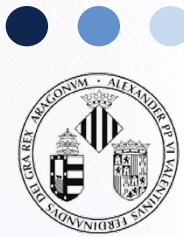
Bloque IV
Seguridad

Arquitectura de Redes de Computadores
2012-2013

Rafael Sebastian
Departamento de Informática
Escuela Técnica Superior de Ingenierías
Universitat de València



Índice de contenido

- **Seguridad WLAN**

Objetivos sección

- ☑ Entender el funcionamiento de WLAN y sus vulnerabilidades
- ☑ Comprender WEP y la seguridad básica
- ☑ Comprender los protocolos seguros para WLAN
- ☑ Comprender el funcionamiento de 802.1X y EAP

Material Multimedia

- ☑ Podcast iTunes Seguridad Red (hacking wifi)

Seguridad WLAN

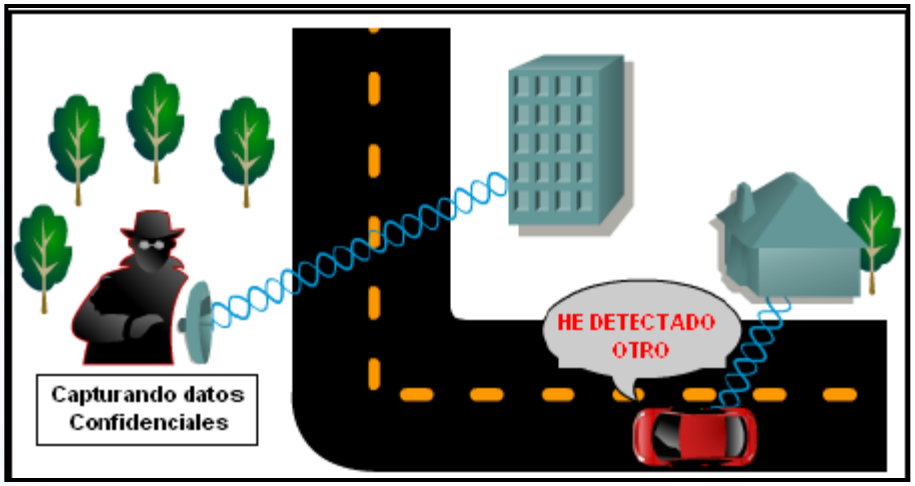
- **Problemas de seguridad WLAN**
  - Conceptos
  - Seguridad definida en IEEE 802.11
  - Vulnerabilidades en IEEE 802.11
- Protocolos seguridad WLAN

Objetivos principales

- Los principales objetivos de seguridad son
  - **Confidencialidad**: Queremos que los datos viajen seguros y solo puedan ser leídos por el destinatario
  - **Integridad:** No queremos que los datos que enviamos puedan ser modificados en el camino y reenviados como si fuesen nuestros
  - **Disponibilidad:** Queremos que la red esté disponible en todo momento y que no pueda ser atacada y desactivada

Confidencialidad

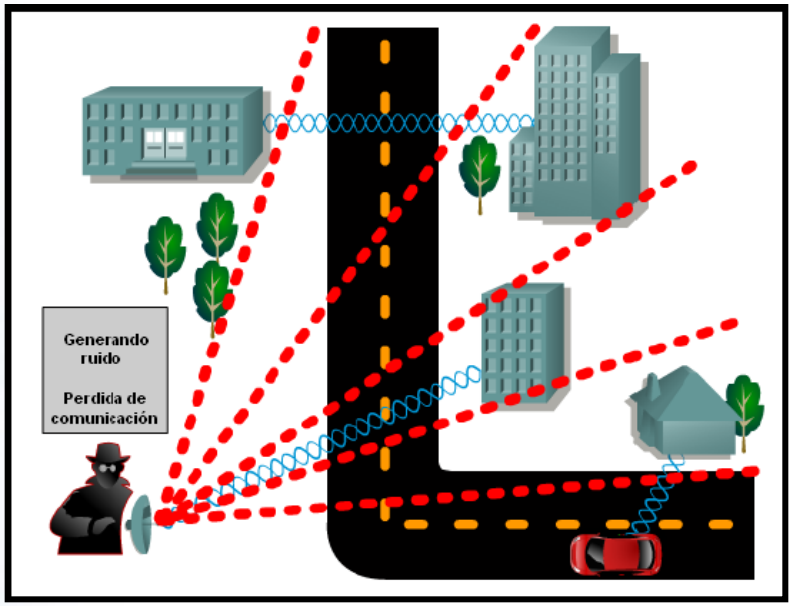
- Para garantizar la confidencialidad deberemos utilizar:
  - Sistemas criptográficos para que los mensajes solo puedan ser leídos por el emisor y el receptor
  - Debemos tratar de evitar que las difusiones de RF salgan de un área mínima necesaria, para evitar el wardriving y eavesdropping



- Es posible capturar el tráfico utilizando antenas que permiten el cambio a modo promiscuo.

Disponibilidad

- En un entorno donde las comunicaciones juegan un papel importante es necesario asegurar que la red esté siempre disponible
- Puesto que ahora el medio físico es el aire, es posible introducir interferencias que dejen la red fuera de servicio



Si generamos mucho ruido en la misma banda de frecuencias que se utiliza en una red, esta puede dejar de funcionar o funcionar al mínimo de sus posibilidades

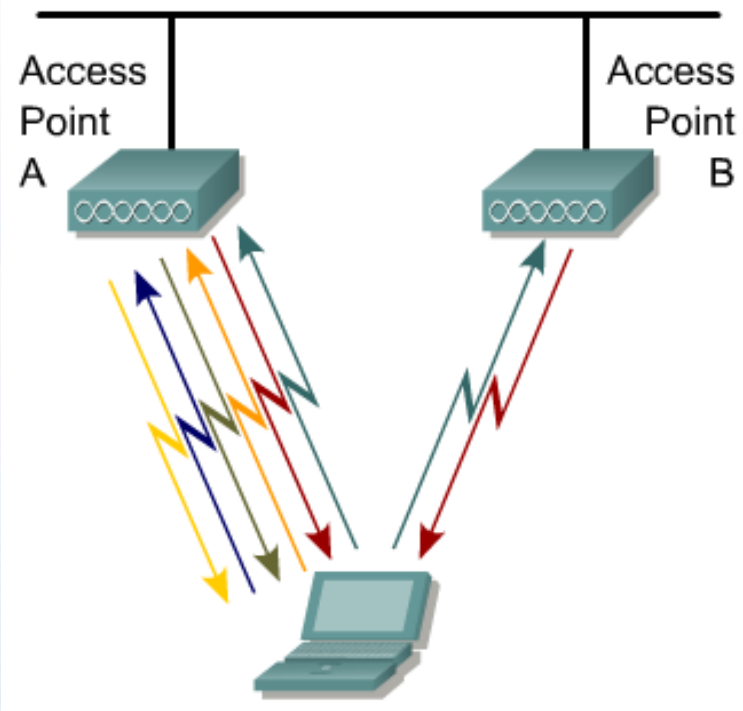
Mínimas garantías

- Vamos a exigir las siguientes medidas a los equipos de la red
  - **Autenticación:** Es decir, un usuario ha de demostrar que es quien dice ser, cuando quiera usar la red
  - **Autorización:** Indica que cosas puede hacer un usuario en la red, dependiendo de su nivel de privilegios
  - **Contabilización:** En determinados entornos es interesante tener un servidor que almacene las operaciones que está realizando un determinado usuario, para luego investigar acciones sospechosas

Autenticación en 802.11

- Las redes Wireless LAN requieren una seguridad especial debido a su naturaleza broadcast:
  - Identificación de usuario para control de acceso a recursos
  - Privacidad de los datos que se transmiten
- El SSID (Service Set Identifier) es una palabra de hasta 32 caracteres asociada a un Access Point o Bridge
  - El cliente de la red debe conocer el SSID para acceder a la red.
- Permite definir que direcciones MAC tienen permiso de acceso a la red
- No permite encriptación

Proceso de autenticación



La autenticación en 802.11 se realiza a nivel de equipo y no de usuario.

- El cliente envía una prueba broadcast
- El AP le contesta (el cliente la evalúa)
- El cliente envía una solicitud de autenticación al AP (A)
- El AP (A) confirma la autenticación
- El cliente solicita una asociación a AP(A)
- El AP (A) confirma la asociación y registra al cliente en su BD.

Mensajes *Beacon* o Sonda

Probe Request Frame

```
DLC: ----- DLC Header -----
DLC:
DLC: Frame 201 arrived at  10:18:59.4328; frame size is 39 (0027 hex) bytes.
DLC: Signal level                    = 100%
DLC: Channel                         = 1
DLC: Data rate                       = 2 ( 1.0 Megabits per second)
DLC:
DLC: Frame Control Field #1 = 40
DLC:                .... ..00 = 0x0 Protocol Version
DLC:                .... 00.. = 0x0 Management Frame
DLC:                0100 .... = 0x4 Probe request (Subtype)
DLC: Frame Control Field #2 = 00
DLC:                .... ...0 = Not to Distribution System
DLC:                .... ..0. = Not from Distribution System
DLC:                .... .0.. = Last fragment
DLC:                .... 0... = Not retry
DLC:                ...0 .... = Active Mode
DLC:                ..0. .... = No more data
DLC:                .0.. .... = Wired Equivalent Privacy is off
DLC:                0... .... = Not ordered
DLC: Duration                        = 0 (in microseconds)
DLC: Destination Address             = BROADCAST FFFFFFFFFFFF, Broadcast
DLC: Source Address                  = Station Aironet500292
DLC: Basic Service Set ID            = BROADCAST FFFFFFFFFFFF, Broadcast
DLC: Sequence Control                = 0x6F30
DLC: ...Sequence Number              = 0x6F3 (1779)
DLC: ...Fragment Number              = 0x0   (0)
DLC: Element ID                      = 0 (Service Set Identifier)
DLC: ...Length                       = 7 octet(s)
DLC: ...Service Set Identity         = "sliders"
```

Autenticación abierta

- Utiliza un algoritmo de autenticación NULA
- El punto de acceso permite cualquier acceso a la red
- Diseñado para sistemas como unidades de adquisición de código de barras o sistemas sencillos con poca potencia de CPU
- Define dos tipos de mensajes:
  - Solicitud de autenticación
  - Respuesta de autenticación

Autenticación abierta

Petición de autenticación

```
DLC: Destination Address              = Station Aironet31669C
DLC: Source Address                   = Station Aironet500292
DLC: Basic Service Set ID             = Aironet31669C
DLC: Sequence Control                 = 0x0A40
DLC: ...Sequence Number               = 0x0A4 (164)
DLC: ...Fragment Number               = 0x0   (0)
DLC: Authentication algorithm number  = 0 (Open System)
DLC: Authentication transaction sequence number = 1
DLC: Status code                      = 0 (Reserved)
```

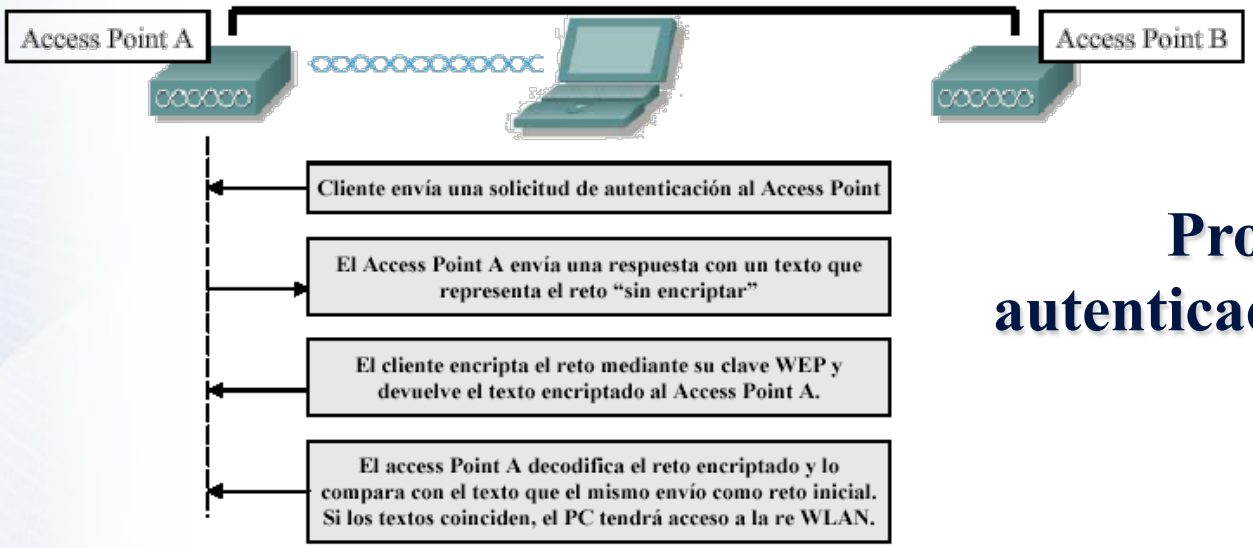
Respuesta de autenticación

```
DLC: Destination Address              = Station Aironet500292
DLC: Source Address                   = Station Aironet31669C
DLC: Basic Service Set ID             = Aironet31669C
DLC: Sequence Control                 = 0xED50
DLC: ...Sequence Number               = 0xED5 (3797)
DLC: ...Fragment Number               = 0x0   (0)
DLC: Authentication algorithm number  = 0 (Open System)
DLC: Authentication transaction sequence number = 2
DLC: Status code                      = 0 (Successful)
```

