Wireless Security

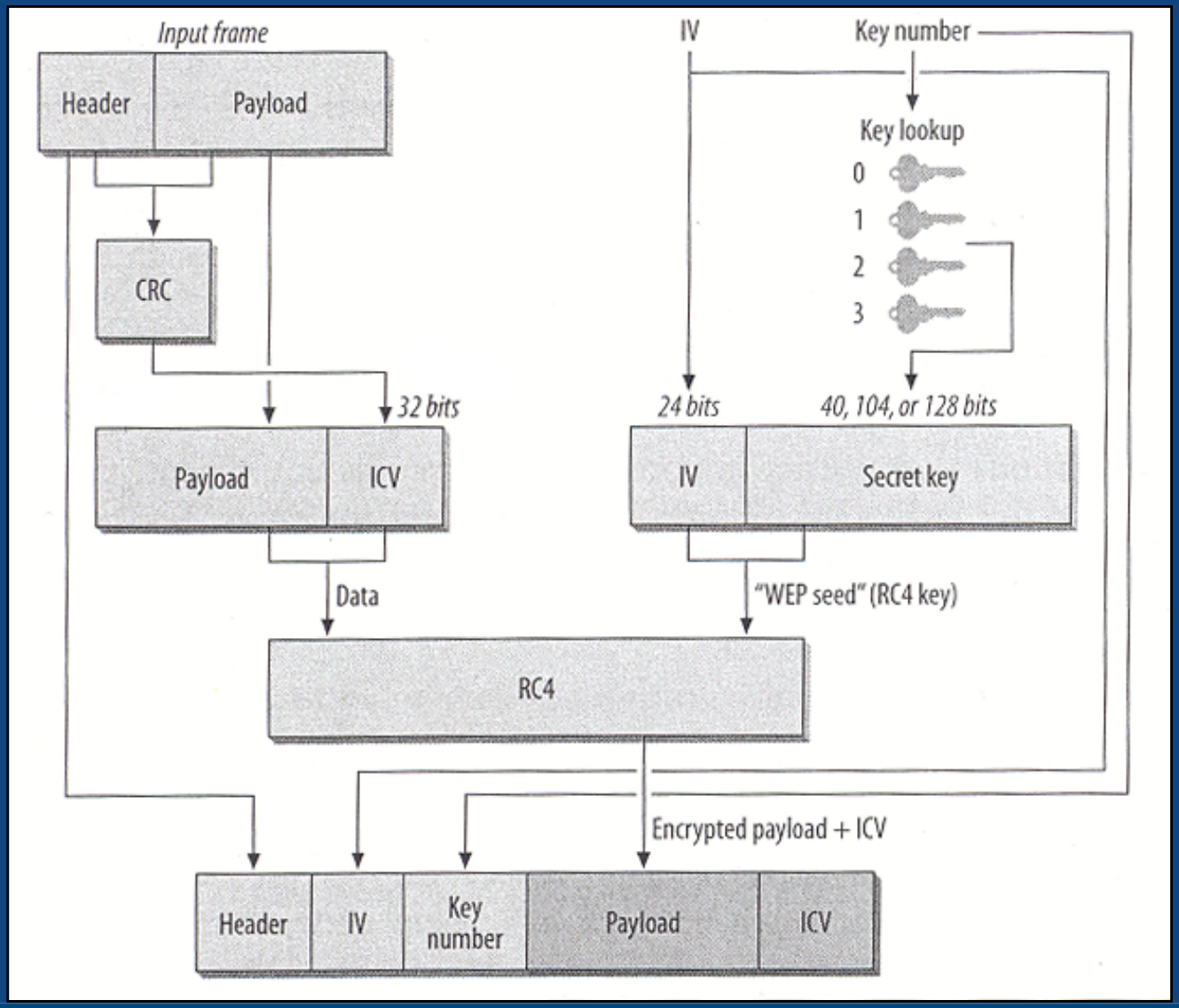
EECE 571B “Computer Security”

Konstantin Beznosov

The final nail in WEP's coffin

Bittau, A., Handley, M., Lackey, J., "The final nail in WEP's coffin," Security and Privacy, 2006 IEEE Symposium on, vol., no.pp. 15 pp.-, 21-24 May 2006.

