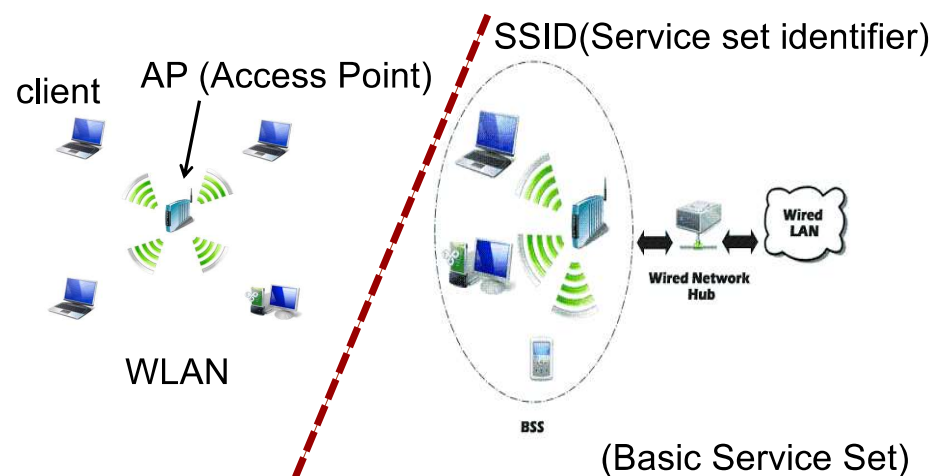


Wireless Security: WEP and GSM

Wireless Local Area Network (WLAN)



1

WLAN security standards

- **WEP:** Wired Equivalent Privacy (introduced in 1999)
 - Original 802.11 standard for WLAN
 - Can be broken within a few minutes
- **WPA:** Wi-Fi Protected Access (2003)
 - Draft 802.11i standard
 - Based on the same hardware for WEP
- **WPA2:** Wi-Fi Protected Access II (2004)
 - Full 802.11i standard
 - Requires more powerful hardware
- **WPA3:** Wi-Fi Protected Access III (2018)

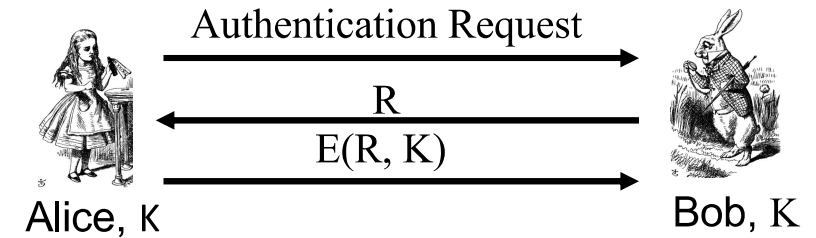
WEP

- WEP — Wired Equivalent Privacy
- The stated goal of WEP was to **make wireless LAN as secure as a wired LAN**
- According to Tanenbaum:
 - “The 802.11 standard prescribes a **data link-level** security protocol called WEP (Wired Equivalent Privacy), which is designed to make the security of a wireless LAN as good as that of a wired LAN. Since the default for a wired LAN is no security at all, this goal is easy to achieve, and WEP achieves it as we shall see.”