Uso de clave compartida

Wired Equivalent Privacy

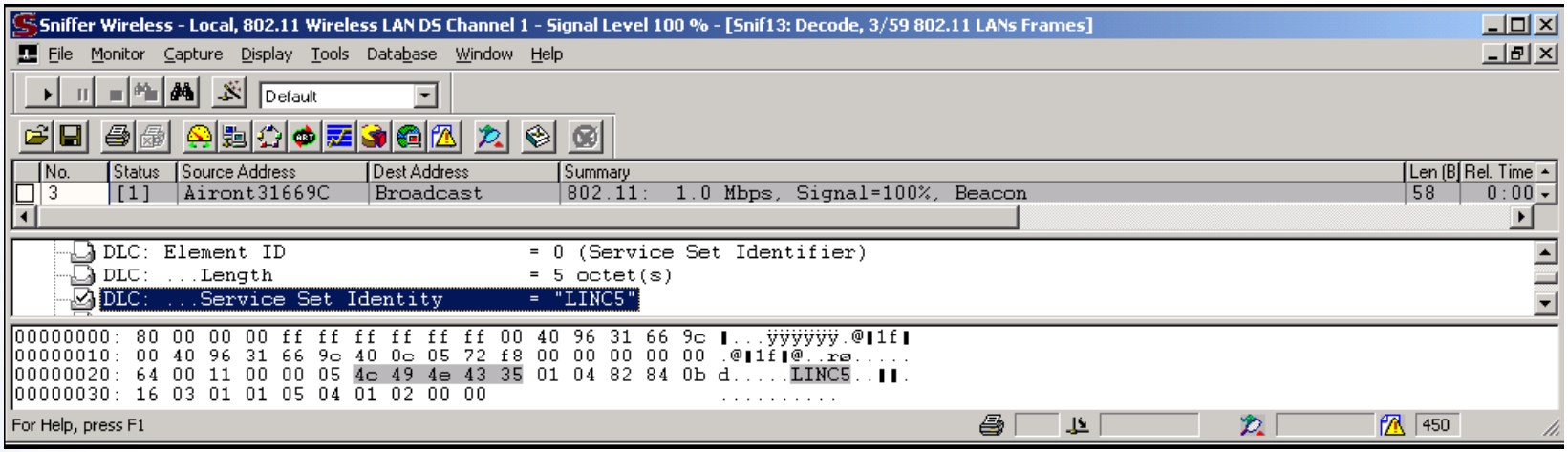
- Con WEP podemos utilizar una clave estática previamente acordada entre los Access Point y los clientes (siempre la misma).
- Esta clave se usa para **autenticar** a los clientes, es decir, para comprobar si se pueden conectar, y para **encriptar** los datos que se envían por la red.



Proceso de autenticación con WEP

Vulnerabilidades del SSID

- El SSID (Service Set Identifier) se publica en texto claro en los mensaje *beacon* emitidos por el punto de acceso.



- Algunos equipos permiten deshabilitar la opción de emitir *beacon* broadcast, pero el problema no se soluciona.

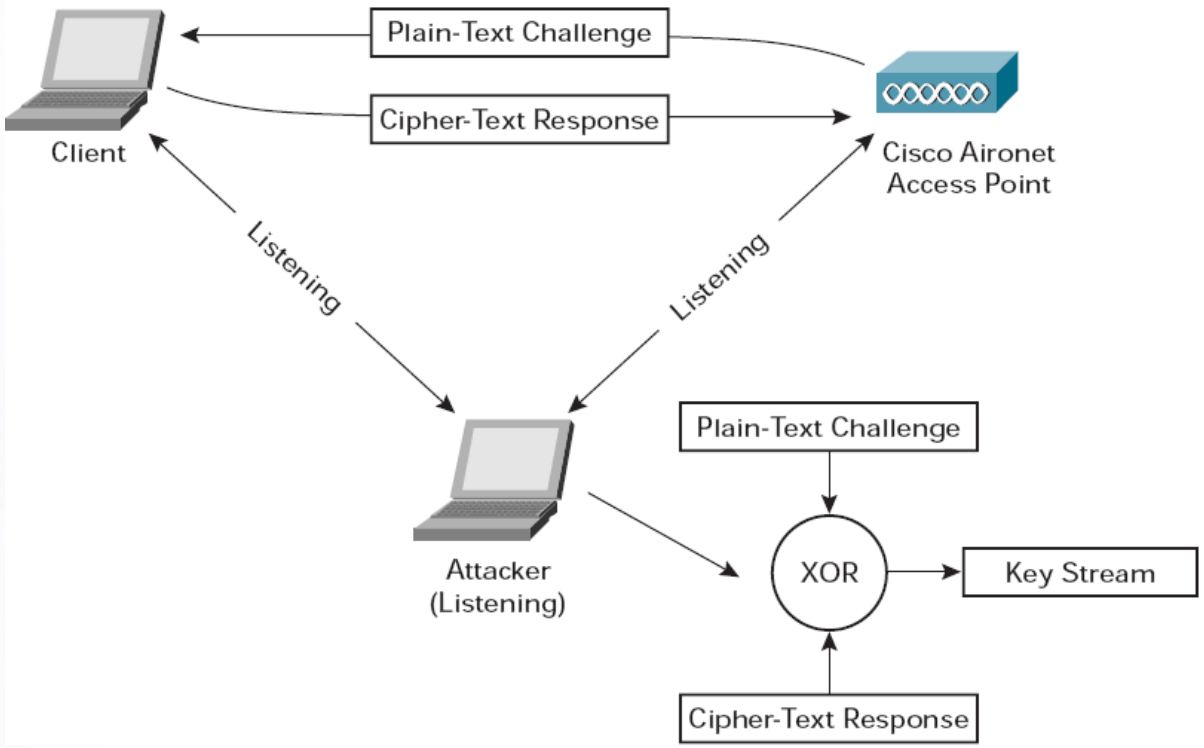
Notas sobre WEP

Problemas del sistema de autenticación compartida

- *Wired Equivalent Privacy (WEP)* es un algoritmo de encriptación basado en RC4 que se encarga de la autenticación y la encriptación de datos en WLAN
- El algoritmo RC4 fue secreto hasta septiembre de 1994.
- WEP es un algoritmo simple = **Débil**
- RC4 no fue implementado apropiadamente en WEP, lo cual creo un agujero de seguridad en redes WLAN
- El problema de WEP radica en su vector de inicialización fijo de 24 bits

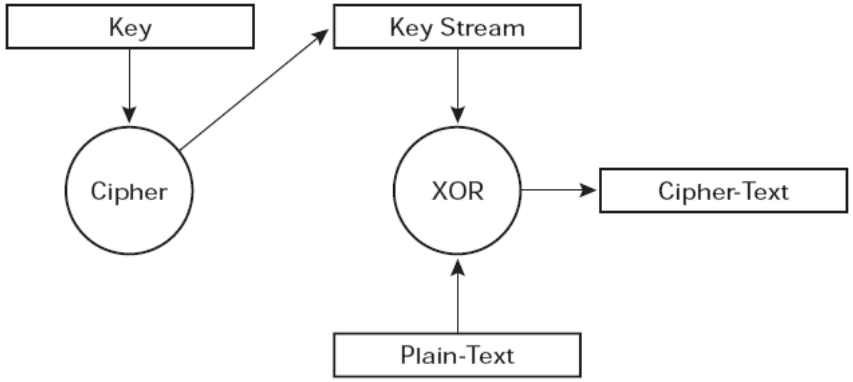
Derivación del Key Stream

- Ataques Man-in-the-middle

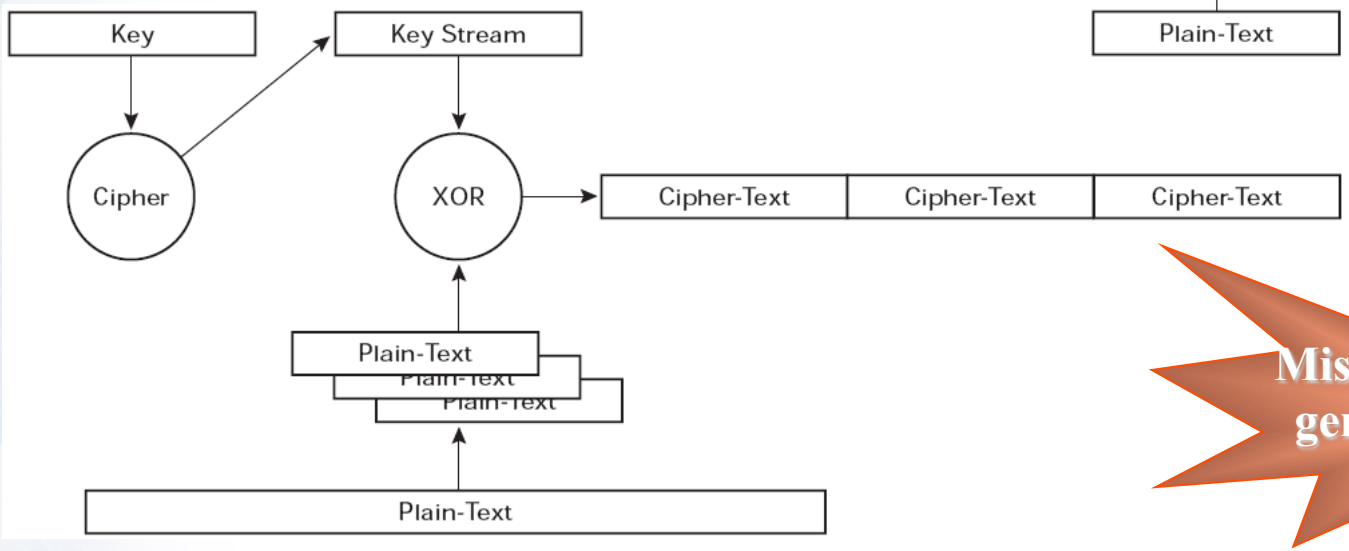


Funcionamiento Stream/ Block Ciphers

Stream Cipher



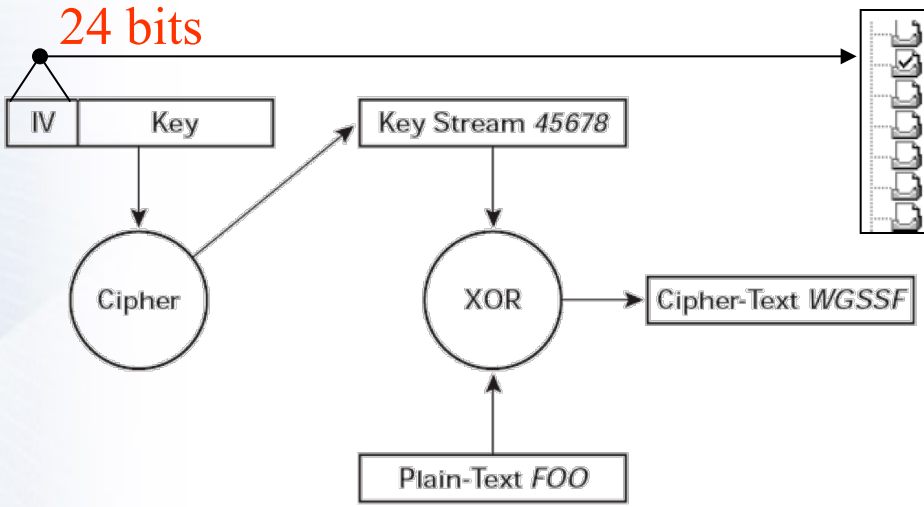
Block Cipher



Mismo texto siempre genera mismo texto cifrado!!

Solución a *stream blocks*

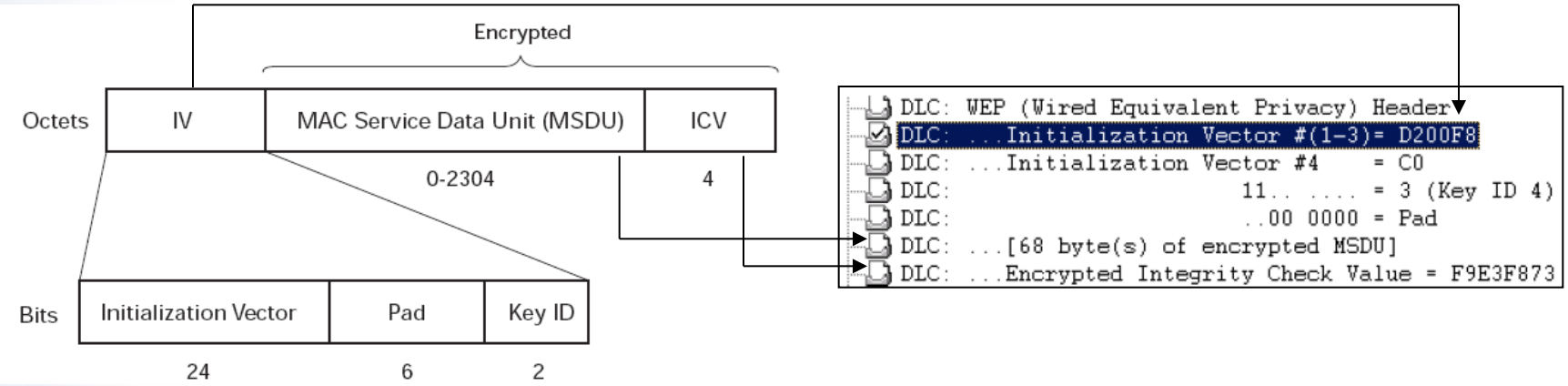
- Vector de inicialización
  - Se utilizan para alterar el Key Stream
  - Ahora antes de generar un Key Stream concatenamos un Vector de Inicialización (IV) a nuestra clave original
  - Por cada nuevo IV tendremos un nuevo key stream




```
DLC: WEP (Wired Equivalent Privacy) Header
DLC:  ...Initialization Vector #(1-3)= D200F8
DLC:  ...Initialization Vector #4     = C0
DLC:                          11.. .... = 3 (Key ID 4)
DLC:                          ..00 0000 = Pad
DLC:  ...[68 byte(s) of encrypted MSDU]
DLC:  ...Encrypted Integrity Check Value = F9E3F873
```