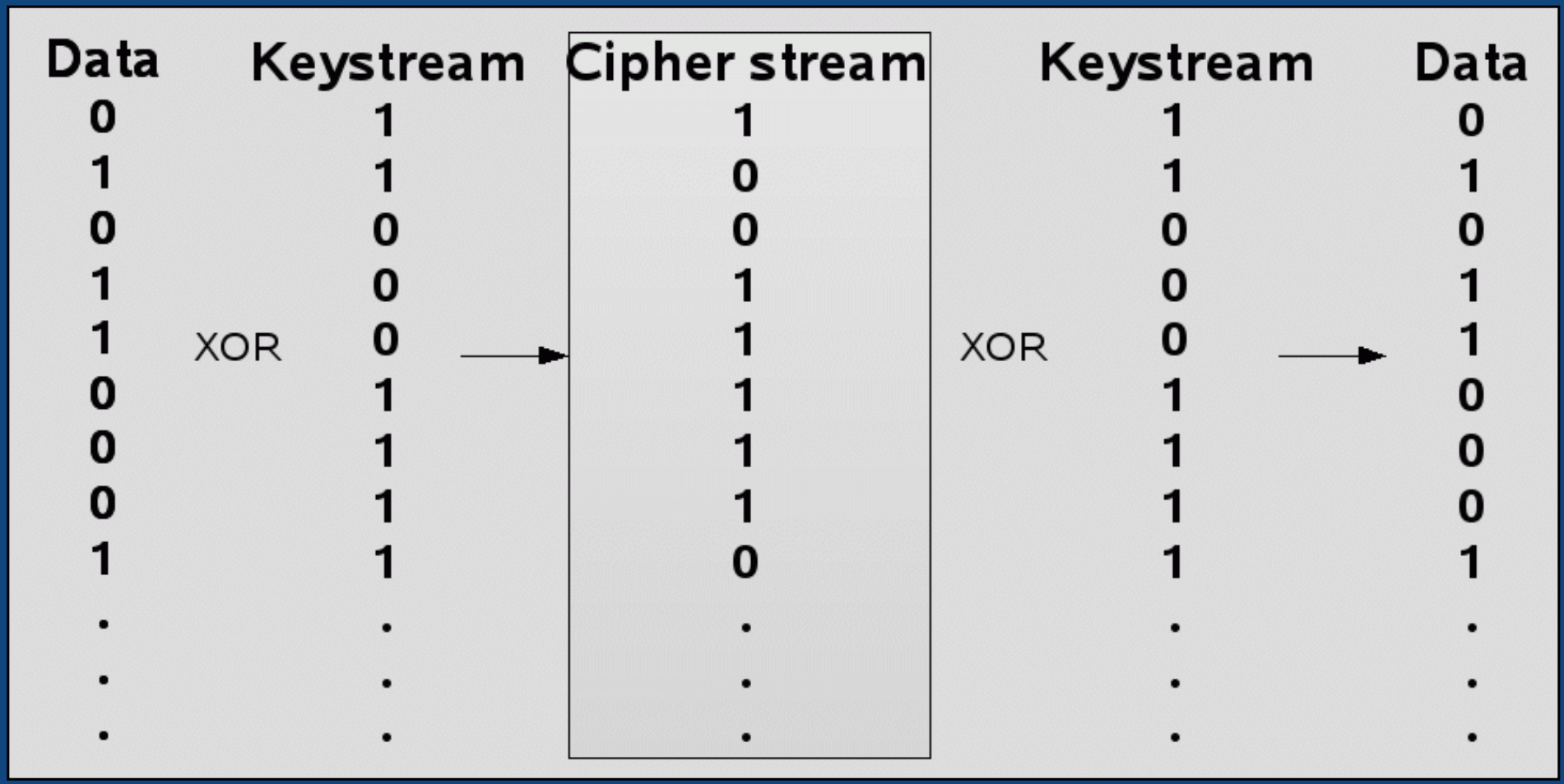
how WEP works



WEP encapsulation summary

- A master key shared between the end points
- Encryption Algorithm = RC4
- Per-packet encryption key = 24-bit IV concatenated to a master key
- WEP allows IV to be reused with any frame
- Data integrity provided by CRC-32 of the plaintext data (the “ICV”)
- Data and ICV are encrypted under the per-packet encryption key

symmetric key cipher



History of WEP Attacks

- Brute-force
  - Try all possible combinations (40-bit key)
    - Less than a month on a single computer
  - Passphrase generated keys lessen the effort
    - Keys connected to actual meaningful words

Solution: 104-bit WEP defends against attack

History of WEP Attacks

- Keystream Re-use
  - Known cleartext then keystream recovered

    ciphertext xor cleartext → keystream
  - "shared key authentication"
    - AP sends cleartext challenge "encrypt this…"
    - Peer sends encrypted version back to AP
    - Snooping attacker now has keystream for IV
      - 802.11 standard says "don't reuse keystream for IV"
    - However, attacker can now transmit indefinitely

  Solution: SSID cloaking & MAC address filters

History of WEP Attacks

- Weak IV Attacks
  - Key could be calculated (based on RC4 properties)
  - Takes approx. 1,000,000 packets
  - Major threat – automated tool (anyone can hack)

