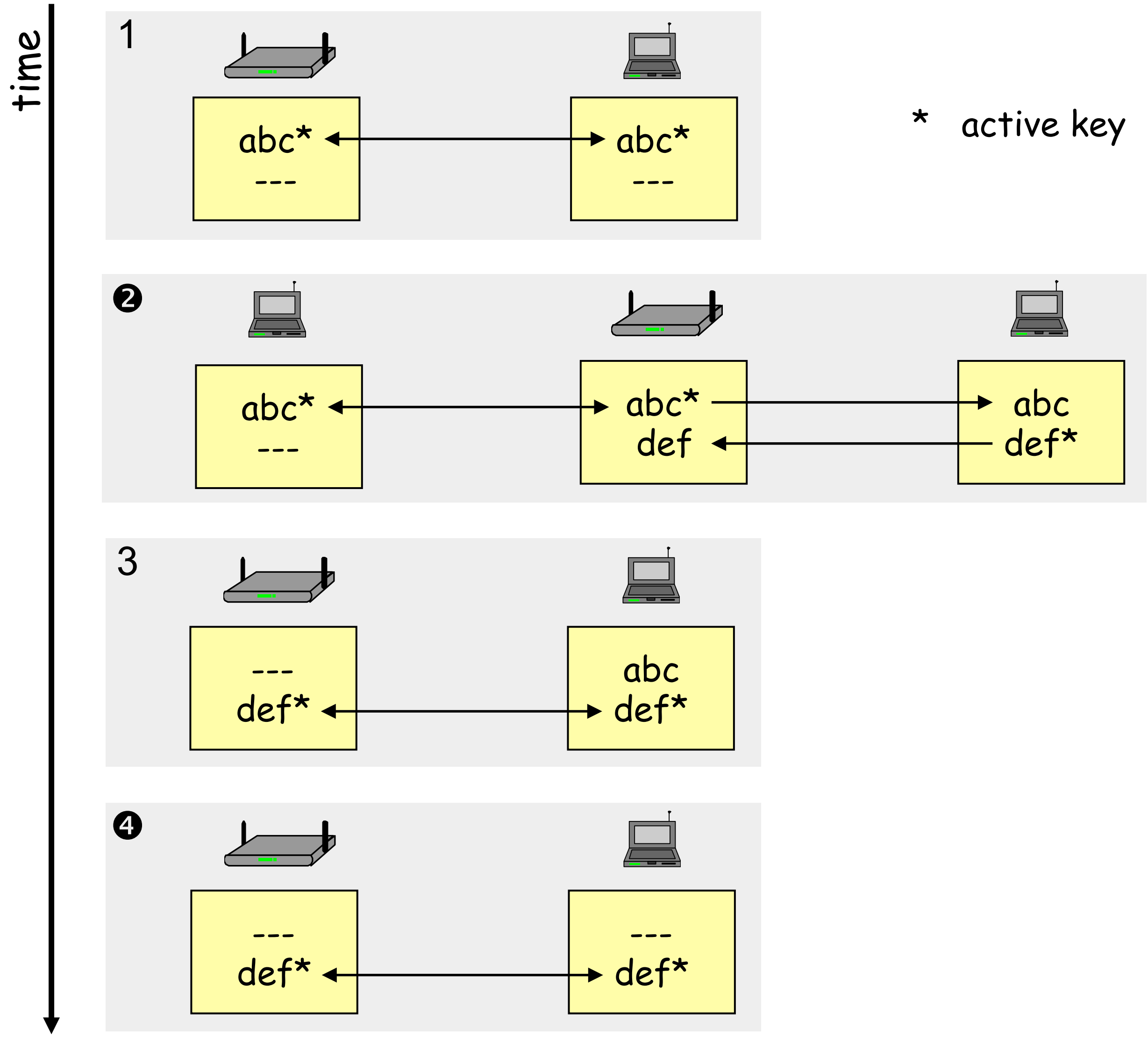


- In practice, **often only default** keys are supported
  - The default key is manually installed in every STA and the AP
  - Each STA uses the same shared secret key → in principle**, STAs can decrypt each other's messages**

WEP – Management of Default Keys

- The default key is a group key, and group keys need to be changed when a member leaves the group
  - e.g., when someone leaves the company and shouldn't have access to the network anymore
- It is practically impossible to change the default key in every device simultaneously
- Hence, WEP **supports multiple default keys** to help the smooth change of keys
  - one of the keys is called the active key
  - the active key is used to encrypt messages
  - any key can be used to decrypt messages
  - the message header contains a key ID that allows the receiver to find out which key should be used to decrypt the message

WEP – The key change process



WEP Flaws – Authentication and Access Control

- Authentication is **one-**way only
  - AP is not authenticated to STA
  - STA is at risk to associate to a rogue AP
- The same **shared secret key** is used for **authentication and encryption**
  - weaknesses in any of the two protocols can be used to break the key
- **No session** key is established during authentication
  - Access control is not continuous
  - Once a STA has authenticated and associated to the AP, an attacker send messages using the MAC address of STA
  - Correctly encrypted messages cannot be produced by the attacker, but replay of **other STA messages** is still possible
- **STA can be impersonated**
  - … next slide

WEP Flaws – Authentication and Access Control

- Recall that authentication is based on a challenge-response protocol:

  ...
  AP → STA: r
  STA → AP: IV | r $\oplus$ K
  ...

  where K is a 128 bit RC4 output (Pseudo Random Sequence) on IV and the shared secret

- An attacker can compute: r $\oplus$ (r $\oplus$ K) = K

- Then it can use K to impersonate STA later:

  ...
  AP → attacker: r'
  attacker → AP: IV | r' $\oplus$ K
  ...

