ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA
GRADO EN INGENIERÍA INFORMÁTICA

Análisis de seguridad del protocolo 6LoWPAN

Security analysis of 6LoWPAN protocol

Realizado por
Ángel Jesús Cañete Valverde
Tutorizado por
Francisco Javier López Muñoz
Departamento
Lenguajes y ciencias de la computación

UNIVERSIDAD DE MÁLAGA
MÁLAGA, SEPTIEMBRE DE 2015

Fecha defensa:
El Secretario del Tribunal

Resumen

En la actualidad, cada vez es más común que objetos de la vida cotidiana tengan la capacidad para conectarse a Internet, lo que se denomina el Internet de las Cosas (IoT). 6LoWPAN es un protocolo encargado de dotar a dichos objetos de esta capacidad, de forma que cada uno de ellos cuente con una dirección IPv6. Los recursos de los dispositivos que forman estas redes son limitados, tanto su memoria como la capacidad de cómputo, y suelen estar alimentados por baterías; 6LoWPAN hace que, pese a ello, puedan conectarse a Internet directamente.

El Internet de las Cosas es la base de una gran cantidad de servicios y esta tendencia irá en aumento, por lo que muchas operaciones que realicemos dependerán de la disponibilidad y la confianza en estas redes. La seguridad de la información se convierte en un concepto muy importante. Debido a las limitaciones de los dispositivos que forman las redes 6LoWPAN, la implantación de seguridad en éstas se hace aún más complicada. En este trabajo se investigan las vulnerabilidades que presentan las redes 6LoWPAN y se tratan varias posibles soluciones de seguridad en diferentes capas del protocolo, realizando un estudio de las mismas con el fin de comprobar su viabilidad en el escenario en el que nos encontramos. Para ello, se evalúa la eficiencia de los diferentes mecanismos de seguridad, atendiendo a los aspectos cruciales en este tipo de dispositivos.

Abstract

Nowadays, it is increasingly common that daily life objects have the ability to connect to the Internet, which is called the Internet of Things (IoT). The 6LoWPAN protocol is responsible for providing this connectivity, so each device has an IPv6 address assigned. The device's resources that compose this networks are limited, both in memory and computing capacity, and they are usually powered by batteries; 6LoWPAN provides a mechanism so they can connect directly to the Internet despite those limitations.

The Internet of Things is the basis of a big amount of services, and this trend will increase, so the operations we will perform will depend on the availability and reliability of these network. Here is where the information security becomes a key concept. In this paper we have investigated the vulnerabilities discovered in 6LoWPAN networks and

we will provide several possible security solutions in different protocol layers by doing a study of them in order to prove their viability in our current scenario. To do so, we will measure the efficiency of the different security mechanisms, taking into account the crucial aspects of this kind of devices.

Palabras clave

Seguridad de la información, autenticidad, confidencialidad, integridad, 6LoWPAN, Internet de las Cosas, IoT, WSN, IPSec, IEEE 802.15.4, CoAP, DTLS, IP, SVELTE, RPL, 6Mapper, 6BR, DODAG.

Key words

Security of information, Authenticity, confidentiality, integrity, 6LoWPAN, Internet of Things, IoT, WSN, IPSec, IEEE 802.15.4, CoAP, DTLS, IP, SVELTE, RPL, 6Mapper, 6BR, DODAG.

Índice

Índice de figuras

Índice de tablas

Índice de gráficos

Índice de algoritmos

1. Introducción

El llamado Internet de las Cosas (IoT) se trata de la interconexión de manera digital de los objetos cotidianos con Internet. En la construcción del IoT entran en juego una gran cantidad de tecnologías que ayudan a que exista un mundo en el que los objetos que conforman dicha red sean capaces de comunicarse entre ellos (Machine-to-Machine, M2M), que en las interacciones entre las personas y éstos ofrezcan información a dicha persona o que la interacción del usuario haga que el objeto lleve a cabo una función en concreto. Las redes de sensores (Wireless Sensor Networks, WSN) es una de estas tecnologías que conectan el mundo virtual y el físico, donde los nodos tienen la capacidad para comunicarse automáticamente unos con otros y con otros sistemas inteligentes.

En una red de sensores convencional, los nodos de la misma tienen la capacidad de recopilar información ambiental y enviar éstos datos a un nodo receptor para el posterior procesado de los mismos. Sin embargo, las redes de nodos actuales se encuentran en entornos mucho más cercanos para los humanos y están destinados a tareas mucho más complejas, como domótica, videovigilancia, automatización industrial y control, etc.

Estos nodos que forman las redes que nos ocupan se tratan de dispositivos con unas capacidades de cómputo limitadas, con una baja capacidad de almacenamiento y alimentadas con baterías. El protocolo IP (Internet Protocol) está propuesto también para las redes de nodos, aunque se asume que es demasiado pesado para utilizarlo en WSN ya que la cabecera que utiliza IPv6 [7] es de 40 bytes que se agrega a cada uno de los paquetes. Sin embargo, IP ofrece interoperabilidad, escalabilidad, fácil programación y cuenta con hardware preparado para el uso de dicha tecnología.

Como tarea principal, 6LoWPAN ajusta paquetes IPv6 a las características únicas y requisitos del salto múltiple inalámbrico entre dispositivos de baja potencia.

El objetivo que nos ocupa es el análisis de varias soluciones de seguridad con el propósito de comprobar su viabilidad en redes de nodos 6LoWPAN, atendiendo a las limitaciones de los dispositivos que las forman. Para ello, se estudian distintos mecanismos de seguridad, comprobando a partir de datos obtenidos de sus implementaciones tales como el incremento en el uso de energía o del tamaño de los mensajes si es factible su implantación.

En el TFG se incluyen detalles de la implementación de los propios mecanismos de seguridad que en él se tratan, pese a que no sea necesario para lograr el objetivo del propio trabajo. El fin de este nivel de detalle es el de proporcionar al lector que esté

interesado en la implementación de uno o varios de ellos las herramientas necesarias para poder llevar a cabo su propósito, así como que aquel usuario al que le interese el funcionamiento interno de los mismos tenga material con el que saciar su interés.

1.1 El internet de las cosas e IPv6

Como se ha mencionado anteriormente, el protocolo IP ofrece una serie de características que hacen idóneo su uso en WSN; sin embargo, es demasiado pesado para ser utilizado en dispositivos con recursos limitados.

6loWPAN (IPv6 over Low power Wireless Personal Area Network) se trata de un estándar que permite el uso de IPv6 sobre redes basadas en el estándar IEEE 802.15.4 [1]. 6loWPAN permite que los dispositivos de una red inalámbrica de área personal se puedan conectar por IP, lo cual lo hace compatible con el modelo OSI.

Dicho protocolo se utiliza como principal bloque de construcción para una serie de escenarios de red en el Internet de las Cosas incluyendo domótica, sistemas de control industrial o ciudades inteligentes. Por consiguiente, una amplia gama de aplicaciones en estos escenarios emplean el protocolo 6LoWPAN para la comunicación basada en IP a través del estándar o protocolos de la capa superior de propósito especial.

6LoWPAN ajusta paquetes IPv6 a las características de dispositivos de baja potencia. La variedad de las aplicaciones que pueden llevarse a cabo de esta forma requiere que 6LoWPAN proporcione una solución tanto para las transmisiones de pequeño tamaño, como pueden ser los datos de los sensores o de control comandos, como para grandes transmisiones de información, como pueden ser las actualizaciones de firmware o handshakes de protocolos de seguridad.

6LoWPAN hace uso de otros subprotocolos tales como RPL (Routing Protocol for Low-power and Lossy Networks) en la capa de red y CoAP (Constrained Application Protocol) en la capa de aplicación:

RPL

El enrutamiento en redes con restricciones (LLNs, Low-Power and Lossy Networks) en el Internet de las Cosas, en el que sus nodos cuentan con una energía y capacidad del canal limitadas, se consigue mediante el protocolo de enrutamiento RPL, que se trata de un protocolo de enrutamiento pensado para este tipo de redes [2]. El protocolo RPL crea un grafo dirigido acíclico coorientado al destino (DODAG) que tiene como objetivo establecer un coste hasta la llegada al router, 6BR. RPL es compatible tanto con el tráfico unidireccional a una raíz DODAG (típicamente el 6BR) y el tráfico bidireccional

entre los nodos y una raíz DODAG. Cada nodo en el DODAG tiene un ID de nodo (una dirección IPv6), uno o más padres (a excepción de la raíz DODAG), y una lista de nodos vecinos.

Los nodos tienen un rango que determina su ubicación con respecto a los vecinos y con respecto a la raíz DODAG. El rango aumentará desde la raíz DODAG hacia los nodos. Dentro de la red, las tablas de enrutamiento mantienen separados a los paquetes que se dirigen hacia arriba y los paquetes dirigidos a la parte baja de la red; se trata del llamado storing mode, modo de almacenamiento. RPL también es compatible con el modo de no-almacenamiento donde los nodos intermedios no almacenan ninguna ruta.

CoAP

Debido a la naturaleza de baja potencia y con pérdida de paquetes de las redes inalámbricas en el IoT, se utiliza UDP en lugar de TCP, ya que este último está orientado a la conexión. Al tratarse de un protocolo orientado a la conexión (necesita mensajes de establecimiento y terminación, three-wayhandshake) y ser además fiable (realiza retransmisiones cada vez que se pierde un paquete), no es adecuado para este tipo de redes en las que lo interesante es realizar el mínimo envío de paquetes posible. El Protocolo HTTP está diseñado para TCP y no es factible su utilización en el IoT basado en UDP. Por este motivo, el IETF define CoAP [3], un subconjunto de HTTP que está siendo estandarizado como un protocolo web para el IoT. CoAP está adaptado a dispositivos limitados y a la comunicación máquina a máquina.

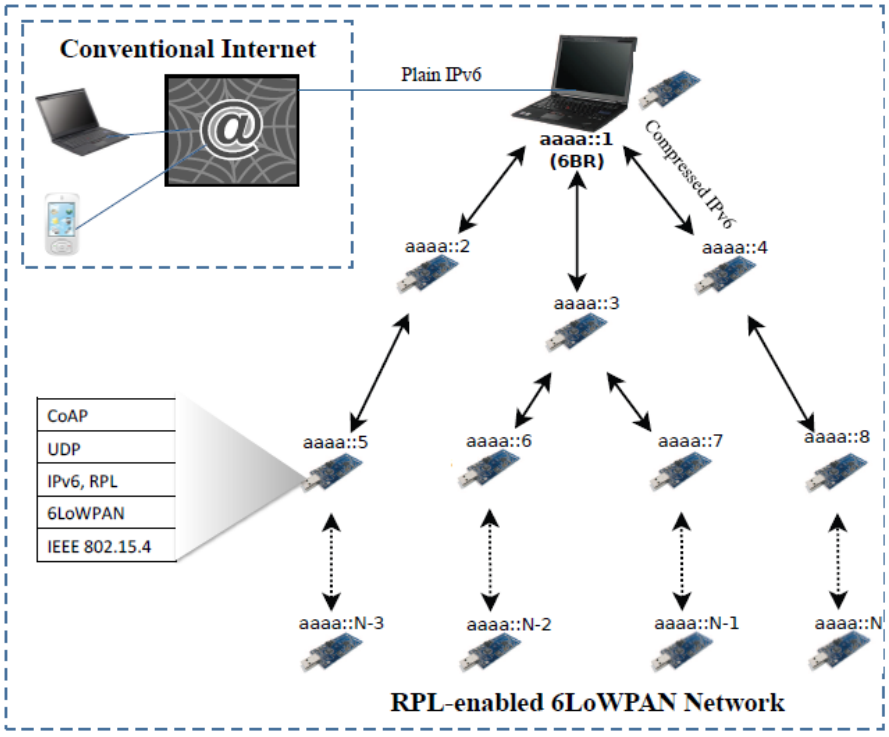


Figura 1: Entorno de funcionamiento 6LoWPAN

1.2 Características

Las características principales de 6LoWPAN son:

- Bajos anchos de banda con tasas de transferencia de hasta 250Kbps, 40 Kbps y 20Kbps definida por la capa física.
- Tamaño de paquetes pequeños (menor o igual a 127 bytes).
- Soporta direcciones MAC de 16 bits cortas o 64 bits extendidos.
- Las topologías soportadas son las de malla y estrella.
- Trabaja con un bajo consumo de energía y su construcción es de bajo costo.
- Soporta un gran número de dispositivos conectados a la red.
- La localización de estos dispositivos no está predefinida en algunos casos, ya que éstos pueden cambiar de posición dinámicamente.
- Los dispositivos en una red 6LoWPAN suelen estar largos periodos de tiempo en modo de hibernación (Sleep Mode) para ahorrar energía.

1.3 Seguridad en el IoT

Las exigencias de seguridad de la información han ido creciendo en las últimas décadas motivadas principalmente por una mayor exposición de los sistemas. Antes del uso extendido de equipos de proceso de datos, la seguridad de la información se garantizaba por medios físicos y administrativos.

En el contexto en el que nos encontramos, con entornos distribuidos y descentralizados, se hace indispensable la transmisión constante de grandes volúmenes de datos a través de redes públicas, atravesando multitud de medios. La necesidad de seguridad ha ido creciendo a medida que crecía esta interconexión global, así como por el tipo de información en tránsito.

La seguridad en el protocolo 6LoWPAN es el tema que nos ocupa en este trabajo. Es decir, las diferentes vulnerabilidades que presenta el protocolo, así como los ataques que pueden llevarse a cabo por parte de terceros haciendo uso de esas debilidades.

A continuación se realiza una pequeña introducción en el concepto en sí de “seguridad de la información”, así como los requisitos que se consideran exigibles para preservar ésta.

1.3.1 Concepto de seguridad

La definición de este concepto viene dada por los requisitos necesarios para preservarla. Para ilustrar estos requisitos, entenderemos el sistema como una función que transfiere información entre dos agentes, un agente emisor y un receptor.

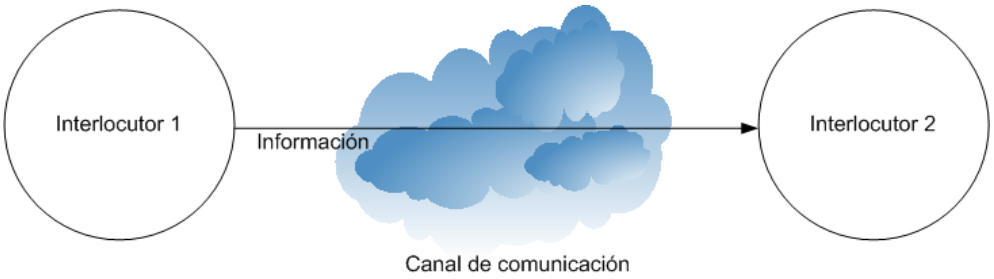


Figura 2: Comunicación entre dos agentes

A continuación se exponen los requisitos de seguridad que se pueden considerar exigibles y necesarios para el estándar que nos ocupa, de forma que se garantice una comunicación segura entre nodos.

Confidencialidad:

La confidencialidad de datos exige que la información de un sistema de computadores sea accesible para lectura solamente a aquellas personas o sistemas que se encuentren autorizados. Este tipo de acceso incluye la visualización y otras formas de revelación, incluyendo el simple revelado de la existencia del objeto.

La amenaza a la confidencialidad se encuentra en la intercepción de la comunicación por un agente no autorizado, ilustrado en la Figura 3. La probabilidad de que esto ocurra dependerá del medio físico de la comunicación, o de los elementos intermedios ubicados entre los dos extremos de la comunicación.

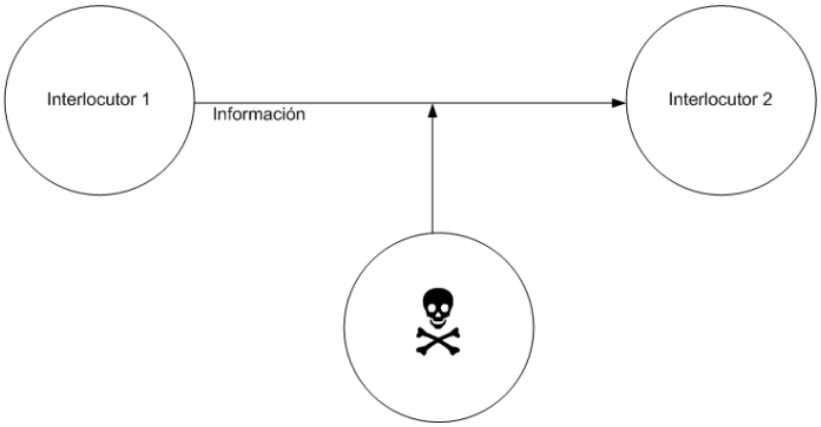


Figura 3: Compromiso de la confidencialidad

Integridad:

Los elementos de un sistema de computadores deben ser modificados únicamente por personas o sistemas con la autorización necesaria para ello. La modificación de los mismos incluye escritura, cambio, cambio de estado, borrado y creación (Figura 4).

La amenaza de que estas modificaciones sean llevadas a cabo por terceros vienen dadas por un acceso no autorizado y la alteración de la información que se encuentra en tránsito; por lo tanto, la probabilidad de éxito de esta amenaza viene dada por la facilidad que tenga el atacante para acceder al canal. Sus efectos pueden llegar a ser muy perjudiciales si no es detectado.

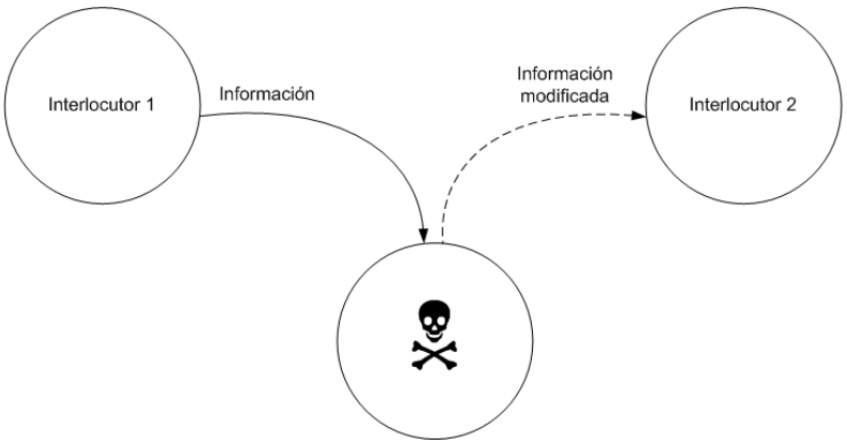


Figura 4: Modificación de información

Otro tipo de ataque a la integridad consiste en la inserción de información falsa en el sistema, como puede ser la retransmisión de un paquete (inyección de información), ilustrado en la Figura 5.

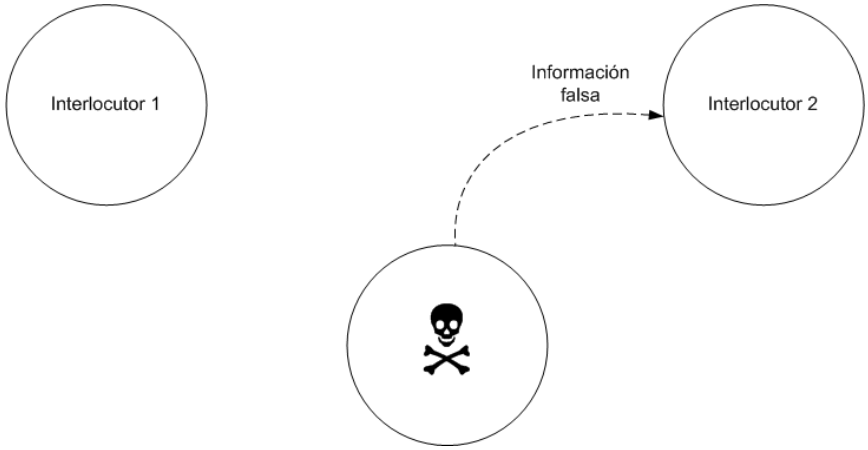


Figura 5: Inyección de información

Como aspecto de este requisito podemos definir el concepto de autenticidad.

Autenticidad:

Es la propiedad que permite identificar el generador de la información, es decir, el receptor debe tener la certeza de que la información que recibe es de ese alguien que la ha mandado, y no una tercera persona haciéndose pasar por la otra (suplantación de identidad).

Disponibilidad:

Garantiza la supervivencia de servicios de red para su utilización por parte de los grupos autorizados en el momento en que sean necesarios, es decir, la supervivencia del sistema ante un ataque de tipo DoS (Denial of Service), un ataque al sistema que causa que un servicio o recurso sea inaccesible a los usuarios legítimos (Figura 6).

La amenaza a la disponibilidad se encuentra en la interrupción de las comunicaciones, ya sea interviniendo sobre el medio, sobre los interlocutores o sobre los elementos intermedios involucrados en la comunicación.

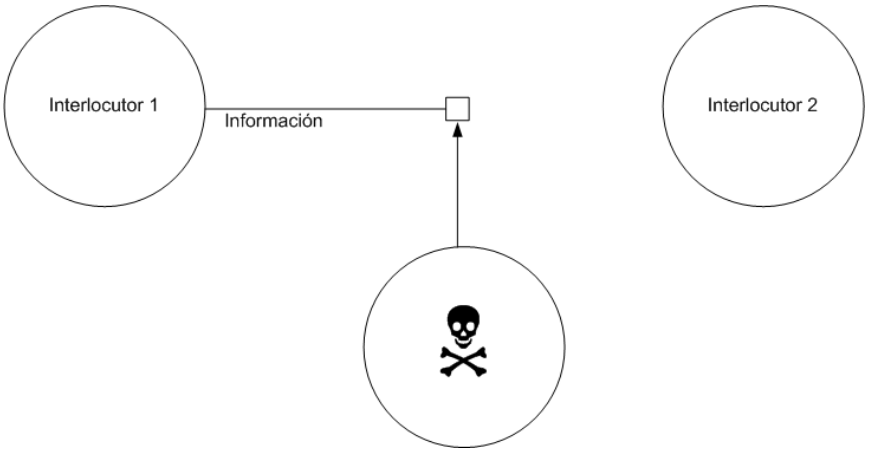


Figura 6: Denegación de servicio

A partir de los anteriores y teniendo en cuenta el escenario descentralizado en el que nos encontramos, aparecen otra serie de requisitos de seguridad, expuestos a continuación.

Otros requisitos aplicables a redes de nodos:

Frescura de datos: implica que cada dato es reciente, de forma que se pueda estar seguro de que ningún adversario repite los mensajes antiguos y puedan darse como recientes por el nodo receptor de los mismos

Tenacidad: debe ser capaz de establecer y mantener un nivel aceptable de seguridad en caso de que se vean comprometidos algunos nodos

Resistencia: es la capacidad de la red para evitar que el adversario obtenga el control completo de ella a partir de un ataque en la replicación de un nodo en el caso de que haya varios nodos implicados en la comunicación.

Eficiencia energética: el esquema de seguridad debe ser eficiente desde el punto de vista energético, con el fin de maximizar la vida de la red. Teniendo en cuenta que nos encontramos ante dispositivos con unos recursos muy limitados, éste se convierte en un requisito imprescindible.

1.3.2 Seguridad en las comunicaciones

La comunicación en el IoT debe ser protegida, con la finalidad de prestar los requisitos de seguridad descritos anteriormente. Utilizando mecanismos de seguridad estandarizados podemos garantizar una comunicación segura en diferentes capas de la pila IP; cada solución tiene sus pros y sus contras. En términos generales, la seguridad de las comunicaciones puede proporcionar seguridad desde el origen hasta el destino, u ofrecer seguridad entre saltos entre nodos vecinos. La Tabla 1 muestra una pila IoT con soluciones de seguridad estandarizadas en diferentes capas.

Capa de enlace: Seguridad en IEEE 802.15.4

Las redes 6LoWPAN utilizan el protocolo IEEE 802.15.4 como capa de enlace. La seguridad en la capa de enlace es la solución de seguridad actual para el IoT. Dicha seguridad protege la comunicación entre dos nodos vecinos, donde cada nodo de la ruta de comunicación debe ser de confianza. Para proteger todas las comunicaciones, se utiliza una única clave pre-compartida. En caso de que un atacante tenga acceso a la clave de un dispositivo, la seguridad per-hop (entre saltos) puede detectar la modificación de dicho mensaje en cada salto al contrario que en el uso de seguridad E2E, es decir, extremo a extremo (end-to-end), donde los paquetes modificados atravesarían toda la ruta hasta el destino, donde nos percataríamos de dicha modificación. Este tipo de seguridad es una buena opción para utilizarse en redes 6LoWPAN con el propósito de evitar el acceso no autorizado a través de un medio de radio, y para defenderse de los ataques lanzados con la finalidad de consumir los recursos de los nodos que conforman la red al evitar tráfico de paquetes modificados por intrusos. Aunque la seguridad de capa de enlace se limita a asegurar el enlace de comunicación entre dos dispositivos vecinos, es una opción flexible y puede operar con múltiples protocolos en las capas superiores.

Capa de red: Seguridad en el protocolo IP

En Internet y por lo tanto en el IoT, la seguridad en la capa de red es proporcionada por el conjunto de protocolos de seguridad IP (IPsec). IPsec en modo transporte proporciona seguridad de extremo a extremo con servicios de autenticación y protección contra ataques de repetición, además de confidencialidad e integridad. Al operar en la capa de red, IPsec puede ser utilizado con cualquier protocolo de capa de transporte

incluyendo TCP, UDP, HTTP, y CoAP. IPsec garantiza la confidencialidad y la integridad de la carga útil IP utilizando el protocolo Encapsulated Security Payload (ESP) [6], y la integridad de la cabecera IP además de la carga útil utilizando el protocolo Authentication Header (AH) [5]. El soporte para IPsec es obligatorio en el protocolo IPv6 [7, 8] lo que significa que todos los dispositivos preparados para IPv6 de forma predeterminada tienen la capacidad de utilizar IPsec, que puede ser activado en cualquier momento. Al ser una solución de capa de red, los servicios de seguridad proporcionados por IPsec son compartidos entre todas las aplicaciones que se ejecutan en una máquina en particular. Sin embargo, siendo obligatoria su soporte en IPv6, IPsec es una de las opciones más adecuadas para la seguridad E2E en el IoT, ya que en la mayoría se ejecuta sólo una aplicación en un dispositivo y las políticas de seguridad por defecto son suficientes para tales escenarios. Además, habilitar IPsec requiere relativamente poco esfuerzo para los desarrolladores de aplicaciones en los hosts de IPv6.

Capa IoT	**Protocolo IoT**	**Solución de seguridad**
Aplicación	CoAP	Definido por el usuario
Transporte	UPD	DTLS
Red	IPv6, RPL	IPSec, seguridad RPL
6LoWPAN	6LoWPAN	Ninguno
Enlace	IEEE 802.15.4	Seguridad IEEE 802.15.4

Tabla 1: Pila IoT con soluciones de seguridad estandarizadas

Capa de transporte: Seguridad en CoAP

Aunque IPsec se puede utilizar en el IoT, no está diseñado específicamente para protocolos como HTTP o CoAP. Para protocolos web, Transport Layer Security (TLS) o su antecesor Secure Sockets Layer (SSL) son las soluciónes de seguridad más comunes. El protocolo TLS orientado a la conexión sólo se puede utilizar a través de TCP; debido a la naturaleza con pérdida de las redes inalámbricas de baja potencia, es difícil mantener una conexión continua en las redes 6LoWPAN, por lo que no es demasiado apropiado la utilización de TCP. Existe una adaptación de TLS para UDP, el llamado Datagram TLS (DTLS) [9]. DTLS garantiza la seguridad E2E de diferentes aplicaciones en una máquina operando entre las capas de transporte y aplicación. DTLS además de

TLS que proporciona autenticación, confidencialidad, integridad y protección de repetición, también proporciona protección contra ataques de denegación de servicio (DoS) con el uso de cookies. Aunque DTLS proporciona seguridad E2E a nivel de aplicación, sólo se puede utilizar en el protocolo UDP; TLS se utiliza a través de TCP. El protocolo de internet seguro para el IoT, Secure CoAP (CoAPs), exige el uso de DTLS como la solución de seguridad subyacente para CoAP. Por lo tanto, es necesario para habilitar el soporte DTLS en el IoT.