WEP router

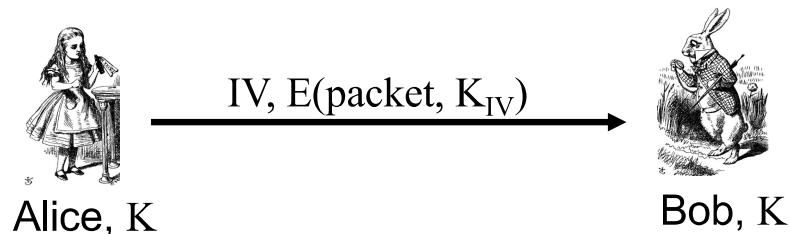


WEP Authentication



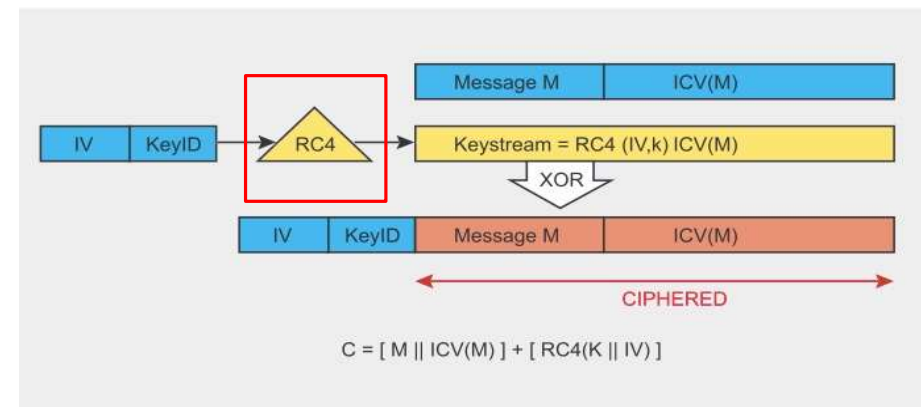
- Bob is **wireless access point**
- Key K shared by access point and **all users**
 - Key K seldom (if ever) changes

WEP Encryption

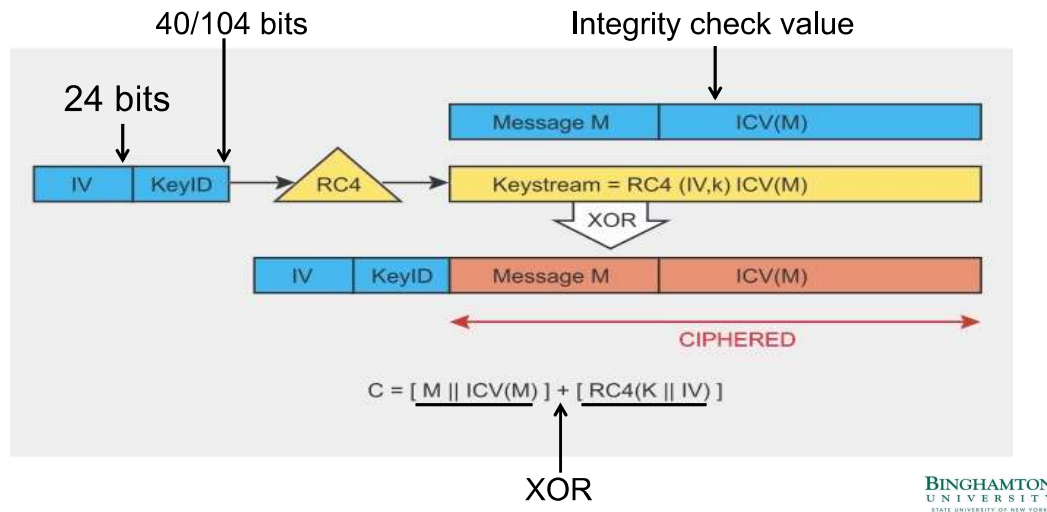


- $K_{IV} = (IV, K)$
 - Encrypted using RC4 stream cipher
 - RC4 key is K with 3-byte IV pre-pended
- Note that the IV is known to Trudy
- Goal: Encrypt packets with distinct keys

WEP Encryption



WEP Encryption



WEP Issues

- WEP uses RC4 cipher for confidentiality
 - RC4 is considered a strong cipher
 - But WEP introduces a subtle flaw...
 - ...making cryptanalytic attacks feasible
- WEP uses **CRC** (Circular Redundancy Check) for “integrity”
 - Should have used a MAC or HMAC instead
 - CRC is for error detection, not crypto integrity
 - Everyone in security knows **NOT** to use CRC for this...

(1) WEP Integrity Problems

- WEP “integrity” gives no crypto integrity
 - CRC is linear, so is stream cipher (XOR)
 - Trudy can change **ciphertext and CRC** so that checksum remains correct
 - Then Trudy’s introduced errors go undetected
 - Requires no knowledge of the plaintext!
- CRC does **not** provide a cryptographic integrity check
 - CRC designed to detect random errors
 - Not able to detect intelligent changes

More WEP Integrity Issues

- Suppose Trudy knows destination IP
- Then Trudy also knows keystream used to encrypt IP address, since...
 - ... **C** = destination IP address \oplus **keystream**
- Then Trudy can replace **C** with...
 - ... **C'** = Trudy’s IP address \oplus **keystream**
 - How?**
- And change the CRC so no error detected!
 - Then what happens??
- Moral: Big problem when integrity fails

(2) WEP confidentiality issue: WEP Key Repeat

- Recall WEP uses a long-term secret key: K
- RC4 is a stream cipher, so each packet must be encrypted using a different key
 - Initialization Vector (IV) sent with packet
 - Sent in the clear, that is, IV is **not** secret
- Actual RC4 key for packet is (IV, K)
 - That is, IV is **pre-pended** to long-term key K

WEP IV Issues

- WEP uses 24-bit (3 byte) IV
 - Each packet gets a new IV
 - Key: IV pre-pended to long-term key, K
- Long term key K seldom changes
- If long-term key and IV are the same, then the same keystream is used
 - This really bad!
 - Why?



