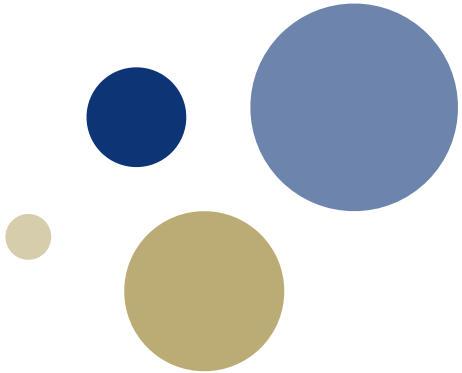
NTNU

Norwegian University of Science and Technology



TTM4137 - Wireless Network Security
Lecture 3: WEP/WPA

Caroline Sandsbråten
Department of Information Security and Information Technology
NTNU, Trondheim

Fall 2021

Credit to Anamaria Costache whos lecture slides this is based on.

Who am I?

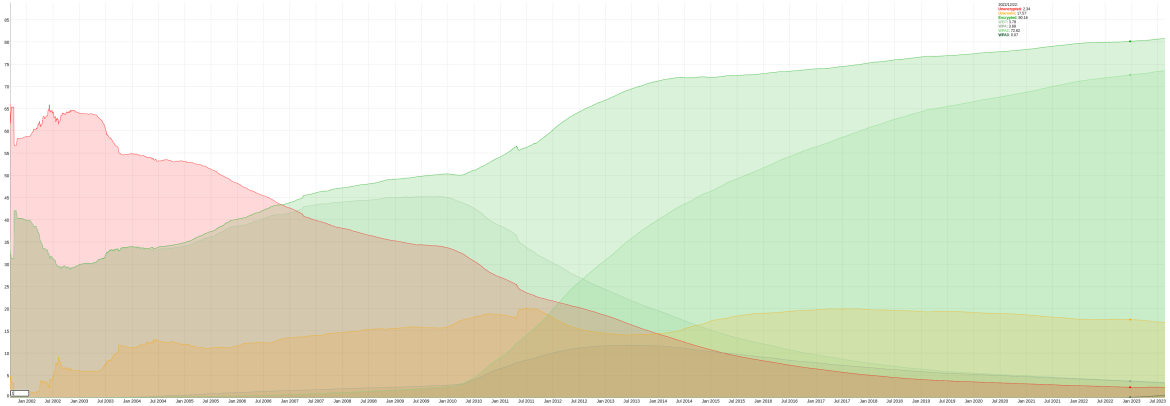
- 2nd year PhD student at IIK
- Researching lattice-based crypto
- Master thesis on practical attacks on ECDSA/EC-Schnorr
- Volunteer at ITK and Fotogjengen at Studentersamfundet
- Also a substitute teacher in TTM4205

802.11 Security

- By default off
- Authorisation to AP based on MAC-address
- IEEE 802.11: Wired Equivalent Privacy (WEP)
- IEEE 802.11i-draft: Wi-Fi Protected Access (WPA)
- IEEE 802.11i: Robust Security Network (RSN/ WPA2)

WEP

- Original security mechanism of 802.11
- Great example of a bad example
- Deprecated in 2008
- Still used a bit



2023/02/24:
Unencrypted: 2.29
Unknown: 17.46
Encrypted: 80.32
WEP: 3.7
WPA: 3.6
WPA2: 72.85
WPA3: 0.17

Source: https://wigle.net/stats

Security Goals

- Data Confidentiality
  - Protect against eavesdropping
  - Only "the right people" have access to their data
- Data Integrity
  - Data should not be changeable
  - Message injection and modification
- Network Access Control
  - Restrict unauthorized users access to a network
- What other goals should we have?

