Wireless Privacy: Analysis of 802.11 Security

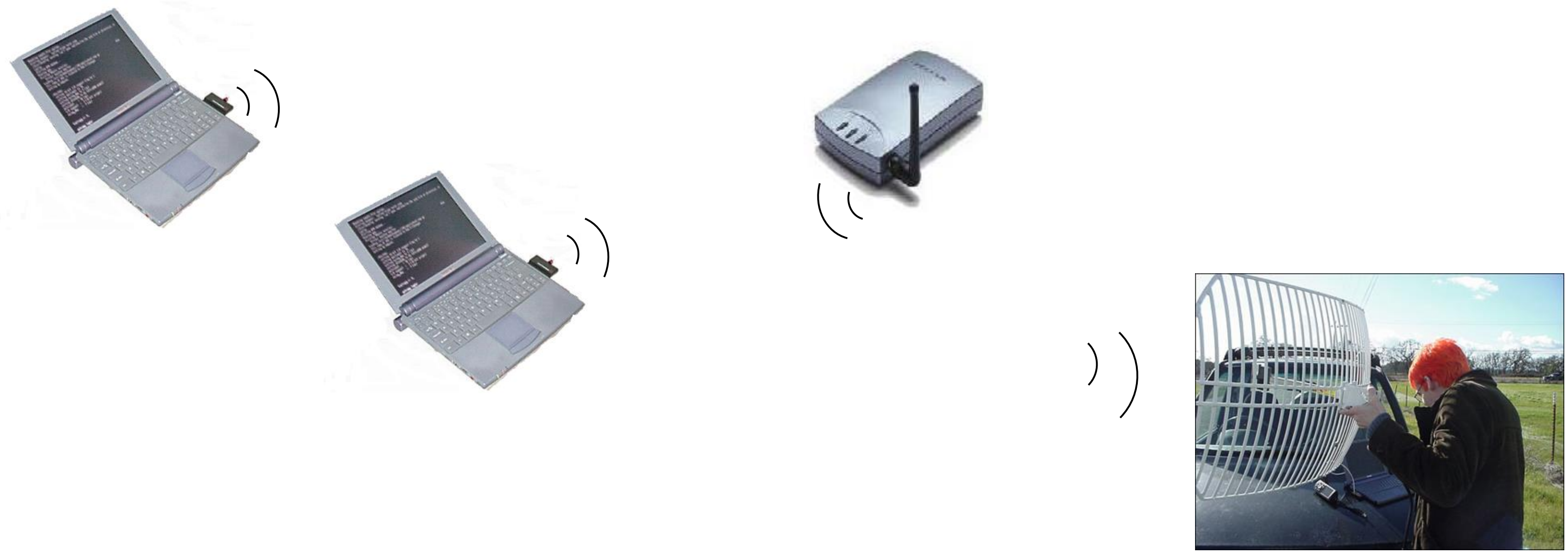
Nikita Borisov
UC Berkeley
nikitab@cs.berkeley.edu

Wireless Networking is Here

802.11 wireless networking is on the rise

- installed base: ~ 15 million users
- currently a $1 billion/year industry

The Problem: Security



Wireless networking is just radio communications

- Hence anyone with a radio can eavesdrop, inject traffic

Wireless Security

- Wireless networks becoming prevalent
- New security concerns
  - More attack opportunities
    - No need for physical access
  - Attack from a distance
    - 1km or more with good antennae
  - No physical evidence of attack
- Typical LAN protection insufficient
  - Need stronger technological measures

More Motivation

Wireless LANs: Trouble in the Air

By Bob Brewin, Dan Verton and Jennifer DiSabatino

(Jan. 14, 2002) As the airline industry scrambles to meet a Jan. 18 deadline to screen every checked bag for explosives, security experts, analysts and government officials are raising serious concerns about the security of wireless technology that's integral to the effort.

