Seguridad en Sistemas Informáticos

Seguridad en redes inalámbricas

Área de Ingeniería Telemática
Dpto. Automática y Computación
http://www.tlm.unavarra.es/

En clases anteriores...

- La cadena de seguridad
- Redes inalambricas
  - 802.11
  - WEP

Hoy:

- Problemas con WEP
- WPA

Seguridad en redes 802.11

- Primer intento: **Wired Equivalen Privacy (WEP)**

  Conseguir en la red inalámbrica el mismo nivel de privacidad que en una de cable

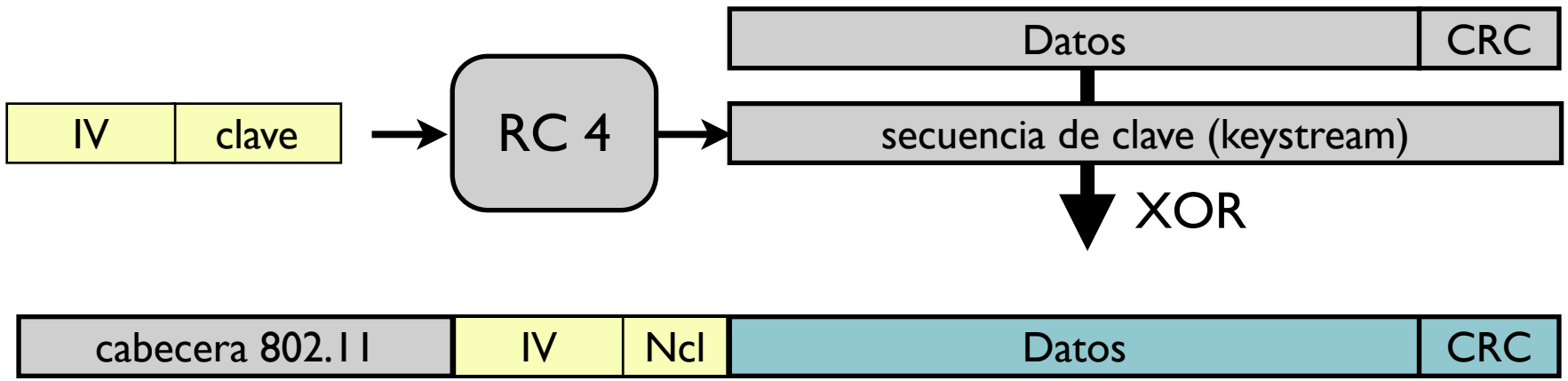
- Se cifran las tramas con el algoritmo RC4
  - Algoritmo de cifrado de tipo clave secreta

    Se basa en generar una serie pseudo-aleatoria a partir de la clave secreta. El mensaje se cifra con una clave de la misma longitud que el mensaje pero que depende de la clave original (intento de hacer un cifrado de Vernan)
- Originalmente era un algoritmo propietario de RCA Security
  - Pero se publicó de forma anónima en Internet y se popularizó

    El algoritmo cifra a gran velocidad y parecía muy seguro
  - Con el tiempo se le han ido encontrando algunos problemas...

WEP

- A los datos de la trama se les añade un CRC para proteger la integridad y se cifran con RC4



- Se usan una clave de 64 o 128 bits
  - Vector de inicialización de 24 bits
  - Secreto compartido de 40 o 104 bits
- El vector de inicialización se cambia en cada paquete para cifrar cada paquete con diferente secuencia. Se envía en cada paquete para que el destinatario sea capaz de descifrar.

Ventajas

- Autentificación sencilla: los usuarios que conozcan la clave pueden usar la red inalámbrica
- Protección de integridad y confidencialidad "razonable"
  - o no?

Desventajas

- Múltiples vulnerabilidades del sistema
  - Contra la confidencialidad
    - La clave se reutiliza. El vector de inicialización de 24 bits solo hay que esperar 16777216 paquetes para que se repita y tener dos paquetes encriptados con la misma clave
    - RC4 tiene claves debiles. Algunos IVs generan claves en las que ciertas partes de la clave secuencia dependen solo de unos pocos bits de la clave original
    - Ataques de fuerza bruta (el secreto compartido depende de una clave introducida por el usuario)
  - Contra la integridad
    - El CRC que se usa fue diseñado para detectar errores no para integridad así que no es un buen hash
    - No hay protección contra inyección de paquetes

      Si repito un paquete que veo en el canal sigue siendo un paquete valido
  - Contra la autentificación
    - Autentificación falsa
    - Ataques de desautentificación

