Encryptions in Different Network Layers

- Application Layer
  - Secure Email
- Transport Layer
  - TLS
- Network Layer
  - IP SEC
- Physical Layer
  - IEEE 802.11 WiFi Security
  - 4G/5G Security

Why do we need security in different layers?

Mutual Authentication and Shared Symmetric Key Derivation: Brief History

1. Wired Equivalent Privacy (**WEP**): Designed in 1999, but attacked and hacked in 2001
2. WiFi Protected Access (**WPA1**): **Developed in 2003**, introducing message integrity checks and avoid attacks that allowed a user to infer encryption keys after observing the stream of encrypted messages for a period of time
3. **WPA2 (2004)**: Mandated the use of AES
4. **WPA3 (2018):** Solve an attack to WPA2 when we reuse a nounce.

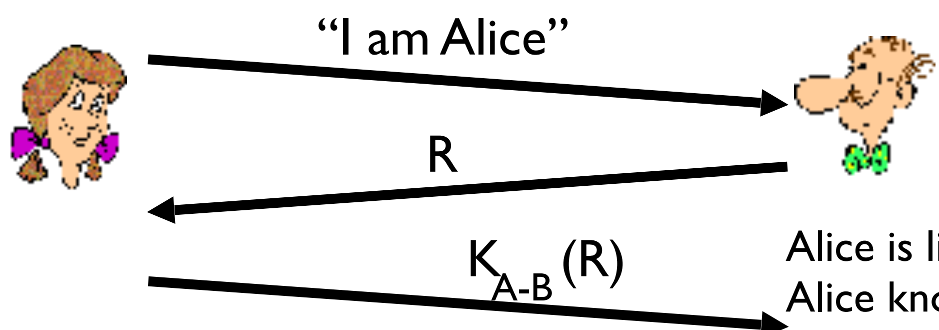
WEP design goals

- **Symmetric key crypto**
  - confidentiality
  - end host authorization
  - data integrity
- **Self-synchronizing: each packet separately encrypted**
  - given encrypted packet and key, can decrypt; can continue to decrypt packets when preceding packet was lost (unlike Cipher Block Chaining (CBC) in block ciphers)
- **Efficient**
  - implementable in hardware or software

End-point authentication w/ nonce

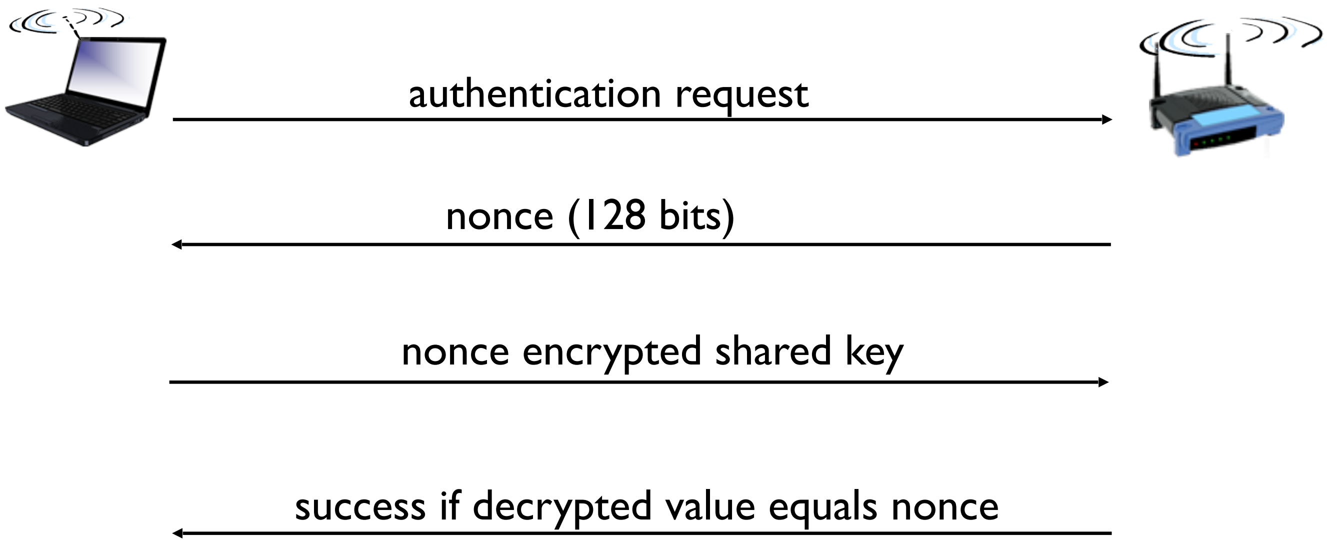
Nonce: number (R) used only *once –in-a-lifetime*

How to prove Alice "live": Bob sends Alice *nonce*, R. Alice must return R, encrypted with shared secret key



Alice is live, and only Alice knows key to encrypt nonce, so it must be Alice!

WEP authentication



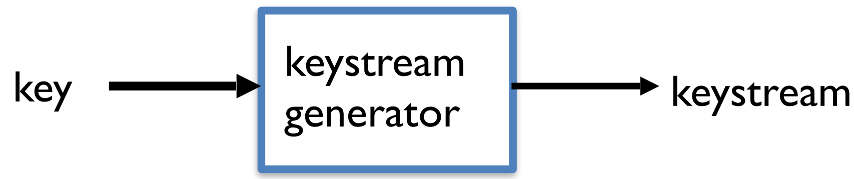
Notes:

- not all APs do it, even if WEP is being used
- AP indicates if authentication is necessary in beacon frame
- done before association

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

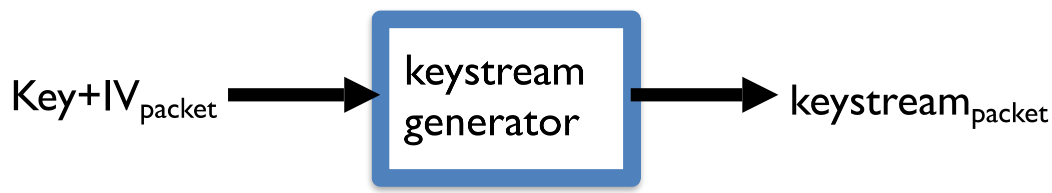
Review: Symmetric Stream Ciphers



- *combine each byte of keystream with byte of plaintext to get ciphertext:*
  - $m(i)$ = i^{th} unit of message
  - $ks(i)$ = i^{th} unit of keystream
  - $c(i)$ = i^{th} unit of ciphertext
  - $c(i) = ks(i) \oplus m(i)$ ($\oplus$ = exclusive or)
  - $m(i) = ks(i) \oplus c(i)$
- **WEP uses RC4**

