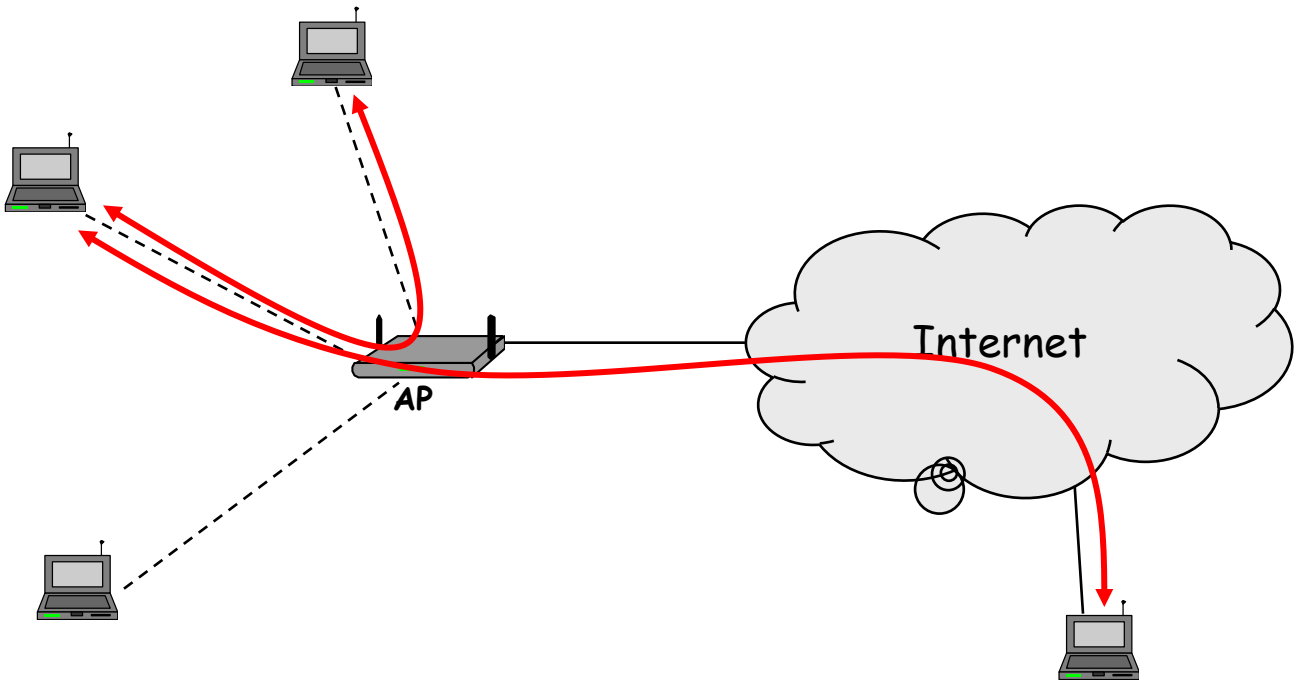
Securing 802.11 (WiFi) networks

Srdjan Čapkun

Department of Computer Science
ETH Zurich

Introduction to WiFi



Why security is 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.)

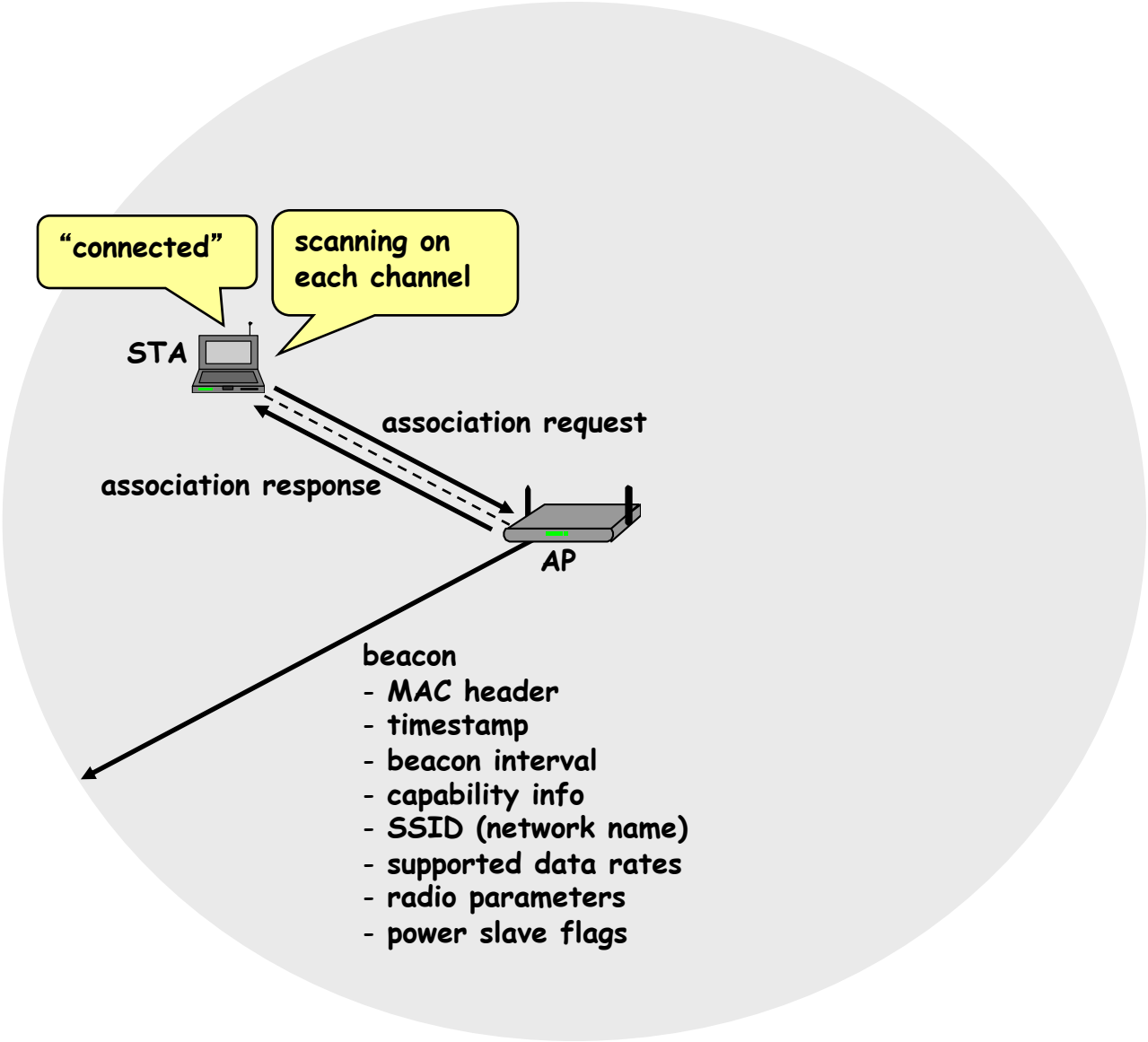
- broadcast communications
  - wireless usually means radio, which (generally) has a broadcast nature
  - transmissions can be overheard by anyone in range
  - anyone can generate transmissions,
    - which will be received by other devices in range
    - which will interfere with other nearby transmissions 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

Introduction to WiFi



Access mechanisms

open network (no protection)

- assumption: there are no unauthorized users in the range of the network
- problems: range is hard to determine (unpredictable propagation of the signals, directional antennas, ...)

closed network

- using SSIDs for authentication (Service Set Identifier)
- MAC filtering
- shared keys
- authentication servers

*** Password-based authentication**

MAC filtering

- MAC address filtering
  - only devices with certain MAC addresses are allowed to associate
  - **needs pre-registration of all device at the AP**

