Universitat Oberta de Catalunya

uoc.edu

Seguridad en la Internet de las cosas:
Propuesta de implantación segura de un sistema de seguridad con dispositivos IoT en una PYME

Autor: Adrián Segura Gavilán

Máster Universitario en Seguridad de las Tecnologías de la Información y de las Comunicaciones

Seguridad en la Internet de las Cosas

Consultor: Amadeu Albós Raya
PRA: Helena Rifá Pous

Junio 2019

FICHA DEL TRABAJO FINAL

Título del trabajo:	*Propuesta de implantación segura de un sistema de seguridad en una PYME*
Nombre del autor:	*Adrián Segura Gavilán*
Nombre del consultor/a:	*Amadeu Albós Raya*
Nombre del PRA:	*Helena Rifá Pous*
Fecha de entrega (mm/aaaa):	06/2019
Titulación:	*Máster Universitario en Seguridad de las Tecnologías de la Información y de las Comunicaciones*
Área del Trabajo Final:	*Seguridad en Internet de las Cosas*
Idioma del trabajo:	*Español*
Palabras clave	*IoT, ecosistemas seguros, ciberseguridad*

Resumen del Trabajo (máximo 250 palabras): *Con la finalidad, contexto de aplicación, metodología, resultados y conclusiones del trabajo.*

El Internet de las Cosas (IoT) ha supuesto una revolución que nos ha brindado la posibilidad de conectar todo tipo de objetos y aparatos del hogar (o la empresa) a Internet. Mediante el uso de nuestras propias manos, o incluso la voz, tenemos la posibilidad de dar órdenes, comprobar y tener una estancia ultra conectada. Cada uno de estos dispositivos IoT, además de estar conectado a la red de redes, cuenta con su propio software y hardware, que en muchos casos ha sido desarrollado de una manera demasiado ágil, o incluso que en unos pocos años deja de recibir soporte, dando lugar a la posible exposición de nuestra privacidad, hogar, familias o información valiosa a delincuentes que sin acceso físico a nuestra estancia puede hacer lo que quieran con nosotros.

En este TFM se pretende llevar a cabo un análisis tanto de las posibles vulnerabilidades existentes para los dispositivos IoT, así como el análisis del proceso de securización de las comunicaciones de un sistema de seguridad en una organización (con cierre de puerta, cámaras, alarmas...). Para ello, se llevará a cabo en primer lugar un análisis de las diferentes vulnerabilidades que puede tener un sistema IoT, seguido de un análisis de los diferentes dispositivos que formarán el sistema de seguridad que se desea implantar y finalizando con las medidas a implementar necesarias para securizar las comunicaciones de estos dispositivos.

Abstract (in English, 250 words or less):

The Internet of Things (IoT) has brought about a revolution that has given us the possibility of connecting all kinds of household objects (or the company) to the Internet. Through the use of our own hands, or even the voice, we have the possibility to give orders, check and have a home connected to Internet. Each of these IoT devices, in addition to being connected to the network of networks, has its own software and hardware, which in many cases has been developed in a too agile way, or even that in a few years stops receiving support, giving rise to the possible exposure of our privacy, home, families or valuable information to criminals who without physical access to our home can do whatever they want with us.

In this TFM, we intend to carry out an analysis of the possible vulnerabilities existing for the IoT devices, as well as the analysis of the securization process of the communications of a security system in an organization (with closing of doors, cameras, alarms...). For this purpose, an analysis of the different vulnerabilities that an IoT system can have will be carried out, followed by an analysis of the different devices that will form the security system that is to be implemented and ending with the necessary measures to be implemented. secure the communications of these devices.