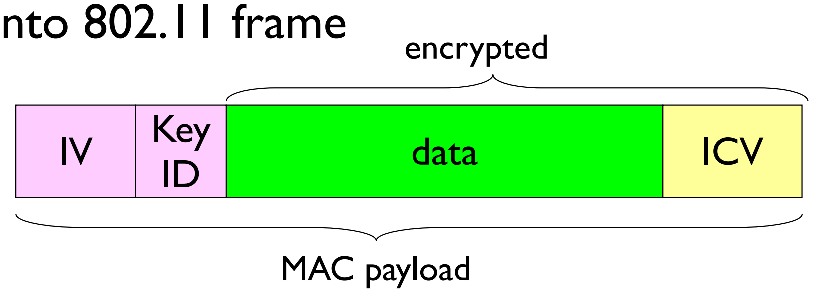
Stream cipher and packet independence

- Recall design goal: each packet separately encrypted
- If for frame n+1, use keystream from where we left off for frame n, then each frame is not separately encrypted
  – need to know where we left off for packet n
- **WEP approach:** initialize keystream with key + new IV for each packet:

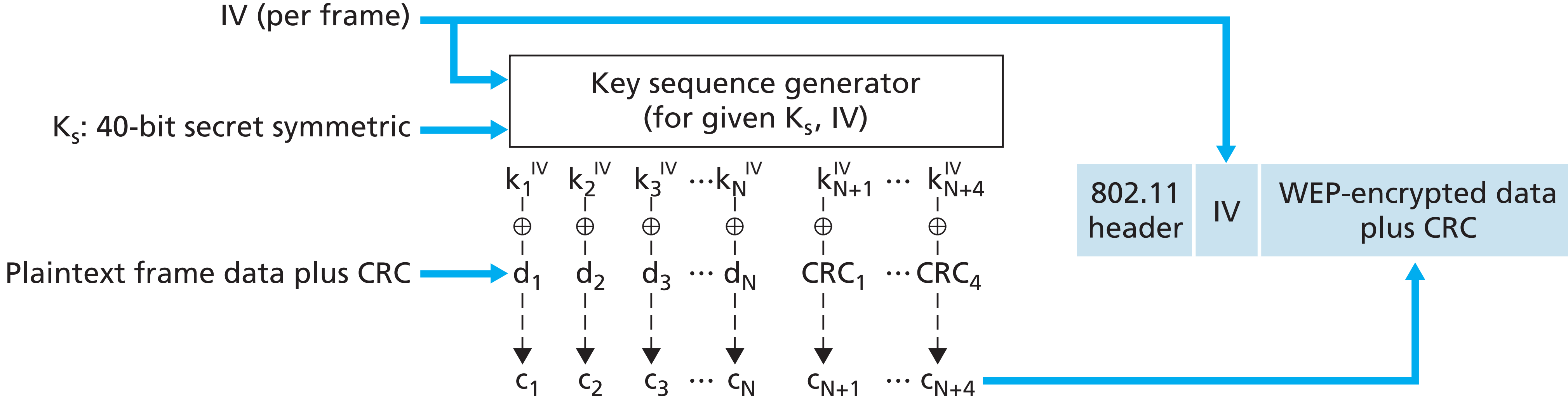


WEP Encryption (1)

- Sender calculates Integrity Check Value (ICV) over data
  - Four-byte hash/CRC for data integrity
- Each side has 40-bit shared key
- Sender creates 24-bit initialization vector (IV), appends to key: gives 64-bit key
- Sender also appends keyID (in 8-bit field)
- 64-bit key inputted into pseudo random number generator to get keystream
- Data in frame + ICV is encrypted with RC4:
  - B\bytes of keystream are XORed with bytes of data & ICV
  - IV & keyID are appended to encrypted data to create payload
  - payload inserted into 802.11 frame

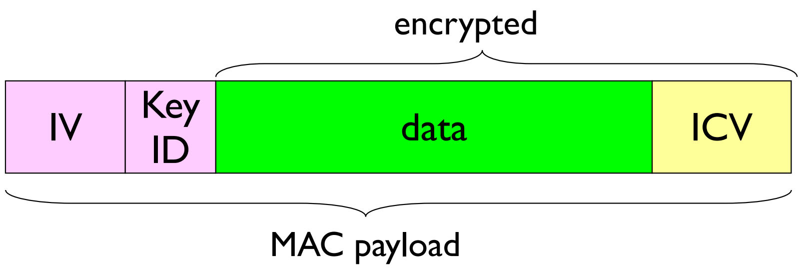


WEP Encryption (2)



