

# Wireless Security

Cyber Security

Spring 2008

# 802.11 or Wi-Fi

- IEEE standard for wireless communication
  - Operates at the physical/data link layer
  - 802.11b – 11Mb/s – 35-100 meter range – 2.5GHz
  - 802.11a – 54Mb/s – 25-75 meter range – 5GHz
  - 802.11g – 54Mb/s – 25-75 meter range – 2.5GHz
  - 802.11n – 540Mb/s – 50-126 meter range – 2.4 or 5 GHz
- Wireless Access Point is the radio base station
  - The access point acts as a gateway to a wired network e.g., ethernet
- Laptop with wireless card uses 802.11 to communicate with the Access Point

# External Security Mechanisms

- MAC restrictions at the access point
  - Protects servers from unexpected clients
  - Unacceptable in a dynamic environment
  - Of course MAC isn't really secure. You can reprogram your card to pose as an "accepted" MAC.
- IPSec
  - To access point or some IPSec gateway beyond
  - Protects clients from wireless sniffers

# Basic Wireless Sniffing

- Wired cards go into “promiscuous” mode
  - See all packets that pass on wire even if not destined to that host
- Wireless can also go “promiscuous”
  - See all packets on the associated AP
- Can also go into “monitor” mode
  - See all packets in radio range

# Wireless under linux

- Use iwconfig to control
  - iwconfig <if>
- Use iwlist to get more details
  - iwlist <if> scanning
- Use ifconfig to set IP level information as for a wired interface

# More sniffing

- Can use standard packet sniffers in monitoring or promiscuous mode
- Can use wardrive tools to gather greater intelligence
  - Unadvertised SSIDs
  - Suspicious behaviour
  - Weak keys

# Kismet

- One wardriving tool
  - Press H or h to get to help screen
  - M or m to mute
  - Q to quit Kismet. q to quit a current menu
- Saves logs in `/var/log/kismet`
- Configuration file in `/etc/kismet/kismet.conf`

# Wired Equivalent Privacy (WEP)

- Excellent example of how security system design can go wrong.
  - Flaws widely published in late 2000
  - (In)Security of the WEP algorithm.  
<http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html>
  - Unsafe at Any Key Size. Tech. Rep. 00/362  
<http://grouper.ieee.org/groups/802/11/Documents/>
- Took secure elements and put them together poorly
  - RC4 stream ciphers and per packet initialization vectors
  - Encrypting 32 bit CRC for message authentication

# RC4 Stream Cipher

- Takes a key value as input and generates a key stream
  - Key stream is XOR'ed with plaintext to create ciphertext
  - $ci = pi \oplus ki$ , for  $i = 1, 2, 3$
  - Ciphertext is XOR'ed with key stream to create plaintext,
  - $pi = ci \oplus ki$ , for  $i = 1, 2, 3$
- Knowing two of key stream, plaintext, and ciphertext lets you easily compute the third
  - Reusing a key value is a really, really bad idea. A well known fact for RC4
  - Enables trivial attacks if you can inject traffic
  - Enables somewhat less trivial attacks from passive sniffing.

# WEP's use of RC4

- RC4 seed is created by concatenating a shared secret with a 24 bit initialization vector (IV)
  - Frames can be lost and stream ciphers do not deal with missing bits, so the stream must be reset with each packet.
  - Therefore, a new IV is sent in the clear with each packet
- Since the IV is reset and the IV is only 24 bits, the time to repeat IV's (and thus keys) with high probability is very short
  - Randomly select IV's and probability of reuse  $p_k = p_{k-1} + (k-1) \cdot 1/n \cdot (1 - p_{k-1})$ , where  $n=2^{24}$
  - 99% likely that you get IV re-use after 12,430 frames or 1 or 2 seconds of operation at 11 Mbps.
- WEP defines no automatic means of updating the shared key
  - In practice folks do not frequently update WEP keys
  - Ideally should be changing shared key after 6 frames to keep low probability of IV collision (99.999% probability of no IV reuse)
- RC4 has weak keys
  - Use of weak keys greatly aid crypto analysis
  - There are standard techniques to avoid the weak keys but WEP does not employ these techniques.