Reaching our security goals

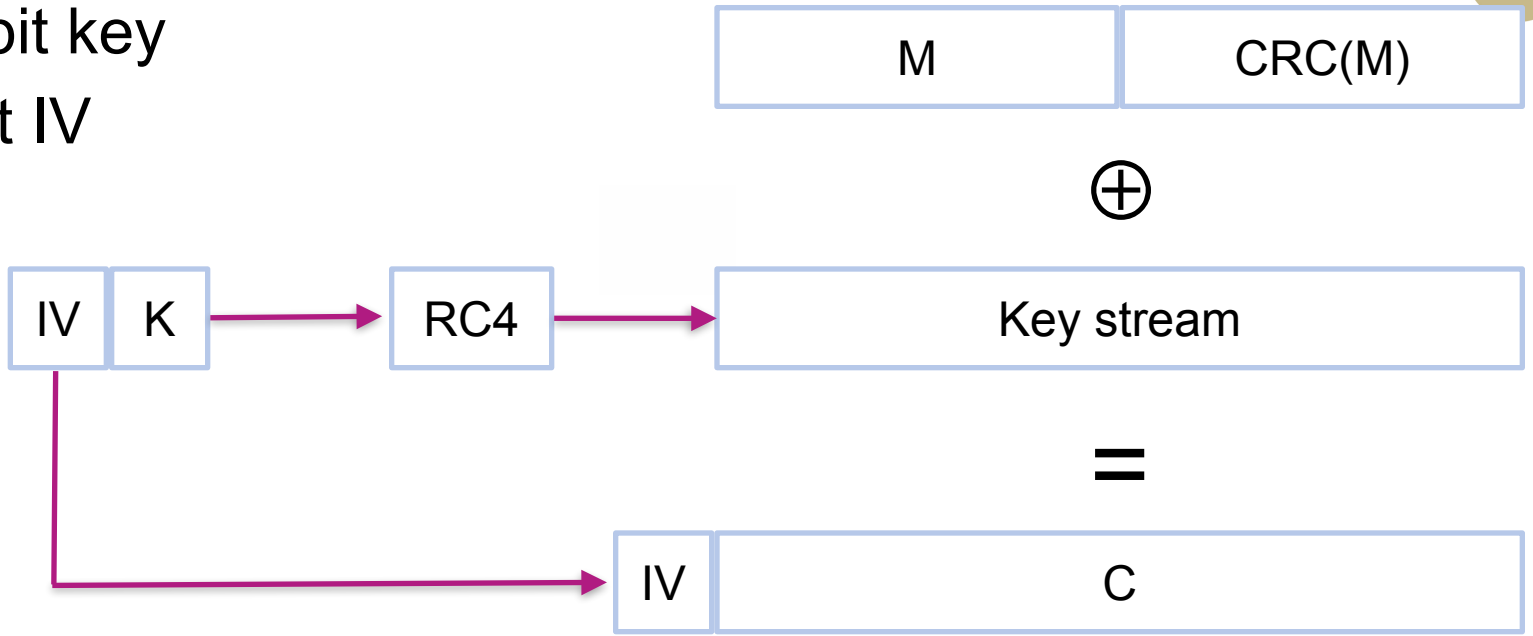
- Key management
  - Single cryptographic key among all devices in a BSS
  - Export restrictions: 40 bit at the time of WEP
- Data confidentiality
  - Encrypt data frames
- Integrity
  - Add a checksum

Limitations of our WEP design

- CRC is not suitable for detecting intentional modifications
- The small keysize due to export limitations
- A whole BSS sharing just 1 key can't possibly be a good idea?
  - Why?
- Only protecting data frames not control frames

