Logistics

- Reading: Chapter on Wireless Networks
  - Homework and programming assignment
  - This evening
- Today:
  - Wireless routing / Security

IEEE 802.11 MAC Protocol: CSMA/CA

802.11 sender

1 if sense channel idle for **DIFS** then

transmit entire frame (no CD)

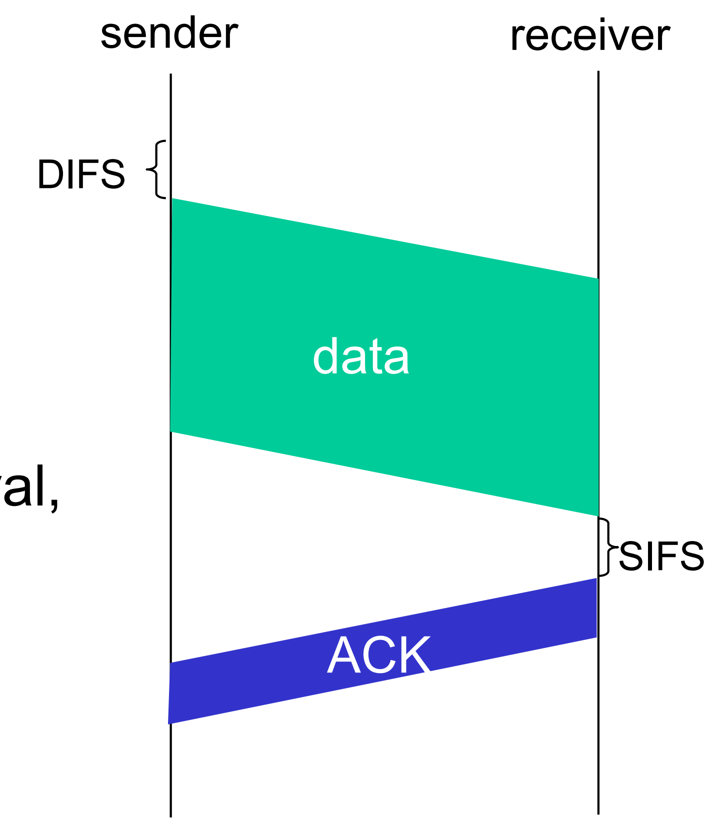
2 if sense channel busy then

start random backoff time

timer counts down while channel idle

transmit when timer expires

if no ACK, increase random backoff interval, repeat 2

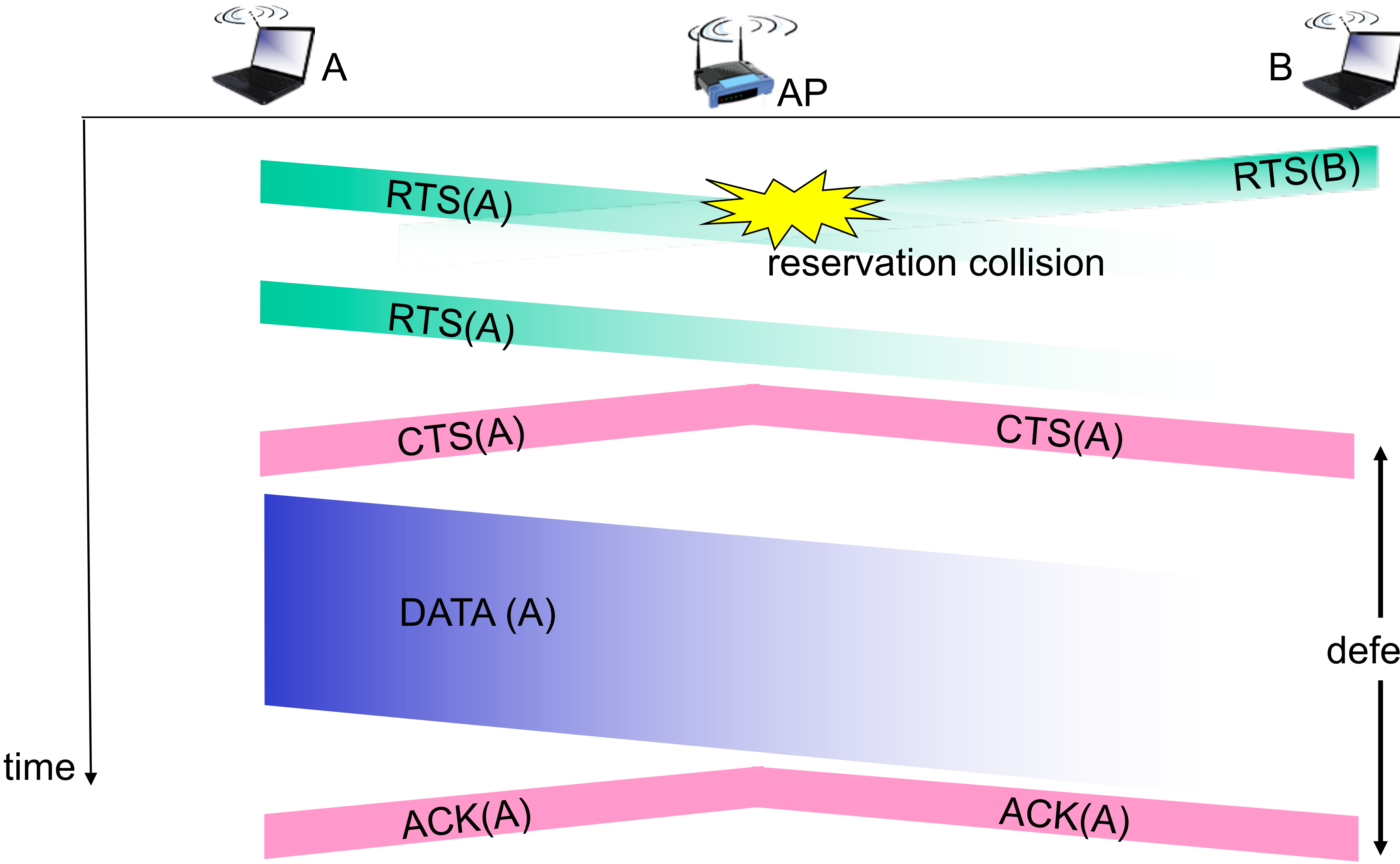


802.11 receiver

- if frame received OK

return ACK after **SIFS** (ACK needed due to hidden terminal problem)

Collision Avoidance: RTS-CTS exchange



802.11: advanced capabilities

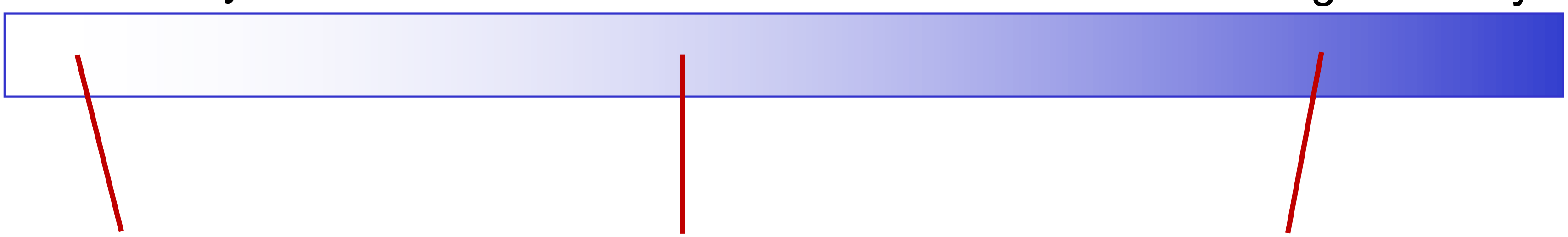
power management

- node-to-AP: "I am going to sleep until next beacon frame"
  - AP knows not to transmit frames to this node
  - node wakes up before next beacon frame
- beacon frame: contains list of mobiles with AP-to-mobile frames waiting to be sent
  - node will stay awake if AP-to-mobile frames to be sent; otherwise sleep again until next beacon frame

What is mobility?