Ataques de fuerza bruta/diccionario

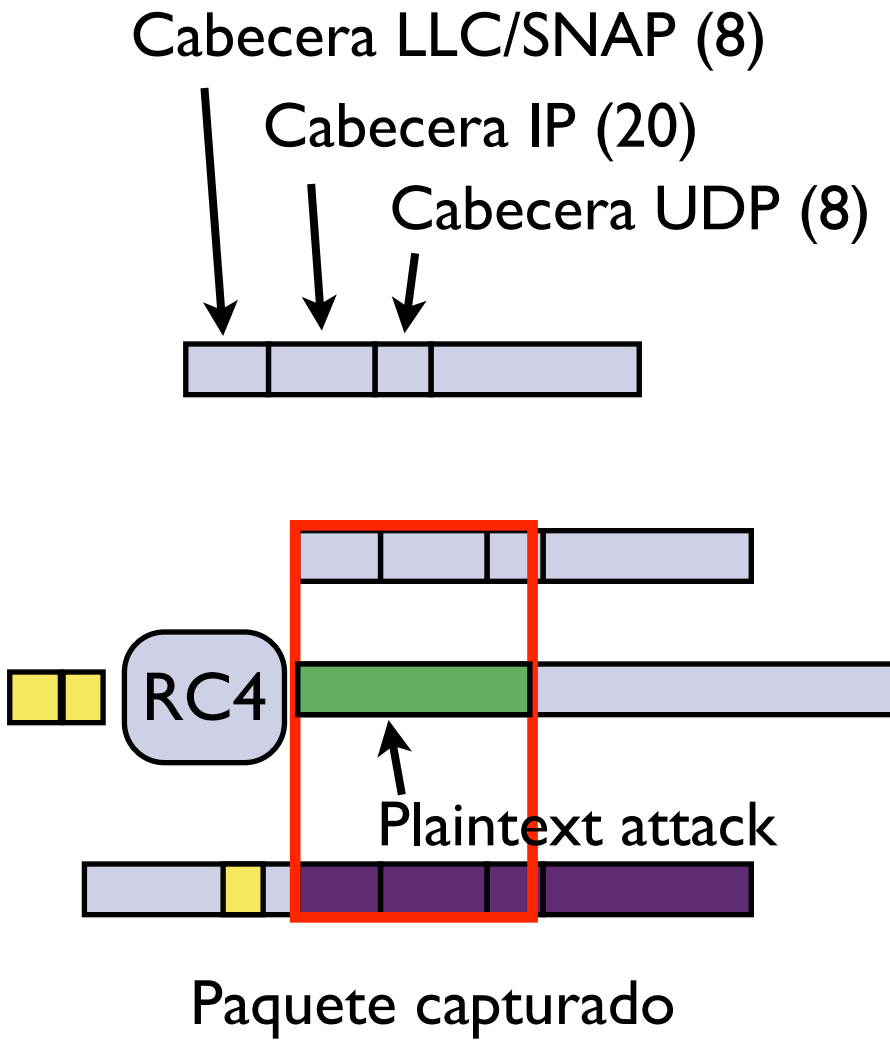
- Podemos hacer ataque de fuerza bruta contra la autentificación?
- El objetivo no es tanto conseguir autentificación como conseguir saber la clave compartida que la necesitamos para descifrar los paquetes wep de otros hosts

- Problemas:
  - Los access points pueden limitarnos el numero de intentos que podemos hacer... es lento
  - La mayoría de los access points no tienen activada la autentificación aunque usen WEP. Para poder hacer fuerza bruta necesito saber si acierto o si no
  - Buena razón para no usar autentificación :-)

Ataques inductivos: Arbaugh (2001)

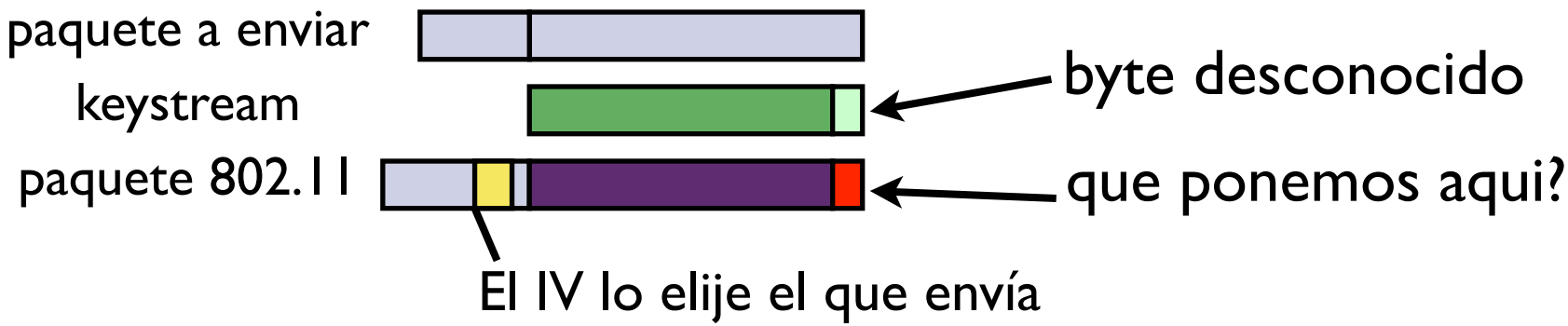
- Suposición: conozco un trozo inicial de secuencia de clave (keystream)
- Por ejemplo si enviamos un paquete de DHCP request para pedir IP
  - Cabecera LLC/SNAP es siempre igual para cualquier paquete IP
  - Los campos de la cabecera IP son fijos o varian entre pocos valores
  - Los campos de la cabecera UDP que transporta DHCP son conocidos
- Si vemos un paquete que imaginamos que es DHCP (por ser el primero que se ve de un nuevo cliente) tenemos un plaintext attack y por tanto 34 bytes de la keystream