Índice

1. INTRODUCCIÓN 1

1.1 CONTEXTO Y JUSTIFICACIÓN DEL TRABAJO 1
1.2 OBJETIVOS DEL TRABAJO 1
1.3 ENFOQUE Y MÉTODO SEGUIDO 2
1.4 PLANIFICACIÓN DEL TRABAJO 3
1.5 BREVE SUMARIO DE PRODUCTOS OBTENIDOS 5
1.6 BREVE DESCRIPCIÓN DE LOS OTROS CAPÍTULOS DE LA MEMORIA 5

2. ESTADO DEL ARTE 7

2.1. DEFINICIÓN DE IOT 7
2.2 SECTORES DONDE SE PUEDE ENCONTRAR IOT 7
2.3. TENDENCIAS Y ESTADO DE LOS ECOSISTEMAS IOT EN LA ACTUALIDAD 9
2.5. EL PELIGRO DE LAS BOTNETS Y REPERCUSIONES DE SUS ATAQUES EN EL IOT 10
2.6. PLATAFORMAS HARDWARE IOT 11
2.6.1. Arduino 11
2.6.2. Waspmote 13
2.6.3. Intel Galileo 14
2.6.4. Raspberry Pi 16
2.7. PLATAFORMAS SOFTWARE IOT 17
2.7.1. ThingSpeak 17
2.7.2. Electric Imp 20
2.7.3. Amazon Web Services IoT 22
2.7.4. Google Cloud IoT 25

3. ANÁLISIS DE LAS AMENAZAS Y VULNERABILIDADES MÁS COMUNES EN IOT 27

3.1 ANÁLISIS DE AMENAZAS EN IOT 27
3.1.1 Ataques DDoS 27
3.1.2 Espionaje y vigilancia 27
3.1.3 Ransomware 28
3.1.4 Movimientos laterales 28
3.2 ANÁLISIS DE VULNERABILIDADES EN IOT 28
3.2.1 Interfaces web no seguras 29
3.2.2 Autenticación/autorización insuficiente 29
3.2.3 Servicios de red inseguros 30
3.2.4 Ausencia de cifrado en las comunicaciones 31
3.2.5 Interfaz en la nube insegura 32
3.2.6 Interfaz móvil insegura 32
3.2.7 Configurabilidad de seguridad insuficiente 33
3.2.8 Software/Firmware inseguro 34
3.2.9 Seguridad física insuficiente 35