WEP Flaws – Integrity and Replay Protection

- **There's no replay protection at all**
  - IV is not mandated to be incremented after each message

- The attacker **can manipulate messages** despite the ICV mechanism and encryption
  - CRC is a linear function wrt to XOR:

$$CRC(X \oplus Y) = CRC(X) \oplus CRC(Y)$$

  - Attacker observes $(M \mid CRC(M)) \oplus K$ where K is the RC4 output
  - For any ΔM, the attacker can compute $CRC(\Delta M)$
  - Hence, the attacker can compute:

$$((M \mid CRC(M)) \oplus K) \oplus (\Delta M \mid CRC(\Delta M)) =$$
$$((M \oplus \Delta M) \mid (CRC(M) \oplus CRC(\Delta M))) \oplus K =$$
$$((M \oplus \Delta M) \mid CRC(M \oplus \Delta M)) \oplus K$$

WEP Flaws – Confidentiality

- **IV reuse**
  - IV space is too small
    - IV size is only 24 bits → there are 16,777,216 possible IVs
    - **after around 17 million messages, IVs are reused**
    - a busy AP at 11 Mbps is capable for transmitting 700 packets per second → IV space is used up in around 7 hours
  - In many implementations **IVs are initialized with 0 on startup**
    - if several devices are switched on nearly at the same time, they all use the same sequence of IVs
    - if they all use the same default key (which is the common case), then IV collisions are readily available to an attacker

- **Weak RC4 keys**
  - for some seed values (called weak keys), the beginning of the RC4 output is not really random
  - if a weak key is used, then the first few bytes of the output reveals a lot of information about the key → breaking the key is made easier
  - for this reason, crypto experts suggest to always throw away the first 256 bytes of the RC4 output, but WEP doesn't do that
  - due to the use of IVs, eventually a weak key will be used, and the attacker will know that, because the IV is sent in clear

  → **WEP encryption can be broken by capturing a few million messages !!!**

WEP – Lessons Learnt

1. **Engineering security protocols is difficult**
   - One can combine otherwise strong building blocks **in a wrong way** and obtain an insecure system at the end
     - Example 1:
       - stream ciphers alone are OK
       - challenge-response protocols for entity authentication are OK
       - but they shouldn't be combined
     - Example 2:
       - encrypting a message digest to obtain an ICV is a good principle
       - but it doesn't work if the message digest function is linear wrt to the encryption function
   - **Don't do it alone** (unless you are a security expert)
     - functional properties can be tested, but security is a non-functional property → it is extremely difficult to tell if a system is secure or not
   - **Using an expert in the design phase pays out** (fixing the system after deployment will be much more expensive)
     - experts will not guarantee that your system is 100% secure
     - but at least they know many pitfalls
     - they know the details of crypto algorithms

2. **Avoid the use of WEP (as much as possible)**

WPA

IEEE 802.11I

Overview of 802.11i

- After the collapse of WEP, IEEE started to develop a new security architecture → 802.11i

- Main novelties in 802.11i wrt to WEP
  - Access control model is based on 802.1X
  - Flexible authentication framework (based on EAP – Extensible Authentication Protocol)
  - Authentication can be based on strong protocols (e.g., TLS – Transport Layer Security)
  - Authentication process results in a shared session key (which prevents session hijacking)
  - Different functions (encryption, integrity) use different keys derived from the session key using a one-way function
  - Integrity protection is improved
  - Encryption function is improved

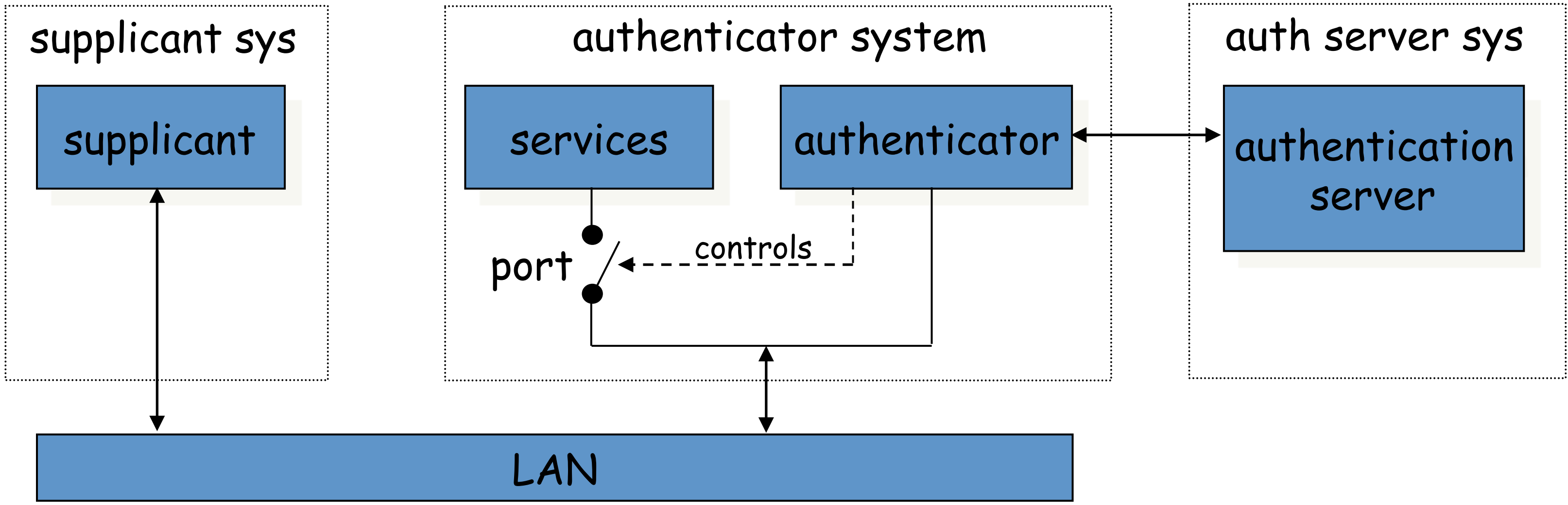
Overview of 802.11i

- 802.11i defines the concept of RSN (Robust Security Network)
  - integrity protection and encryption is based on AES (and not on RC4 anymore)
  - nice solution, but needs new hardware → cannot be adopted immediately

- 802.11i also defines an optional protocol called TKIP (Temporal Key Integrity Protocol)
  - integrity protection is based on Michael (we will skip the details of that)
  - encryption is based on RC4, but WEP's problems have been avoided
  - ugly solution, but runs on old hardware (after software upgrade)

- Industrial names
  - TKIP → WPA (WiFi Protected Access)
  - RSN/AES → WPA2

802.1X Authentication Model



- The supplicant requests access to the services (wants to connect to the network)
- The authenticator controls access to the services (controls the state of a port)
- The authentication server authorizes access to the services
  - the supplicant authenticates itself to the authentication server
  - if the authentication is successful, the authentication server instructs the authenticator to switch the port on
  - the authentication server informs the supplicant that access is allowed