- MAC can be sniffed and forged
  - **sent in clear text in each packet (can be sniffed)**
  - can be forged

Device identification – MAC addresses

- "Hardcoded" addresses in WiFi cards ("unique device identifiers")
  - *all* devices have different addresses

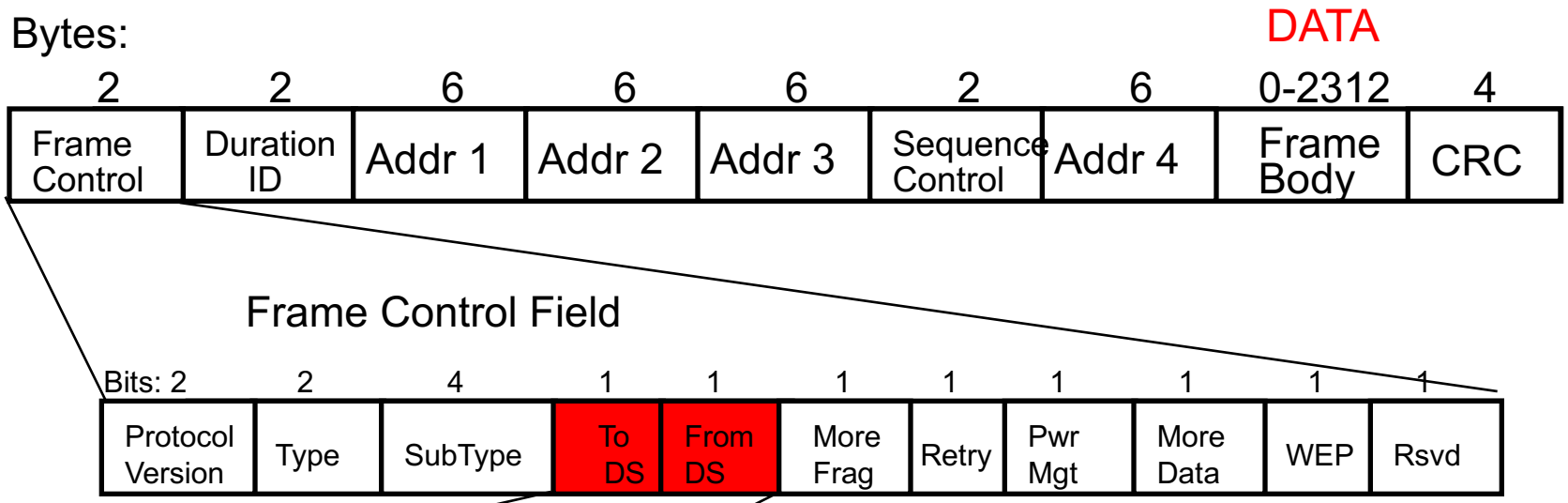
- Concept taken over from ethernet addresses

```
>ipconfig /all
```

```
Ethernet adapter Wireless Network Connection:

        Media State . . . . . . . . . . . : Media disconnected
        Description . . . . . . . . . . . : Intel(R) PRO/Wirele
k Connection
        Physical Address. . . . . . . . . : 00-13-02-B8-9A-1B
```

802.11 MAC header



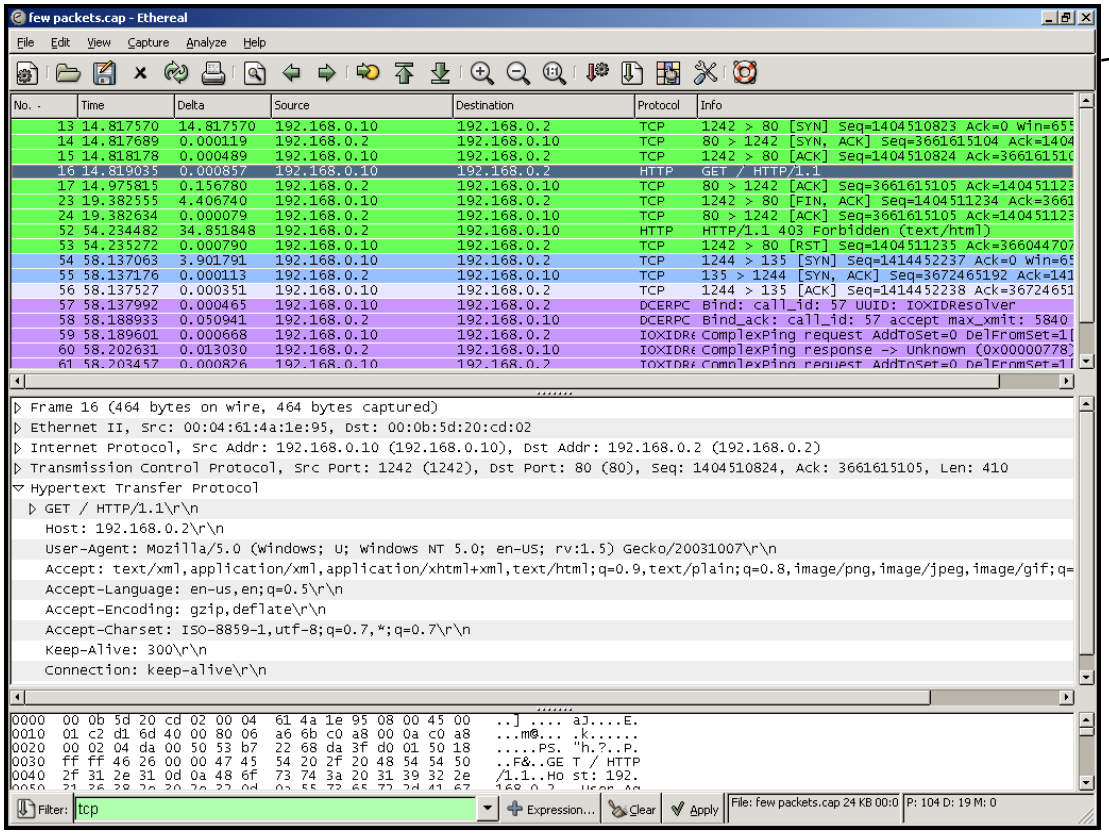
To DS	From DS	Address 1	Address 2	Address 3	Address 4
0	0	DA	SA	BSSID	N/A
0	1	DA	BSSID	SA	N/A
1	0	BSSID	SA	DA	N/A
1	1	RA	TA	DA	SA

- Addr. 1 = All stations filter on this address.
- Addr. 2 = Transmitter Address (TA).
- Addr. 3 = Dependent on *To* and *From DS* bits.

3 simple steps for overcoming MAC filtering

1. Put your card in promiscuous mode (accepts all packets).

2. Sniff the traffic and find out which MAC addresses are accepted by the AP



Ethereal

3. Change your MAC address (need a card that can do that)

```
# ifconfig ath0 hw ether <mac address of C>
```

SSID-based access control

- SSID = Service Set IDentifier (network name)
- a 32-character unique identifier
- attached to the header of packets
- acts as a password when a mobile device tries to connect to the WLAN
- SSID differentiates one WLAN from another
- all devices attempting to connect to a specific WLAN must use the same SSID

SSID-based access control

- SSIDs can be sniffed *(e.g. http://www.ethereal.com)*
  - advertised by the APs
  - contained in SSID response frames

- Overcomming SSID-based access control
  - Sniff SSID (either sent by the clients or advertised by the AP)
  - Set your SSID to the same value ...

- MAC/SSID access control: not a bad protection from unskilled neighbors (much better than no authentication/protection)

MAC/SSID sniffing and MAC forging will be shown in the exercises.