1.3.3 Seguridad en la red

Incluso con la seguridad de las comunicaciones que protege los mensajes con los servicios de confidencialidad e integridad, las redes pueden sufrir otra serie de ataques, principalmente los que atentan contra su disponibilidad. Estos ataques están dirigidos a interrumpir el funcionamiento de las redes mediante la detención, por ejemplo, de la topología de enrutamiento o el lanzamiento de ataques de denegación de servicio (DoS). Los sistemas de detección de intrusiones (IDS) deben ser capaces de detectar impostores y actividades maliciosas en la red y se necesita el uso de firewalls para bloquear el acceso no autorizado a las redes. En el IoT, las redes 6LoWPAN son vulnerables a una serie de ataques desde Internet y desde el interior de la red. Además, las redes 6LoWPAN pueden convertirse en fuente de ataques contra los servidores de Internet, ya que es relativamente más fácil comprometer un nodo inalámbrico con recursos limitados que un host de internet típico.

RPL también es propenso a una serie de ataques de enrutado dirigidos a alterar la topología. El IoT con redes 6LoWPAN corriendo RPL, como se muestra en la Figura 1, forma una configuración de red diferente de los WSNs típicos. En el IoT, se supone que siempre hay accesible una 6BR, por lo tanto la seguridad de los mensajes de extremo a extremo se convierte en un requisito, y los nodos se identifican por una dirección IP única. En WSN típica no hay gestor centralizado ni un controlador, generalmente la seguridad se pasa por alto, y los nodos son identificables sólo dentro de una WSN. Teniendo en cuenta las nuevas características del IoT vale la pena investigar la aplicabilidad de IDS actuales y técnicas de firewall en el IoT, o el diseño de un nuevo IDS y firewall explotando las características y protocolos propios del internet de las cosas.

1.3.4 Seguridad de los datos

Es importante proteger no sólo la comunicación y las redes, sino también salvaguardar los datos sensibles almacenados en un dispositivo IoT. La mayoría de los dispositivos IoT son pequeños nodos de recursos limitados, conectados de forma inalámbrica, y prácticamente no es posible proteger físicamente a cada dispositivo, ni para protegerlos con las tecnologías resistentes a la manipulación basados en hardware como con el uso de tarjetas inteligentes o Módulos de Plataforma Segura (TPM) [10]. Existen varias soluciones basadas en software que pueden aplicarse a los datos almacenados en los nodos con el fin de protegerlos criptográficamente. Por ejemplo, Codo [11] es una solución de almacenamiento seguro diseñado para el sistema de Coffee File System [12] de Contiki. También existe la necesidad de diseñar nuevos mecanismos de almacenamiento seguro en el contexto del IoT.

2. Amenazas a la seguridad en redes de nodos

La mayoría de los ataques y amenazas contra la seguridad del usuario y los datos en 6LoWPAN son plausibles y su efecto puede llegar a ser muy destructivo. El análisis de la seguridad de 6LoWPAN comienza con el reconocimiento de las diferentes amenazas existentes en las respectivas capas del modelo ISO OSI. En esta sección, procederemos a clasificar y analizar las posibles amenazas por capas. El supuesto modelo de amenaza supone que el atacante es completamente capaz en todo momento de realizar cualquier actividad, excepto durante la fase de despliegue.

6LoWPAN es altamente susceptible a los ataques físicos, es decir, amenazas debidas a la reubicación física, destrucción del nodo y enmascaramiento. Mediante este tipo de ataques, uno o varios nodos 6LoWPAN pueden ser eliminados de forma permanente, por lo que las pérdidas son irreversibles. El ataque físico puede extraer secretos criptográficos de la circuitería asociada al nodo, modificar la programación de los mismos y puede hacer que un nodo malicioso tome el control. Esto puede ejecutarse modificando el código del nodo o cambiando el rol al que el nodo estaba destinado.

En el entorno 6LoWPAN, pueden llevarse a cabo varios tipos de ataques DoS en diferentes capas. En la capa física, los ataques de denegación de servicio pueden ser llevados a cabo por señales electromagnéticas (EM). Teniendo en cuenta los limitados recursos de los nodos de este tipo de redes, los ataques de denegación de servicio pueden ejecutarse con facilidad.

2.1 Ataques a la capa MAC

Los ataques a la capa MAC (Medium Aceess Control) implican colisión, el agotamiento y la falta de equidad. Al tratarse de dispositivos con recursos limitados, tratan de entrar en modo de reposo tan a menudo como sea posible, a fin de conservar la batería. Tales limitaciones abren la puerta al atacante, que puede enviar paquetes innecesarios con el propósito de agotar la batería de los nodos, el llamado *sleep deprivation torture*. Tal ataque también puede conducir a agotar la batería del nodo coordinador del PAN, debido a que los paquetes descendentes deben ser solicitados expresamente al coordinador del PAN, lo que mantiene ocupado tanto al coordinador como al nodo final al que va destinado el paquete.

Un ataque contra la disponibilidad de la red puede consistir en inundar la red, transmitiendo un gran número de paquetes de gran tamaño. En tal caso, el atacante

puede degradar el funcionamiento de la red y reducir drásticamente el rendimiento de la misma.

En esa misma línea se encuentra el denominado ataque *Sybil* (Figura 7), derivado de la suplantación de un nodo. El nodo atacante presenta varias identidades a la red, invalidando la información de los nodos legítimos y modificando la información de enrutado. Existe un método de protección para solventar esta amenaza, basado en la protección mutua de los nodos y la contabilidad de las tramas emitidas por cada uno, comprobando si existe alguna diferencia entre lo que se ha declarado en la red y lo que se ha contado.

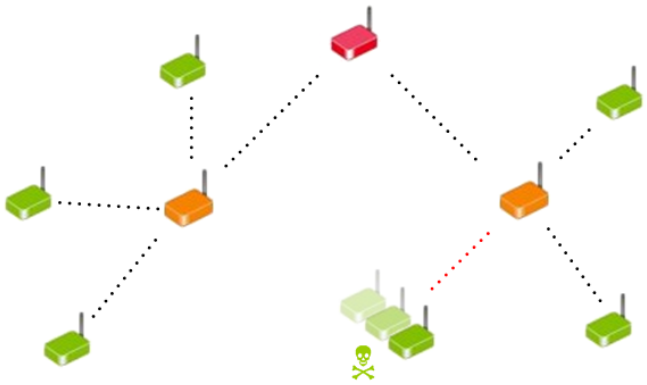


Figura 7: Ataque Sybil

Un dispositivo malicioso también puede atacar el proceso de distribución de claves llevado a cabo por el coordinador de la WPAN; éste anuncia los identificadores de los dispositivos que se encuentran en la red en texto plano, con el fin de llevar a cabo el intercambio de claves. Por lo tanto, el atacante puede enviar un paquete de solicitud de intercambio de claves con la ID de otro nodo. El objetivo de dicha solicitud es hacer que el coordinador de la red comience un proceso de intercambio de claves, mientras que el destinatario legítimo no se percata del mismo.

2.2 Ataques a la capa de red

Existen diferentes ataques contra la capa de red:

Falsificado, alterado o reproducción de información de enrutamiento:

En este tipo de ataque, el nodo malicioso utiliza la suplantación, alteración y / o reproducción de la información de enrutamiento para el intercambiado de mensajes

entre los nodos, en un intento de crear bucles de enrutamiento, atraer o repeler el tráfico de red, ampliar o acortar rutas, generar mensajes de error falsos, etc.

Expedición selectiva:

En este ataque, ilustrado en la Figura 8, el dispositivo malicioso podrá negarse a facilitar ciertos mensajes. En este caso, los dispositivos vecinos pueden concluir que el dispositivo malicioso ha fracasado y, por tanto, tratar de buscar otra ruta. Una forma más sutil de este ataque es cuando el dispositivo malicioso envía hacia adelante de forma selectiva, en cuyo caso los nodos vecinos no serán capaces de llegar a la conclusión de que se necesita otra ruta, por lo que volverán a enviar los paquetes de datos pasando por el dispositivo que trata de efectuar el ataque.

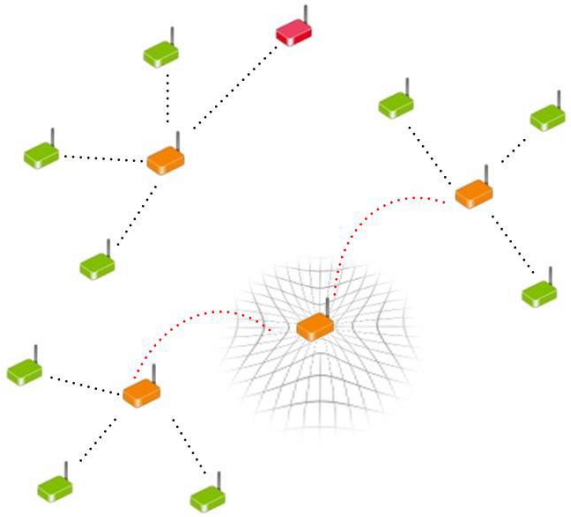


Figura 8: Expedición selectiva

Ataque de agujero negro:

Se trata de una mejora del anterior. En un ataque de agujero negro (Figura 9), los nodos maliciosos intentan obtener todo el tráfico de un área en particular. Con el fin de lanzar este tipo de ataques, el dispositivo malicioso puede escuchar las solicitudes de rutas y, a continuación, las respuestas a los nodos que solicitan las mejores rutas a la estación base. Para ello se utilizan enlaces fuera de banda de baja latencia para falsear la distancia entre los nodos, ya que se tuneliza la información. Una vez que el dispositivo malintencionado es capaz de insertarse entre los nodos de comunicación, es capaz de hacer cualquier cosa con los paquetes que pasan a través de él.

Los nodos malintencionados no requieren formar parte de la red, ni tener una identidad en la comunicación. Simplemente actuando como *relays* pueden modificar la información de rutado, aun empleando criptografía y autenticación.

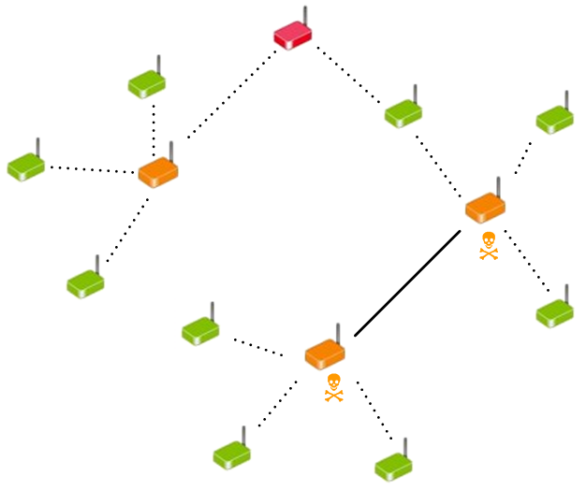


Figura 9: Ataque de agujero negro

Ataques al mecanismo de descubrimiento de vecinos (ND, Neighbor Discovery):

El descubrimiento de vecinos es el mecanismo por el cual un nodo que se incorpora a una red, descubre la presencia de otros en su mismo enlace, determina sus direcciones en la capa de enlace, localiza los routers y mantiene la información de conectividad acerca de las rutas a los vecinos activos.

Existe una versión modificada de este protocolo de IPv6 específica para WPAN. Sin embargo, ésta hereda amenazas que se aplica en el despliegue WPAN. Esto incluye un aviso de enrutador inseguro, que repercute en un posible ataque de denegación de servicio (DoS).

En este ataque, el nodo malicioso comienza con la fabricación de direcciones con el prefijo de subred y empieza a enviar paquetes de forma continuada a esta dirección. El último router al que llega el paquete dentro de la red de nodos está obligado a resolver estas direcciones mediante el envío de paquetes de solicitud de vecino. Un dispositivo que intente entrar en la red puede no ser capaz de obtener el servicio de descubrimiento de vecinos del último router al que llegan los paquetes, ya que se encontrará ocupado con el envío de otras solicitudes, que no encontrarán un nodo destino dado que el sufijo de la dirección es incorrecto, es decir, no corresponde a ningún nodo de la red. Este

ataque DDoS es diferente de los otros en que el atacante puede estar fuera de la red. El recurso atacado en este caso es la caché de vecinos conceptual, que se llena de intentos de resolver direcciones IPv6 que tienen un prefijo válido pero sufijo inválido. Este es un ataque de denegación de servicio (DoS).

Si el acceso al enlace está restringido a los nodos registrados y el router de acceso hace un seguimiento de los nodos que se han registrado para acceso en el enlace, el ataque puede ser solventado. Sin embargo, ningún mecanismo de este tipo está estandarizado actualmente.

Existe otra variación de este ataque en el que el atacante puede encontrarse fuera de la red. Mediante el uso de una antena de alta ganancia, el atacante es capaz de confundir a la red, presentándose como vecino de un número de nodos, cuando estos no tienen la capacidad de emisión suficiente para comunicarse con la antena. Se denomina *Hello flood* (Figura 10) y provoca el consumo de las baterías, ya que los nodos tratan de responder al anuncio, emitiendo señales al vacío.

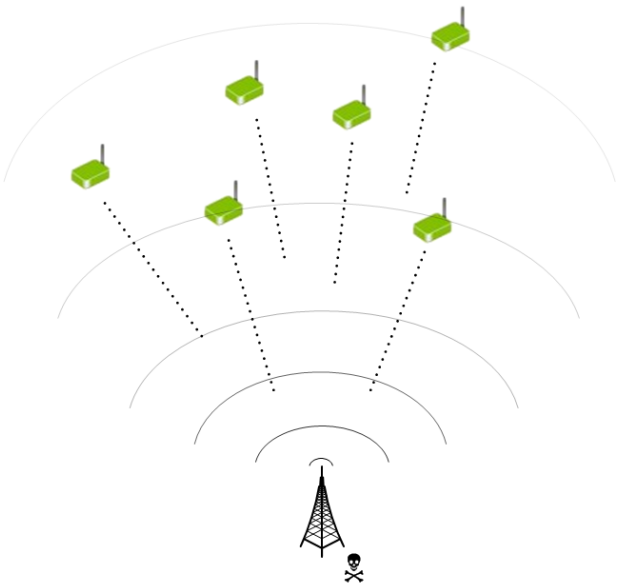


Figura 10: Hello flood

Ataque a la fragmentación

Para habilitar la transmisión de paquetes IPv6 grandes a través de tecnologías de tamaño limitado de enlace tales como IEEE 802.15.4, 6LoWPAN cuenta con soporte de fragmentación en la capa de adaptación. Sin embargo, el diseño del mecanismo de fragmentación de 6LoWPAN hace uso de un buffer, y el reenvío y tramitación de paquetes fragmentados son operaciones desafiantes al ser ejecutadas sobre dichos

dispositivos, que cuentan con recursos limitados. Los nodos malintencionados pueden enviar fragmentos duplicados o superpuestos. Debido a la falta de autenticación en la capa 6LoWPAN, los destinatarios no pueden distinguir estos fragmentos no deseados de los que sí lo son en el momento del reensamblado de paquetes. Por otra parte, los nodos encargados del reensamblado deben almacenar todos los fragmentos de un paquete y contar con un mecanismo de tiempo de espera para descartar los paquetes incompletos. Esto, sin embargo, puede hacer que la escasa memoria de un nodo pueda ser ocupada con los paquetes incompletos, debido a los fragmentos que faltan.

Así, los enlaces perdidos, bien por nodos maliciosos o bien por nodos desconfigurados, pueden llegar a bloquear el proceso de almacenamiento de paquetes fragmentados recibidos recientemente debido a la ocupación del buffer de los nodos por causa de fragmentos erróneos.

3. Soluciones de seguridad en redes 6LoWPAN

Dado el diverso número de amenazas de las que pueden ser víctimas las redes de nodos, nos encontramos ante la necesidad de llevar a cabo una serie de mecanismos de seguridad, con el fin de evitar que nuestro sistema sea víctimas de ataques. Para ello, se llevará un análisis de diversas soluciones de seguridad para las diferentes capas del protocolo; dichas medidas deberán estar adaptadas a los dispositivos ante los que nos encontramos, que cuentan con unos recursos limitados, tanto de cómputo y memoria como de batería.

3.1 Comunicación segura en 6LoWPAN utilizando IPsec comprimido

Las redes inalámbricas de sensores pueden integrarse con las infraestructuras existentes con base IP utilizando IPv6 sobre redes 6LoWPAN. Utilizando 6LoWPAN, los nodos de la red pueden comunicarse directamente con hosts activos IPv6 y, por ejemplo, el procesamiento de datos del sensor puede llevarse a cabo por servidores estándar. Por lo tanto, 6LoWPAN simplifica en gran medida el funcionamiento y la integración de WSNs en infraestructuras de internet existentes.

Las implementaciones de redes de sensores inalámbricos (WSNs) requieren una comunicación segura. Los host IPv6 soportan por defecto el uso de IPsec para dotar a la comunicación de seguridad. Por lo tanto, los flujos de datos entre hosts IPv6 y nodos sensores 6LoWPAN deben hacer provecho de las capacidades existentes y proteger el tráfico mediante IPsec. Por lo tanto, se propone añadir soporte IPsec para 6LoWPAN como se ilustra en la Figura 10.

IPsec define un encabezado de autenticación (AH) y una carga útil de seguridad de encapsulación (ESP). El AH proporciona integridad de datos y autenticación mientras que la cabecera ESP proporciona confidencialidad de datos, integridad y autenticación. Tanto AH como ESP o el uso de ambas pueden asegurar los paquetes IPv6 en tránsito. Dependiendo de la aplicación, serán necesarios unos servicios de seguridad u otros. 6LoWPAN utiliza técnicas de compresión de cabecera para lograr una reducción del tamaño de la cabecera de los paquetes. Al utilizar IPsec, en cada datagrama se incluirán las cabeceras AH y ESP; es por eso que debemos utilizar una técnica de compresión lo más eficiente posible.

Independiente de las tasas de compresión conseguidas en las cabeceras AH y ESP, es obvio que el apoyo IPsec en 6LoWPAN aumentará los tamaños de los paquetes al

incluirse encabezados adicionales a los datagramas. Sin embargo, mediante el uso de IPsec no necesitamos utilizar los mecanismos de seguridad que 802.15.4 nos proporciona en la capa de enlace, lo que hace que a su vez libere algo de espacio del encabezado existente.

A continuación trataremos:

- **Especificación 6LoWPAN-IPsec:** Definición de la especificación de IPsec para 6LoWPAN incluyendo definiciones para las cabeceras de extensión AH y ESP (Secciones 3.1.1 y 3.1.2).

- **Implementación 6LoWPAN-IPsec:** Se presenta la implementación de IPsec para redes 6LoWPAN. Se demuestra que es práctico y factible la protección de la comunicación WSN usando IPsec (Secciones 3.1.3 y 3.1.4).

- **Evaluación de la conjunción 6LoWPAN-IPsec:** Se realizará una evaluación de la implementación de IPsec en 6LoWPAN en términos de tamaño de código, los gastos generales de paquetes y el rendimiento de comunicación. Nuestros resultados muestran que los gastos generales son comparables a los gastos generales del uso de la seguridad 802.15.4 a nivel de enlace a la vez que ofrece el beneficio de una verdadera seguridad E2E (Secciones 3.1.3 y 3.1.4).

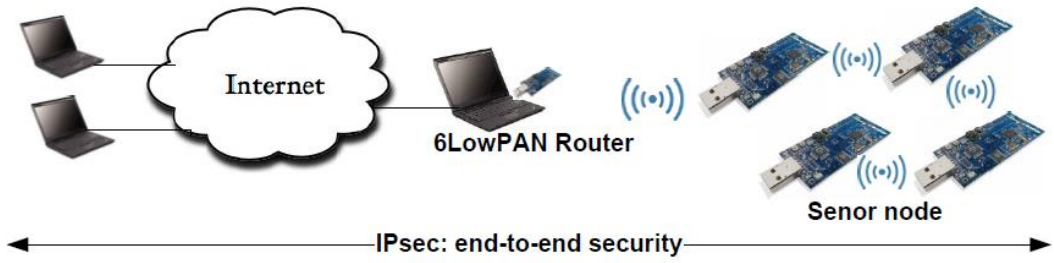


Figura 11: Seguridad IPSec.

IPSec proporciona una comunicación segura entre los nodos que conforman la red y el router (6BR). IPSec dota a la red de seguridad E2E

3.1.1 IPv6 e IPSec

Con el avance del IoT y la visión de cualquier aparato físico como un dispositivo conectado a internet, se espera que todos ellos lo hagan utilizando IP [13]. Dado el gran número de dispositivos, esto requiere el uso de IPv6 [14], una versión del Protocolo de

Internet que aumenta el tamaño de la dirección de 32 bits a 128 bits con respecto a su predecesor. Además del aumento del espacio de direcciones IPv6 ofrece en comparación con IPv4 un formato de cabecera simplificada, soporte mejorado para extensiones y opciones, la capacidad de flujo de etiquetado y la capacidad de autenticación y privacidad.

Tanto la autenticación como la privacidad en IPv6 es proporcionada por IPsec . IPsec define un conjunto de protocolos para asegurar la comunicación IP: los protocolos de seguridad Authentication Header (AH) y Encapsulating Security Payload (ESP), los algoritmos de autenticación y cifrado, mecanismos de intercambio de claves y las llamadas asociaciones de seguridad (SA) [8]. Una SA especifica cómo un flujo IP en particular debe ser tratado en términos de seguridad. Para establecer las SA, IPSec especifica una clave pre-compartida así como un protocolo de intercambio de claves, Internet Key Exchange (IKE).Esto significa que todos los nodos habilitados para funcionar en internet utilizando IPv6 soporta el uso de claves pre-compartidas. El protocolo IKE utiliza criptografía asimétrica que se supone que es el computacionalmente costoso para nodos limitados.

La finalidad de AH es proporcionar integridad sin conexión y autenticación del origen de datos para los datagramas IP, así como protección contra repeticiones. El Message Authentication Code (MAC) se utiliza para producir la información de autenticación. El MAC se aplica a la cabecera IP, encabezado AH y carga IP. La cabecera de autenticación se muestra en la Figura 11. Todos los hosts deben soportar el algoritmo de código de autenticación de mensajes basado en hash AES-XCBC-MAC-96 [16] para calcular los datos de autenticación, que tiene un tamaño de 12 bytes. Por lo tanto, como se muestra en la Figura 12, el encabezado básico AH tiene un tamaño de 24 bytes.

ESP proporciona autenticidad del origen, integridad y confidencialidad de los paquetes IP. ESP se utiliza para cifrar la carga útil de un paquete IP pero en contraste con AH, no proporciona seguridad a la cabecera IP.

Si se aplica ESP la cabecera IP es seguido por la cabecera de extensión ESP IP que contiene la carga útil cifrada. ESP incluye un SPI que identifica el SA usado, un número de secuencia para evitar ataques de repetición, la carga útil cifrada, el padding, variable, que puede ser requerido por algunos bloques, una referencia a la siguiente cabecera y opcionalmente datos de autenticación.

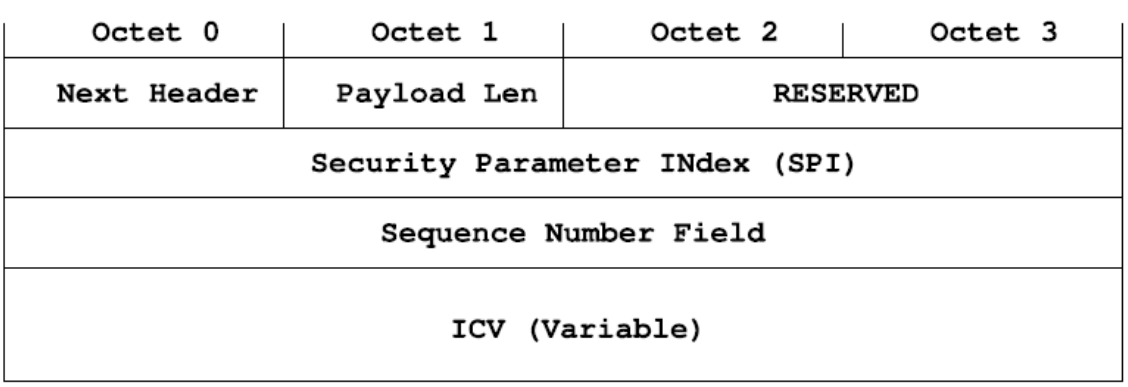


Figura 12: Cabeceras de autenticación (AH) IP

El cifrado en ESP incluye datos de carga útil, el padding, la longitud de este padding y la referencia a la siguiente cabecera; si se elige el uso de autenticación, incluye todos los campos de cabecera en el ESP. Si asumimos obligatorio AES-CBC como un algoritmo de cifrado ESP con la alineación de bloques perfecta tendremos una sobrecarga de 18 bytes (10 bytes para ESP y 8 bytes para vector de inicialización). Si la autenticación adicional utiliza AESXCBC-MAC-96, la sobrecarga ESP es de 30 bytes, ya que la longitud mínima de AES-XCBC-MAC-96 es de 12 bytes.

Los protocolos AH y ESP soportan dos modos de funcionamiento diferentes: el modo de transporte y el modo túnel. En el modo de transporte la cabecera IP y la carga útil se fijan directamente como se describió anteriormente. En el modo de túnel, una nueva cabecera IP se coloca en frente de las funciones originales de paquetes y de seguridad IP se aplican a la (túnel) paquete IP encapsulado. En el contexto 6LoWPAN, el modo túnel parece no ser práctico, ya que las cabeceras adicionales aumentarían aún más el tamaño de paquete.

3.1.2 Mecanismos de compresión 6LoWPAN

6LoWPAN tiene por objeto la integración de las infraestructuras existentes de IP y las redes de sensores especificando cómo los paquetes IPv6 deberán transmitirse a través de una red IEEE 802.15.4. El tamaño máximo de paquete en la capa física de 802.15.4 es de 127 bytes y el tamaño máximo de encabezado de la trama es de 25 bytes. Por lo tanto, un paquete IPv6 ha de encajar en 102 bytes. Dado que los encabezados de los paquetes consumen 48 bytes de los 102 bytes disponibles, es obvio que los mecanismos de compresión de cabecera son un componente esencial en 6LoWPAN.

HC13 [15] propone dos mecanismos de compresión basada en el contexto: LOWPAN_IPHC (nos referiremos al mismo como IPHC) para la compresión de la

cabecera IPv6 y LOWPAN_NHC, (al que nos referiremos como NHC) para la compresión de la cabecera siguiente. La cabecera IPHC aparece en la Figura 13:

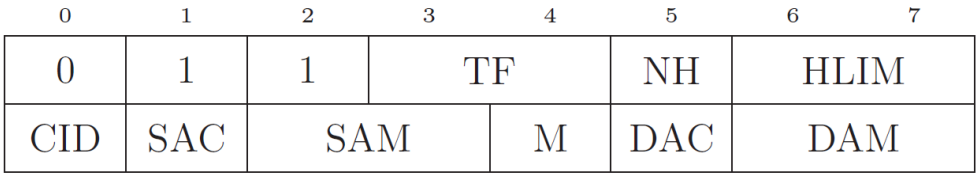


Figura 13: Formato de la cabecera IPHC

Dicha cabecera contiene información acerca de cómo se ha llevado a cabo la compresión de la cabecera IP Contiene los siguientes campos:

- TF (*Trafic Class, Flow Label*): Indican si se suprime el campo Clase de tráfico y/o Etiqueta de flujo de la cabecera IPv6.

- NH (*Next Header*): Indica si se suprime el campo NH de la cabecera IPv6 y se usa NHC.

- HLIM (*Hop Limit*): Indica si el campo Hops de la cabecera IPv6 se comprime además de limitar el contador a un valor máximo de 1, 64 o 255.

- CID (*Context Identifier Extension*): Informa sobre el uso opcional del campo de extensión de identificador de contexto.

- SAC (*Source Address Compression*): Indica si la dirección IPv6 de la fuente va a usar compresión *stateful* (basada en contexto) o compresión *stateless*.