  Solo para un IV por supuesto



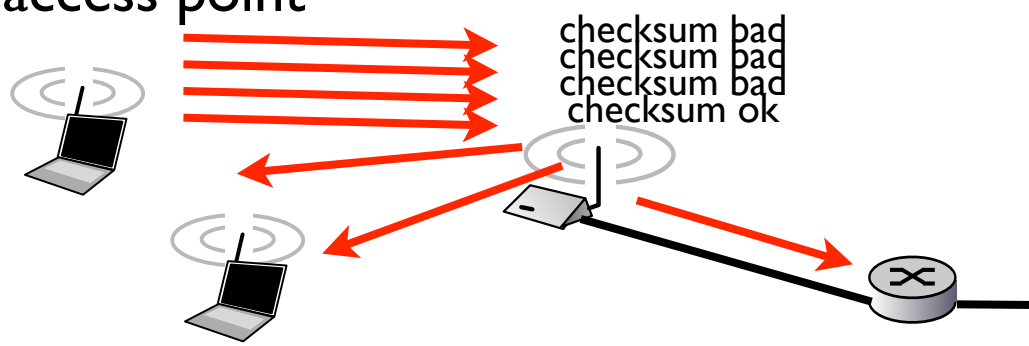
Ataques inductivos: Arbaugh (2001)

- Tenemos n bytes del keystream
- Ya podemos cifrar paquetes de menos de n bytes y enviarlos a la red inalámbrica
- Construimos un paquete valido de n+1 bytes pero solo podemos cifrar n



- Enviamos los 256 posibles paquetes resultantes
- Y observamos cual es reenviado por el access point
- Cuando veamos el paquete reenviado

  Tenemos un byte más del keystream



Ataques inductivos: Arbaugh (2001)

- Procedimiento de ataque
  - Esperar a capturar algo que parezca DHCP request
  - Enviar paquetes (de ping o arp) con un byte más cada vez, en media por cada 128 paquetes averiguamos 1 byte del kystream.
  - Cuando llegamos a 1500 bytes ya tenemos keystream suficiente para mandar cualquier paquete

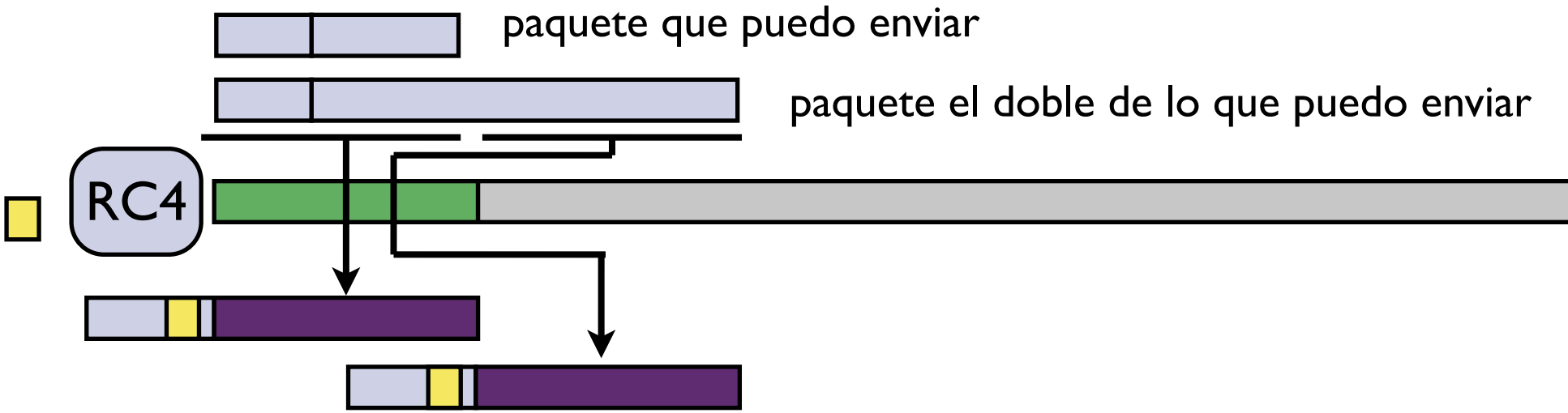
    Para un IV pero podemos elegir el IV de lo que enviamos
  - Problema al recibir: Los paquetes que nos envíen otros vendran con otros IVs de los que no sabemos el keystream...
- podemos averiguarlo?
  - Construir un paquete valido de 1500 bytes.

    Enviarlo con el IV que concemos. El access point lo reenviara con otro IV

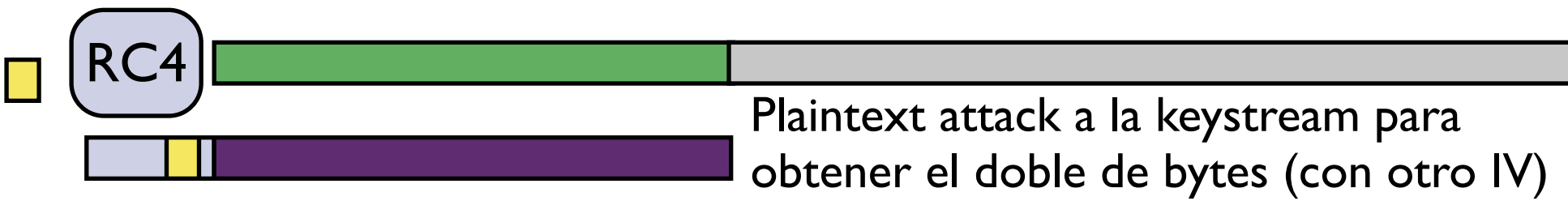
    Con un plaintext attack tenemos toda la secuencia keystream para el nuevo IV
  - Repetir hasta tenerlos todos
- Inconvenientes
  - Hay que enviar 2^{24}, unos 16millones de paquetes más
  - Hay que guardar 2^{24} keystreams de 1500 bytes (unos 25GB)

Ataques inductivos: fragmentacion (2005)

- Supongamos que tenemos los n primeros bytes del keystream (para un IV). Como puedo avanzar más rápido que byte a byte?