Solución a Stream Blocks

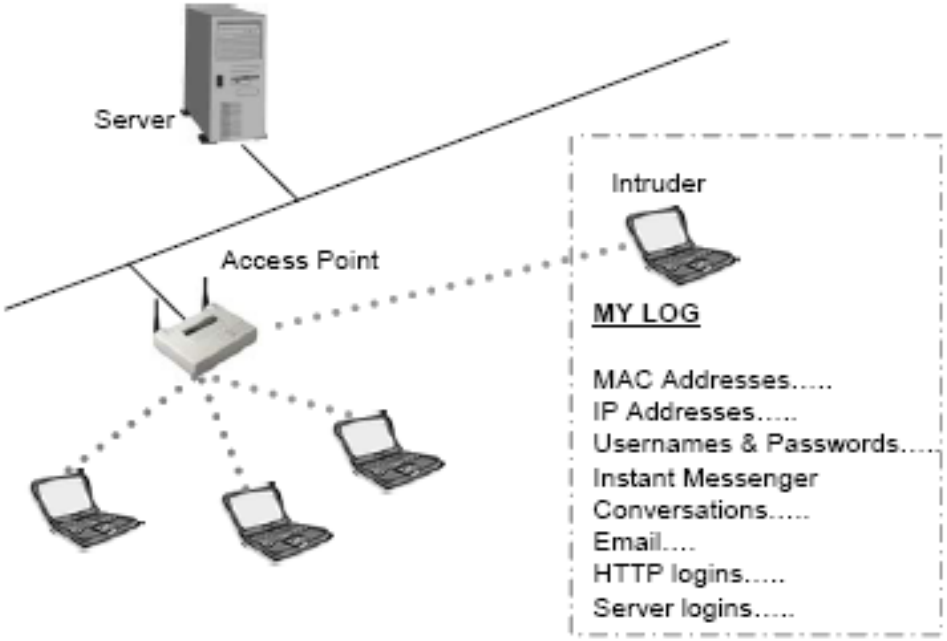
- Vector de inicialización
  - Los vectores de Inicialización tienen una longitud fija de 24 bits
  - Las claves WEP son de 40 bits (+24 IV = 64 bits) o de 104 bits (+ 24 IV = 128 bits)
  - El IV se envía en texto claro en cada mensaje



Ataques a sistema basados en WEP

- Un usuario capturando tráfico podría desencriptar la clave WEP en unas pocas horas. Además existe software que realiza esta tarea!!

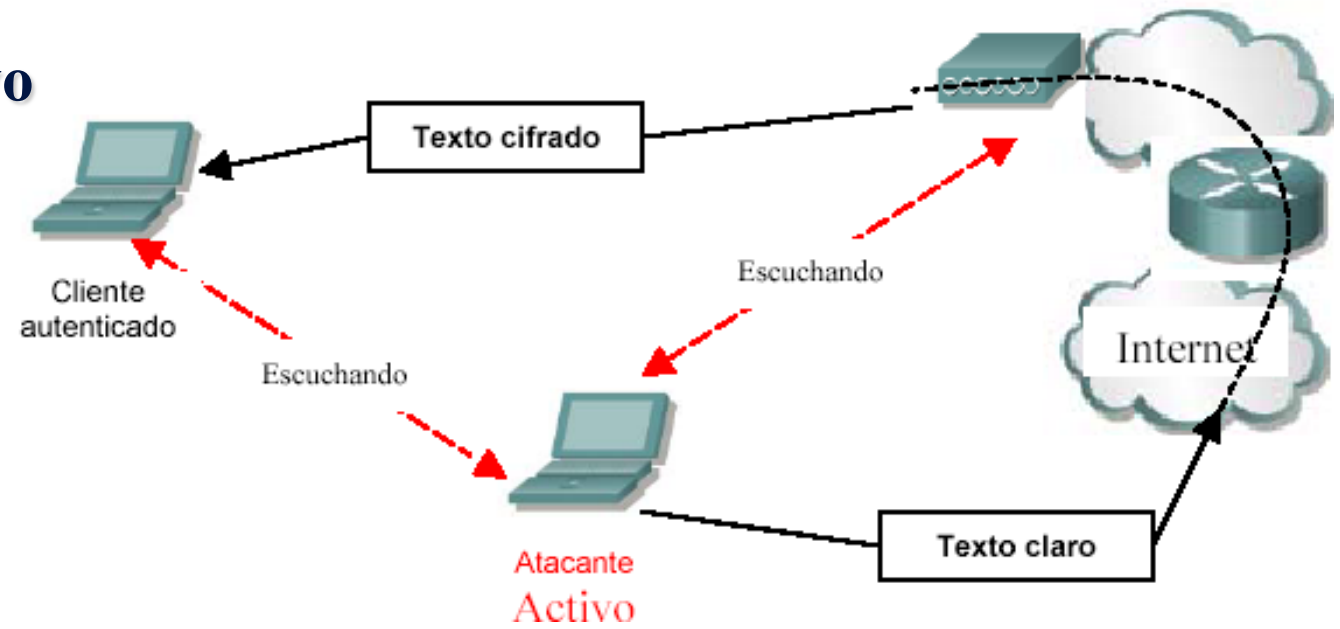
Ataque Pasivo



Ataque pasivo:
Fluhrer, Mantin y Samir (Agosto 2001)

Ataques a sistema basados en WEP

Ataque Activo



Publicado:
Nikita Borisov,
Ian Goldberg
David Wagner
(Febrero 2001)

Ataques a sistema basados en WEP

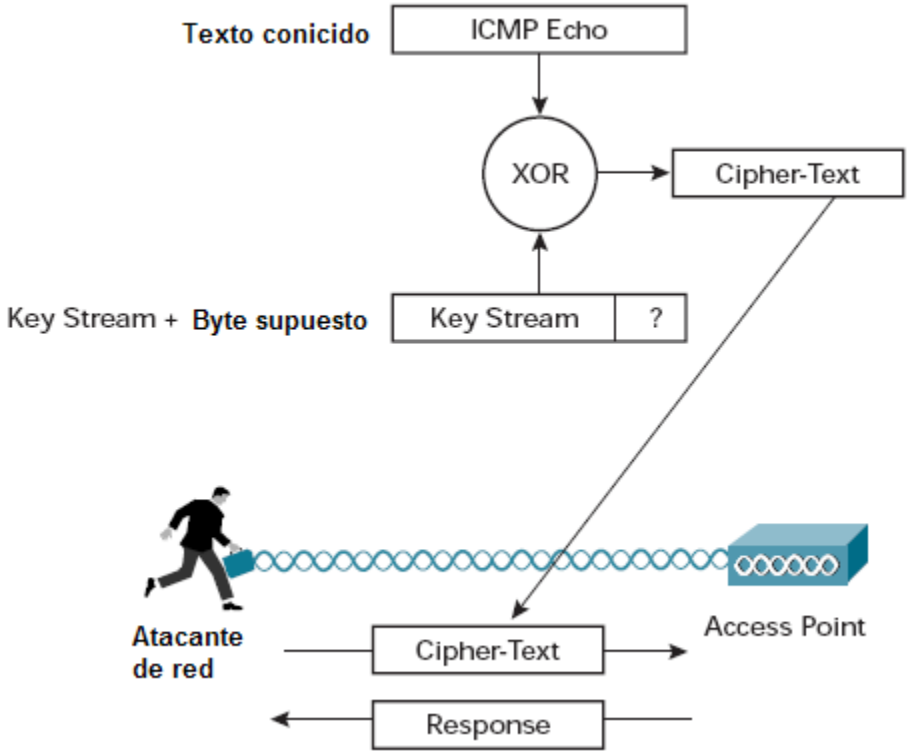
- Ataques Activos: Repetición del IV
  1. Enviamos a un cliente de la red Wireless un mensaje que conocemos (ej: Correo electrónico)
  2. El atacante hace *sniffing* de la red wireless buscando el mensaje cifrado correspondiente al mensaje que ha enviado
  3. El atacante obtiene el *key stream* utilizado
  4. El atacante puede "engordar" el *key stream* obtenido usando la misma combinación IV/WEP

Ataques a sistema basados en WEP

Ataques Activos: Repetición del IV

"Engordando" el Key stream

1. El atacante genera un mensaje un byte más grande que el *key stream* obtenido. (ICMP Echo)
2. Supone el valor del byte extra (al azar).
3. Envía mensaje y espera ECHO. Solo lo recibirá si ha acertado el valor (256 posibilidades, 2^8).



Ataques a sistema basados en WEP

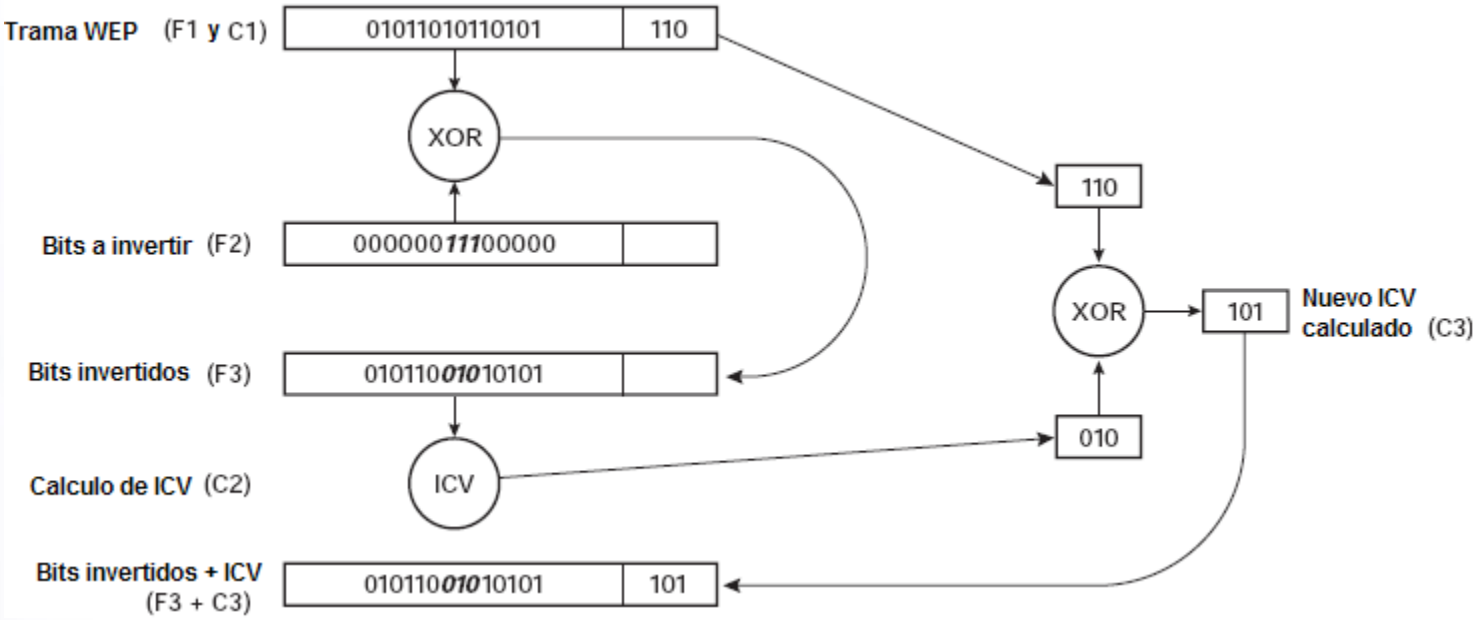
- Ataques Activos: Bit flipping (inversión de bit)

1. El atacante captura una trama encriptada de la red wireless
2. El atacante invierte aleatoriamente algunos bits del Payload de capa 3 (dentro de la trama)
3. El atacante ajusta el valor del campo ICV (para que sea correcto)
4. El atacante transmite la trama modificada
5. El cliente o AP recibe la trama y comprueba el ICV que será correcto
6. Desencapsula el paquete de dentro de la trama y detecta fallos en el checksum (puesto que hemos invertido algunos bits)
7. El receptor envía un paquete de error predecible que puede usar para obtener el key stream y realizar los pasos del ataque anterior