WEP IV Issues

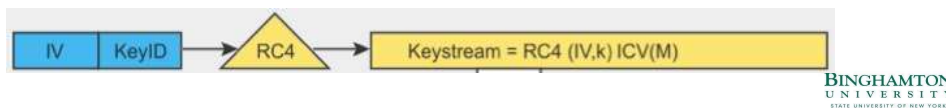
- Assume 1500 byte packets, 11 Mbps link
- Suppose IVs generated in sequence
 - Since $1500 \cdot 8 / (11 \cdot 10^6) \cdot 2^{24} = 18,000$ seconds...
 - ...an IV must repeat in about 5 hours
- Again, repeated IV (with same K) is bad!

(3) WEP : Another Active Attack

- Suppose Trudy can **insert traffic** and **observe corresponding ciphertext**
 - Then she knows the keystream for some IV
 - She can decrypt any packet(s) that uses that IV
- If Trudy does this many times, she can then decrypt data for lots of IVs
 - Remember, IV is sent in the clear

(4) Cryptanalytic Attack

- 3-byte IV pre-pended to key
- Denote the RC4 key **bytes**...
 - ...as $K_0, K_1, K_2, K_3, K_4, K_5, \dots$
 - Where $IV = (K_0, K_1, K_2)$, which Trudy knows
 - Trudy wants to find $K = (K_3, K_4, K_5, \dots)$
- FMS attack
 - Designed by Fluhrer, Mantin and Shamir
 - With **certain IVs**, an attacker knowing **the first byte of the keystream** and **the first m bytes of the key** can derive the **$(m+1)$ -th byte of the key**



GSM (In)Security

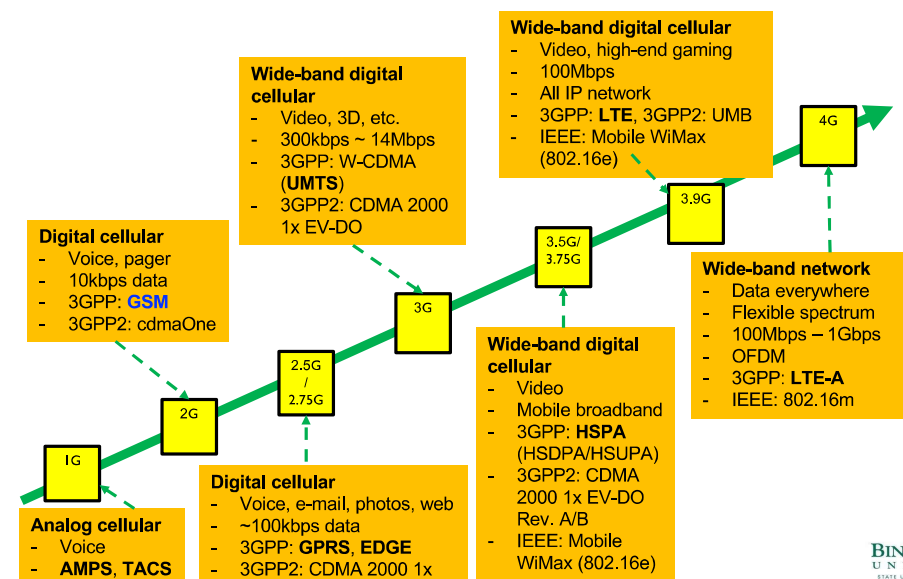


WEP Conclusions

- Many attacks are practical
 - AirSnort
 - AirCrack
 - WepLab
 - ...
- Attacks have been used to recover keys and break real WEP traffic
- How to prevent WEP attacks?
 - Don't use WEP
 - Good alternatives: WPA, WPA2, etc.