- Se envia el paquete fragmentado en dos paquetes 802.11 con el mismo IV. Utilizan el mismo trozo de keystream
- El access point lo reenviará sin fragmentar, o con un ping un host cualquiera nos devolvera un paquete previsible sin fragmentar



Ataques inductivos: fragmentacion (2005)

- Doblando cada vez el número de bytes de la keystream vamos de 34 a 1500 bytes enviando 6 paquetes
- Una vez que tenemos 1500 bytes de keystream para un IV podemos sacar la keystream para todos los IVs como antes

- Hay otros ataques inductivos parecidos

  KoreK. chopchop (2004)

- Pero el mayor problema es que intentan calcular todas las keystreams para todos los posibles

  No podríamos usar esto para adivinar la clave?

Ataques de fuerza bruta

- Conociendo el comienzo de la keystream para un IV tenemos material para hacer ataques de fuerza bruta sobre la clave
  - Para cada clave posible generamos el principio de la secuencia y comparamos hasta obtener la conocida

- Combinaciones a probar
  - Clave de 40bits: 1 billon aprox
  - Clave de 104 bits: 20282409603651670423947251286016 combinaciones

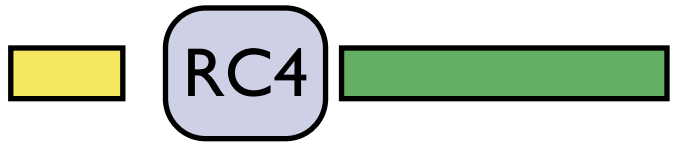
    Impracticable
  - Pero al menos da una manera de probar con diccionario

Ataques estadisticos

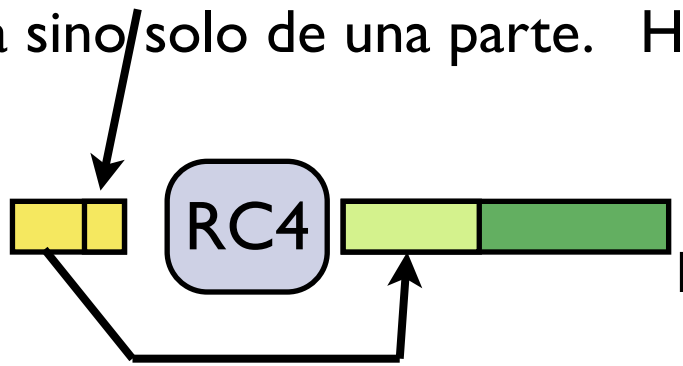
- También en 2001 se publica

  S. Fluhrer, I. Mantin and A. Shamir, *Weaknesses in the Key Scheduling Algorithm of RC4*

- Debilidades:
  - Las secuencias que genera RC4 no son tan aleatorias si se analizan bien
  - Los comienzos de keystream generados con semillas aleatorias tienen propiedades que permiten distinguirlos de numeros aleatorios



  - Propiedades colaterales. Si puedo elegir una parte de la semilla (IV) con ciertas condiciones el comienzo del keystream no depende de toda la semilla sino solo de una parte. Hay IVs débiles que causan esto



Parte de la secuencia de informacion de un unico byte de la clave

Ataques estadisticos

- Se puede hacer un ataque basado en esto
  - De todos los paquetes que vemos podemos sacar los primeros bytes de la keystream (porque la cabecera LLC/SNAP es conocida)
  - Observamos el trafico y si el IV que lleva un paquete es debil lo guardamos junto con el principio de la keystream
  - Cada caso que recolectamos nos dice información probabilística de una parte de la clave
  - Vamos recogiendo información hasta que podemos reducir las claves a probar a unas pocas

- Problema
  - El método funciona muy bien pero necesita que pase tráfico para generar muestras. Ningun problema en redes muy cargadas se puede recoger información suficiente en unos 10 minutos
  - En redes descargadas hay técnicas para aumentar artificialmente el tráfico

Reinyección de paquetes

- No hace falta saber la clave para poder enviar paquetes a una red WEP
- Podemos usar tecnicas inductivas para conseguir el keystream de un IV y construir el paquete que queramos
- Tambien podemos buscar paquetes con tecnicas heuristicas:

  por ejemplo: primer paquete que envie un host tipicamente será una petición de ARP

  si lo envío como si fuera un paquete mío generaré una respuesta con otro IV
- Packet reinjection (aireplay) permiten con técnicas de este tipo generar tráfico artificialmente y por tanto IVs, muchos de ellos seran debiles

- Este es un problema importante de WEP. Integridad de mensajes
  - Un usuario que no este autentificado no deberia poder enviar mensajes
  - Un paquete de un host no deberia ser válido si lo envía otro host (encriptación/CRC deberían depender de la dirección del host)
  - Un paquete no deberia ser válido dos veces consecutivas