Protected access using WEP

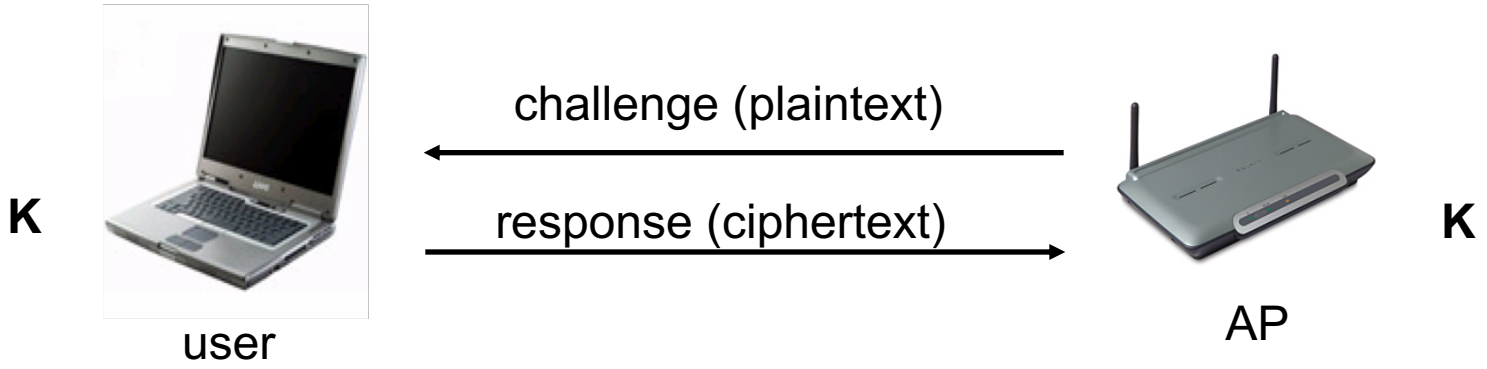
- WEP = Wired Equivalent Privacy
  - part of the IEEE 802.11 specification

- goal
  - make the WiFi network *at least as secure as a wired LAN* (that has no particular protection mechanisms)
  - WEP has never intended to achieve strong security
  - (at the end, it hasn't achieved even weak security)

- services
  - access control to the network
  - message confidentiality
  - message integrity

WEP-authentication (1)

- Based on a shared key between the station and the AP (40 bit or 104 bit)
- Based on the RC4 symmetric stream cipher
- 24-bit Initialization Vector (IV)



- The payload of every packet is encrypted (confidentiality) with its CRC value (integrity)
- Authentication through 'classical' challenge-response authentication protocol ... using the shared key ...

WEP-authentication (2)

user

AP

probe request
(sent on all channels, trying to find AP with matching SSID)

response (if in range)

auth. request (choosing best AP)

challenge

response

auth. decision

.... if authentication successful

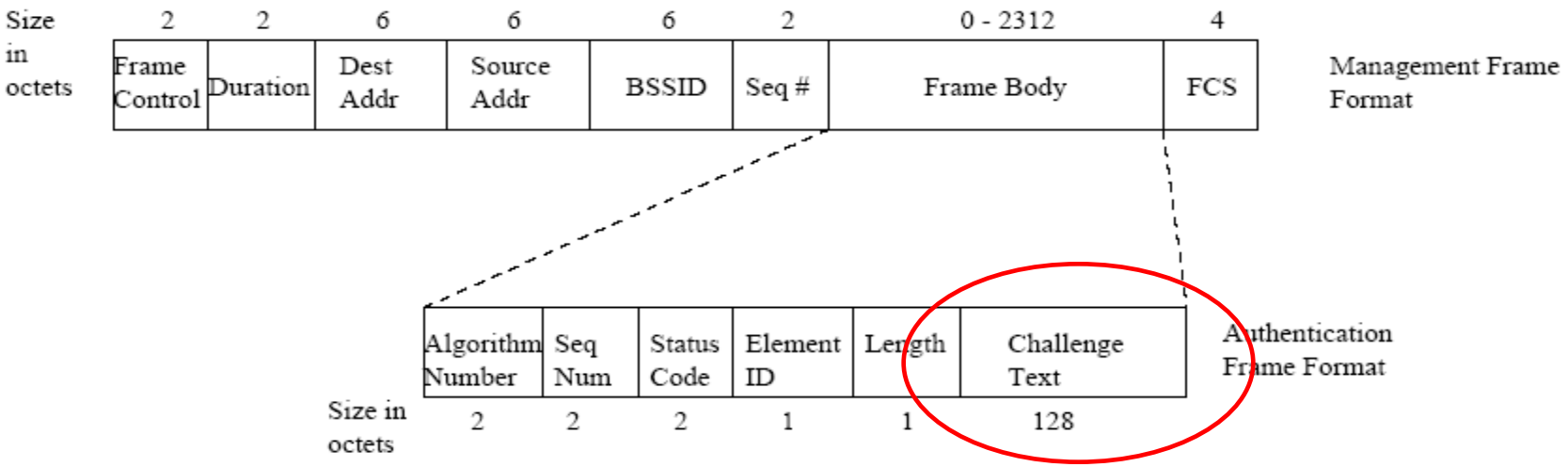
association request

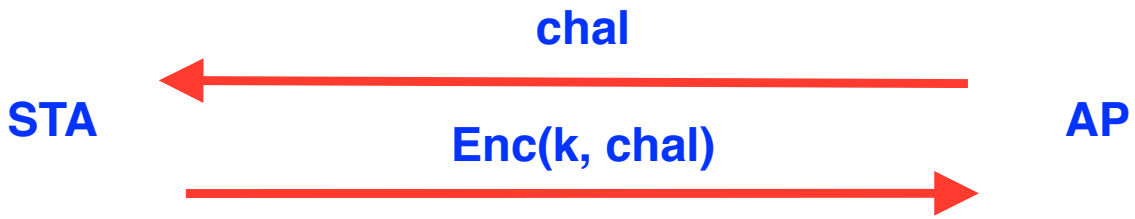
association reply

user can now send data

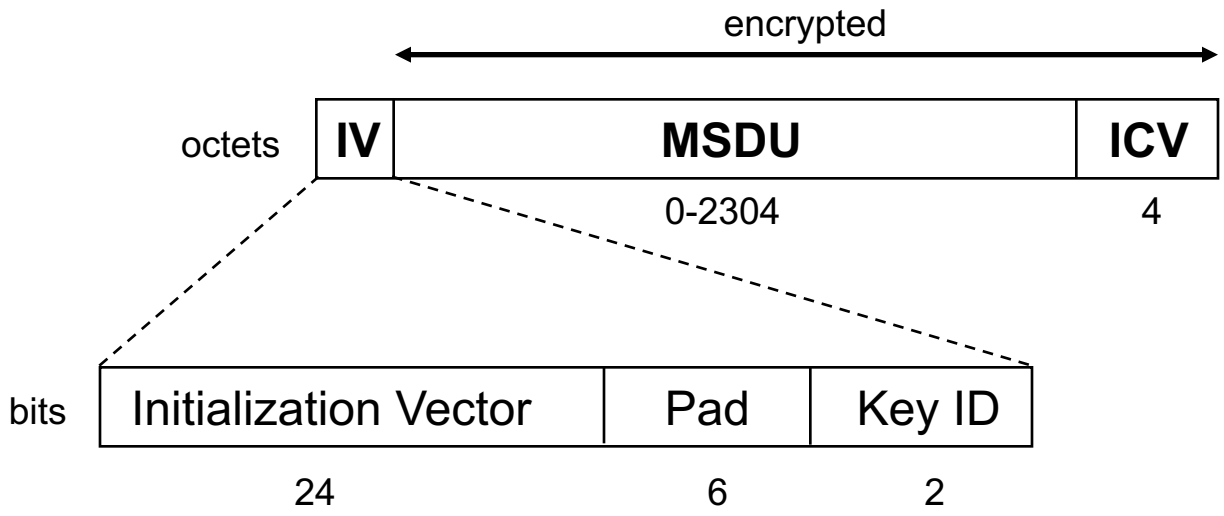
WEP-authentication (3)