- SAM (*Source Address Mode*): En función del bit comentado anteriormente, se tienen 4 modos de compresión para cada tipo (stateless o stateful). Las direcciones IPv6 pueden no comprimirse, comprimirse a 64 bits, 16 bits, o comprimirse completamente.

- M (*Multicast Compression*): Indica si la dirección IPv6 destino es *multicast*.

- DAC (*Destination Address Compression*): Similar al SAC pero para el destino.

- DAM (*Destination Address Mode*): Similar al SAM pero para el destino. Si el bit M está activado se sigue una compresión idónea (48, 32 u 8 bits) para direcciones *multicast*.

Para una compresión eficiente de la cabecera IPv6, IPHC elimina campos de la cabecera IP que los nodos de la red conocen de manera implícita.

La cabecera IPHC tiene una longitud de 2 bytes de los cuales 13 bits se utilizan para la compresión de cabecera. Sin comprimir los campos de cabecera IPv6 siguen la codificación IPHC en el mismo orden como aparecerían en la cabecera IPv6 normal. En un escenario multihop IPHC puede comprimir la cabecera IPv6 a 7 bytes. 6LoWPAN especifica que el tamaño de NHC debe ser múltiplo de octetos, por lo general de 1 byte, donde los primeros bits de longitud variable representa un ID de NHC y los bits restantes se utilizan para codificar / comprimir cabeceras. 6LoWPAN ya define NHC para UDP e IP Extension Header [15]

3.1.3 LoWPAN NHC

IPsec requiere la compresión de cabeceras para mantener los tamaños de paquete razonables para su uso en redes 6LoWPAN. Sin embargo, no existe ninguna especificación para las codificaciones de cabecera de extensión AH y ESP. A continuación, veremos una solución a dichas codificaciones de cabeceras.

Como se ha descrito anteriormente, HC13 define un mecanismo de compresión basado en el contexto usando IPHC para la compresión de la cabecera IP y NHC para la compresión de la siguiente cabecera. El mecanismo de compresión NHC ya definido para la cabecera IP se puede utilizar para codificar las cabeceras de extensión AH y ESP. La cabecera opcional IPv6 NHC, que proporciona mecanismos de extensión de IPv6 consiste en un octeto donde tres bits (los bits 4, 5 y 6) se utilizan para codificar la IPv6 Extension Header (EID).

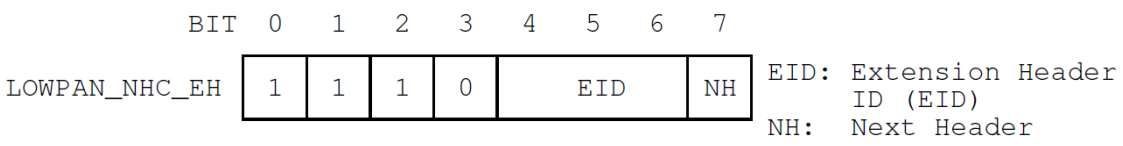


Figura 14: Codificación LOWPAN_NHC_EH para la cabecera de mecanismos extensión IPv6

- EID (*Extension Header ID*): Indica el tipo de información añadida que va en el campo sin compresión de IPv6 NHC: opciones de enrutamiento, fragmento, salto, movilidad...
- NH (*Next Header*): Indica si el encabezado siguiente se comprime usando NHC.

De los ocho valores posibles para el campo EID, el valor de seis de ellos está especificado por HC13. Los dos restantes (101 y 110) se encuentran reservados. Por lo tanto, dichos huecos se pueden utilizar para la codificación de AH y ESP. Esto hace necesario el cambio del último bit en la cabecera IPv6 de mecanismos de extensión a 1, indicando que la siguiente cabecera (AH o ESP) se codificará utilizando NHC.

Codificación LOWPAN_NHC_AH

Se propone el uso de la codificación NHC para AH. El formato de dicha cabecera se muestra en la Figura 15:

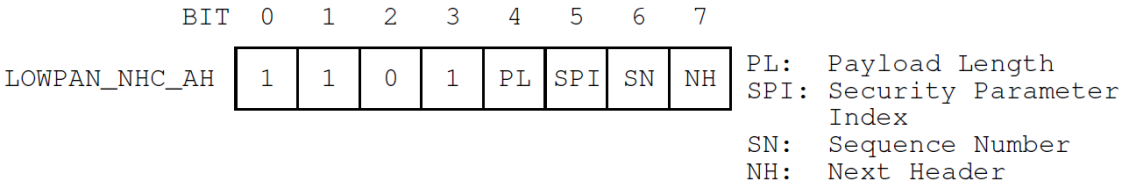


Figura 15: Codificación LOWPAN_NHC_AH para la cabecera de autenticación IPv6

La función de cada una de las celdas se define:

- Los primeros cuatro bits en NHC_AH representan el ID NHC que se define para AH y se modifica a 1101. Esto es necesario para cumplir con el estándar 6LoWPAN.

- PL: Si su valor es 0, la longitud de la carga útil se omite. Esta longitud puede obtenerse por el valor de SPI ya que la longitud de la información de autenticación depende del algoritmo utilizado y es fijo para cualquier tamaño de entrada. Si es 1, la longitud se incluye después de la cabecera NHC_AH.

- SPI: Si su valor es 0, el SPI se utiliza el valor por defecto de la red de sensores y el campo SPI se omite. El valor por defecto del SPI será 1. Esto no quiere decir que todos los nodos de la red compartan la misma asociación de seguridad (SA), sino que cada nodo tiene su propio SA identificado como SPI 1. En este caso, el SPI se especifica explícitamente.

- SN: Si su valor es 0, se utiliza un número de secuencia de 16 bits se y los 16 restantes se toman como 0. Si su valor es 1, se utilizan 32 bits como número de secuencia, que se define explícitamente.

- NH: Si está a 0, el siguiente campo de la cabecera AH se utilizará para especificar la siguiente cabecera y se hará en la propia línea. Si el valor es de 1, el campo de siguiente cabecera en AH se salta. La siguiente cabecera se codificará utilizando NHC.

La longitud mínima de una cabecera AH estándar utilizando HMC-HA1-96 es de 24 bytes. Después de la compresión, obtenemos un tamaño de 16 bytes. La siguiente figura muestra un paquete comprimido IPv6/UDP seguro con AH y HMAC-SHA1-96. En la Figura 16 puede verse un ejemplo de un paquete IPv6/UDP comprimido utilizando AH:

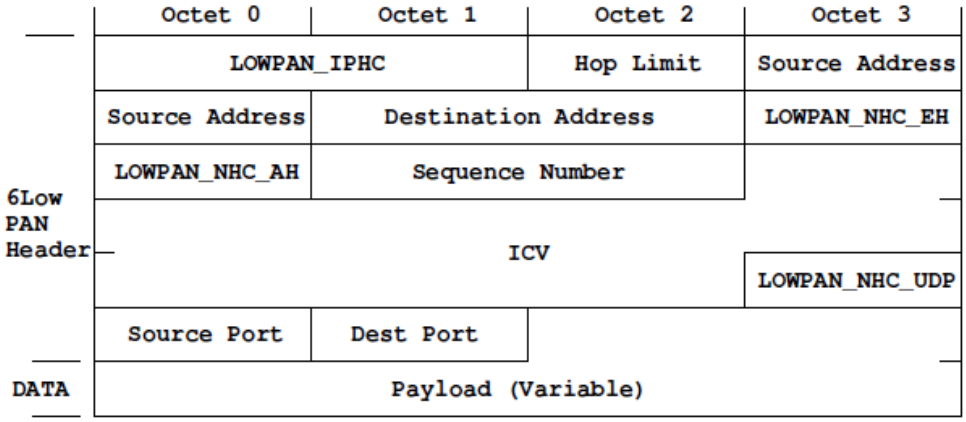


Figura 16: Paquete comprimido IPv6/UDP utilizando AH.

Codificación LOWPAN_NHC_ESP

También la codificación NHC para ESP codifica el índice de parámetros de seguridad, el número de secuencia, los siguientes campos de cabecera y el ID de NHC para ESP. En el caso de ESP, se propone 1110 como ID NHC. El formato se muestra en la Figura 17:

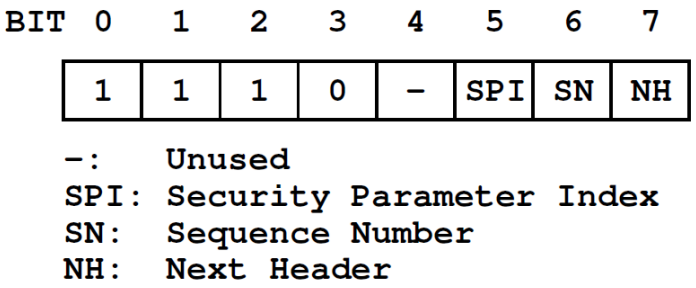


Figura 17: Codificación Codificación LOWPAN_NHC_ESP para ESP IPv6

- Los cuatro primeros bits representan el NHC ID que se define para ESP. En nuestro caso será 1110.

- El siguiente bit no se utiliza. Se ha mantenido el bit para mantener cierta similitud con la codificación de la cabecera AH. También puede utilizarse si el índice del parámetro de seguridad (SPI) lo requiere.

- Si SPI es 0, se utiliza el SPI por defecto de la red 802.15.4 y el campo SPI es omitido. El valor de SPI por defecto es 1, lo que no quiere decir que todos los nodos usen la misma asociación de seguridad (SA), sino que todos los nodos tienen un único SA predefinido, identificado como 1.
  Si SPI es 1, se utilizan los 32 bits que indican el SPI.

- Si SN es 0, se utilizan los primeros 16 bits del número de secuencia. Los demás se asumen como ceros.
  Si SN es 1, los 32 bits del número de secuencia se incluyen.

- Si NH es 0, el campo siguiente cabecera en ESP se utilizará para especificar la siguiente cabecera.
  Si NH es 1, la siguiente cabecera se codificará utilizando NHC. En el caso de ESP no podemos saltarnos la siguiente cabecera a menos que el host final sea capaz de ejecutar la compresión/descompresión 6LoWPAN y el codificado/decodificado.

Destacar que la sobrecarga mínima sin autentificación, utilizando AES-CBC y un alineamiento perfecto de bloques es de 18 bytes. Tras la compresión, dicha sobrecarga se reduce a 12 bytes. ESP con autenticación (HMAC-SHA1-96) tiene una sobrecarga de 30, que se reduce a 24 si se utiliza la compresión.

3.1.4 Combinación de AH y ESP

Tanto AH como ESP pueden utilizarse de manera simultánea, así como haciendo ambos uso de NHC como método de compresión; sin embargo, obviamente es más eficiente desde el punto de vista del tamaño de las cabeceras el uso de ESP con autentificación en lugar de añadir ambas cabeceras a un mismo paquete. El tamaño de paquete es importante, ya que puede ser la diferencia entre poder utilizar IPsec en WSNs o ser imposible su implementación debido a las limitaciones de los nodos.

3.1.5 Requisitos en el host final

Los nodos 6LoWPAN que utilizan AH pueden comunicarse directamente con hosts convencionales en internet sin modificar IPsec. Cuando se utiliza ESP en nodos también disponen de esta capacidad. Sin embargo, si se utiliza ESP no es posible comprimir las cabeceras de capas superiores, como UDP. Una pasarela 6LoWPAN entre redes de sensores y la red IP no puede acceder y descomprimir la cabecera UDP encriptada. Para habilitar la compresión UDP con ESP se necesita especificar un nuevo algoritmo de cifrado para ESP que sea capaz de realizar la compresión y el cifrado de cabeceras UDP. De nuevo, si esta optimización utiliza IPsec, los host deben incluir y soportar este protocolo de cifrado.

3.1.6 Evaluación y resultados de IPsec en redes 6LoWPAN

A continuación, se cuantifica el rendimiento del uso de IPSec en redes 6LoWPAN. Tras realizar una descripción de la implementación llevada a cabo por Raza et al.[23], se evaluará el impacto del uso de IPSec en cuanto a memoria, tamaño de paquetes, consumo de energía y rendimiento utilizando diferentes configuraciones.

De esta forma, se llegará a la conclusión de si realmente es conveniente el uso de dicha tecnología para la protección de la red, observando con datos objetivos la repercusión del uso de IPSec sobre redes 6LoWPAN.

3.1.6.1 Implementación y configuraciones experimentales

La implementación de AH y ESP de IPSec se llevó a cabo sobre el sistema operativo Contiki [17]. Para ello, llevaron a cabo una modificación de la pila IP de Contiki que ofrece la funcionalidad 6LoWPAN, introduciendo las codificaciones NHC EH, NHC AH y NHC ESP en la capa SICLoWPAN, el componente 6LoWPAN de la pila IP.

IPSec utiliza las implementaciones de SHA1 y AES de MIRACL [24], una librería open source e implementa todos los modos de operación criptográficos necesarios para las funcionalidades que IPSec ofrece. Para AH, se utiliza HMAC-SHA1-96 y AES-XCBC-MAC-96. Para ESP, se utiliza AES-CBC para el encriptado y HMAC-SHA1-96 para la autenticación.

Además en ESP también se incluye AES-CTR para el encriptado y AES-XCBC-MAC-96 para la autenticación. Dicha implementación utiliza claves pre-compartidas para establecer los SAs con los que trabajar con todos los nodos IPv6 en internet y un mecanismo precompartido obligatorio.

La evaluación está formada por cuatro sensores Tmote Sky [19], un puente software 6LoWPAN y una máquina Linux corriendo un sistema operativo Ubuntu con IPSec hablilitado. Los cuatro sensores forman una red multihop; ejecutan una única aplicación que escucha un puerto UDP. Cuando se recibe un paquete, es procesado por la capa 6LoWPAN, interpretado por la capa IPsec y por µIP. Luego la carga útil es reenviada a la aplicación. Como respuesta, un nuevo datagrama del mismo tamaño es enviado de vuelta, siguiendo el proceso contrario. IPSec se utiliza para garantizar la seguridad E2E entre el sensor y el host de internet. Se utiliza NullMAC de Contiki en los experimentos, de manera que los nodos mantienen sus señales de raido activas todo el tiempo, con la finalidad de evitar el retardo en la activación.

3.1.6.2 Impacto del uso de IPSec en memoria

Para el estudio de la repercusión en la memoria del nodo al utilizar IPsec utilizaremos la Tabla 2, donde se utiliza el sistema Contiki, así como µIP y SICSLoWPAN.

Configuración	ROM (kB)		RAM (kB)	
	Total	Diferencia	Total	Diferencia
Sin IPSec	32.9	-	8.0	-
AH con MAC-SHA1-96	36.8	3.9	9.1	1.1
AH con XCBC-MAC-96	38.4	5.5	8.5	0.5
ESP con AES-CBC	41.4	8.5	8.3	0.3
ESP con AES-CTR	39.8	6.9	9.1	0.3
ESP con AES-XCBC-MAC-96	39.8	6.9	8.3	0.3
ESP con AES-CBC + AES-XCBC-MAC-96	41.9	9.0	8.3	0.3

Tabla 2: La tabla muestra que AH y ESP consumen 3.9kB y 9kB para los algoritmos IPsec obligatorios.

En el caso de la memoria ROM, vemos que la utilización de la misma se encuentra entre 3.8kB (AH con HMAC-SHA1) y 9kB (ESP con AES-CBC + AES-XCBC-MAC), por lo que se encuentra siempre por debajo de 48kB, el tamaño de la memoria ROM flash de Tmote Sky que recordemos es el tipo de nodo que se ha simulado.

El impacto en la memoria RAM se calcula como la suma de todos los datos globales y el uso de la pila en tiempo de ejecución. Con un uso adicional de 1.1 kB, la configuración AH con MAC-SHA1-96 se muestra como la más consumista desde el punto de vista de la memoria RAM. Utilizando otros modos de operación, el uso de la memoria RAM se encuentra entre un rango de 0.3 y 0.5 kB, lo que nos lleva a la conclusión de que AH IPSec y ESP puede utilizarse en dispositivos de bajos recursos.

3.1.6.3 Comparación del incremento de tamaño de la cabecera

Actualmente, la protección en WSN se lleva a cabo mediante el uso de seguridad a nivel de enlace 802.15.4. Sin embargo, al contrario que en el uso de IPSec, dicha seguridad se garantiza a nivel de salto y no extremo a extremo (E2E). La Tabla 3 muestra el incremento en el tamaño de cabecera en el uso de IPSec sin comprimir, IPSec comprimido y seguridad 802.15.4.

Configuración	IPSec sin comprimir		IPsec comprimido		802.15.4	
	Modo	Bytes	Modo	Bytes	Modo	Bytes
Autenticación AH	HMACSHA1- 96	24	HMACSHA1-96	16	AES-CBC-MAC- 96	12
Encriptado ESP	AES-CBC	18	AES-CBC	12	AES-CTR	5
Encriptado y autenticación ESP	AES-CBC y HMAC-SHA1-96	30	AES-CBC y HMAC-SHA1-96	24	AES-CCM-128	21

Tabla 3: Incremento en el tamaño de la cabecera con diferentes configuraciones de seguridad.

En la tabla se puede apreciar como el tamaño de paquete utilizando IPsec comprimido es parecido al tamaño de paquete en el uso de la seguridad 802.15.4, mientras que el primero proporciona seguridad E2E.

Al utilizar la seguridad a nivel de enlace, el incremento en el tamaño del datagrama destinado a la autenticación es el tamaño que ocupa la MAC. En el caso del uso de IPSec utilizando AES-XCBCMAC-96, el tamaño de la MAC es de 12 bytes. El campo adicional AH incrementa la cabecera en otros 12 bytes, lo que hace un total de 24. Sin embargo, utilizando el la compresión de cabecera IPSec definida, dicho tamaño adicional se reduce a 16 bytes. Por lo tanto, la seguridad E2E que IPSec comprimido proporciona tiene un incremento en el tamaño de la cabecera de 4 bytes.

Si se requiere únicamente la encriptación del mensaje, 802.15.4 proporciona AES-CTR con un incremento en el tamaño de la cabecera de 5 bytes. IPSec con ESP y AES-CBC la incrementa en 18 bytes. La compresión de dicho protocolo la reduce a 12, lo que se traduce en un incremento de 7 bytes con el fin de la proporción de la seguridad E2E que IPSec proporciona.

Con el uso de AES-CCM-128, el incremento de la cabecera 802.15.4 es de 21 bytes, mientras que con IPSec ESP sería de 30 bytes, que se ven reducidos a 24 bytes. La diferencia entre ambos es de 3 bytes, a lo que se debe añadir la seguridad extremo a extremo de IPSec.

A todo esto debe añadirse un dato a favor del uso de IPSec; cuando se envían datagramas IP grandes, se utiliza la fragmentación en la capa intermedia entre la de enlace y de red de 6LoWPAN. Utilizando seguridad a nivel de enlace, el incremento de la cabecera se paga en cada uno de los fragmentos, mientras que la cabecera IPSec se incluye una vez para todos los fragmentos de un datagrama, Esto se traduce en que en el caso en que se necesiten dos o más fragmentos, IPSec ofrece un incremento de cabecera menor que en el caso de la seguridad a nivel de enlace proporcionada por IEEE 802.15.4.

3.1.6.4 Rendimiento de la criptografía

En esta sección, se llevará a cabo una evaluación de la eficiencia de los diferentes algoritmos criptográficos de la implementación IPSec.

El Gráfico 1 muestra los rendimientos y consumos de energía de los diferentes modos de operación dependiendo del tamaño de la carga útil de los datagramas IP.

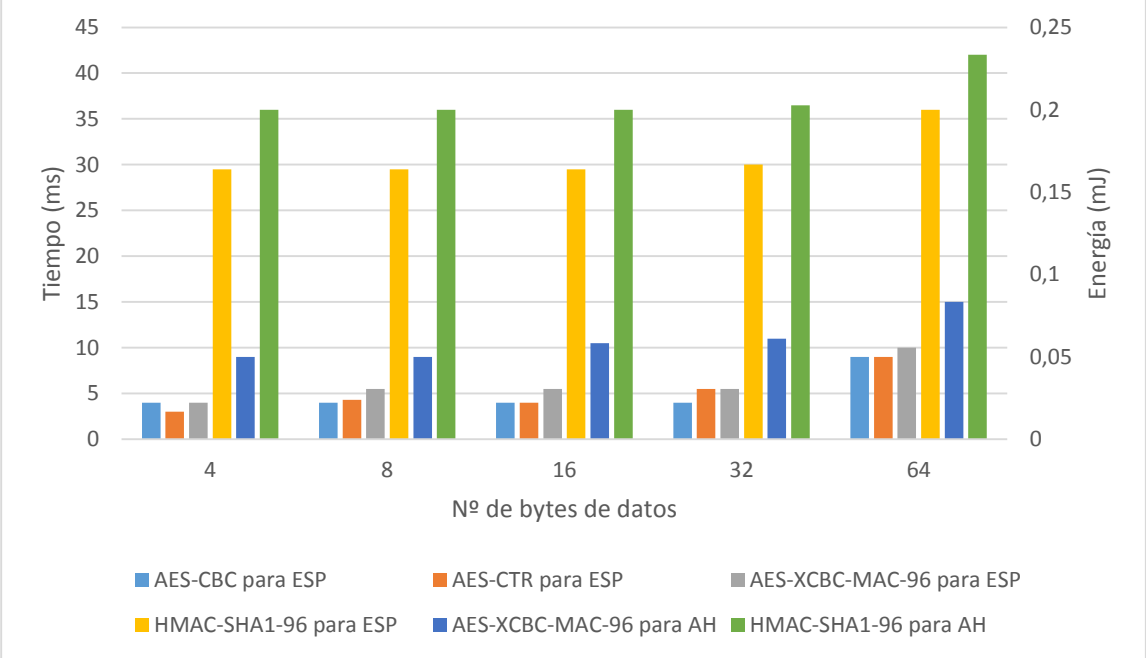


Gráfico 1: Comparación de diferentes algoritmos criptográficos

El gráfico muestra que AES-CBC para ESP y AES-XCBC-MAC-96 para ESP son los más eficientes desde el punto de vista del rendimiento y el consumo de energía.

Vemos como para el encriptado (ESP), tanto AES-CBC como AES-CTR muestran un rendimiento similar. En cuanto a la autenticación, obviamente el resultado es mayor, debido al mayor número de datos de la cabecera AH. Aun así, el coste es reducido y su crecimiento es lineal con respecto al tamaño de los datos a procesar. El uso de HMAC-SHA1-96 no se presenta como una alternativa competente dado al gran tiempo que requiere para el procesamiento cuando se trata con pequeñas cantidades de datos.

3.1.6.5 Consumo de energía

La instauración de un sistema de seguridad en el IoT supone un coste de energía por parte de los nodos que conforman la red que debe ser medido y estudiado, dada la naturaleza de los dispositivos que, como se ha mencionado en varias ocasiones, suelen estar provistos de baterías.

En la recogida de datos, se tienen en cuenta los ticks de CPU, es decir, las operaciones que realiza el procesador desde la recepción del primer fragmento de un mensaje, cuando comienza el desencriptado en la capa de enlace, parando el recuento cuando el desencriptado del último paquete de la capa de enlace se ha completado, por lo que se ignora el tiempo de conexión entre paquetes. Se mide utilizando AH, ESP y sin utilizar IPSec. Hay que tener en cuenta que ESP se está utilizando dotando a la red de autenticación y de cifrado. Aunque la energía consumida por IPSec es notablemente más alta que en el caso en el que no se utiliza, hay destacar que es insignificante en comparación con el consumo de energía en los nodos por otro tipo de actividades, como el consumo que se produce en los chips de radio. En el peor de los casos, AH con un tamaño de paquete de 64 bytes de datos consume una energía de 0.5 mJ. El chip de radio de Tmote Sky consume la misma cantidad de energía tras 8 ms escuchando.

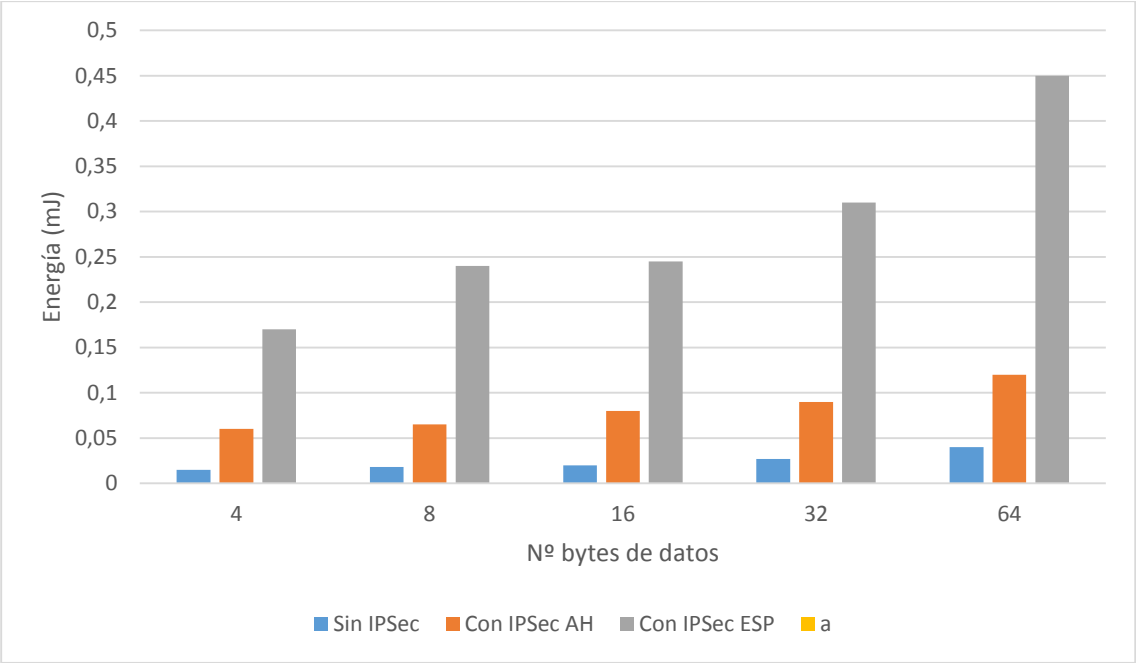


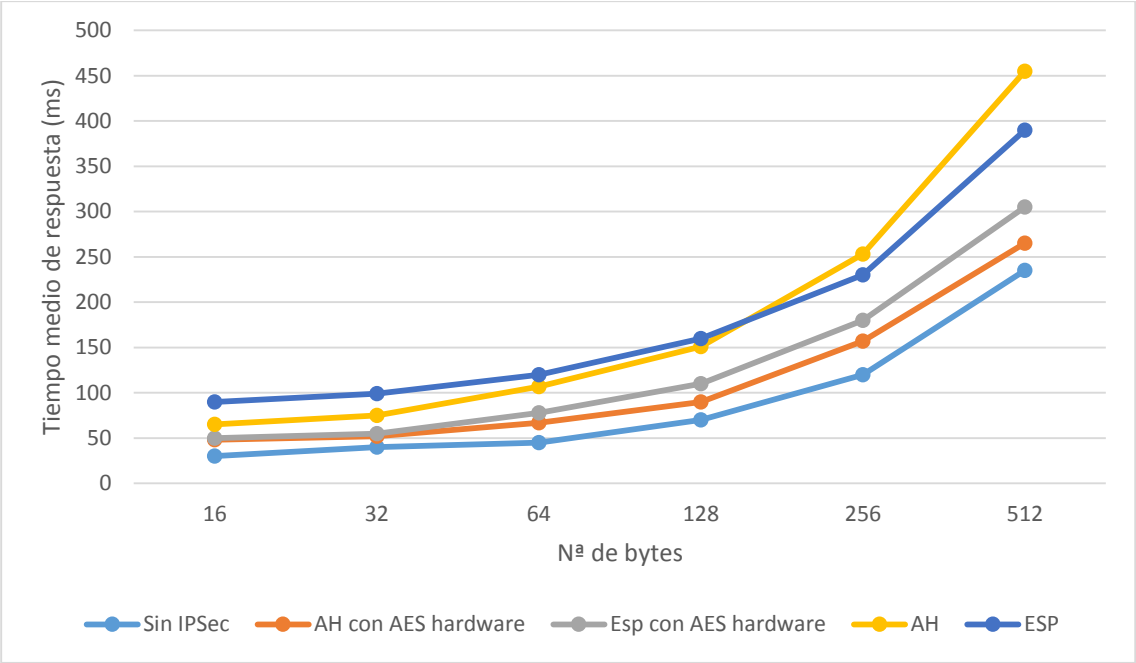
Gráfico 2: Consumo de energía en el uso de IPSec

El consumo de energía sin utilizar IPSec es menor, ascendiendo con el uso de AH y ESP se consolida como el mayor. Sin embargo, comparado con otras actividades de los nodos es insignificante.