- spectrum of mobility, from the *network* perspective:

no mobility high mobility



mobile wireless user, using same access point

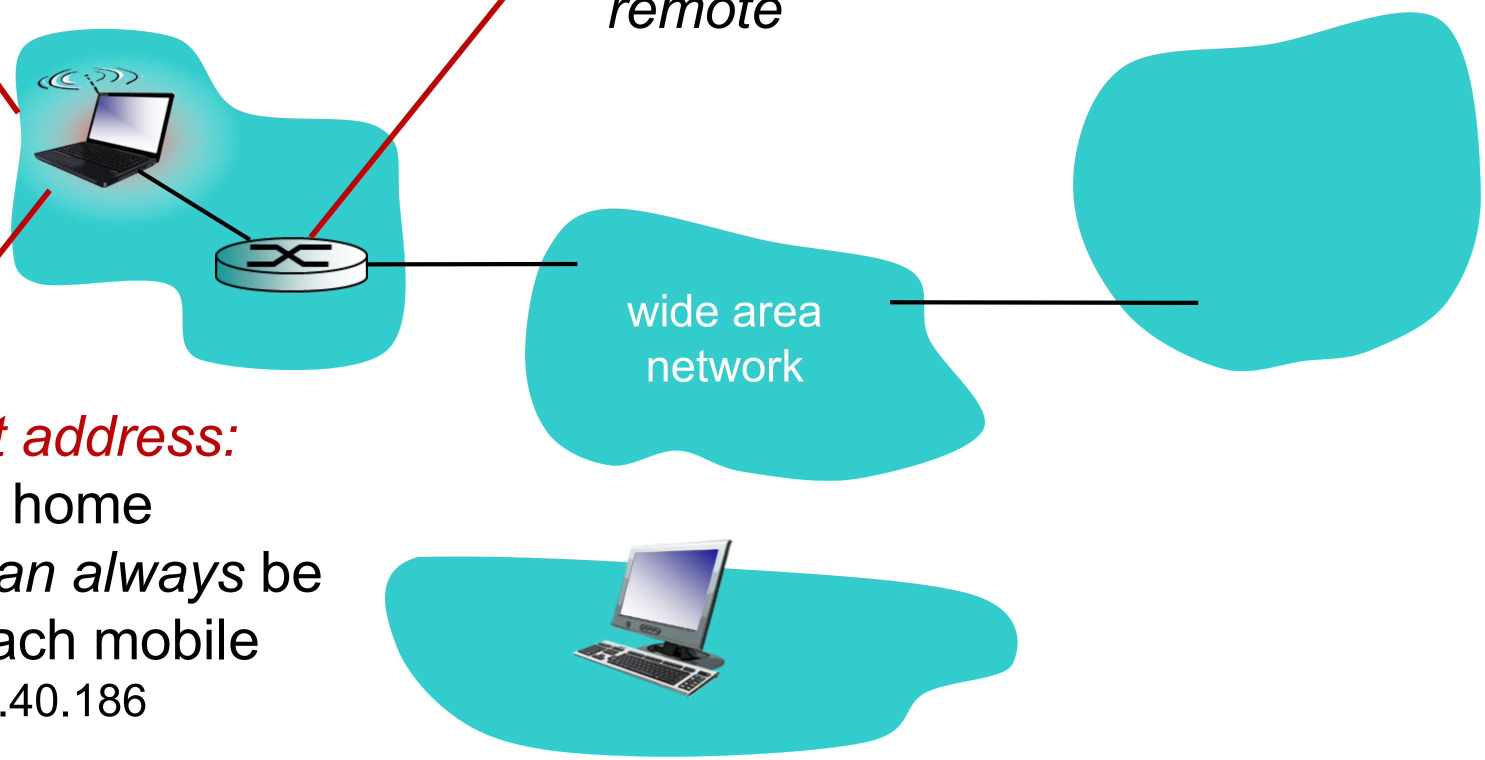
mobile user, connecting/ disconnecting from network using DHCP.

mobile user, passing through multiple access point while maintaining ongoing connections (like cell phone)

Mobility: vocabulary

home network: permanent "home" of mobile
(e.g., 128.119.40/24)

home agent: entity that will perform mobility functions on behalf of mobile, when mobile is remote



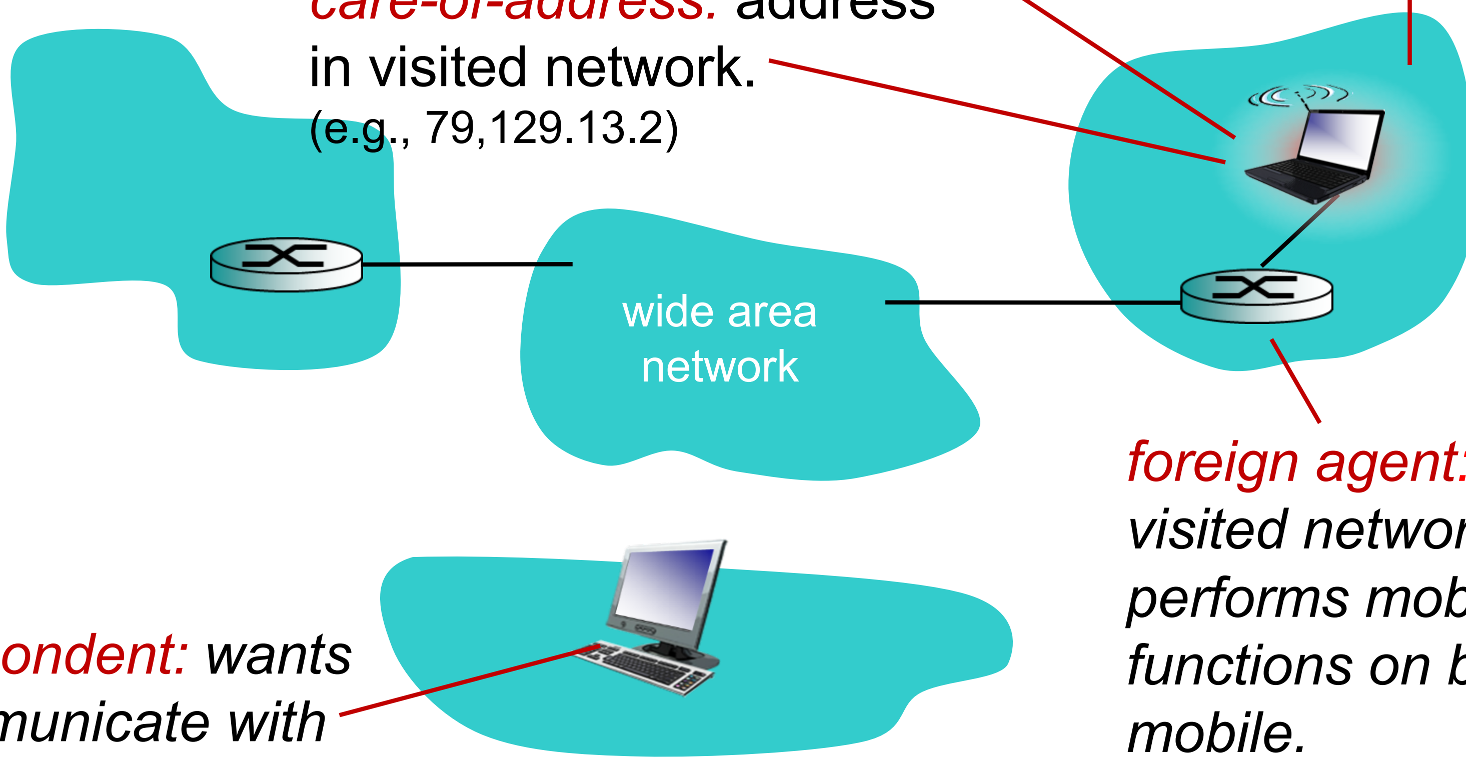
permanent address:
address in home network, *can always* be used to reach mobile
e.g., 128.119.40.186