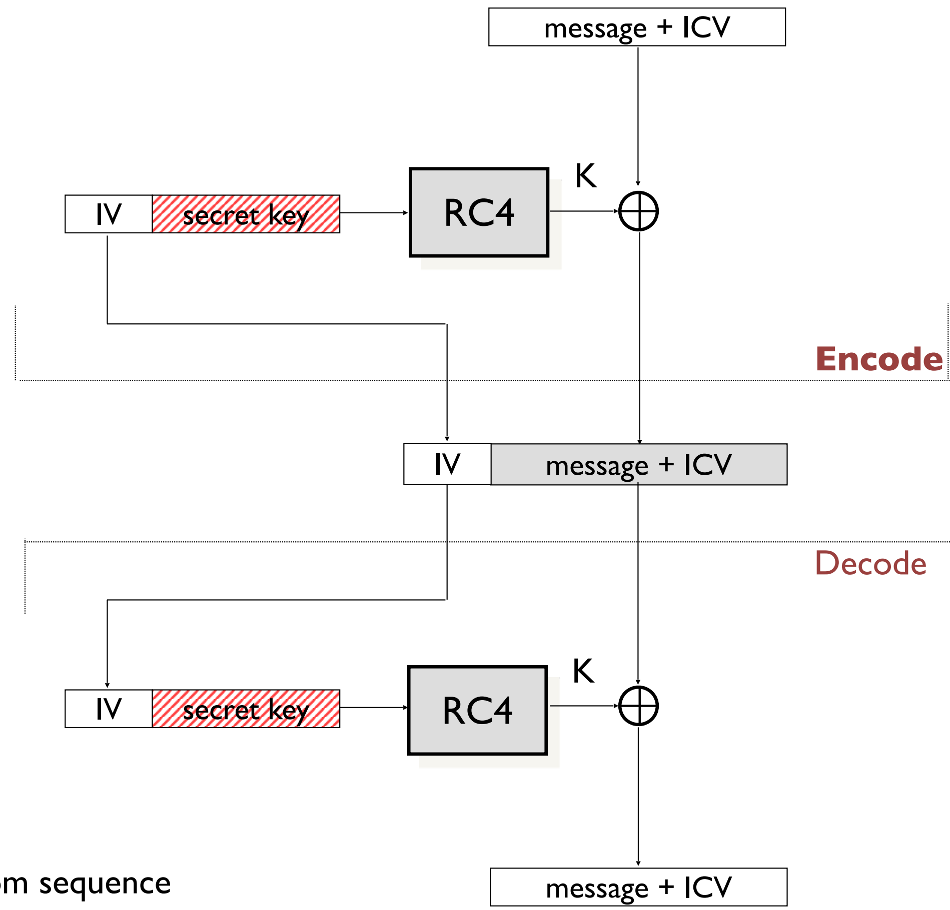
Note: New IV for each frame

WEP decryption overview



- Receiver extracts IV
- Inputs IV, shared secret key into pseudo random generator, gets keystream
- XORs keystream with encrypted data to decrypt data + ICV
- Verifies integrity of data with ICV
  - **Note:** message integrity approach used here is different from MAC (message authentication code) and signatures (using PKI).

WEP – Message Confidentiality and Integrity



K: pseudo-random sequence

Breaking 802.11 WEP encryption

security hole:

- 24-bit IV, one IV per frame, → IV's eventually reused
- IV transmitted in plaintext → IV reuse detected

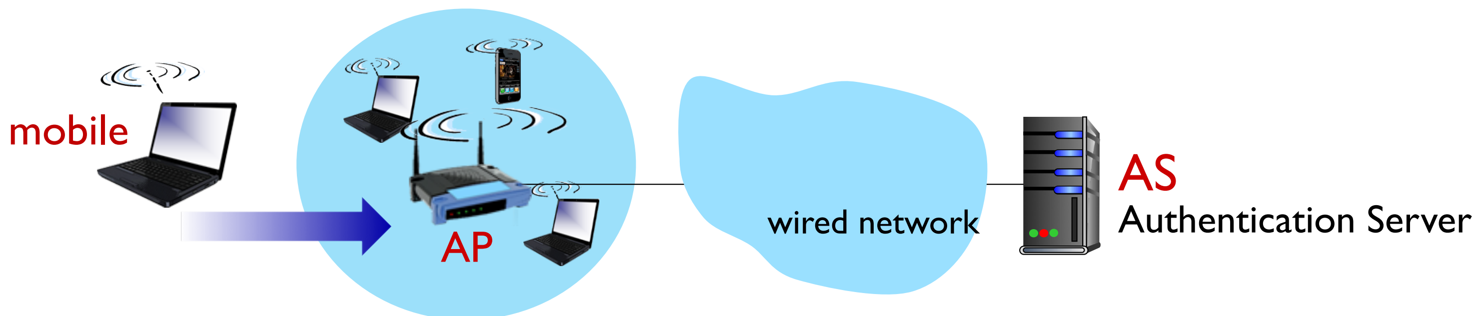
attack:

- Trudy causes Alice to encrypt known plaintext d_1 d_2 d_3 d_4 …
- Trudy sees: $c_i = d_i$ XOR k_i^{IV}
- Trudy knows c_i and d_i, so can compute k_i^{IV}
- Trudy knows encrypting key sequence k_1^{IV} k_2^{IV} k_3^{IV} …
- Next time IV is used, Trudy can decrypt!

802.11i: Improved Security

- Numerous (stronger) forms of encryption possible
- Provides key distribution
- Uses authentication server separate from access point

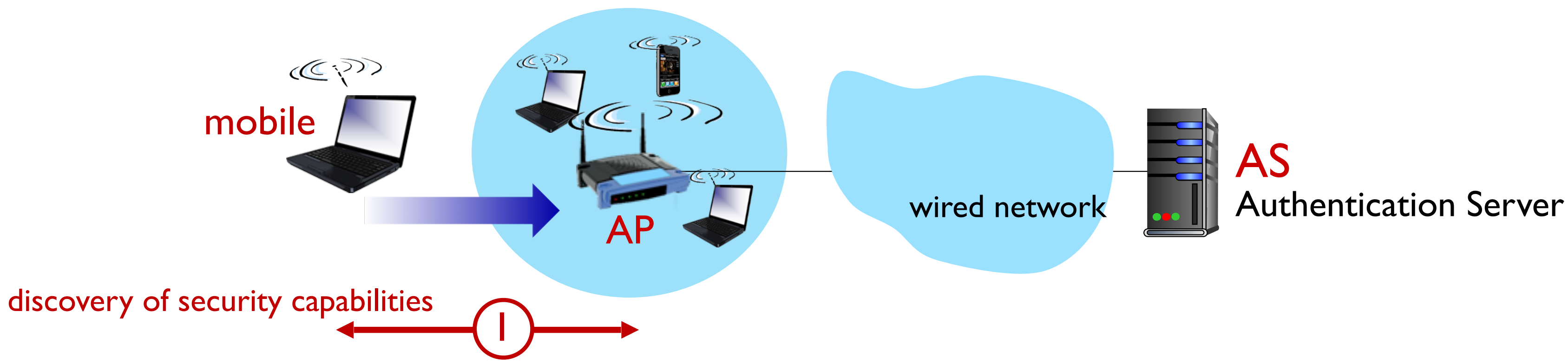
802.11: Authentication, Encryption



Arriving mobile must:

- associate with access point: (establish) communication over wireless link
- authenticate to network

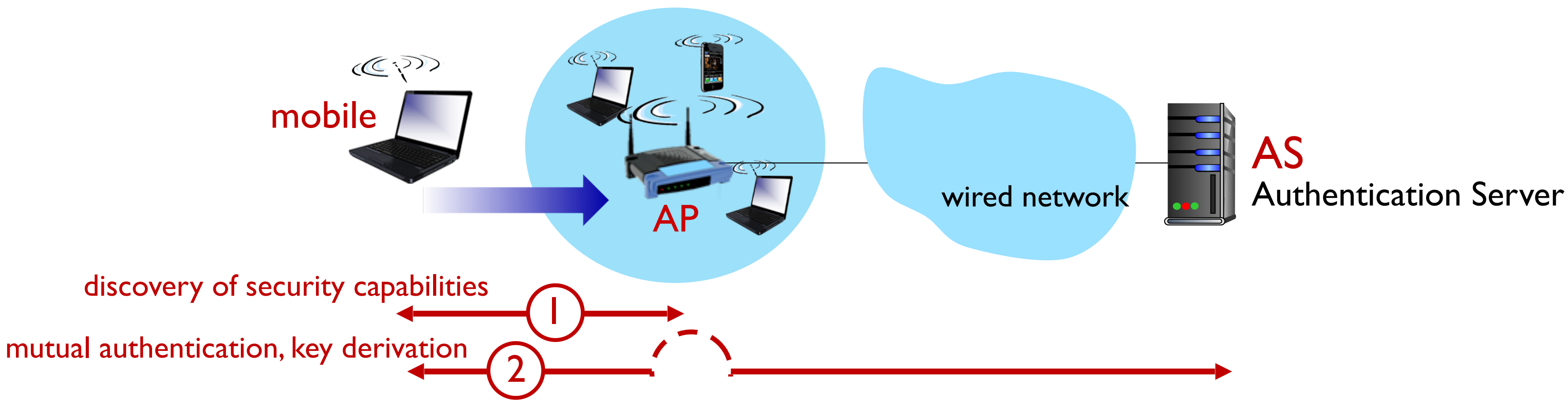
802.11i: Authentication, Encryption



1. discovery of security capabilities:
   - AP advertises its presence, forms of authentication and encryption provided
   - device requests specific forms authentication, encryption desired

although device, AP already exchanging messages, device not yet authenticated, does not have encryption keys

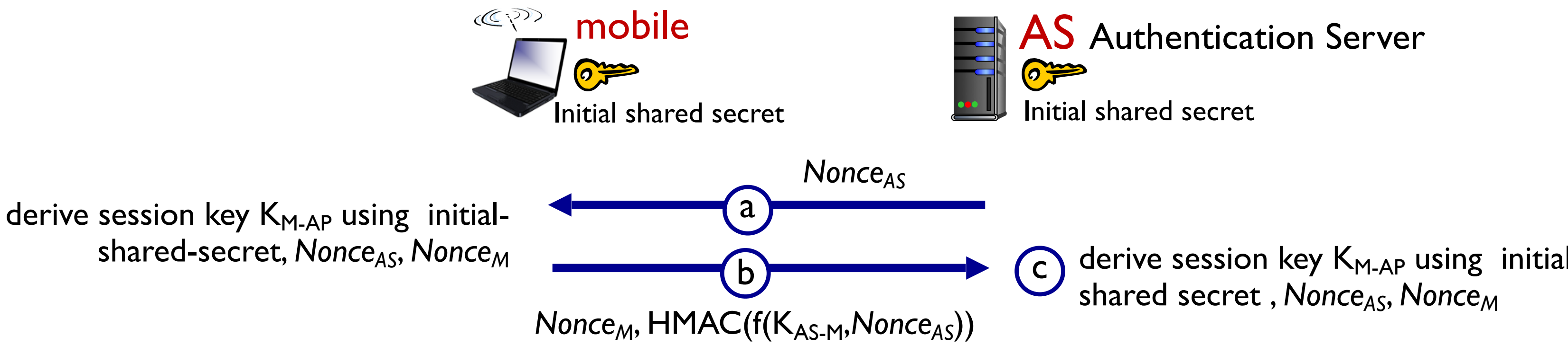
802.11: Authentication, Encryption



2 mutual authentication and shared symmetric key derivation:

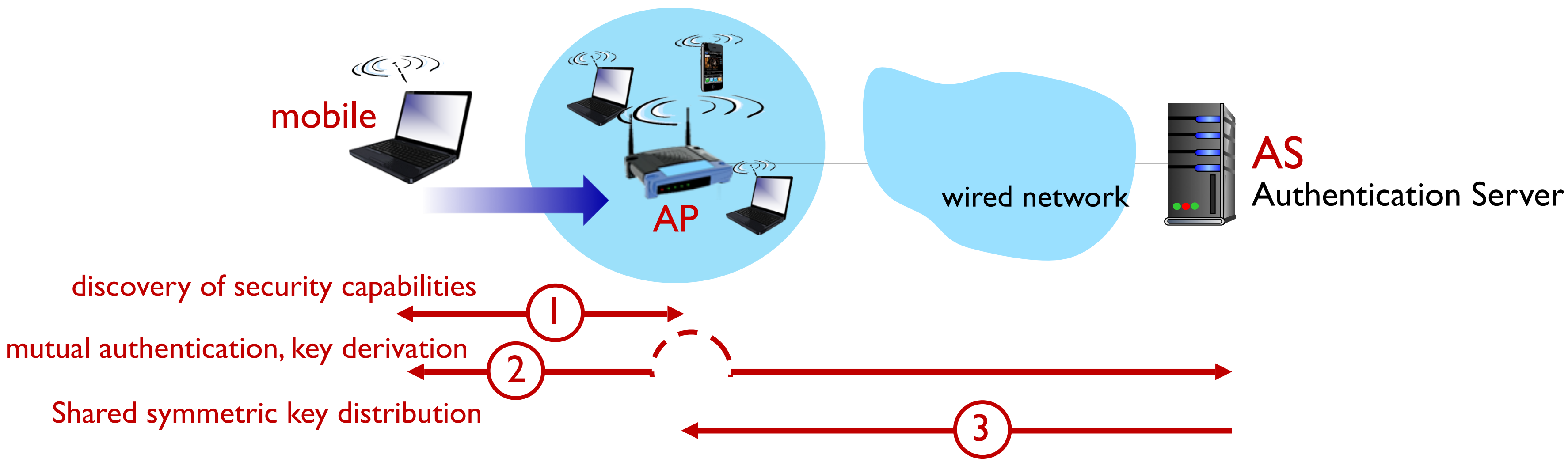
- [AS, mobile] already have shared common secret (e.g., password)
- [AS, mobile] use shared secret, nonces (prevent relay attacks), cryptographic hashing (ensure message integrity) to authenticating each other
- [AS, mobile] derive symmetric session key