- Challenge text sent in payload in cleartext (128 octets), random IV used
- Response sent in payload encrypted with the key shared between the AP and the station



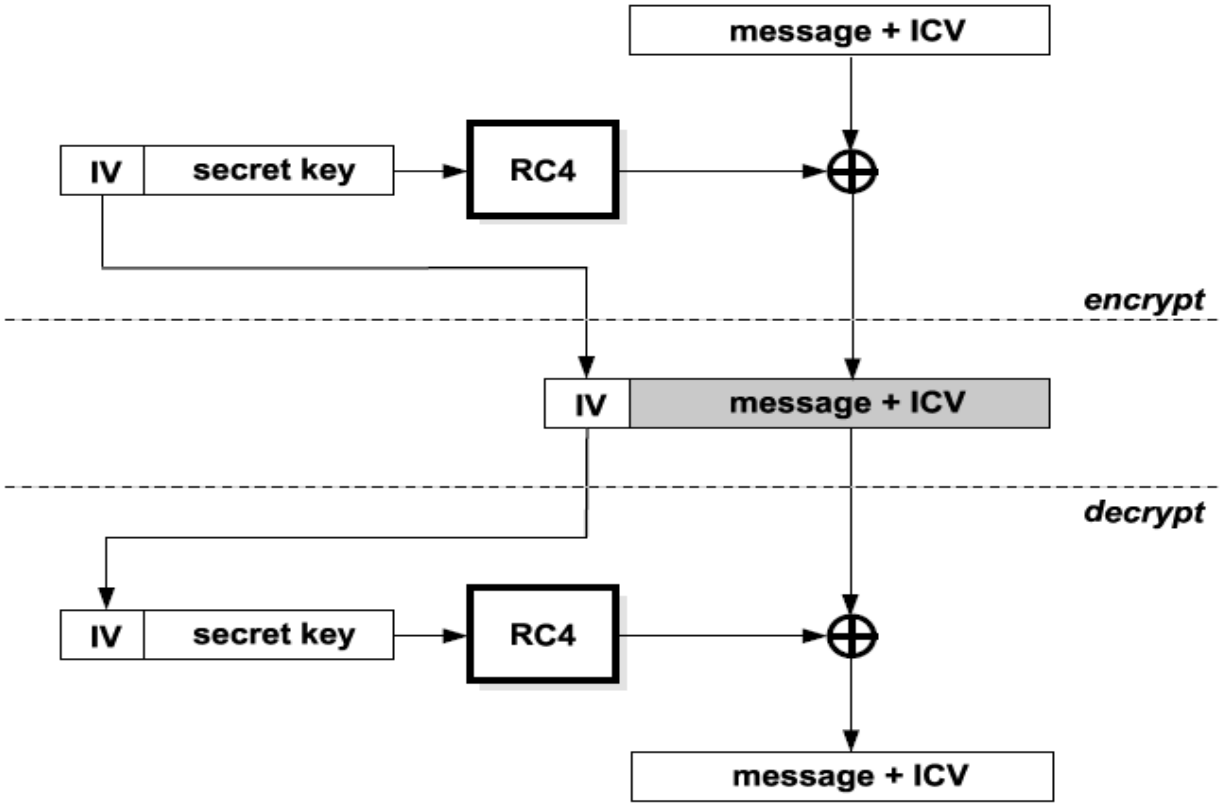


WEP - Authentication frame format



- MSDU = MAC Service Data Unit
  - this is the message
- ICV = Integrity Check Value (encrypted CRC)
  - protects message integrity
- IV = Initialization Vector is used to expand the key
  - expands the key

WEP confidentiality/integrity protection (1)

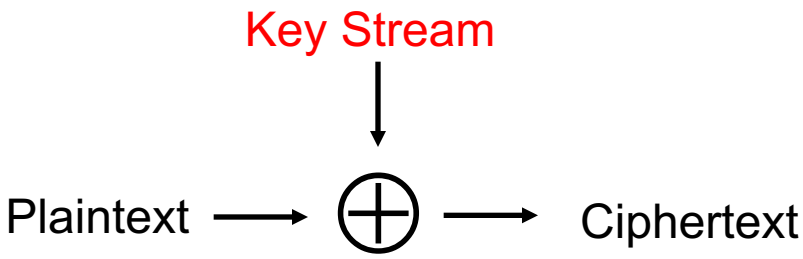


- RC4 Generates a key stream of a desired length from the key
- The key stream is XORed with plaintext data
- The result is ciphertext data

WEP confidentiality/integrity protection (2)

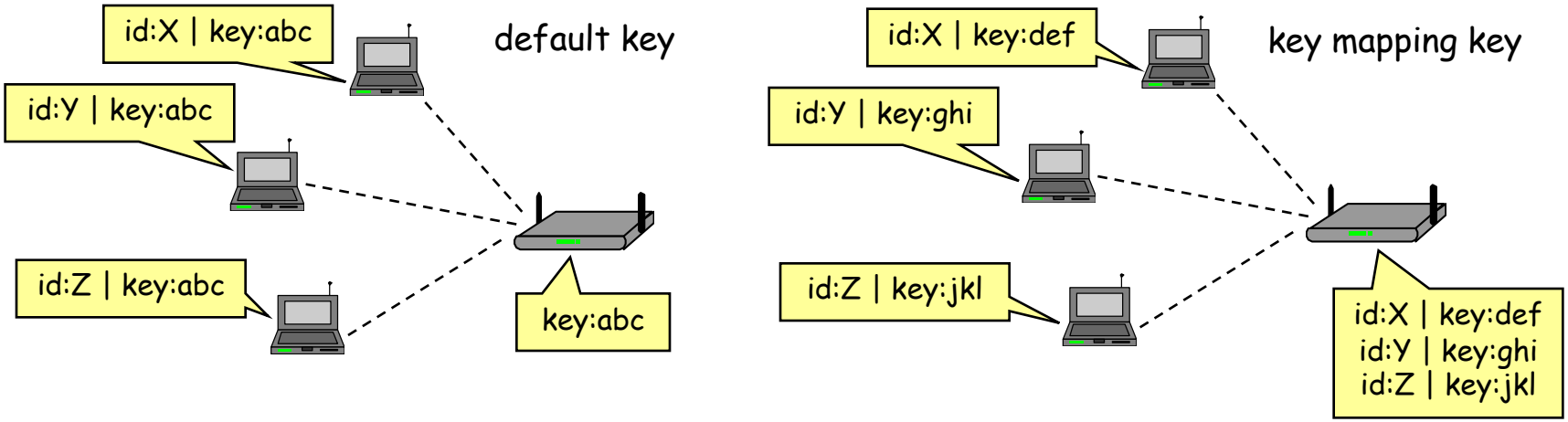
- RC4 is a stream cipher
  - given a short input key, it produces a pseudorandom sequence (key stream)
  - the key stream is always the same for the same key

- The output of the key stream is XORed with the plaintext to obtain a ciphertext:



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)