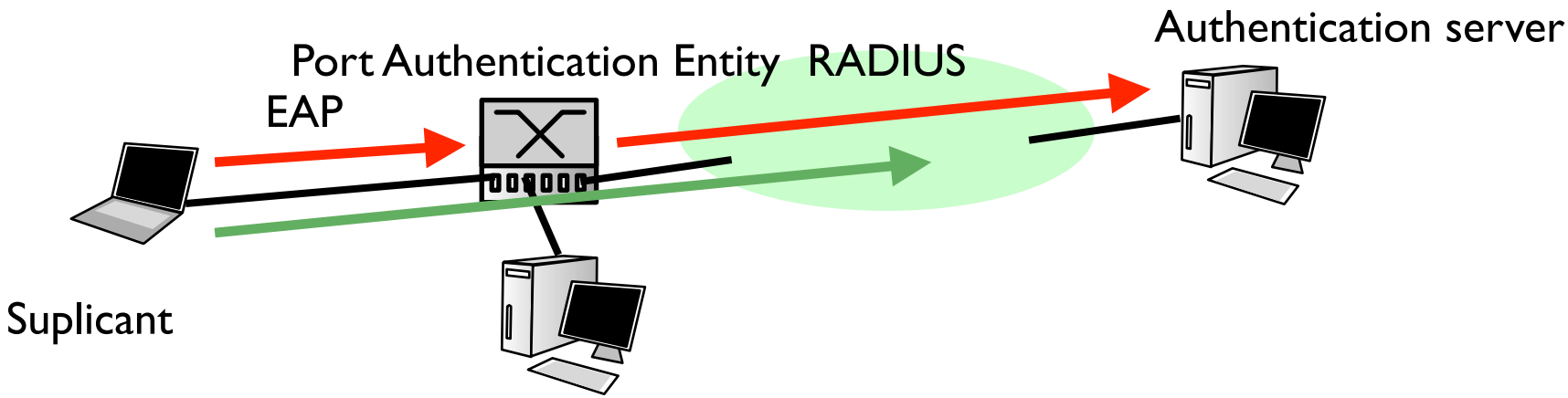
    (Número de secuencia no falsificable)

Mejorando acceso de WEP

- Usar un secreto compartido para autentificar el acceso a la red inalámbrica tiene ciertos problemas
  - Difícil de controlar que no la conozca quien no debe
  - Difícil de cambiar cuando alguien que la conoce deja de ser de confianza (se va de la empresa)
  - Difícil de cambiar cuando el administrador lo decide (por ejemplo para evitar que la misma clave se use durante mucho tiempo
- Y eso unido a que WEP hace bastante fácil atacar y averiguar la contraseña. (Tradicionalmente en horas ahora ya en minutos)
- Otras técnicas para controlar el acceso no basadas en la contraseña WEP?
  - Lista de acceso en el access point con las direcciones MAC permitidas. (solo reenvía tramas de las estaciones declaradas)
  - Control de acceso en redes ethernet: 802.1x