Mobility: more vocabulary

permanent address: remains constant (e.g., 128.119.40.186)

visited network: network in which mobile currently resides (e.g., 79.129.13/24)

care-of-address: address in visited network. (e.g., 79,129.13.2)



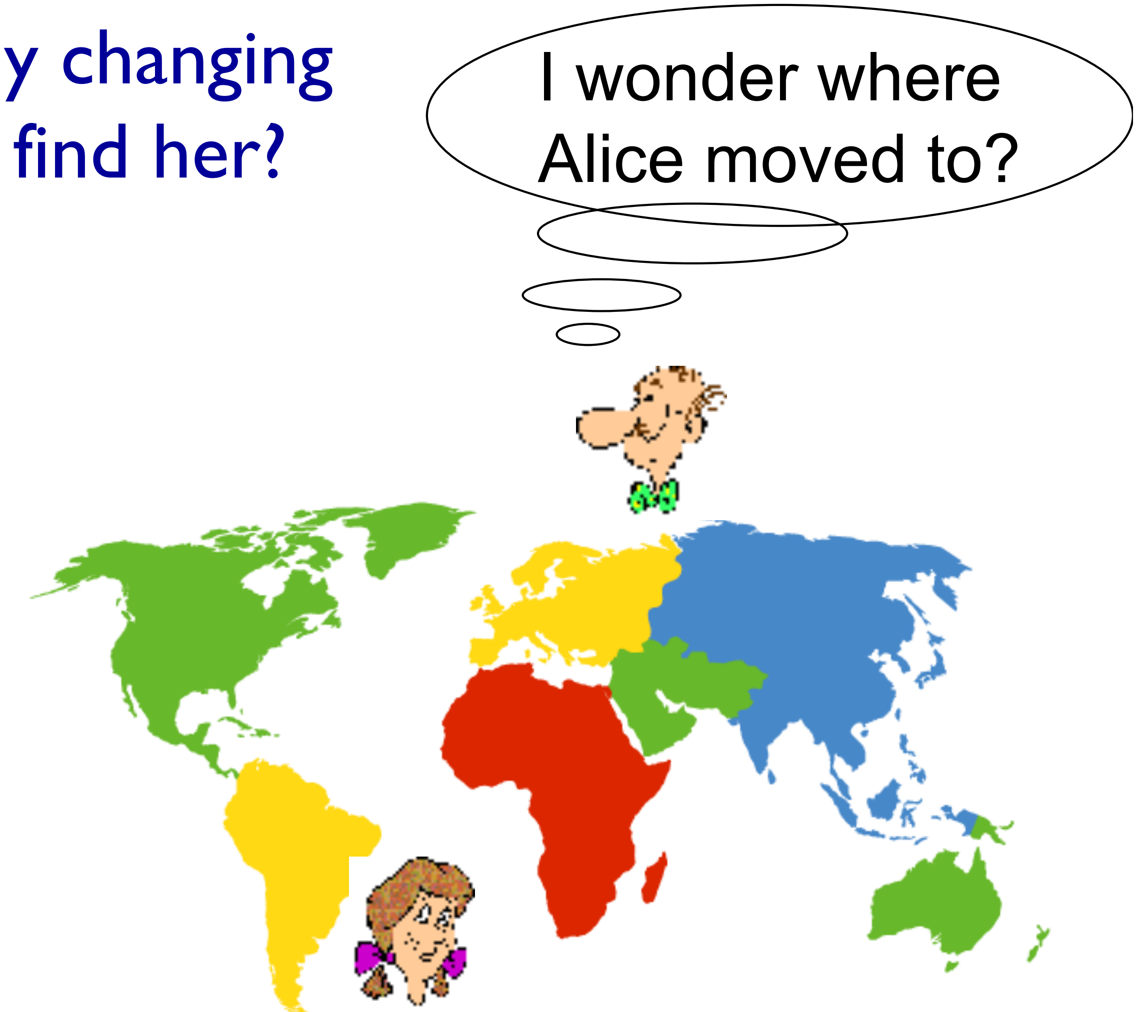
correspondent: wants to communicate with mobile

foreign agent: entity in visited network that performs mobility functions on behalf of mobile.

How do *you* contact a mobile friend:

Consider friend frequently changing addresses, how do you find her?

- search all phone books?
- call her parents?
- expect her to let you know where he/she is?



