Implementación de un ataque DoS a redes WPAN 802.15.4

Aleix Dorca
Estudis d'Informàtica,
multimedia i Teleciomunicació
Universitat Oberta de Catalunya
Email: adorca@uoc.edu

Jordi Serra-Ruiz
Estudis d'Informàtica,
multimedia i Teleciomunicació
Universitat Oberta de Catalunya
Email: jserrai@uoc.edu

***Resumen*—Las redes industriales y, concretamente, las redes de sensores son hoy en día una realidad emergente y con muchas expectativas de cara al futuro sobre todo en entornos empresariales o públicos. Los grandes ayuntamientos están creando las *smart cities* con este tipo de estructuras. Entre los estándares que existen parece que hay dos que se imponen, los estándares 802.15.4 y ZigBee. Conjuntamente proporcionan un conjunto de protocolos y servicios a los usuarios para permitir una comunicación fiable y segura. Todo y eso, como la mayoría de redes inalámbricas, estas redes no están exentas de potenciales peligros que pueden ponerlas en un compromiso. El objetivo de este artículo es mostrar la implementación de un ataque de denegación de servicio mediante un caso real. Como prueba de concepto, se muestra como dejar inhabilitado un nodo de una red que utiliza la especificación 802.15.4.**

***Palabras clave*—Seguridad (security), Wi-Fi, 802.15.4, ZigBee, Smart Cities, DOS.**

I. Introducción

Antes de tratar específicamente los estándares que son el objeto de este artículo es interesante hacer una breve presentación de lo que son las redes adhoc y, como caso especial, las redes de sensores que se utilizan es este tipo concreto de infraestructuras. Estos dos tipos de redes tienen muchas similitudes pero a la vez también presentan algunas diferencias importantes.

[1] describe una red adhoc como un conjunto de nodos que se comunican entre sí mediante unos enlaces radioeléctricos. Cada dispositivo de esta red tiene libertad total de movimientos por el espacio, y eso hace que la red se tenga que adaptar a los cambios de manera autónoma y automática. Los nodos pueden aparecer y desaparecer en cualquier momento. Por eso cada nodo se tiene que comportar como un encaminador de la información para el resto de los nodos, ya que los cambios en la estructura de la red pueden necesitar de esa característica, y tiene que hacer circular por la red el tránsito que recibe y del cual no es el destinatario final.

Las redes de sensores son una particularidad formada por dispositivos autónomos que por lo general se encargan de monitorizar condiciones ambientales o físicas. Así, por ejemplo, podemos encontrar sensores destinados en el control de temperatura, presión, peso, sonido, vibraciones, etc. Estos dispositivos inalámbricos envían la información que sus sensores detectan a través de la red hacia nodos de control. Estas redes son interesantes puesto que implementar una solución similar utilizando redes cableadas podría suponer un problema de presupuesto y un problema de logística haciéndola inviable en la mayoría de casos. Utilizando el aire como medio de comunicación da la posibilidad de instalar todos aquellos dispositivos o nodos necesarios a un precio muy económico en comparación con levantar toda una calle para colocar unos sensores de presencia de vehículos estacionados.

A grandes rasgos, los dos tipos de redes presentan las siguientes similitudes:

- Permiten la comunicación entre dispositivos mediante el envío de datos con encaminamiento multi-salto (*multi-hop*).
- Lo más usual es tratar con dispositivos que disponen de recursos mínimos, ya sea de proceso, memoria o almacenamiento, en los dos casos. Los dispositivos acostumbran a ser pequeños y alimentados con pequeñas baterías que con el tiempo hay que reemplazar.

