ELEC5616 COMPUTER & NETWORK SECURITY

Lecture 20:
Wireless Security

MOBILE COMPUTING

Main forms:

3G, 4G, 5G Mobile (CDMA2000, LTE, Samsung)
802.11 Wireless Ethernet (Wireless LANs)
802.15 Wireless Personal Area Networks (e.g. Bluetooth)
802.16 Wireless Broadband

Of main concern to wireless networking is 802.11

802.11b operating at 2.4GHz ISM band (11Mbps)
802.11a operating at 5GHz ISM band (54Mbps)
802.11g Mixed mode operation (a & b)

802.11c Bridging.
802.11f Roaming, Access Point (AP) Hand Off.
802.11i Security / WPA2
.. MIMO etc..

PARADIGM CHANGE

Physical access to the network is no longer required

Most wireless networks are inside the firewall

No more network perimeter

Most wireless networks link to insecure machines

Particularly laptops, soon PDAs and mobile phones

Passive and active attacks are easier to launch

Less audit trails

Less security mechanisms (for now)

Attackers can get away with relative impunity

Denial of service

WAR DRIVING

The wireless equivalent of

- war dialing
  - scanning all carriers within an area code with a modem
- port scanning
  - scanning all machines and ports on a network

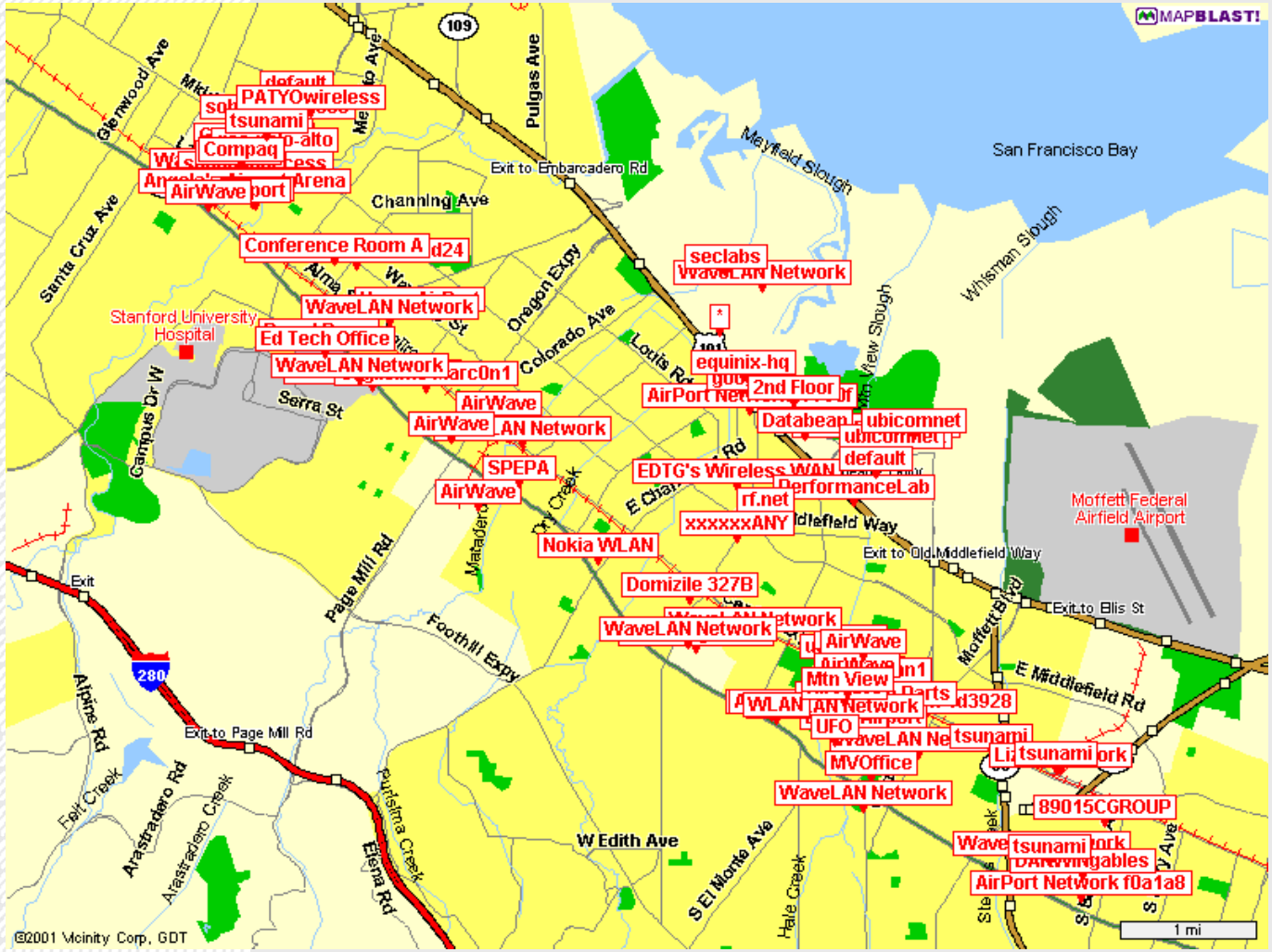
The concept is simple:

- Drive around in a car listening for 802.11 networks.
- Plot signal strengths on a map using a hand held GPS unit.