Mobility: approaches

- *let routing handle it:* routers advertise permanent address of mobile-nodes-in-residence via usual routing table exchange.
  - routing tables indicate where each mobile located
  - no changes to end-systems
- *let end-systems handle it:*
  - *indirect routing:* communication from correspondent to mobile goes through home agent, then forwarded to remote
  - *direct routing:* correspondent gets foreign address of mobile, sends directly to mobile

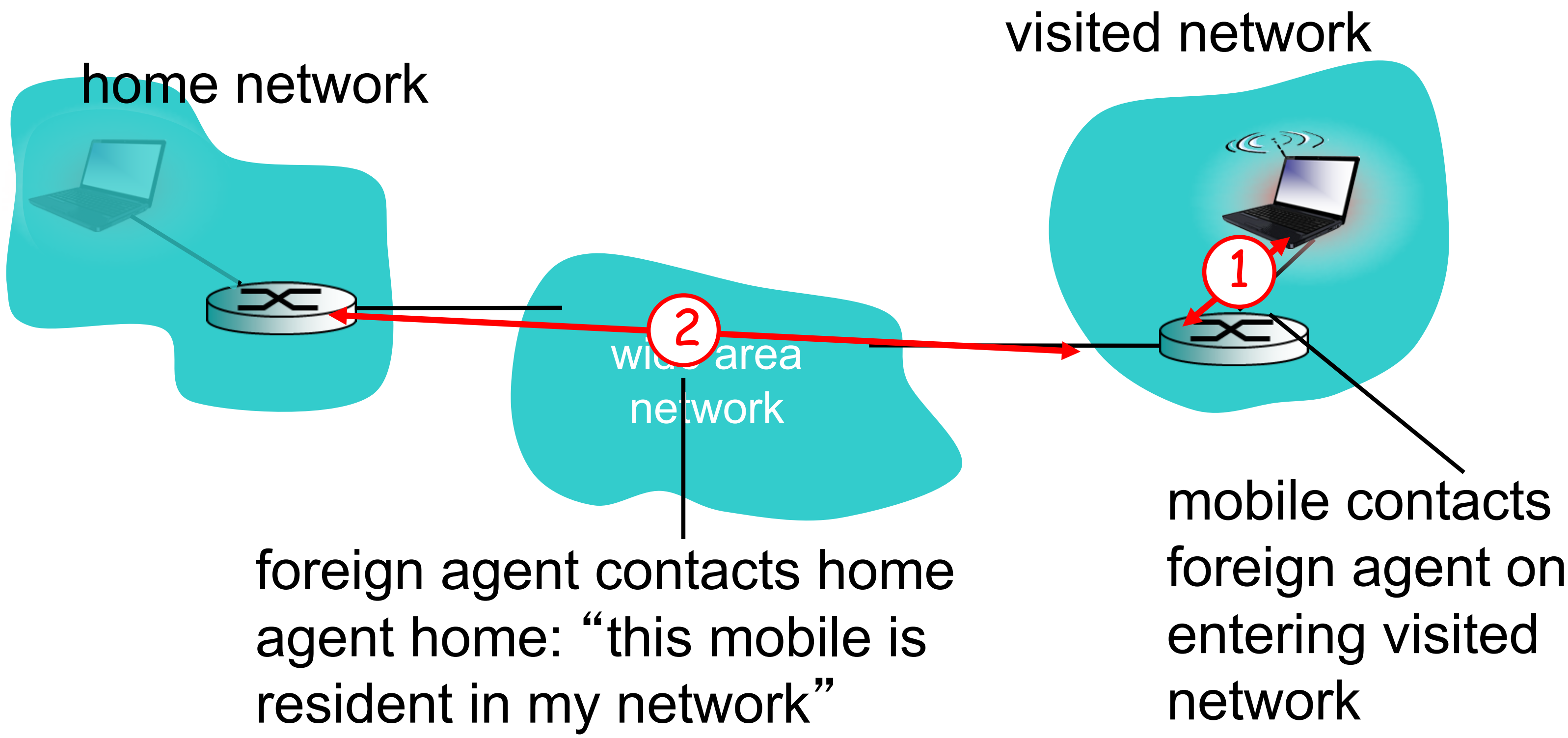
Mobility: approaches

- *let routing handle it:* ro… dvertise permanent address of mobile-nodes-in-re… sual routing table exchange.
  - routing tables … each mobile located
  - no changes to …

not scalable to millions of mobiles

- *let end-systems handle it:*
  - *indirect routing:* communication from correspondent to mobile goes through home agent, then forwarded to remote
  - *direct routing:* correspondent gets foreign address of mobile, sends directly to mobile

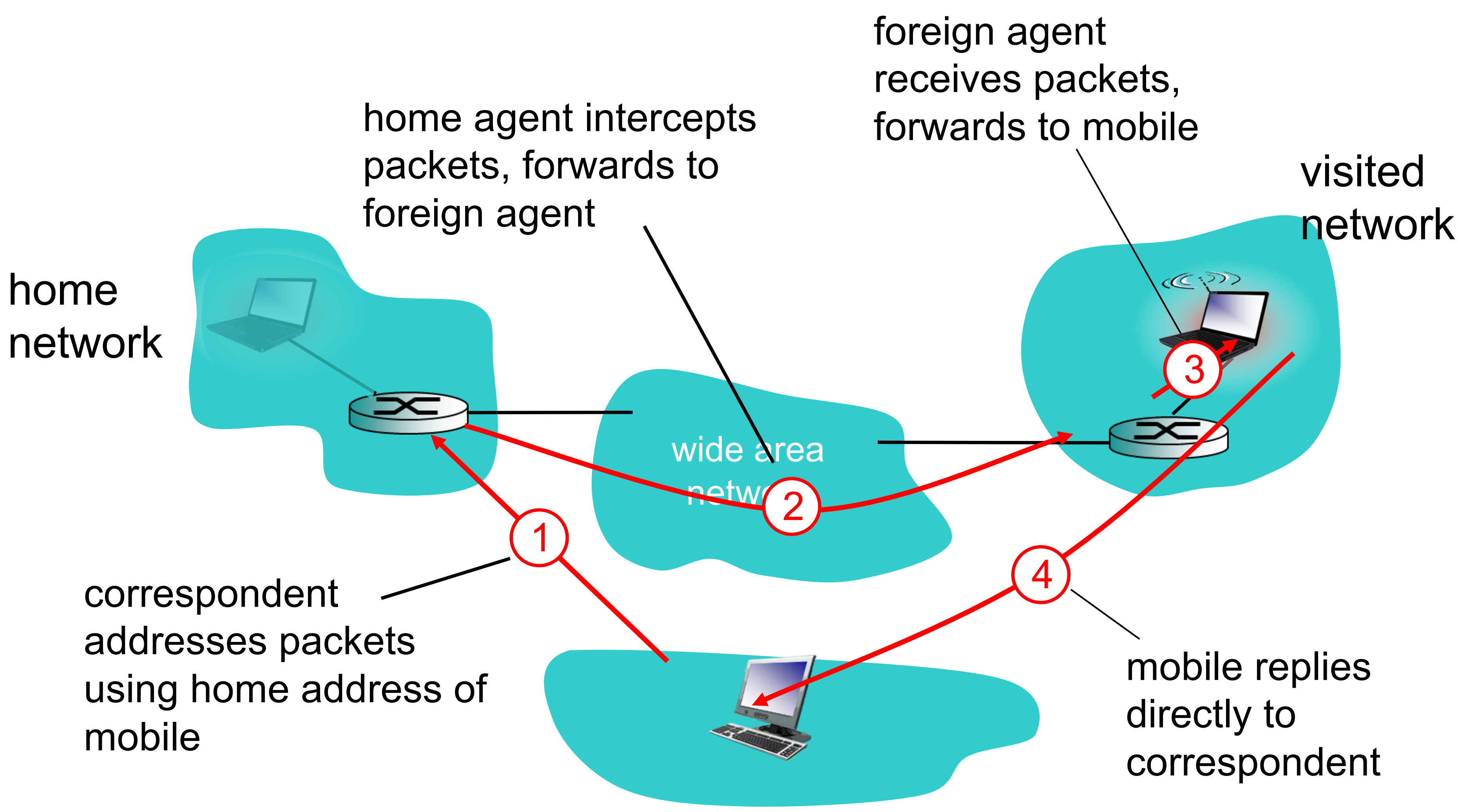
Mobility: registration



end result:

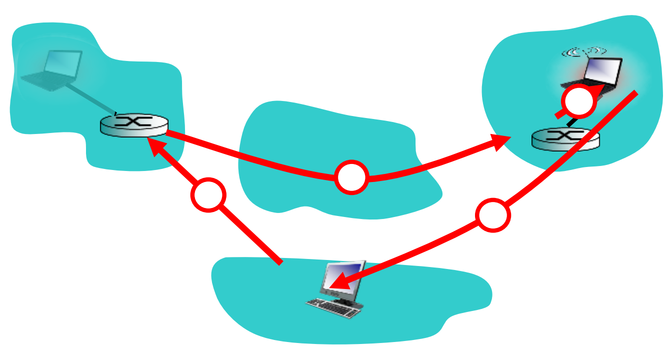
- foreign agent knows about mobile
- home agent knows location of mobile

Mobility via indirect routing



Indirect Routing: comments

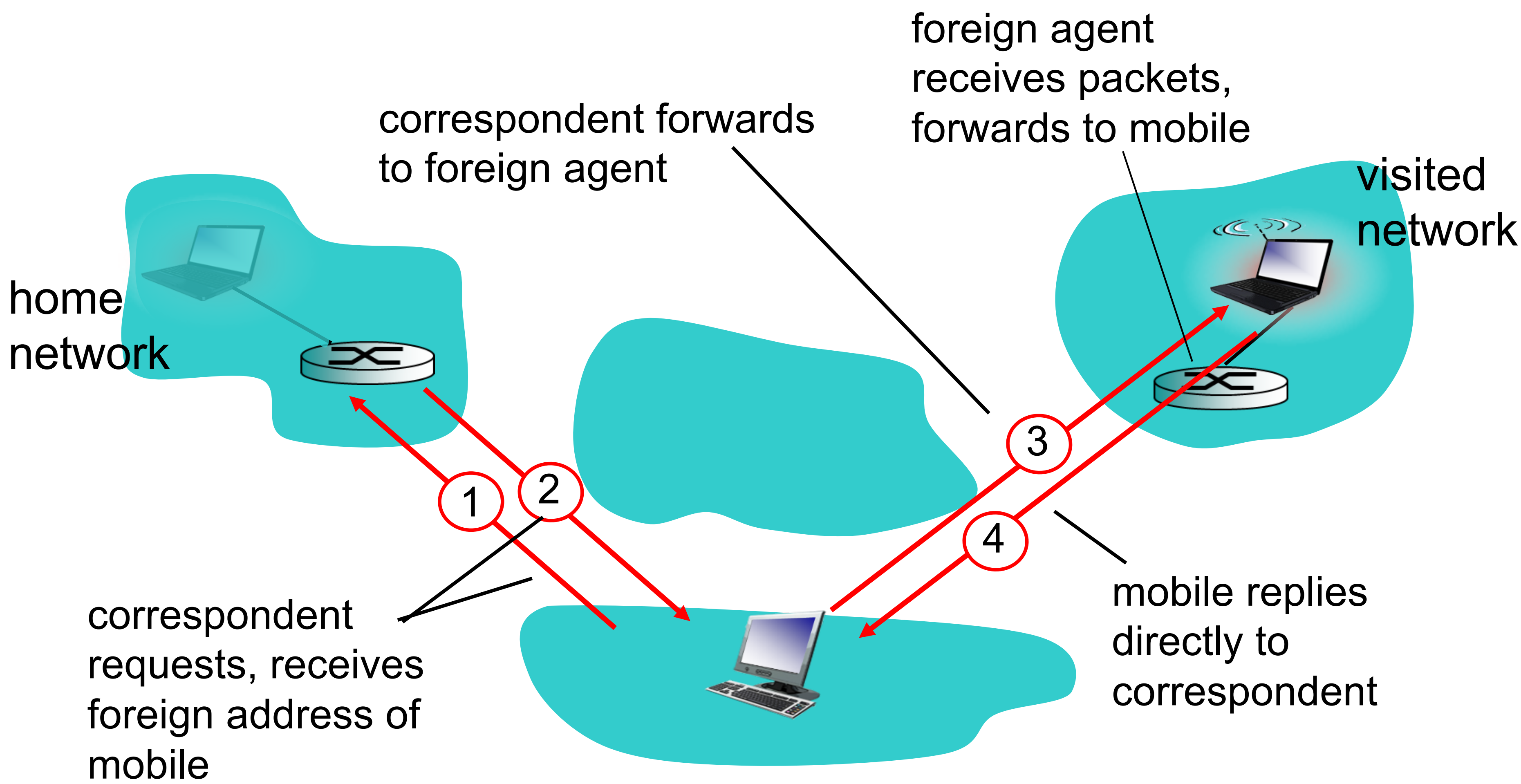
- mobile uses two addresses:
  - permanent address: used by correspondent (hence mobile location is *transparent* to correspondent)
  - care-of-address: used by home agent to forward datagrams to mobile
- foreign agent functions may be done by mobile itself
- triangle routing: correspondent-home-network-mobile
  - inefficient when correspondent, mobile are in same network



Indirect routing: moving between networks

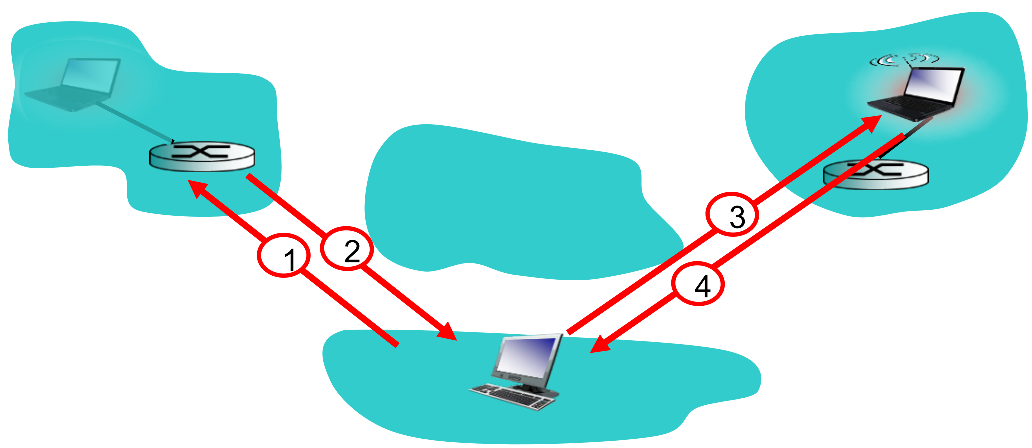
- suppose mobile user moves to another network
  - registers with new foreign agent
  - new foreign agent registers with home agent
  - home agent update care-of-address for mobile
  - packets continue to be forwarded to mobile (but with new care-of-address)
- mobility, changing foreign networks transparent: *on going connections can be maintained!*