Las principales diferencias son las siguientes:

- Las redes adhoc permiten la comunicación entre cualquier par de dispositivos mientras que las redes de sensores definen tipos de encaminamiento específicos.
- A pesar de que los dispositivos acostumbran a tener recursos mínimos, esta característica se hace todavía más patente en las redes de sensores donde los dispositivos, una vez asociados a la red, han que estar largos períodos tiempos (meses o años) sin ser recargados o reemplazados. Es evidente que la gestión de la energía es un punto clave en la gestión y diseño de estas redes. Algunos ejemplos podrían ser:
  - ◇ Calles donde los sensores de ocupación de plazas de estacionamiento se encuentran bajo tierra.
  - ◇ Monitorización de espacios naturales. (humedad, luz, lluvia...)
  - ◇ Detección de incendios, terremotos o inundaciones.
  - ◇ Control del tráfico.
- Los nodos en redes de sensores a menudo tienen relaciones de confianza entre nodos cercanos puesto que no es extraño que todos ellos recojan información similar o redundante, por lo que enviarla por la red sería una pérdida de recursos. Este comportamiento de complicidad no se

ISBN: 978-84-9717-323-0

encuentra en las redes adhoc.

I-A. Protocolo 802.15.4 y ZigBee

En esta sección se mostrará el comportamiento general del estándar 802.15.4 y el protocolo ZigBee, que permiten comunicar dispositivos remotos de bajo rendimiento.

I-A1. Estándar 802.15.4: El estándar 802.15.4 define las capas de comunicación física y de acceso al medio de la pila de protocolos. Otras características de este estándar son el hecho que presenta una alta flexibilidad en cuanto a la configuración de red, un bajo coste computacional y a la vez un muy bajo consumo [3].

La capa física además de enviar y recibir paquetes a la red se encarga de toda una serie de tareas como por ejemplo la activación del enlace, la detección de energía o el indicador de baja calidad.

Canales de transmisión

La capa física se puede configurar para transmitir en diferentes canales o bandas de frecuencia dependiendo de las necesidades de cada caso. Se definen los siguientes canales: Banda de 2450 MHz de 16 canales con una velocidad máxima de 250 kbps. Banda de 915 MHz de 10 canales con una velocidad máxima de 40 kbps y banda de 868 MHz de 1 canal con una velocidad máxima de 20 kbps.

Se ha que tener en cuenta que la comunicación en la banda de 2450 MHz trabaja en el misma zona de frecuencia que los dispositivos Wi-Fin 802.11. Es por eso que se recomienda escoger los canales 15, 20, 25 o 26 del estándar 802.15.4 para no provocar interferencias.

Capa de acceso al medio

Esta capa define como se realiza la comunicación a bajo nivel entre dispositivos. Se definen aspectos como la generación de los *beacons*, la duración de la transmisión de estos, el establecimiento de una política de *slots* equitativa, la asociación de nodos y la validación de las tramas [4].

El protocolo de acceso al medio se implementa mediante el algoritmo CSMA-CA.

La capa de control de acceso al medio define dos tipos de dispositivos que se pueden encontrar en una red 802.15.4 [5]:

- Los nodos de función completa (*Full Function Device*–FFD): Estos vienen equipados con una serie completa de funciones en la capa de acceso al medio, cosa que les permite actuar como coordinadores de la red o como dispositivos finales. Cuando estos nodos actúan como coordinadores pueden enviar *beacons*, o señalizaciones para proveer la red de servicios de sincronía, comunicación y procesos de acceso a la misma.
- Los nodos de función reducida (*Reduced Function Device*–RFD): Este tipo de nodos solo pueden actuar como nodos finales y no como coordinadores. Vienen equipados con sensores, actuadores, transductores, interruptores, etc. Y solo pueden interactuar con dispositivos que sean nodos de función completa.

Todas las redes 802.15.4 han de tener como mínimo un dispositivo FFD que actúe como coordinador. Uno de estos dispositivos es elegido coordinador de la PAN (*Personal Area Network*), responsable de las tareas de control de la red y de la seguridad.

Cualquier dispositivo RFD siempre tiene que estar asociado a un FFD para el correcto funcionamiento de la red. En el apartado de topología de la red de ZigBee se ven algunos ejemplo de asociación de nodos.