Mapping the 802.1X Model to WiFi

- Supplicant → mobile device (STA)
- Authenticator → access point (AP)
- Authentication server → server application running on the AP or on a dedicated machine
- Port → logical state implemented in software in the AP

- One more thing is added to the basic 802.1X model in 802.11i:
  - Successful authentication results not only in switching the port on, but also in a session key between the mobile device and the authentication server
  - The session key is sent to the AP in a secure way
    - This assumes a shared key between the AP and the auth server
    - This key is usually set up manually

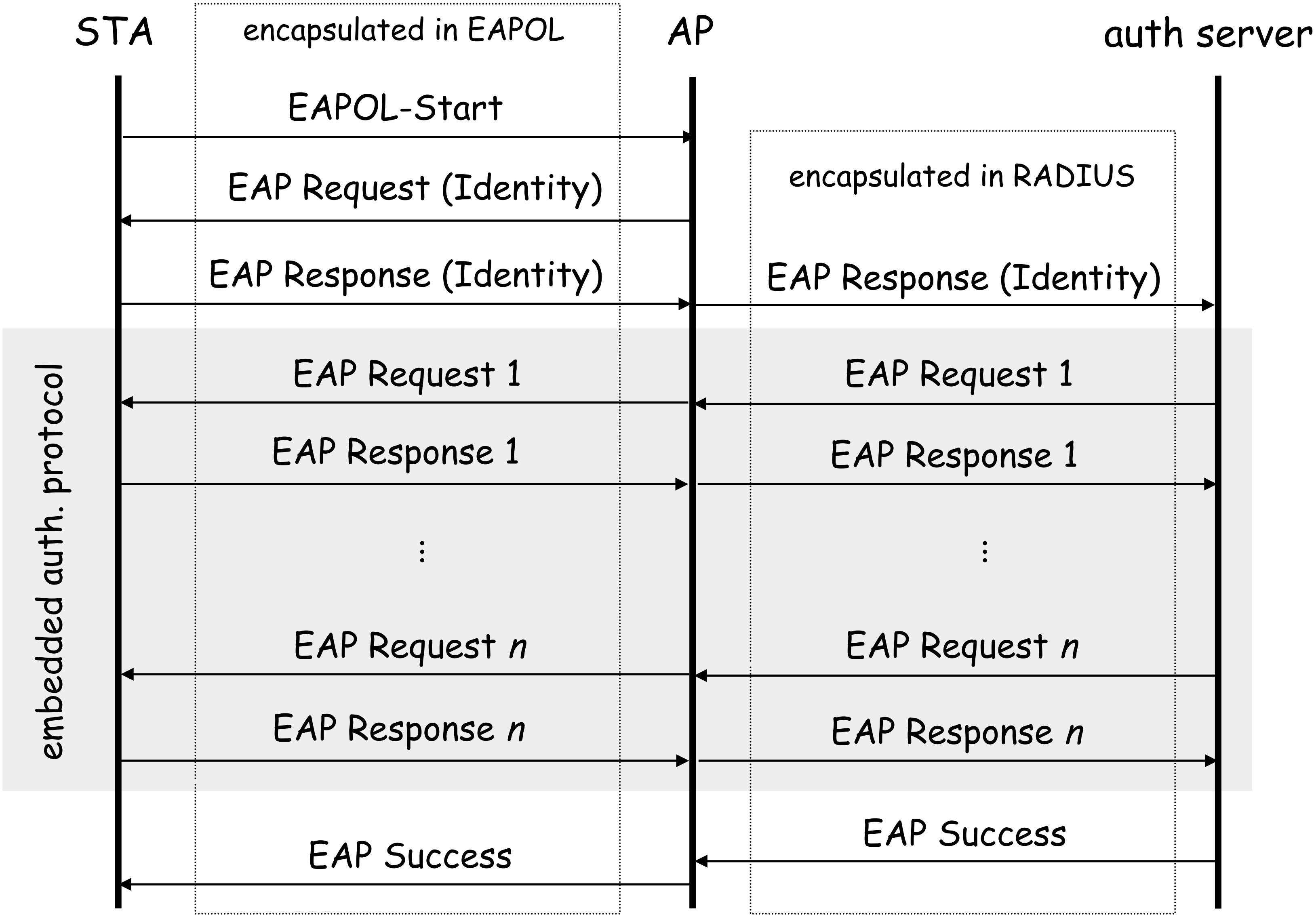
Protocols – EAP, EAPOL, and RADIUS

- EAP (Extensible Authentication Protocol) [RFC 3748]
  - carrier protocol designed to transport the messages of "real" authentication protocols (e.g., TLS)
  - very simple, four types of messages:
    - EAP request – carries messages from the supplicant to the authentication server
    - EAP response – carries messages from the authentication server to the supplicant
    - EAP success – signals successful authentication
    - EAP failure – signals authentication failure
  - authenticator doesn't understand what is inside the EAP messages, it recognizes only EAP success and failure

- EAPOL (EAP over LAN) [802.1X]
  - used to encapsulate EAP messages into LAN protocols (e.g., Ethernet)
  - EAPOL is used to carry EAP messages between the STA and the AP

- RADIUS (Remote Access Dial-In User Service) [RFC 2865-2869, RFC 2548]
  - used to carry EAP messages between the AP and the auth server
  - MS-MPPE-Recv-Key attribute is used to transport the session key from the auth server to the AP
  - RADIUS is mandated by WPA and optional for RSN