Mobility via direct routing



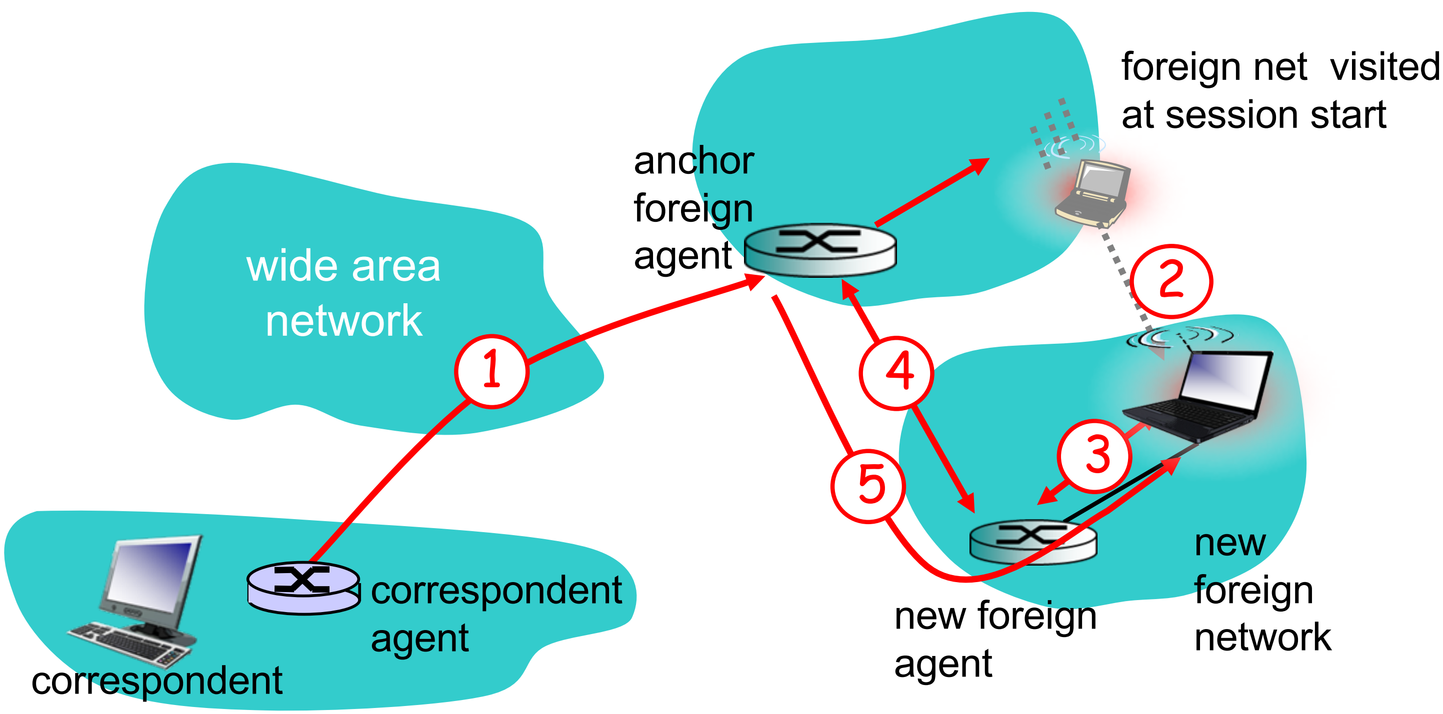
Mobility via direct routing: comments

- overcome triangle routing problem
- *non-transparent to correspondent:* correspondent must get care-of-address from home agent
  - what if mobile changes visited network?



Accommodating mobility with direct routing

- anchor foreign agent: FA in first visited network
- data always routed first to anchor FA
- when mobile moves: new FA arranges to have data forwarded from old FA (chaining)



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

RC4 Key scheduling

S[]: A permutation of all 256 byte values
key[]: Array containing a secret key (length "keylength")

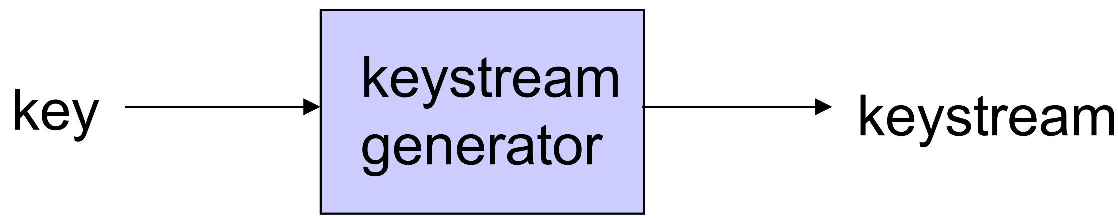
```
for i from 0 to 255
    S[i] := i
endfor
j := 0
for i from 0 to 255
    j := (j + S[i] + key[i mod keylength]) mod 256
    swap values of S[i] and S[j]
endfor
```

RC4 Generation

S[]: A permutation of all 256 byte values
key[]: Array containing a secret key (length "keylength")

```
i := 0
j := 0
while GeneratingOutput:
    i := (i + 1) mod 256
    j := (j + S[i]) mod 256
    swap values of S[i] and S[j]
    K := S[(S[i] + S[j]) mod 256]
    output K
endwhile
```

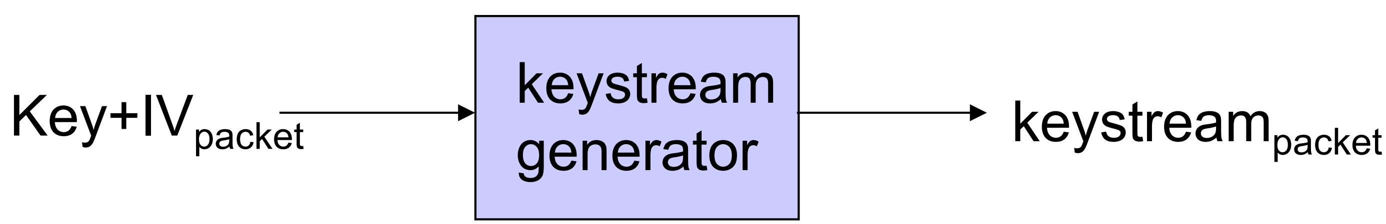
Symmetric stream ciphers



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

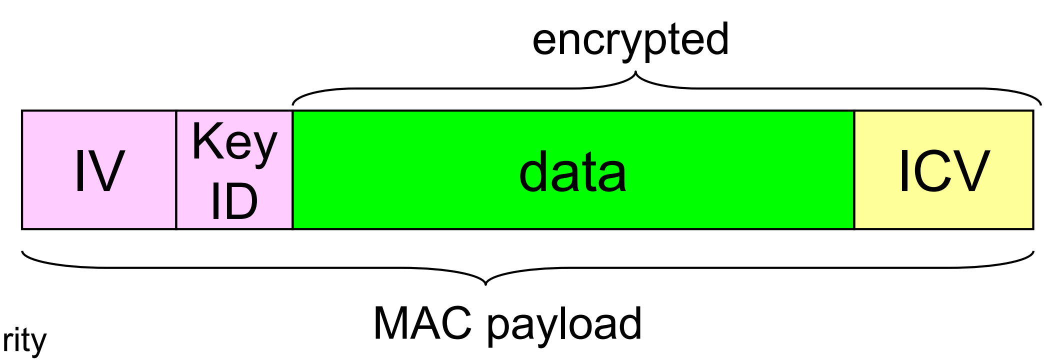
Stream cipher and packet independence

- 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:

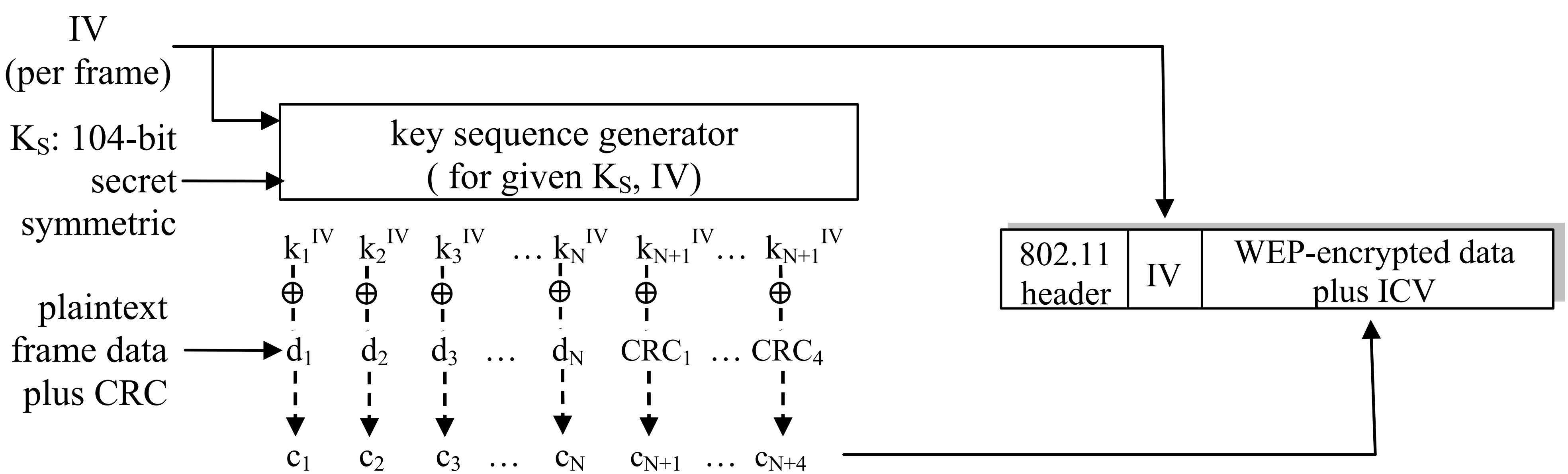


WEP encryption (1)

- sender calculates Integrity Check Value (ICV) over data
  - four-byte hash/CRC for data integrity
- each side has 104-bit shared key
- sender creates 24-bit initialization vector (IV), appends to key: gives 128-bit key
- sender also appends keyID (in 8-bit field)
- 128-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

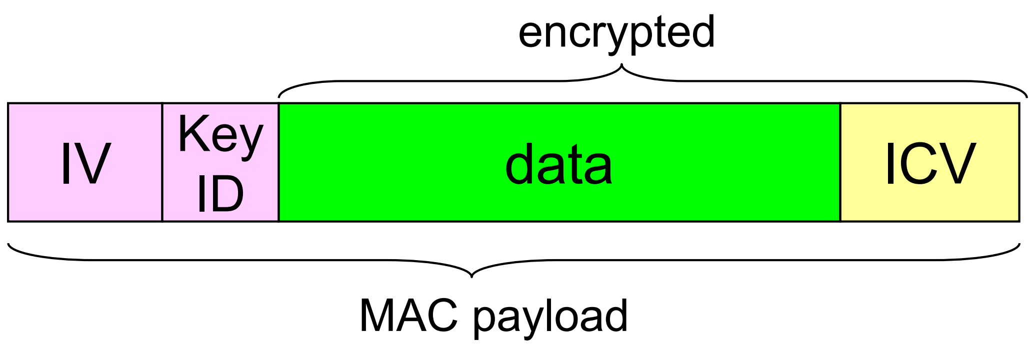


WEP encryption (2)



new IV for each frame

WEP decryption overview

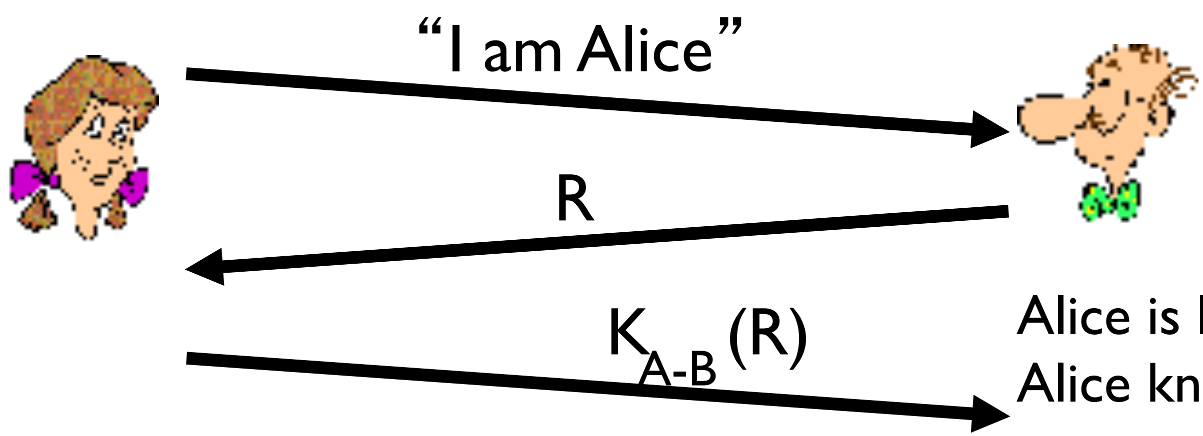


- receiver extracts IV
- inputs IV, shared secret key into pseudo random generator, gets keystream
- XORs keystream with encrypted data to decrypt data + ICV
- verifies integrity of data with ICV
  - note: message integrity approach used here is different from MAC (message authentication code) and signatures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