Formato de trama

La tabla I muestra el esquema de la estructura del formato de una trama MAC.

Donde: MHR es la cabecera, MSDU los datos (o *Payload*) y un final de trama (MFR)

- Campo de control: Este campo de longitud 16 bits contiene toda la información de control del paquete. Eso incluye el tipo de trama (Datos, ACK, etc.), si la seguridad está habilitada, si se necesario un ACK para esta trama. Además se define si los campos de direccionamiento estarán todos presentes y la longitud de estos. Por ejemplo, si el campo Intra-PAN está activo entonces el campo de PAN origen no estará presente. La tabla II muestra la estructura de estos campos.
- Control de secuencia: Este campo se utiliza para verificar el orden de llegada de los paquetes y para evitar ataques de reenvío. Este valor aparecerá en los paquetes ACK conforme el paquete con el código de secuencia especificado ha sido recibido.
- Campos de direccionamiento: Estos cuatro valores no son siempre obligatorios y dependerá del tipo de trama. Existen cuatro campos que corresponden a las PAN de origen y destino y la dirección origen y destino a nivel MAC. Las direcciones MAC pueden tener diferentes medidas según los estándares IEEE: 16 o 64 bits.
- Carga (Payload): En este campo se almacena los datos del paquete, concretamente, estará los datos del protocolo ZigBee, a pesar de que no tendría que ser siempre así puesto que este estándar está preparado para encapsular otros protocolos. La longitud de este campo es variable siempre y cuando no sobrepase la longitud máxima de una trama MAC (127 bytes).
- Código de verificación (FCS): 16 bits que almacenan los datos de verificación de la trama. Se utiliza un algoritmo CRC de 16 bits.

I-A2. La especificación ZigBee: La especificación ZigBee se encarga de definir en detalle las capas superiores de la pila de protocolos. Concretamente se trata de las capas de red y de aplicación. Además, ZigBee también define los siguientes aspectos:

- Tipo de dispositivos descritos en el apartado I-A1
- Topología de la red.
- Procedimiento para acceder o abandonar la red.
- Algoritmos de encaminamiento.

Topología

En en cuanto a la topología de la red se definen tres tipos de distribución de los nodos [5]. Se pueden apreciar en la figura 1

- Estrella: Existe un nodo central que a la vez actúa de coordinador y gestor. El nodo central es un dispositivo

Bytes:2	1	0/2	0/2/8	0/2	0/2/8	variable	2
Control	Secuencia	PAN destino	Destino	PAN origen	Origen	Datos	FCS
		Campos de direccionamiento					
MHR						MSDU	MFR

Tabla I
FORMATO DE LA TRAMA MAC

Bits: 0-2	3	4	5	6	7-9	10-11	12-13	14-15
Tipo de trama	Segur. habilitada	Trama pendiente	ACK necesario	Intra PAN	Reser.	Tipo direc. destino	Reser.o	Tipo direc. origen

Tabla II
FORMATO DEL CAMPO DE CONTROL DE LA CAPA MAC

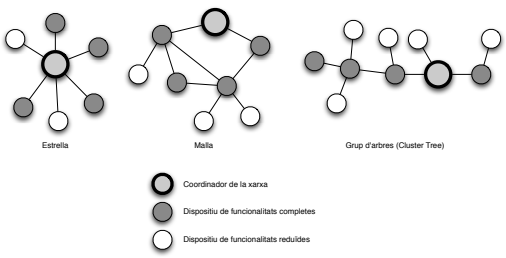

Figura 1. Topología de las redes de sensores

FFD mientras que el resto son, o pueden ser, RFD.

- Malla: Este tipo de topología permite que entre encaminadores FFD del tipo Cluster Tree también haya comunicación sin tener que depender del nodo central.
- Grupo de árboles: En este tipo de topología existe un nodo central que a la vez actúa como coordinador de la red y gestor. De este nodo dependen toda otra serie de nodos que pueden ser tanto FFD como RFD. En este caso los nodos FFD actuarán como encaminadores para otros dispositivos RFD. Es escalable siempre y cuando los encaminadores sean FFD.