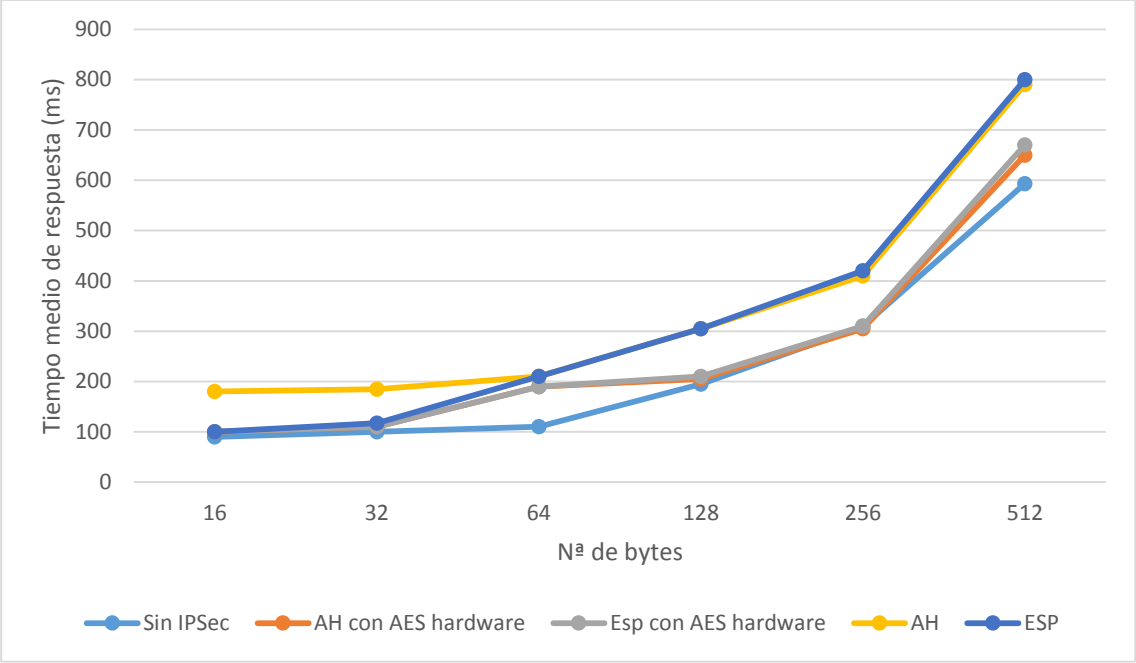
3.1.6.6 Tiempo de respuesta

Se entiende como tiempo de respuesta al tiempo transcurrido desde el envío de un mensaje hasta que se recibe una respuesta por parte del receptor.

Se realiza un estudio del tiempo de respuesta con distintos modos de funcionamiento de IPSec, así como sin utilizarlo, con diferentes tamaños de mensajes y con un único salto y salto múltiple.



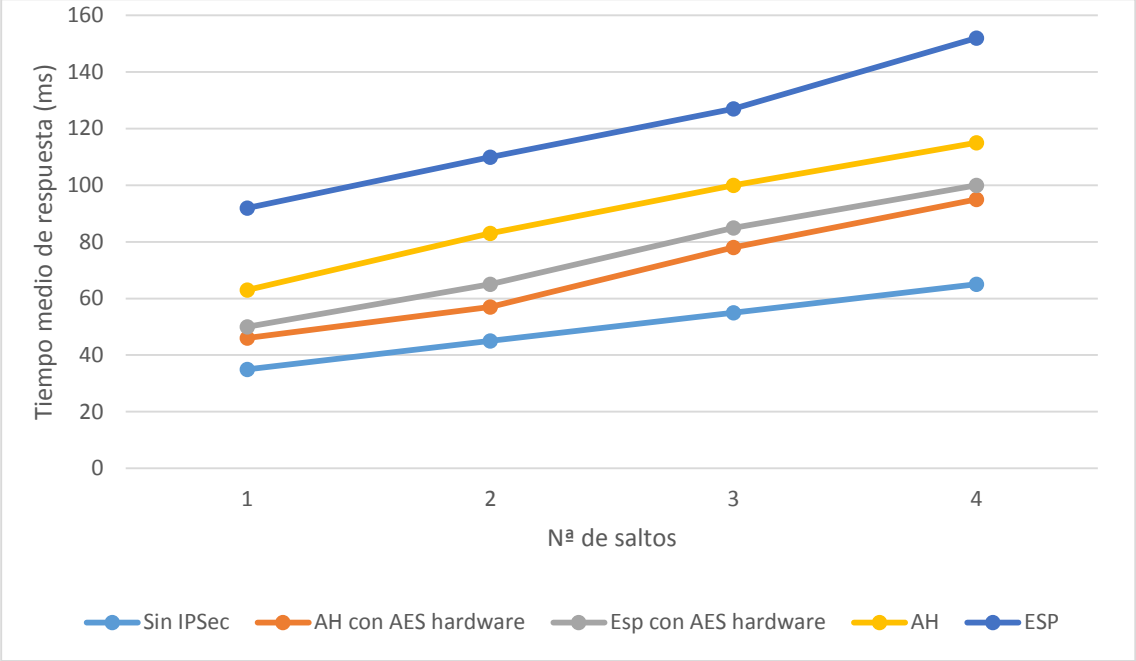
a) Salto simple con diferentes tamaño de datos



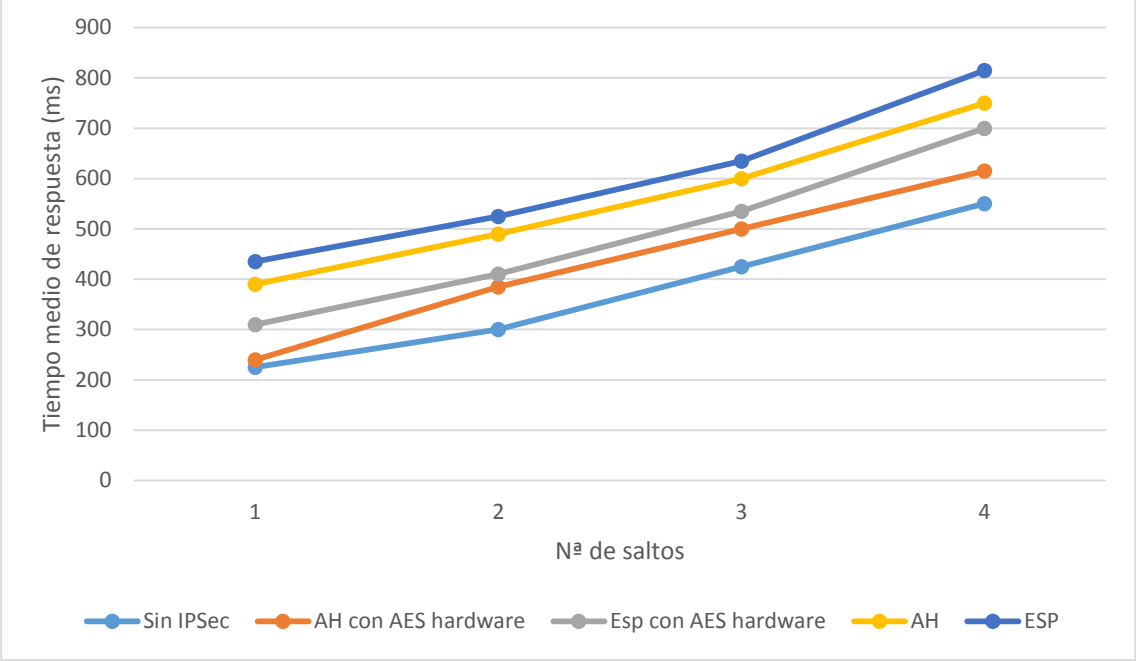
b) Salto múltiple (4) con diferente tamaño de datos.

Gráfico 3 (a y b): Relación tiempo de respuesta y tamaño del datagrama con AH, ESP y sin IPSec.

En el Gráfico se observa que ESP es más rápido que AH para datagramas pequeños debido a que procesa los 40 bytes de la cabecera IP. Sin embargo, AH es más rápido para tamaños grandes ya que procesa la autenticación pero no el cifrado.



a) Salto múltiple con un tamaño de datos de 16 bytes



b) Salto múltiple con un tamaño de datos de 512 bytes

Gráfico 4 (a y b): Relación tiempo de respuesta versus número de saltos utilizando AH, ESP y sin utilizar IPSec

En el gráfico vemos como el coste del uso de IPSec se mantiene constante sin importar el número de saltos.

Podemos observar como para un tamaño de datos dado, el coste del uso de AH o ESP se mantiene constante, sin importar del número de saltos. Esto es debido a que para los nodos intermedios, el coste de reenviar datos con y sin IPSec es exactamente el mismo, por lo que el coste computacional se produce en los nodos finales. En el peor de los casos, el incremento del tiempo de respuesta, utilizando un tamaño de datos de 512 bytes, es de 216 ms.

3.1.6.7 Mejoras utilizando soporte hardware

La eficiencia en el uso de IPSec puede verse incrementada si se utilizan funcionalidades criptográficas proporcionadas por el hardware del sensor. Por ejemplo, el chip de radio CC2420 cuenta con dicha funcionalidad en algunos de sus sensores.

En los gráficos 3 y 4 de los apartados anteriores se puede observar el impacto del uso de soporte hardware con este fin; en todos los casos las implementaciones basadas en hardware son más rápidas que las que se basan únicamente en software. Si nos ceñimos a los datos del estudio, en datagramas de 512 bytes y salto único, el incremento del coste computacional es del 65%, que se reduce al 12% con la ayuda de un coprocesador criptográfico. En el caso de ESP, se reduce del 64% al 37%.

3.2 Comparación entre seguridad en la capa de enlace e IPSec para 6LoWPAN

Como se ha mencionado ya anteriormente en este trabajo, el internet de las cosas (IoT) está compuesto por una gran cantidad de dispostivos interconectados entre sí y conectados a internet; la seguridad en internet es un aspecto sumamente importante, y en el Internet de las Cosas no es una excepción, aunque este tipo de dispositivos tenga cierto tipo de limitaciones.

Estos objetos inteligentes de los que se compone el IoT normalmente se encuentran interconectados mediante IEEE 802.15.4. Un router final es el encargado de conectar la red 802.15.4 a internet con el fin de habilitar la comunicación IPv6 entre los objetos inteligentes que conforman la red e internet. Dado que los paquetes 6LoWPAN se encuentran comprimidos para que estos nodos de la red con bajos recursos sean capaces

de comunicarse, es el nodo final el encargado de comprimir/descomprimir los paquetes que recibe.

Actualmente, 6LoWPAN utiliza los mecanismos de seguridad que 802.15.4 proporciona. Se utiliza una única clave en la red para la seguridad en la transferencia de datos y ésta se proporciona entre saltos y no extremo a extremo. Este esquema de seguridad puede considerarse adecuado en el caso en que la red no se encuentra conectada a internet y el robo de la clave no es posible. Sin embargo, en el caso en el que nos encontramos, se debe tener la capacidad de asegurar los datos E2E, dotándolos de autenticación, integridad, no repudio y confidencialidad.

Dicho método para establecer una seguridad E2E es IPSec, tratado en la Sección 3.1. IPSec define extensiones de seguridad al protocolo IP con el fin de implementar servicios de seguridad.

La extensión de seguridad 6LoWPAN/IPSec estudiada en este trabajo para dotar a la red de seguridad extremo a extremo entre los objetos inteligentes de la red y los host de internet se encuentra ilustrada en la Figura 18. Se definen una compresión de cabecera para las extesiones que IPSec propone: Authentication Header (AH) y Encapsulating Security Payload (ESP).

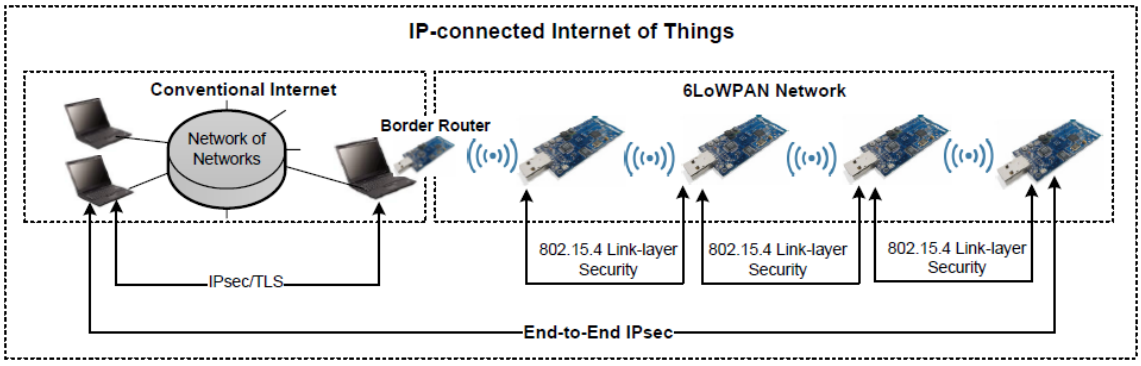


Figura 18: Seguridad 802.15.4 e IPSec.

La seguridad proporcionada por 802.15.4 puede asegurar la comunicación entre dispositivos 6LoWPAN utilizando una clave compartida. IPSec/TLS asegura la comunicación entre el router final (6BR) y un host de internet.

En esta sección se llevará a cabo una comparación entre la seguridad ofrecida por 802.15.4 y 6LoWPAN/IPSec. Para este fin se analizará una implementación de la seguridad 802.15.4 a nivel de enlace para el sistema operativo Contiki.

Anteriormente en este trabajo ya se ha comentado el funcionamiento de IPSec para redes 6LoWPAN, así como la seguridad que los mecanismos de 802.15.4 ofrecen a la red.

3.2.1 Implementación

Implementación de la seguridad a nivel de enlace:

La implementación del mecanismo de seguridad de IEEE 802.15.4 estudiada, llevada a cabo por Raza et al.[24] soporta la construcción de la cabecera para todos los modos de seguridad descritos en el estándar. La construcción del frame IEEE 802.15.4 es llevado a cabo por software mientras que las operaciones criptográficas se ejecutan mediante hardware, en el chip de radio CC2420. Como clave se utiliza una clave predefinida.

Como ya se ha comentado con anterioridad, hay que destacar que en la seguridad a nivel de enlace, en caso en que un fragmento que cuente con este tipo de seguridad sea necesario la fragmentación, el aumento en la cabecera se llevará a cabo en cada uno de los fragmentos que forma el paquete original.

Implementación de IPSec:

Al igual que en la Sección 3.1, la implementación de AH y ESP de IPSec se realizó sobre el sistema operativo Contiki. Para ello, llevaron a cabo una modificación de la pila IP de Contiki que ofrece la funcionalidad 6LoWPAN, introduciendo las codificaciones NHC EH, NHC AH y NHC ESP en la capa SICLoWPAN, el componente 6LoWPAN de la pila IP.

Se utiliza las implementaciones de SHA1 y AES de MIRACL [18], una librería de código abierto.

Con el fin de realizar un estudio de la diferencia entre realizar las operaciones criptográficas vía software o hardware, el equipo que lleva a cabo la implementación propone el uso de las capacidades de criptografía del chip de radio CC2420 con este fin.

La implementación de Contiki 6LoWPAN/IPSec utiliza claves pre-compartidas para establecer las asociaciones de seguridad.

3.2.2 Uso actual

La seguridad en la capa de red y en la capa de enlace no son excluyentes. IPSec garantiza la seguridad extremo a extremo, mientras que en la capa de enlace es entre saltos. Por este motivo, mediante el uso de seguridad en la capa de enlace detecta antes los ataques contra la modificación de los datos, por lo que es importante su uso. En este caso, el uso de CCM-128 en la capa de enlace puede ser innecesaria y CBC-128 sería suficiente.

3.2.3 Evaluación y resultados

De forma análoga al apartado 3.1.6, se realiza el estudio de una implementación de IPSec utilizando IPSec comprimido ya definido con anterioridad.

La configuración que se muestra en la Figura 18 se tratan de cuatro nodos Tmote Sky, un nodo Tmote Sky actuando como un router final 6LoWPAN y un host de internet corriendo un sistema operativo Ubuntu. El host de internet corre la pila IPSec del kernel de Linux. Los cuatro nodos de la parte derecha de la figura forman una red multihop. Utilizando esta disposición se realizan los diferentes test con configuraciones diferentes. En estos test, los nodos ejecutan una única aplicación que escucha un puerto UDP. Los mensajes se crean por un programa cliente en el host Linux y son reenviados por la red hasta el nodo destino, que cambia en función del experimento. Todos los nodos procesan los paquetes utilizando 6LoWPAN y µIP. Si se utiliza IPSec, tanto el host Linux como el nodo destino deben tener capacidades criptográficas (seguridad extremo a extremo). Sin embargo, si se utiliza la seguridad en la capa de enlace (seguridad 802.15.4), todos los nodos incluidos en el camino hacia el nodo destino deben tener dicha capacidad, dado que la seguridad en este caso es a nivel de salto. La carga útil se envía a la aplicación en el nodo destino, que genera un nuevo datagrama del mismo tamaño como respuesta. Dicha respuesta se envía al host Linux, siguiendo el camino inverso. Para algunos test se utiliza la seguridad IPSec mientras que en otros casos se utiliza la seguridad 802.15.4.

3.2.3.1 Comparación en el uso de memoria

En la Tabla 5 se muestra el uso de memoria, tanto ROM como RAM de cada una de las configuraciones de seguridad que se muestran en la primera columna de la misma.

Configuración	ROM (kB)		RAM (kB)	
	Total	Diferencia	Total	Diferencia
Sin seguridad	26.1	-	8.0	-
Seguridad 802.15.4	27.3	1.2	8.0	-
AH con MAC-SHA1-96	30.7	4.6	9.1	1.1
AH con XCBC-MAC-96	32.3	6.2	8.5	0.5
AH con XCBC-MAC-96 (con hardware)	27.6	1.5	8.3	0.3
ESP con AES-CBC	34.8	8.7	8.3	0.3
ESP con AES-CBC (con hardware)	30.0	3.9	8.3	0.3
ESP con AES-CTR	33.2	7.1	9.1	0.3
ESP con AES-CTR (con hardware)	28.4	2.9	9.1	0.3
ESP con AES-XCBC-MAC-96	32.7	6.6	8.3	0.3
ESP con AES-XCBC-MAC-96 (con hardware)	28.0	1.9	8.3	0.3
ESP con AES-CBC + AES-XCBC-MAC-96	35.3	9.2	8.3	0.3
ESP con AES-CBC + AES-XCBC-MAC-96 (con hardware)	30.5	4.4	8.3	0.3
ESP con AES-CTR + AES-XCBC-MAC-96	33.7	7.6	8.3	0.3
ESP con AES-CTR + AES-XCBC-MAC-96 (con hardware)	28.9	2.8	8.3	0.3

Tabla 4: Impacto en memoria del uso de IPSec y seguridad 802.15.4

La tabla muestra que AH y ESP consumen 3.9kB y 9kB para los algoritmos IPsec obligatorios. Puede observarse como el impacto en memoria en el caso de IPSec es considerablemente mayor que en el caso del uso de la seguridad 802.15.4. Sin embargo, este impacto se ve reducido en el caso del uso de apoyo hardware para el cifrado y descifrado. Asimismo, puede compararse cada una de las configuraciones con la ausencia de seguridad que, obviamente, es la configuración que menos memoria consume.

Merece la pena mencionar que a diferencia de AES-CBC, el modo AES-CTR de operación sólo se basa en el cifrado AES. Por lo tanto, la configuración de AES-CTR + AES-XCBC-MAC-96 se puede implementar sin el descifrado AES, lo que se traduce en una huella en memoria especialmente baja. La sobrecarga adicional en la memoria ROM es de 2,8 Kb, comparable con los 1,2 Kb que usa la seguridad a nivel de enlace. La huella de la memoria RAM se calcula como la suma de los segmentos de datos globales y el uso de la pila de tiempo de ejecución que se mide mediante la ejecución

de Contiki en el emulador MSPSim [20]. Con una huella adicional de 1,1 kB, la configuración AH HMAC-SHA1 es la configuración que más RAM consume. Cuando se utilizan otros modos de operación, el uso de la RAM se encuentra entre 0.3 y 0.5 kB.

Con el estudio de los resultados anteriores podemos llegar a la conclusión de que pese a la utilización de mecanismos de seguridad como IPSec o la seguridad a nivel de enlace, el dispositivo sigue contando con memoria suficiente para los programas de la capa de aplicación (la memoria ROM del nodo utilizado en la simulación, Tmote Sky, es de 48 kB). Por lo tanto, puede aplicarse este tipo de seguridad en dispositivos de capacidades restringidas.

3.2.3.2 Comparación en el tamaño de la cabecera

El uso de un mecanismo de seguridad se traduce en un incremento del tamaño de la cabecera.

La tabla 5 muestra el incremento en la cabecera utilizando IPSec y utilizando seguridad en la capa de enlace.

Configuración	IPsec comprimido		802.15.4	
	Modo	Aumento de cabecera	Modo	Aumento de cabecera
Integridad	HMACSHA1 -96	16 Bytes	AES-CBC-MAC- 96	12 Bytes por flag
Confidencialidad	AES-CBC	14 Bytes	AES-CTR	5 Bytes por flag
Integridad y autenticidad	AES-CBC y HMAC-SHA1-96	24 Bytes	AES-CCM-128	21 Bytes por flag

Tabla 5: Incremento en el tamaño de cabecera con IPSec y seguridad 802.15.4

El incremento de la cabecera utilizando IPSec comprimido es mayor que en el caso de 802.15.4. Sin embargo, en el momento en el que el mensaje necesite ser fragmentado una única vez, el incremento en la cabecera del mensaje original sería mayor.

Solo integridad:

Al utilizar la seguridad a nivel de enlace, el incremento en el tamaño del datagrama destinado a la autenticación es el tamaño que ocupa la MAC. En el caso del uso de IPSec utilizando AES-XCBCMAC-96, el tamaño de la MAC es de 12 bytes. El campo adicional AH incrementa la cabecera en otros 12 bytes, lo que hace un total de 24. Sin embargo, utilizando el la compresión de cabecera IPSec definida, dicho tamaño adicional se reduce a 16 bytes. Por lo tanto, la seguridad E2E que IPSec comprimido proporciona tiene un incremento en el tamaño de la cabecera de 4 bytes.

Solo confidencialidad:

Si se requiere únicamente la encriptación del mensaje, 802.15.4 proporciona AES-CTR con un incremento en el tamaño de la cabecera de 5 bytes. IPSec con ESP y AES-CBC la incrementa en 18 bytes. La compresión de dicho protocolo la reduce a 12, lo que se traduce en un incremento de 7 bytes con el fin de la proporción de la seguridad E2E que IPSec proporciona, incremento de tamaño que se amortiza en el caso en que un datagrama necesite ser fragmentado en 3 o más ocasiones.

Integridad y confidencialidad:

Con el uso de AES-CCM-128, el incremento de la cabecera 802.15.4 es de 21 bytes, mientras que con IPSec ESP sería de 30 bytes, que se ven reducidos a 24 bytes. La diferencia entre ambos es de 3 bytes, a lo que se debe añadir la seguridad extremo a extremo de IPSec.

3.2.3.3 Comparación en el consumo de energía

La implementación de mecanismos de seguridad en el Internet de las Cosas supone un coste energético que, debido a la naturaleza de los dispositivos que forman este tipo de redes, debemos tener en cuenta.

Las mediciones del consumo de energía se realizan para diferentes configuraciones de seguridad. Para llevarlas a cabo, el equipo que lleva a cabo la implementación utiliza Powertrace, una herramienta integrada en Contiki. Para todas las mediciones se considera un voltaje constante de 3V, de manera que se proporciona una comparación equitativa de todos los experimentos. No se puede hablar en términos de la vida útil debido a que la descarga de la batería del dispositivo varía en función de diferentes factores, como el clima.

En la recogida de datos, se tienen en cuenta los ticks de CPU desde la recepción del primer fragmento de un mensaje, cuando comienza el desencriptado de la capa de enlace, parando el recuento cuando el procesamiento de la capa de enlace del último fragmento se ha completado.

Se ignora el tiempo de procesamiento por parte de la red; el tiempo que un nodo consume esperando un nuevo fragmento y el que emplea para la transmisión de un fragmento. Se incluye las mediciones del procesamiento de la capa de enlace, procesamiento 6LoWPAN, manejo de la pila µIP y procesamiento de la capa de aplicación.

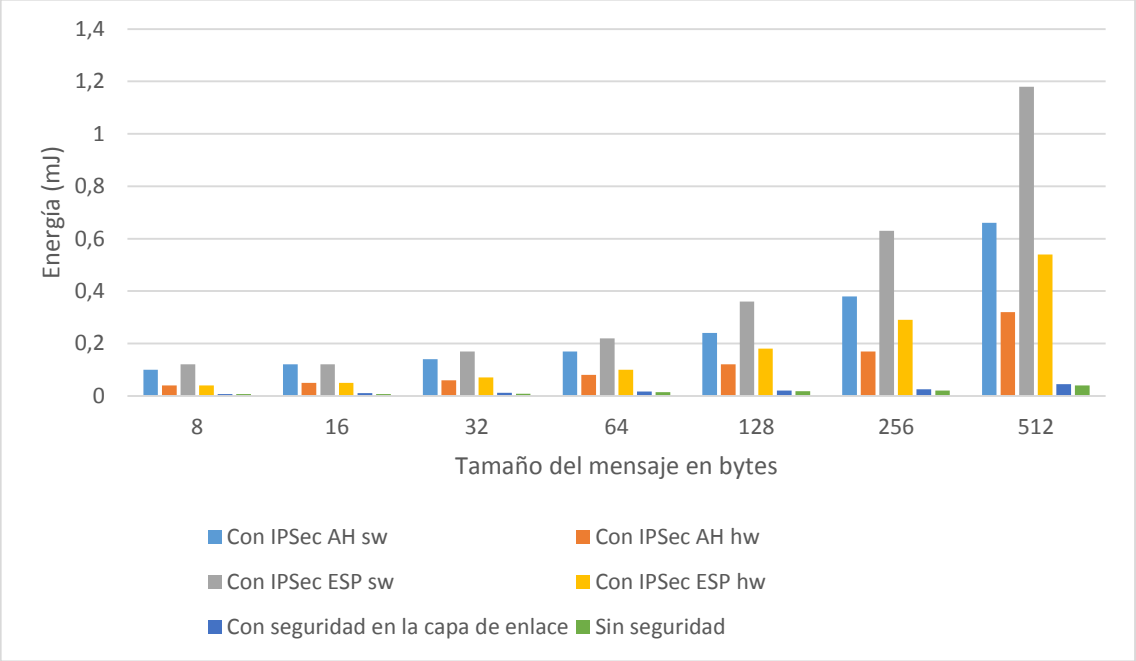


Gráfico 5: Relación tamaño de mensaje y energía consumida con IPSec y seguridad IEEE 802.15.4.

El tamaño del mensaje con la energía consumida por el nodo con diferentes configuraciones de seguridad. El consumo de energía es mayor utilizando IPSec, siendo mayor con ESP que con AH. Vemos como el consumo de energía utilizando IPSec es considerablemente mayor que en el caso en que se utilice seguridad a nivel de enlace. Sin embargo, este incremento en el consumo de energía puede ser asequible en el caso en el que se necesite seguridad E2E. Además, dicho consumo de energía es pequeño comparado con otros procesos llevados a cabo en los nodos de la red, como el tiempo de escucha. Por ejemplo, un nodo escuchando 2 segundos consumo mucha más energía que en el peor de los casos del ejemplo, utilizando paquetes de 512 bytes con IPSec AH sin soporte hardware.