Ataques a sistema basados en WEP

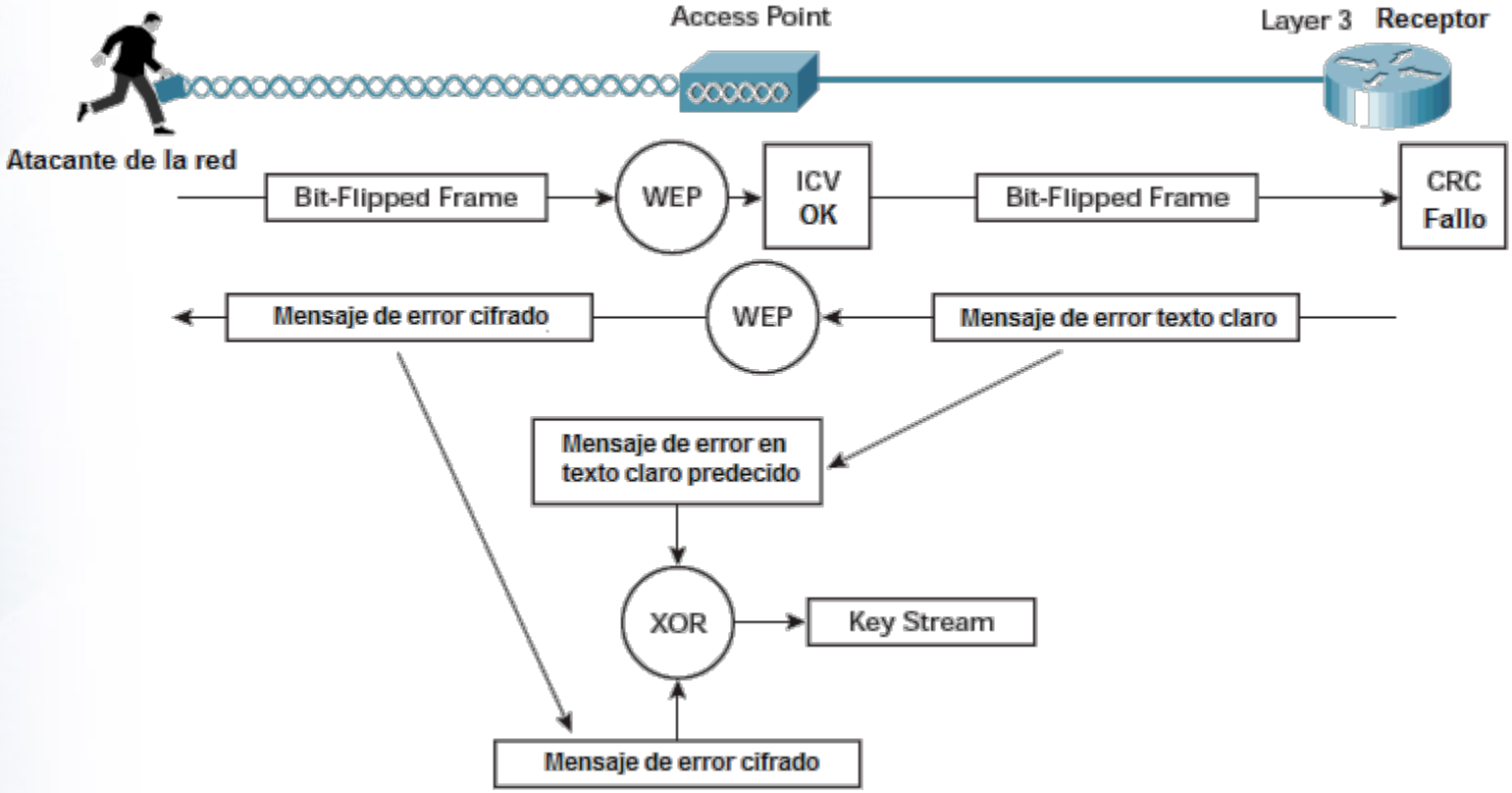
Ejemplo de ataque de bit-flipping

Cálculo del nuevo ICV



Ataques a sistema basados en WEP

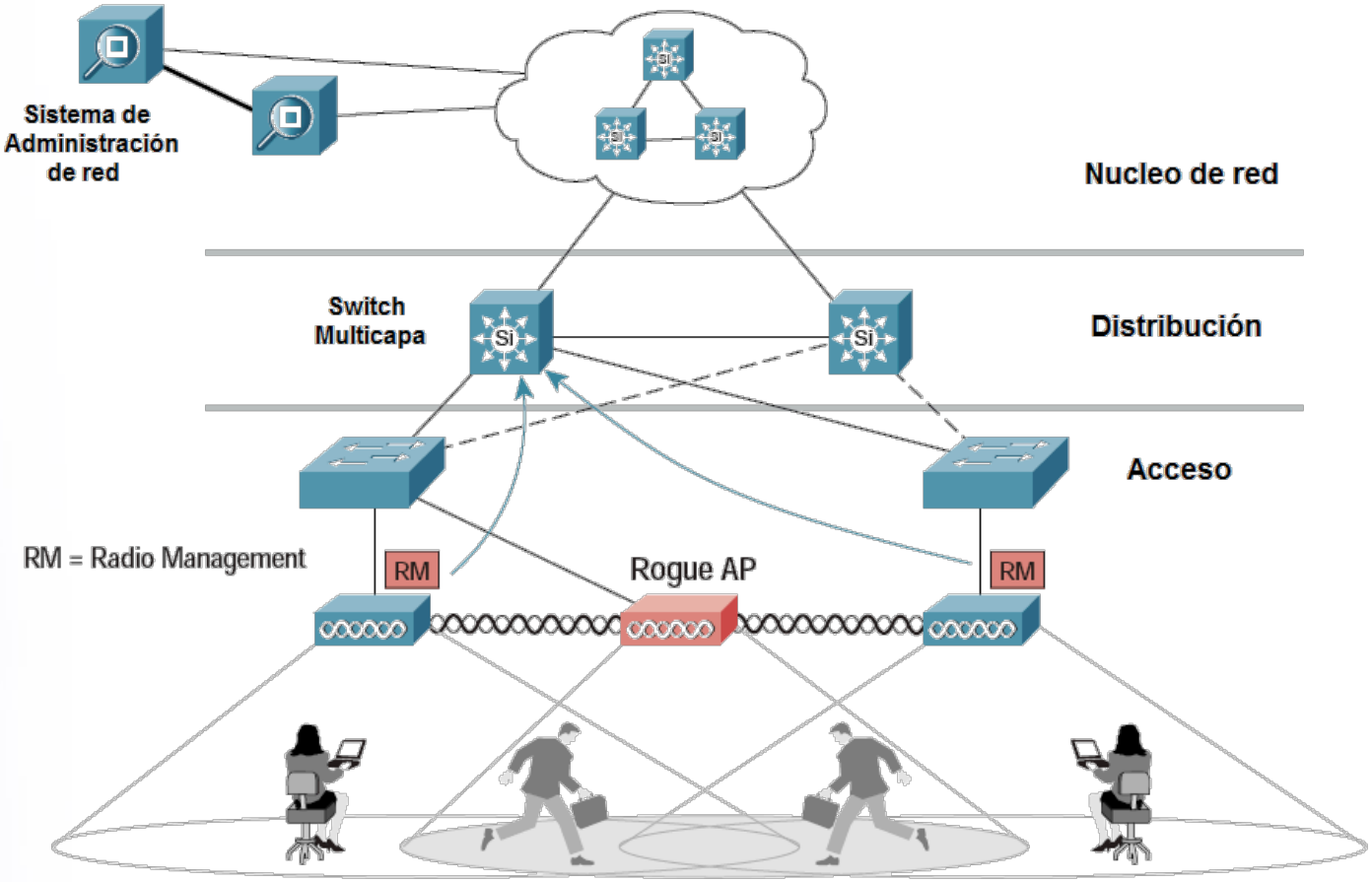
Ejemplo de ataque de bit-flipping



Consecuencias de WEP

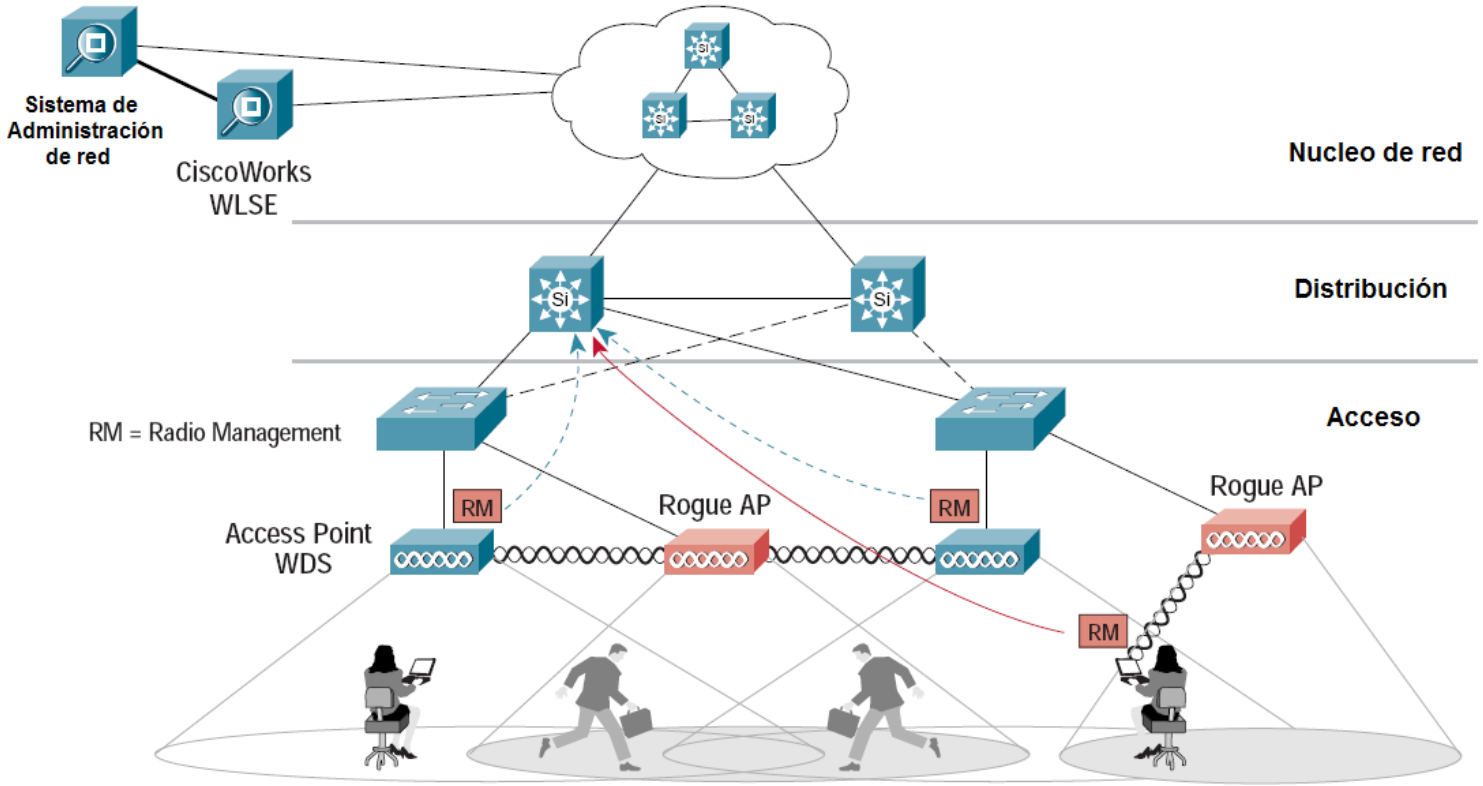
- No define ningún mecanismo de administración de claves:
  - Las claves deben configurarse estáticamente
  - Si la clave secreta es descifrada → El administrador debe cambiarla en todos los PCs y APs.
  - La autenticación es por equipo y no por usuario. Si algún PC o equipo es robado la red puede ser comprometida
- Si algún empleado con conocimiento de la clave es despedido, la clave debería de ser modificada
- La clave solo sirve para identificar el acceso de una máquina a la red, pero no permite diferenciar entre diferentes usuarios

Rogue Access Points



Solución a Rogue Access Points

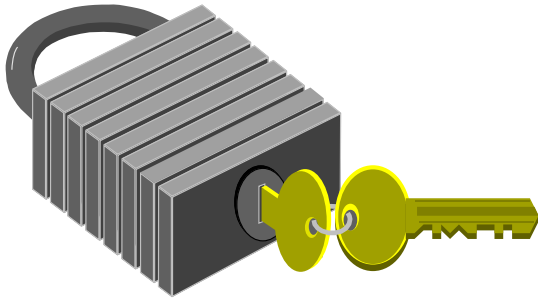
- Solución propietaria



Seguridad WLAN

- Problema de seguridad WLAN
- **Protocolos seguridad WLAN**

Mejoras al estándar

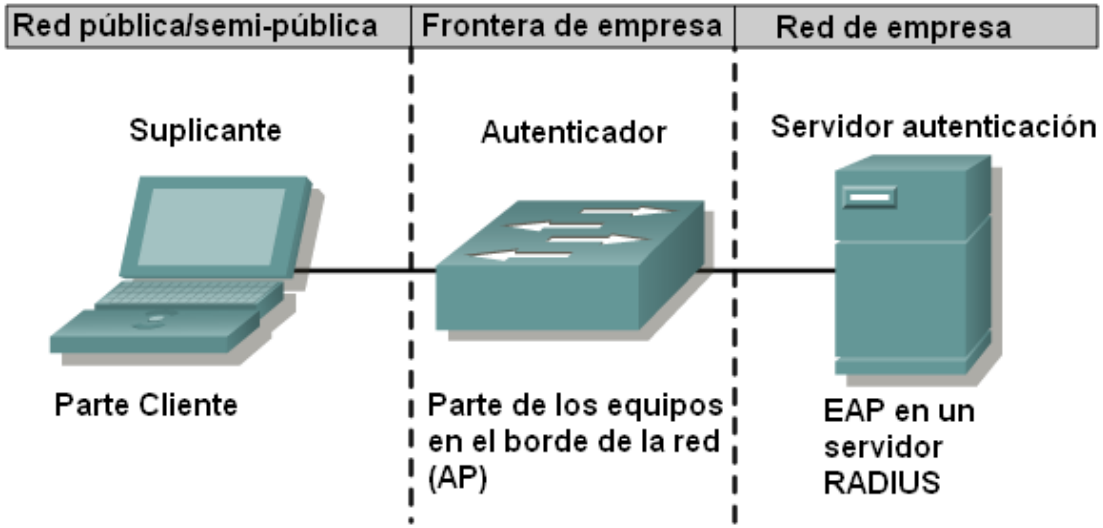


- Nuevos requisitos
  - Entorno de autenticación
  - Algoritmo de autenticación
  - Privacidad de datos mediante nuevos algoritmos de encriptación
- Cambio de arquitectura
  - 802.1X → Arquitectura de autenticación a nivel de capa de enlace
  - EAP → Algoritmo de autenticación flexible
- Mejoras a WEP
  - TKIP o AES → Algoritmos robustos de encriptación de datos

Entorno para seguridad WLAN

- Estándar 802.1X
  - El estándar 802.1X define un marco de autenticación, dentro de una arquitectura de seguridad
  - Define tres entidades: El suplicante, el autenticador y el servidor de autenticación.