Crack 128-bit WEP key in less than 10 minutes

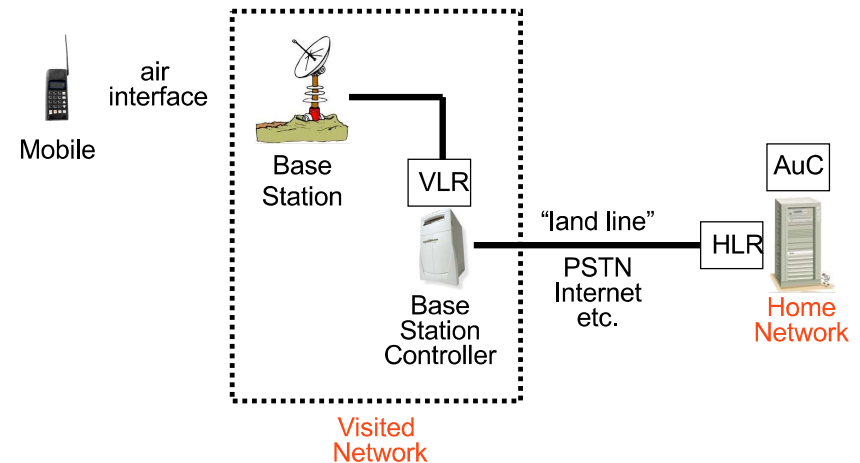
Evolution of cellular communications standards



Cell Phones

- First generation cell phones
 - Brick-sized, analog, few standards
 - Little or *no* security, and susceptible to **cloning**
- Second generation cell phones: **GSM**
 - Began in 1982 as “*Groupe Spécial Mobile*”
 - Now, Global System for Mobile Communications
- Third generation: UMTS (Universal Mobile Telecommunications System)
- 4th gen (LTE)
- 5th gen, ...

GSM System Overview



GSM System Components

- Mobile phone
 - Contains SIM (Subscriber Identity Module)
- SIM is the **security module**
 - IMSI (International Mobile Subscriber ID)**
 - User key: K_i (128 bits)**
 - Tamper resistant (smart card)**
 - PIN activated (usually not used)**



GSM System Components

- Visited network** — network where mobile is currently located
 - Base station — one “cell”
 - Base station controller — manages many cells
 - VLR** (Visitor Location Register) — info on all visiting mobiles currently in the network
- Home network** — “home” of the mobile
 - HLR** (Home Location Register) — keeps track of most recent **location** of mobile
 - AuC** (Authentication Center) — has **IMSI** and **K_i**

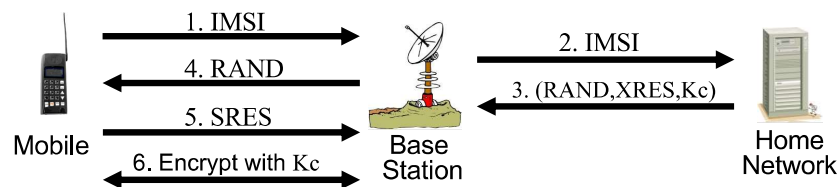
GSM Security Goals

- Primary design goals
 - **Make GSM as secure as ordinary telephone**
 - **Prevent phone cloning**
- Not designed to resist active attacks
 - At the time this seemed infeasible
 - Today such attacks are feasible...
- Designers considered biggest threats to be
 - Insecure billing
 - Other low-tech attacks

GSM Security Features

- **Anonymity**
 - Intercepted traffic does not identify user
 - Not so important to phone company
- **Authentication**
 - Necessary for proper billing
 - Very, very important to phone company!
- **Confidentiality**
 - Confidentiality of calls over the air interface
 - Not important to phone company
 - May be important for marketing

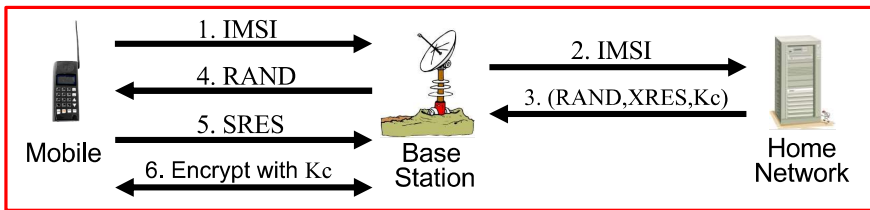
GSM Security Protocol



GSM: Anonymity

- IMSI used to initially identify caller
- Then TMSI (**T**emporary Mobile Subscriber ID) used
 - TMSI changed frequently
 - TMSI's encrypted when sent
- Not a strong form of anonymity
- But probably sufficient for most uses