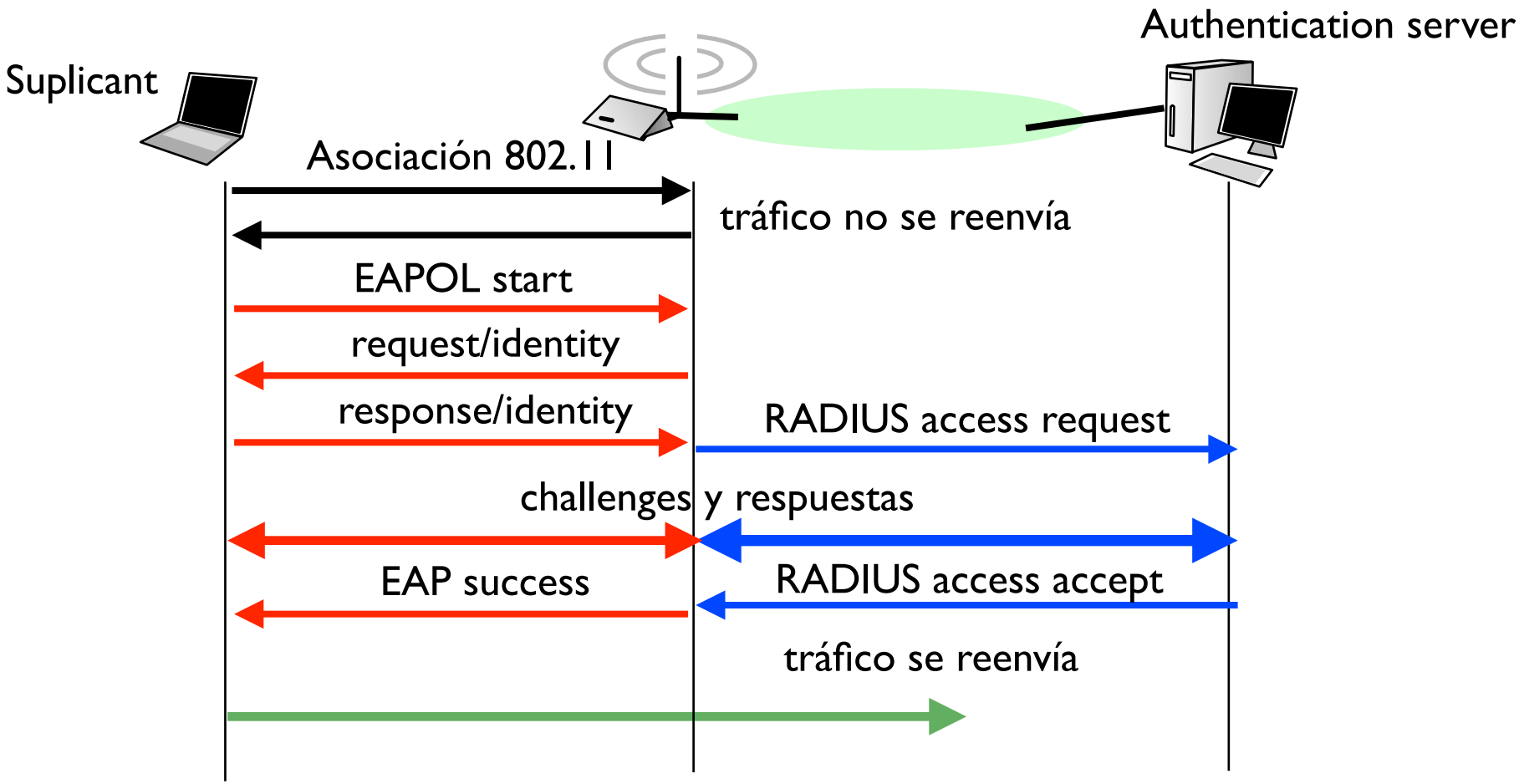
Autentificación

- Ya había un estandar del IEEE para la autentificación en redes de tipo Ethernet es parte de la normativa 802.1
- Arquitectura de autentificación 802.1x



  - El conmutador sólo acepta el traffico 802.1x del suplicante
  - El suplicante se autentifica usando EAP en el servidor de autentificación
    - EAP over LAN : EAPOL para transmitir EAP en ethernet
    - el conmutador utiliza RADIUS para verificar la autenticación
  - Tras completar el proceso el PAE acepta todo el tráfico en el puerto del suplicante

802.1x en inalámbricas



- Se puede usar 802.1x en una red de acceso inalámbrica
- 802.1x autentifica al usuario aunque cambie de maquina
- Acceso protegido por 802.1x no importa que se averigüe la clave WEP
  - Salvo para confidencialidad

Problemas

- Lista con direcciones MAC permitidas
  - Trabajo extra de gestión: mantener lista actualizada
  - Si un usuario cambia de máquina ya no puede acceder hasta obtener el visto bueno del administrador de red
  - Se puede cambiar la dirección MAC de la tarjeta inalámbrica. Solo hay que observar el canal y copiar la de una maquina que tenga acceso
- 802.1x
  - Se puede copiar la dirección MAC de alguien que haya conseguido acceso?
  - Mecanismos que lo impiden
    - EAP incluye mensajes para que el servidor y cliente generen claves comunes para usar en WEP
    - El servidor puede cada cierto tiempo volver a exigir autentificación al usuario o bien perderá el acceso. También puede regenerar nuevas claves WEP

Mejorando confidencialidad de WEP

- 802.11i Estandar del IEE sobre seguridad mejorada en redes 802.11

  Añade:

- Autentificación basada en 802.1x
- 2 nuevos protocolos de cifrado para sustituir a WEP:
  - TKIP: protocolo basado en RC4 pero corrigiendo los problemas de WEP (iba a ser WEP2)

    Fácil de cambiar en hardware que ya soporte WEP
  - CCMP: protocolo completamente rediseñado para nuevo hardware, basado en AES