802.11: WPA3 handshake



- a AS generates $Nonce_{AS}$, sends to mobile
- b mobile receives $Nonce_{AS}$
  - generates $Nonce_M$
  - generates symmetric shared session key K_{M-AP} using $Nonce_{AS}$, $Nonce_M$, *MAC addresses of mobile and AS,* and initial shared secret
  - sends $Nonce_M$, and HMAC-signed value using $Nonce_{AS}$ and initial shared secret
- c AS derives symmetric shared session key K_{M-AP}

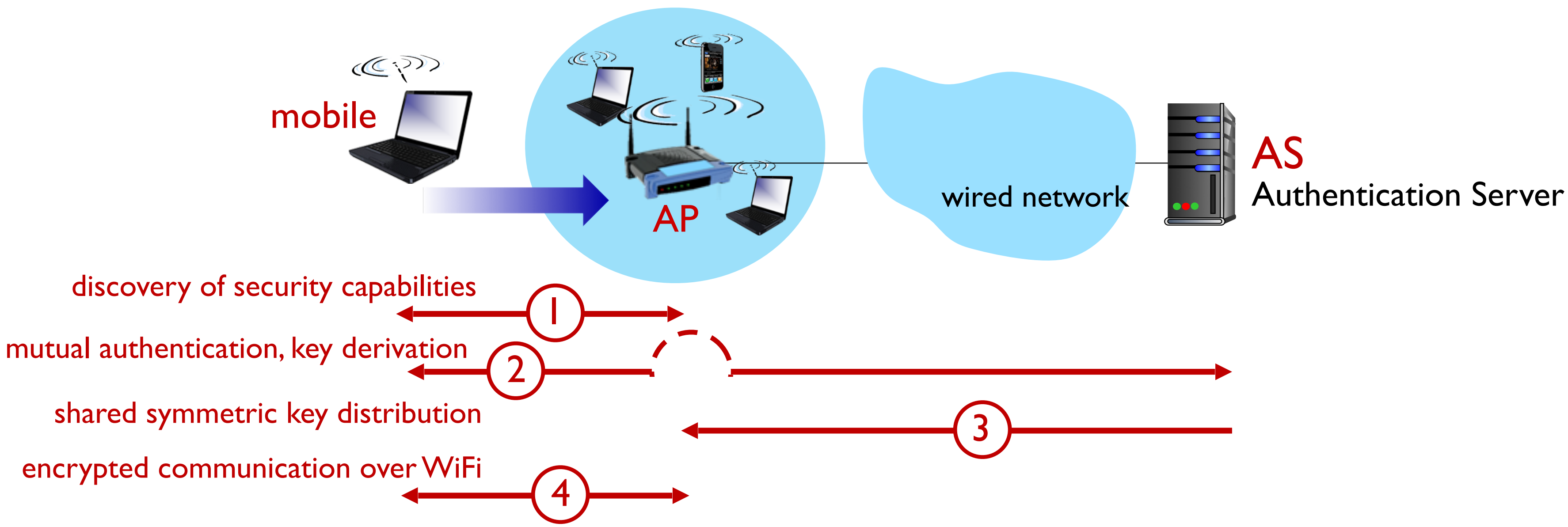
802.11: authentication, encryption



3 shared symmetric session key distribution (e.g., for AES encryption)

- same key derived at [mobile, AS]
- AS informs AP of the shared symmetric session