Se puede apreciar en la figura como, mediante el uso de soporte hardware para las operaciones criptográficas, el consumo de energía se ve reducido en torno al 50%.

También debe considerarse que en el caso de utilizar IPSec, las operaciones criptográficas las realizan únicamente los nodos finales, mientras que en el uso de la seguridad a nivel de enlace éstas se realizan en cada uno de los nodos que se encuentran en el path (ruta) de destino hacia el nodo final.

3.2.3.4 Resultados globales de la red

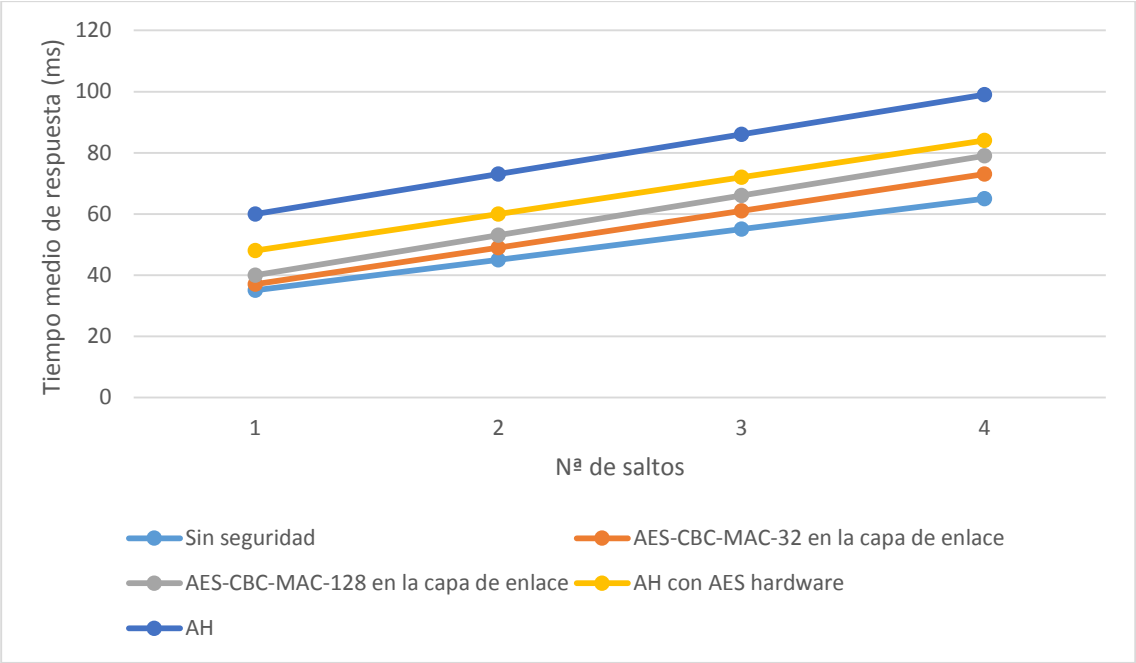
En esta sección se evalúa la respuesta del sistema ante diferentes configuraciones y tamaño de paquetes, así como con diferentes número de saltos. En este caso, el tiempo de respuesta se trata del tiempo que tarda un mensaje en ser respondido al emisor del mismo desde que es enviado, es decir, un servicio de echo.

Se evalúan diferentes modos de operación, en función de las necesidades de seguridad de la red.

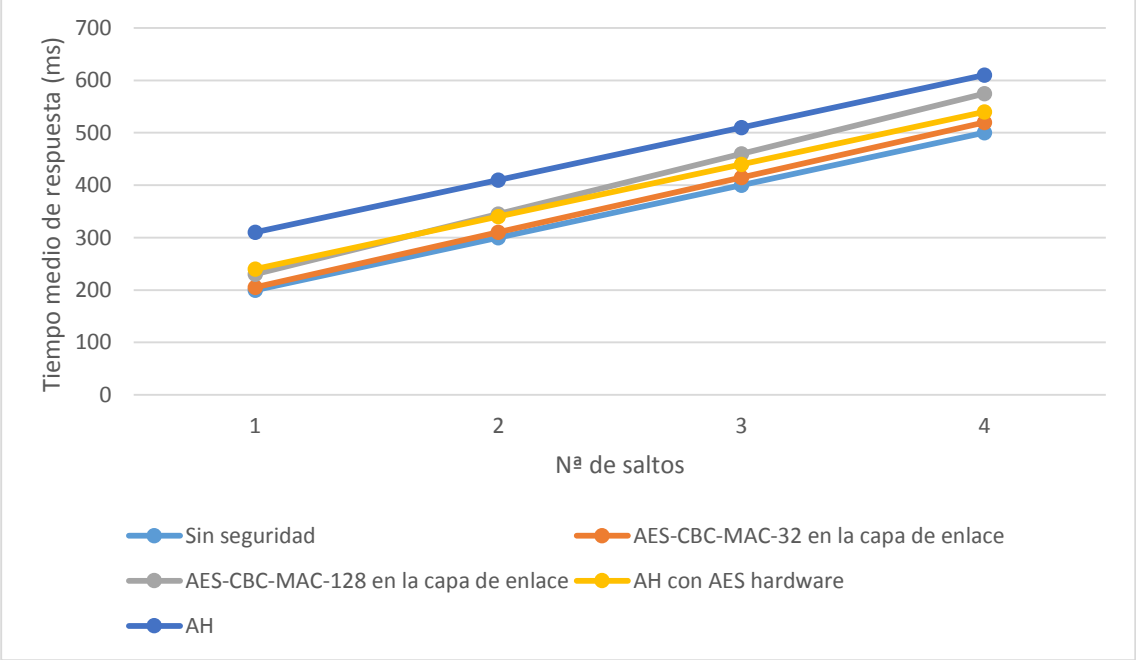
Impacto de la implantación de integridad:

En el primer caso nos centramos en la integridad. IEEE 802.15.4; proporciona servicios de seguridad con AES-CBC-MAC con un tamaño de MIC de 4, 8 y 16 bytes. IPSec proporciona integridad con AH y opcionalmente con ESP con un tamaño de MIC de 12 bytes.

El Gráfico 6 muestra el tiempo de respuesta con diferentes configuraciones que dotan a los mensajes de integridad. AES-CBC-MAC en la capa de enlace con diferentes tamaños de MIC, IPsec AH con XCBC-MAC-SHA1-96 y sin seguridad. Con un tamaño de datos de 512 bytes y 4 saltos, AH software aumenta el tiempo de respuesta en un 26%, tiempo que se ve reducido al 11% con la ayuda de criptografía AES vía hardware. El incremento del tiempo de respuesta se mantiene constante en función del número de saltos en el caso de IPSec, ya que las operaciones criptográficas se llevan a cabo en el emisor y en el destinatario del paquete. Con el uso de la seguridad en el nivel de enlace, esto no es así, pudiéndose observar como AH con soporte hardware, dados 2 saltos o más, resulta ser más rápido que AES-CBC-MAC-128.



a) Salto múltiple con 16 bytes de datos



b) Salto múltiple con 512 bytes de datos

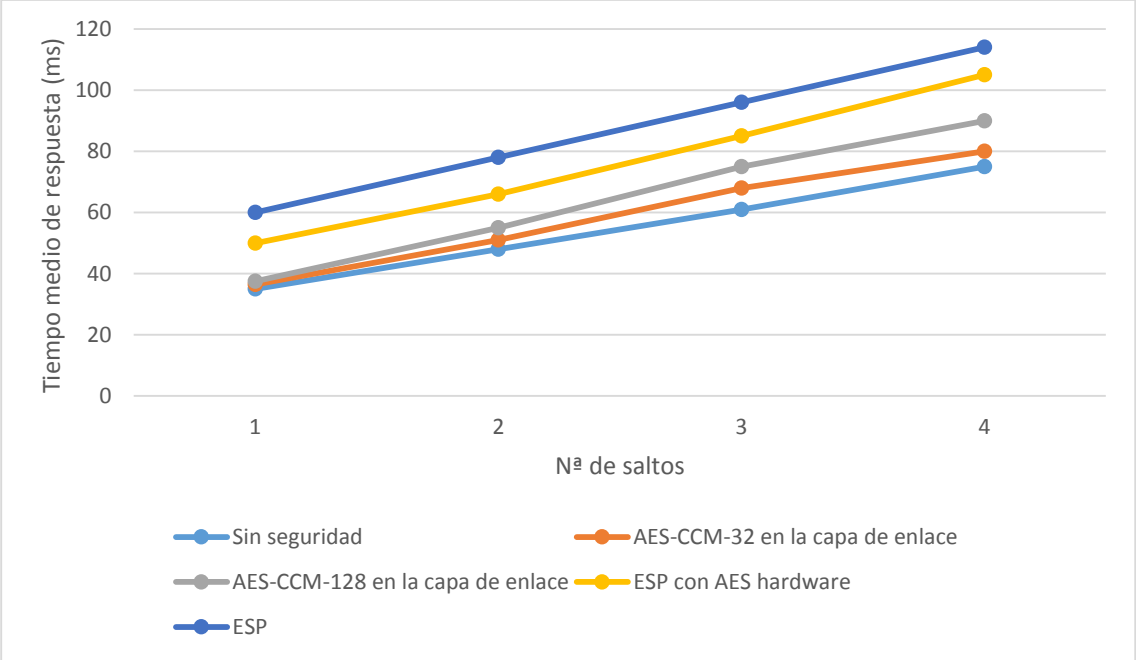
Gráfico 6: Relación tiempo de respuesta y número de saltos con IPSec y seguridad 802.15.4

Se puede apreciar como para tamaños de datos pequeños, conviene el uso de seguridad en la capa de enlace mientras que para tamaños de datos grandes IPSec ofrece mejores resultados. También podemos llegar a la conclusión de que dependiendo del número de saltos que sean necesarios para la comunicación de nuestra red, claramente esto

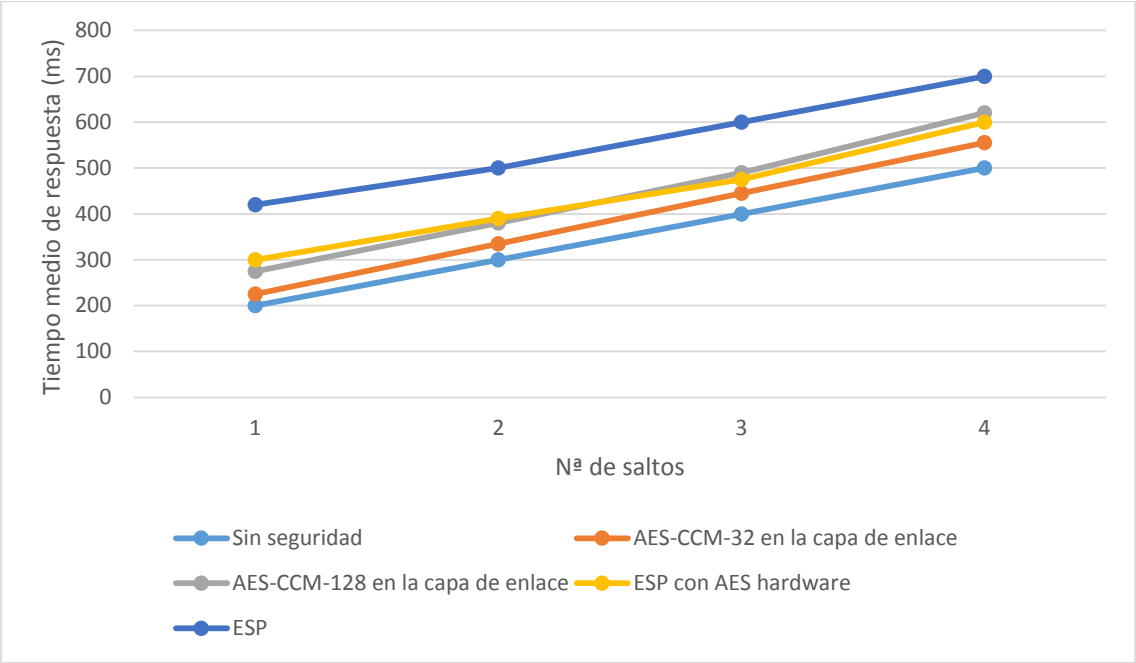
dependerá del número de nodos que conformen la misma, es recomendable utilizar una configuración u otra. En el caso de la seguridad IEEE 802.15.4, con salto único es más veloz que IPSec y esta ventaja se va desvaneciendo en función del número de saltos.

Impacto de la implantación de integridad y confidencialidad:

En este caso nos centramos en el contexto en el que se necesita tanto integridad de los datos como confidencialidad de los mismos. Estos servicios los proporciona tanto 802.15.4 con AES-CCM como ESP combinando integridad y confidencialidad. En este caso se emplea AES-CTR con AES-XCBC-MAC.



a) Salto múltiple con 16 bytes de datos



b) Salto múltiple con 512 bytes de datos

Gráfico 7: Relación tiempo de respuesta y número de saltos

Al igual que en el caso en el que se provee únicamente de integridad, la seguridad IEEE 802.15.4 se comporta mejor en salto único mientras que IPSec funciona mejor en salto múltiple; dicha diferencia se acentúa conforme van aumentando el número de saltos.

La evaluación muestra que AES-CTR y AES-XBC-MAC-96 son las implementaciones más rápidas y más eficientes energéticamente. IPSec es más eficiente para los nodos no finales, mientras que la seguridad en la capa de enlace es al contrario.

En cuanto al tiempo de respuesta, como se podía intuir, IPSec proporciona una mayor escalabilidad que la seguridad IEEE 802.15.4. La seguridad 802.15.4 se consagra como una mejor solución para paquetes de pequeño tamaño, así como para salto único, pero dado que nos encontramos ante redes de nodos esto ocurrirá en el menor número de casos. IPSec se comporta mejor con un tamaño de datos mayor y número de saltos, dado que, como se ha comentado en varias ocasiones, los nodos que se encargan únicamente del envío a otro nodo de la información recibida, es decir, forman parte del path hacia el nodo final pero no lo son, no realizan operaciones criptográficas, por lo que el tiempo de respuesta aumenta de forma constante con respecto al número de saltos. Además, como se ha podido observar, la eficiencia de IPSec puede verse aumentada de forma sustancial con el uso de soporte hardware con capacidades criptográficas, como el coprocesador incluido en los chips de radio actuales como CC2420.

3.3 Solución de seguridad ligera para CoAP en el IoT

6LoWPAN trabaja sobre UDP, utilizando CoAP en la capa de aplicación. Para dotar de seguridad a dicho protocolo, CoAP propone el uso de DTLS como protocolo de seguridad con el fin de dotar al sistema de una distribución automática de claves, autenticación, integridad y confidencialidad. El término empleado para dicha asociación es CoAPs. Sin embargo, dicho protocolo de seguridad está pensado para escenarios en los que el tamaño de los mensajes no supone ningún inconveniente, situación que no se da en el caso de las redes que tratamos en este trabajo. Es por eso que es necesario el uso de un mecanismo capaz de comprimir dichos mensajes con el fin de que sean lo más escuetos posibles. 6LoWPAN ya cuenta con un mecanismo para comprimir el protocolo IP con el fin de utilizarlo en este tipo de dispositivos; dicho mecanismo puede utilizarse para comprimir los mensajes necesarios para el uso de DTLS en CoAP.

En esta sección se estudia el uso de dichos mecanismos para la compresión de paquetes DTLS, conformando un CoAPs ligero (CoAPs Lithe). El resultado del uso de esta tecnología tiene una función doble:

- Aumentar la eficiencia energética reduciendo el tamaño de los paquetes.
- Evitar la fragmentación de paquetes requerida cuando éstos adquieren cierto tamaño, superan la MTU (Unidad Máxima de Transferencia), que en nuestro caso es de 127 bytes. Esto es importante dado las vulnerabilidades que 6LoWPAN presenta frente a ataques a la fragmentación [21].

DTLS comprimido ofrece seguridad E2E entre los host que lo tengan activado y el host de internet que utilice DTLS estándar, es decir, no comprimido. En la figura xx se muestra una configuración típica de uso, donde los nodos 6LoWPAN utilizando CoAPs se encuentran conectados a un 6BR que hace de puente entre ellos e internet.

3.3.1 CoAP y DTLS

CoAP es un protocolo de servicio que utiliza UDP como capa de transporte; proporciona una interfaz REST similar a HTTP y es adecuada para su uso en dispositivos limitados y comunicación máquina a máquina. Para proveer seguridad dicho protocolo, se propone el uso de DTLS adaptado para este tipo de escenarios, el DTLS comprimido.

DTLS consta de dos subcapas, la alta y la baja. La subcapa alta contiene:

- Handshake Protocol: permite que los puntos de comunicación se autentiquen mutuamente, y que además negocien un cipher suite y (opcionalmente) un método de compresión
- Change Cipher Spec Protocol: permite a los puntos de comunicación activar el cipher suite
- Alert Protocol: permite a los puntos de comunicación indicar posibles problemas potenciales e intercambiar los correspondientes mensajes de alerta
- Application Data Protocol: es el propio protocolo de la capa de aplicación (ej: HTTP), y alimenta al SSL Record Protocol

La subcapa baja contiene:

- SSL Record Protocol: fragmenta los datos de la capa de aplicación y los procesa de forma individual

La Figura 19 muestra la estructura de un mensaje DTLS en un datagrama IP/UDP.

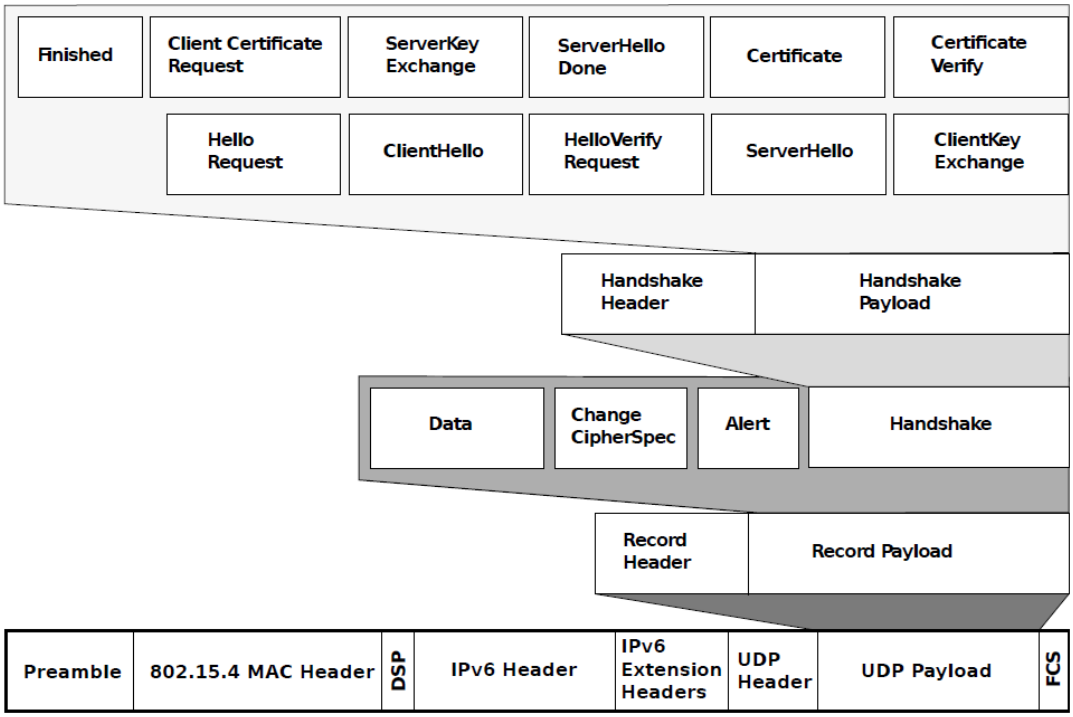


Figura 19: Estructura de un paquete asegurado utilizando DTLS

El Handshake Protocol se utiliza antes de transmitir ningún dato de la capa de aplicación. Permite al servidor y al cliente autenticarse mutuamente, negociar un algoritmo de cifrado y una función MAC y negociar las claves que se utilizarán para

proteger los datos del Record Protocol. La Figura 20 muestra un proceso de hadshake completo.

El Alert Protocol se usa para informar al otro punto de comunicación las alertas relacionadas con la comunicación.

El Record Protocol [9] toma los datos de la subcapa alta, los fragmenta en bloques manejables, los comprime de forma opcional, añade el MAC, cifra, y añade una cabecera. En recepción, los datos recibidos son descifrados, verificados, descomprimidos y reensamblados antes de entregarlos a la capa de aplicación

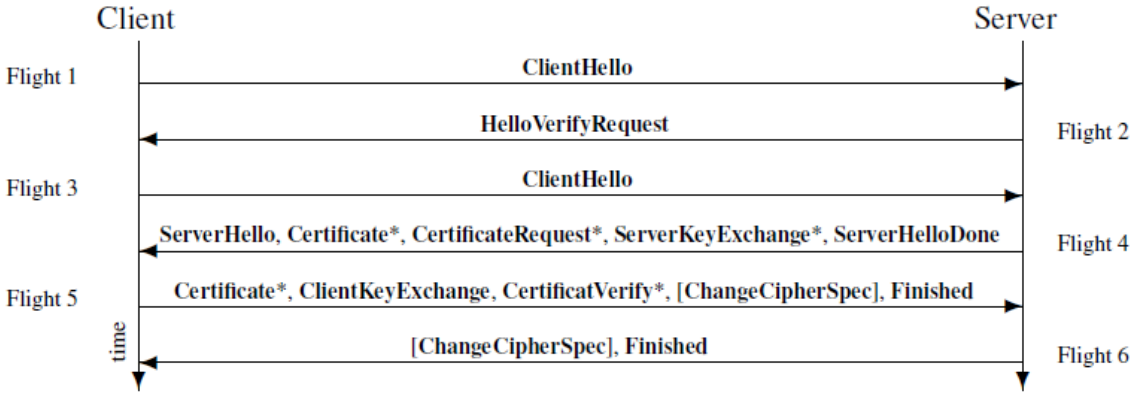


Figura 20: Proceso de handshake DTLS. Los mensajes marcados con * son opcionales

3.3.2 Compresión DTLS

El protocolo 6LoWPAN define la fragmentación de los datagramas IPv6 en redes 6LoWPAN, así como la compresión de la cabecera IP mediante los mecanismos de compresión IP Header Compression (IPHC) y Next Header Compression (NHC). La codificación IPHC comprime la cabera IPv6 dejándola en 2 bytes en el caso de un solo salto y 7 bytes en el caso de salto múltiple. Existe un bit en IPHC, el NH, que indica si el encabezado siguiente se encuentra comprimido utilizando NHC. NHC se utiliza para codificar las cabeceras de extensión de IPv6 y la cabecera UDP. El tamaño de la compresión NHC es múltiplo de octetos (normalmente 1 byte) que contiene una ID de tamaño variable y los bits codificados de una cabecera concreta. El estándar NHC definido puede comprimir cabeceras por debajo de UDP y no capas superiores, dado que las codificaciones NHC para UDP no cuentan con el bit NH que indica si el encabezado siguiente se encuentra comprimido. Para la implantación de DTLS a las

redes 6LoWPAN, puede utilizarse estos protocolos de compresión con el fin de hacer menos pesado DTLS para dispositivos 6LoWPAN.

La compresión de la cabecera se utiliza dentro de la red 6LoWPAN; el encargado de la compresión de datos entrantes a la red, descompresión de datos salientes de la misma, así como de la fragmentación/desfragmentación de datagramas entrantes y salientes es el router 6BR, que hace las veces de puente entre la red de nodos e internet entre host convencionales.

La fragmentación de paquetes ocurre cuando el tamaño del mismo excede la unidad de transmisión máxima (MTU). En el caso de 6LoWPAN, que utiliza el protocolo IEEE 802.15.4, el MTU es de 127 bytes.

3.3.3 Integración DTLS-6LoWPAN

Debido a que el estándar actual de NHC no permite utilizarlo para la compresión de la carga útil UDP ni sus capas superiores, es necesaria una modificación del mismo con el fin de utilizar dicho protocolo para la compresión DTLS. Por lo tanto, utilizaremos otro NHC que nos permita llevar a cabo estas operaciones.

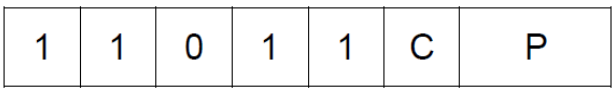
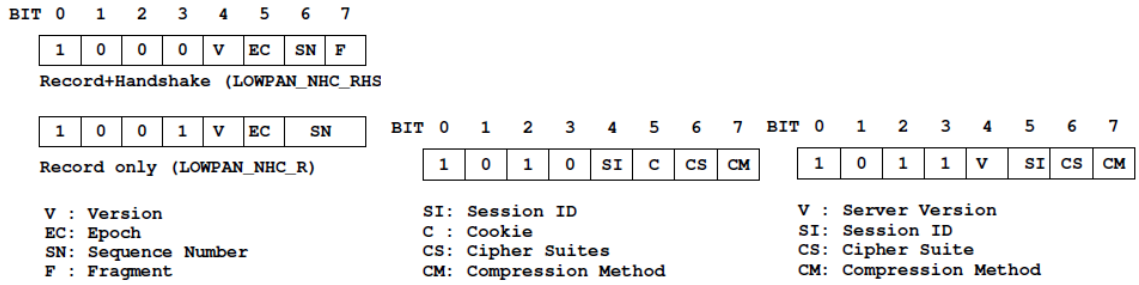


Figura 21: 6LoWPAN-NHC para UDP, con capacidad para compresión de la carga útil de UDP

Los bits ID en NHC son 11110 para UDP, como se define en el estándar NHC, que indican que la carga útil UDP no se encuentra comprimida. Utilizando el ID 11011 (Figura 21), que no se utiliza en 6LoWPAN, indicamos que la carga útil se encuentra comprimida utilizando NHC.

3.3.3.1 6LoWPAN-NHC para las cabeceras Record y Handshake

De forma análoga al apartado anterior, el protocolo NHC puede adaptarse para su uso con diferentes cabeceras.



(a) Record and the Handshake header. (b) ClientHello message. (c) ServerHello message.

Figura 22: Codificación NHC para diferentes cabeceras DTLS

El protocolo Record añade a la cabecera de cada paquete 13 bytes, mientras que el protocolo HandShake añade 12 bytes de cabecera a cada uno de los mensajes handshake. Utilizando NHC, esta longitud se reduce a 5 y 3 bytes respectivamente. Sin embargo, en el caso del handshake, solo se comprimirán los mensajes durante el primer handshake, debido a que tras este se utilizara el mecanismo de cifrado y la clave que ambas parten acuerden durante el proceso. En cualquier caso, la cabecera Record es siempre comprimida utilizando este método.

Para comprimir la cabecera Record y Handshake se tienen en cuenta dos situaciones: que la cabecera Record contenga un fragmento de un mensaje handshake, en cuyo caso ambos se codificarán utilizando un único byte (Record + HandShake, 6LoWPAN-NHC-RHS). En el otro caso, el fragmento de la cabecera Record se trata de datos de la aplicación (6LoWPAN NHC-R).

Los bits codificados tienen las siguientes funciones:

Los primeros 4 bits representan el campo ID que se utiliza para distinguir las diferentes configuraciones que se emplean y para cumplir con el esquema 6LoWPAN-NHC. Para 6LoWPAN-NHC-RHS se emplea el ID 1000 y para LoWPAN-NHC-R el 1001.

- Version (V): Si es 0, la versión es la última DTLS y el campo se omite. Si es 1, la versión se especifica explícitamente.

- Epoch (EC): Si es 0, se utilizan 8 bits para definir el instante de envío y los 8 restantes se omiten. Si es 1, se especifican los 16 bits.

- Sequence number (SN): El número de secuencia consta de 48 bits, aunque algunos de ellos pueden ser ceros. Si SN es 0, se usan 16 bits y se omiten los 32 restantes. Si es 1, se especifican los 48 bits. En el caso de 6LoWPAN-NHC-R,

como se aprecia en la Figura 22 (a), se utilizan 2 bits para definirlo. Si SN es 00, se usan 16 bits para definirlo. Si es 01, se usan 32, si es 10 se usan 24 y si es 11 se usan los 48 bits para definir el número de secuencia.