WEP encryption

- Use RC4
- 104 bit key
- 24 bit IV

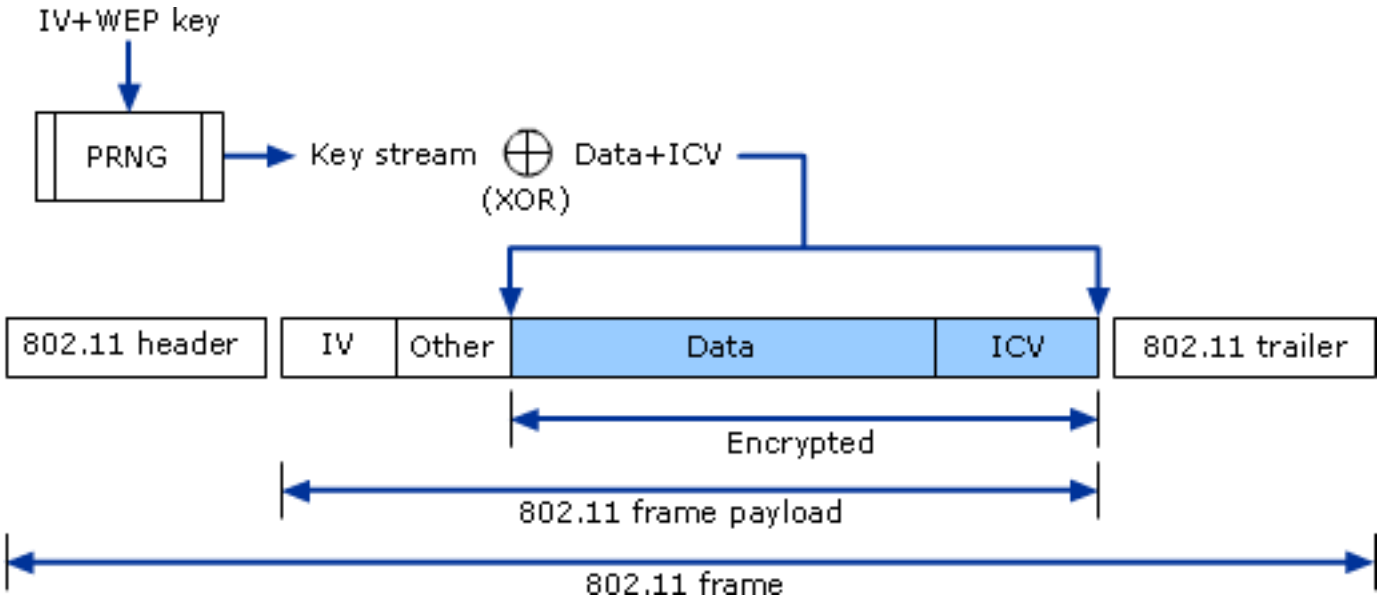


RC4

- Encryption
  - Plaintext ⊕ Keystream = Ciphertext
- Decryption
  - Ciphertext ⊕ Keystream = Plaintext

WEP in Practice (1)

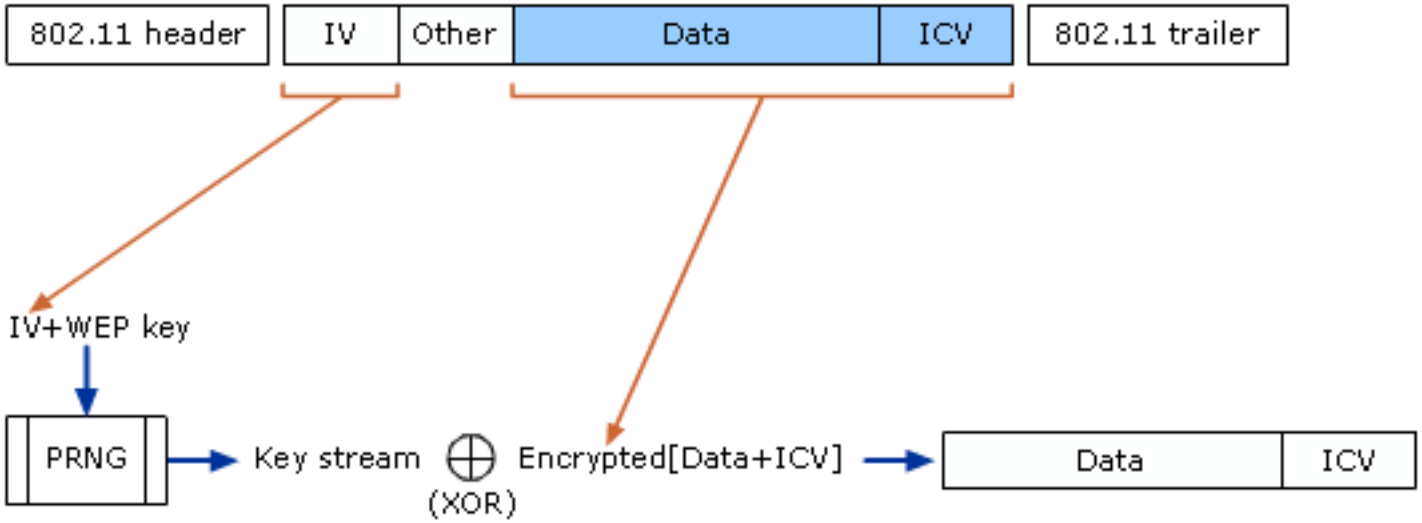
- Encryption



Source: https://learn.microsoft.com/pt-pt/previous-versions/windows/server/cc757419(v=ws.10)?redirectedfrom=MSDN