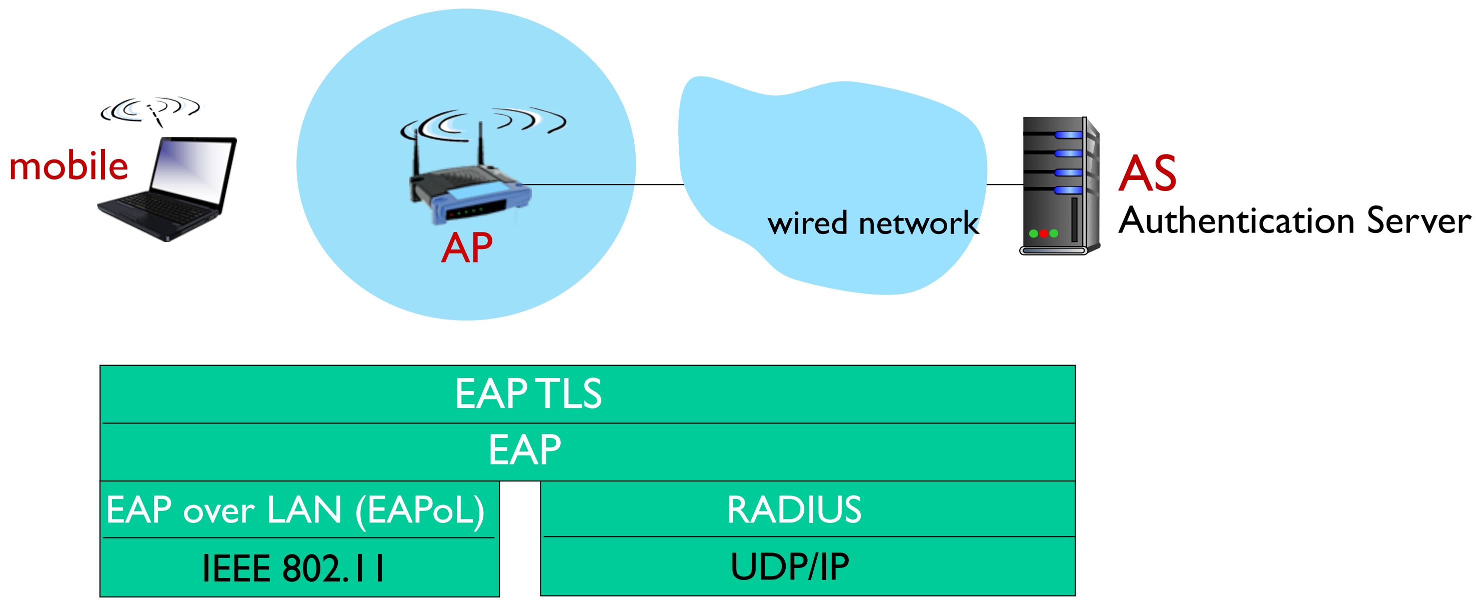
802.11: authentication, encryption



4 encrypted communication between mobile and remote host via AP

- same key derived at mobile, AS
- AS informs AP of the shared symmetric session

802.11: authentication, encryption



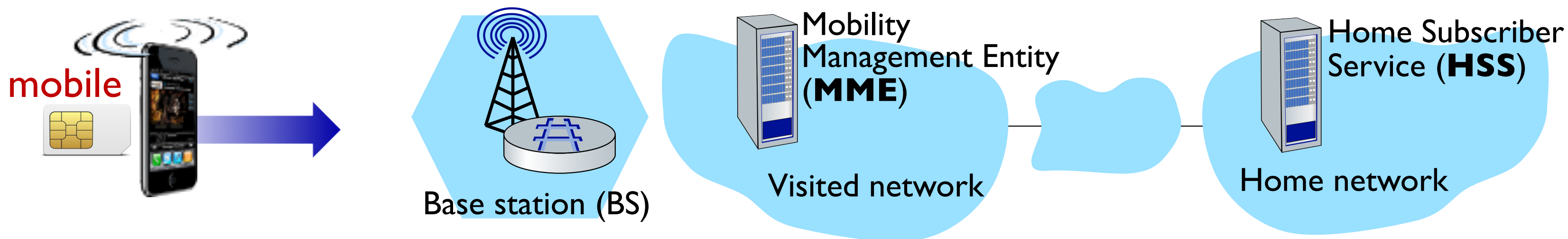
- Extensible Authentication Protocol (EAP) [RFC 3748] defines end-to-end request/response protocol between mobile device, AS

Encryptions in Different Network Layers

- Application Layer
  - Secure Email
- Transport Layer
  - TLS
- Network Layer
  - IP SEC
- Physical Layer
  - IEEE 802.11 WiFi Security
  - 4G/5G Security

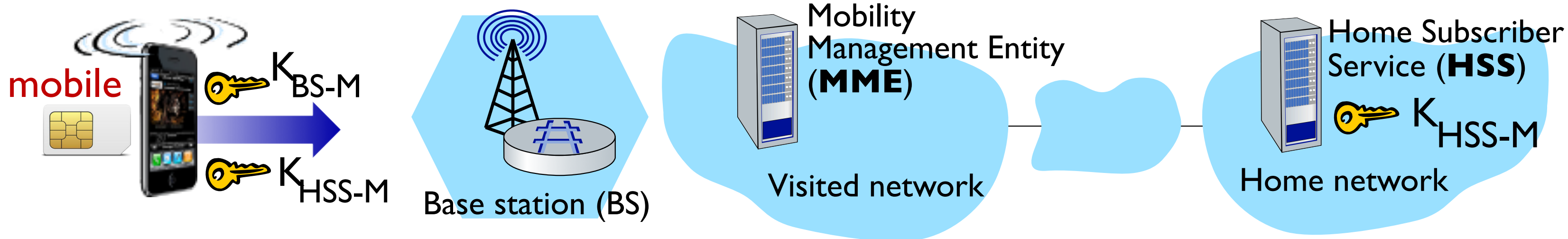
Why do we need security in different layers?

Authentication, encryption in 4G LTE



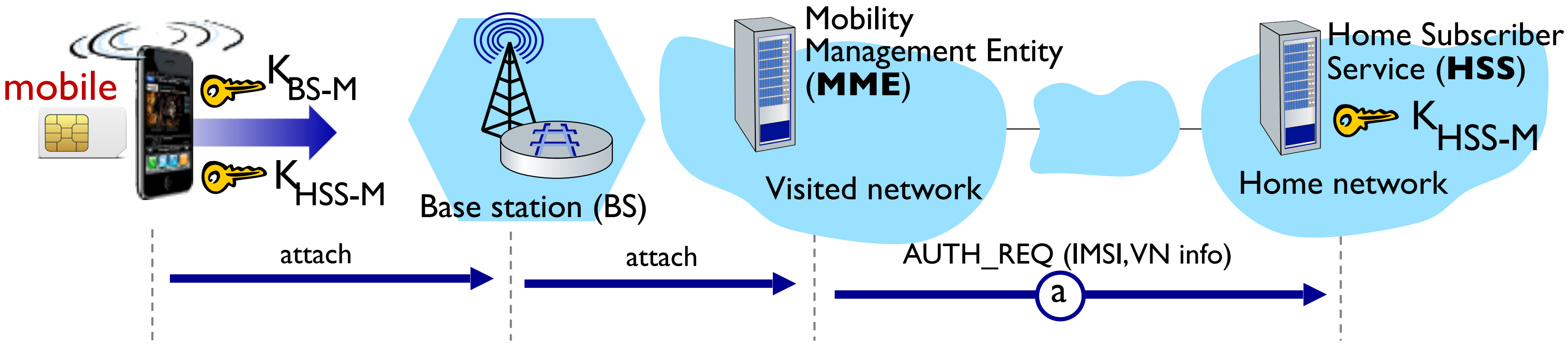
- arriving mobile must:
  - associate with BS: (establish) communication over 4G wireless link
  - authenticate itself to network, and authenticate network
- notable differences from WiFi
  - mobile's SIMcard provides global identity, contains shared keys
  - services in visited network depend on (paid) service subscription in home network

Authentication, encryption in 4G LTE



- mobile, BS use derived session key K_{BS-M} to encrypt communications over 4G link
- MME in visited network + HHS in home network, together play role of WiFi AS
  - ultimate authenticator is HSS
  - trust and business relationship between visited and home networks