Tools

- Net stumbler
- Airsnort
- WEPcrack
- Antenna (21dB directional $200)
- Amplifier (up to 10W over the Internet ~$1,000)
- Laptop (war driving)
- Palmtop (war walking)

WAR DRIVING


MAPBLAST!
San Francisco Bay
Mayfield Slough
Whisman Slough
Pulgas Ave
Exit to Embarcadero Rd
Channing Ave
Santa Cruz Ave
Oregon Expy
Colorado Ave
Stanford University Hospital
Serra St
Page Mill Rd
Foothill Expy
Exit to Page Mill Rd
Alpine Rd
Arastradero Rd
Moffett Federal Airfield Airport
Exit to Old Middlefield Way
Exit to Ellis St
E Middlefield Rd
W Edith Ave
Hale Creek
1 mi
©2001 Vicinity Corp. GDT
default
PATYOwireless
tsunami
Compaq
AirWave
Conference Room A
WaveLAN Network
Ed Tech Office
SPEPA
Nokia WLAN
Domizile 327B
seclabs
equinix-hq
2nd Floor
ubicomnet
default
xxxxxxANY
Mtn View
UFO
MVOffice
89015CGROUP
AirPort Network f0a1a8

802.11B

802.11b is protected by the Wired Equivalent Privacy (WEP) protocol.

- Claimed to be "equivalent security" to a fixed wired network but in fact is much worse.

WEP Security Goals:

- Confidentiality
  - Prevent an attacker from eavesdropping
- Access Control
  - Prevent an attacker from accessing your network
- Integrity
  - Prevent an attacker modifying messages in transit

The following is an exercise in how security protocols should not be designed.

WEP OVERVIEW

A master key k_0 (either 40 or 104 bits) is shared between two parties wishing to communicate a priori.

Each 802.11 packet (header|data) is then protected by:

An integrity check field IC = h(header|data)

A random initialisation vector IV

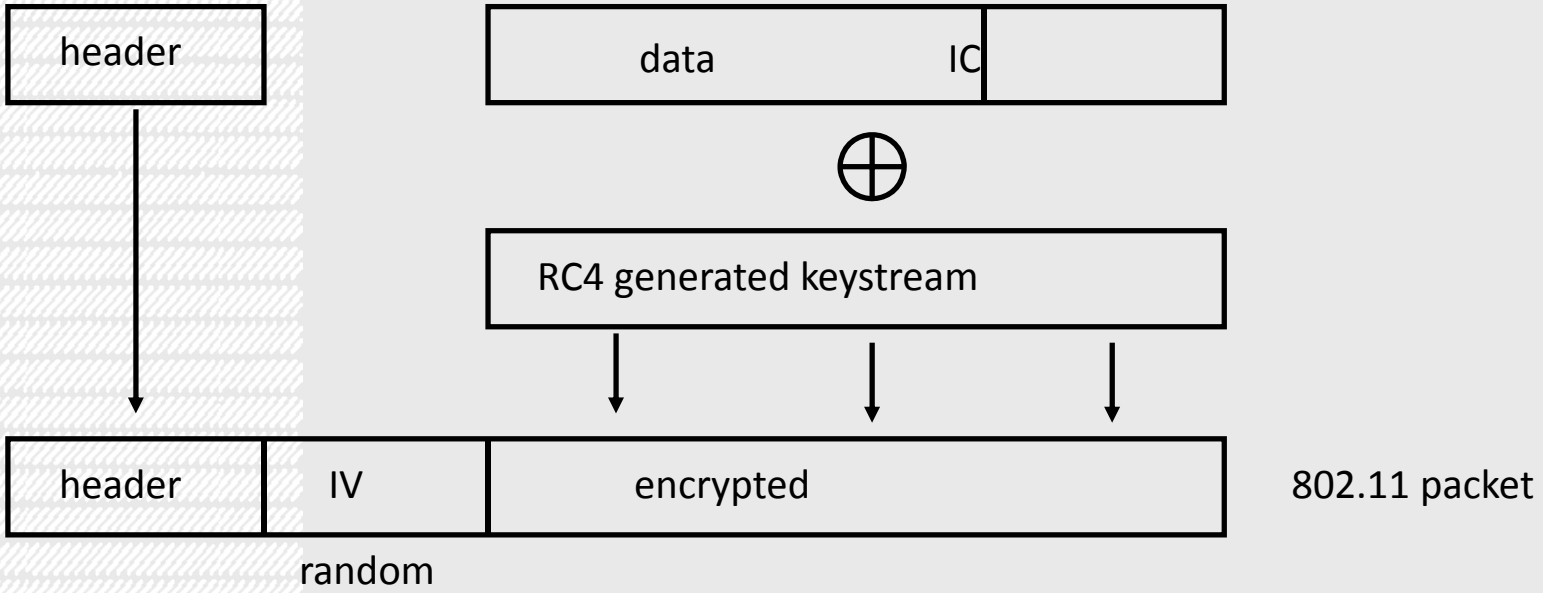
The master key and IV are used to generate a keystream using RC4 in stream cypher mode