EAP in Action

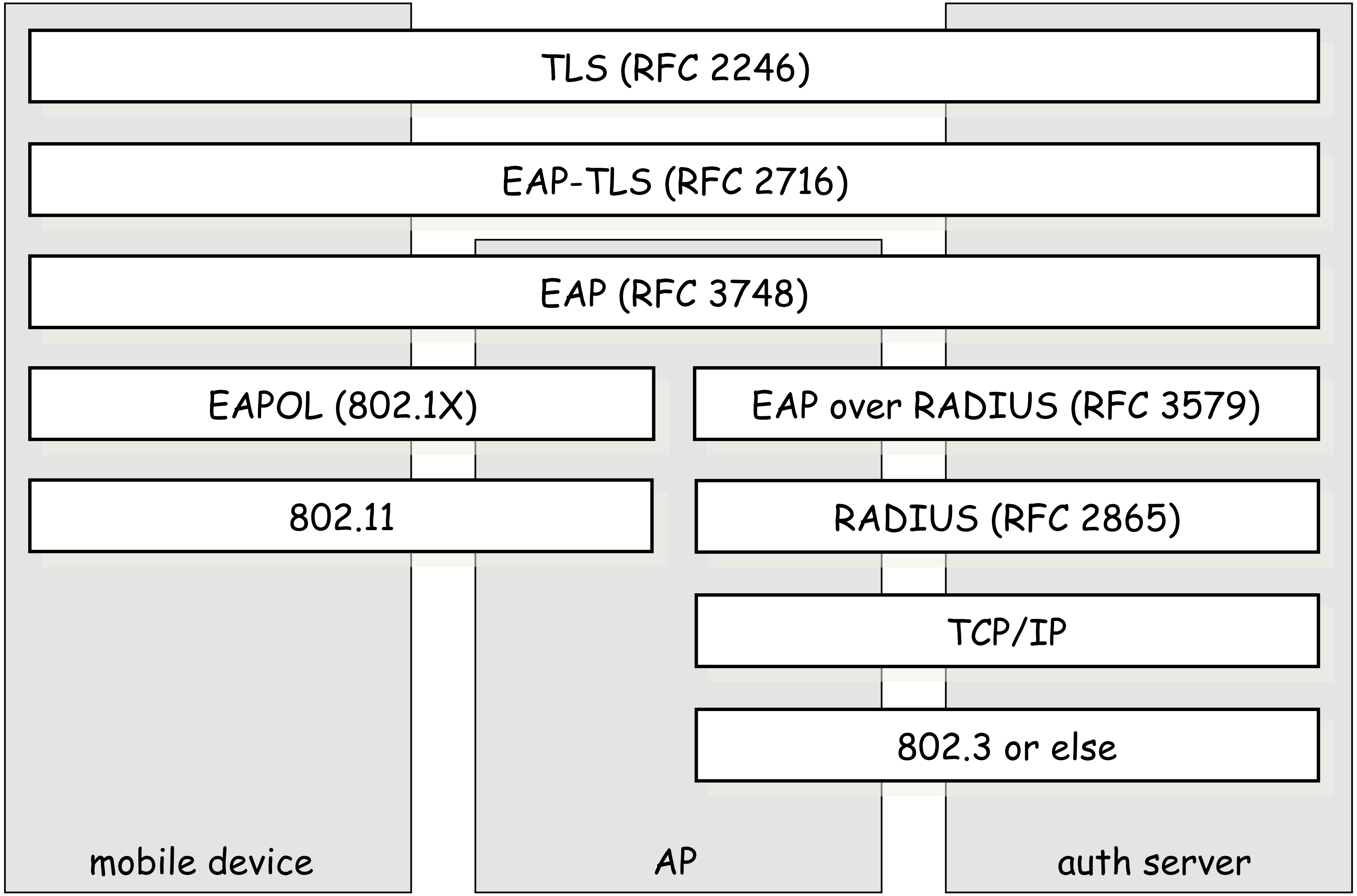


Protocols – LEAP, EAP-TLS, PEAP, EAP-SIM

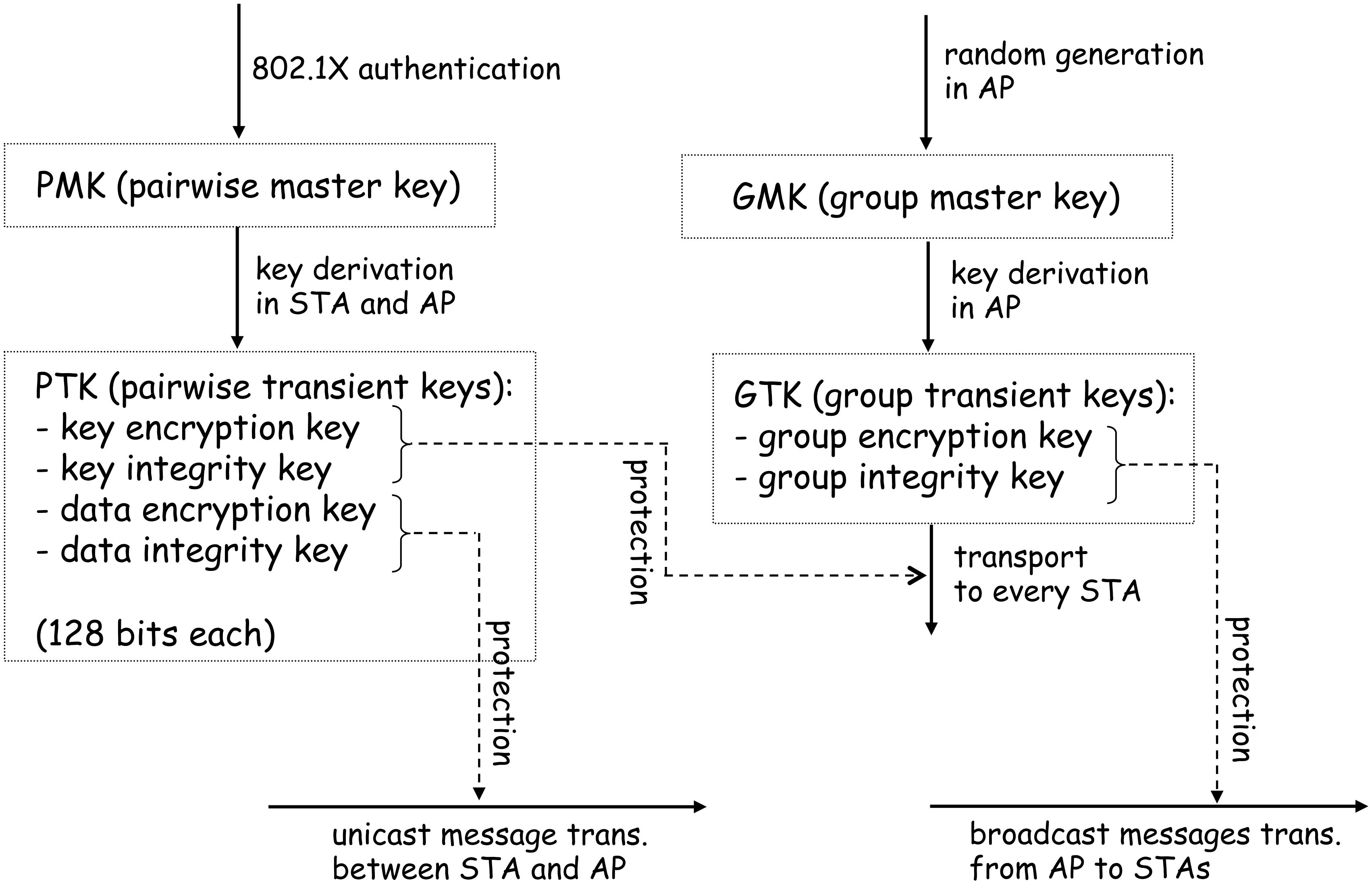
- LEAP (Light EAP)
  - developed by Cisco
  - similar to MS-CHAP extended with session key transport
- EAP-TLS (TLS over EAP)
  - only the TLS Handshake Protocol is used
  - server and client authentication, generation of master secret
  - TLS maser secret becomes the session key
  - mandated by WPA, optional in RSN
- PEAP (Protected EAP)
  - phase 1: TLS Handshake without client authentication
  - phase 2: client authentication protected by the secure channel established in phase 1
- EAP-SIM
  - extended GSM authentication in WiFi context
  - protocol (simplified) :