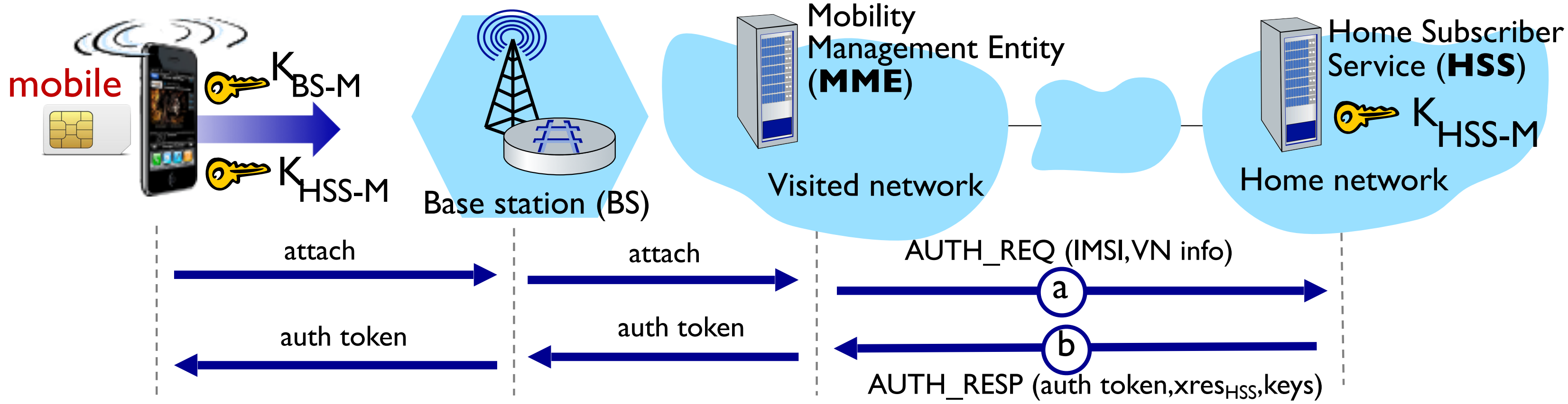
Authentication, encryption in 4G LTE



a authentication request to home network HSS

- mobile sends attach message (containing its IMSI, visited network info) relayed from BS to visited MME to home HHS
- IMSI identifies mobile's home network

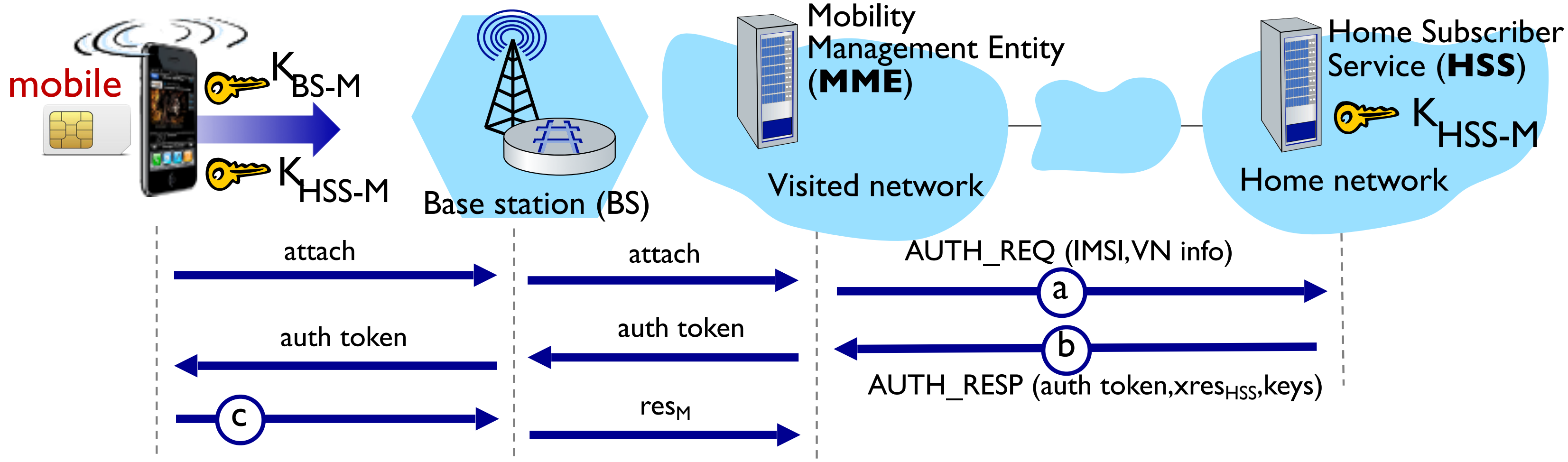
Authentication, encryption in 4G LTE



ⓑ HSS use shared-in-advance secret key, K_{HSS-M}, to derive authentication token, *auth_token*, and expected authentication response token, $xres_{HSS}$

- *auth_token* contains info encrypted by HSS using K_{HSS-M} , allowing mobile to know that whoever computed *auth_token* knows shared-in-advance secret
- mobile has authenticated network
- visited HSS keeps $xres_{HSS}$ for later use