$$k = RC4(k_0, IV)$$

The data and IC are then encrypted by this keystream

$$E_k(m) = m \oplus k$$

WEP PACKET



packet = header | IV | E_k(data | IC)

RC4 STREAM CYPHER

WEP protects the confidentiality of the payload through RC4 in stream cypher mode.

Senders use RC4 seeded with the IV and master key k_0 to generate a keystream. This keystream is then xord with the plaintext.

Receivers likewise generate the same keystream using the master key (shared a priori) and the received IV (sent in the clear). They then xor this with the cyphertext to obtain the plaintext (the keys cancel):

$$m = c \oplus k = m \oplus k \oplus k$$

ATTACKS ON WEP

WEP is broken.

There are a surprising large number of attacks possible on the protocol:

- Passive attacks to decrypt traffic based on statistical analysis.
- Active attacks to decrypt traffic, based on tricking the access point.
- Active attack to inject new traffic from unauthorized mobile stations.
- A memory tradeoff attack that allows real-time automated decryption of all traffic.
- An active inductive chosen plaintext attack which allows decryption of traffic.
- An attack on the key scheduling algorithm of RC4.

STREAM CYPHER PROBLEMS

RC4 is effectively being used as a pseudo-one time pad.

<u>Problem:</u>

Two messages must never be sent using the same key or you end up with a two time pad:

$$c_1 \oplus c_2 = m_1 \oplus k \oplus m_2 \oplus k$$
$$= m_1 \oplus m_2$$

This is effectively a running key cipher with English as the key.

As the messages have a low entropy (parts are very easily guessed), an attacker can trivially decode both messages.

Even worse, an attacker can obtain the original keystream.

STREAM CYPHER PROBLEMS

They keystream in this mode of RC4 depends on only an IV and k_0.

The master key k_0 is a long-term, fixed key

In many setups all users share this key (so much for WEP at a "hot spot")

As it is user chosen it is most likely guessable (dictionary attack).

Thus the keystream is only really dependent on IV

Which is 24 bits long (16 million values)

If any two packets ever have the same IV, the keystream is reused (hence packets can be decrypted).

The IV is transmitted in the clear, making it simple for an attacker to know when a collision occurs.

BIRTHDAY ATTACK ON THE IV

To attack the IV in WEP, any packet collision will do.

According to the birthday paradox, if C(N,q) is the probability of collision throwing q balls randomly into N different buckets then if also $1 \leq q \leq \sqrt{(2N)}$ we know:

$$C(N,q) \geq 0.3\ q(q-1)/N$$

Solving for C(N,q) = 0.5 and $N = 2^{24}$ gives

$$q = 5{,}288 \text{ packets}$$

Thus on average a collision will occur every 5,288 packets.

IV IMPLEMENTATION IS BROKEN

In reality, the problem is much worse. Most cards initialise the IV as zero on power on and increment per packet sent rather than use random values.

Finding a collision becomes trivial as they will occur every time a laptop is powered on.

Furthermore, in most arrangements the master key k_0 is shared between all users on the network.

Thus an attacker can find collisions between any user on the network

Any direction of all users on all channels.

A MEMORY TRADEOFF ATTACK ON THE IV

An adversary can mount a known plaintext attack on the IV in WEP easily:

Send a WEP user a known message (e.g. via email)

The adversary records the IV for the message

They then XOR the plaintext and the cyphertext to store the keystream

This keystream is stored in a table, indexed by the IV value

Next time a message is sent with that IV, the message can be fully decrypted.