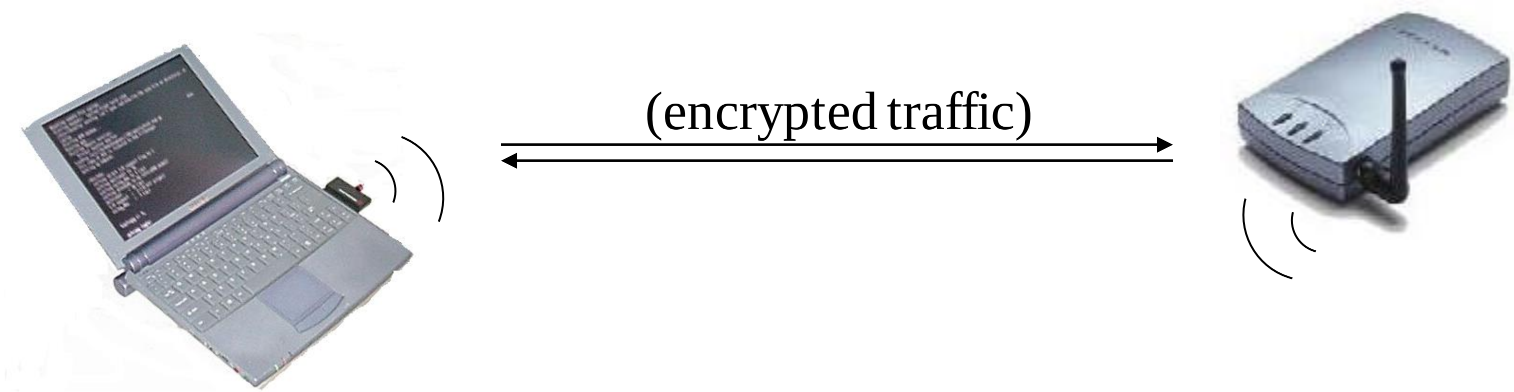
At issue is the adoption by airlines of industry-standard 802.11b, or Wi-Fi, wireless LANs operating in the 2.4-GHz band. These systems, which are widely viewed as inherently insecure, are being used to support such applications as bag matching and curbside and roving-agent check-in.

The concerns appear to be justified, based on two investigations that were conducted last week by professional security firms that analyzed airline wireless LAN systems at Denver International Airport and San Jose International Airport.

Overview of the Talk

- In this talk:
  - The history: WEP, and its (in)security
  - Where we stand today
  - Future directions