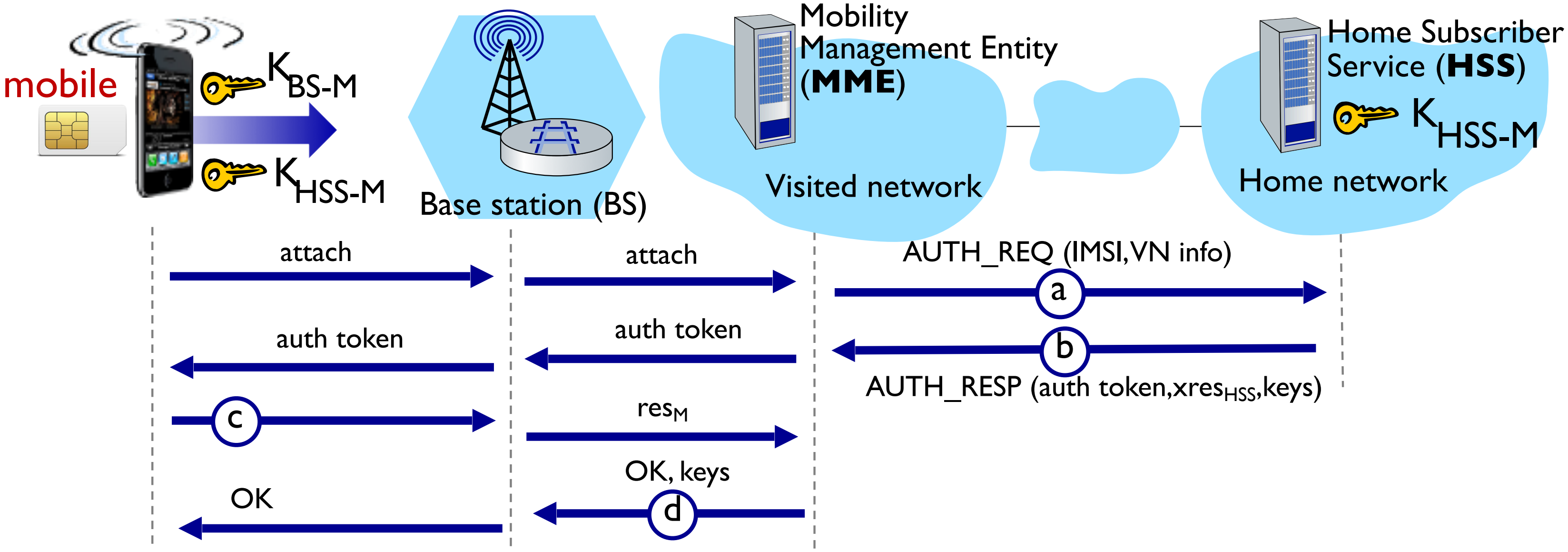
Authentication, encryption in 4G LTE



ⓒ authentication response from mobile:

- mobile computes res_M using its secret key to make same cryptographic calculation that HSS made to compute $xres_{HSS}$ and sends res_M to MME

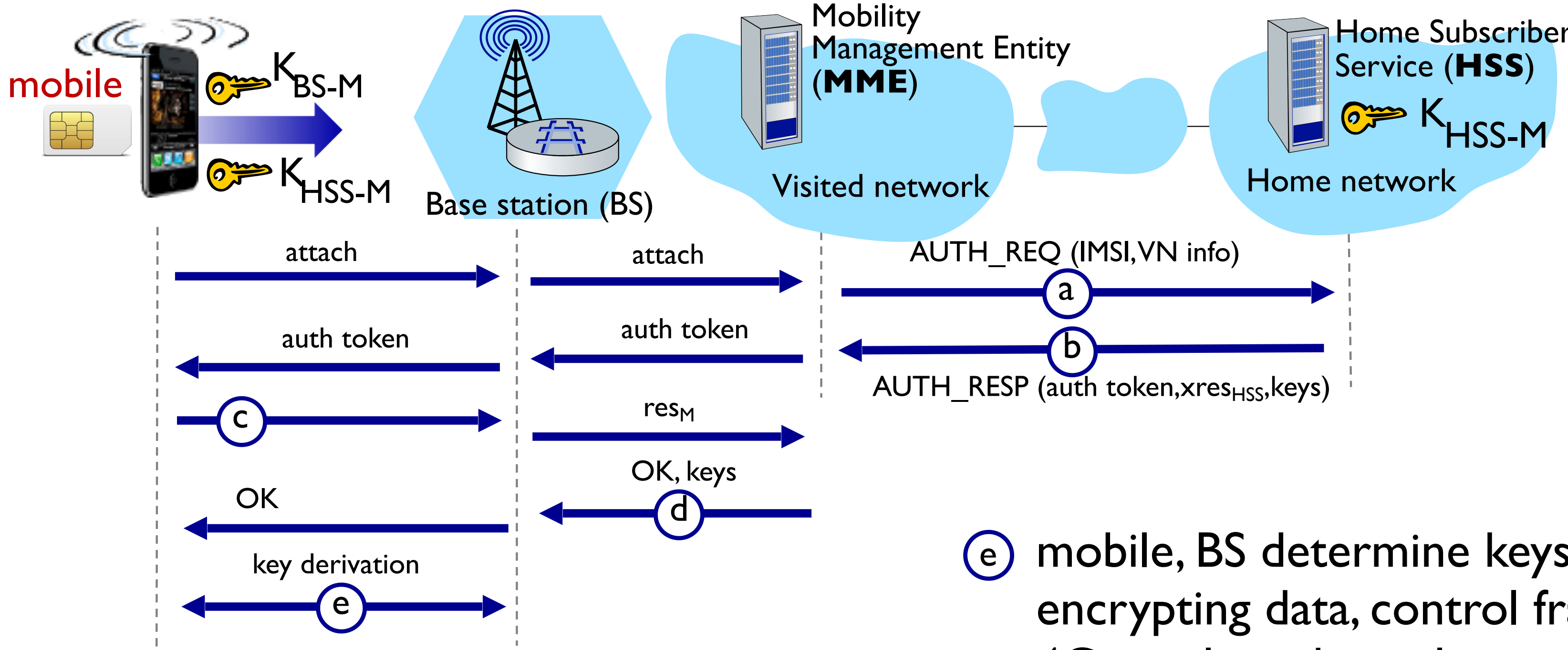
Authentication, encryption in 4G LTE



ⓓ mobile is authenticated by network:

- MME compares mobile-computed value of res_M with the HSS-computed value of $xres_{HSS}$. If they match, mobile is authenticated ! (why?)
- MME informs BS that mobile is authenticated, generates keys for BS

Authentication, encryption in 4G LTE



(e) mobile, BS determine keys for encrypting data, control frames over 4G wireless channel

- AES can be used

Authentication, encryption: from 4G to 5G

- 4G: MME in visited network makes authentication decision
- 5G: home network provides authentication decision
  - visited MME plays "middleman" role but can still reject

- 4G: uses shared-in-advance keys
- 5G: keys not shared in advance for IoT

- 4G: device IMSI transmitted in cleartext to BS
- 5G: public key crypto used to encrypt IMSI