Chapter 1: The security of existing wireless networks

cellular networks:
- GSM;
- UMTS;
- WiFi LANs;
- Bluetooth;

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

Chapter outline

1.3.1 Cellular networks
1.3.2 WiFi LANs
1.3.3 Bluetooth
Security and Cooperation in Wireless Networks
Chapter 1: The security of existing wireless networks

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,...)
1.3.1 Cellular networks

GSM security

**Cryptographic algorithms of GSM**

- **Random number**: R
- **User’s secret key**: K_i

**Triplets**

- **Authentication**: R, S, K_i
- **Ciphering algorithm**: A5

- **A3**: subscriber authentication (operator-dependent algorithm)
- **A5**: ciphering/deciphering (standardized algorithm)
- **A8**: cipher generation (operator-dependent algorithm)

**Authentication principle of GSM**

- **Mobile Station**: IMSI (or TMSI)
- **Visited network**: IMSI, Triplets (K_c, R, S)

- **Home network**: K_c, R, S, S’

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

\[ K_c \quad \text{FRAME NUMBER} \]

\[ \text{A5} \quad \text{CIPHERING SEQUENCE} \]

\[ \text{PLAINTEXT SEQUENCE} \quad \oplus \quad \text{CIPHERTEXT SEQUENCE} \]

Sender
(Mobile Station or Network)

Receiver
(Network or Mobile Station)

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
### 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 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

Mobile Station  
Visited Network  
Home Environment

Sequence number $(SQN)_i^{RAND}(i)$

K: User’s secret key

IMSITMSI

Generation of cryptographic material

Authentication vectors

User authentication request $RAND(i) \oplus AUTN(i)$

Verify $AUTN(i) \oplus AUTN(i)$

Compute $RES(i)$

User authentication response $RES(i)$

Compare $RES(i)$ and $XRES(i)$

Select $CK(i)$ and $IK(i)$

Compute $CK(i)$ and $IK(i)$

K

Generation of the authentication vectors

$AUTN \Rightarrow (SQN \oplus ACK) || AMF \parallel MAC$

$AV \Rightarrow RAND \parallel XRES \parallel CK \parallel IK \parallel AUTN$

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

Security and Cooperation in Wireless Networks
Chapter 1: The security of existing wireless networks
1.3.1 Cellular networks
UMTS security

Security and Cooperation in Wireless Networks
Chapter 1: The security of existing wireless networks
13/60

Security and Cooperation in Wireless Networks
Chapter 1: The security of existing wireless networks
1.3.1 Cellular networks
UMTS security

Security and Cooperation in Wireless Networks
Chapter 1: The security of existing wireless networks
14/60
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*

<table>
<thead>
<tr>
<th>SQN</th>
<th>AMF</th>
<th>OP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>rotate by r1</td>
<td>OP</td>
</tr>
<tr>
<td></td>
<td>c1</td>
<td>EK</td>
</tr>
<tr>
<td></td>
<td>f1</td>
<td>f1*</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RAND</th>
<th>OP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>rotate by r2</td>
</tr>
<tr>
<td></td>
<td>c2</td>
</tr>
<tr>
<td></td>
<td>f2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OP</th>
</tr>
</thead>
<tbody>
<tr>
<td>rotate by r3</td>
</tr>
<tr>
<td>c3</td>
</tr>
<tr>
<td>f3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OP</th>
</tr>
</thead>
<tbody>
<tr>
<td>rotate by r4</td>
</tr>
<tr>
<td>c4</td>
</tr>
<tr>
<td>f4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OP</th>
</tr>
</thead>
<tbody>
<tr>
<td>rotate by r5</td>
</tr>
<tr>
<td>c5</td>
</tr>
<tr>
<td>f5*</td>
</tr>
</tbody>
</table>

OP: operator-specific parameter
t1, ..., r5: fixed rotation constants
c1, ..., c5: fixed addition constants

E_k: Rijndael block cipher with128 bits text input and 128 bits key

Signalling integrity protection method

<table>
<thead>
<tr>
<th>SIGNALLING MESSAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>IK</td>
</tr>
<tr>
<td>MAC-I</td>
</tr>
</tbody>
</table>

Sender
(Mobile Station or Radio Network Controller)

<table>
<thead>
<tr>
<th>FRESH</th>
<th>COUNT-I</th>
<th>DIRECTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>FRESH</td>
<td>COUNT-I</td>
<td>DIRECTION</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SIGNALLING MESSAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>IK</td>
</tr>
<tr>
<td>XMAC-I</td>
</tr>
</tbody>
</table>

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

COUNT || BEARER || DIRECTION || 0...0

CK ⊕ KM → KASUMI

Register

BLKCNT=0 → BLKCNT=1 → BLKCNT=2 → BLKCNT=BLOCKS−1

CK → KASUMI CK → KASUMI CK → KASUMI CK → KASUMI

Details of Kasumi

KL, KO, KI : subkeys used at ith round
S7, S9: S-boxes

Fig. 1: KASUMI

Fig. 2: FO Function

Fig. 3: FI Function

Fig. 4: FL Function

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
Chapter outline

1.3.1 Cellular networks
1.3.2 WiFi LANs
1.3.3 Bluetooth

Introduction to WiFi

- Beacon
- MAC header
- Timestamp
- Beacon interval
- Capability info
- SSID (network name)
- Supported data rates
- Radio parameters
- Power slave flags
Introduction to WiFi