# WEP CRC Problems

- We encrypt the CRC, so it is secure, right?
- Wrong. CRC is linear
  - Flipping bits in the ciphertext can be fixed up in the CRC even if the CRC is RC4 encrypted
- This means that an attacker can change the cipher text and fix up the CRC
  - Cannot do this with crypto hashes used by IKE

# WEP Active Attacks

- Insert known plaintext
  - Send email (probably forged or anonymized) to someone on the access point and sniff the stream
  - Knowing both plain and ciphertext getting the key stream for that IV is just an XOR
- Sniff both the wireless stream and the wire after the access point
  - Correlate the two streams to get plain and ciphertext pairs

# WEP Passive Attacks

- Each frame contains one IP packet
  - Use knowledge about IP headers to get partial key recovery for all packets
- XORing ciphertext streams using the same key will result in the XOR of the two plaintext streams
  - Knowing how plaintext streams differ can help in the analysis
  - Use natural language facts to determine the likely plain text

# WEP crackers

- Numerous tools will crack WEP given enough traffic
  - Airtsnort
    - We'll play with airtsnort
  - Wepcrack

# How do other security protocols avoid these problems?

- SSL uses RC4 without these problems
  - Over a reliable data stream so the 128 bit key does not need to be reset with each packet
  - Would need to capture  $2^{64}$  streams rather than  $2^{12}$  streams to get key reuse with 50% probability
  - New keys potentially change all bits not just the bottom 24 bits.
