Last Time

- Internet Application Security and Privacy
  - Authentication
  - Security controls using cryptography

This time

[Stinson, Shmatikov-Boneh]

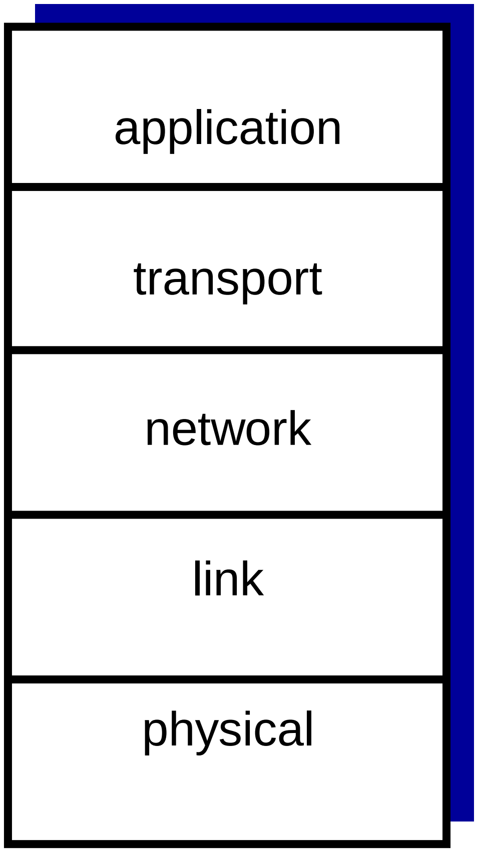
- Internet Application Security and Privacy
  - Link-layer security: WEP, WPA, WPA2

Network security and privacy

- The primary use for cryptography
  - “Separating the security of the medium from the security of the message”
- Entities you can only communicate with over a network are inherently less trustworthy
  - They may not be who they claim to be

Internet protocol stack

- *application:* supporting network applications
  - FTP, SMTP, HTTP
- *transport:* process-process data transfer
  - TCP, UDP
- *network:* routing of datagrams from source to destination
  - IP, routing protocols
- *link:* data transfer between neighboring network elements
  - Ethernet, 802.111 (WiFi)
- *physical:* bits "on the wire"



Network security and privacy

- Network cryptography is used at every layer of the network stack for both security and privacy applications:
  - Link
    - WEP, WPA, WPA2
  - Network
    - VPN, IPSec
  - Transport
    - TLS / SSL, Tor
  - Application
    - ssh, PGP, OTR

Link-layer security controls

- Intended to protect local area networks
- Most common example today: WEP (Wired Equivalent Privacy)
- WEP was intended to enforce three security goals:
  - Confidentiality
    - Prevent an adversary from learning the contents of your wireless traffic
  - Access Control
    - Prevent an adversary from using your wireless infrastructure
  - Data Integrity
- None of these is actually enforced!

WEP description

Brief description:

- The sender and receiver share a secret *k*
  - *The secret k is either 40 or 104 bits long*
- In order to transmit a message *M*:
  - Compute a checksum *c(M)*
    - this does not depend on *k*
  - Pick an IV (a random number) *v* and generate a keystream *RC4(v,k)*
  - XOR <*M,c(M)*> with the keystream to get the ciphertext
  - Transmit *v* and the ciphertext over the radio link

WEP description

- Upon receipt of *v and the ciphertext*:
  - Use the received v and the shared *k* to generate the keystream *RC4(v,k)*
  - XOR the ciphertext with *RC4(v,k)* to get <*M′,c′*>
  - Check to see if $c' = c(M')$
  - If it is, accept *M′* as the message transmitted

- Problem number 1: *v is 24 bits long*
  - *Why is this a problem?*
    - RC4 is a stream cipher: the same traffic key must never be used twice.
    - The purpose of an IV: to prevent any repetition
    - A 24-bit IV is not long enough to ensure this on a busy network. For a 24-bit IV, there is a 50% probability the same IV will repeat after 5000 packets.