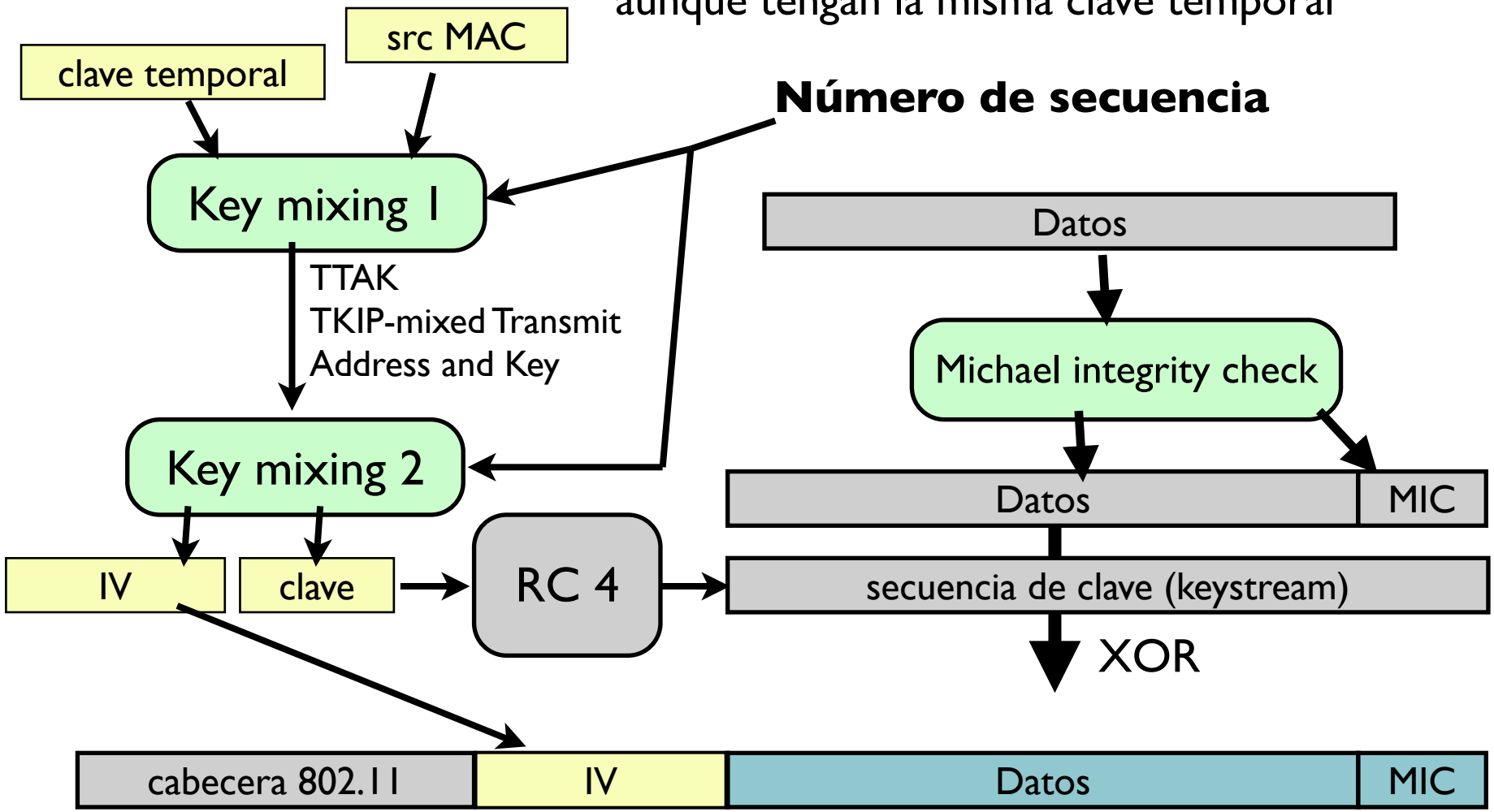
TKIP

- TKIP (Temporal Key Integrity Protocol)
- Usa RC4 pero reforzando los puntos debiles
  - Jerarquía y distribución de claves

    Master keys que se usan para generar **transient keys** que se usan para generar las claves para cifrar cada paquete. Incluye protocolos de comunicación de claves entre el BSS y el usuario inalámbrico
  - Se genera una clave RC4 de entrada para cada paquete
  - Numero de secuencia para evitar ataques de replay (reinyección)
  - Control de integridad no basado en CRC
  - Contramedidas contra fallos de integridad (denegar el acceso por un tiempo)

TKIP

Dos tarjetas WiFi no usaran la misma clave aunque tengan la misma clave temporal



- Ahora la clave RC4 no se reutiliza entre paquetes, depende de la dirección de origen y cambia cada vez que envío algo
- El IV en el paquete se construye de forma que no se permiten claves débiles de RC4

CCMP

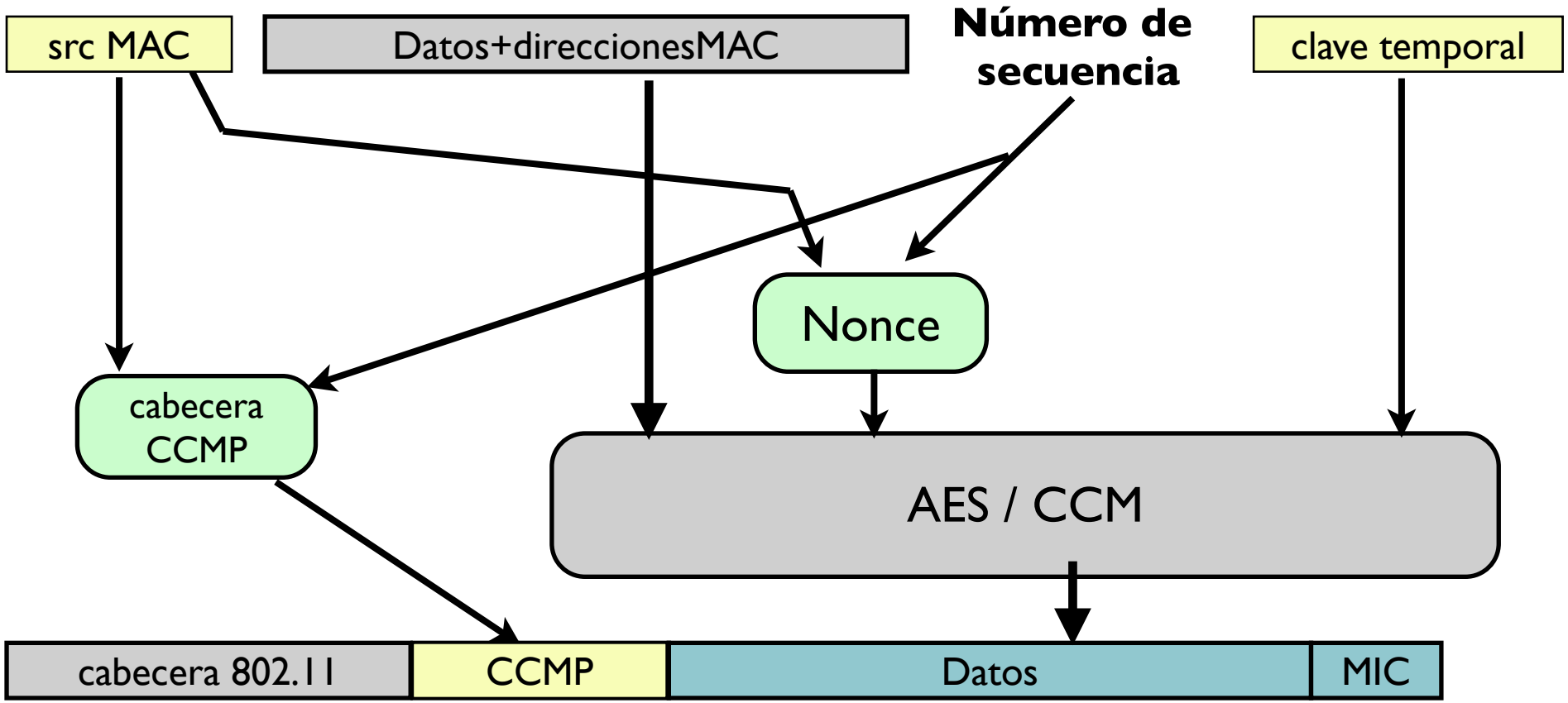
- Algoritmo de cifrado basado en AES (Advanced Encryption Standard).
  - Cifrador de bloque de clave secreta