Encaminamiento

El encaminamiento en las redes ZigBee se realiza dependiendo del tipo de topología que se ha escogido. En cualquier caso intervienen procesos de descubrimiento de rutas así como encaminamiento mediante tablas de rutas. El algoritmo de encaminamiento es sencillo y se puede resumir en que si la información es para el propio nodo se pasa a la pila de protocolos, sino se mira si es para un nodo hijo y se envía al nodo o a la ruta preestablecida.

Para realizar el descubrimiento de la ruta idónea se utiliza el algoritmo AODV (*AdHoc on Demand Distance Vector Routing*) que consiste en enviar un paquete a todos los nodos vecinos, que también propagarán, con el fin de llegar al nodo destino. A medida que el paquete pasa por los nodos se actualiza el coste de la ruta por la cual el paquete ha pasado. Y cuando el paquete llega al destino se envía una respuesta al nodo origen con la ruta más óptima.

II. ASPECTOS DE SEGURIDAD

Aparte de estas características básicas se ha puesto especial énfasis en la seguridad de los estándares. Las dos especificaciones establecen una serie de opciones que hacen que la seguridad ya no dependa del propio código de la aplicación. El protocolo puede cifrar, por ejemplo, el contenido de las tramas.

II-A. Seguridad en IEEE 802.15.4

El estándar 802.15.4 define tres modos de seguridad [5]–[6]:

1. Sin seguridad.
2. Modo ACL (*Access Control Lists*). No presenta cifrado pero solo se aceptan paquetes de dispositivos en listas de control de acceso.
3. Modo Seguro. Algunas de las características que puede incluir este modo son: Integridad, Confidencialidad, Control de acceso, etc.

A la vez se definen cuatro servicios de seguridad:

1. Control de acceso vía ACL.
2. Cifrado de datos mediante el algoritmo AES de 128bits.
3. Integridad de las tramas.
4. Control de secuencia para evitar ataques de reenvío.

Finalmente se definen ocho configuraciones de seguridad posibles en las que se puede escoger el algoritmo de cifrado así como el modo y la longitud del código de integridad. Algunas configuraciones incluyen solo autenticación de las tramas mientras que otras más completas añaden la opción de cifrado según los requerimientos de la aplicación.

II-B. Seguridad en ZigBee

Aparte de los elementos de seguridad del estándar 802.15.4 de los que ZigBee también se puede beneficiar, define toda una serie de conceptos orientados a la seguridad. De entrada la seguridad en ZigBee se basa en los siguientes principios [7]:

- Simplicidad: Cada capa se encarga de su seguridad.
- Es directa: Las claves de cifrado se intercambian directamente entre origen y destino.
- Extremo a extremo: Los datos circulan cifrados de origen a destino sin tener que ser descifrados en cada salto.

ZigBee define tres tipos diferentes de claves de cifrado que se utilizan, cada una de ellas, en casos muy diferenciados [5]. Los tres tipos de claves son:

1. Clave maestra: esta clave no se utiliza para cifrar información, sino para la generación de otras claves. Esta clave se establece en el momento de la construcción, pero puede ser entrada por el propio usuario o bien asignada por un centro de confianza. Todos los dispositivos disponen de una clave maestra propia y única.
2. Clave de red: la disponen todos los dispositivos de la red y se utiliza para enviar mensajes a toda la red. Los mensajes cifrados *broadcast* se cifran y se descifran mediante esta clave cuando la seguridad está habilitada. Esta clave se establece en el momento de la unión a la red o bien mediante procesos de renovación de claves.
3. Clave de enlace: se utiliza para establecer comunicaciones seguras entre dos dispositivos. Se obtiene a partir de la clave maestra mediante un proceso llamado SKKE.