WEP

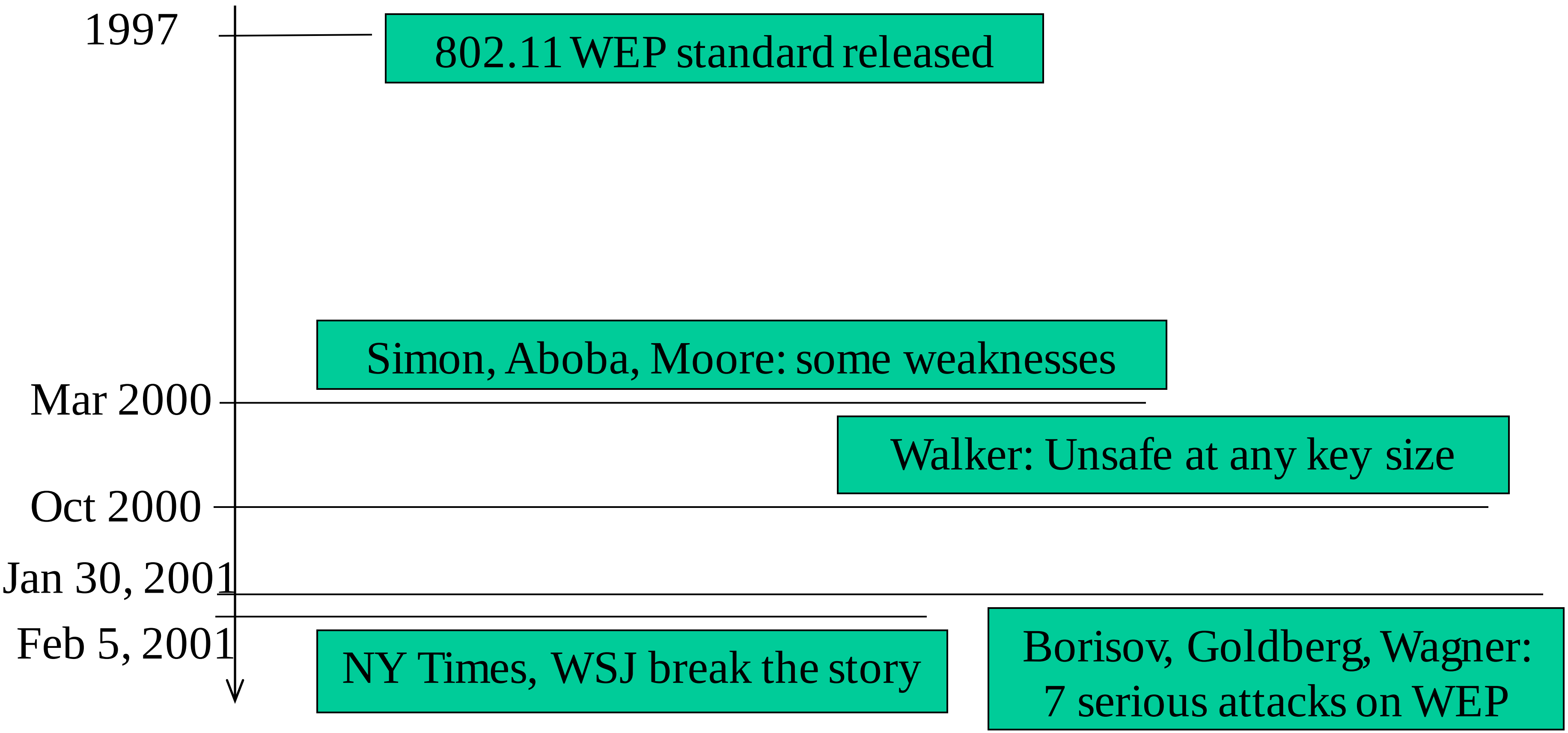


- The industry's solution: WEP (Wired Equivalent Privacy)
  - Share a single cryptographic key among all devices
  - Encrypt all packets sent over the air, using the shared key
  - Use a checksum to prevent injection of spoofed packets