- Fragment (F): Si es 0, el mensaje handshake no está fragmentado y los campos fragment offset y fragment length se omiten. Si es 1, ambos campos se especifican.

En el caso de 6LoWPAN-NHC-R, el campo contet type siempre se especifica, y en el caso de 6LoWPAN-NHC-RHS, content type siempre se omite ya que se obvia que se trata de un mensaje handshake lo que se encuentra fragmentado. Además, los campos message type y message seq de la cabecera Handshake siempre se especifican. El campo lenght en las cabeceras Record y Handshake siempre se omite ya que puede deducirse de las capas inferiores, tanto de la cabecera 6LoWPAN como de la IEEE 802.15.4.

3.3.3.2 6LoWPAN-NHC para ClientHello

En la Figura 22 (b) se muestra la estructura de la codificación NHC para el mensaje ClientHello (6LoWPANNHC-CH). Durante el handshake se envía dos veces, una sin la cookie del servidor y la otra con la cookie. Los campos que lo conforman son:
Los primeros 4 bits representan la ID 6LoWPAN-NHC-CH, 1010.

- Session ID (SI): Si es 0, la session id es no está disponible y los 8 bits que el tamaño de la misma se omiten. El campo session id en el ClientHello va de 0 a 255 bits, siempre precedidos por 8 bits que indican el tamaño de la misma. El mensaje ClientHello utiliza la ssesion id solo en el caso de que el cliente DTLS pretenda reanudar una sesión antigua. Si el campo Session ID es 1, esta se especifica.

- Cookie (C): Si es 0, no está disponible y los 8 bits que informan de la longitud se omiten. La cookie contiene de 0 a 255 bits, siempre precedida del campo que informa de su longitud. Si es 1, se especifica.

- Cipher Suite (CS): Si es 0 se utiliza el cipher suite por defecto (TLS ECDHE ECDSA WITH AES 128 CCM 8) de CoAP que soporta un manejo de claves automático y los 16 bits de prefijo de longitud se omiten. El tamaño de cipher suite va desde los 16 hasta los 2^{16} bits. Si es 1, estos bits se especifican.

- Compression Method (CM): Si es 0, se utiliza el método de compresión por defecto y los 8 bits de prefijo de longitud se omiten. Si es 1, éstos 8 bits se utilizan y el método de compresión se especifica.

El campo random siempre se especifica. En caso de que el cliente especifique una versión que el servidor no soporta, el ServerHello contiene un campo que muestra la versión que este soporta.

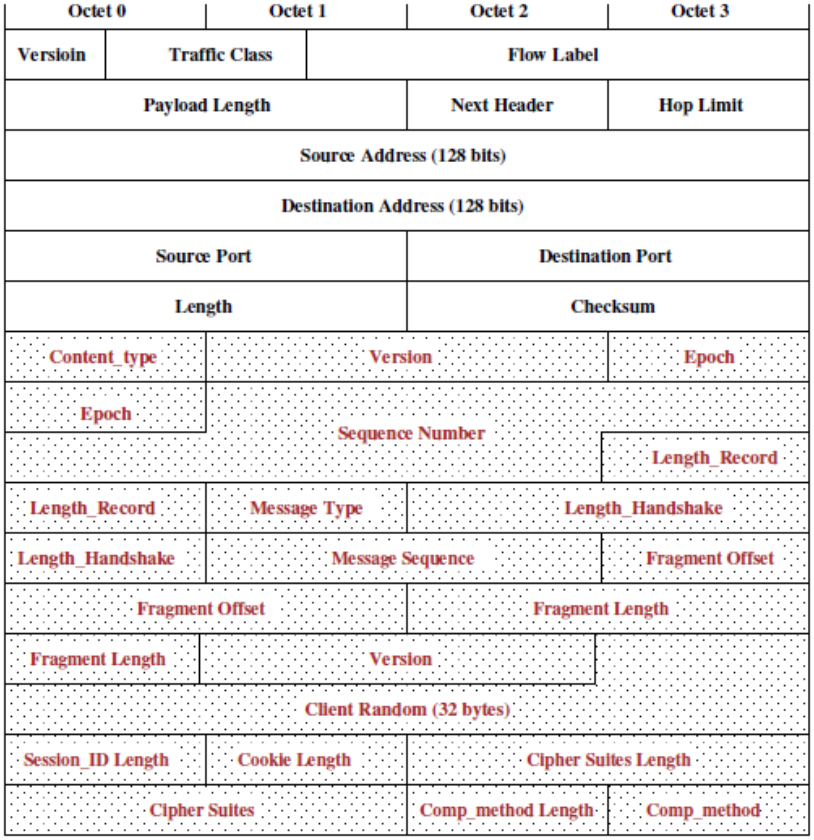


Figura 23: Mensaje ClientHello sin comprimir

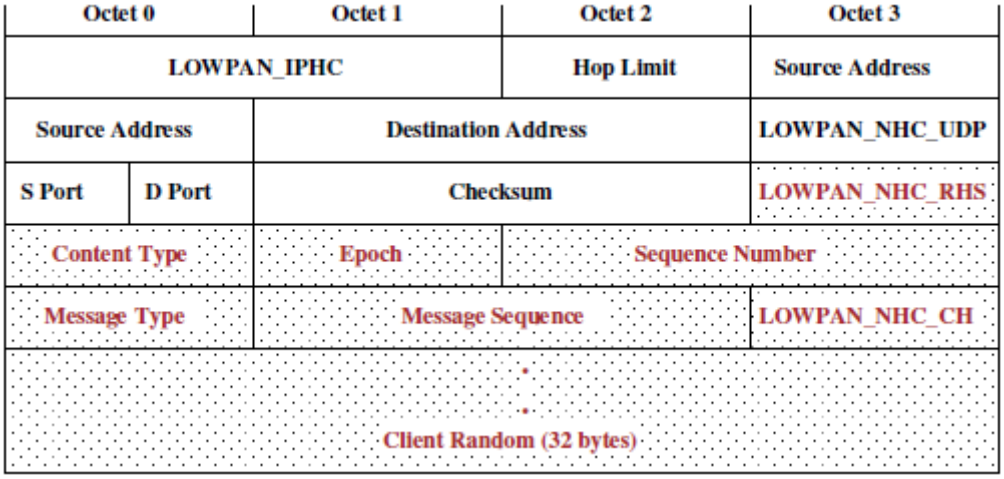


Figura 24: Mensaje ClientHello comprimido

3.3.3.3 6LoWPAN-NHC para ServerHello

En la Figura 22 (c) se puede observar la estructura de la cabera del mensaje de ServerHello. Es muy similar a la de ClientHello con la particularidad del tamaño del cipher suites y compression methods, de 16 y 8 bits respectivamente.

Los primeros 4 bits representan el ID de 6LoWPAN-NHC-SH, 1011.

Version (V): Si es 0, se asume que la versión soportada por el servidor es la DTLS 1.0. Si es 1, la versión a utilizar para la comunicación entre el cliente y el servidor se espeficia.

Los campos Session ID (SI), Cipher Suite (CS), and Compression Method (CM) se codifican de forma similar que en el ClientHello. El random siempre se especifica.

3.3.4 Implementación

La implementación de las funcionalidades descritas anteriormente, como en las demás secciones de este trabajo se realizan sobre Contiki, y la evaluación de los resultados se hará sobre dicha implementación.

La implementación, llevada a cabo por Raza et al.[25], consta de cuatro componentes principales: DTLS, CoAP, un módulo de integración y la compresión de la cabecera DTLS.

En la implementación se añade el mecanismo de compresión de cabecera descrito como una extensión de la implementación 6LoWPAN de Contiki. La capa 6LoWPAN se encuentra entre la capa IP y la capa de acceso de control al medio (MAC). Los paquetes de la capa IP se consideran de entrada y los de la capa MAC de salida. Decir que pese a que se utilice el método de compresión para DTLS descrito anteriormente, esto no quiere decir que se pierda la seguridad E2E que DTLS ofrece. Dentro de la red 6LoWPAN, los paquetes se envían comprimidos por motivos de eficiencia pero una vez llegado el paquete al 6BR, se descomprimen y no pierden ninguna propiedad.

3.3.5 Evaluación

La evaluación se realiza sobre sensores reales que corren el sistema operativo Contiki. Se utilizan WiSMote [22] como plataforma hardware. Cuentan con 16 un microcontrolador de 16 MHz MSP430 5-Series 16-bit RISC, 16 kB de RAM, 128 kB de

ROM y un transmisor IEEE 802.15.4 (CC2520). Dicho sensor cuenta con la capacidad necesaria para el uso de DTLS. El transmisor CC2520, comentado anteriormente, aunque cuenta con capacidad criptográfica por hardware, para la evaluación se utiliza AES software; el objetivo de la evaluación es comprobar la viabilidad del uso de DTLS, ya se ha comentado anteriormente la mejora que supone el uso de soporte hardware en el caso de realizar operaciones criptográficas.

3.3.5.1 Reducción en el tamaño de paquete

Mediante el uso de DTLS comprimido se reduce significativamente el tamaño de paquete. En la Tabla 6 se observa la reducción del número de bits de las cabeceras, lo que se traduce en una reducción de tiempo necesario para el envío, además de reducir la fragmentación de paquetes. Además, se consigue alargar la vida útil de la red reduciendo el número de bits que se transmiten por la misma, ya que las comunicaciones entre nodos consumen alrededor de 10 veces más energía que la computación dentro del propio nodo.

Cabecera DTLS	Sin comprimir (Bits)	Comprimido (Bits)	Ratio de compresión
Record	104	40[1]	62%
Handshake	96	24[1]	75%
ClientHello	336[2]	264[2]	23%
ServerHello	304	264[3]	14%
CertificateRequest	40	0	100%

Tabla 6: Reducción en tamaño de cabeceras usando 6LoWPAN-NHC

[1] Se requiere un byte adicional para codificar las cabeceras Record y Handshake

[2] Algunos campos tienen una longitud variable. Se consideran aquellos que se envían siempre

[3] Se envía el tamaño completo del random por motivos de seguridad. Los demás campos pueden ser omitidos

Se observa cómo disminuye el número de bits cuando la compresión DTLS se encuentra activada.

3.3.5.2 Requisitos de memoria RAM y ROM

Característica	ROM (Bytes)	RAM (Bytes)
DTLS Crypto (SHA-256, CCM, AES)	6590	2868
DTLS	10662	989
Contiki OS	32145	4979
CoAP	8632	582
Compresión DTLS	2820	1
Total	60849	9419

Tabla 7: Requisitos de memoria ROM y RAM

La implementación de DTLS incluyendo las funcionalidades criptográficas requiere 16.8 kB de ROM y 3.7 kB de RAM.

El mecanismo de compresión de DTLS requiere 2820 bytes de ROM y 1 de RAM. La ROM total utilizada en Contiki para la fragmentación y la compresión (sin el uso de la compresión DTLS) es de 3782 bytes, lo que significa que la compresión DTLS utiliza prácticamente la misma cantidad de estos recursos que el estándar 6LoWPAN. Los sensores de hoy en día no tienen problema para soportar este uso de memoria y dejar memoria suficiente para el uso por parte de las aplicaciones.

3.3.5.3 Rendimiento en tiempo de ejecución

En esta sección se lleva a cabo una evaluación de la implementación de DTLS comprimido, desde el punto de vista energético y temporal.

Para llevar a cabo el cálculo de la energía consumida por el nodo para este fin, se utiliza la siguiente ecuación:

$$Energy\ [mJ] = \frac{ticks\ x\ I\ [mA] x\ Voltaje\ [V]}{ticks\ por\ segundo}$$

Coste de la compresión

A continuación se evalúa la eficiencia energética en el uso de DTLS comprimido. El Gráfico 8 muestra el consumo de energía en la compresión/descompresión de los mensajes DTLS que forman parte del handshake.

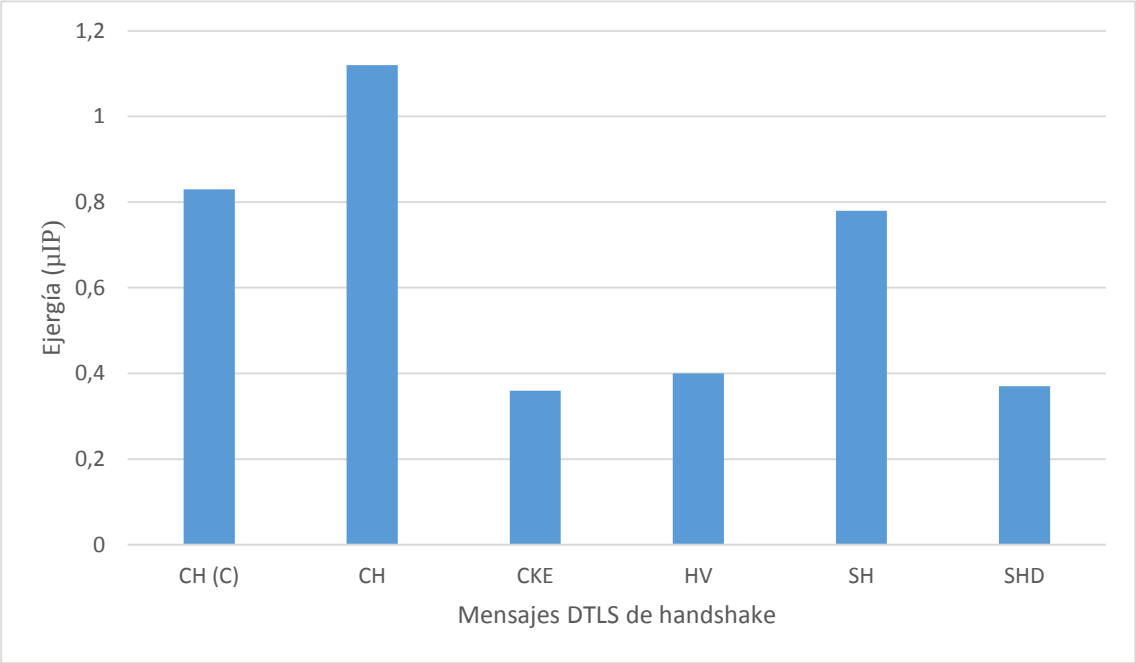


Gráfico 8: Consumo de energía por los mensajes que forman el handshake, individualmente

Cada mensaje handshake consiste en las dos cabeceras Record y Handshake. Un handshake DTLS que utiliza claves pre-compartidas se produce un coste de 4.2 µJ por la compresión.

Inicialización CoAP

Durante la inicialización CoAP se establece una sesión seguro entre dos puntos mediante el protocolo DTLS Handshake. Para llevar a cabo este proceso se utilizan tanto la cabecera Record como Handshake, lo que significa que ambas pueden comprimirse. Debe evaluarse la diferencia entre el aumento de la energía consumida en el nodo por llevar a cabo la compresión de las cabeceras y la disminución del consumo de ésta por utilizar paquetes de menor tamaño y, por lo tanto, disminuir el número de bits que se envía a lo largo de la red. La Tabla 8 muestra el consumo de energía producido en un handshake, tanto utilizando compresión como sin hacerlo.

Compresión	Servidor	Cliente	Total (µJ)
Sin compresión	1756.66	1311.65	3068.31
Con compresión	1467.54	1143.47	2611.01

Tabla 8: Energía consumida por los nodos cliente y servidor durante el handshake. Se produce un aumento de la eficiencia en torno al 15% utilizando compresión

Anteriormente se ha visto que el hecho de llevar a cabo la compresión por parte de un nodo de las cabeceras que forman el handshake DTLS tienen un coste de 4.2 µJ, un valor insignificante relacionado con el coste de transmisión de datos que supone.

Petición-Respuesta CoAP

Una vez que se ha completado la fase de inicialización, un nodo puede enviar y recibir mensajes CoAP seguros usando el protocolo Record DTLS. Aunque el protocolo handshake es más costoso, éste solo se suele realizar una vez.

Las medidas se han llevado a cabo desde que el cliente se prepara la respuesta CoAP y para tras la recepción de la repuesta por parte del servidor y el procesamiento de la misma.

En el Gráfico 9 puede apreciarse el consumo energético dependiendo del número de bytes de datos que se transfieren por parte de los nodos. Llegando a 48 bytes, en el caso de CoAP plano, comienza a observarse el problema de la fragmentación, disparándose el consumo a causa de ésta. El consumo energético en el caso de las peticiones utilizando CoAP comprimido es siempre del 10%, mientras que en el caso de los mensajes de respuesta depende del tamaño de la carga útil. En cualquier caso, el ahorro energético en este último caso en los ejemplos propuestos oscila entre el 4 y el 26%.

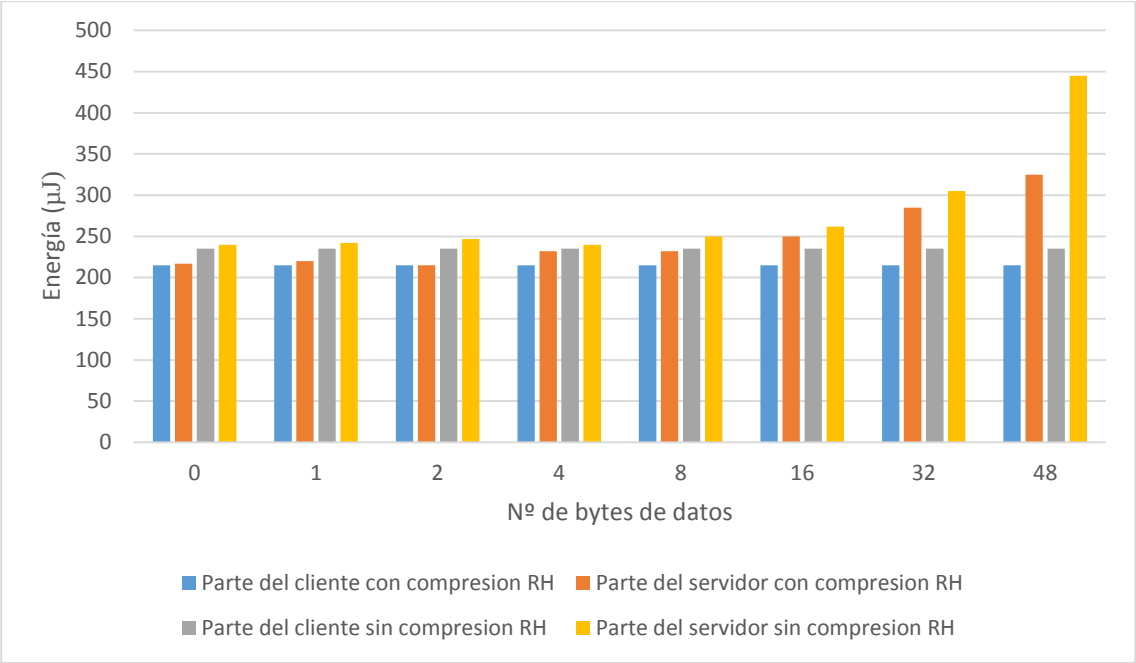


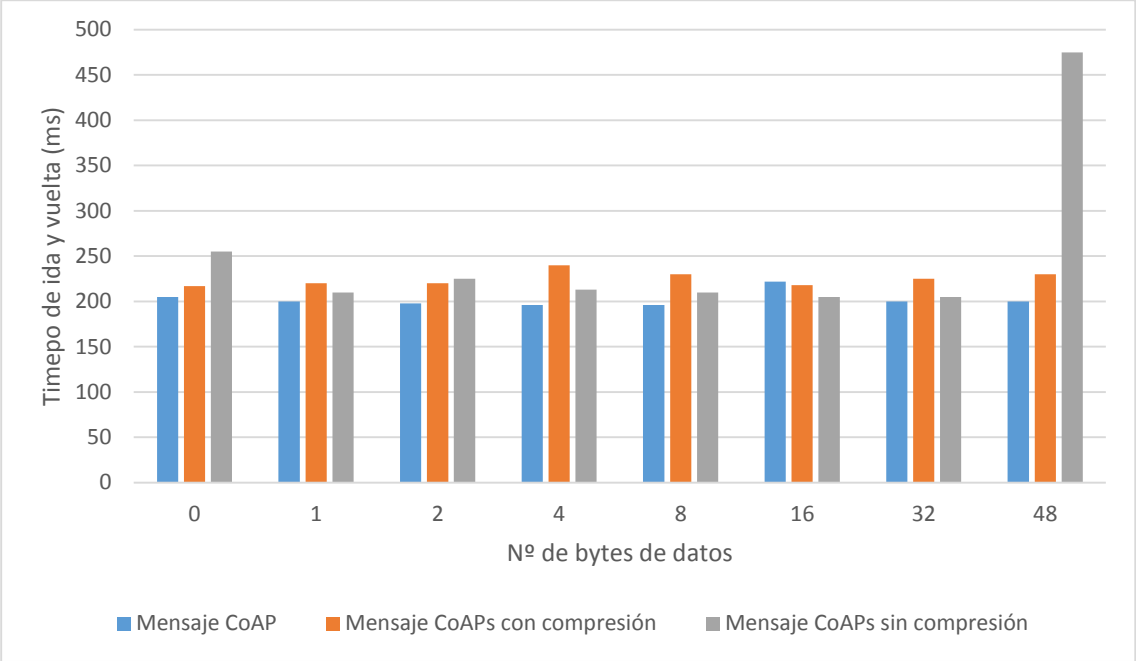
Gráfico 9: Energía consumida por el cliente y el servidor en el envío de mensajes CoAP comprimidos y descomprimidos de diferentes tamaños

La reducción del tamaño de paquete afecta también al RTT (Round Trip Time) de petición-respuesta (tiempo de ida y vuelta de un mensaje). En el caso en que no se emplea RDC (Radio Duty Cycling) el RTT es un 1.5% menor, excepto en el caso en que los paquetes son de 48 bytes de carga útil, donde es un 77% menor. Esto es debido a la fragmentación necesaria para enviar dicho paquete, ya que el original excede el MTU.

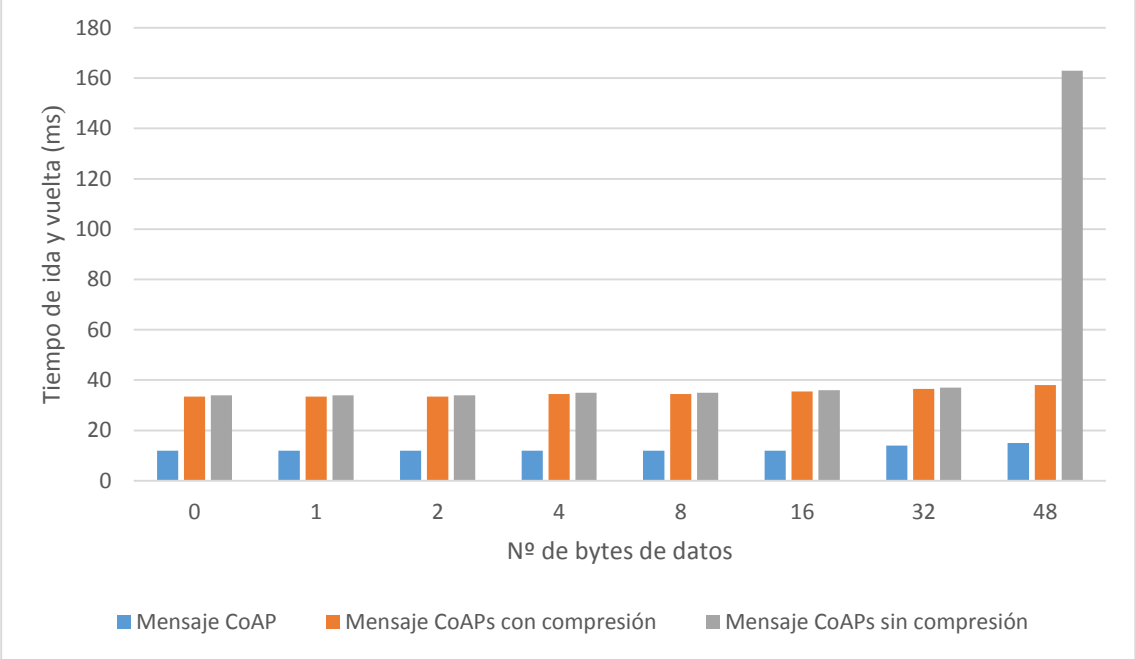
El RTT de los mensajes CoAP sin seguridad es 1/3 del RTT en el caso de los mensajes con CoAPs comprimido. Este valor se mantiene, ya que la versión comprimida de DTLS impide que se necesite fragmentación para este tamaño de paquete.

En el caso del uso de RDC Gráfico 10 a), vemos que los valores tanto de CoAP como CoAPs plano y comprimido se mantienen más o menos constantes, excepto en el caso de un tamaño de 48 bytes. Esto se debe al efecto de RCD; RCD se trata de un mecanismo de ahorro de energía que hace que el chip de radio del nodo se encuentre en estado de reposo la mayor parte del tiempo. Este mecanismo hace que se produzca un ahorro energético, con el consiguiente coste de latencia. El emisor envía paquetes, pero éstos no son procesados por el receptor hasta que el chip de radio se activa, por lo que si el receptor se encuentra en estado de reposo, si el tiempo requerido para el envío no excede este intervalo de reposo, este tiempo no

se ve incrementado. En el peor de los casos, la espera será del intervalo de tiempo que el receptor tarda en activarse. Aunque en este caso la mejora de Lithe con respecto a CoAPs plano es menor, podemos observar que en el caso de paquetes con 48 bytes de datos, la diferencia se encuentra entorno al 50%.



a) Con Duty Cycling (RDC)



b) Sin Duty Cycling (RDC)

Grafico 10: Comparación de RTT de CoAP, CoAPs ligero (Lithe) y CoAPs plano

3.3.5.4 Conclusión

Como se puede observar con los datos obtenidos en la evaluación, la implantación de CoAPs en redes 6LoWPAN es una opción de seguridad viable. Mediante el método de compresión de DTLS se obtiene un menor tiempo de respuesta, diferencia que va en aumento en función de los fragmentos que evita el mecanismo de compresión. Hay que tener en cuenta que, debido a los dispositivos que forman este tipo de redes, lo más probable es que éstas utilicen Radio Duty Cycling ya que la escucha por parte del chip de radio del nodo consume mucha más energía que la computación interna en el mismo (en torno a 10 veces más). Por lo tanto, como puede verse en el Gráfico 10 (a), el incremento en el tiempo de respuesta con el uso de RDC y CoAP comprimido es insignificante.

3.4 SVELTE: Detección de intrusos en tiempo real en el IoT

A diferencia de las redes convencionales de sensores, en las redes 6LoWPAN cada uno de los nodos que la forman tienen capacidad para conectarse a internet de forma directa, lo que las hace aún más vulnerables a la intrusión.

Pueden utilizarse mecanismos de detección de intrusos (IDS) para WSNs, aunque la mayoría de ellos asumen que no existe un punto de control central, no existe seguridad en los mensajes de la red y que los nodos no se identifican de manera global. En redes 6LoWPAN existe el 6BR central, necesitamos añadirle seguridad a los mensajes E2E y cada uno de los nodos cuenta con una dirección IP que los identifica. Por lo tanto, la solución empleada debe tener en cuenta que los nodos son accesibles globalmente, tienen recursos limitados, la naturaleza de la red es de pérdida de paquetes y utiliza protocolos como 6LoWPAN, RPL o CoAP.