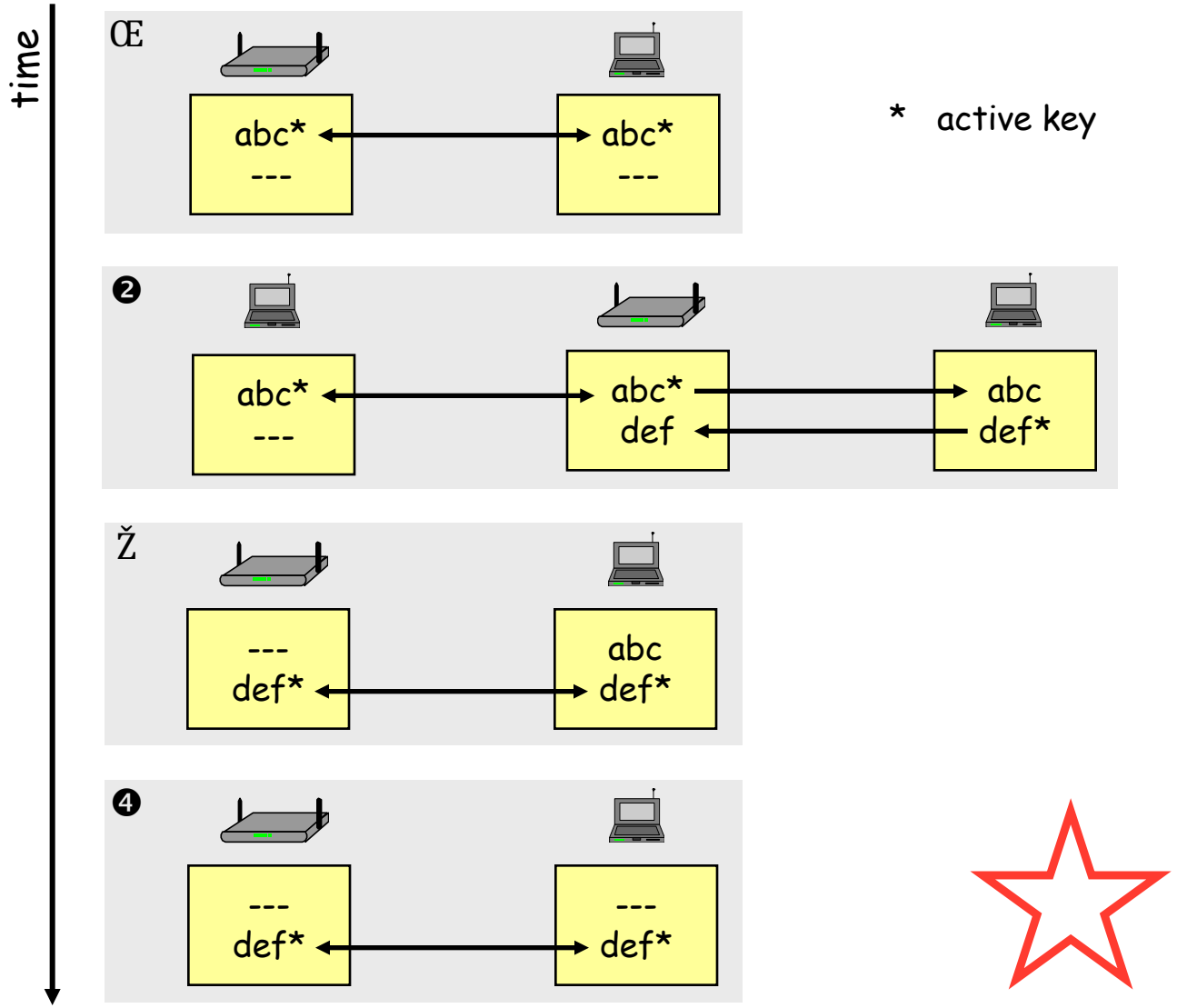


- in practice, often only default keys are supported
  - the default key is manually installed in every STA and the AP
  - each STA uses the same shared secret key → in principle, STAs can decrypt each other's messages

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 may associate to a rogue AP

- the same shared secret key is used for authentication and encryption
  - weaknesses in any of the two protocol can be used to break the key
  - different keys for different functions are desirable

- 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 → STA: r
  STA → AP: IV | r ⊕ K
  …

  where K is a 128 bit RC4 output on IV and the shared secret

- an attacker can compute r ⊕ (r ⊕ K) = K

- then it can use K to impersonate STA later:

  …
  AP → attacker: r'
  attacker → AP: IV | r' ⊕ K
  …

Authentication spoofing

- The adversary sees a single exchange between the AP and the station, (plaintext, ciphertext)
- Computes the key stream
- Then tries to associate with the same AP:
  - AP sends a challenge N
  - Station sends a reply K⊕<N,CRC(N)>
  - AND AP ASSOCIATES the station AS THE IV is chosen by the station
  - Catastrophic design !!!

WEP flaws – Integrity and replay protection

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

- 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:

$$\begin{aligned}&((M \mid CRC(M)) \oplus K) \oplus (\Delta M \mid CRC(\Delta M)) = \\ &((M \oplus \Delta M) \mid (CRC(M) \oplus CRC(\Delta M))) \oplus K = \\ &((M \oplus \Delta M) \mid CRC(M \oplus \Delta M)) \oplus K\end{aligned}$$

Message modification

Station and AP share a key (K);

Message M is encrypted into C = RC4(K,IV)⊕<M,CRC(M)>

- An attacker chooses D and wants the AP to accept that a fake message F=D⊕M came from the Station
- Attacker uses CRC-32 and RC4 linearity (all stream ciphers are linear), therefore
  - C' = C ⊕ <D,CRC(D)>
- Receiver will decrypt the packet

  <M',CRC(M')> = RC4(K,IV) ⊕ C'

  = C ⊕ <D,CRC(D)> ⊕ RC4(K,IV)

  = <M,CRC(M)> ⊕ <D,CRC(D)>

  = <M ⊕ D, CRC(M) ⊕ CRC(D)>

  = <M ⊕ D, CRC(M ⊕ D)> = <F, CRC(F)>

WEP flaws – Confidentiality

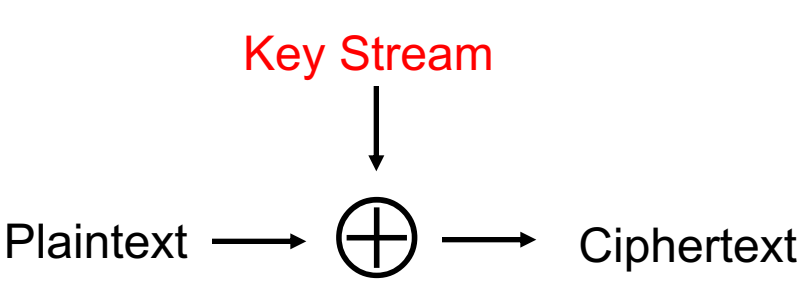
- IV reuse
  - IV space is too small
    - IV size is only 24 bits → there are 16,777,216 possible IVs
    - after around 17 million messages, IVs are reused
    - a busy AP at 11 Mbps is capable for transmitting 700 packets per second → IV space is used up in around 7 hours
  - in many implementations 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

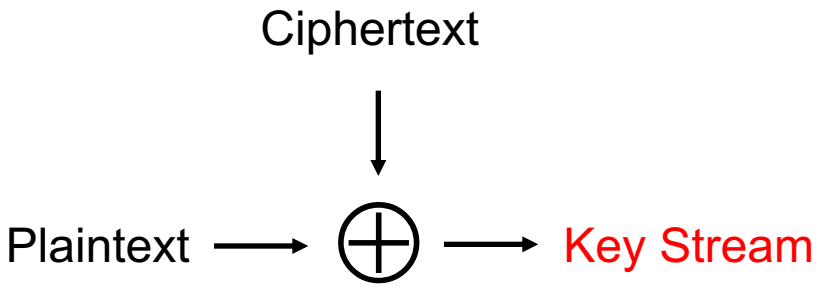
- weak RC4 keys
  - for some seed values (called weak keys), the beginning of the RC4 output is not really random
  - if a weak key is used, then the first few bytes of the output reveals a lot of information about the key → breaking the key is made easier
  - for this reason, crypto experts suggest 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 !!!