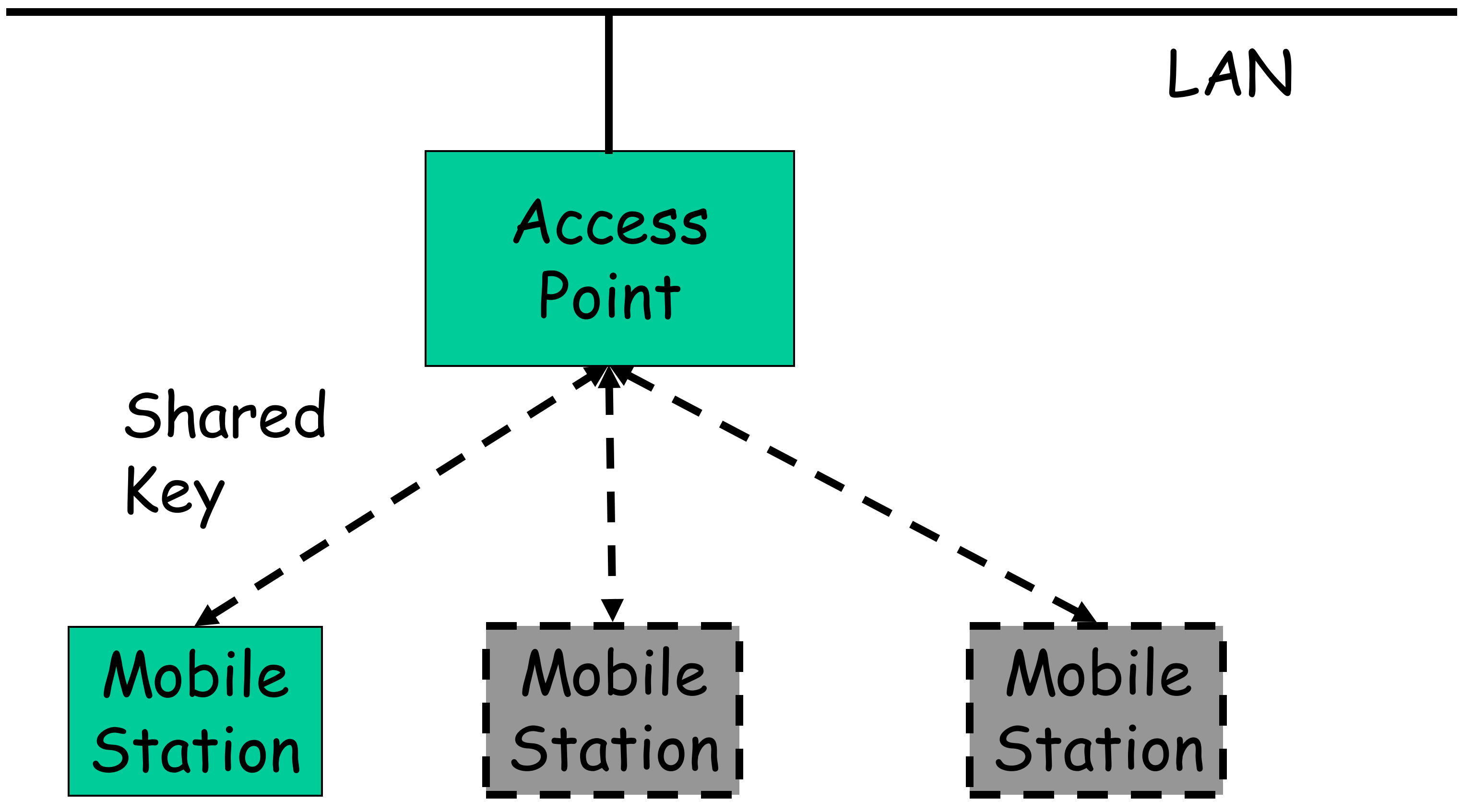
802.11 Security

- “Wired Equivalent Privacy” protocol (WEP)
- Protects wireless data transmissions
- Security goals:
  - Prevent eavesdropping [privacy]
  - Prevent message modification [integrity]
  - Control network access [access control]
- Essentially, equivalent to wired security
- Only protects the wireless link
  - … not an end-to-end solution

Early History of WEP



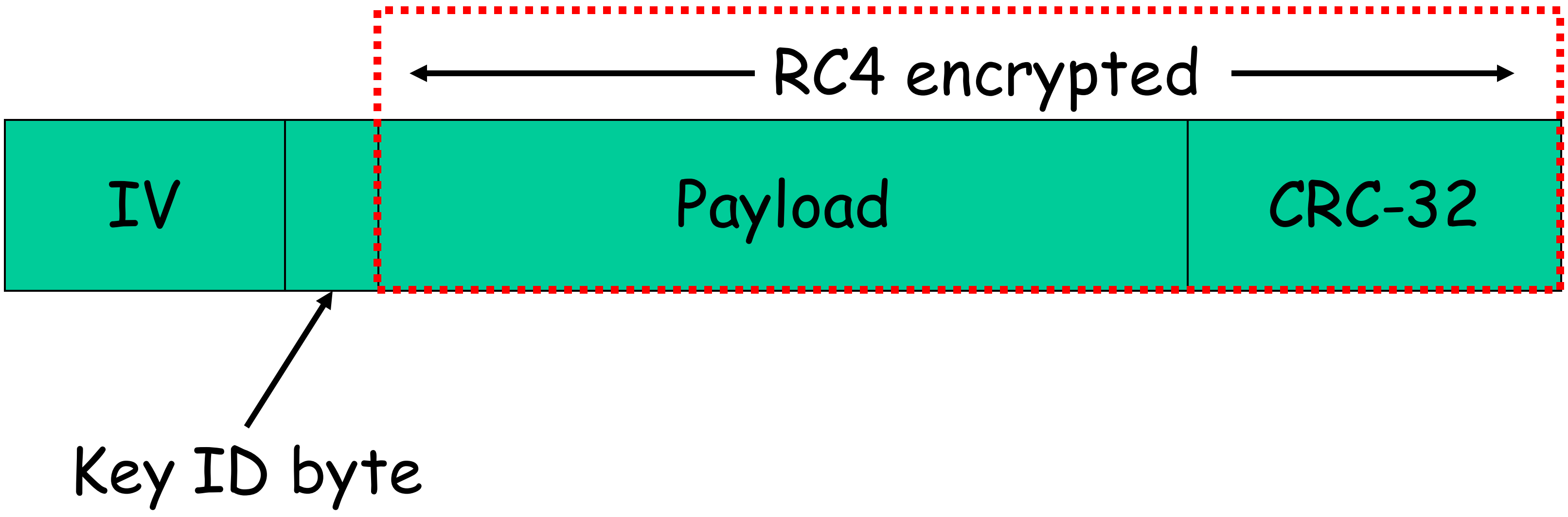
Protocol Setup



Protocol Setup

- Mobile station shares key with access point
  - Various key distribution strategies
  - One shared key per installation is common
- Integrity check (CRC) computed over packet
- Packet + CRC are encrypted with shared key
  - … together with an IV
- Receiver decrypts and verifies CRC
- Packet accepted if verification succeeds