Likewise an adversary can mount this attack with no known plaintext if they see a packet collision (thus can decrypt the third packet sent).

REFINING THE IV MEMORY TRADEOFF ATTACK

A full table for all IVs for a given master key k_0 will take at most 1,500 bytes * 2^{24} = 24GB (a cheap hard drive).

Most likely one won't need the full 1,500 bytes (500 may do).

Note the table is independent of the size of the master key k_0.

If the cards are using non random IVs (e.g. initialised to zero), then the IVs (and hence the tables) will be much smaller, making the attack much easier.

Furthermore the 802.11 standard dictates that changing the IV with each packet is optional!

THE INTEGRITY CHECK FIELD

In WEP, the Integrity Check field (IC) is a 4 byte value used to verify message integrity (and, in fact message authentication).

Thus a receiver will accept a message if the IC is valid.

The issue with WEP is that the IC is the CRC-32 cyclic redundancy check, a simple checksum.

- CRCs are good for detecting transmission errors
- CRCs do nothing to stop malicious errors

There are two major problems here

- CRCs are linear i.e. $h(m \oplus k) = h(m) \oplus h(k)$
- The CRC is independent of the master secret k_0 and the IV

MODIFICATION ATTACK ON THE IC

The attacker records a message (known or not known)

The attacker then modifies m in a known way to produce m'

$$m' = m \oplus \Delta$$

Since CRC-32 is linear, they can compute a new valid integrity check field IC :

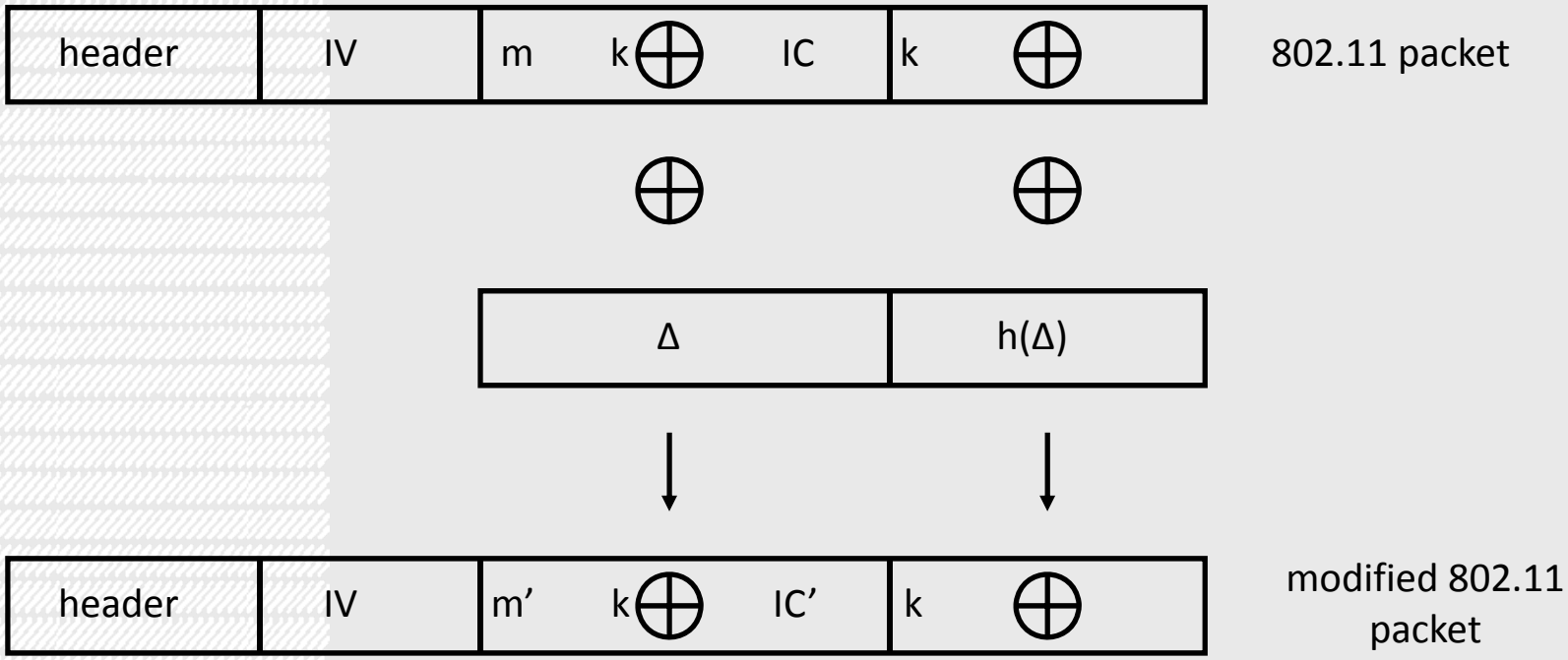
$$IC' = IC \oplus h(\Delta)$$

Which will be valid for the new cyphertext c'

$$c' = c \oplus \Delta = k \oplus (m \oplus \Delta) = k \oplus m'$$

Thus an attacker xors the original packet by (Δ | h(Δ))