GSM: Authentication

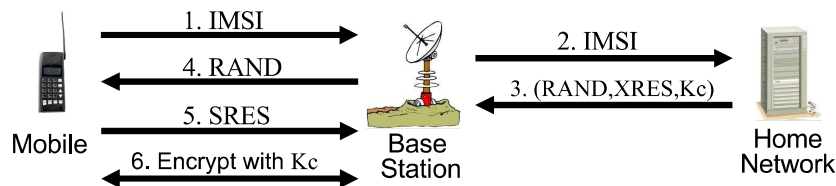


- Caller is authenticated to base station (not mutual)
- Authentication via **challenge-response**
 - Home network generates RAND and computes $XRES = A3(RAND, Ki)$ where A3 is a hash function and Ki the mobile's 128 bit user key
 - Then (RAND, XRES) sent to base station
 - Base station sends **challenge** RAND to mobile
 - Mobile's **response** is $SRES = A3(RAND, Ki)$
 - Base station verifies $SRES = XRES$

GSM: Confidentiality

- Data encrypted with stream cipher
 - Error rate estimated at about 1/1000
- Encryption key Kc
 - Home network computes $Kc = A8(RAND, Ki)$ where A8 is a hash
 - Then Kc sent to base station with (RAND, XRES)
 - Mobile computes $Kc = A8(RAND, Ki)$
 - Keystream generated from $A5(Kc)$
- **Note:** Ki never leaves home network!

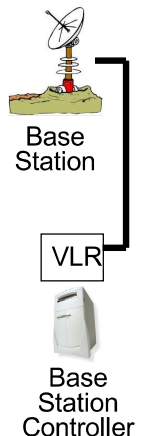
GSM Security



- $SRES = XRES = A3(RAND, Ki)$, $Kc = A8(RAND, Ki)$
- **SRES and Kc must be uncorrelated**
 - Even though both are derived from RAND and Ki
- Must not be possible to deduce Ki from known RAND/SRES pairs (known plaintext attack)
- Must not be possible to deduce Ki from chosen RAND/SRES pairs (chosen plaintext attack)
 - With possession of SIM, attacker can choose RAND's

GSM Insecurity – Crypto Flaws

- Hash used for A3/A8 is **COMP128**
 - Broken by 160,000 chosen plaintexts
 - With SIM, can get Ki in 2 to 10 hours
- Encryption between mobile and base station but **no encryption** from base station to base station controller
 - Often transmitted over microwave link
- Encryption algorithm A5/1
 - Feasible attacks on A5/1 are known

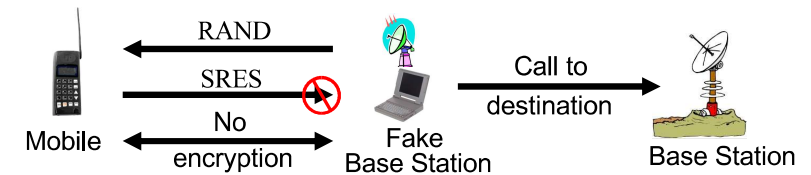


GSM Insecurity – SIM Card

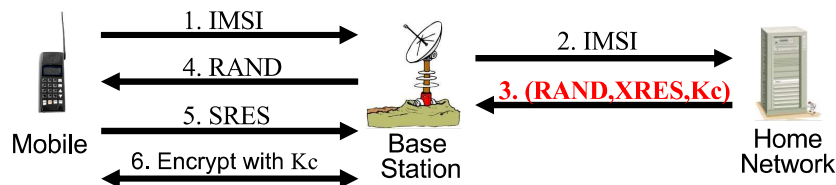
- Attacks on SIM card
 - Optical Fault Induction** — could attack SIM with a flashbulb to recover Ki
 - Partitioning Attacks** — using timing and power consumption, could recover Ki with only 8 adaptively chosen “plaintexts”

GSM Insecurity - Base Station

- Fake base station** exploits two flaws
 - Encryption not automatic
 - Base station not authenticated



GSM Insecurity - Replay



- Can replay triple: (RAND, XRES, Kc)
 - One compromised triple gives attacker a key Kc that is valid forever
 - No replay protection here

GSM Conclusion

- Did GSM achieve its goals?
 - Eliminate cloning? **Yes, as a practical matter**
 - Make air interface as secure as PSTN? **Perhaps...**
- But design goals were clearly too limited
- GSM insecurities — weak crypto, SIM issues, fake base station, replay, etc.
- GSM a (modest) security success?

Protocols Summary

- Generic authentication protocols
 - Protocols are subtle!
- SSH
- SSL
- IPSec
- Kerberos
- Wireless: GSM and WEP