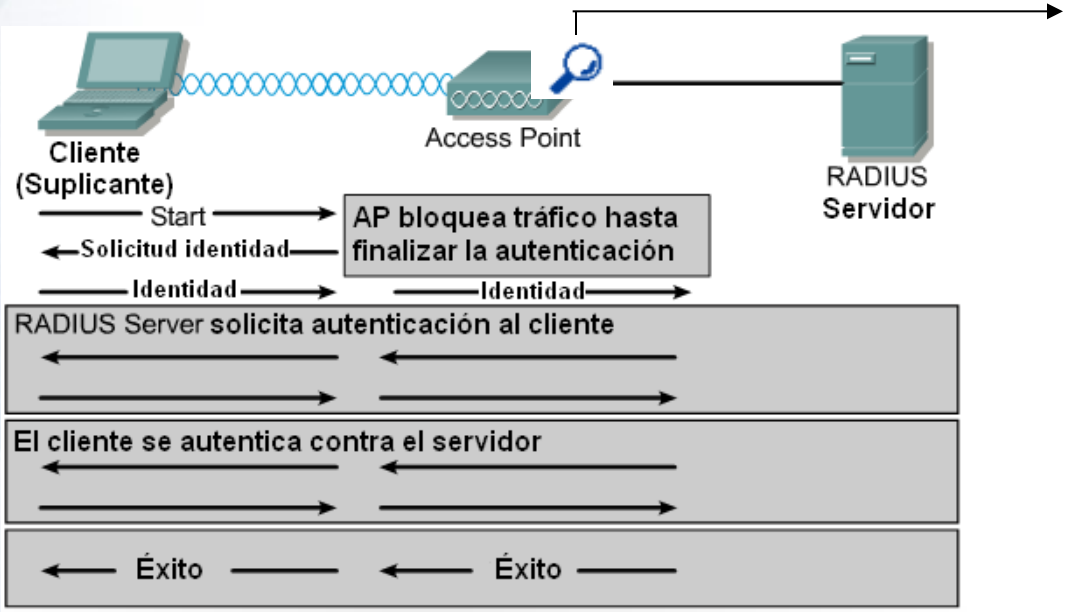
Entidades

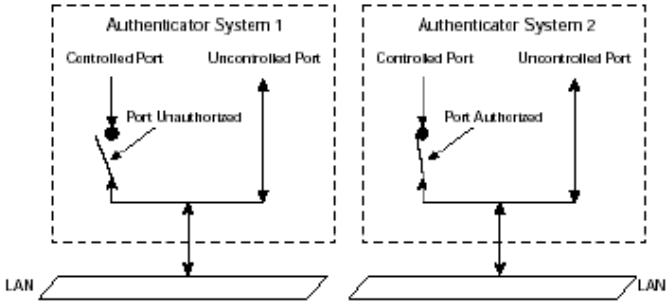


Funcionamiento de 802.1X

- Sobre la arquitectura 802.1X podemos utilizar diferentes protocolos basados en EAP (Extensible Autentication Protocol)
- El intercambio de mensajes de los protocolos basados en EAP es muy similar

Proceso de autenticación



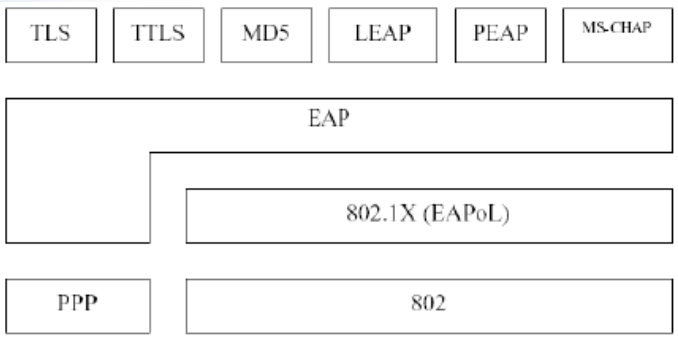


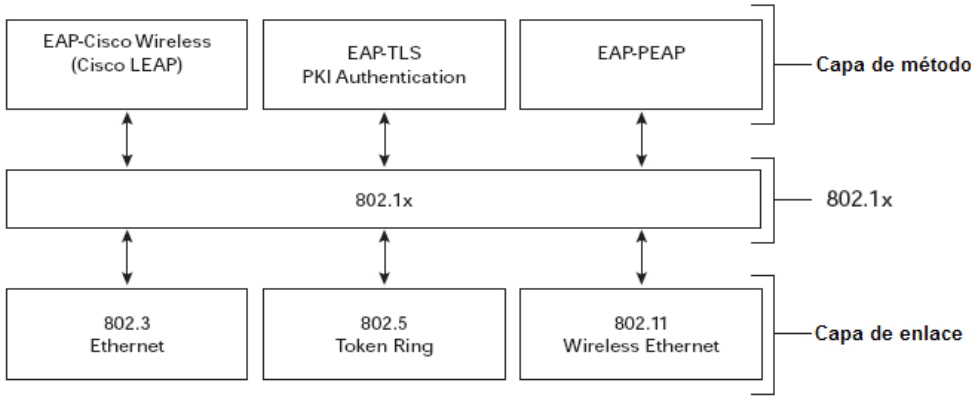
Puerto controlado: Solo puede ser usando por clientes autenticados.

Protocolos para seguridad WLAN

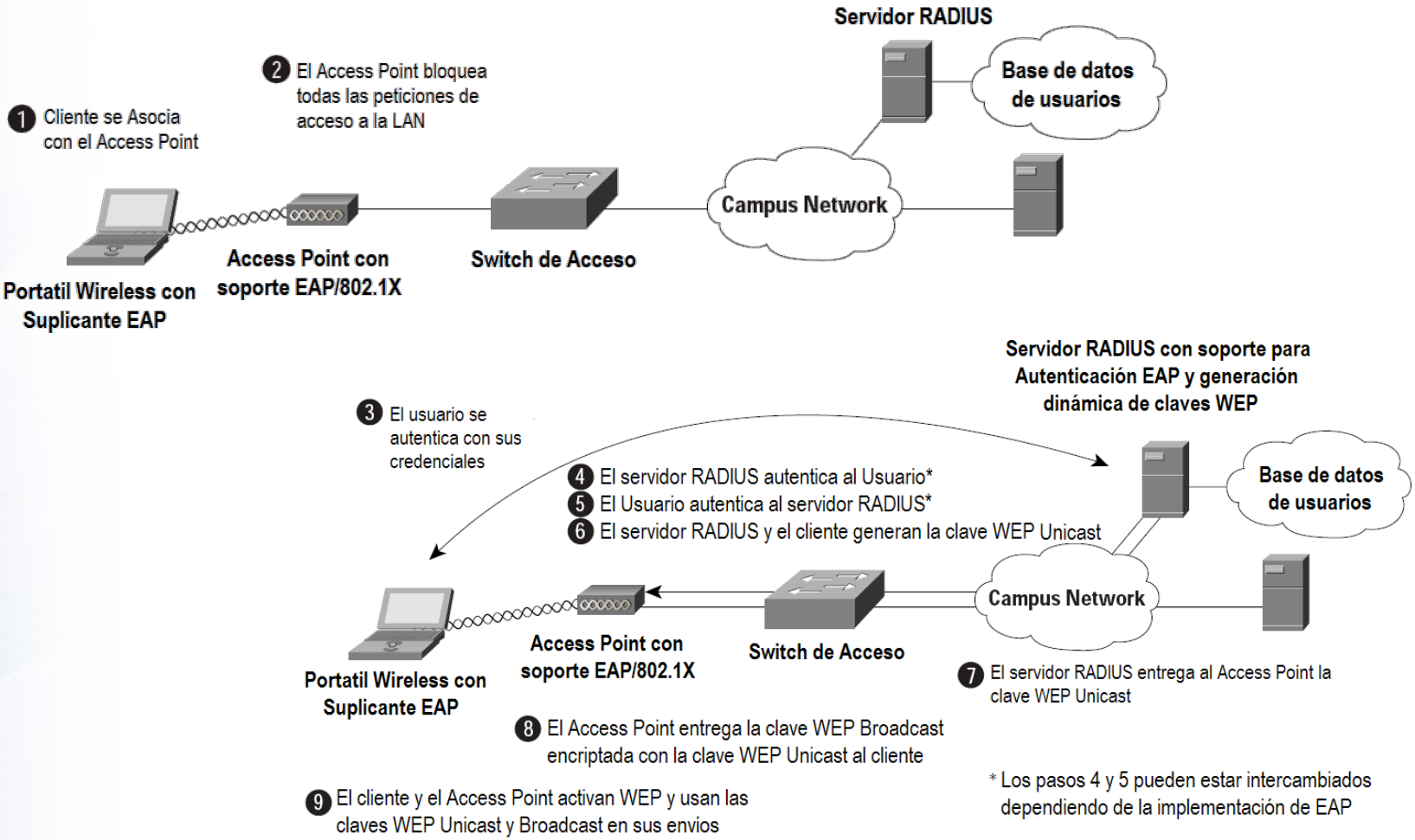
- Extended Authentication Protocol (EAP)
  - EAP es un protocolo que permite la autenticación centralizada de usuarios cuando se combina con entornos como 802.1X
  - Algunas mejoras son: la autenticación mutua, la administración centralizada y la creación de claves WEP dinámicas

Estructura de capas





Funcionamiento de EAP



Algunas implementaciones de EAP

	LEAP	EAP-TLS	EAP-PEAP
Server Authentication	Password	Certificates/Public Key Infrastructure (Certs/PKI)	Certs/PKI
Client Authentication	Password	Certs/PKI	Password[1]
Single Sign On	Si	Si	No[2]
Vulnerable to Password Attack	No[3]	No	No
OTP/LDAP Support	No	N/A	Si
Additional Infrastructure	No	Si /Certificate Authority (CA)	Si /CA

[1] Los esquemas de autenticación no están limitidos, este es el que está disponible actualmente
[2] MS suplicante nativo soporta SSo con EAP-MS-CHAP v2
[3] Requiere passwords fuertes

Ventajas de 802.1X y EAP

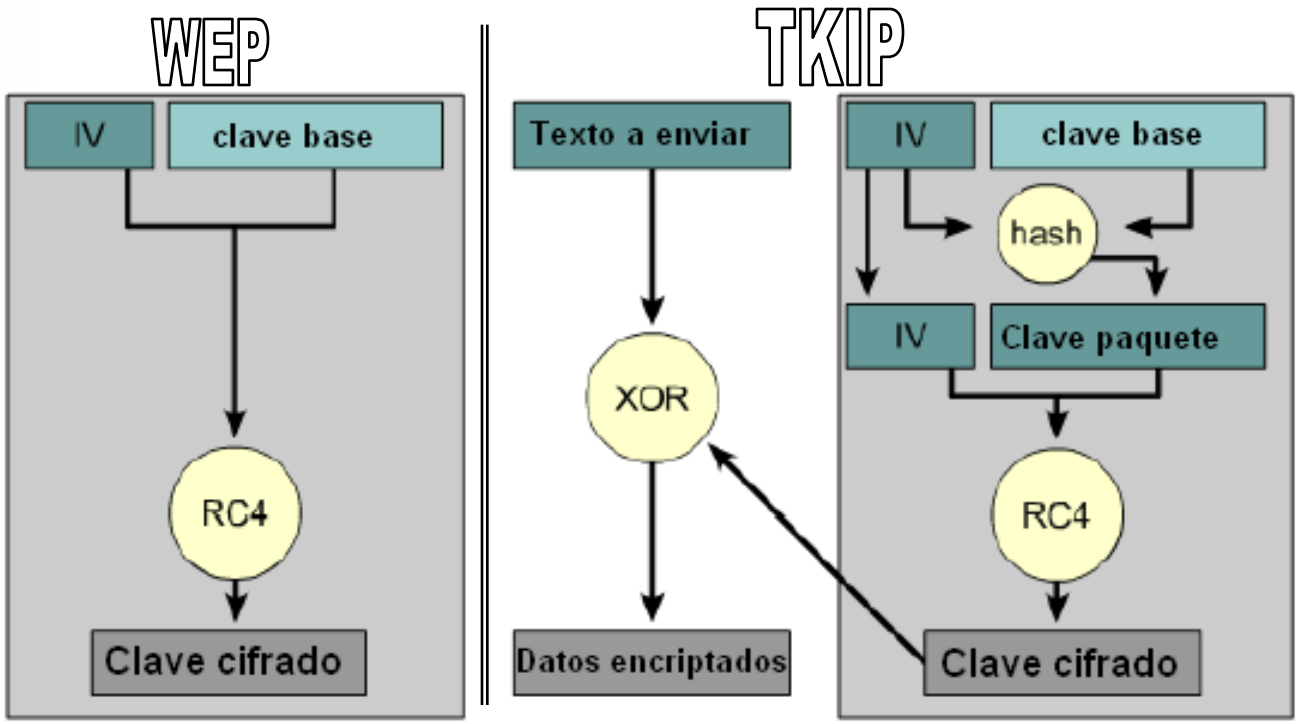
- Permiten autenticación dependiendo de qué usuario quiera conectarse, y no de que PC (ya que podrían usarlo diferentes personas)
- Permiten autenticación mutua, es decir, se autentican tanto el cliente como el servidor
- Es posible disponer de una base de datos centralizada en el servidor RADIUS con diferentes passwd, y no una base de datos diferente en cada AP
- Permite el uso de claves individualizadas y no de una única clave WEP compartida por todos

WPA, mejoras criptográficas

- Otro punto de vista para mejorar las debilidades de WEP es utilizar mejoras en el proceso criptográfico
- En el estándar 802.11i, se recogen dos alternativas:
  - TKIP: Temporal Key Integrity Protocol
  - AES: Advanced Encryption Standard
- Ambos protocolos impiden que un posible atacante pueda descifrar las claves que están utilizando en AP y el cliente para comunicarse

