NSS Wireless Security Tutorial

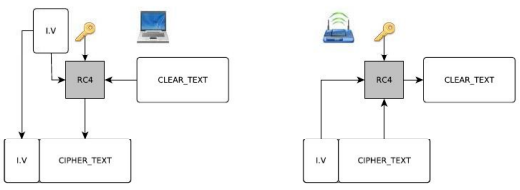
WEP uses RC4 cipher to encrypt data:

- Symmetric: The device and the A.P share the same key
- Stream Cipher: sequence of clear-text → sequence of cipher-text
- Simple
- Fast
- Weak

NSS Wireless Security Tutorial

- Shared key is 128 bits but..,
- Actual key size is 104 bits
- The 24 remaining bits serve as an Initialization Vector (IV) → not Secret
- The IV is different for each frame
- The IV is in clear-text in every frame (recall: self-synchronizing)

NSS Wireless Security Tutorial



NSS Wireless Security Tutorial

- RC4 is a **stream cipher:**
  - Creates a stream of data that is combined (XOR) with the data
  - 00110101 xor 11100011 = 11010110 (if the same then 0)
  - But: 00110101 xor 11100011 = 11010110 and 11010110 xor 11100011 = 00110101
  - Which means that:
    - Plaintext xor Random = Cipher text
    - Ciphertext xor Random = Plaintext

NSS Wireless Security Tutorial

Problem with XOR

$$C_1 = P_1 \oplus K$$
$$C_2 = P_2 \oplus K$$
$$C_1 \oplus C_2 = P_1 \oplus P_1$$

NSS Wireless Security Tutorial

NSS Wireless Security Tutorial

- What is the problem here?

NSS Wireless Security Tutorial

- Many <ciphertext, plaintext> pairs
- Remember:
  - ***P xor R = C → C xor P = R***
  - Attacker asks for a new challenge
  - Xors it with R and uses the same IV as the captured one...

NSS Wireless Security Tutorial

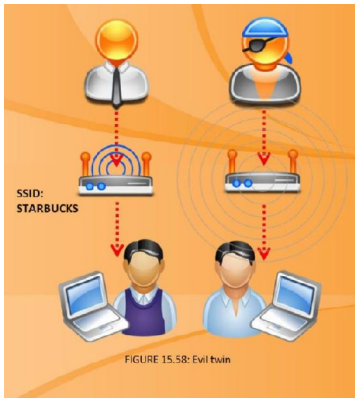
- Even worse:
  - 2^24→ 17 million IVs will be exhausted really fast
  - Collisions mean that IV + Key remains the same
    - C1 xor C2 = … = P1 xor P2 (for a collision)
    - As more collisions occur bigger parts of the key are decoded
    - Some packet fields do not change :) (e.g., IP addresses)

NSS Wireless Security Tutorial

- Now that you know the key what can you do?

NSS Wireless Security Tutorial

NSS Wireless Security Tutorial



NSS Wireless Security Tutorial

- But if you don't have the key, what else can you do (besides authenticate)

NSS Wireless Security Tutorial