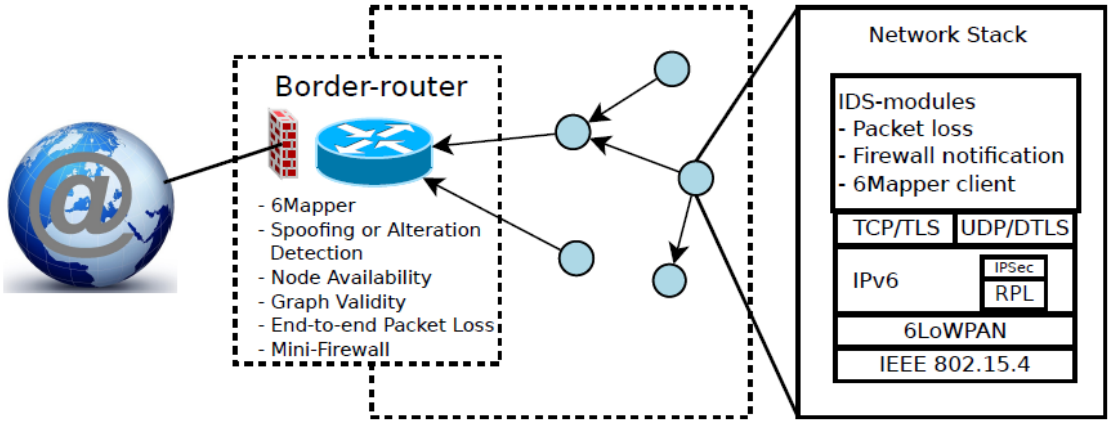


Figura 25: Configuración de IoT donde los módulos con IDS están centralizados

SVELTE se trata de un IDS diseñado específicamente para el IoT. Los ataques a la capa de red y los ataques de enrutamiento son los más comunes en redes inalámbricas, por lo que este sistema protege contra ataques de tipo sybil y la clonación del ID del nodo. Este mecanismo se evaluará para diferentes ataques, como el de agujero negro o el de expedición selectiva, todos ellos comentados en la sección 2 de este trabajo.

SVELTE utiliza RPL como protocolo de enrutamiento. Consta de dos partes principales, el 6LoWPAN Mapper (6Mapper) y de módulos de detección de intrusos.

Una de las decisiones más importantes a la hora de instaurar un sistema de este tipo es el lugar en el que se sitúa el IDS en la red. En este trabajo se emplea una posición híbrida del sistema, localizando SVELTE de forma intensiva en el 6BR y utilizando los módulos ligeros correspondientes en los nodos.

3.4.1 Intrusion Detection System (IDS)

Un IDS es un mecanismo para detectar ataques contra el sistema o la red analizando la actividad en la misma o en el sistema en sí. Cuando se detecta un ataque, el IDS almacena información acerca del mismo y/o reporta una alarma. Los mecanismos de detección en IDS están basados o en firmas o en la detección de anomalías en la red.

Las detecciones basadas en firmas compara el comportamiento actual de la red con patrones de ataques predefinidos. Éstas firmas se preconfiguran en el dispositivo, de manera que cada una de ellas coincide con un ataque en concreto. Esta forma de funcionamiento es fácil de utilizar, aunque se necesita conocer el comportamiento específico de la red ante cada uno de los ataques, así como meter datos acerca de los ataques de manera manual, lo que lo hace más estático. Además, el coste de almacenamiento crece en función del número de ataques que conoce el mecanismo.

Las detecciones basadas en anomalías establecen un comportamiento base, estudiando la red, con lo que todo comportamiento que se separe de la línea de funcionamiento normal se trata como sospechoso. Aunque tienen la capacidad de percibir prácticamente cualquier tipo de fallo, también tienen una enorme cantidad de falsos positivos, es decir, trata como anomalía comportamientos en los que no hay ningún tipo de ataque. De la misma forma, también producen falsos negativos en aquellos ataques que no cambian el funcionamiento de la red de forma brusca.

En la solución que se estudia se utiliza una mezcla de los dos mecanismos mencionados, buscando un punto intermedio entre el coste en memoria del mecanismo basado en firmas y el coste computacional del mecanismo basado en anomalías.

Cuando se detecta un ataque, el objetivo es mitigar su efecto y eliminar el atacante de la red, lo que requiere la identificación del nodo atacante. La manera más sencilla de eliminar un nodo malicioso es ignorándolo. Para detectarlo, el sistema no puede basarse ni en la dirección IP ni la MAC, ya que ambas pueden falsificarse fácilmente. Una manera de ignorarlos es el uso de blacklist (lista negra), donde se incluyen los nodos maliciosos y una whitelist (lista blanca), donde se encuentran definidos los nodos confiables. Para que estos mecanismos sean efectivos, debe evitarse que el nodo sea capaz de cambiar su identidad. SVELTE utiliza whitelist.

3.4.2 SVELTE: Un IDS para el IoT

Como se ha mencionado ya en varias ocasiones en este trabajo, cualquier mecanismo que dote de seguridad a este tipo de redes debe tener en cuenta su naturaleza no fiable, así como los bajos recursos de los nodos que la forman. SVELTE está diseñado para una red 6LoWPAN que utilice tecnologías que doten a los mensajes de seguridad, como IPSec y DTLS, que proveen seguridad extremo a extremo (E2E).

Colocación de SVELTE

La colocación de un IDS es una decisión importante. Dada la naturaleza de las redes 6LoWPAN y su distribución (Figura 25), se utiliza una situación del sistema híbrida, centralizada y distribuida, situándolo tanto en el 6BR como en los nodos.

SVELTE tiene tres módulos centralizados que se alojarán en el 6BR. El primero, llamado 6Mapper, reúne información sobre el RPL y reconstruye la red en el 6BR (sección 3.4.2.1). El segundo módulo es un componente de detección de intrusos que analiza la información de mapeo con el objetivo de detectar intrusión en la red (sección 3.4.2.2). El tercero se trata de un mini-firewall distribuido, diseñado para filtrar tráfico

no deseado antes de que éste entre en la red de nodos (sección 3.4.2.3). Los módulos centralizados cuentan con dos módulos ligeros descentralizados en cada nodo. El primero proporciona información de mapeo al 6BR para que éste detecte intrusos; el segundo trabaja con el firewall centralizado. Cada nodo cuenta con un tercer módulo para manejar la pérdida de paquetes extremo a extremo (sección 3.4.2.2).

3.4.2.1 6LoWPAN Mapper

Un componente esencial en SVELTE es 6LoWPAN Mapper (6Mapper) que construye el Destination-Oriented Directed Acyclic Graph de RPL en el 6BR y lo complementa con información acerca de los vecinos de cada nodo. Para llevar a cabo esta operación, 6Mapper manda peticiones periódicas a los nodos. El paquete de respuesta contiene la información necesaria para identificar un DODAG RPL. Incluye el ID de la instancia RPL (IID), el ID DODAG y el número de versión DODAG. También incluye una marca de tiempo (timestamp) para saber cuándo se recibió la información de mapeo. El tamaño de estos paquetes de petición es de 5 bytes.

Cada nodo contesta a este mensaje de petición añadiendo su ID, su rango, la ID de sus padres y todos los IDs de sus vecinos y sus rangos. El tamaño básico de estos paquetes de respuesta es de 13 bytes, a los que se le añade 3 bytes por cada vecino. El formato de estos paquetes queda ilustrado en la Figura 26.

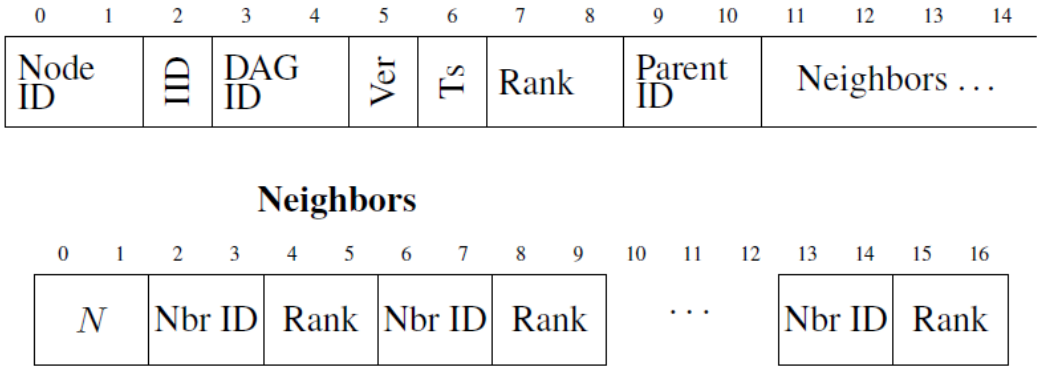


Figura 26: Formato de paquete de respuesta de mapeo

6Mapper con comunicación segura y autenticada

Es probable que la cabecera de autenticación de IPSec (AH) o la seguridad en la capa de enlace IEEE 802.15.4 esté activada dotando de protección a la integridad de las

cabeceras IP. En este caso no se incluye la ID en la respuesta, ya que coincide con la ID de origen de la cabecera IP. Cuando el host 6Mapper tiene la misma dirección IP que la raíz DODAG no es necesario incluir la DODAGID. En los paquetes de petición el origen sería el DODAG, mientras que el destino sería el nodo al que se le pide la información. Si los paquetes de mapeo se transfieren de forma fiable, por ejemplo mediante el uso de CoAPs que emplea reconocimientos, no hay necesidad de enviar la marca de tiempo ya que estamos seguros de que éstos llegarán en el tiempo de espera programado por el protocolo subyacente. Cuando la comunicación en 6LoWPAN es autenticada y confiable, el tamaño de los paquetes de petición y respuesta se reducen a 1 y 8 bytes respectivamente.

6Mapper RPL unidireccional

Algunas implementaciones RPL admiten únicamente el tráfico destinado a la raíz DODAG, normalmente el 6BR. En este caso no se envían peticiones de mapeo a los nodos, sino que se espera a que éstos manden dichos paquetes, que lo hacen de forma periódica. De esta forma se reduce el tráfico en la red y, por lo tanto, el consumo de energía.

Inconsistencias válidas en 6Mapper

Es posible que en el 6Mapper las respuestas de mapeo sean incompatibles entre sí, lo que puede dar lugar a falsos positivos si no se manejan de forma adecuada. Esto sucede cuando la información de mapeo proporcionada por un nodo se convierte en obsoleta o cuando un atacante cambia la información de forma intencionada.

Esto puede verse en el siguiente ejemplo, donde $D_a(N)$ representa la distancia actual a la que se encuentra el nodo N y $D_m(N)$ representa la distancia a la que el 6Mapper piensa que se encuentra el nodo N y el nodo P es padre del nodo C;

- El nodo P envía información acerca de su distancia al 6Mapper $D_a(P)=1024$ y $D_m(P)=1024$.
- P recalcula su distancia: $D_a(P)=512$ y $D_m(P)=1024$.
- El nodo C recibe la actualización de la información de posición de su padre P.
- P recalcula su distancia $D_a(C)=768$.
- P envía información acerca de su posición al 6Mapper: $D_a(P)=768$ y y $D_m(P)=768$.
- El estado de la red sería el siguiente:

- D_a(P)=512 y D_m (P)=1024
- D_a(C)=768 y D_m (C)=768

Tal como se encuentra la situación en la red: $D_a(P) < D_a(C)$; sin embargo, el 6Mapper tiene otra información, con lo que piensa que el nodo C se encuentra más cercano que el nodo P, ya que : $D_m(P) > D_m$ (C).

Requisitos en el mapeo

Para que 6Mapper sea completamente efectivo se necesita que los paquetes destinados al mapeo de la red sean indistinguibles del resto; de lo contrario, un atacante podría filtrar estos paquetes y modificarlos.

Lo primero sería el encriptado de los paquetes. Como se ha comentado anteriormente, se asume que se está utilizando seguridad en capas superiores del protocolo, como IPSec o DTLS. Otro requisito sería evitar que el encabezado del paquete revele que éste va dirigido al 6Mapper. Si se utiliza una única dirección IP para el 6Mapper, dado que éste recibirá información acerca de la posición de forma asidua por parte de todos los nodos de la red, el atacante podría determinar que los paquetes cuyo destino es dicha dirección contienen información acerca del mapeo. Una solución para dicho problema sería utilizar una dirección IP para 6Mapper por cada uno de los nodos de la red.

3.4.2.2 Detección de intrusos en SVELTE

En este trabajo se muestran tres técnicas de detección que utilizan 6Mapper. Primordialmente, éstas técnicas detectan información alterada o falsa, agujeros negros y ataques de repetición selectiva (Sección 2).

Detección de inconsistencias en el grafo de la red

Existen diferentes tipos de ataques que pueden repercutir en inconsistencias en el grafo, como un nodo malicioso que proporcione información incorrecta acerca de la posición de sus vecinos, o que dada la pérdida de paquetes de estas redes, la información de posición enviada por el nodo no llegue al destino.

Con el fin de detectar información incorrecta y estar seguros de que ésta es consistente se comprueban todos los nodos de la red. 6Mapper proporciona información acerca de los nodos de la red, su distancia e ID, la de sus padres y la de sus vecinos. Se comprueba

que la información proporcionada por 6Mapper coincide con la situación real de la red, comprobando los nodos uno por uno. Es posible que se produzca una alerta debida a una inconsistencia válida, como se describe en el apartado anterior (Sección 3.4.2.1).

Para distinguir entre inconsistencias válidas e inválidas en la red, nos fijamos en información como el número de fallos detectados y la diferencia entre la distancia real del nodo $D_a(N)$ y la distancia que del nodo en 6Mapper $D_m(N)$. Se utiliza un límite, denominado como FaultThreshold en el Algoritmo 1, tomando el nodo como defectuoso si el número de inconsistencias de este nodo con los demás supera el límite establecido. Para no tomar las inconsistencias válidas como inválidas, se consideran inválidas únicamente cuando la diferencia entre la información de distancias entre dos nodos son un 20% superior de la media. Éste valor se ha tomado teniendo en cuenta una evaluación empírica de SVELTE.

```
Require: N - A list of nodes
    for Node in N do
        for Neighbor in Node.neighbors do
            Diff = |Node.neighborRank(Neighbor) Neighbor.rank|
            Avg = (Node.neighborRank(Neighbor)+Neighbor.rank)=2 {If the absolute
            difference is greater than 20% of the ranks average}
        if Diff > Avg * 0:2 then
            Node.fault = Node.fault + 1
            Neighbor.fault = Neighbor.fault + 1
        end if
    end for
end for
for Node in N do
    if Node.fault > FaultThreshold then
    Node.rank = Rank reported for Node by any neighbor
        for Neighbor in Node.neighbors do
            Node.neighborRank(Neighbor)= Neighbor.rank
        end for
    end if
end for
```

Algoritmo 1: Detección y corrección de inconsistencias RPL DODAG

Corregimos la información cuando se producen las dos condiciones anteriores, es decir, una vez que tengamos inconsistencias grandes en un nodo. La información errónea correspondiente a un nodo se corrige cambiando la distancia conocida para 6Mapper sustituyéndola con la información enviada por uno de sus vecinos.

La información de vecinos se actualiza con la información reportada directamente por sus vecinos. Una vez que se detecta que una inconsistencia de enrutamiento es un resultado de ataque deliberado, SVELTE o elimina el nodo defectuoso o corrige la inconsistencia. SVELTE mantiene un registro de incidencias y si es la primera vez que se detecta un nodo como malicioso no se elimina inmediatamente, ya que puede ser una falsa alarma o como resultado de un ataque pasivo; en este caso la información defectuosa se corrige como se comentada anteriormente. Sin embargo, si el mismo nodo se detecta como defectuoso de nuevo, es eliminado mediante la supresión de su entrada de la lista blanca de 6Mapper.

Comprobación de la disponibilidad de un nodo

Es importante detectar si un nodo o un conjunto de ellos se encuentran disponibles y operan de forma correcta. Cuando un nodo se encuentra comprometido puede lanzar ataques con el fin de interrumpir los procesos de la red, como ataques de expedición selectiva o de agujero negro.

Dependiendo de la implementación RPL y la configuración, podemos utilizar la tabla de enrutamiento RPL en la raíz RPL DODAG como base para los nodos disponibles en la red. Como se utiliza una lista blanca de nodos válidos en la red para el control de acceso, también podríamos utilizar esa lista como base para la detección.

Cuando comparamos los nodos de la lista blanca con los nodos de nuestra RPL DODAG, las diferencias se tratan de nodos fuera de línea o nodos no autorizados. Es decir se cumple la siguiente ecuación, donde W es la lista de nodos pertenecientes a la lista blanca y R los nodos conocidos por el RPL en la raíz DODAG RPL y siendo / el complemento relativo, se cumple que O:

$$W \setminus R = O$$

Es decir, O es el conjunto de nodos incluidos en la lista blanca que no son conocidos por el RPL porque se encuentra offline y todos los nodos que no se encuentran en la lista blanca porque no son confiables. Por lo tanto, los nodos que no se encuentran en O no es posible determinar si no lo están porque son nodos que funcionan correctamente (están en la lista blanca y online) o son nodos que se encuentran offline.

Extendiendo el método anterior con la información del 6Mapper, puede realizarse un filtrado y conocer cuáles pertenecen a cada grupo, lo que nos ayuda a detectar ataques de expedición selectiva, ya que dicho ataque envía información RPL aunque se filtren mensajes de aplicación para el mismo, lo que hace que parezca operativo. Puede

utilizarse la siguiente realización, donde M representa los nodos conocidos por el 6Mapper y F los nodos filtrados (nodos con un funcionamiento correcto):

$$W \setminus M = F$$

Como 6Mapper contiene un registro acerca de la última vez que obtuvo un paquete desde un nodo, puede saberse que nodos forman parte de los confiables. Para evitar los errores debidos a la pérdida de paquetes de estas redes, se define un límite sobre el tiempo en que se recibió el último paquete desde un nodo, entendiéndolo como el número de peticiones sin respuesta acerca de la información de posición de un nodo.

```
Require: W - Set of whitelisted nodes
Require: M - Set of nodes known to the 6Mapper
    F = {F will contain the filtered nodes}
    for Node in W do
        if Node in M and M[Node].lastUpdate() > RecencyThreshold then
        F.add(Node)
        end if
    end for
    return F
```

Algoritmo 2: Detección de nodos filtrados

Validez del grafo de enrutamiento

Alterando el grafo de enrutamiento, el atacante puede remodelar la topografía de la red y controlar el tráfico a su antojo. Esta acción se emplea en ataques como los de agujero negro y expedición selectiva (Sección 2).

Utilizando SVELTE puede detectarse la mayoría de los ataques de agujero negro mediante el análisis de la topología de la red, ya que si se detecta una inconsistencia en el grafo significa que se está produciendo uno de estos ataques. Por ejemplo, en ningún caso un nodo hijo tendrá menos distancia que su padre; un caso que muestre dicha información nos asegura que se está produciendo una modificación del grafo de enrutamiento y, por lo tanto, un ataque.

Hay que recordar los casos en los que se producen inconsistencias válidas, como se ha comentado anteriormente en este trabajo (Sección 3.4.2.1). Por lo tanto, es necesario evitar que se produzcan estos falsos positivos. Para ello se toma como positivo cuando

se da esta situación en el mismo nodo más de una vez. Esto se describe en el Algoritmo 3, donde FaultThreshold es un estado global que se mantiene en distintas ejecuciones del algoritmo. En RPL la distancia entre cualquier nodo y su padre es de al menos MinHopRankIncrease.

```
Require: N - A list of nodes
    for Node in N do
        if Node.rank +MinHopRankIncrease < Node.parent:rank then
            Node.fault = Node:fault + 1
        end if
    end for
    for Node in N do
        if Node.fault > FaultThreshold then
            Raise alarm
        end if
    end for
```

Algoritmo 3: Búsqueda de inconsistencias en las distancias

En la mayoría de los casos los ataques de agujero negro se detectarán usando este método. Cuando un atacante quiere dirigirse el tráfico intentará hacer ver que existe una menor distancia hacia él que hacia su padre.

Adaptación a la pérdida de paquetes extremo a extremo

Dada la naturaleza de estas redes de pérdida de paquetes, si se introduce un nodo malicioso en la red que esté filtrando paquetes, el protocolo de la capa de transporte no es capaz de detectarlo. Por lo tanto, se necesita de un mecanismo auxiliar que sea capaz de mitigar estos efectos. El objetivo es que si un nodo se encuentra filtrando paquetes, modificar la ruta de manera que no pase información por el mismo.

Todos los nodos tratarán de enviar paquetes al nodo principal, por lo que los nodos que se encuentren en la ruta del nodo malicioso se darán cuenta de que éste filtra datos de aplicación, por lo que se realiza un cambio de ruta de los mismos, es decir, el siguiente salto para ese paquete (en el Algoritmo 4 se emplea un cambio del 20%).

Require: dest - The destination with packet loss
nexthop =getNexthop(dest)
nexthop.metric = nexthop.metric * 0.8

Algoritmo 4: Adaptación extremo a extremo a la pérdida de paquetes

Protección contra ataques de tipo Sybil y CloneID

En un ataque de tipo Sybil, un nodo atacante presenta varias identidades a la red, situadas todas ellas en un mismo nodo físico. El ataque de clonación de identidad (Clone ID) se basa en la copia de una misma identidad en varios nodos físicos. Ambas tienen como objetivo hacerse con el control de la red.

6Mapper toma como identidad de un nodo la última que recibe acerca del mismo; por lo tanto, el ataque de tipo CloneID no repercute en el funcionamiento de la red de nodos, ya que no habría variación con la situación en que un nodo enviase su información dos veces al 6Mapper. En un ataque de tipo Sybil, cada uno de los nodos virtuales que se encuentran en un mismo nodo físico son tratados como nodos separados, por lo que no afecta al funcionamiento normal de la red.

3.4.2.3 Mini-firewall distribuido

SVELTE protege las redes 6LoWPAN contra ataques de intrusión en la intranet, sin embargo es necesario proteger cada uno de los nodos que la forman contra ataques globales, muchos más potentes. Los cortafuegos normalmente filtran ciertos mensajes de entrada y salida, sin embargo, dado que dotamos a la red de confidencialidad e integridad E2E, es difícil distinguir entre tráfico malicioso y seguro.

Para llevar a cabo dicha protección se utiliza un mini-fireswall distribuido por los nodos; dicho cortafuegos tiene un módulo en el 6BR y en los nodos y está integrado con SVELTE; provee protección sobre atacantes externos conocidos especificados manualmente por el administrador de la red, que puede bloquear ataques en tiempo real.

El nodo de destino dentro de la red 6LoWPAN puede descifrar la información, ya que es él el destinatario, y analiza si se trata de tráfico malicioso. En ese caso, notifica al 6BR en tiempo real para que filtre la información procedente del emisor de la misma. El nodo envía un paquete de petición de bloqueo, incluyendo la IP del nodo que se quiere bloquear. Si se utiliza IPSec con AH y 6Mapper, no es necesario que se incluya la IP del nodo emisor, lo que repercute en un ahorro de 2 bytes en el tamaño de paquete.

El tamaño de este paquete es de 16 o 18 bytes, dependiendo de si se incluye explícitamente la IP del paquete o no.

Para que ningún nodo malicioso mande peticiones de bloqueo falsas con intención de filtrar el tráfico de hosts internos, el filtrado se realiza al nodo emisor de la petición únicamente, hasta que varios nodos pidan el filtrado del mismo host (ReportThreshold Algoritmo 5), momento en el que el filtrado se realiza para sobre ese host para todos los nodos de la red 6LoWPAN.

Require: Host - The host to report
Require: Source - The node that sent the report
Require: GlobalFilter - A set of external hosts to filter towards all nodes
Require: LocalFilter - A map mapping an external host to a set of local nodes. The set describes all nodes that have reported that specific external host.

```
if Host in GlobalFilter then
        return Host already filtered
end if
if Host in LocalFilter then
        Filter = LocalFilter.get(Host)
        {Add Source to the list of nodes blaming Host}
        Filter.add(Source)
        if Filter.size() ≥ ReportThreshold then
        GlobalFilter.add(Host)
        LocalFilter.remove(Host)
        end if
end if
```

Algoritmo 5: Mini-firewall distribuido

3.4.3 Implementación

La implementación, llevada a cabo por el grupo de trabajo Raza et al.[26], se realiza sobre el sistema operativo Contiki, el cual tiene una implementación de RPL (ContikiRPL) sobre la que se implementa 6Mapper, el cortafuegos distribuido y los módulos de detección de intrusos. En la implementación de RPL de Contiki cada nodo mantiene un registro de sus hijos. Mediante el uso de esta función debería saberse qué nodos deberían estar disponibles en la red.

Para probar su funcionamiento, se simula los ataques de agujero negro y expedición selectiva.

3.4.4 Evaluación

En esta sección se evaluará el resultado de la inclusión de SVELTE a una red 6LoWPAN. Para ello se evaluarán aspectos como los positivos reales y el ratio de detección. También se mide el incremento de tamaño de cabecera por el uso de SVELTE, en términos tanto de consumo de memoria como de energía.

Se considera que el 6BR no es un nodo de recursos limitados, sino que se trata de un PC, corriendo 6Mapper nativo.

3.4.4.1 Detección de SVELTE y ratio de positivos reales

En esta sección se evalúa el ratio de detecciones, es decir, el número de nodos maliciosos detectados en comparación con el total de nodos infectados de la red, así como el ratio de positivos reales, es decir, el número de alarmas correctas dividido entre el número total de alarmas dadas por el sistema.

Se utilizan diferentes configuraciones, todas ellas utilizando 6Mapper en modo petición (6Mapper pide información de la localización de los nodos de forma periódica), en este caso cada dos minutos. Tanto las peticiones como los análisis se realizan cada dos minutos, por lo que la primera detección posible se produce a los 4 minutos de funcionamiento de la red, a lo que hay que añadir que un nodo no se toma como malicioso a no ser que se produzca más de un incidente en el mismo (Algoritmo 5), siendo el número de veces 2; por lo tanto, la primera detección que conlleva eliminación de un nodo del mapa se puede producir a los 6 minutos. Lo primero que comprueba el sistema son las inconsistencias en el grafo de nodos, como se describe en la sección 3.4.2.2 (validez del grafo de enrutamiento). Los experimentos se realizan en escenarios con pérdidas y sin ellas (escenario ideal para 6Mapper).

Ataques de agujero negro con y sin pérdidas

El resultado de los descubrimientos en escenarios sin pérdidas muestra un resultado del alrededor del 100% de positivos reales y ningún falso positivo durante la simulación; en este escenario, no se producen pérdidas de paquetes y transmisión de los paquetes es muy rápida, lo que se traduce en que el mapa de la red es una representación exacta del estado actual de la red.

En escenarios con pérdidas de paquetes, el ratio en redes pequeñas es menor conforme aumenta el número de nodos que la forman; esto es debido a que, conforme aumenta

el tamaño, más inexacto es la representación de la red que tiene el RPL, dada la naturaleza de pérdida de paquetes de la red. En el Gráfico 11 puede observarse lo efectivo que es el sistema de protección así como, a medida que aumenta el tiempo de actividad de la red, el número de ataques detectados con éxito en relación con las alarmas dadas por el sistema aumenta; es decir, cuando la red RPL se estabiliza.