Packet Format



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 $\langle M, c(M) \rangle$ with the keystream to get the ciphertext
  - Transmit v and the ciphertext over the wireless 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 $\langle M', c' \rangle$
  - Check to see if $c' = c(M')$
  - If it is, accept M' as the message transmitted

Notes:

- V is 24 bits long
- CRC is linear
  - I.e. $CRC(X \oplus Y) = CRC(X) \oplus CRC(Y)$

Example

"WIRELESS"	=	**574952454C455353**	
RC4("foo")	=	**0123456789ABCDEF**	
			XOR
		566A1722C5EE9EBC	
RC4("foo")	=	**0123456789ABCDEF**	
			XOR
"WIRELESS"	=	**574952454C455353**	

Group Discussion:

- How to attack WEP protocol?

Initialization Vectors

- Encrypting two messages with the same part of RC4 keystream is disastrous:
  - $C1 = P1 \oplus RC4(key)$
  - $C2 = P2 \oplus RC4(key)$
  - $C1 \oplus C2 = P1 \oplus P2$
  - Keystream cancels out!
- Use initialization vector to augment the key
  - Key = base_key || IV
  - Different IVs produce different keystreams
- Include IV (unencrypted) in header

Problem 1: IV collision

- What if two messages use the same IV?
- Same IV $\Rightarrow$ same keystream!
- $C1 \oplus C2 = P1 \oplus P2$
- If P1 is known, P2 is immediately available
- Otherwise, use expected distribution of P1 and P2 to discover contents
  - Much of network traffic contents predictable
  - Easier when three or more packets collide

Finding IV collisions

- 802.11 doesn't specify how to pick IVs
  - Doesn't even require a new one per packet
- Many implementations reset IV to 0 at startup and then count up
- Further, only 2^{24} IV choices
  - Collisions guaranteed after enough time
  - Several hours to several days
- Collisions more likely if:
  - Keys are long-lived
  - Same key is used for multiple machines

Decryption Dictionary

- Once a packet is successfully decrypted, we can recover the keystream:
  - RC4(k,IV) = P xor C
- Use it to decrypt packets with same IV
- If we have 2^{24} known plaintexts, can decrypt every packet
- Store decryption dictionary on a cheap hard drive
- For counting IVs starting at 0, smaller dictionaries can be effective

Problem 2: Linear Checksum

- Encrypted CRC-32 used to check integrity
  - Fine for random errors, but not deliberate ones
- CRC is linear
  - I.e. $CRC(X \oplus Y) = CRC(X) \oplus CRC(Y)$
- $RC4(k,X \oplus Y) = RC4(k,X) \oplus Y$
- $RC4(k,CRC(X \oplus Y)) = RC4(k,CRC(X)) \oplus CRC(Y)$
  - Hence we can change bits in the packet