Two-time pad problem

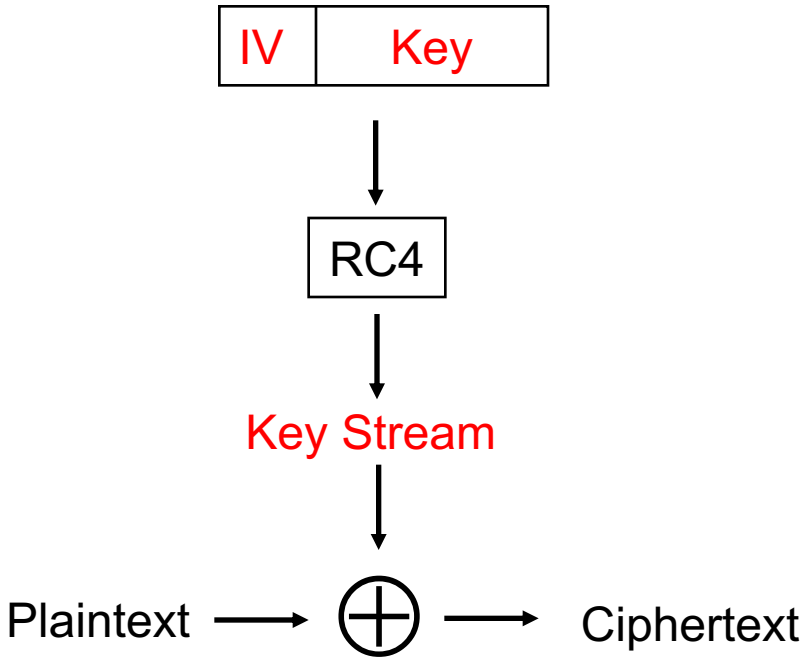
If the key stream is the same for each message, then it can be retrieved easily (with known plaintext) and can be reused for generating new messages ...





Initialization Vector

- Initialization vector (IV) alters the key stream
- Augments the key to generate a new key stream
- As the IV changes, so does the key stream change
- *Takes care of the key reuse ?*



Two-time pad problem

- In 802.11 IV is a 24 bit integer value
- 40 bit keys is augmented to 64 bits and a 104 bit key to 128 bits
- IV is sent in the clear (the receiver needs to know the vector to decrypt the message)

- 24 bits = only 16 million packets before the IV repeats itself

Implications to confidentiality

- If $P1 \oplus K = C1$ and $P2 \oplus K = C2$
  then $C1 \oplus C2 = P1 \oplus K \oplus P2 \oplus K = P1 \oplus P2$

- If the attacker learns part of one plaintext he can recover the other.
- Usually, having an XOR of two plaintexts enables the attacker to recover both of them.

More trouble

- Attacker knows the plaintext and the ciphertext (i.e. challenges the station and gets encrypted replies)
- It can then store all 2^24 (IV,Key stream) pairs (2^24x1500 = 24 GB)

- Next time that an attacker sees an encrypted packet, it can match it to the (IV, Key stream) pair (IV is in the packet) and XOR with the Key stream to obtain the plaintext !!!

WEP – Lessons learnt

1. engineering security protocols is a **very** risky business
   - you may combine otherwise strong building blocks in a wrong way and obtain an insecure system at the end
     - example:
       - stream ciphers alone are OK
       - challenge-response protocols for entity authentication are OK
       - but they shouldn't be combined
     - example:
       - 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 that you don't
     - they know the details of crypto algorithms better than you do
2. **avoid the use of WEP (as much as possible)**

Further WEP weakness (RC4 vulnerabilities)

- *Scott R. Fluhrer, Itsik Mantin, Adi Shamir, Weaknesses in the Key Scheduling Algorithm of RC4 Proceedings of SAC 01*
- *William A. Arbaugh. Narendar Shankar. YC Justin Wan, Your 802.11 Wireless Network has No Clothes, IEEE Wireless Communications, December 2002.*

- 802.11 WEP is flawed*
- A WEP key can be derived in 1M to 4M frames using statistical analysis
- Attacker is passive, and only 'listens' to wireless LAN
- Implemented in the AirSnort application

WEP crack will be shown during the exercises.

WEP Summary

- WEP is insecure
- publicly available tools are available to crack it
- it is still better than no protection

- Two major problems:
  - IV size/use
  - CRC does not depend on the key

- Some solutions
  - extending the size of the IV
  - replacing CRC with a strong MAC that depends on the key

- Lessons:
  - use existing (tested) protocol design
  - make the design process open and transparent (the secrecy is in the key NOT in the design)

After WEP - back to the basics

Wireless authentication

- User-based, centralized, strong authentication
- Mutual authentication of client and network

Wireless privacy

- Strong, effective encryption
- Effective message integrity check
- Centralized, dynamic key management

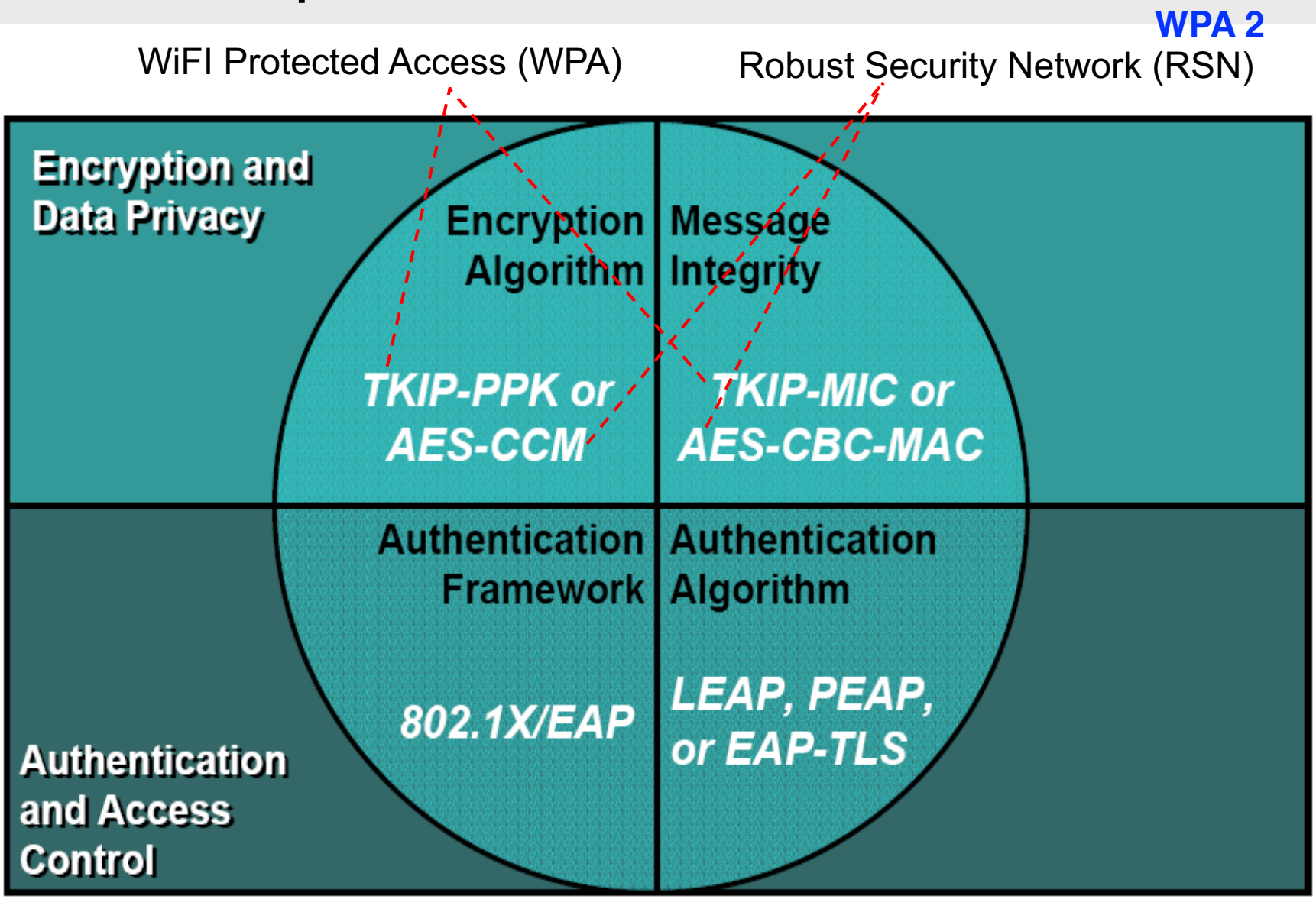
We need ...