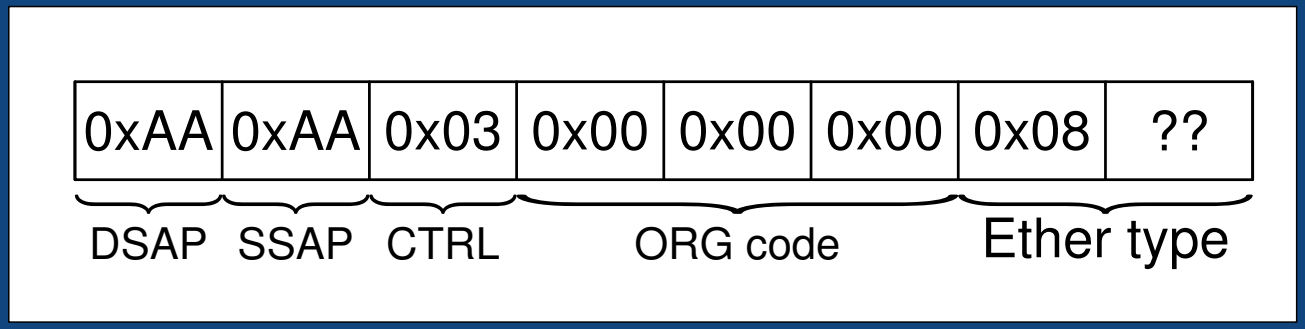
Solution: NIC hardware filters weak IVs

- could now take days
- Problem: fewer keystreams (< 2^{24} keystreams)

A single legacy host (without filter) can compromise network

new attacks

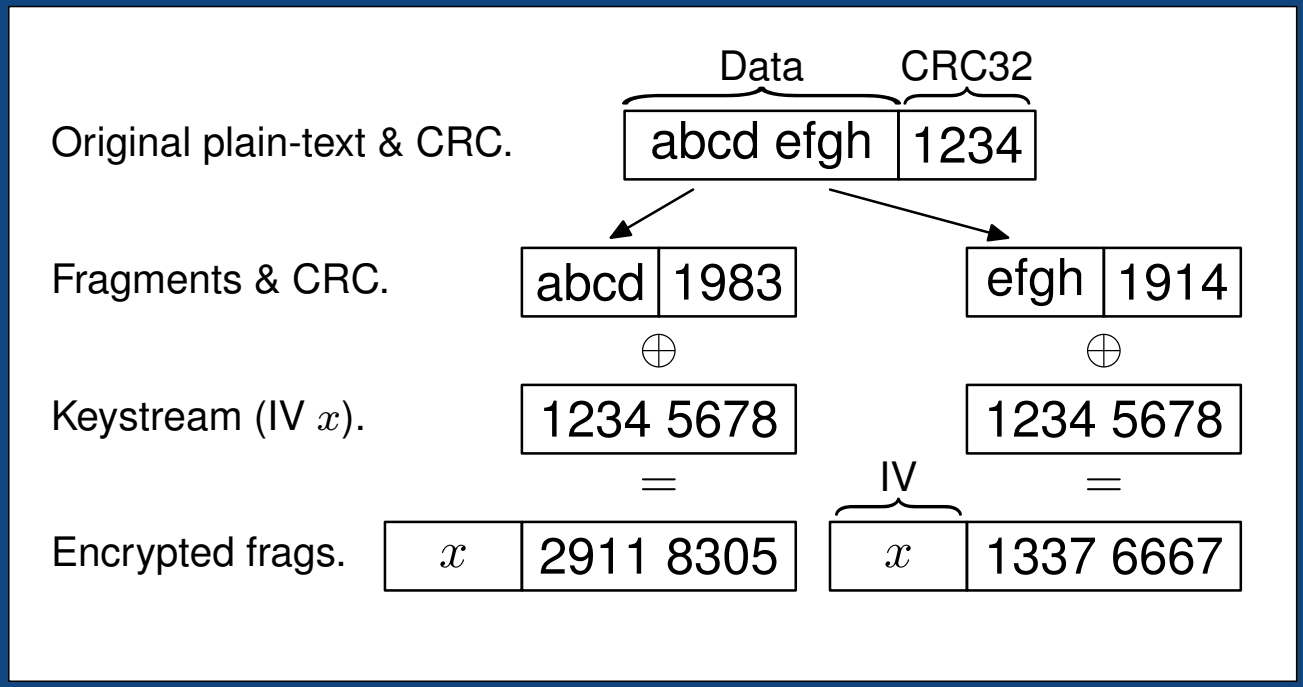
known plaintext in packets



LLC/SNAP header contained in practically all 802.11 data frames

- first 8 bytes of keystream can be calculated (clear text XOR cipher text)
- possible to send 8 bytes of encrypted payload

fragmentation in 802.11



transmitting single packet in multiple frames

- send up to 16 802.11 fragments, each using the same keystream
- inject 4x16 = 64 bytes of data