WEP PACKET


header
IV
m
k
IC
k
802.11 packet
Δ
h(Δ)
header
IV
m'
k
IC'
k
modified 802.11 packet

KEYSTREAM RECOVERY ATTACK

If an attacker knows the plaintext of a single WEP protected packet, they can inject any packet into the network

An attacker records a packet $c = m \oplus k$ where m is known

e.g. the attacker emails the victim

The attacker then recovers the keystream $k = c \oplus m$ for that IV

Say an attacker wishes to inject message m'. They compute:

$$IC' = h(m') = CRC32(m')$$

The attacker then computes the encrypted part of the packet

$$c = (m'|IC') \oplus k$$

The attacker now has a valid packet

$$\text{header} \mid IV \mid (m' \mid IC') \oplus k$$

KEYSTREAM RECOVERY ATTACK

The fundamental problem here is that they checksum is not dependent on any shared secret.

As a result, if CRC-32 is replaced by a secure hash function (e.g. MD5) this attack would still be possible.

Far better would have been to use a keyed MAC dependent on some secret.

ATTACK ON THE AUTHENTICATION PROTOCOL

The authentication protocol in WEP is used to prove that a client wishing to access the network knows master secret k_0

The base station sends a challenge [x | h(x)] to the client.

The client sends back the challenge encrypted with k_0

$$[x \mid h(x)] \oplus k \qquad \text{where } k = RC4(IV, k_0)$$

The base station verifies the response is encrypted with k_0.

Problem:

An eavesdropper has just seen a plaintext/cyphertext pair (and hence can use it in any of the attacks mentioned before - including extracting the keystream).

An eavesdropper can replay the response to gain access to the network, spoofing the authentication protocol.

AUTHENTICATION SPOOFING

Alice tries connecting to the network.

Bob (the base station) sends out a challenge [x | h(x)].

Alice replies with [IV, (x | h(x)) $\oplus$ k].

Eve extracts IV and k from this message by xoring the challenge with the response.

Now Eve tries connecting to the network.

Bob sends out a challenge string y.

Eve replies with [IV, (y | h(y)) $\oplus$ k].

Bob accepts Eve onto the network.

MESSAGE DECRYPTION ATTACKS

Although an adversary does not know k_0 through any of the attacks so far, there are several attacks in which they can trick the base station to decrypt messages for them:

Decryption by double encryption.

WEP decapsulation through message redirection.

Reaction attacks.

DOUBLE ENCRYPTION

An attacker records a packet they wish to decrypt. Say this packet has the value IV = v as the initialisation vector.

The attacker waits until the base station resets (or wraps) and the base station IV = v-1.

The attacker then forwards this packet over a separate connection through the base station (joined through authentication spoofing).

The base station will encrypt the encrypted packet:

$$[m \mid h(m)] \oplus RC4(v, k_0) \oplus RC4(v, k_0) = [m \mid h(m)]$$

The plaintext is thus sent over the air.

MESSAGE REDIRECTION

This attack is even easier than double encryption in that it removes timing issues.

An attacker records a packet they wish to decrypt.

They then modify the header so that the destination IP address is a machine they control somewhere on the Internet.

The attacker then calculates a new IC checksum:

Remember if $m' = m \oplus \Delta$, then $IC' = IC \oplus h(\Delta)$ (CRC-32 is linear)

The attacker then joins the network using authentication spoofing.

The attacker then injects this packet onto the network.

The base station will forward the packet to the Internet, stripping the WEP encapsulation (decrypting it).

REACTION ATTACKS

This attack allows an adversary to decrypt a packet even if the base station is not connected to the Internet.

The target packet to decrypt needs to be a TCP packet (though others can likely be sent as TCP packets).

<u>Lemma:</u> It is possible using the TCP checksum to make the checksum be valid or invalid depending on whether a particular bit in the message is a 0 or 1.

An attacker modifies the recorded packet to check if bit0 of the message is a 0 and sends it on the network.

- If the base station responds with an ACK, bit0 is 0.
- If the base station responds with a NACK, bit0 is 1.

The adversary repeats for each bit in the message.

INDUCTIVE CHOSEN PLAINTEXT ATTACK

Principle:

Guess at some plaintext in an encrypted message.