- Encryption Algorithm
  - Mechanism to provide data privacy
- Message Integrity Check
  - Ensures data frames are tamper free and truly from the source address
- Authentication Framework
  - Framework to facilitate authentication of messages between clients, access point, and AAA server
- Authentication Algorithm
  - Mechanism to validate client credentials

802.11i – wireless security done well

- AES replaces RC4 w/TKIP
- Dubbed “WPA2” by WiFi Alliance
- Robust Security Network (RSN) for establishing secure communcations
  - Uses 802.1x for authentication
  - Replaces TKIP
- Counter Mode with Cipher Block Chaining (CCMP) for encryption
  - CCM mode of AES
  - 128-bit keys, 48-bit IV
  - CBC-MAC provides data integrity/authentication
  - CCMP mandatory with RSN
  - WRAP was initial selection, licensing rights/problems got in the way

802.11i components



from cisco.com

802.11i

- IEEE 802.11i defines a new type of wireless network called a robust security network (RSN)
- Transitional security network (TSN): Both RSN and WEP systems can operate in parallel.
- Most existing Wi-Fi cards cannot be upgraded to RSN as AES is not supported in all hardware
- All cards can be upgraded to WPA (RC4 supported in all hardware due to WEP)

WPA / RSN

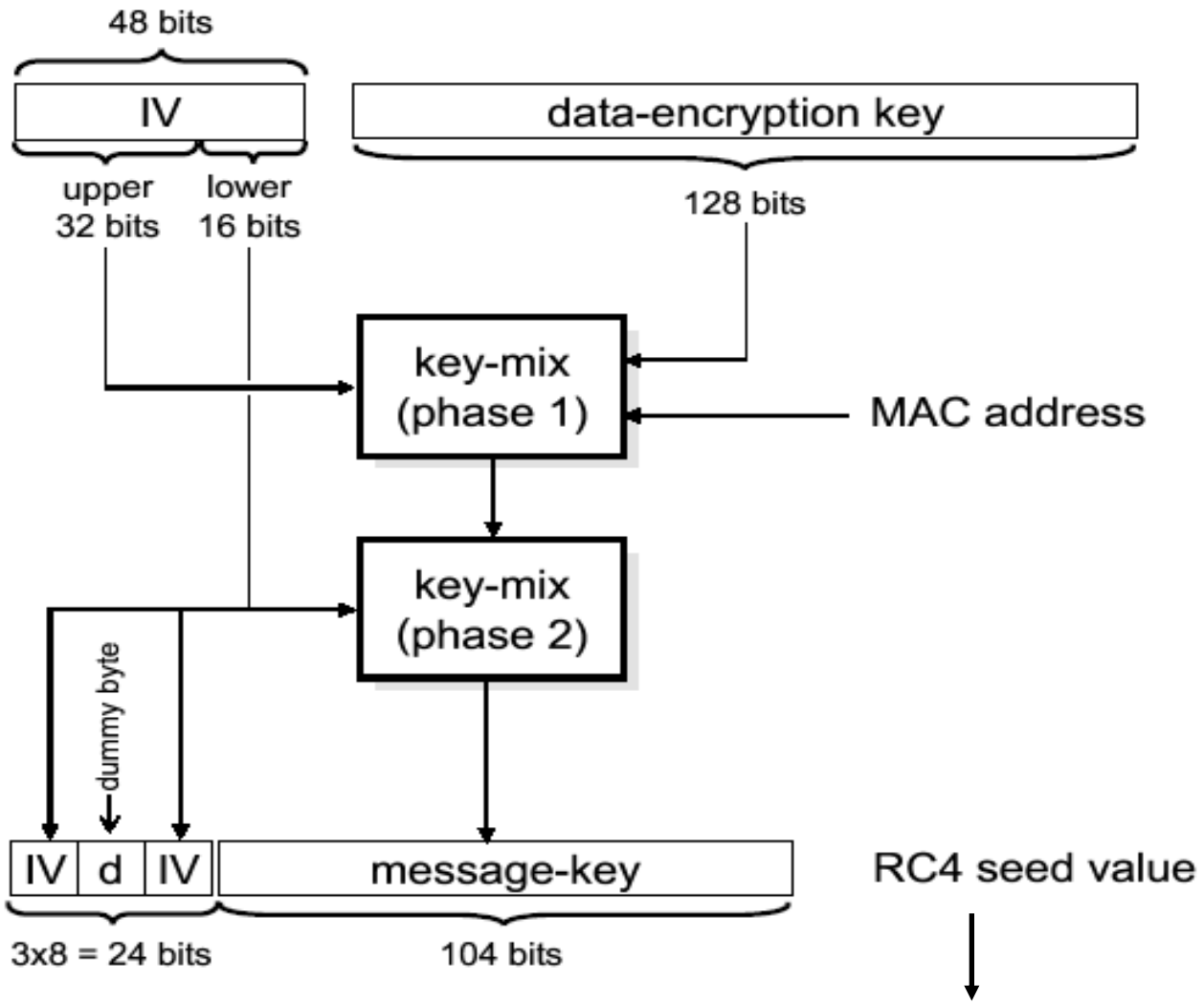
- Temporal Key Integrity Protocol (TKIP): based on the capabilities of existing Wi-Fi products.
- TKIP is allowed as an optional mode under RSN.
- Wi-Fi alliance adopted a new security approach based on the draft RSN but only specifying TKIP. This subset of RSN is called as WPA
- RSN also supports the AES cipher algorithm in addition to TKIP, whereas WPA focuses on TKIP.
- RSN and WPA share a single security architecture under which TKIP or AES based security protocols can operate.
- This architecture includes upper-level authentication, secret key distribution, and key renewal.

TKIP – enhanced encryption / integrity

- TKIP has to be secure and available as an upgrade to WEP systems.
- The implementation of WEP almost depends on the hardware assist functions.
- The hardware assist functions in these earlier systems cannot support AES-CCMP.
- **TKIP uses existing RC4 and upgrades the firmware.**

TKIP – new generation of RC4 seed value

Per-packet key mixing



Changes from WEP to TKIP

- Message integrity: add a message integrity protocol.
- IV selection and use: as counter (sequence no)
- Per-packet key Mixing
- Increase the size of IV.
- Key management.

IV selection and use

- IV size: 24 bits -> 48 bits
- IV use as a sequence number to avoid replay attacks.
- IV is constructed to avoid certain “weak keys.” (RC4 has some weak keys)

TKIP Message Integrity Code (MIC)

- ICV offers no real protection at all.
- All the well-known methods need a new cryptographic algorithm or require fast multiply operation.
- Michael uses no multiplications, just shift and add operations.
- Michael is vulnerable to brute force attacks.

Can be reversed!