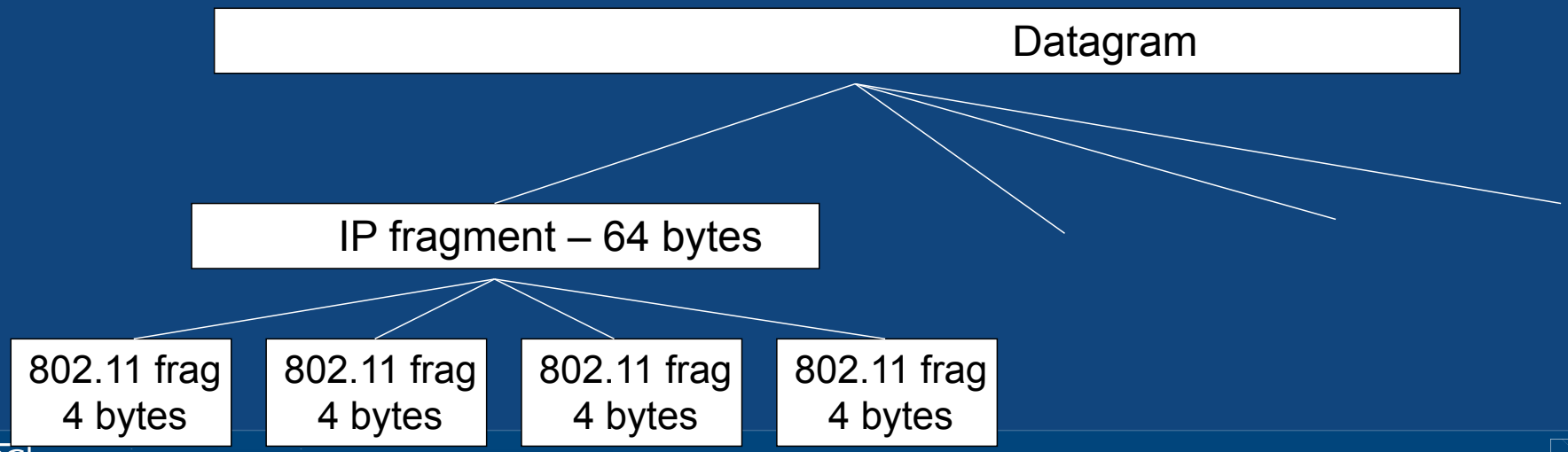
Pure Fragmentation Attack

- Transmission

Steps

1. eavesdrop one packet – get 8 bytes
2. send data up to 64 bytes

   use **IP fragmentation** to send larger payloads



Pure Fragmentation Attack

- Decryption (Forwarding to the Internet)

Steps

1. capture packet to decrypt
2. prepend additional IP header & forward to AP
3. AP assembles into single packet & decrypts
4. AP sends cleartext packet to destination host
5. Attacker recovers packet from controlled host