Based on this we know n bytes of the keystream.

Leverage redundancy in the CRC-32 checksum to learn more information (one byte at a time) about the keystream.

INDUCTIVE CHOSEN PLAINTEXT ATTACK

Example:

Wait for a DHCP discover message (where we know the source address is 0.0.0.0 and the destination address is 255.255.255.255).

We now have 24 bytes of keystream for a particular IV (if we xor the known plaintext with the cyphertext we get the keystream).

Create a new packet now (say a "ping" packet) that is 24 - 3 = 21 bytes long. Xor this part with the first 21 bytes of the keystream we know.

INDUCTIVE CHOSEN PLAINTEXT ATTACK

Example:

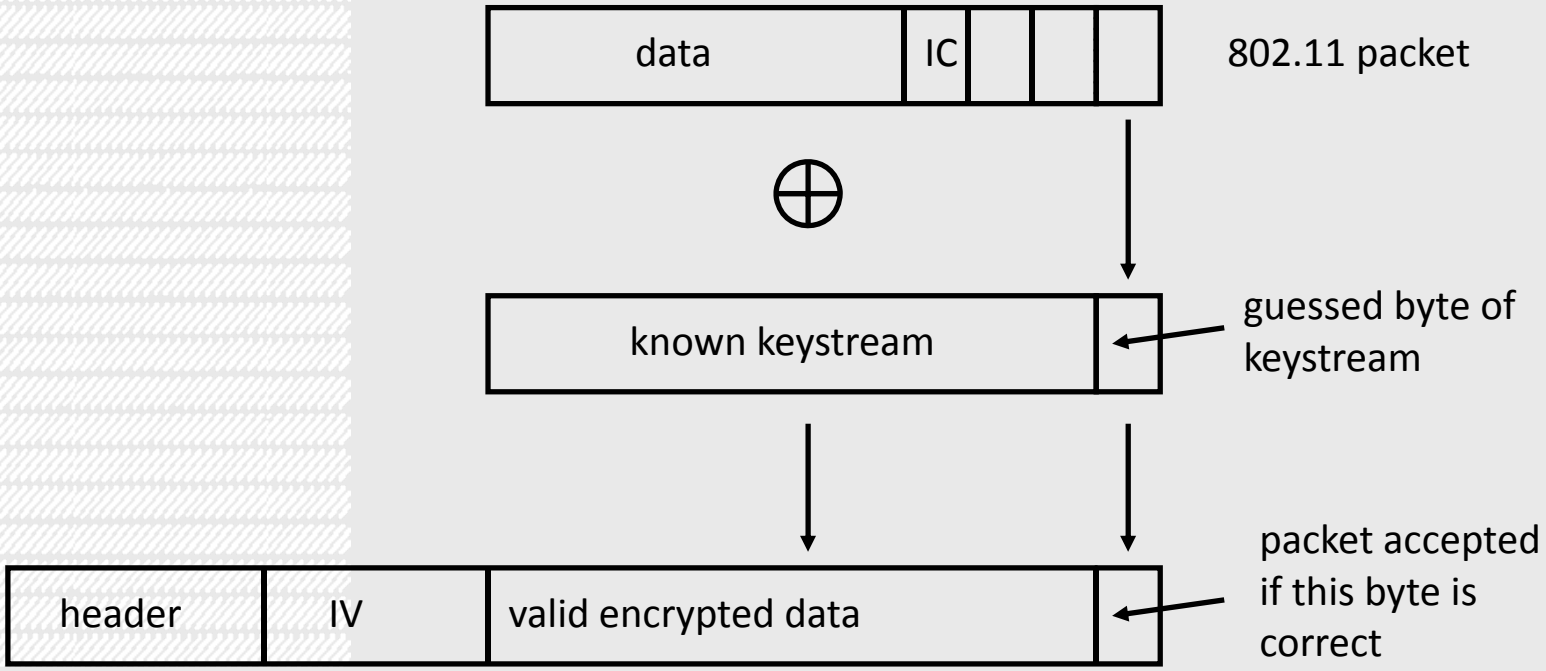
Compute the checksum IC for the message, but only append the first 3 of 4 bytes to the packet. Xor this with the remaining bytes of the keystream we know.

Add the last byte of the checksum and guess at the next byte of keystream to xor.

If the packet is accepted we got it right (repeat 256 times until we get it correct).

When we get it right we learn one more byte of the keystream (for a given IV).

INDUCTIVE CHOSEN PLAINTEXT ATTACK



INDUCTIVE CHOSEN PLAINTEXT ATTACK

Discussion:

This attack is possible regardless of the length of the IV or the key size.