    Cifra bloques de 128 bits con una clave de 128 bits

    AES está muy estudiado y se considera un cifrador bueno
  - Tiene varios modos de funcionamiento, para cifrar series continuas de datos de forma parecida a RC4.
  - Se utiliza modo de funcionamiento llamado Counter(CRT) con con codigo de autenticidad de mensaje (integridad) mediante Cipher Block Chaining.

    Counter with **C**BC-MAC **M**ode **P**rotocol = CCMP
- Todo esto quiere decir que AES ya proporciona un cifrador que es capaz de generar códigos de integridad

CCMP



- Se cifra con AES el paquete de datos obteniendo los datos encriptados y con prueba de integridad (MIC)
- Para cifrar cada paquete se utiliza un vector de inicialización (Nonce) que depende de la dirección origen y el número de secuencia. En la cabecera se de información suficiente para reconstruir el Nonce en destino

Comparativa de cifradores WiFi

	WEP	TKIP	CCMP
Cifrador usado	RC4	RC4	AES
Tamaño de clave	40 o 104	128 cifrado 64 autentificacion	128
Tamaño IV	24	48	48
Una clave por paquete?	No (cambia el IV)	Si	No hace falta
Integridad de la cabecera?	No	Src y Dst Addr	CCM sobre la cabecera
Integridad de los datos?	CRC32	Michael Integrity	CCM
Detección de reinyección	No	Comprobación de secuencia, no puede haber reinyección/replay	Comprobación de secuencia, no puede haber reinyección/replay
Gestión de claves	Ninguna	802.1x	802.1x

- WEP: considerado malo
- TKIP: aceptable, parcha a WEP pero funciona bien
- CCMP: lo mejor que hay... de momento

Comercialmente

- Nombres de la WiFi alliance para los equipos reales
- **WPA (WiFi protected access)** nombre comercial de TKIP. Se definió a partir del borrador de 802.11i cuando aun se trabajaba en el standar.

  TKIP se implemento antes debido a que estaba basado en el hardware de WEP
- **WPA2** = 802.11i estandar. Con CCMP
- Ambos tienen dos formas de funcionamiento
  - **WPA personal**

    Basado en secreto compartido (las claves se calculan a partir de una clave definida en los BSS y en los PCs)
  - **WPA enterprise**

    Clave basada en TLS y certificados

Es WPA suficiente?

- Es mucho más difícil de atacar aunque hay propuestas de ataques basados en fuerza bruta
- En WPA personal sigue pudiéndose hacer ataques de diccionario a la autentificación
- Se siguen pudiendo hacer ataques de bajo nivel
  - Inundación de paquetes de deautentificación o desasociación
  - Robo de ancho de banda
  - Denegación de servicio por Jamming/interferencia

Un caso particular...

- Todo lo anterior está muy bien en el caso de que yo quiera impedir que los usuarios desconocidos se conecten

  Pero y si lo que quiero es vender el acceso?

  - Cualquiera puede conectarse a mi red pero solo ver mi servidor
  - Si en la pagina web de mi servidor paga por un tiempo de uso puede usar la red inalámbrica para conectarse al exterior
  - Si no solo vera la publicidad de mi servidor
- Desde el punto de vista de 802.11 esto se hace con una red abierta y la autentificación suele hacerse mediante web + SSL
- Mientras no este autentificado las peticiones web se redirigen a un portal web cautivo
- Para casos más complicados se puede usar combinaciones de los anteriores + VPNs

Conclusiones

- El WEP esta definitivamente a extinguir. La duración de una clave WEP de 128 bits ante un ataque es menor de 1 hora
- Si se quiere una mínima seguridad WPA
- WPA2 preferible
- WPA si los equipos no soportan WPA2

- Otras soluciones
  - Portales cautivos sobre redes abiertas
  - VPNs sobre redes abiertas