WEP in Practice (2)

- Decryption



Source: https://learn.microsoft.com/pt-pt/previous-versions/windows/server/cc757419(v=ws.10)?redirectedfrom=MSDN

WEP: WEAKNESSES AND ATTACKS

Authentication

- Challenge
  - AP sends a random 128-bit challenge text
- Response
  - STA encrypts the challenge text using WEP and sends the CT to the AP
- Success
  - AP decrypts and compares the plaintext with the challenge

Authentication (2)

- What can the adversary (A) find by a passive attack?
  - A eavesdrops on both the plaintext and ciphertext and find the keystream corresponding to the IV
  - $c \oplus m = m \oplus keystream \oplus m = keystream$
- Is the authentication secure?
  - No, why?
  - A knows the keystream for a given IV and can reuse the IV to XOR the challenge with the recovered keystream and return a valid keystream

Authentication (3)

- Do we have mutual authentication?
  - No, AP does not authenticate to the STA
- What type of attack does this facilitate?
  - Feasibility of rogue entities
  - A rogue AP sends challenges and receives answers

Linearity

- Consider the case where there is no CRC. What can an adversary do?
  - Flip bits of the plaintext at will
- With CRC, how?
  - Intercept c
  - change c
    - c' = c⊕m'
  - Send c' instead of c
  - Receiver gets:
    - c'⊕keystream = c⊕m'⊕keystream=m⊕k⊕m'⊕k=m⊕m'
    - m' has 1 on the positions of the bits to be flipped in m, and 0 otherwise

IV Reuse

- What is the first thing that happens if we reuse an IV with a key K?
  - We get the same keystream
- If this happens, is the system secure?
  - c1 = m1 || CRC(m1) ⊕ keystream
  - c2 = m2 || CRC(m2) ⊕ keystream
  - c1⊕c2 = m1 || CRC(m1) ⊕ m2 || CRC(m2)
- Knowing m2 then leads to knowing m1
- But will this actually happen?
  - Yes, we only have 2^{24} possible IVs, repeating every few hours

Other Problems

- RC4 is not a secure PRG
  - Weak keys
  - Direct attacks
- Replay attacks
  - No replay protection
- Message injection
  - For one IV, once a keystream is known
- Man-in-the-Middle attacks
  - Find a new key, rogue AP

Other problems (2)

- Key recovery
- It takes just a few seconds to recover the 104 bit key

- Smashing WEP in a passive attack: https://www.youtube.com/watch?v=JDG9ZAmfIBs
  A bazillion other attacks on WEP: https://scholar.google.com/scholar?hl=no&as_sdt=0%2C5&q=wep&btnG=