Packet Modification

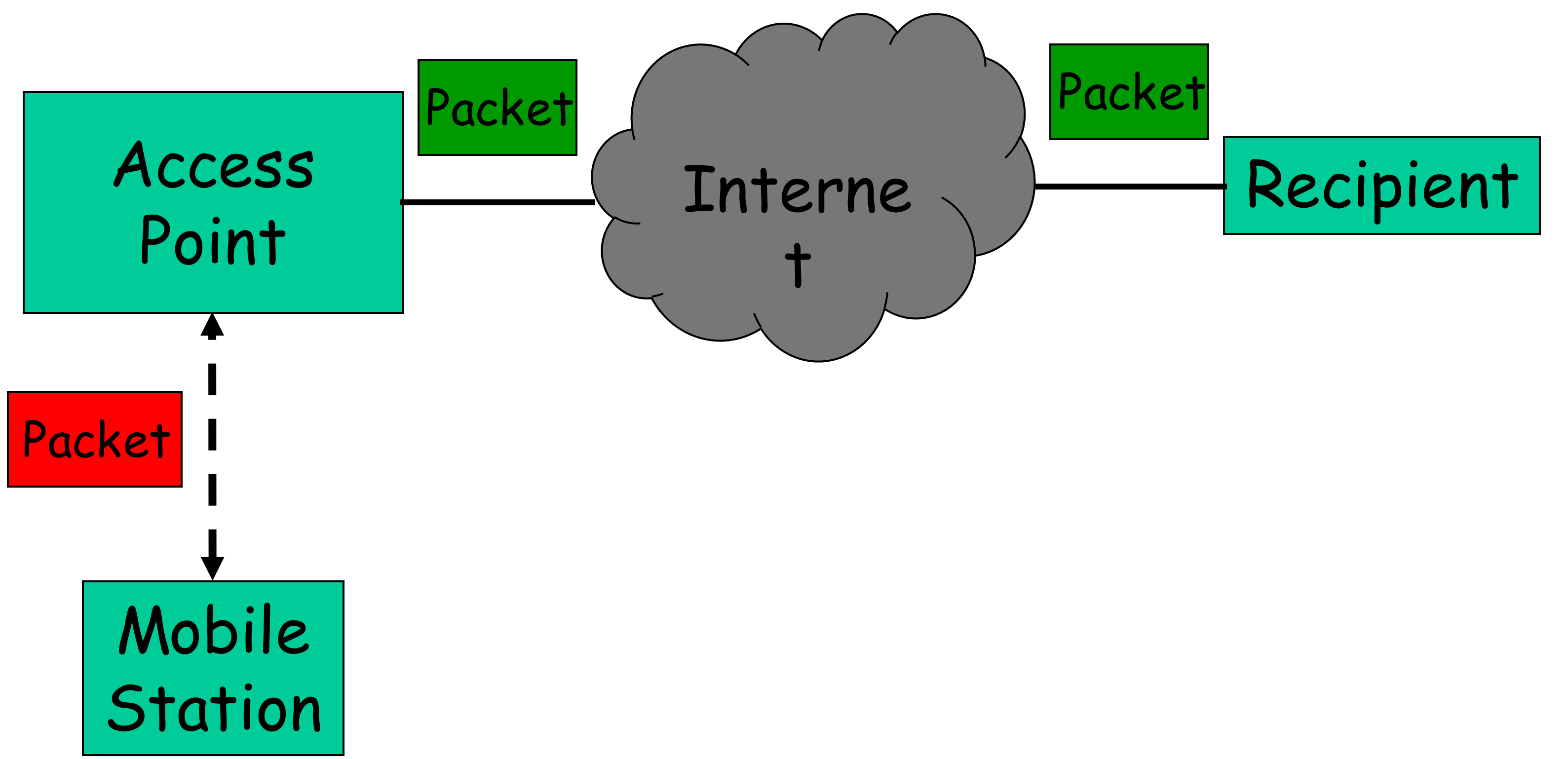
	Payload	CRC-32	
	011010010100……………………………………	10110…………	
RC4	101101110101………………………………………………………		XOR
	110111100001……………………………………	11011…………	
	010000000000……………………………………	00110…………	XOR
	100111100001……………………………………	11101…………	
	Modified Packet		