This attack is stopped by use of a keyed MAC for the hash function (again, instead of CRC-32).

Replay prevention would also help.

An attacker making 100 guesses/second will, on average obtain a 1,500 byte keystream (for a given IV) in 32 minutes.

Note: failures are not logged by the OS (hence attackers are not noticed).

IV CASCADING

Once an attacker has one IV, the others are trivial to obtain.

An adversary needs only transmit a packet which is echoed back by the access point (e.g. a ping packet).

The access point will pick a new IV to encrypt the known plaintext. Hence an attacker can quickly fill the remaining values from the 2^{24} possible combinations.

Broadcast pings are even better, returning many packets for each one sent.

THE KEY SCHEDULING ALGORITHM IN RC4

After all this, RC4 is used poorly in the protocol.

There are large numbers of "weak" keys where a few bits in the key leads to large numbers of determined bits out of the key scheduling algorithm (KSA) and output stream.

Combined with this is a related key attack which allows an adversary to obtain the rest of the secret bits when they have access to parts of the input key to RC4. In WEP they can modify the IV; remember the stream cypher is RC4(IV, k_0).

This attack is only linear in complexity with increasing key size. Hence 128-bit WEP2 keys are also vulnerable.

PROBLEMS WITH 802.11

Significant problems (you should have picked up from this class)

- The IC hash should be a keyed MAC, not a linear checksum.
- 24 bit initialisation vectors are too small, and should be randomly chosen.
- The master secret k_0 is likewise too small (at 40 bits) and should be arranged to be different for each machine - and not user chosen.
- The key scheduling algorithm of RC4 is broken. The cypher should be replaced with another (many alternatives).
- Nonces should be incorporated to avoid replay issues.
- The authentication protocol is weak and keys used should be separate from those used to protect confidentiality.
- New versions should not allow backwards compatibility!

Other major problems:

- The underlying 802.11b management frames are unauthenticated and may be spoofed
  - Whole slew of problems (AIR-JACK, WLAN-JACK, MONKEY-JACK, KRACKER-JACK)…

WEP SECURITY REALITY

Confidentiality

Your network is vulnerable from 10 kilometres away.

All your traffic can easily be decrypted.

Access Control

Anyone can join your network whenever they feel like it.

Most likely your internal network.

Integrity

All your traffic is vulnerable to modification and replay.

I own your DHCP server- all traffic now routes via my laptop

Reliability

Your network can be taken down at a moment's notice.

WIFI PROTECTED ACCESS (WPA/WPA2)

Temporal Key Integrity Protocol (TKIP)

• Per-packet 128-bit dynamic key

Replaces CRC-32 with an integrity check algorithm "Michael"

http://arstechnica.com/security/2008/11/wpa-cracked/

• Sniff a packet, make modifications to the checksum and send back to the AP, allows keystream recovery allowing decryption of small packets.. which potentially leads to ARP or DNS poisoning. Required Quality of Service (802.11e) to be enabled initially.

Further refinements of the attack have lead to full recovery without QoS.

WPA2 replaces "Michael" with CCMP, which is AES-based security.

Extensible Authentication Protocols (EAP) to WPA/WPA2
802.1X Authentication Server

802.1X

Standard for passing EAP over wired/wireless LAN

- EAP encapsulation over LANS (EAPOL)
- "Network Port Authentication"

Extensible Authentication Protocol (EAP)

- General framework for many authentication schemes
  - Passwords, challenge-response tokens, public-key infrastructure certificates ..