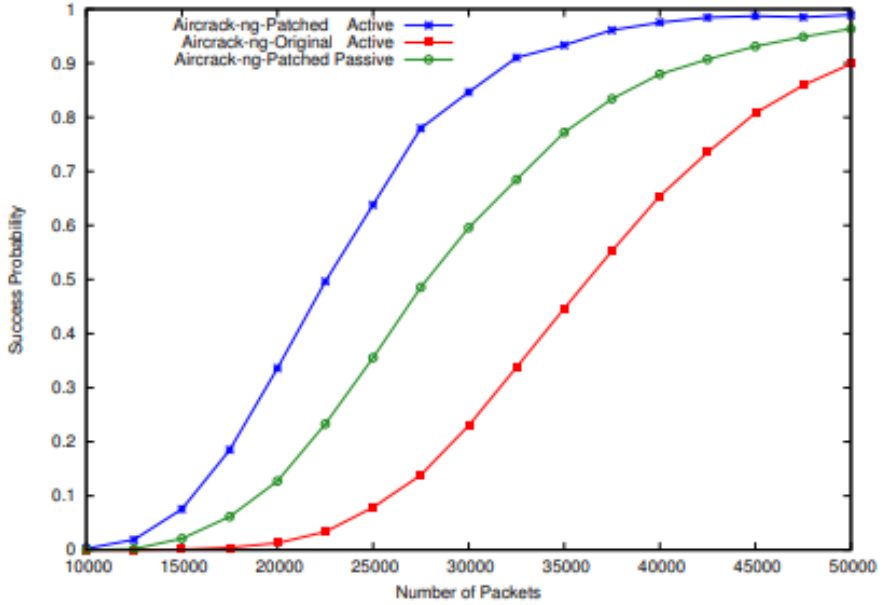


Fig. 5. Our attacks success probability (both active and passive attacks) with respect to the number of packets compared to Aircrack-ng in active attack mode.

What can we learn from the design of WEP?

- Clear design goals ✓
- Open standard ✓
- Use well studied crypto primitives and protocols ✗
- Public review and competition ✗

Other lessons

- Designing security protocols is hard
- Involve cryptographers in your work
- Make the system public
- Use precise definitions and security analysis
- Remember that attackers are not limited by models

WPA

Outline

- Improvements from WEP
- TKIP
- Authentication
- Key management

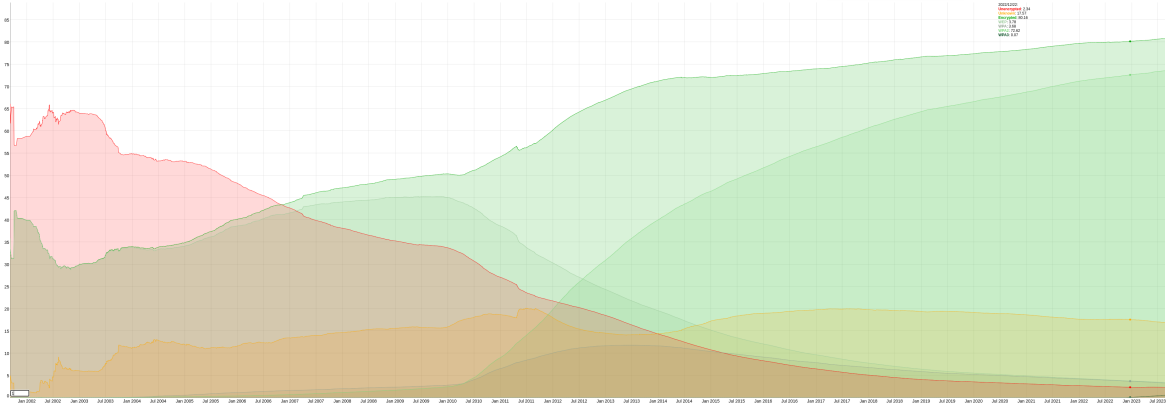
802.11 Security

- By default off
- Authorisation to AP based on MAC-address
- IEEE 802.11: Wired Equivalent Privacy (WEP)
- IEEE 802.11i-draft: Wi-Fi Protected Access (WPA)
- IEEE 802.11i: Robust Security Network (RSN/ WPA2)

24

WPA

- Fix to WEP by the Wi-Fi alliance
- Addition of TKIP
- Deprecated, but still sees about the same use as WEP
- Easy to adopt



2023/02/24:
Unencrypted: 2.29
Unknown: 17.46
Encrypted: 80.32
WEP: 3.7
WPA: 3.6
WPA2: 72.85
WPA3: 0.17

Source: https://wigle.net/stats

WPA - Remember from WEP

- No (bad) confidentiality
  - Short IVs leads to reuse of RC4 keystream
  - Key recovery possible in seconds
- No integrity
  - CRC is not a MAC
- No replay protection
- No key management
  - Single master key shared by all users connected to the WLAN
- Pointless user authentication

WPA - improvements to WEP

- Authentication
  - 802.1X authentication is required
- Key Generation and Distribution
  - IEEE 802.1X
- Integrity
  - We finally see some integrity, Michael to the rescue
- Encryption
  - TKIP
    - A new key is used for each frame
  - Optional AES

TEMPORAL KEY INTEGRITY PROTOCOL (TKIP)

Temporal Key Integrity Protocol (TKIP)