Los servicios de seguridad en ZigBee incluyen métodos para el establecimiento de claves, el envío de estas claves, la protección de tramas y la gestión de dispositivos. Como el control de secuencia (*freshness*), que mediante contadores que se regeneran cada vez que se renuevan las claves se permite controlar la secuencia de los mensajes para que no se realicen ataques de reenvío. La integridad de los mensajes, que asegura que los mensajes enviados no han sido modificados durante la transmisión por ningún tercero. La autenticación, que mediante claves de red o enlace, los dispositivos pueden estar seguros que el origen de los mensajes es de quién dicen ser, evitando la suplantación por parte de intrusos. El cifrado, que mediante el algoritmo AES de 128 bits la protección se extiende a nivel de red o dispositivo. El cifrado es opcional sin necesidad de afectar otras características de seguridad. Las tramas están encapsuladas en la especificación 802.15.4, por lo que las cabeceras no van cifradas, como se puede observar en la tabla I

II-C. Vulnerabilidades

Las redes adhoc y, por extensión, las redes de sensores pueden ser vulnerables a toda una serie de ataques que se pueden categorizar de la siguiente manera [2]–[5]:

1. Denegación de servicio (*Denial of Service*-DOS): Estos ataques hacen que un nodo deje de funcionar mientras dura el ataque o indefinidamente.
2. Escucha de la red (*eavesdropping*): Como su nombre indica, un dispositivo escucha la red a la espera de recibir información. Evidentemente utilizando cifrado en la red o sobre los datos este ataque pasa a ser inútil, siempre y cuando no se combine con algún ataque para obtener las claves de cifrado.
3. Usurpación de identidad (*spoofinng*): Este ataque consiste al hacerse pasar por otro dispositivo, ya sea a nivel MAC, de red u otras. De este modo se pueden obtener paquetes por el dispositivo atacante. Si este usurpa un encaminador y actúa de manera ”legal”, podrá capturar toda la información que encamine.
4. Reenvío de paquetes (*replay*): Este ataque consiste a reenviar paquetes capturados para que el destino actúe de manera errónea. Por ejemplo, si un sensor manda un mensaje de incremento de temperatura, el nodo atacante podría reenviar un decremento del valor provocando que el nodo destino actúe de manera inversa a la deseada. Para evitar este tipo de ataques se utilizan los códigos de secuencia, el cifrado, etc.

III. Ataques

En esta sección se muestran la descripción de los ataques que se han realizado en este trabajo.

III-A. Ataques de denegación de servicio

Los ataques de denegación de servicio se pueden categorizar según la capa de la pila de protocolos a la que van dirigidos [8].

Posibles ataques a la capa física:

- Interferencias (Jamming): consiste en saturar un canal de comunicación con información errónea para que ningún otro dispositivo pueda utilizarlo. En general este tipo de ataque se cancela mediante diferentes canales en los que transmitir.
- Alteración de datos (*Data tampering*): consiste en modificar la información que circula por la red, capturando los datos y modificándolos. Este tipo de ataque se puede frenar con códigos de verificación de datos.

Posibles ataques a la capa de enlace:

- Colisión: en este caso, similar al *jamming*, se modifica cierta información del origen provocando que la verificación del paquete provoque un error. De este modo se provoca un reenvío de los paquetes que puede llevar al límite los recursos. No se conoce un procedimiento totalmente fiable para evitar este tipo de ataques.
- Ruido en el canal: existen muchos errores en la transmisión que implican un gran reenvío de paquetes. Si se consigue que un dispositivo agote todos sus recursos y quede aislado o inoperativo el ataque se considera satisfactorio. Para evitarlo se puede establecer un umbral a partir del cual no se retransmite.
- Longitud de las tramas: este ataque deja la red inservible ocupando el canal enviando muy poca información a intervalos regulares y constantes.