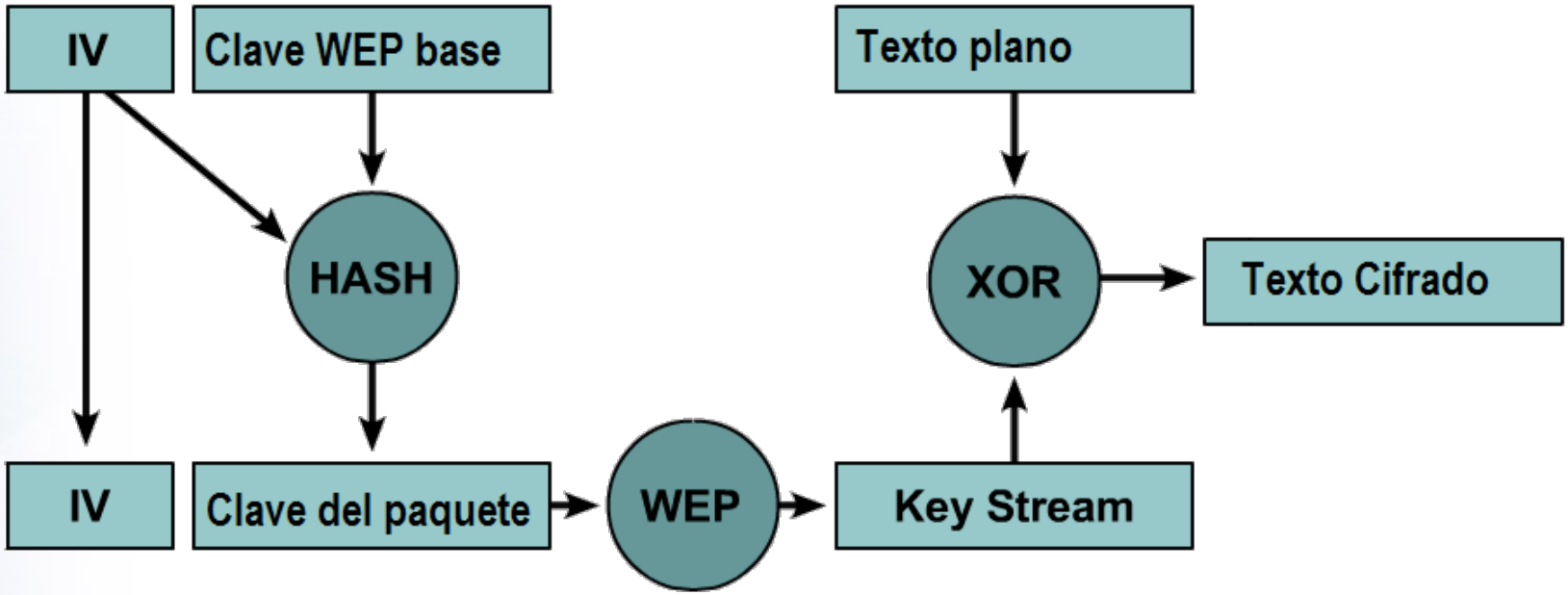
Protocolo TKIP

- TKIP se basa en WEP pero introduce una serie de mejoras que le dan más robustez



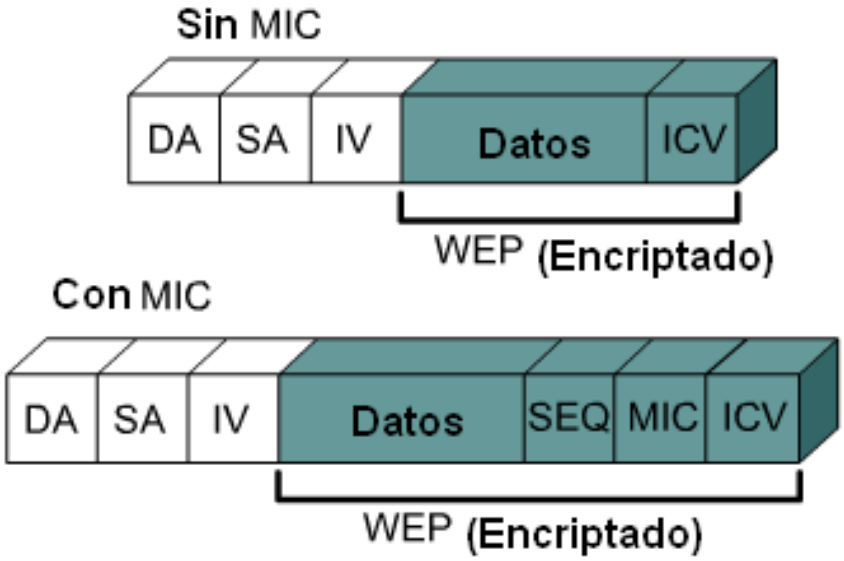
Protocolo TKIP

- Una mejora importante es que se genera una nueva clave WEP por paquete enviado



Protocolo TKIP

- MIC: Message Integrity Check
  - Soluciona el problema de bit-flipping que aparecía por el uso de un CRC-32 en el campo ICV de WEP

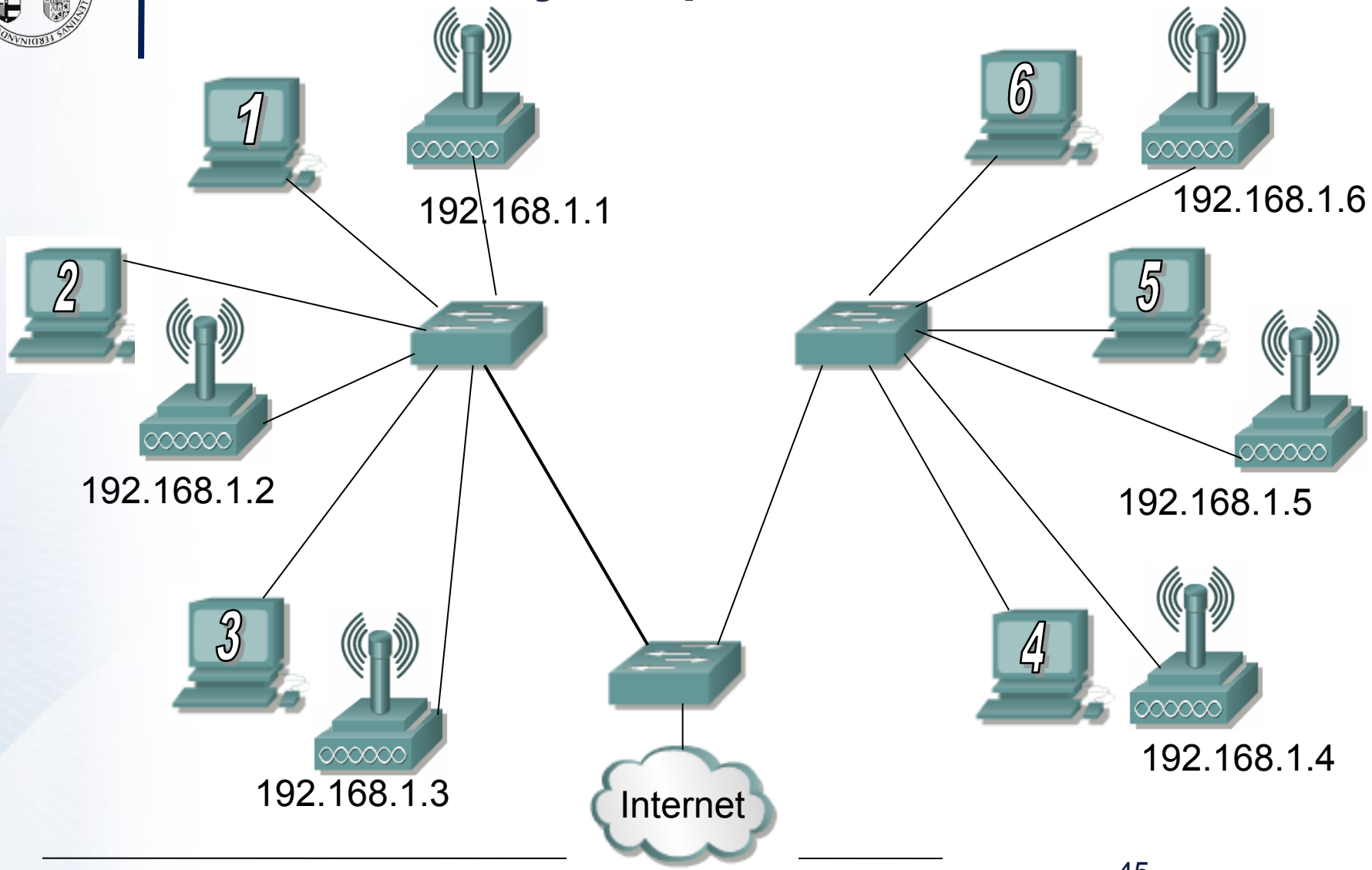


MIC

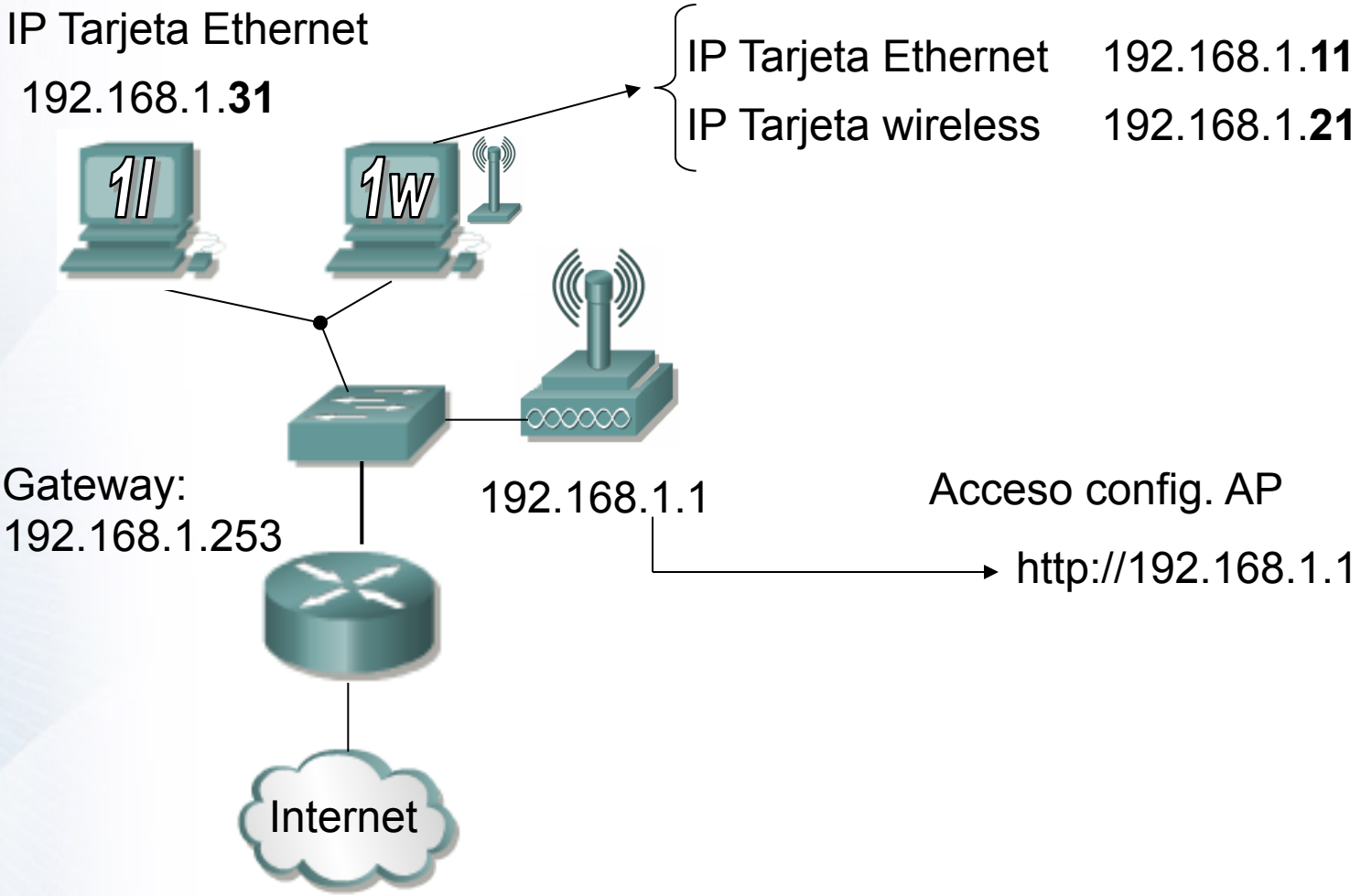
Protocolo AES

- El protocolo AES (Advanced Encryption Standard) se desmarca totalmente de WEP y presenta una alternativa radicalmente diferente y más potente
- El protocolo AES tiene un nivel de encriptación muy potente y se está pensando en su integración en todas las áreas de las telecomunicaciones por algunos gobiernos
- Soporta claves de hasta 256 bits, claves dinámicas y soporta MIC

Red de Ejemplo



Configuración para AP1



Implementación seguridad

- Implementación de una red wireless segura
  - Configuración de un servidor RADIUS.
  - Implementación basada en EAP/TLS.
  - Configuración de los equipos wireless.