- "Easy" solution to replace WEP
  - Solution for already deployed hardware
- Temporal keys generated by the protocol
- Device specific keys
  - Temporal key is mixed with the transmitter MAC address
- Frame specific keys
  - IV is used as input for the key
- Longer IV: 48 bits
- TKIP Sequence Counter (TSC)
- Michael introduced, replacing CRC/ICV for integrity

Michael

- Uses two 64-bit keys, one in each direction of communication
- Lightweight
  - Uses only substitutions, XOR and rotations
- Input
  - Source and destination MAC addresses, plaintext
- Output
  - Output 64 bit MIC

Randomised attack on Michael

- Michael provides 20 bits of security against a randomised attack
- Is this good security?
  - No!
  - $1/2^{20}$ = 1 in a million chance of a random MIC value being accepted
- Solutions to this?
  - Rate limiting
  - Attack detection
  - Closing down communications for a while
  - Downsides of this?
    - Denial-of-Service (DoS)

802.1X

802.1X Architecture

- Supplicant
  - An entity that requests access
    - Laptops, phones, etc.
- Authenticator
  - An entity that enforces authentication before allowing access
    - AP
- Authentication server
  - Authenticates supplicants
  - Remote Authentication Dial-In User Service (RADIUS)
  - Can be contained in an AP, but this is not typical

EAP - Authentication sequence

- Extensiple Authentication Protocol (EAP): A Point-to-Point Protocol (PPP) for authentication
- IEEE 802.1X defines EAP over LAN (EAPOL) which is an encapsulation of EAP over LAN

EAP Authentication

- Generic authentication framework
- Not a standalone protocol on its own
- Encapsulates other concrete protocols

- Defines:
  - Four generic messages
    - Request, Response, Success, Failure
  - A key derivation process

EAPOL

- EAP RFC does not specify how messages should be passed around
- IEEE 802.1X provides the description for EAPOL
- The EEE 802.1X committee described 5 EAPOL messages:
  - EAPOL-start
  - EAPOL-Key
  - EAPOL-Packet
  - EAPOL-Logoff
  - EAPOL-Encapsulated-ASF-Alert (not used by WPA)

RADIUS

- Remote Authentication Dial-In User Service (RADIUS)
- Defines:
  - A set of functionality that should be common across authentication servers
  - A protocol that allows other devices to access these capabilities
- Specified by IETS and is designed for TCP/IP
- Lots of updates, f.ex. EAP over RADIUS (RC 2869)

RADIUS (2)

- Challenge-Response mechanism
- Core protocol is very simple
  - Access-Request (AP => AS)
  - Access-Challenge (AP <= AS)
  - Access-Accept (AP <= AS)
  - Access-Reject (AP <= AS)
- To use with EAP we change the purpose of some of the messages
  - Works because RADIUS is flexible
  - RADIUS messages is composed of attributes

WPA KEY HIERARCHY

Pairwise WPA Key Hierarchy

- Pairwise Master Key (PMK)
  - 256 bit symmtric key
  - Pre-shared or supplied from upper layer (authentication server)
- Pairwise Transient Key (PTK)
  - PTK = f(PMK, Nonce1, Nonce2, MAC1, MAC2)
- PTK:
  - Up to 4 keys
    - EAPOL-keys (encryption and integrity keys)
    - Data encryption key
    - Data integrity key

Group WPA Key Hierarchy

- Used for broadcast and multicast
- Group Master Key (GMK)
  - 256 bit symmetric key
  - Generated by AP
- Group Transient Key (GTK)
  - GTK = f(GMK, Nonce, MAC_AP)
- GTK
  - Encryption Key (128 bit)
  - Integrity Key (128 bit)

Summary from Today

- WEP, WPA (TKIP, Michael, EAP, RADIUS)

- Next Lecture:
  - WPA2, WPA3

Further Reading

- Chapter 4, 6 of Real 802.11 Security: Wi-Fi Protected Access and 802.11i
- Chapter 7, 8, 10, 11 of Real 802.11 Security: Wi-Fi Protected Access and 802.11i

- Chapter 1, 2, 3, 4, 5 of Serious Cryptography: A Practical Introduction to Modern Encryption
  - Not included as course material, but I highly recommend this book