    STA → AP: EAP res ID (IMSI / pseudonym)
    STA → AP: EAP res (nonce)
    AP: [gets two auth triplets from the mobile operator's AuC]
    AP → STA: EAP req (2*RAND | MIC_{2*Kc} | $\{\text{new pseudonym}\}_{2*Kc}$)
    STA → AP: EAP res (2*SRES)
    AP → STA: EAP success

Summary of the Protocol Architecture



Key Hierarchies



Four-way Handshake

- Objective:
  - prove that AP also knows the PMK (result of authentication)
  - exchange random values to be used in the generation of PTK

- Protocol:

AP : generate ANonce
AP → STA : ANonce | KeyReplayCtr
STA : generate SNonce and compute PTK
STA → AP : SNonce | KeyReplayCtr | MIC_{KIK}
AP : compute PTK, generate GTK, and verify MIC
AP → STA : ANonce | KeyReplayCtr+1 | $\{GTK\}_{KEK}$ | MIC_{KIK}
STA : verify MIC and install keys
STA → AP : KeyReplayCtr+1 | MIC_{KIK}
AP : verify MIC and install keys

MIC_{KIK} : Message Integrity Code (computed by the mobile device using the key-integrity key)
KeyReplayCtr: used to prevent replay attacks

PTK and GTK Computation

- for TKIP

PRF-512(PMK,
"Pairwise key expansion",
MAC1 | MAC2 | Nonce1 | Nonce2) =
= KEK | KIK | DEK | DIK

PRF-256(GMK,
"Group key expansion",
MAC | GNonce) =
= GEK | GIK

- for AES-CCMP

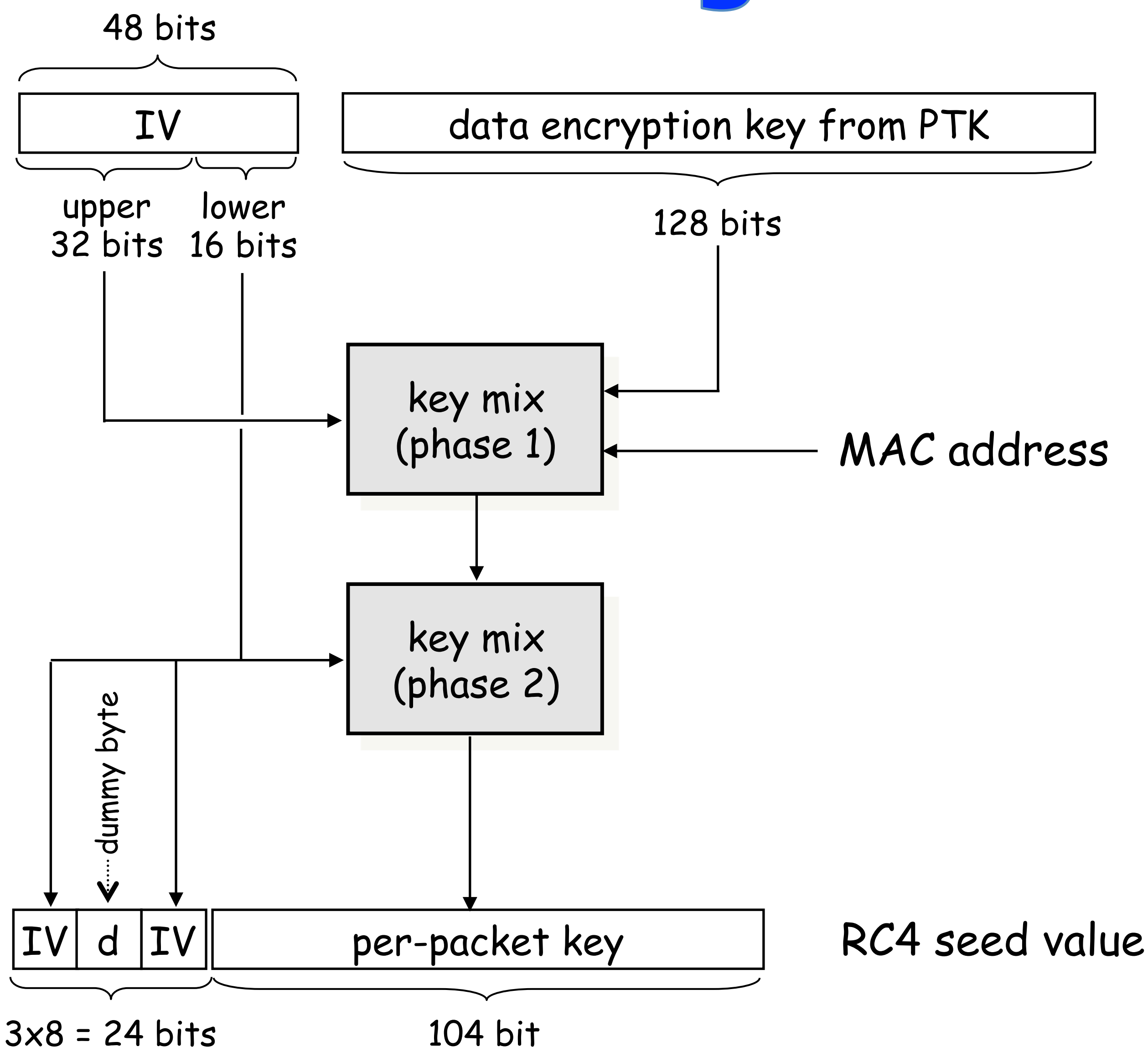
PRF-384(PMK,
"Pairwise key expansion",
MAC1 | MAC2 | Nonce1 | Nonce2) =
= KEK | KIK | DE&IK