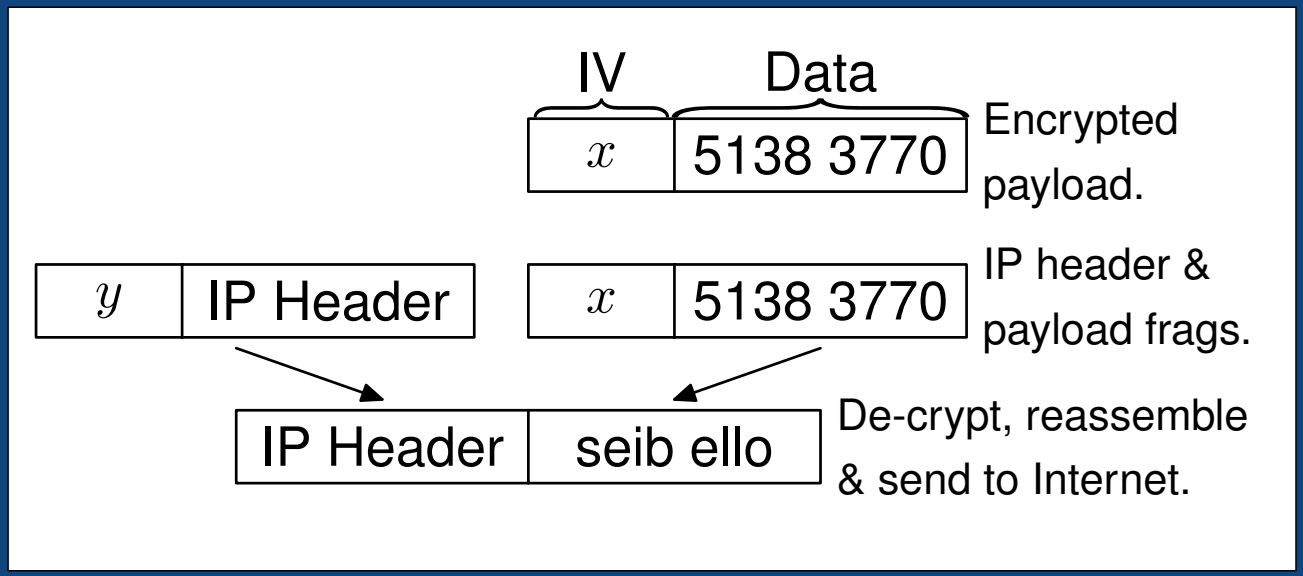
Wait, it's not quite that easy…

Pure Fragmentation Attack

Why is it not quite that easy?

- Problem 1: Need Router's MAC address and proper source address for network…to send.
- Problem 2: How about packets which meet the MTU limit?

Using the AP to decrypt & send along



Pure Fragmentation Attack

Problem 1: Need Router's MAC address and proper source address for network…to send.

- Need MAC address of router
  - Often AP is router (look for beacon frames)
  - Most popular MAC used
- Need correct source IP
  - In some networks this is not needed

Pure Fragmentation Attack

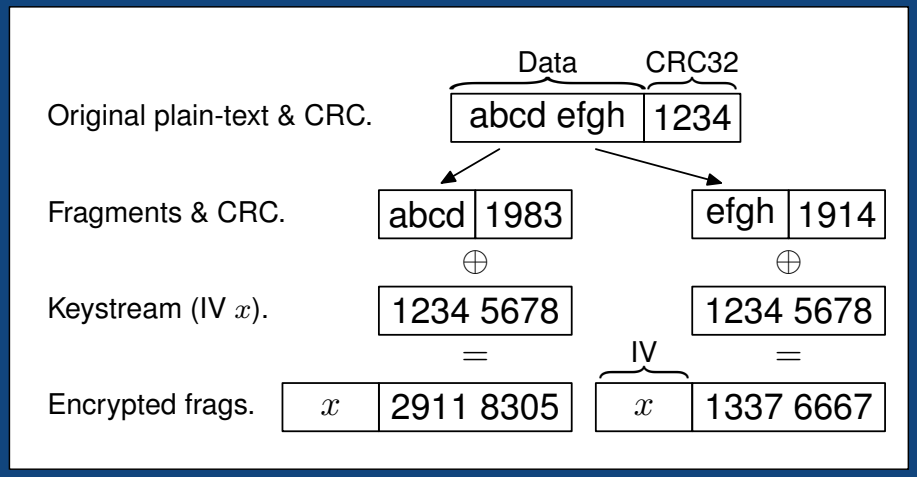
Problem 2: How about packets which meet the MTU limit?

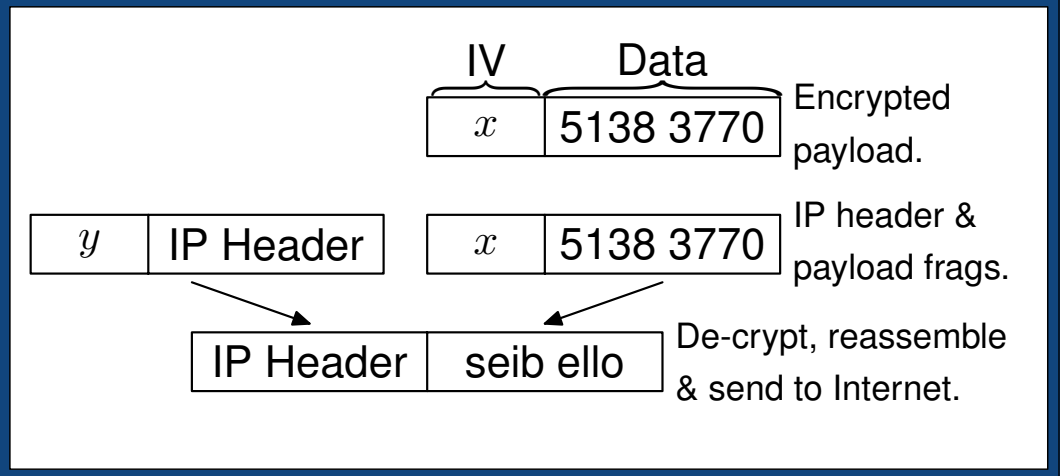
- MTU packets (if packet > MTU-28 bytes then trouble)
  - Techniques
    - Bit-flip destination address
    - Chop-chop
    - Spoof ICMP "packet too big" message

Pure Fragmentation requires access to internet controlled host – not possible with private network

Solution: **Keystream Based Attacks**

summary of new attacks so far





Keystream Attacks

ATTACKER's GOALS

1. Discover all possible keystreams (Dictionary attacks) - however the result is one long list of keystreams
2. Discover one specific keystream

Remember: if keystream known & packet snooped with corresponding IV then plaintext is now known.