Posibles ataques a la capa de red y encaminamiento:

- *Homing*: este ataque obtiene información sobre nodos que son de especial importancia en la red. En las redes ZigBee, el ataque se centra en el PAN Coordinator. El cifrado de datos puede mitigar el ataque.
- Encaminamientos erróneos o selectivos: se implementa sobre encaminadores que rechazan o encaminan erróneamente paquetes. Para evitar este tipo de ataques el dispositivo puede probar de encaminar los paquetes por otra ruta.
- Agujeros negros (*Black holes*) y agujeros de gusano (*Wormholes*): en las redes que utilizan el protocolo de descubrimiento de rutas basado en el coste del enlace, este ataque puede provocar la construcción de rutas erróneas si un dispositivo siempre anuncia la calidad de su enlace como la mejor. De este modo la mayoría de

Figura 2. mota Z1 de Zolertia

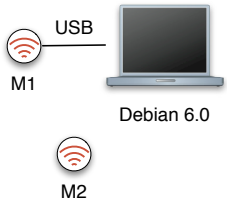

Figura 3. Esquema de montaje

paquetes serán enviados a través de él y este podrá aplicar decisiones de encaminamiento erróneas o simplemente descartarlos.

- *Sybil*: se basa en que un nodo pueda presentar varias identidades a la vez. Así se pueden romper esquemas de encaminamiento múltiple o bien causar problemas en entornos geográficamente dispares.

Posibles ataques a la capa de aplicación:

- Inundar la red (*Flooding*), *HELLO attacks*: una vez introducido el nodo malicioso en la red, envía peticiones de conexión que pueden llevar al nodo remoto a agotar los recursos y quedar inoperantivo. Para evitar este ataque se presentan soluciones del tipo limitar el número de conexiones establecidas o presentar rompecabezas al cliente que tiene que resolver antes no se le otorgue el recurso.
- De-sincronización: Mediante el envío con códigos de secuencia erróneos a nodos que ha establecido conexión con un tercero se puede forzar el reenvío de tramas. Si además se sigue el ataque con insistencia se pueden agotar sus recursos. Estos ataques no tienen sentido si los nodos pueden comprobar la veracidad de los paquetes mediante cifrado o códigos MAC.

IV. Resultados experimentales

Este apartado pretende mostrar una prueba de concepto en el que se ha llevado a cabo un ataque de denegación de servicio sobre una red 802.15.4/ZigBee. El proceso que se ha seguido es muy simple y lo que se desea mostrar es la facilidad con la que ha sido posible dejar sin recursos un nodo de la red.

Entre los posibles ataques que se han mostrado, este ataque recaería sobre el agotamiento de recursos en la capa de aplicación y la capa de enlace puesto que ambos tienen gran parte de implicación.

Para montar la red ZigBee se han utilizado dos sensores o *motas*. Concretamente se ha utilizado la plataforma Z1 de la marca Zolertia [9], mostrada en la figura 2.

Por otro lado, para simular el dispositivo atacante se ha utilizado un sniffer/inyector de la marca Atmel: el dispositivo RZUSBSTICK [10].

En esta prueba de concepto es necesario que un sensor actúe como nodo encaminador (llamado M1 en la figura 3) y que el otro actúe como dispositivo RFD hoja (M2). La topología escogida es la de estrella. Y el nodo al que se atacará agotando los recursos es el que actúa como encaminador (la mota M1). Por lo que en el caso de tener más sensores conectados a la mota M1, todos dejarían de poderse comunicar. Para programar las motas se han de conectar inicialmente a la máquina Debian con un cable USB/MicroUSB, después ya no es necesario esa conexión y pueden operar mediante pilas y por tanto ser un nodo completamente autónomo. Con las aplicaciones base que hay disponibles, la mota M1 se ha programado con la aplicación IPBaseStation, mientras que la mota M2 utiliza el servicio TCPEcho.

El siguiente paso consiste en examinar los paquetes que se envían en el momento de la asociación de la mota M2 en la red y los que se envían durante la transmisión de paquetes al servicio Echo para poder, posteriormente, inyectar los paquetes previamente modificados desde el dispositivo atacante.