PRF-128(GMK,
"Group key expansion",
MAC | GNonce) =
= GE&IK

TKIP

- Runs on old hardware (supporting RC4), but …
- WEP weaknesses are corrected
  - new message integrity protection mechanism called Michael
    - MIC value is added at SDU level before fragmentation into PDUs
    - implemented in the device driver (in software)
  - use IV as replay counter
  - increase IV length to 48 bits in order to prevent IV reuse
  - per-packet keys to prevent attacks based on weak keys

TKIP – Generating RC4 Keys



AES-CCMP

- CCMP means CTR mode and CBC-MAC
  - integrity protection is based on CBC-MAC (using AES)
  - encryption is based on CTR mode (using AES)
- CBC-MAC
  - CBC-MAC is computed over the MAC header, CCMP header, and the MPDU (fragmented data)
  - mutable fields are set to zero
  - input is padded with zeros if length is not multiple of 128 (bits)
  - CBC-MAC initial block:
    - flag (8)
    - priority (8)
    - source address (48)
    - packet number (48)
    - data length (16)
  - final 128-bit block of CBC encryption is truncated to (upper) 64 bits to get the CBC-MAC value
- CTR mode encryption
  - MPDU and CBC-MAC value is encrypted, MAC and CCMP headers are not
  - format of the counter is similar to the CBC-MAC initial block
    - "data length" is replaced by "counter"
    - counter is initialized with 1 and incremented after each encrypted block

Summary on WiFi Security

- Security has always been considered important for WiFi
- Early solution was based on WEP
  - seriously flawed
  - not recommended to use
- The new security standard for WiFi is 802.11i
  - access control model is based on 802.1X
  - flexible authentication based on EAP and upper layer authentication protocols (e.g., TLS, GSM authentication)
  - improved key management
  - TKIP
    - uses RC4 → runs on old hardware
    - corrects WEP's flaws
    - mandatory in WPA, optional in RSN (WPA2)
  - AES-CCMP
    - uses AES in CCMP mode (CTR mode and CBC-MAC)
    - needs new hardware that supports AES

Outline

- Cellular networks
- WiFi LANs
- Bluetooth

Bluetooth

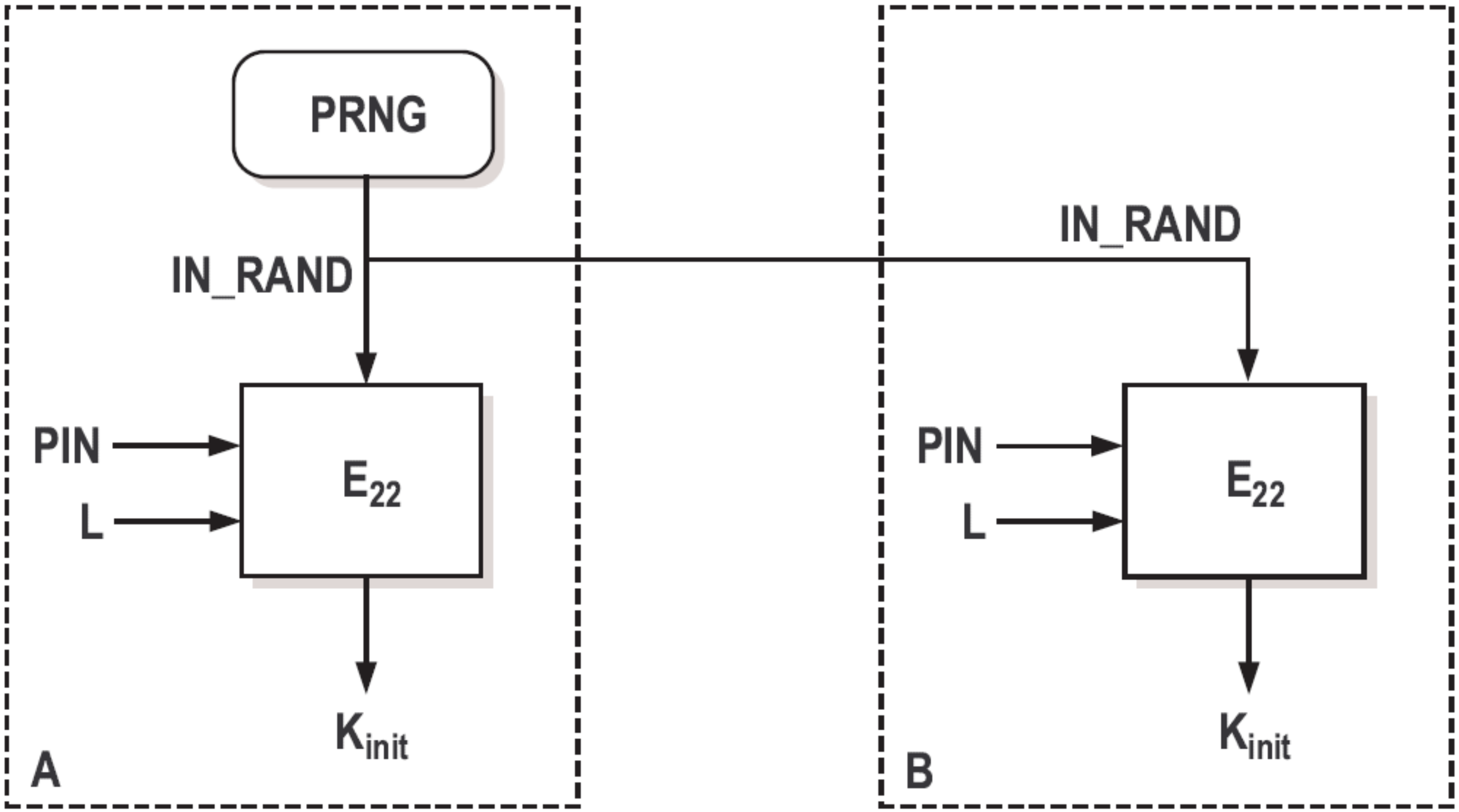
- Short-range communications, master-slave principle