$$RC4(k, CRC(X \oplus Y)) = RC4(k, CRC(X)) \oplus CRC(Y)$$

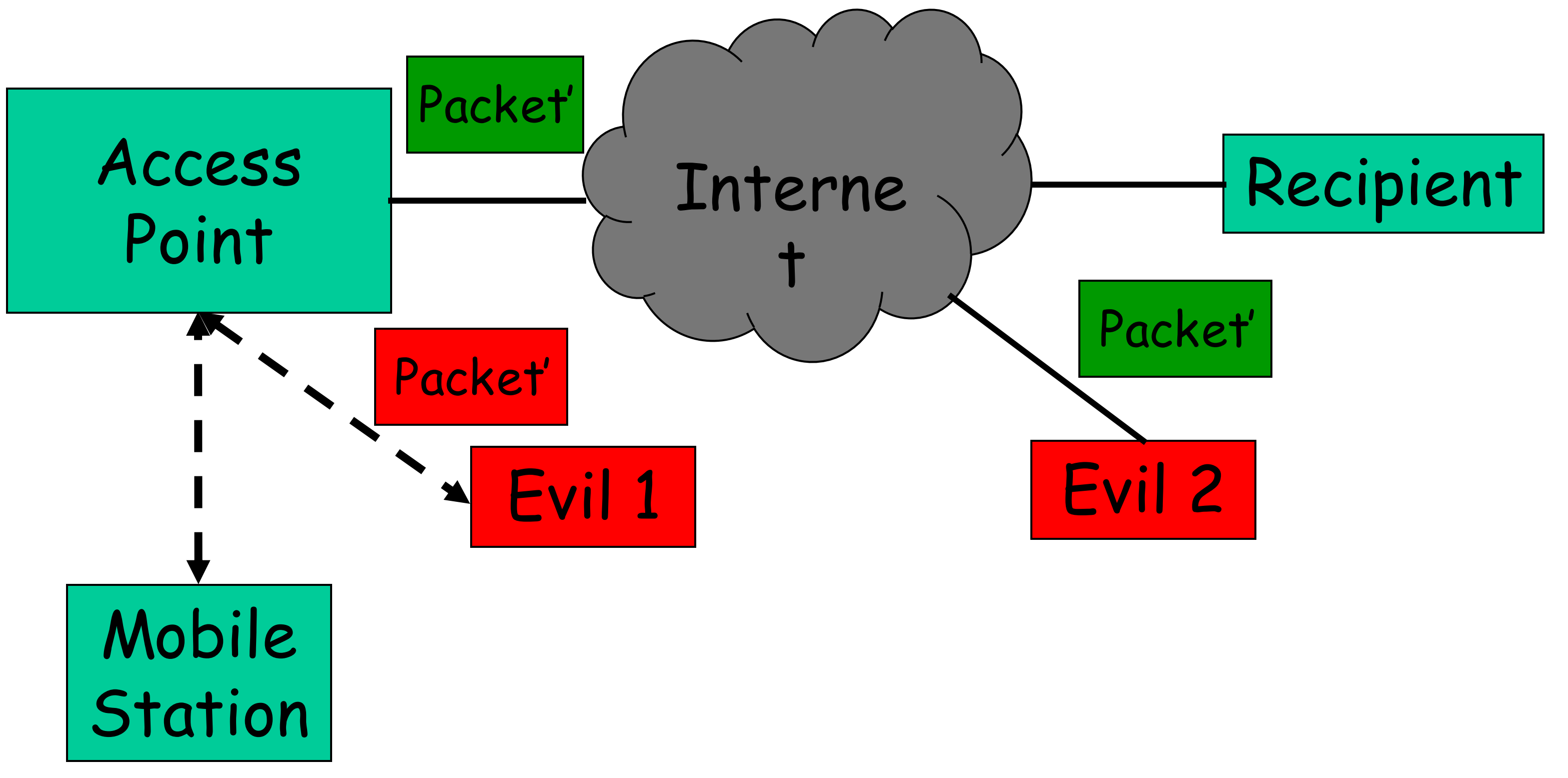
Can modify packets!

- "Integrity check" does not prevent packet modification
- Can maliciously flip bits in packets
  - Modify active streams
  - Bypass access control
- Partial knowledge of packet is sufficient
  - Only modify the known portion

Typical Operation



Redirection Attack



Redirection Attack

- Suppose we can guess destination IP in encrypted packet
- Flip bits to change IP to Evil 2, send it to AP
  - Tricks to adjust IP checksum (in paper)
- AP decrypts it, then forwards it to Evil 2

- Incorrect TCP checksum not checked until Evil 2 sees the packet!

Reaction Attacks

- Send encrypted packet to the *AP*
- *AP* decrypts it for further processing
- System reacts to the **decrypted** data
- Monitor reaction
  - Learn information about decrypted data
  - Usually only a few bits
- Reaction becomes a *side channel*
- Learn more data with multiple experiments

TCP reaction attack

- Carefully modify an intercepted packet
- TCP checksum will be correct or incorrect depending on the decrypted contents
- Reinject packet, watch reaction
  - ACK received $\Rightarrow$ TCP checksum correct
  - Otherwise, checksum failed
- Learn one bit of information about packet
- Repeat many times to discover entire packet

Fluhrer et al Attack on RC4

- Designer's worst fear: new flaw in encryption algorithm
- Attack:
  - Monitor encrypted traffic
  - Look for special IV values that reveal information about key state
  - Recover key after several million packets

(many technical details omitted)

Practical Considerations

- Park van outside of house or office
  - With good antenna and line of sight, can be many blocks away
- Use off-the-shelf wireless card
- Monitor and inject traffic
  - Injection potentially difficult, but possible
- Software to do Fluhrer et al attack readily available