- IPSec has the unreliable transport issues too, but its security has stood up
  - Uses separate keys in each direction
  - Uses 64 bit (for 3DES) or 128 bit (for AES) IV's
  - Uses the IV as a salt not as part of the key
  - Forces a rekey after at most  $2^{32}$  packets

# LEAP: One WEP Patch

- Cisco and Microsoft driven
  - Tried to get a fix out to market quickly because of all the flap over WEP
  - But didn't get it quite right
  - LEAP: a Looming Disaster <http://www.lanarchitect.net/Articles/Wireless/LEAP/>
- Problem is the use of EAP-MD5
  - Not appropriate for use over an unsecured physical layer like wireless
  - MS-Chapv2 is used to protect the credentials and is weak
    - User name sent in clear
    - No salt in the hashes
    - 2 byte DES key
  - So the sniffer can easily launch an offline dictionary attack
  - ASLEAP attack tool <http://asleap.sourceforge.net/> Implements the attack
- One solution is to force your users to use strong passwords
  - Probably not a good basis for security though
- EAP-TLS came out about the same time, but it requires certificate deployment and so was not as popular
  - EAP-TTLS and PEAP came later and only required access point certificates

# 802.11i

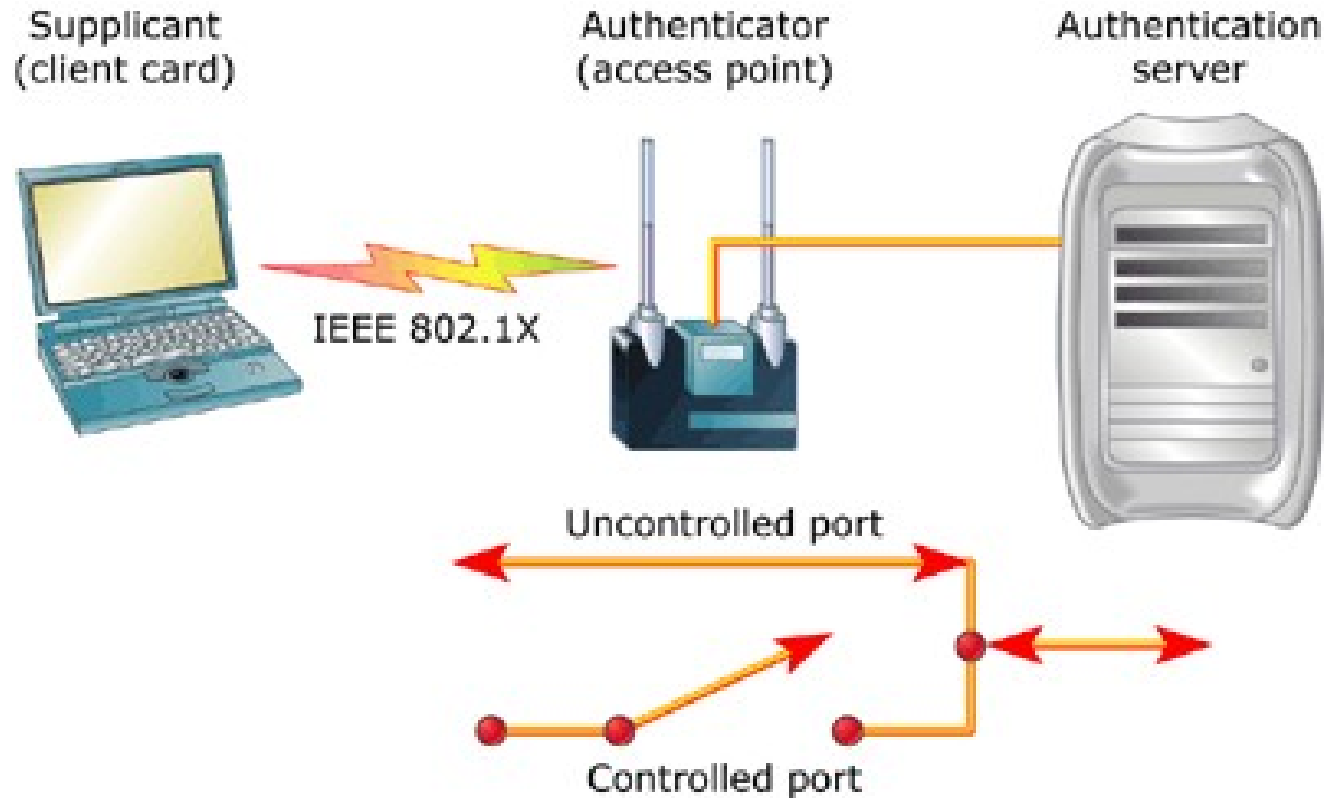
- IEEE effort to improve security of the 802.11 spec
  - Using 802.1X for authentication
- Wi-Fi Alliance promoting interim standards
  - WPA, a shorter term solution that uses existing hardware
  - WPA2, an implementation of the full 802.11i standard