WEP data integrity

- Problem 2: the checksum used in WEP is CRC-32 (cyclic redundancy check)
  - Quite a poor choice; there's already a CRC in the protocol to detect random errors, and a CRC can't help you protect against malicious errors.
- The CRC has two important properties:
  - It is independent of *k* and *v*
  - It is linear: $c(M \text{ XOR } D) = c(M) \text{ XOR } c(D)$
- *Why is linearity a pessimal property for your integrity mechanism to have when used in conjunction with a stream cipher?*

When CRC is encrypted with a stream cipher, both the message and the associated CRC can be manipulated without knowledge of the encryption key;

WEP access control

- What if the adversary wants to inject a new message *F* onto a WEP-protected network?

- All he needs is a single plaintext/ciphertext pair
- This gives him a value of *v* and the corresponding keystream *RC4(v,k)*
- Then $C' = \langle F, c(F) \rangle$ XOR $RC4(v,k)$, and he transmits v, C'.
- C' is in fact a correct encryption of *F*, so the message must be accepted.

WEP authentication protocol

- How did we get that single plaintext/ciphertext pair we needed just now?
  - Problem 3: It turns out the authentication protocol gives it to the adversary for free!
- The authentication protocol is supposed to prove that a certain client knows the shared secret *k*
  - Four step challenge-response handshake
- But if I watch you prove it, I can turn around and execute the protocol myself!
  - “What’s the password?”

WEP authentication protocol

- Here's the protocol:
  - The access point sends a challenge string to the client
  - The client sends back the challenge, WEP-encrypted with the shared secret *k*
  - The base station checks if the challenge is correctly encrypted, and if so, accepts the client
- So the adversary has just seen both the plaintext and the ciphertext of the challenge

WEP decryption

- Problem number 4: this is enough not only to inject packets (as in the previous attack), but also to execute the authentication protocol himself!

- Somewhat surprisingly, the ability to modify and inject packets also leads to ways to adversary can decrypt packets!
  - The access point knows k; it turns out the adversary can trick it into decrypting the packet for him and telling him the result.

Recovering a WEP key

- Note that none of the attacks so far:
  - Used the fact that the stream cipher was RC4 specifically
  - Recovered *k*
- Since 2002, there have been a series of analyses of RC4 in particular
  - Problem number 5: it turns out that when RC4 is used with similar keys, the output keystream has a subtle weakness
    - And this is how WEP uses RC4!

Replacing WEP

- These observations have led to programs that can recover either a 104-bit or 40-bit WEP key in under 60 seconds, most of the time

- Wi-fi Protected Access (WPA) was rolled out as a short-term patch to WEP while formal standards for a replacement protocol (IEEE 802.11i, later called WPA2) were being developed

Replacing WEP

- WPA:
  - Replaces CRC-32 with a real MAC (here called a MIC to avoid confusion with a Media Access Control address)
  - IV is 48 bits
  - Key is changed frequently
  - Ability to use 802.11x authentication server
    - But maintains less-secure PSK (Pre-Shared Key) mode for home users
  - Able to run on most older WEP hardware

Replacing WEP

- The 802.11i standard was finalized in 2004, and the result (called WPA2) has been required for products calling themselves “Wi-fi” since 2006
- WPA2:
  - Replaces the RC4 and MIC algorithms in WPA with the CCMP algorithm, which uses AES
  - Considered strong, except in PSK mode
    - Dictionary attacks still possible

Network-layer security

- Suppose every link in our network had strong link-layer security
- Why would this not be enough?

- We need security across networks
  - Ideally, end-to-end

- At the network layer, this is usually accomplished with a Virtual Private Network (VPN)

Recap

- Internet Application Security and Privacy
  - Link-layer security: WEP, WPA, WPA2

Next time

- Internet Application Security and Privacy
  - Network-layer security: VPN, IPSec
  - Transport-layer security and privacy: TLS / SSL, Tor