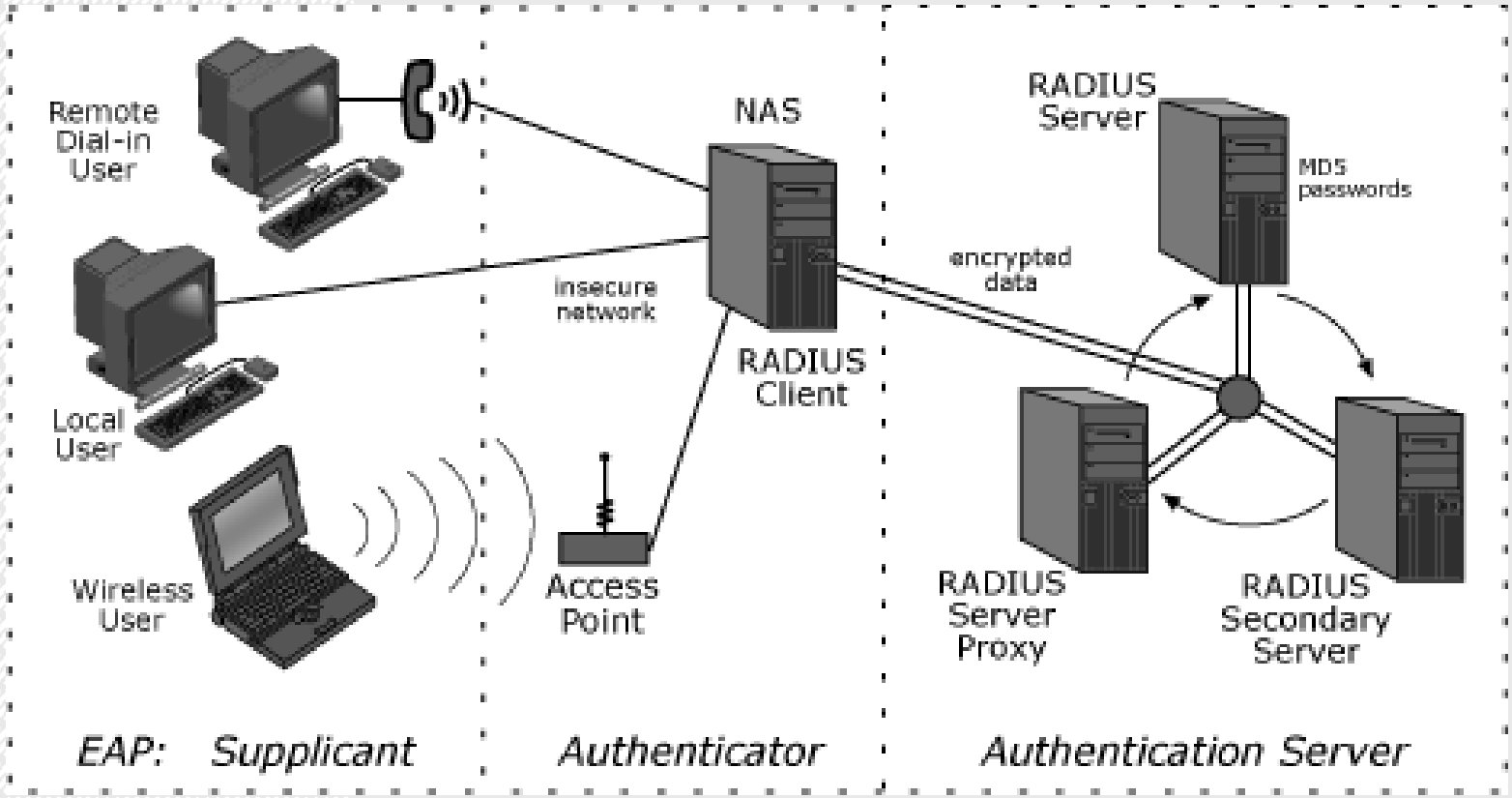
No per-packet overhead

Requires only firmware update

Fits well with existing infrastructure

- EAP originally designed as part of PPP authentication

802.1X


Remote
Dial-in
User
Local
User
Wireless
User
insecure
network
NAS
RADIUS
Client
Access
Point
RADIUS
Server
MD5
passwords
encrypted
data
RADIUS
Server
Proxy
RADIUS
Secondary
Server
EAP: Supplicant
Authenticator
Authentication Server

802.1X

1. Authenticator sends an "EAP-Request/Identity" packet to the supplicant as soon as it detects that the link is active
2. Supplicant sends an "EAP-Response/Identity" packet to the authenticator, which is then passed on to the authentication (RADIUS) server.
3. The authentication server sends back a challenge to the supplicant via the authenticator using EAPOL
4. The supplicant responds to the challenge via the authenticator and passes the response onto the authentication server.
5. If the supplicant provides proper identity, the authentication server responds with a success message, which is then passed onto the supplicant.
6. The authenticator now allows access to the LAN- - possibly restricted based on attributes that came back from the authentication server.
   - For example, the authenticator might switch the supplicant to a particular virtual LAN or install a set of firewall rules.

802.1X PROBLEMS

Is not a complete replacement for WEP

Confidentiality is not provided for, only key negotiation and management

Poor authentication protocols are vulnerable to attack

e.g. dictionary attacks on password authentication

Session Hijacking

After authentication, force supplicant to disconnect and steal session

Man in the middle

There is no mutual authentication, thus access points can be spoofed

802.1x authentication mechanisms are vendor-implemented

Variety of denial-of-service attacks

Sending spoofed EAPOL Start, Identifier, Success and Failure packets

OTHER

Hide SSID (security through obscurity)

MAC Filtering, IP Filtering

Wireless IDS

Monitor suspicious activity on the network

RF Signal Shaping

Directional antennae

Low access point power