Servidor de autenticación

Usaremos un programa emulador de sistemas

- Microsoft virtual PC (software libre)

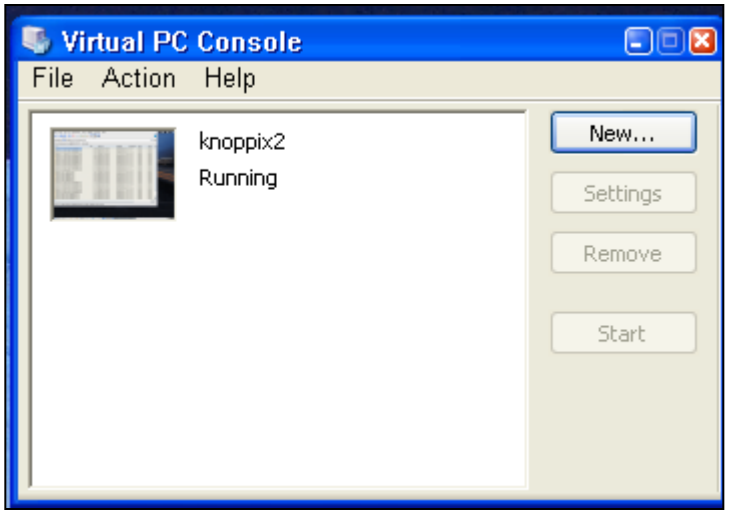
SERVIDOR DE AUTENTICACIÓN

- Plataforma LINUX: Linux SUSE 10.0 (software libre)
  1. Servidor RADIUS: FreeRadius
  2. OpenSSL: Certificados digitales X.509

Microsoft Virtual PC

Ejecución de Linux

→ Microsoft virtual PC (software libre)



Configuración IP grupo1

Cada grupo necesitará 3 direcciones:

1. PC con windows.
2. PC con Linux.
3. Punto de acceso.

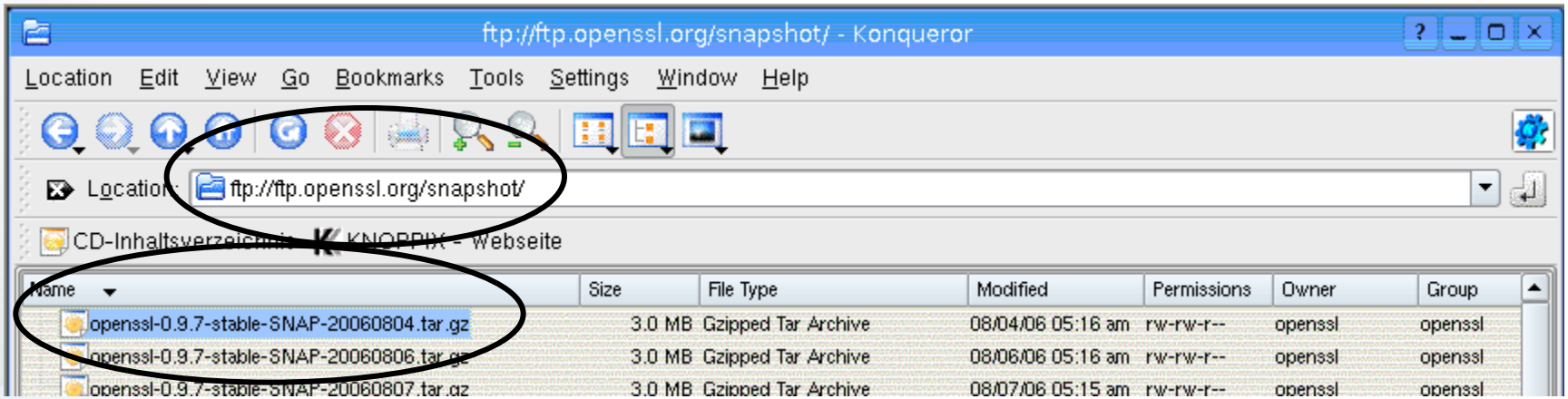
PC Virtual con Linux
Servidor RADIUS (Servidor de autenticación)
IP: 192.168.1.31 / Mascara: 255.255.255.0
Gateway: 192.168.1.253 / DNS: 80.58.0.33

PC con Windows
Cliente (Suplicante) (wireless)
IP: 192.168.1.11

Punto de Acceso
Autenticador
IP: 192.168.1.1

EJEMPLO CONFIGURACIÓN GRUPO 1

Obtención de openSSL



Abre una shell para ejecutar comandos

Nos permite explorar diferentes directorios del PC

Instalación de openSSL

Secuencia de comandos

- Creamos el directorio para el programa:
  - mkdir -p /usr/src/802/openssl
- Movemos al nuevo directorio el fichero descargado:
  - mv *openssl-0.9.7-SNAP. .. .tar.gz* /usr/src/802/openssl
  - cd /usr/src/802/openssl
- Descomprimimos el fichero:
  - tar zxvf openssl-0.9.7-SNAP … .tar.gz
  - cd openssl-SNAP-0.9.7.-…
- Configuramos directorios y distribuimos ficheros:
  - ./config shared --prefix=/usr/local/openssl
- Compilamos el software:
  - make
- Instalamos el software:
  - make install

Creación de CA y certificados digitales

Secuencia de tareas

- **Creación de una entidad de certificación CA.**
- Creación de certificados digitales
  - Clave privada.
  - Clave pública.

Creación de CA

Entidades de Certificación CA

- Un CA es un sistema que actúa como raíz en una infraestructura de clave pública.
- Tiene la autoridad para aprobar/desaprobar certificados digitales.
- Lo utilizaremos para crear los certificados X.509 para los usuarios de nuestra red wireless.
- Utilizaremos OpenSSL para crear nuestro propio CA y así ser capaces de extender certificados digitales.

Creación de CA

Creación de nuestro CA

- Editar el fichero openssl.cnf (en /etc/ssl)

1

Cambiar el directorio por defecto en el que se almacenan los certificados

```
[ CA_default ]
dir = ./UVCA  # Directorio de Certificados raiz
```

2

Cambiar el Pais/Region (más abajo en fichero openssl.cnf)

countryName_default = **ES**
stateOrProvinceName_default = **Valencia**
0.organizationName_default = **UV S.A.**

Creación de CA

Creación de nuestro CA

- Editar el fichero CA.sh (en /usr/share/ssl/misc/)

3 Cambiar el directorio por defecto para que coincida con el indicado antes

CATOP = ./UVCA

4 Crear los certificados RAIZ del CA:
1) Volver al directorio **/etc/ssl**
2) Ejecutar **/usr/share/ssl/misc/CA.sh -newca**

Creación de CA

Creación de nuestro CA

- Crear nuevo fichero ***xpextensions*** (en /etc/ssl)

5 El contenido del fichero será el siguiente

```
[ xpclient_ext]
extendedKeyUsage = 1.3.6.1.5.5.7.3.2
[ xpserver_ext ]
extendedKeyUsage = 1.3.6.1.5.5.7.3.1
```

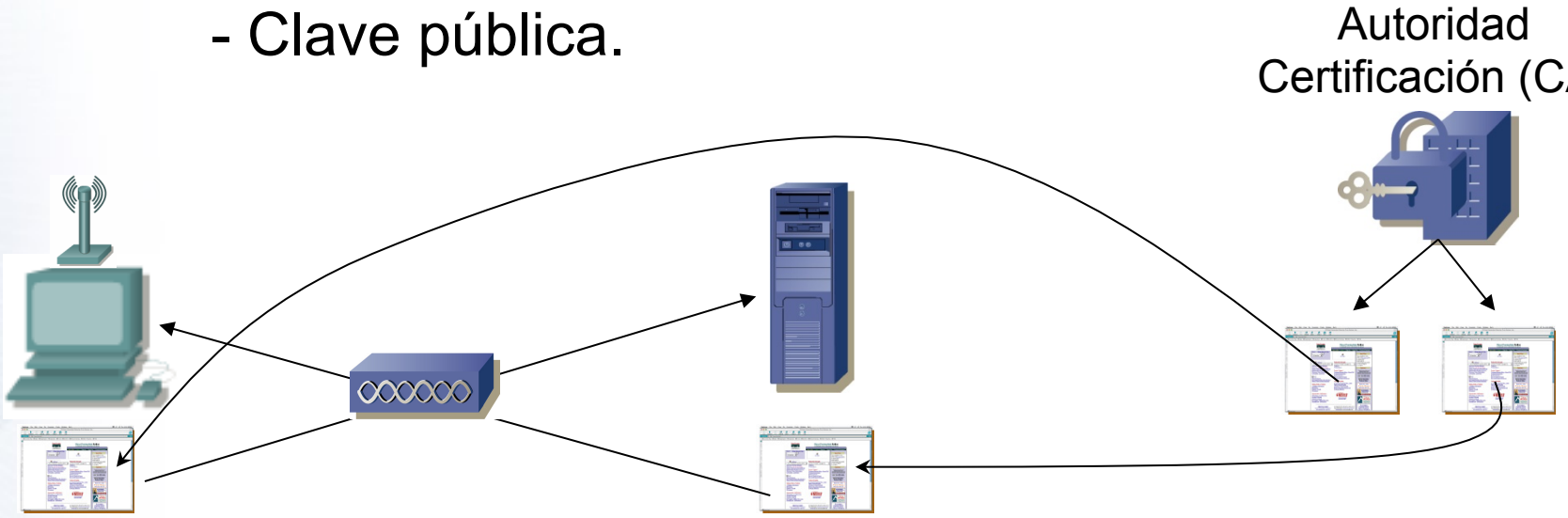
6 Modificar el formato de nuestro CA para exportar después a XP

```
openssl pkcs12 –export – in UVCA/cacert.pem –inkey
    UVCA/private/cakey.pem –out rootUV.p12 –cacerts
openssl pkcs12 -in rootUV.p12 -out rootUV.pem
openssl x509 -inform PEM -outform DER -in rootUV.pem -out rootUV.der
```

Creación de certificados

Secuencia de tareas

- Creación de una entidad de certificación CA.
- **Creación de certificados digitales**
  - Clave privada.
  - Clave pública.



Creación de certificados

- Crear un nuevo certificado sin firmar para RADIUS y para cliente

1

El comando que usaremos será el mismo. Usaremos nombre diferentes para diferenciar los certificados:

SERVIDOR

$ openssl req -new -nodes -keyout server_key.pem -out server_req.pem -days 730 -config ./openssl.cnf

CLIENTE

$ openssl req -new -nodes -keyout client_key.pem -out client_req.pem -days 730 -config ./openssl.cnf

Creación de certificados

Firma de la solicitud de certificado Server

- Vamos a firmar el certificado del servidor con el CA de UV

2

Comando a utilizar: (debemos estar en directorio ***/etc/ssl***)

SERVIDOR

$ openssl ca -config ./openssl.cnf -policy policy_anything -out server_cert.pem -extensions xpserver_ext -extfile ./xpextensions -infiles ./server_req.pem

3

Vamos a combinar el certificado y la clave privada del servidor en un solo fichero:

1º Editamos ***server_cert.pem*** y eliminamos todo lo que hay antes de:

----- BEGIN CERTIFICATE ----

2º Ejecutamos:

$ cat server_key.pem server.cert.pem > server_keycert.pem

Creación de certificados

Firma de la solicitud de certificado Client

- Vamos a firmar el certificado del servidor con el CA de UV

4 Comando a utilizar: (debemos estar en directorio ***/etc/ssl***)

CLIENTE

$ openssl ca -config ./openssl.cnf -policy policy_anything -out client_cert.pem -extensions xpclient_ext -extfile ./xpextensions -infiles ./client_req.pem

5 Vamos a convertir el certificado cliente en un formato que Windows XP entienda, ***client_cert.p12:***

$ openssl pkcs12 -export -in client_cert.pem -inkey client_key.pem -out client_cert.p12 -clcerts

Configuración RADIUS

Servidor FreeRADIUS sobre Linux

- Directorio del servidor: ***/etc/raddb***
- Ficheros a modificar: ***eap.conf, clients.conf, y users***

Clients.conf: Indica que Access Point pueden solicitar información a nuestro servidor RADIUS. Pueden ser Ips individuales o subredes enteras.

eap.conf: *Configura todos los parámetros relevantes del servidor en relación a EAP.*

Users: *Permite especificar los usuarios de la red wireless.*

Configuración RADIUS

Configuración de *Clients.conf*

1

Editar el fichero y añadir al final:

```
client 192.168.1.0/24 {
secret = UV2k06
shortname = accesspoint
}
```

Explicacion del comando anterior

client <subred>/<mascara> {
secret = <password del punto de acceso>
shortname = Nombre
} .