WEP – Wired Equivalent Privacy

- part of the IEEE 802.11 specification
- goal
  - make the WiFi network \textit{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 (eK(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

\[ \text{message + ICV} \]

\[ \text{IV} \quad \text{RC4} \quad \text{K} \]

\[ \text{encode} \]

\[ \text{IV} \quad \text{message + ICV} \]

\[ \text{IV} \quad \text{RC4} \quad \text{K} \]

\[ \text{decode} \]

\[ \text{message + ICV} \]

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 \( \rightarrow \) 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

[Diagram showing the key change process with time progression and active key identification]
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 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 \rightarrow STA: r
  STA \rightarrow AP: IV | r \oplus K
  ...
  where K is a 128 bit RC4 output 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 \rightarrow attacker: r'
  attacker \rightarrow 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:
    \[ \text{CRC}(X + Y) = \text{CRC}(X) + \text{CRC}(Y) \]
    - attacker observes \((M | \text{CRC}(M)) + K\) where \(K\) is the RC4 output
    - for any \(\Delta M\), the attacker can compute \(\text{CRC}(\Delta M)\)
    - hence, the attacker can compute:
      \[
      ((M | \text{CRC}(M)) + \Delta M | \text{CRC}(\Delta M)) + K =
      ((M + \Delta M) | \text{CRC}(M + \Delta M)) + K
      \]

WEP flaws – Confidentiality

- IV reuse
  - IV space is too small
    - IV size is only 24 bits \(\Rightarrow\) 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 \(\Rightarrow\) 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 \(\Rightarrow\) 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
  \(\Rightarrow\) 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)

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

Protocol – 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 master 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 | MIC²,Kc | \{new pseudonym\}²,Kc)
    - STA → AP: EAP res (2*SRES)
    - AP → STA: EAP success
Summary of the protocol architecture

<table>
<thead>
<tr>
<th>TLS (RFC 2246)</th>
<th>EAP-TLS (RFC 2716)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EAP (RFC 3748)</td>
<td>EAP over RADIUS (RFC 3579)</td>
</tr>
<tr>
<td>EAPOL (802.1X)</td>
<td>RADIUS (RFC 2865)</td>
</tr>
<tr>
<td>802.11</td>
<td>TCP/IP</td>
</tr>
</tbody>
</table>

Key hierarchies

PMK (pairwise master key)
- key derivation in STA and AP
- PTK (pairwise transient keys):
  - key encryption key
  - key integrity key
  - data encryption key
  - data integrity key
  (128 bits each)

GMK (group master key)
- key derivation in AP
- GTK (group transient keys):
  - group encryption key
  - group integrity key

802.1X authentication
- random generation in AP

unicast message trans. between STA and AP

broadcast messages trans. from AP to STAs
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:**

  \[
  \begin{align*}
  &\text{AP : generate ANonce} \\
  &\text{AP} \rightarrow \text{STA : ANonce | KeyReplayCtr} \\
  &\text{STA : generate SNonce and compute PTK} \\
  &\text{STA} \rightarrow \text{AP : SNonce | KeyReplayCtr | MIC}_{KIK} \\
  &\text{AP : compute PTK, generate GTK, and verify MIC} \\
  &\text{AP} \rightarrow \text{STA : ANonce | KeyReplayCtr+1 | \{GTK\}_{KEK} | MIC}_{KIK} \\
  &\text{STA : verify MIC and install keys} \\
  &\text{STA} \rightarrow \text{AP : KeyReplayCtr+1 | MIC}_{KIK} \\
  &\text{AP : verify MIC and install keys}
  \end{align*}
  \]

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**

  \[
  \begin{align*}
  &\text{PRF-512( PMK,} \\
  &\quad \text{"Pairwise key expansion",} \\
  &\quad \text{MAC1 | MAC2 | Nonce1 | Nonce2 ) =} \\
  &\quad \text{= KEK | KIK | DEK | DK} \\
  &\text{PRF-256( GMK,} \\
  &\quad \text{"Group key expansion",} \\
  &\quad \text{MAC | GNonce ) =} \\
  &\quad \text{= GEK | GIK}
  \end{align*}
  \]

- **for AES-CCMP**

  \[
  \begin{align*}
  &\text{PRF-384( PMK,} \\
  &\quad \text{"Pairwise key expansion",} \\
  &\quad \text{MAC1 | MAC2 | Nonce1 | Nonce2 ) =} \\
  &\quad \text{= KEK | KIK | DE&IK} \\
  &\text{PRF-128( GMK,} \\
  &\quad \text{"Group key expansion",} \\
  &\quad \text{MAC | GNonce ) =} \\
  &\quad \text{= GE&IK}
  \end{align*}
  \]
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
Chapter outline

1.3.1 Cellular networks
1.3.2 WiFi LANs
1.3.3 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 first time:
  - Set up the temporary initialization key.

Setting up the link key:
Bluetooth

- The authentication protocol:

![Authentication Protocol Diagram]

- Generation of the encryption key and the key stream:

![Encryption Key Generation Diagram]
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 E₀ 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