# 802.11i Reading Material

- Overview of industry slides from 2003  
[http://csrc.nist.gov/wireless/S10\\_802.11i%20Overview-jw](http://csrc.nist.gov/wireless/S10_802.11i%20Overview-jw)
- Cisco white paper  
<http://cisco.com/en/US/products/hw/wireless/ps430/produ>
- Cisco FAQ  
<http://cisco.com/en/US/netsol/ns340/ns394/ns348/ns337/>
- Recent overview article from Embedded.com <http://www.embedded.com/showArticle.jhtml?articleID=34400002>

# 802.1X

- Evolved from PPP authentication
  - Extensible Authentication Protocol (EAP)
- Not an authentication protocol, but an authentication transport
- 802.1X extends EAP mechanisms so they can be used over wires or wireless



# 802.11i Exchanges



**Station**



**Access  
Point**



**Authentication  
Server**



**Security capabilities  
discovery**



**802.1X authentication**



**802.1X key management**



**RADIUS-based key  
distribution**

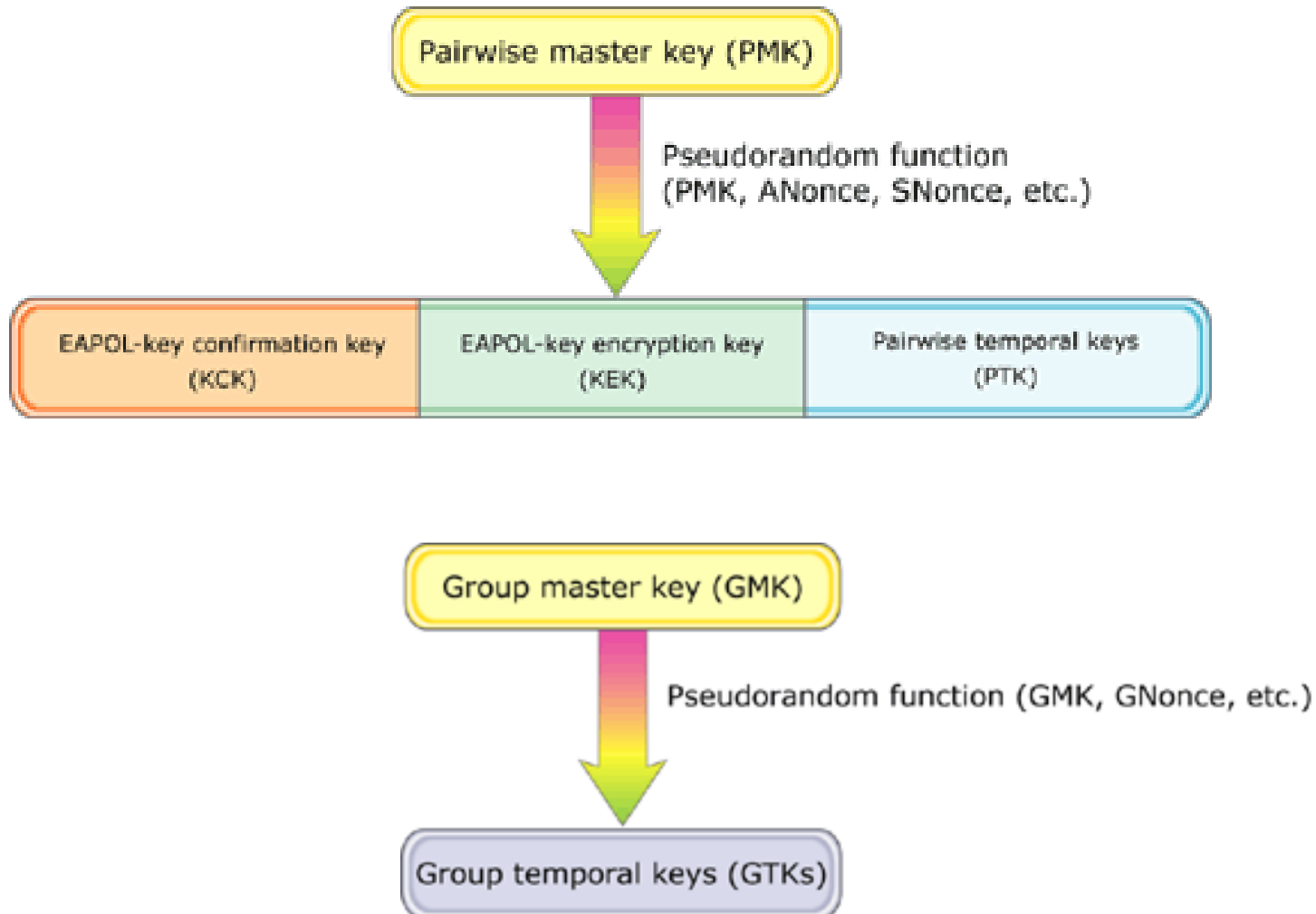


**Data protection**

# Key Management from 802.1x

- EAPOL (Extensible Authentication Protocol over LANs) key exchange
  - 4 way handshake to negotiate pairwise keys
    - A Pairwise Master Key (PMK) is provided by the authentication server
    - The handshake negotiates temporal keys and starts the group key negotiation
    - New session key for each client association
  - Group key handshake to negotiate group (broadcast) keys
    - Protocol forces a periodic rekeying of the group keys

# Key hierarchies





STA

# Step 2: 4-Way Handshake



AP



PMK



PMK

Pick Random ANonce

EAPoL-Key(Reply Required, Unicast, ANonce)

Pick Random SNonce, Derive **PTK** = EAPoL-PRF(**PMK**, ANonce | SNonce | AP MAC Addr | STA MAC Addr)

EAPoL-Key(Unicast, SNonce, **MIC**, STA RSN IE)