Para averiguar qué paquetes circulan por la red durante las fases de asociación de dispositivos y la conexión al servicio Echo se ha utilizado el software Wireshark y la herramienta zbdump. Y para escuchar la red mediante el dispositivo RZUSBSTICK son necesarias las herramientas que se encuentran en KillerBee [11]. Eso es debido a que una simple tarjeta WiFi convencional no puede escuchar las frecuencias de las redes 802.15.4

Mediante el dispositivo RZUSBSTICK, el software Wireshark y la herramienta zbdump se ha determinado cuáles son los paquetes esenciales y necesarios para llevar a cabo la asociación de un dispositivo al encaminador. Y al mismo tiempo se ha obtenido el paquete que se envía cuando se solicita un Echo al servicio de la mota M2. Esta información se ha utilizado para construir posteriormente unos paquetes específicos con los que atacar el sistema.

IV-A. Atacando al sistema

El ataque desarrollado se basa en dos scripts que por un lado asocian una serie de nodos falsos al encaminador M1 y de la otra envían paquetes al servicio TCPEcho de la mota M2 de manera ininterrumpida mediante el encaminador M1. Los paquetes de la Mota M2 con las respuestas solicitadas pasan a través de M1 y llegan a los nodos falsos maliciosos que están programados especialmente para no responder a las peticiones ACK que solicita M2 de sus respuestas. Como el número de nuevas peticiones por parte de los nodos falsos no cesa y la mota M2 está continuamente enviando los datos que supone no han llegado a los nodos destino, la tabla del encaminador se satura y se queda sin recursos, lo que hace bloquear completamente el encaminador, dejando a todos los

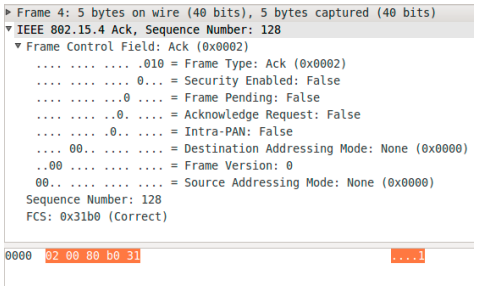

Figura 4. Captura de red, ACK

sensores aislados de la red.

La programación del capturador de tráfico es relativamente sencilla, ya que se aprovecha las tramas reales para inyectar el código modificado y no contestar a los ACK. Básicamente se realizan los siguientes pasos:

1. Inicialización: utilizando la API de Killerbee se define en qué canal retransmitir y el período de tiempo que se dejará pasar entre paquetes. Las herramientas KillerBee todavía no disponen de un programa que escuche todos los canales a la vez por el que se ha de determinar. Para este escenario se ha fijado el canal.
2. Bucle principal: para cada dispositivo se genera el valor del paquete en la variable *origen*. La posición donde se ha de poner este valor se ha determinado del estudio de los paquetes capturados con zbdump y Wireshark y de la especificación del formato de trama MAC. En la figura 4 se puede ver un ejemplo de los paquetes capturados, concretamente, el Ack correspondiente en formato Daintree mediante la herramienta zbdump.
3. Inyección: el paquete se envía a la red(inject).
4. Se duerme el sensor los segundos especificados.
5. El script acaba con la limpieza de la trama enviada.

En el caso del ataque con el envío de solicitudes el script que hace las peticiones al servicio Echo es similar al anterior caso, cambiando únicamente las variables utilizadas de la API.

Una vez ejecutado el ataque el nodo falso envía paquetes de solicitud de Echo de manera indiscriminada desde direcciones falsas de dispositivo asociados a la M1 que no existen provocando que la M1 tenga que responder a todos los Ack solicitados, así como a la propia respuesta del protocolo. Como el nodo falso malicioso no responde a ningún paquete, se crea una situación en la que la mota M1 tiene que reenviar paquetes varias veces.