802.11i overview

Station

Access Point

Auth. Server

Security capabilities discovery

802.1X authentication

802.1X key management

RADIUS key distribution

DATA protection
(TKIP, AES-CCMP)

WPA and RSN key Hierarchy

station

1. authenticate and derive Master Key

4-way handshake

2. derive Pairwise Master Key (PMK)

Auth. Server

distribute PMK

3. Use PMK to enforce 802.11 channel access: derive and use PTK

AP

- MK ≠ PMK or AP could make access control decisions
- instead of AS
- MK is fresh and bound to the session between STA and *AS*
- PMK is bound to *this* STA and *this* AP

Key derivation

Four separate keys for two layers' protection:

- EAP handshake and user's data.
  - Data Encryption key
  - Data Integrity key
  - EAPOL-Key Encryption key
  - EAPOL-Key Integrity key

- Pairwise transient key (PTK): the four keys

- Once that the keys are chosen:
  - TKIP encryption **or AES encryption and MAC**
  - TKIP MIC

In summary: WEP vs. WPA vs. WPA2

	WEP 1997	WPA 2003	WPA2 2004
Encryption	RC4	RC4	AES
Key rotation	None	Dynamic session keys	Dynamic session keys
Key distribution	Manually typed into each device	Automatic distribution available	Automatic distribution available
Authentication	Uses WEP key as AuthC	Can use 802.1x & EAP	Can use 802.1x & EAP

WPA-TKIP security

- **Recall:** TKIP was designed for transitional period
  - A study (from 2013) found that 70% of networks still allow TKIP

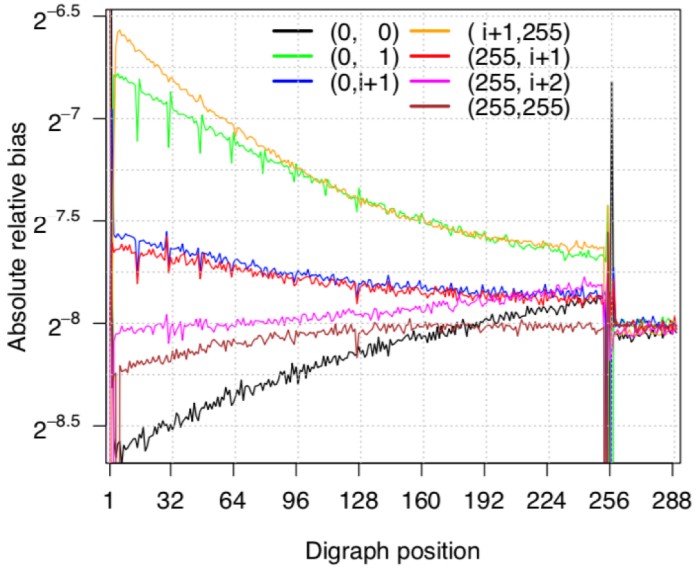
- Fairly successful – for an "ugly fix"
  - Uses old RC4 algorithm with per-packet key mixing

- Example attack:
  - AlFardan, Bernstein, **Paterson**, Poettering, Schuldt. On the security of RC4 in TLS and WPA. In *USENIX Security*, 2013
  - Requires 13 x 2^30 ciphertexts = 2000 hours of recorded data
  - http://www.isg.rhul.ac.uk/tls/RC4biases.pdf

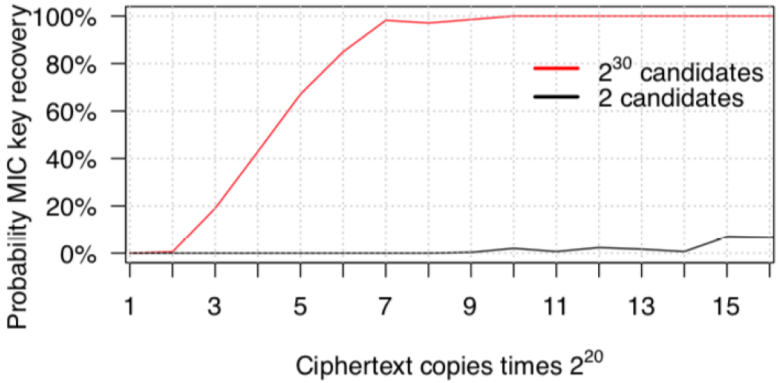
Efficient attack on WPA-TKIP

- **Main idea:** leverage **biases** in the RC4 key stream

- Relatively complicated attack
- **Steps** (simplified):
  - Generate large amounts of keystream
  - Using statistical tests detect biases in it
  - Establish a set of candidate plaintexts
  - Prune candidate plaintexts based on CRC
  - Recover plaintext
  - Recover MIC key by reversing
  - Inject false echo packets with ciphertext
  - Receive (decrypt) arbitrary packets



WPA-TKIP not secure



- **Attack practicality**
  - Capture traffic for ~1 hour
  - 9.5 x 2^20 ciphertext packets

- Vanhoef and Piessens.
  All Your Biases Belong To Us: Breaking RC4 in WPA-TKIP and TLS.
  In *USENIX Security*, 2015.
  https://www.usenix.org/system/files/conference/usenixsecurity15/sec15-paper-vanhoef.pdf

- **Conclusions:**
  - As expected transitional WPA-TKIP is vulnerable to attacks
  - Usage of WPA 2 is advised

WPA2 security

- **Recall:** WPA 2 uses modern and strong cryptographic primitives
  - AES for encryption
  - CBC-MAC for message integrity

- No major flaws found ☺

- However, vulnerability in 4-way handshake implementations identified…
  - Vanhoef and Piessens.
    Key Reinstallation Attacks: Forcing Nonce Reuse in WPA2
    CCS 2017
    https://papers.mathyvanhoef.com/ccs2017.pdf

WPA2 remains secure

- Handshake attack main idea
  - Client executes 4-way handshake to negotiate a fresh session key
  - Client installs this key after receiving message 3
  - However, as messages may be lost, AP may retransmit message 3
  - Each time client receives message 3, it will reinstall the same session key
  - This resets nonce and replay counter needed for confidentiality!!
  - **Interesting side note:** the 4-way handshake was formally proven “secure”

- Attack impact
  - Allowed arbitrary packet decryption
  - Implementation can be patched in backwards-compatible manner
  - **WPA 2 is secure (when used with latest updates)**

WPA3

- Successor of WPA2, announced in 2018
- Does not define new protocols
- Instead mandates which existing protocols must be supported
  - Mandates a handshake protocol called Dragonfly
  - Dragonfly prevents offline dictionary attacks and provides forward-secrecy

- First research results…
  - Minor side channel: AP response times during handshake may leak information about the password
  - Vanhoef and Ronen. *Dragonblood: Analyzing the Dragonfly Handshake of WPA3 and EAP-pwd.* To appear in **IEEE S&P 2020** https://papers.mathyvanhoef.com/dragonblood.pdf

Summary of WiFi security

- MAC and SSID – easy to bypass
- WEP – was designed to improve security, but turned out to be completely broken (many problems)
- WPA – TKIP was intended as transitional solution (not that bad, but should be avoided)
- WPA2 – based on AES and other good practices, remains secure
- WPA3 – recent refinement of the standard

MAC SSID WEP 1997 WPA-TKIP 2003 WPA2 (AES) 2004 WPA3 2018

Reading material

1. Borisov, Goldberg and Wagner
   Intercepting Mobile Communications: The Insecurity of 802.11
   Mobicom 2001
   http://www.isaac.cs.berkeley.edu/isaac/mobicom.pdf

2. Vanhoef and Piessens
   Key Reinstallation Attacks: Forcing Nonce Reuse in WPA2
   CCS 2017
   https://papers.mathyvanhoef.com/ccs2017.pdf