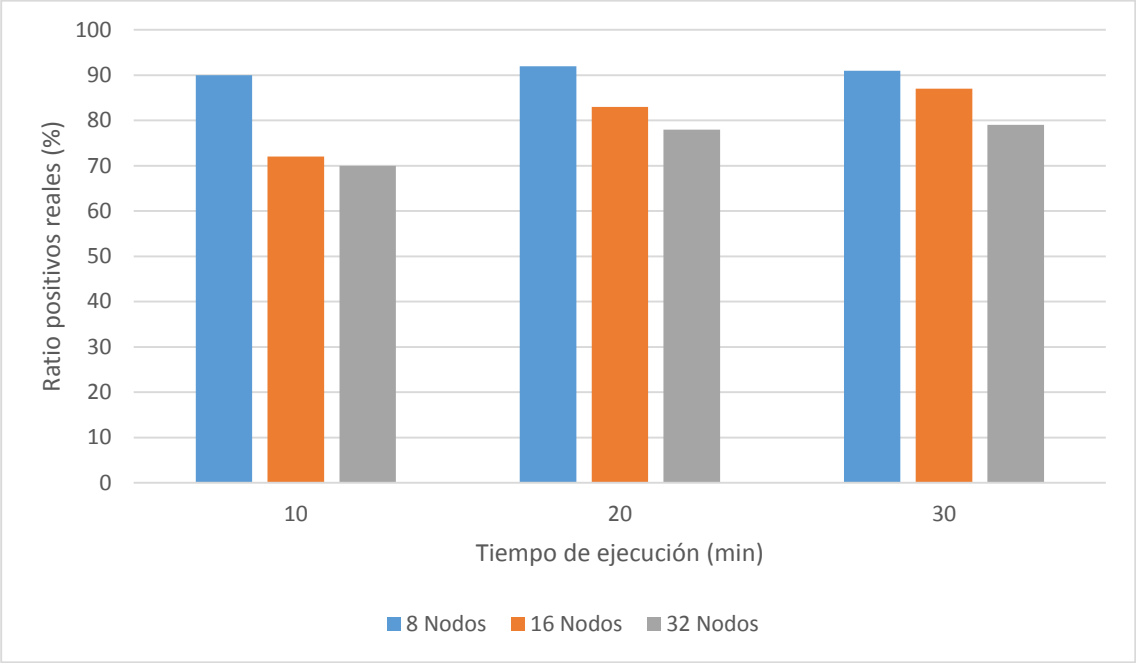


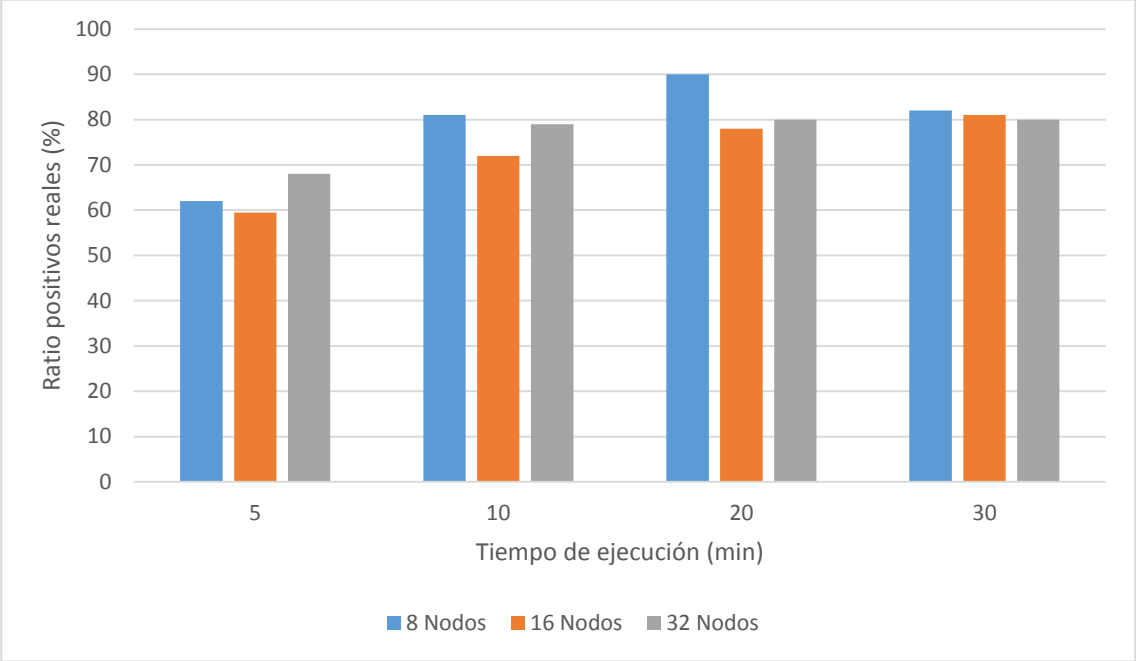
Gráfico 11: Relación tiempo de ejecución y ratio de positivos reales con diferentes números de nodos

Expedición selectiva con y sin pérdida de paquetes

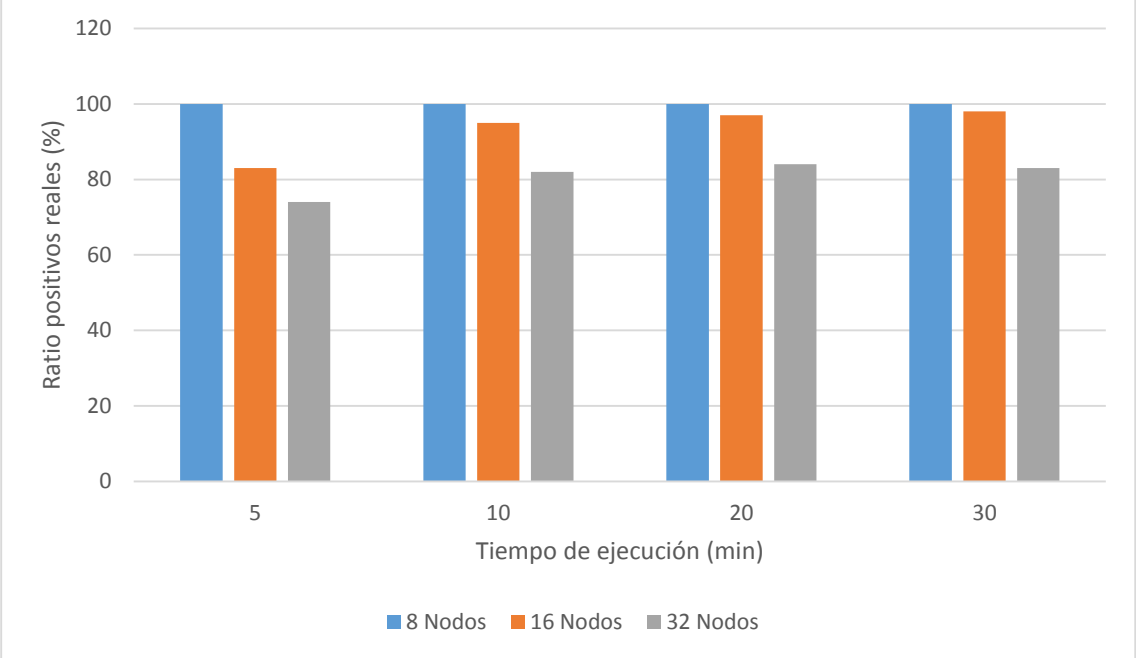
En un ataque de expedición selectiva, el nodo malicioso filtra la información de la red que pasa por él, por lo que 6Mapper no es capaz de obtener información sobre los hijos de este nodo infectado. En este caso, al contrario que en el ataque de agujero negro, en redes sin pérdidas el resultado no es siempre del 100% de aciertos, ya que depende de la topología de la red (Gráfico 12 b).

En el Gráfico 12 a) puede verse el resultado de SVELTE en una red con pérdida de paquetes. Salvo en el caso de una red con 32 nodos y 5 minutos de funcionamiento de ésta (inestabilidad de la red), en todos los casos el ratio de alarmas reales supera el 80 por ciento, ratio que aumenta conforme el tiempo aumenta. Como se puede prever, si aumentamos el número de ocasiones que un nodo debe ser sospechoso para lanzar una alarma y eliminarlo de la lista blanca, el ratio de falsos positivos en relación con el total

de alarmas aumenta; sin embargo, esto conlleva una menor cantidad de ataques detectados, ya que se es más persuasivo con las incoherencias.



a) Red con pérdida de paquetes, detección de ataques de expedición selectiva



b) Red sin pérdida de paquetes, detección de ataques de expedición selectiva

Gráfico 12: Relación tiempo de ejecución y ratio de positivos reales en ataques de expedición selectiva con diferentes números de nodos

3.4.4.2 Rendimiento energético

Como ya se ha comentado en varias ocasiones, los dispositivos que forman estas redes suelen estar alimentados por baterías, por lo que el consumo energético en los mismos es un aspecto crucial con el fin de alargar la vida útil de la red lo máximo posible.

Los dispositivos de recursos limitados de la red se encuentran conectados a un potencial de 3 voltios en todos los casos. Llamaremos como LPM (Low Power Mode) al estado del dispositivo cuando tanto el MCU (microcontrolador) del mismo como el chip de radio de este se encuentran apagados, modo CPU cuando únicamente el MCU se encuentra activo y *escuchando* o *transmitiendo* cuando ambos se encuentran encendidos, llevando a cabo una de las dos operaciones que el propio nombre describe. Así mismo, se contemplan los casos tanto de RDC activo como apagado. Recordar que RDC (Radio Duty Cycling) se refiere al modo en el que el nodo enciende su chip de radio cada cierto tiempo con el fin de ahorrar energía.

Red con Duty Cycling

Se evalúa el consumo en una red RPL una red con Duty Cycling activo, con y sin 6Mapper y detección de intrusos. Se utiliza ContikiMac, accionando el chip de radio 8 veces por segundo y sin tráfico este se encuentra activo el 0.6% del tiempo. Cada uno de los experimentos se lleva a cabo con 8, 16, 32 y 64 nodos simulados Tmote Sky [19], con la misma topología para todas las configuraciones.

El Gráfico 13 muestra el consumo de energía de toda la red de nodos en 30 minutos, calculado de la siguiente manera:

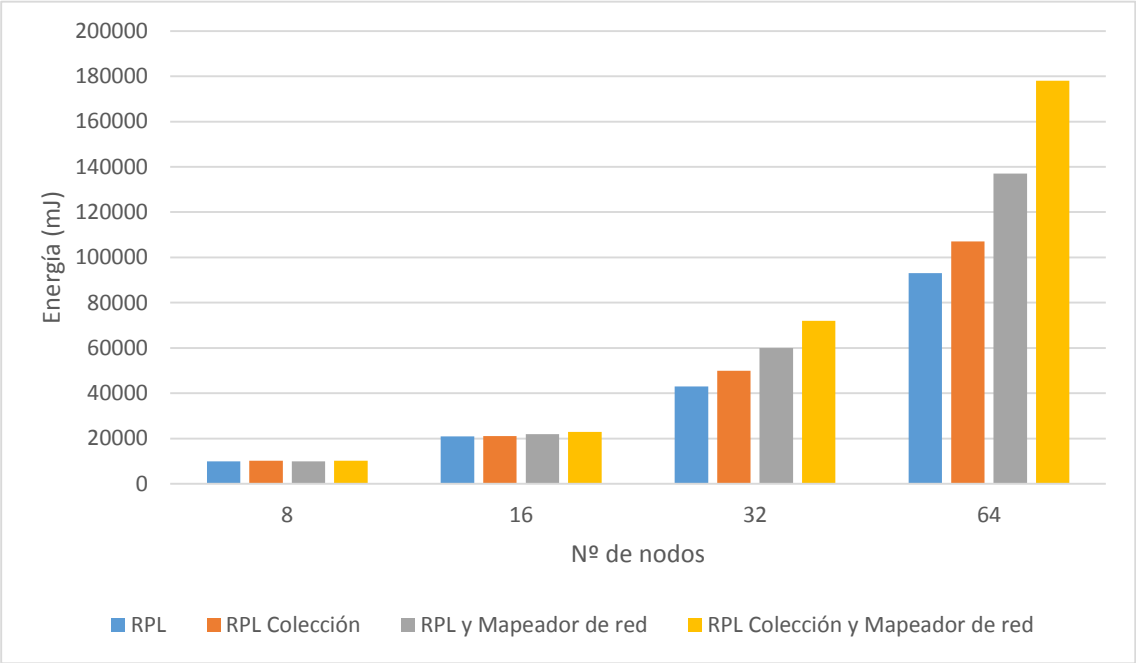
$$\text{Energía (mJ)} = (\text{transmitiendo} * 19{:}5\text{mA} + \text{escuchando} * 21{:}8\text{mA} + \text{CPU} * 1.8\text{mA} + \text{LPM} * 0.0545\text{mA}) * 3\text{V}/4096 * 8$$

La potencia media se calcula:

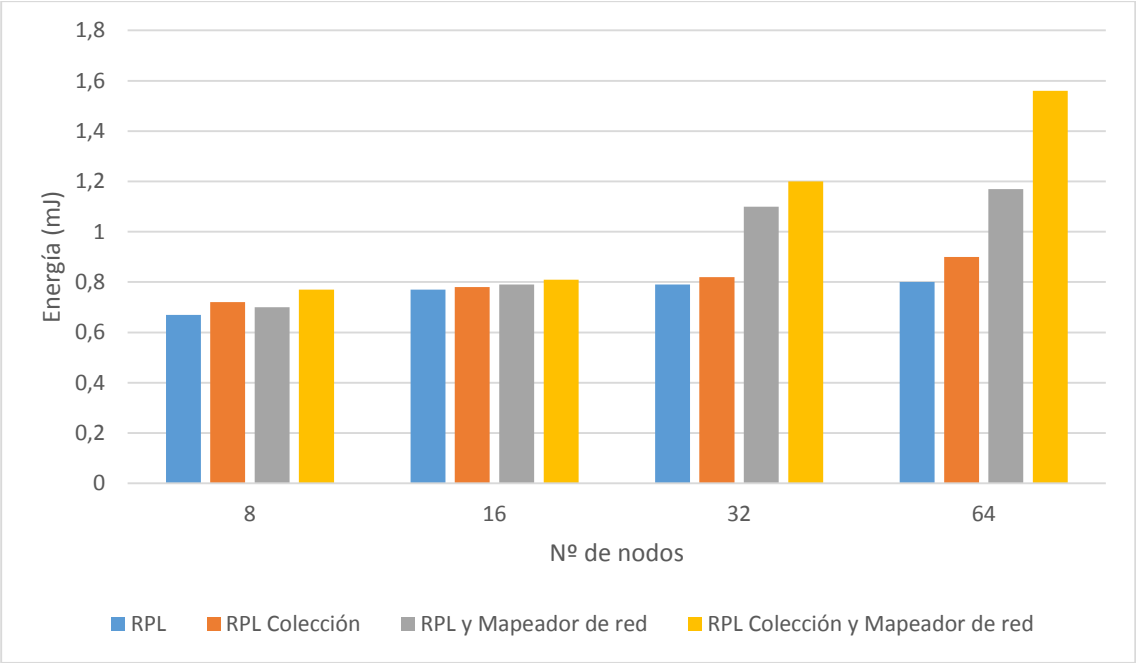
$$\text{Potencia (mW)} = \frac{Energía\ (mJ)}{Tiempo\ (seg)}$$

Al dividir la potencia entre el número de nodos de la red se obtiene la potencia consumida por cada uno de los nodos de la red. El Gráfico 13 b muestra el consumo de energía por nodo. Como puede verse en dicho Gráfico, el incremento en consumo de energía por 6Mapper es prácticamente nulo hasta los 16 nodos y se va incrementando a medida que el número de éstos aumentan, siendo éste incremento de alrededor del 30% en comparación con el uso de RPL en redes de 64 nodos. En duty cycling activo, el chip de radio del nodo se encuentra inactivo el 99% del tiempo.

En el Gráfico 13 (b) puede observarse que el consumo de energía por nodo aumenta también a medida que aumenta el número de nodos de la red. Esto se debe a que al incrementar dicho tamaño de la misma, son más las ocasiones en las que cada uno de los nodos retransmitirá paquetes de sus vecinos.



a) Energía consumida por toda la red durante 30 minutos con duty cycling



b) Energía consumida por nodo durante 30 minutos con duty cycling

Gráfico 13: Relación de consumo de energía con el número de nodos de la red, en redes con duty cycling activo (RDC)

Red sin Duty Cycling

Si se llevan a cabo las pruebas del apartado anterior en una red con RDC inactivo, vemos como el incremento de consumo de energía por el uso de RPL con 6Mapper resulta insignificante en comparación con el consumo del chip de radio quien, en este caso, se encuentra activo el 100% del tiempo.

Es necesario comentar, como ya se ha hecho en varias ocasiones en este trabajo, que las redes 6LoWPAN suelen tener activado el RDC en sus nodos, ya que la escucha continua supone un gasto de energía que este tipo de dispositivos con recursos energéticos limitados no pueden permitirse.

Incremento de energía dentro del nodo

Se miden el consumo de energía de un único evento del 6Mapper y del cortafuegos situado en el interior del nodo. La Tabla 9 muestra la energía necesaria para manejar diferentes funciones, sin incluir la energía para recibir y enviar paquetes.

Como puede observarse en la Tabla 9, el consumo de energía en el interior del nodo por el manejo de las funciones adicionales que son necesarias por el uso de SVELTE resulta insignificante, ya que las funciones principales de este sistema de protección se realizan en el 6BR (quien no tiene problema con el consumo energético).

Evento	**Energía (mJ)**
Manejo de las respuestas al 6Mapper	0.1465
Manejo del firewall	0.0478
Corrección de la pérdida de paquetes	0.0483

Tabla 9: Consumo de energía por el manejo de un único evento en un nodo

3.4.4.3 Requisitos de memoria RAM y ROM

La Tabla 10 muestra el consumo de memoria ROM que SVELTE requiere. Se puede observar que, de un total de memoria de 48k de un nodo Tmote Sky, SVELTE necesita 1.76k. El incremento de memoria ROM en el 6BR aumenta con el número de nodos, aunque teniendo en cuenta que éste se tratará probablemente de un PC, no debería resultar un problema. Por lo tanto, teniendo en cuenta el incremento de memoria producido por SVELTE, podemos llegar a la conclusión de que es viable el uso del mismo desde el punto de vista de la memoria.

Configuración	**ROM total (bytes)**	**Incremento (bytes)**
6Mapper en cliente	44264	1414
Firewall en cliente	43556	0246
Pérdida de paquetes	43264	0122
Servidor 6Mapper (1 nodo, 1 vecino)	46798	3580
Servidor 6Mapper (8 nodo, 1 vecino)	46798	3846
Servidor 6Mapper (16 nodo, 1 vecino)	46800	4152
Servidor 6Mapper (16 nodo, 8 vecino)	46924	4724

Tabla 10: Memoria ROM requerida por SVELTE

La Tabla 11 muestra el uso de memoria RAM de SVELTE dentro de un nodo Tmote Sky. La memoria RAM de estos nodos es de 10kB, de los cuales por el uso de SVELTE en los mismos se produce un incremento de sólo 0.365kB. Por lo tanto, se puede concluir que la memoria RAM no debería suponer un impedimento para la implementación de éste método de detección de intrusos en redes 6LoWPAN.

Evento	**RAM (bytes)**
Manejo de respuestas al 6Mapper	162
Manejo del firewall	24
Corrección de paquetes perdidos	188

Tabla 11: Memoria RAM requerida por SVELTE

3.4.4.4 Conclusión

Como se ha podido comprobar, el uso de memoria no debería ser un problema para el uso de SVELTE en redes 6LoWPAN. Un mecanismo para la detección de intrusos supone una pieza fundamental. Como se comenta en este trabajo en la Sección 2, existen una gran variedad de ataques a la red que suponen una amenaza para la misma. Además, a diferencia de las redes WPAN convencionales, en las redes 6LoWPAN, los nodos que la forman tienen acceso a internet de forma directa, por lo que esto abre una puerta a los atacantes que no existe para otro tipo de redes. Para paliarlo, SVELTE cuenta con cortafuegos localizado en cada uno de los nodos que, como se ha comprobado, es lo suficientemente ligero como para que no suponga ningún problema su utilización, tanto desde el punto de vista de la computación dentro del propio nodo como por el aumento del número de mensajes de la red al incluir dicha función.

4. Conclusión

La seguridad de la información es un concepto crucial. Dado el avance de las tecnologías y el uso cada vez más cotidiano de la red, los sistemas se encuentran cada vez más expuestos y, por lo tanto, pueden ser objetivos de un mayor número de ataques.

Pese a la naturaleza de pérdida de paquetes de las redes 6LoWPAN, de las limitaciones técnicas y la conectividad directa con internet de los nodos que la forman, en este trabajo se demuestra que es posible dotarlas de seguridad.

El uso de sistemas de seguridad para los mensajes se traduce en un incremento de su tamaño, ya que éstos añaden cabeceras con información relevante para el receptor. El incremento de tamaño de los mensajes repercute negativamente en la red, ya que aumenta el tráfico que circula por ella, por lo que los nodos deben reenviar un mayor número de paquetes y, por lo tanto, consumir más energía, además de producirse un incremento del tiempo de respuesta. La relación entre bytes enviados y carga útil se trata de un aspecto muy importante en todo tipo de sistemas, pero en redes 6LoWPAN lo es aún más.

Existen una gran cantidad de ataques de los que pueden ser víctimas las redes 6LoWPAN. Estos ataques pueden desencadenar una denegación de servicio, pueden cambiar información de enrutado a fin de atraer o repeler el tráfico de la red, crear bucles con el propósito de agotar las baterías de los nodos, etc. En este tipo de escenarios distribuidos, además de la necesidad de dotar a los mensajes de integridad, confidencialidad y autenticidad, surge la necesidad de evitar que se produzca intrusión en la red, por lo que se describe el funcionamiento de SVELTE y se evalúa su utilización, comentado especialmente la protección frente a ataques de expedición selectiva y agujero negro, por su frecuencia y su devastador efecto en la red. De su estudio se consagra como un mecanismo viable, detectando la mayor parte de estos ataques y con un número de falsos positivos muy bajo, que puede decrementarse cambiando un parámetro que define nuestra tolerancia en detraimiento de la cantidad de ataques detectados.

Como se comenta en este trabajo, actualmente este tipo de redes se protegen mediante el uso de la seguridad a nivel de enlace que proporciona IEEE 802.15.4, dotando a la red de seguridad entre saltos y no extremo a extremo; esto hace que las operaciones criptográficas necesarias para dotar a las comunicaciones de seguridad deban realizarse en cada uno de los nodos que forman parte del path del mensaje. Además, dicha protección incrementa la cabecera de cada una de las fragmentaciones de un paquete,

mientras que en la seguridad extremo a extremo proporcionada por IPSec se añaden cabeceras al paquete completo, sin importar el número de fragmentaciones que deban llevarse a cabo en su envío, dado que se realiza en la capa de red, superior a la capa de adaptación de 6LoWPAN. Esto supone una mejora de la eficiencia y supone un ahorro energético, ya que para la misma cantidad de carga útil es necesaria la transmisión de un menor número de bytes, reduciéndose el tráfico en la red. Además uno de los problemas de la seguridad per-hop es la falta de escalabilidad.

Aunque cierto es que la seguridad E2E parece ser una mejor solución, ambas se complementan, ya que en el caso de que se produzca un ataque en la red y se esté enviando información sin utilidad o se estén realizando modificaciones de los mensajes, en el caso de la seguridad en la capa de enlace nos percataríamos de la modificación de estos datos en el primer envío, mientras que en la seguridad E2E atravesaría toda la red y sería en el nodo destino donde nos percataríamos de que se trata de un mensaje modificado. Este tipo de ataques en los que se inunda la red con mensajes innecesarios hace que los dispositivos agoten sus baterías y baje el rendimiento.

En el caso de uso de DTLS sobre CoAP (CoAPs), se utilizan mensajes de inicialización entre las partes con el fin de llegar a un acuerdo acerca de las claves empleadas. Esto conlleva el envío de una serie de mensajes cuyo único propósito es el de mantener una comunicación segura entre emisor y receptor, sin que se envíe carga útil. Como en cada uno de los mecanismos de seguridad descritos, se realiza una compresión de los paquetes requeridos para este proceso con el fin de minimizar al máximo el tráfico en la red referente. Para ello, se utiliza el mecanismo de compresión NHC definido por IP, adaptándolo de manera que se pueda utilizar sobre UDP. Realizando una evaluación sobre la implantación de CoAPs llegamos a la conclusión de que la compresión NHC adaptada a UDP hace de DTLS para CoAP un mecanismo de seguridad factible para redes 6LoWPAN, permaneciendo el tiempo de ida y vuelta (RTT) de mensajes CoAP prácticamente inmóvil en el caso de redes con RDC activo, que serán la mayor parte de estas redes, reduciéndose de manera notable en el caso de mensajes con grandes cantidades de datos debido a la fragmentación. Se mantiene el tiempo de respuesta, se aumenta la eficiencia energética con respecto al DTLS puro y su consumo de memoria no es un problema.

El uso de métodos de compresión se traduce en un coste energético interno en el nodo; sin embargo, los análisis muestran que el incremento de coste computacional en el nodo debido a la compresión y, por lo tanto, energético, en relación con el número de fragmentaciones de los paquetes que se dejan de enviar debido a la propia compresión

es insignificante, aumentando la diferencia conforme el tamaño de los bytes de datos de los mensajes y el número de nodos aumentan.

Todos estos mecanismos de seguridad no nos confirman que la red se encuentre libre de ataques, aunque sí se lo ponen más difícil a los atacantes. Se ha comprobado que puede dotarse a las redes 6LoWPAN de seguridad, en diferentes capas y sin que el aumento de tamaño de paquete o de tráfico en la red sea un problema, ni por la memoria del dispositivo ni la energía consumida en el nodo. Pese a tratarse de dispositivos de baja potencia, su comunicación, tanto entre ellos como de estos con internet puede realizarse de forma segura.

Referencias

[1] N. Kushalnagar, G. Montenegro, C. Schumacher. IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals, 2007.

[2] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur y R. Alexander. RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks. RFC 6550.

[3] Z. Shelby, K. Hartke, y C. Bormann. Constrained Application Protocol (CoAP), Marzo 2013. http://tools.ietf.org/html/draft-ietf-core-coap-14.

[4] S. Kent and R. Atkinson. Security architecture for the internet protocol, 1998. http://www.ietf.org/rfc/rfc2401.txt.

[5] S. Kent. IP Authentication Header. RFC 4302, 2005. http://tools.ietf.org/html/rfc4302.

[6] S. Kent. IP Encapsulating Security Payload. RFC 4303, 2005. http://tools.ietf.org/html/rfc4303.

[7] S. Deering and R. Hinden. Internet Protocol, Version 6 (IPv6) Specification. RFC 1883

[8] R. Atkinson. Security Architecture for the Internet Protocol. RFC 1825

[9] E. Rescorla and N. Modadugu. Datagram Transport Layer Security Version 1.2. RFC 6347, enero 2012. http://www.ietf.org/rfc/rfc6347.txt.

[10] Trusted Platform Module (TPM) Work Group. TCG specification architecture verview (TPM 2007), 2007. http://www.trustedcomputinggroup.org

[11] Ibrahim Ethem Bagci, Mohammad Reza Pourmirza, Shahid Raza, Utz Roedig, and Thiemo Voigt. Codo: Confidential data storage for wireless sensor networks.

[12] Nicolas Tsiftes, Adam Dunkels, He Zhitao, and Thiemo Voigt. Enabling large-scale storage in sensor networks with the coffee file system.

[13] J. Vasseur and A. Dunkels. Interconnecting Smart Objects with IP – The Next Internet. Morgan Kaufmann, 2010.

[14] S. Deering and R. Hinden. Internet Protocol, Version 6 (IPv6) Specification.RFC 2460, 1998.

[15] J. Hui and P. Thubert. Compression Format for IPv6 Datagrams in 6LoWPAN Networks. draft-ietf-6lowpan-hc-13, 2010.

[16] V. Manral. Cryptographic algorithm implementation requirements for encapsulating security payload (ESP) and authentication header (ah). RFC 4835, 2007.

[17] A. Dunkels, B. Gr¨onvall, and T. Voigt. Contiki - a lightweight and flexible operating system for tiny networked sensors

[18] Shamus Software. Multiprecision Integer and Rational Arithmetic /C++Library. Web page. Visited 2010-04-17.

[19] J. Polastre, R. Szewczyk, and D. Culler. Telos: enabling ultra-low power wireless research. In IPSN'05, abril de 2005.

[20] J. Eriksson, A. Dunkels, N. Finne, F. O¨ sterlind, and T. Voigt. Mspsim –an extensible simulator for msp430-equipped sensor boards

[21] R. Hummen, J. Hiller, H. Wirtz, M. Henze, H. Shafagh, and K. Wehrle. 6LoWPAN Fragmentation Attacks and Mitigation Mechanisms

[22] LCIS and Aragosystems. WiSMote Sensor Node. Web page: http://wismote.org/

[23] S. Raza, S. Duquennoy, T. Chung, D. Yazar, T. Voigt. Securing Communication in 6LoWPAN with Compressed IPsec

[24] S. Raza, S. Duquennoy, J. Hoglund, U. Roedig, T. Voigt. Secure Communication for the Internet of Things - A Comparison of Link-Layer Security and IPsec for 6LoWPAN

[25] S. Raza, H. Shafagh, K. Hewage, R. Hummen, T. Voigt. Lightweight Secure CoAP for the Internet of Things

[26] S. Raza, L. Wallgren, T. Voigt. SVELTE: Real-time Intrusion Detection in the Internet of Things