- Eavesdropping is difficult:
  - Frequency hopping
  - Communication is over a few meters only

- Security issues:
  - Authentication of the devices to each other
  - Confidential channel
- Based on secret link key

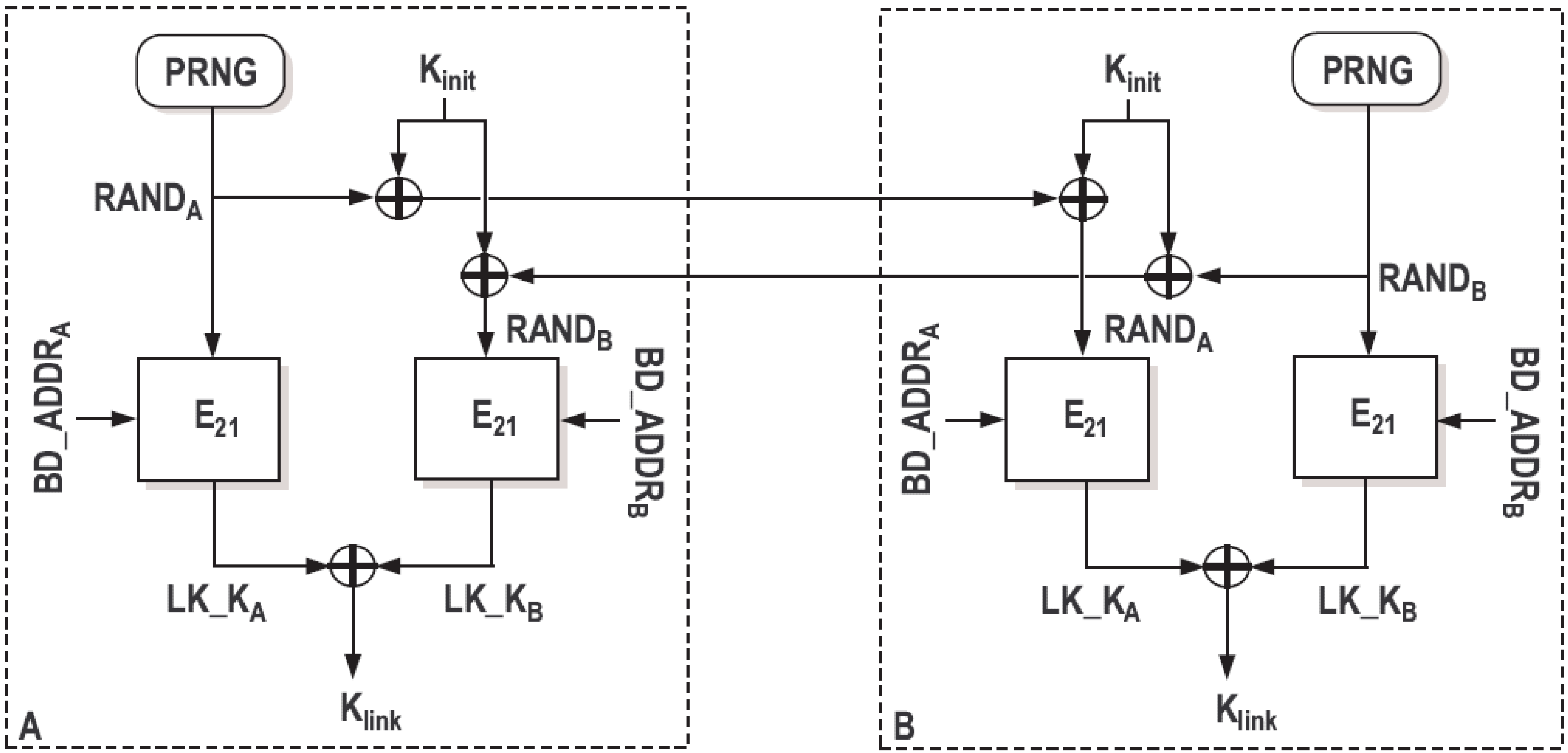
Bluetooth

- When two devices communicate for the



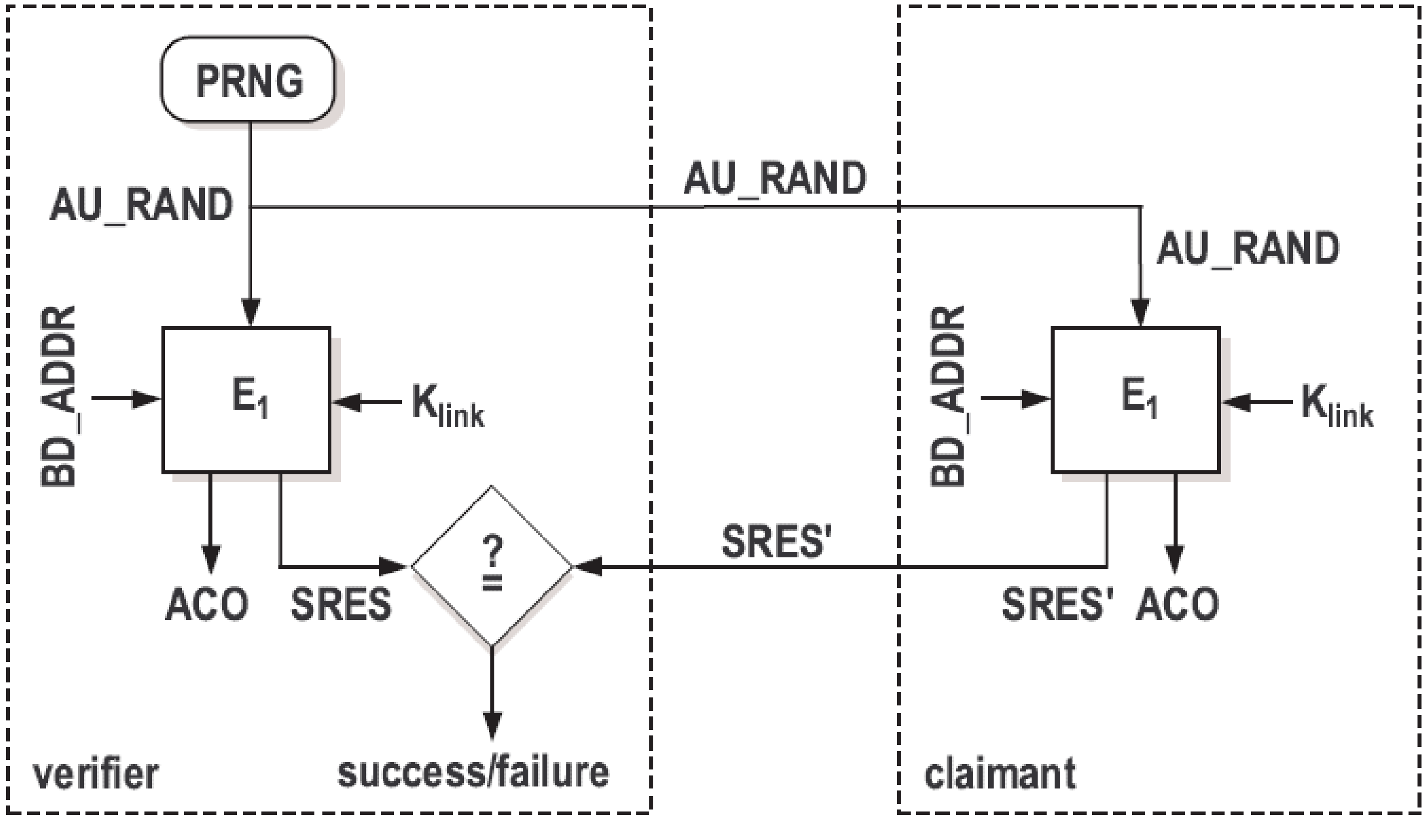
Bluetooth

- Setting up the link key:



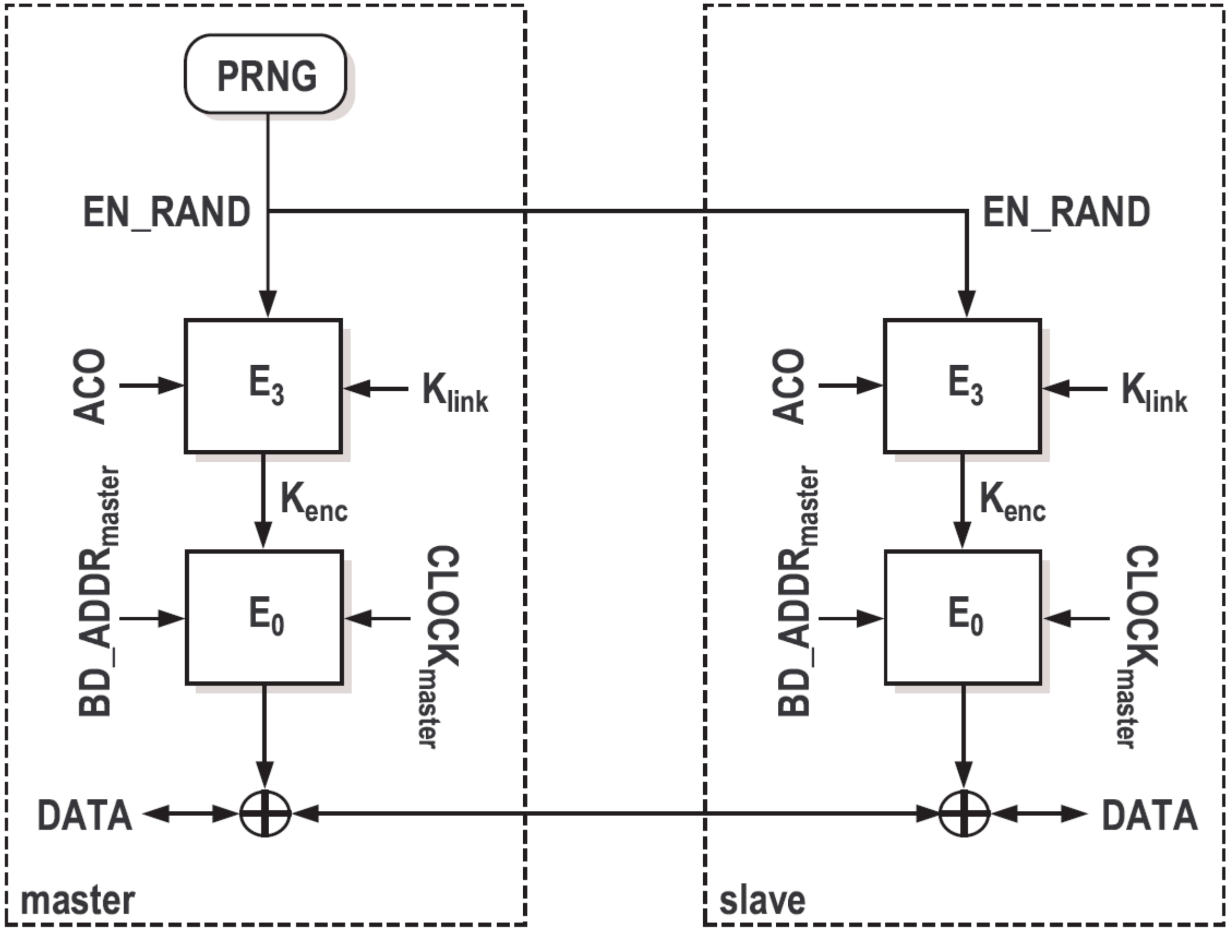
Bluetooth

- The authentication protocol:



Bluetooth

- Generation of the encryption key and the key stream:



Weaknesses

- The strength of the whole system is based on the strength of the PIN:
  - PIN: 4-digit number, easy to try all 10000 possible values.
  - PIN can be cracked off-line.
  - many devices use the default PIN.
- For memory-constrained devices: the link key = the long-term unit key of the device.
- Fixed and unique device addresses: privacy problem.
- Weaknesses in the E_0 stream cipher.

Conclusion

- Security issues of wireless networks:
  - wireless channel: easy to eavesdrop on, jam, overuse
  - Users: usually mobile
- Classical requirements:
  - authentication, confidentiality, integrity, availability
- Location privacy: unique to mobile networks.
- Mobile devices:
  - Limited resources
  - Lack of physical protection
- roaming of users across different networks