End-point authentication w/ nonce

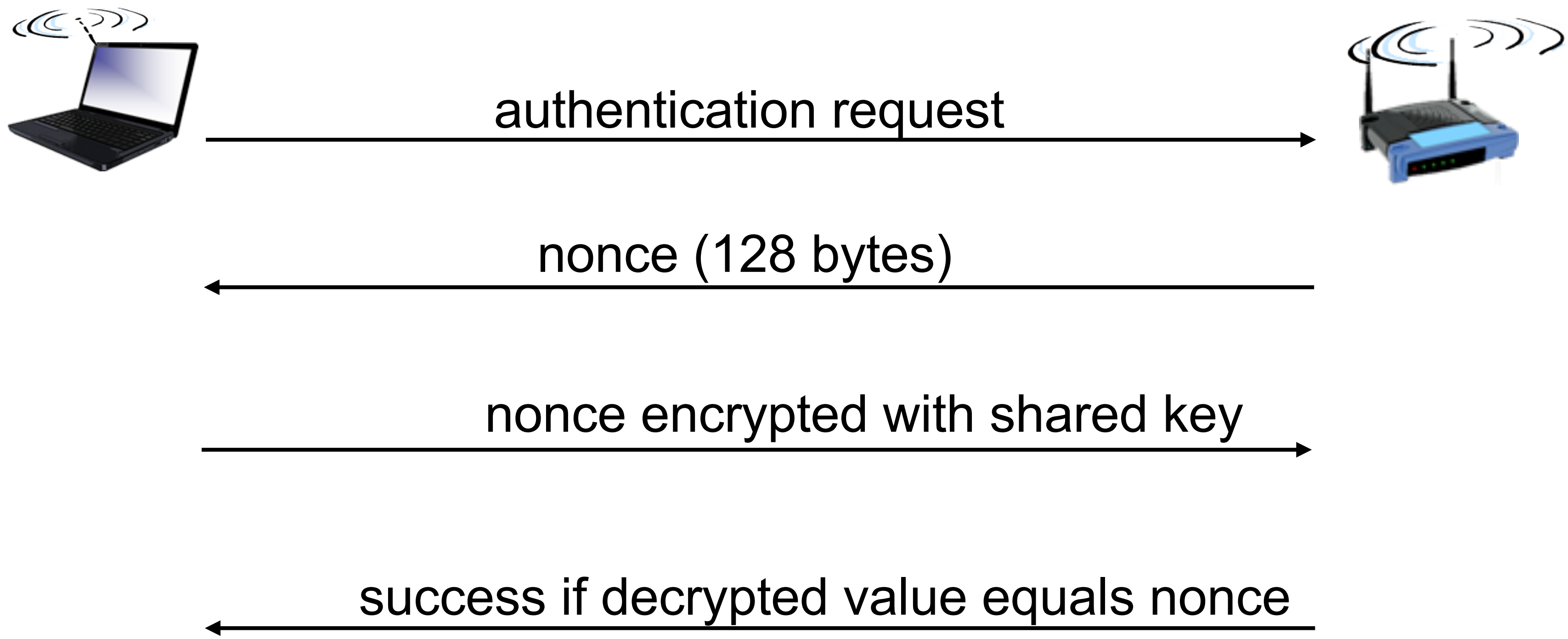
Nonce: number (R) used only *once –in-a-lifetime*

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

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

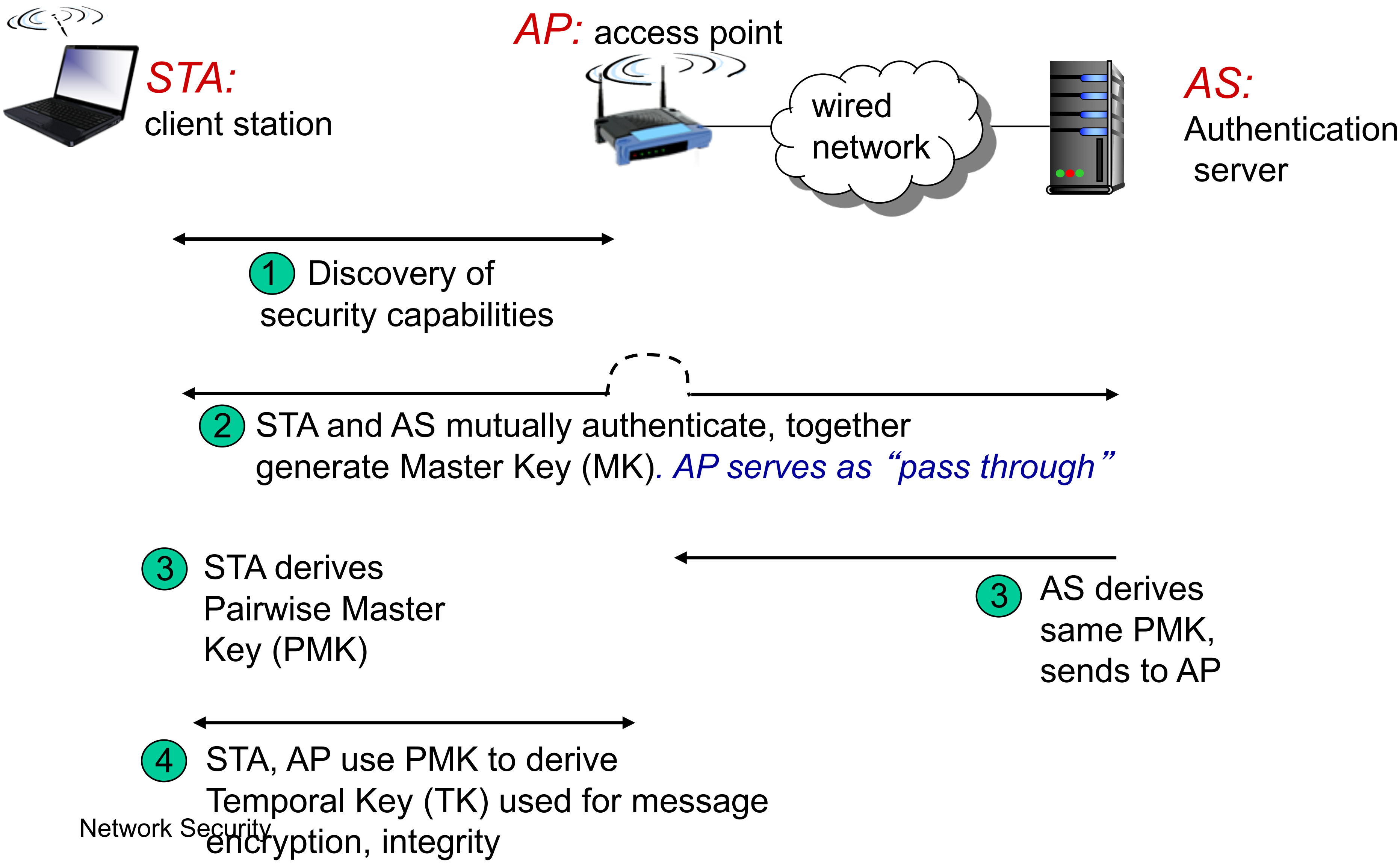
attack:

  - 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 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

802.11i: four phases of operation



EAP: extensible authentication protocol

- EAP: end-end client (mobile) to authentication server protocol
- EAP sent over separate “links”
  - mobile-to-AP (EAP over LAN)
  - AP to authentication server (RADIUS over UDP)