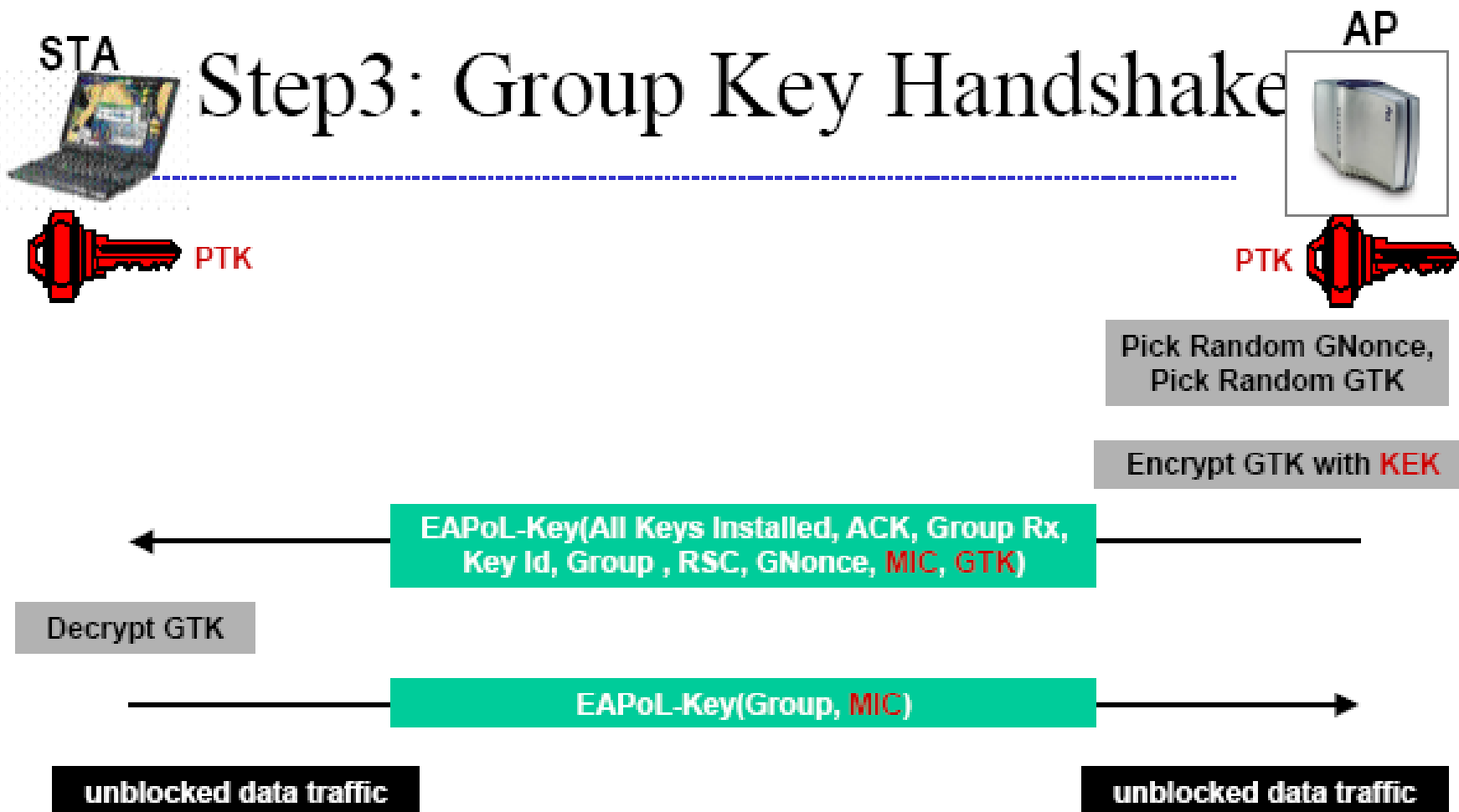
Derive **PTK**

EAPoL-Key(Reply Required, Install PTK, Unicast, ANonce, **MIC**, AP RSN IE)

Install TK

EAPoL-Key(Unicast, **MIC**)

Install TK



# Home and Enterprise Modes

- Independent Basic Service Set (IBSS)
  - For small installations (like a home network)
  - Use pre-shared keys
  - No authentication server
- Extended Service Set (ESS)
  - For larger installations
  - Can use pre-shared keys or certificates
  - Use authentication server

# Wi-Fi Protected Access (WPA)

- Interim solution to run on existing wireless hardware
- Uses Temporal Key Integrity Protocol (TKIP) for data encryption and confidentiality
  - Still uses RC4, 128 bits for encryption
  - Provisions for changing base keys
  - Avoids weak keys
- Includes Michael a Message Integrity Code (MIC)
  - 64 bits
  - Replaces the CRC
  - Observer cannot create new MIC to mask changes to data
- Increases IV from 24 bits to 48
- Mixes the IV and the base key

# WPA2

- Uses AES, specifically Counter-Mode/CBC-MAC Protocol (CCMP)
  - Too computationally intensive in SW for wireless hardware deployed at the time of WEP
- Uses 128 bit key
- Provides data confidentiality by using AES in counter mode
- Provides message authentication using Cipher Block Chaining Message Authentication Code (CBC-MAC)
  - The MAC also covers the packet source and destination

# 802.11i Summary

|                   | <u>WEP</u>      | <u>TKIP</u>                            | <u>CCMP</u> |
|-------------------|-----------------|--|-------------|
| <i>Cipher</i>     | RC4             | RC4                                    | AES         |
| <i>Key Size</i>   | 40 or 104 bits  | 128 bits<br>encryption,<br>64 bit auth | 128 bits    |
| <i>Key Life</i>   | 24-bit IV, wrap | 48-bit IV                              | 48-bit IV   |
| <i>Packet Key</i> | Concat.         | Mixing Fnc                             | Not Needed  |
| <i>Integrity</i>  |                 |  |             |
| <i>Data</i>       | CRC-32          | Michael                                | CCM         |
| <i>Header</i>     | None            | Michael                                | CCM         |
| <i>Replay</i>     | None            | Use IV                                 | Use IV      |
| <i>Key Mgmt.</i>  | None            | EAP-based                              | EAP-based   |