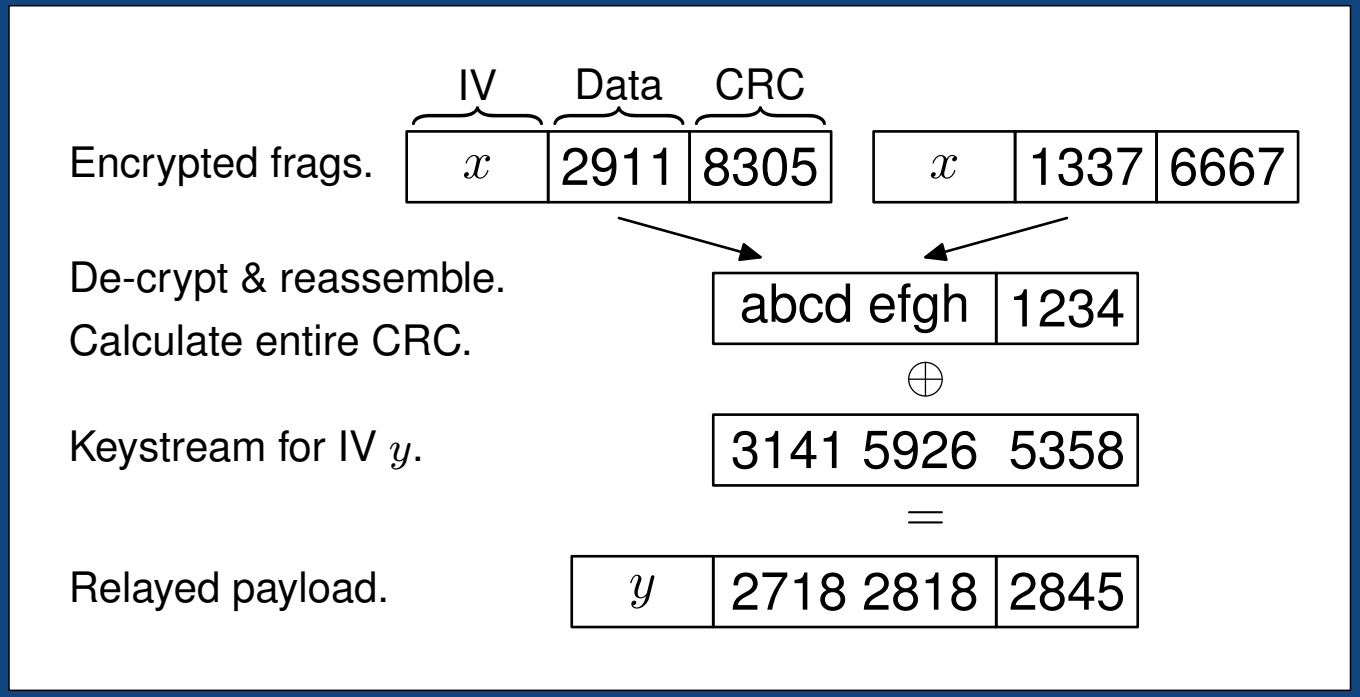
Keystream Attacks

- Discovering Keystreams

  Steps

  1. Acquire ability to send data (discussed earlier)
  2. Send large broadcast frame in small fragments
  3. AP will reassemble it and relay as large frame
  4. Attacker listens & obtains keystream for the new IV chosen by the AP
  5. Attacker does plaintext XOR to get new keystream for IV

Discovering Keystreams



34 fragments → 1500 bytes of keystream

16 frags (4 bytes/frag) → 64 bytes of keystream

16 frags (64 bytes/frag) → 1024 bytes

2 frags (1024 bytes/frag & 476 bytes/frag) → 1500 bytes

Discovering Keystreams

- To recover other keystreams – attacker sends 1500 bytes (without fragmentation) and snoops the relayed version by AP (most likely using a different IV)
- By sending approx. 16M (2^{24}) packets a complete IV dictionary is built.