Cuando se desea realizar una comunicación con la mota M2 esta responde correctamente a algunas peticiones pero a medida que el ataque progresa ésta dejará de responder. Pero en realidad la mota M1 ha dejado de encaminar paquetes. El número de paquetes necesario para que la mota M1 deje de responder varía en cada ejecución y depende de la cantidad de envíos correctos realizados y del tiempo transcurrido antes del ataque.

En general, según las pruebas llevadas a cabo, la mota M2 responde una decena de peticiones de Echo antes de que la mota M1 deje de responder. El bloqueo de la mota M1 es absoluto siendo necesario un reinicio completo del dispositivo para que vuelva a funcionar la comunicación con la mota M2.

El envío de paquetes por parte del nodo malicioso no es en ningún caso exhaustivo. En este caso cada paquete se envía con un retraso de 0.5 segundos respeto la anterior cosa que permite un tiempo suficiente a todas las partes del sistema a responder sin provocar un bloqueo del medio. A pesar de que el ataque de saturación del medio es igualmente factible, este no es demasiado interesante para los protocolos.

V. Conclusión

En este artículo se ha descrito el funcionamiento y las características básicas de los estándares 802.15.4 y ZigBee, el montaje y puesta en funcionamiento de redes de sensores utilizados en redes de ”Smart Cities”.

A continuación se han descrito las opciones de seguridad que estos dos protocolos ofrecen, describiendo cuales son los casos de ataque más comunes en este tipo de redes y dedicando un apartado especial a los ataques de denegación de servicio en el que se basa la prueba de concepto de ataque sobre una red 802.15.4/ZigBee.

Finalmente se ha ejemplificado un caso real en el que es posible dejar sin recursos un dispositivo 802.15.4 mediante únicamente herramientas de libre distribución y la programación en Python de un script que envía solicitudes Echo de manera indefinida hasta que el dispositivo se satura y deja de funcionar.

Como trabajo futuro está el estudio más detallado de las posibles mejoras al protocolo estándar para que no se puedan realizar este tipo de ataques sobre las cabeceras no cifradas.

Agradecimientos

This work was partly funded by the Spanish Government through projects TIN2011-27076-C03-02 “CO-PRIVACY” and CONSOLIDER INGENIO 2010 CSD2007-0004 “ARES”.

Referencias

[1] E. Peig, “Redes abiertas (Cuando los usuarios forman parte de la red),” Barcelona, Ed. UOC, 2012.
[2] C. Karlof, D. Wagner, “Secure routing in wireless sensor networks: Attacks and countermeasures,” en *IEEE SPNA*, 2002.
[3] S. Coleri Ergen, “ZigBee/IEEE 802.15.4 summary,” Unknown, 2004.
[4] IEEE, “802.15.4,” http://www.ieee802.org/15/pub/TG4.html.
[5] P. Baronti, P. Pillai, V.W.C. Chook, S. Chessa, A. Gotta, and Y. Fun Hu, “Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards,” *Computer Comunications*, vol 30(7), pp. 1655–1695, 2007.
[6] H. Li, B. Xue, W. Song, “Application and Analysis of IEEE 802.14.5 Security Services.” en *2nd International Conference on Networking and Digital Society (ICNDS)*, pp. 139–142, 2010.
[7] H. Li, Z. Jia, X. Xue, “Application and Analysis of Zigbee Security Services Specication,” en *Networks Security Wireless Communications and Trusted Computing (NSWCTC)*, pp.494–497, 2010.
[8] A.D. Wood, J.A. Stankovic, “Denial of service in sensor networks”, *Computer*, vol.35(10), pp.54–62, 2002.
[9] Zolertia, “Platform Z1”, http://www.zolertia.com/ti
[10] Atmel, “Rzusbstick”, http://www.atmel.com/tools/rzusbstick.aspx
[11] KillerBee, “Homepage”, http://code.google.com/p/killerbee/