Lesson: Public Review Essential

- IEEE used “open design”
  - Anyone allowed to participate meetings
  - Standard documents freely available (used to cost $$)
- However:
  - Only employees sponsored by companies can afford the time and expense of meetings
  - No review by cryptography community
- Many flaws are not new
  - E.g. CRC attacks, reaction attacks
  - Arguably, even the Fluhrer et al attack could have been prevented

Lesson: Message Integrity Essential

- Message integrity was only a secondary goal
- However, poor integrity can compromise privacy as well:
  - IP redirection attack
  - TCP reaction attack
  - Inductive CRC attack [Arbaugh'01]
- Proper cryptographic authentication necessary
- "Encryption without integrity checking is all but useless" [Bellovin'96]

Recovering a WEP key

- 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 (often) how WEP uses RC4!
- 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
  - See the optional reading for more information on this

Replacing WEP

- 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
- 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 (TKIP)
  - Ability to use 802.1x 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 CCM authenticated encryption mode (using AES)
  - Considered strong, except in PSK mode
    - Dictionary attacks still possible

Is WPA2 security enough?

Home | Call for... | Agenda | Workshops | Registration | Travel & Venue | Organization | Accepted | Proceedings | Awards | Talks

Awards

Real-World Impact Award

This is a newly-created award recognizes papers with substantial immediate impact.

- *The Return of Coppersmith's Attack: Practical Factorization of Widely Used RSA Moduli* [PDF] [Artifact] (H1)
  Matus Nemec, Marek Sys, Petr Svenda, Dusan Klinec, Vashek Matyas
- *Key Reinstallation Attacks: Forcing Nonce Reuse in WPA2* [PDF] [Paper] (F3)
  Mathy Vanhoef, Frank Piessens

ACM CCS 2017 Real-World Impact Award

Key Reinstallation Attacks: Forcing Nonce Reuse in WPA2

Mathy Vanhoef
imec-DistriNet, KU Leuven
Mathy.Vanhoef@cs.kuleuven.be

Frank Piessens
imec-DistriNet, KU Leuven
Frank.Piessens@cs.kuleuven.be

ABSTRACT

We introduce the key reinstallation attack. This attack abuses design or implementation flaws in cryptographic protocols to reinstall an already-in-use key. This resets the key's associated parameters such as transmit nonces and receive replay counters. Several types of cryptographic Wi-Fi handshakes are affected by the attack.

All protected Wi-Fi networks use the 4-way handshake to generate a fresh session key. So far, this 14-year-old handshake has remained free from attacks, and is even proven secure. However, we show that the 4-way handshake is vulnerable to a key reinstallation attack. Here, the adversary tricks a victim into reinstalling an already-in-use key. This is achieved by manipulating and replaying handshake messages. When reinstalling the key, associated parameters such as the incremental transmit packet number (nonce) and receive packet number (replay counter) are reset to their initial value. Our key reinstallation attack also breaks the PeerKey, group key, and Fast BSS Transition (FT) handshake. The impact depends on the handshake being attacked, and the data-confidentiality protocol in use. Simplified, against AES-CCMP an adversary can replay and decrypt (but not forge) packets. This makes it possible to hijack

work, we present design flaws in the 4-way handshake, and in related handshakes. Because we target these handshakes, both WPA- and WPA2-certified products are affected by our attacks.

The 4-way handshake provides mutual authentication and session key agreement. Together with (AES)-CCMP, a data-confidentiality and integrity protocol, it forms the foundation of the 802.11i amendment. Since its first introduction in 2003, under the name WPA, this core part of the 802.11i amendment has remained free from attacks. Indeed, the only currently known weaknesses of 802.11i are in (WPA-)TKIP [57, 66]. This data-confidentiality protocol was designed as a short-term solution to the broken WEP protocol. In other words, TKIP was never intended to be a long-term secure solution. Additionally, while several attacks against protected Wi-Fi networks were discovered over the years, these did not exploit flaws in 802.11i. Instead, attacks exploited flaws in Wi-Fi Protected Setup (WPS) [73], flawed drivers [13, 20], flawed random number generators [72], predictable pre-shared keys [45], insecure enterprise authentication [21], and so on. That no major weakness has been found in CCMP and the 4-way handshake, is not surprising. After all, both have been formally proven as secure [39, 42].