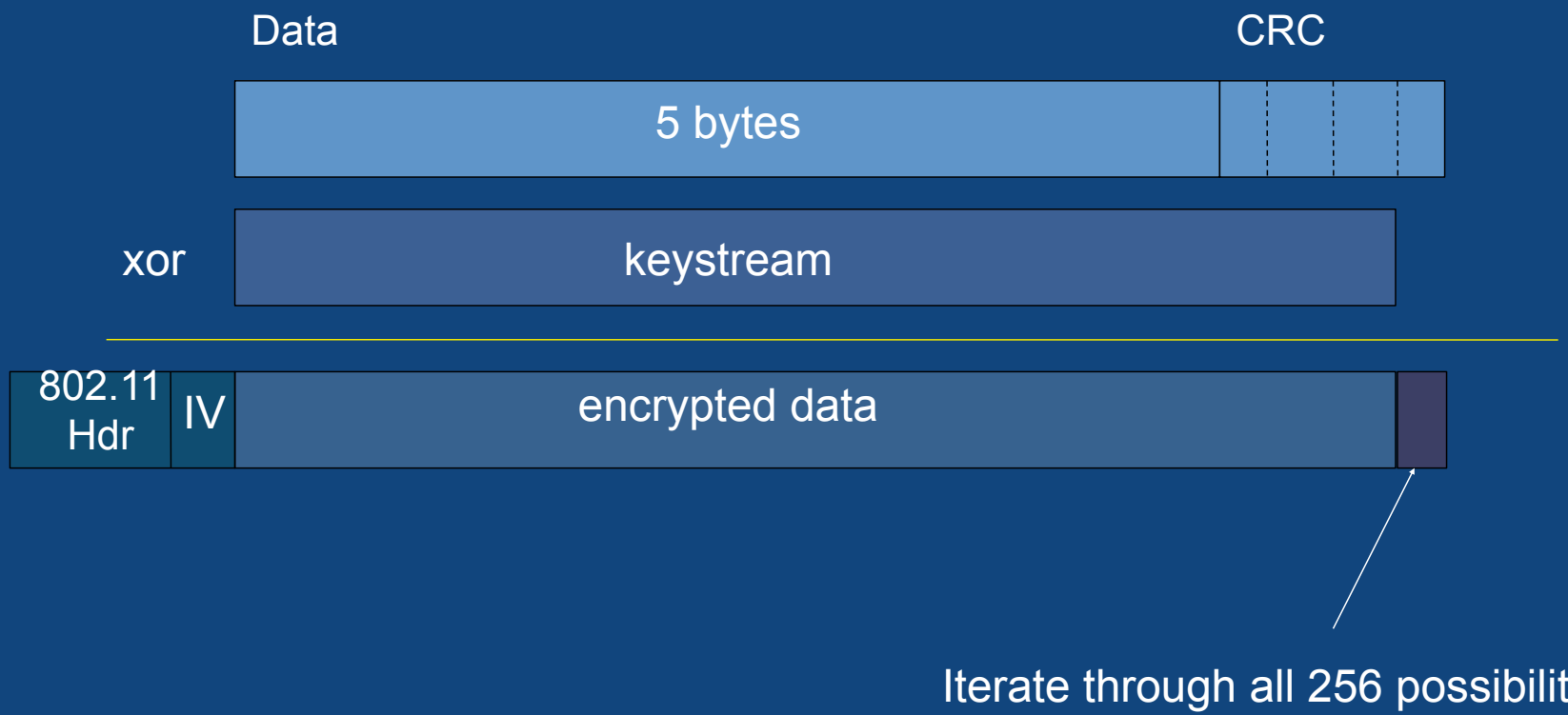
Discovering a Specific Keystream: Linear Keystream Expansion

- What happens if you need a specific keystream in order to decrypt a specific packet?

steps

1. recover keystream (8 bytes) via known plaintext
2. generate datagram of size 5 (8 – 3 bytes)
3. compute 4 byte CRC for payload (use only 3 bytes)
4. XOR with 8 byte pseudo-random stream
5. Append last byte (9th byte – guess)

Discovering a Specific Keystream: Linear Keystream Expansion



Discovering a Specific Keystream: Linear Keystream Expansion

steps (cont.)

6. send frame & wait for AP to broadcast
7. if no response from AP – try again with new last byte

   else

   we know that our last byte guess matches the last byte of the correct CRC (which we know) therefore we can calculate one more byte of the keystream.