Configuración RADIUS

Configuración de *eap.conf*

2

Editar el fichero y modificar:

```
# Extensible Authentication Protocol
#
# For all EAP related authentications
eap {
default_eap_type = tls
timer_expire = 60
```

Configuración RADIUS

Configuración de *eap.conf*

2

Editar el fichero y modificar:

```
tls {
        private_key_password = UV2k6
        private_key_file = ${raddbdir}/certs/server_keycert.pem

        certificate_file = ${raddbdir}/certs/server_keycert.pem
        CA_file = /etc/ssl/UVCA/cacert.pem

        dh_file = ${raddbdir}/certs/dh
        random_file = ${raddbdir}/certs/random
        fragment_size = 1024
        Include_length = yes
}
```

Configuración RADIUS

Configuración de *users*

3 Editar el fichero y añadir al final el usuario que hemos creado: (p, ej)

"Rafael Sebastian" Auth-Type := EAP

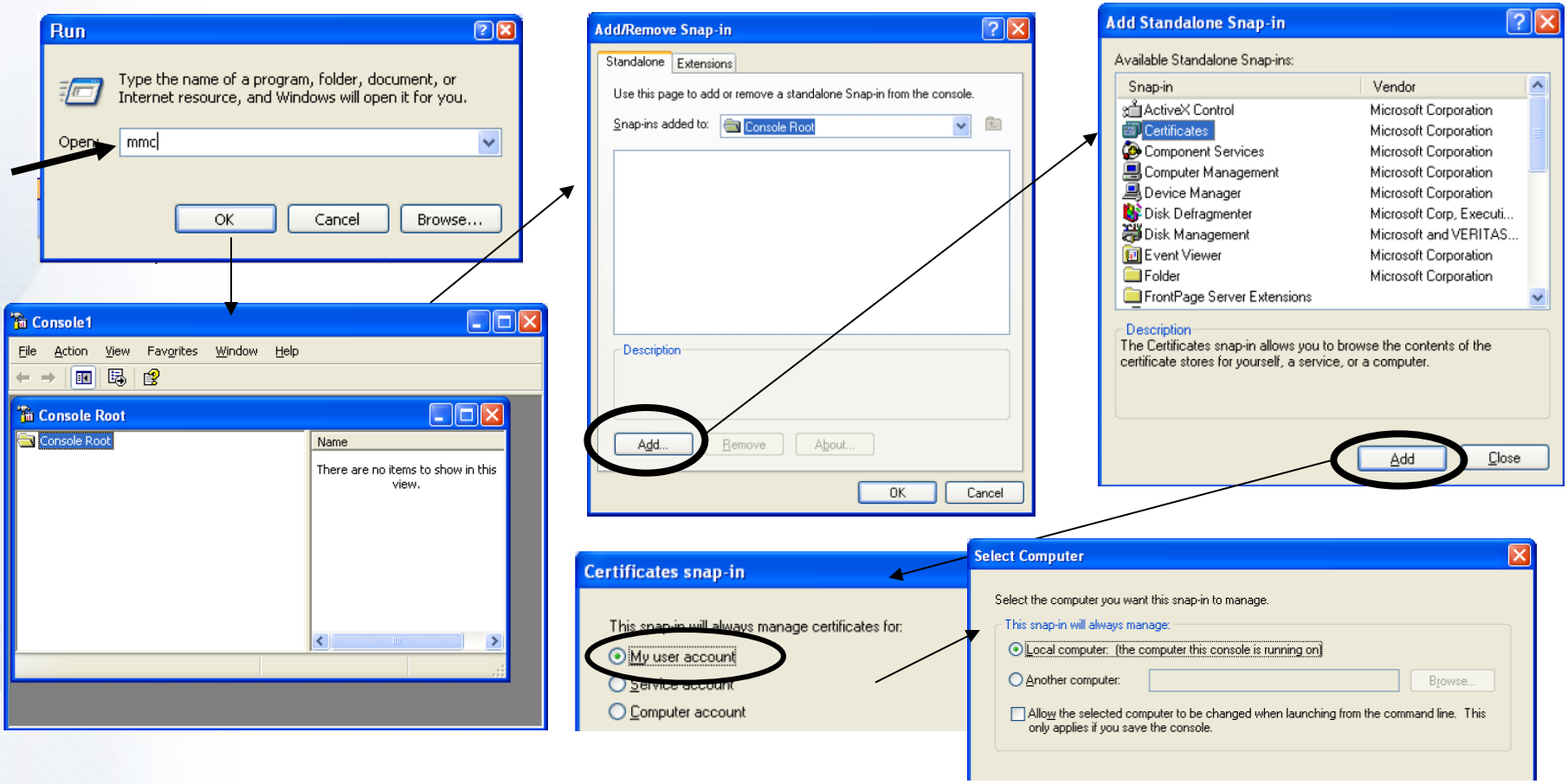
Activar el Servicio Radiusd

4 Finalmente solo tenemos que lanzar el servicio RADIUS.

$ radiusd -X

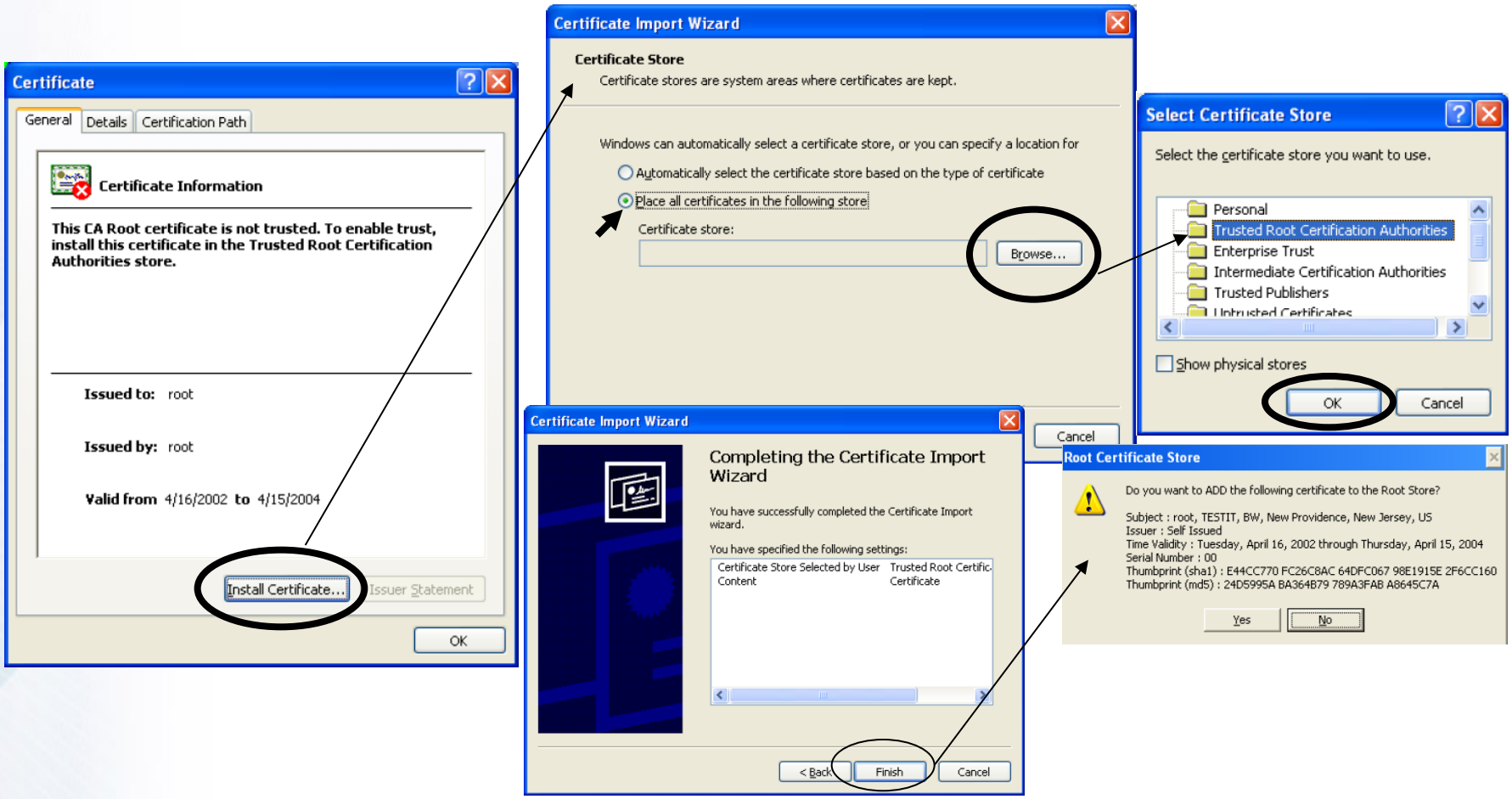
Configuración de *suplicante*

Instalación del CA



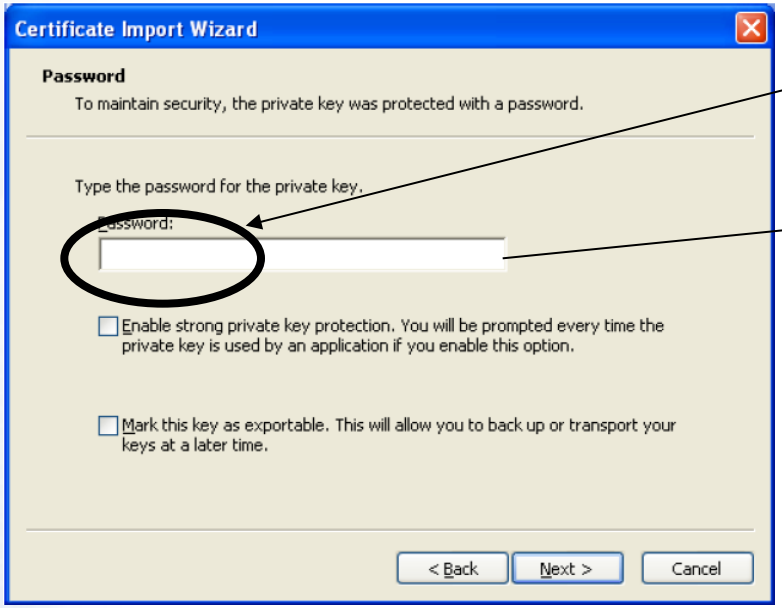
Configuración de *suplicante*

Instalación del CA

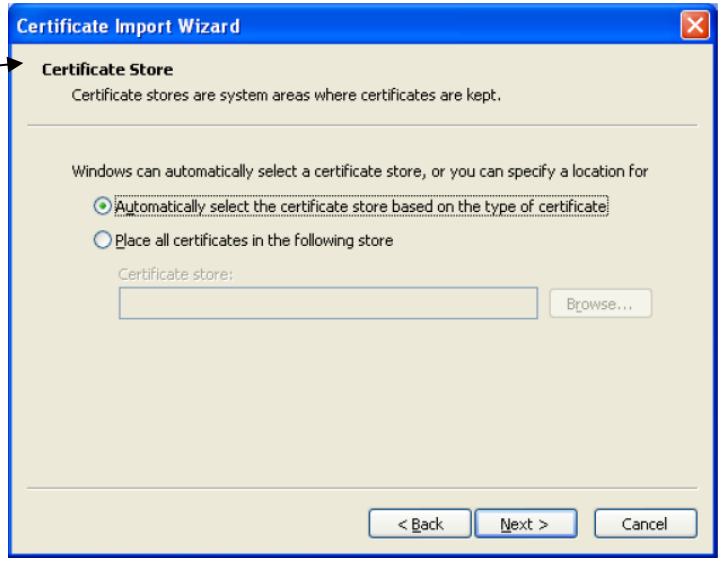


Configuración de *suplicante*

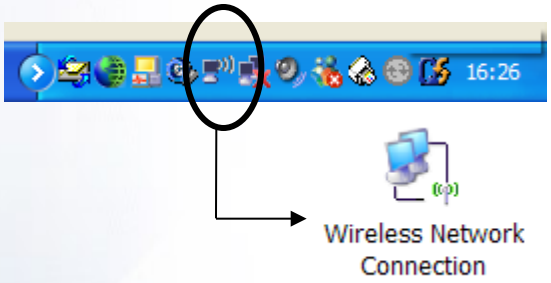
Instalación del Certificado digital de cliente

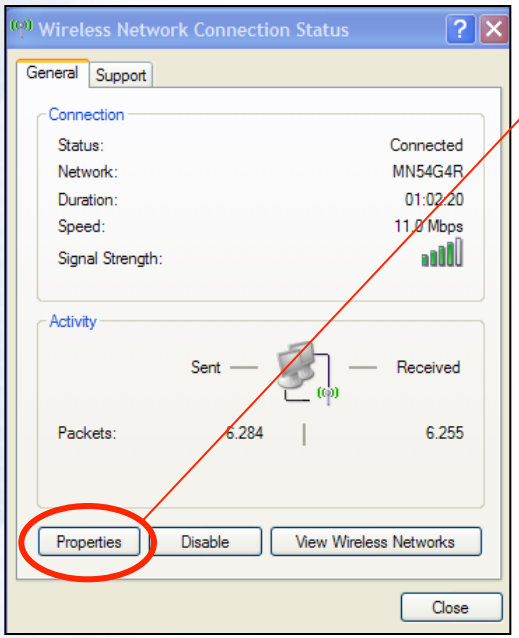


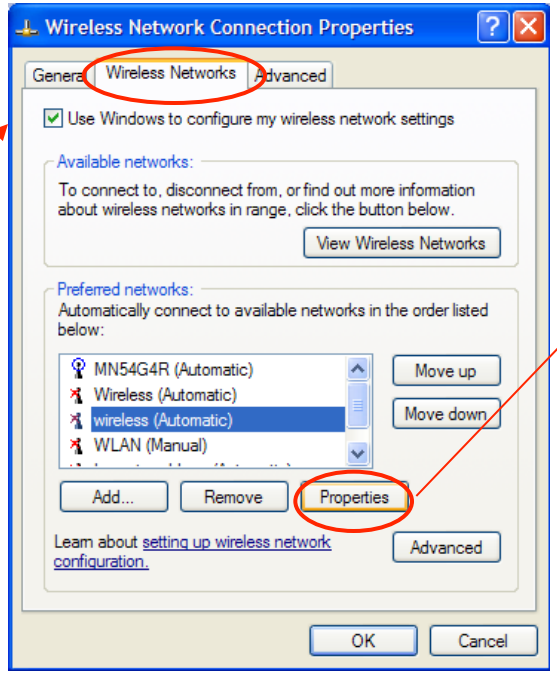
Clave: ***UV2k6***

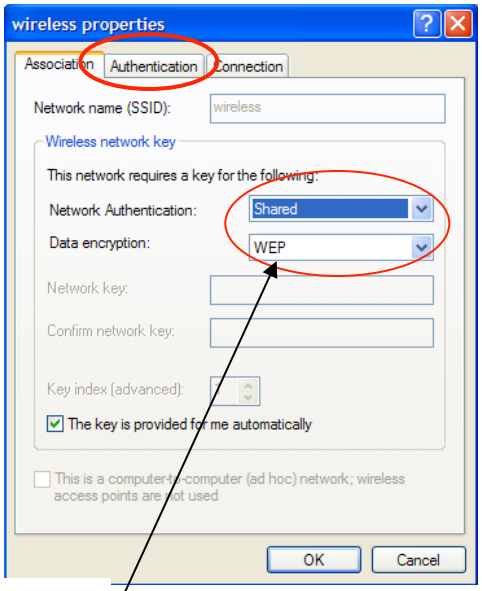


Configuración EAP-TLS









WPA

TKIP

Configuración AP1