next byte of keystream = byte guessed xor last known byte of CRC

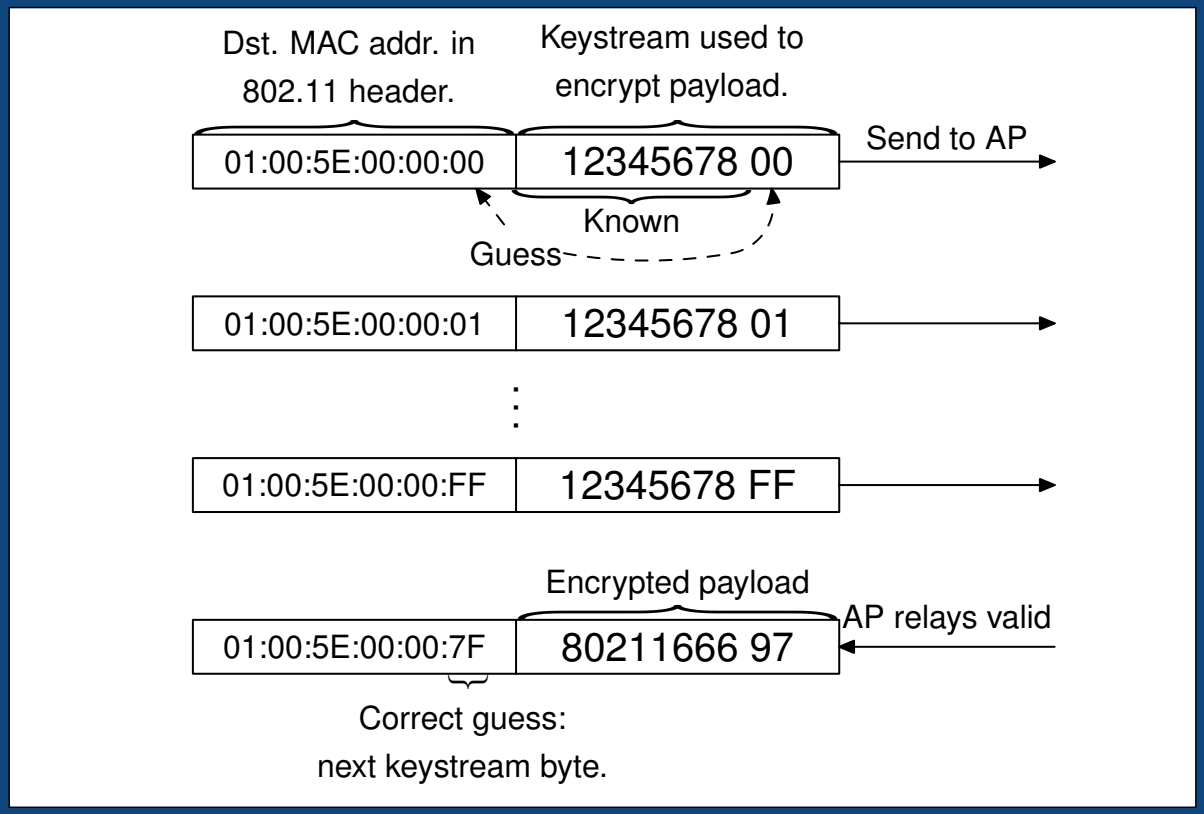
Keystream Attacks

Well-known multicast addresses

Instead of Timing the AP: Use multicast to do this –

All 256 guesses sent in parallel to 256 different multicast addresses

When AP relays one then simply read off the correct guess from the multicast address.



Therefore after sending 380,928 packets (most often less) – an arbitrary packet may be decrypted

summary of the new attacks

1. Eavesdrop a data packet
2. Recover 8 bytes of keystream
   - transmission of arbitrary data (up to 64 bytes) is possible via 802.11 fragmentation.
3. Recover 1,500 bytes of keystream by sending large broadcasts in smaller fragments
   - At this point, transmission of ar- bitrary data (of any length) is possible even without 802.11 fragmentation.
4. If an external communications channel is available
   - re-send an eavesdropped data packet to a controlled Internet host by using fragmentation.
   - The AP will decrypt the packet which is then received by the remote host, and returned to the attacker.
5. Otherwise, obtain the network's IP prefix by decrypting the IP address in a packet by using the linear keystream expansion technique.
6. Obtain the router's MAC address. This is only neces- sary if communication with the Internet is required
7. Decrypt "interesting" data.
8. Generate traffic in the network.
   1. Build an IV dictionary.
   2. Perform the weak IV attack.

Conclusion

Using Fragmentation:

- Takes less than a minute for an attacker to be able to send MTU-sized packets (and find out the IP address range of the network)
- About 15 minutes to recover 40-bit WEP keys
- About 60-120 minutes to recover 104-bit WEP keys

WEP is officially DEAD – fragmentation used in conjunction with these other attacks counters the final countermeasure: frequent re-keying

credits

these slides contain material from

1. presentation on “The Final Nail in WEP’s Coffin” at CPS372 in Gordon College
2. Bittau, A., Handley, M., Lackey, J., "The final nail in WEP's coffin," in IEEE Symposium on Security and Privacy, pp. 15, 21-24 May 2006.