4. ANÁLISIS DE LA SEGURIDAD Y LAS AMENAZAS DE LOS ROUTERS 36

4.1. VULNERABILIDAD FILET-O-FIREWALL 36
4.2. ATAQUE DNS CHANGER 38

5. COMUNICACIONES SEGURAS EN ECOSISTEMAS IOT 41

5.1. DTLS 41
5.2. ECC 41
5.3. HMAC 42
5.4. BLOCKCHAIN 42

6. PROPUESTAS DE MEDIDAS DE SEGURIDAD Y SU IMPLANTACIÓN PARA UN ECOSISTEMA IOT..... 43

6.1. PROPUESTAS PARA SECURIZAR LOS ACCESOS Y LOS DATOS TRANSMITIDOS 43

6.1.1. Políticas para los accesos 44

6.1.2. Autenticación de doble factor..... 45

6.1.3. Infraestructura de clave pública 46

6.2. PROPUESTA DE SECURIZACIÓN DE LAS COMUNICACIONES EN EL ECOSISTEMA..... 48

6.2.1. Uso de una VPN para securizar los accesos externos 48

7. CONCLUSIONES Y TRABAJO FUTURO..... 52

8. GLOSARIO..... 54

9. BIBLIOGRAFÍA..... 55

10. ANEXOS 60

Lista de figuras

Figura 1 Diagrama de Gantt 4
Figura 2 Principales componentes de Waspmote - Parte frontal 13
Figura 3 Principales componentes de Waspmote - Parte trasera 14
Figura 4 Placa Intel Galileo, segunda generación 15
Figura 5 Raspberry Pi 3B 16
Figura 6 Raspberry Pi Zero W 17
Figura 7 Plataforma ThingSpeak 18
Figura 8 Ecosistema de la plataforma Electric Imp 21
Figura 9 Servicios de AWS IoT 24
Figura 10 Funcionamiento normal de router actuando como firewall 37
Figura 11 Explotación de la vulnerabilidad Filet-O-Firewall 38
Figura 12 Funcionamiento de la criptografía de clave pública 46
Figura 13 Ejemplo de una conexión VPN a través de un móvil utilizado como dispositivo de control 49
Figura 14 Paquete encapsulado con PPPTP 50
Figura 15 Paquete encapsulado en la primera capa de L2TP 51
Figura 16 Paquete completamente encapsulado de L2TP 51
Figura 17 Diagrama de cifrado asimétrico SSL/TLS en OpenVPN 60

1. Introducción

1.1 Contexto y justificación del Trabajo

En el presente Trabajo de Fin de Máster (TFM), pretende ser una investigación sobre los peligros existentes en el uso del Internet de las Cosas (IoT) y sobre los métodos de securización aplicables a un entorno IoT, presentando en el caso de este proyecto un sistema de seguridad (alarma, puerta con cierre, cámaras de seguridad…) cuya implementación pueda ser realizada en una PYME o un hogar.

Hoy en día vivimos en una sociedad de la inmediatez en la que mucha tecnología tiene que ser desarrollada con mucha agilidad para que el producto final consiga llegar a tiempo al mercado. Un hecho importante es que, si bien parte de estos productos es hardware, el software que se encuentra embebido en estos dispositivos generalmente está realizado a medida para el dispositivo. Es en este punto donde se generan los problemas a la hora de tener un soporte de actualizaciones que acompañe al dispositivo durante su vida útil y a la hora de que el software desarrollado sea seguro.

En este concepto de sociedad encontramos que los usuarios demandantes de tecnología quieren poder hacer uso de ella de la forma más inmediata y sencilla nada más adquirir un producto. Esto conlleva que, especialmente en el Internet de las Cosas, no se preste atención a diversos aspectos en la configuración de los productos que pueden conllevar una exposición de nuestra red, nuestra privacidad o incluso la integridad de nuestra información.

Con este TFM se pretende presentar una instalación propia de un sistema de seguridad mediante la implantación de un ecosistema IoT junto con su proceso de securización de las comunicaciones entre a nivel de dispositivo y de comunicación entre ellos así como concienciar y verificar que los dispositivos IoT de hoy en día tienen serios problemas para ser seguros, ya sea por vulnerabilidades presentes en su firmware/software, por menús de configuración que han sido mal diseñados o simplemente por la tendencia de los usuarios a configurar este tipo de dispositivos de la forma más rápida y cómoda a la vez que insegura.

1.2 Objetivos del Trabajo

Los objetivos que se buscan cumplir mediante la realización de este TFM son:

- Analizar los riesgos de la implantación de un ecosistema IoT en una PYME o en el hogar: se analizarán los riesgos más típicos desde la perspectiva de la configuración realizada por el usuario medio y las implicaciones que pueden surgir si alguno de los dispositivos implantados se ve comprometido. Como se verá en los apartados siguientes de esta memoria, los ecosistemas IoT son un objetivo interesante para los cibercriminales.

- Análisis de los routers: los routers suponen el medio de conexión de todo el ecosistema IoT implantado. Si un dispositivo IoT cuenta con una vulnerabilidad y además nuestro router cuenta con una configuración errónea, se nos presenta un entorno en el que se potencia el riesgo de nuestros dispositivos IoT. Otro caso que se puede dar es que, aunque nuestro router se encuentre bien configurando, el mismo presente vulnerabilidades en su software, llevándonos también a un entorno de alto riesgo.

- Mejorar las comunicaciones entre los dispositivos del ecosistema IoT implantado: algunas tecnologías que se pueden utilizar para implementar un mecanismo de seguridad adecuado para la protección de las comunicaciones y los datos de los dispositivos de un ecosistema IoT.

- Propuesta de implantación del ecosistema de seguridad IoT de forma segura basándose en lo analizado en los apartados anteriores.

1.3 Enfoque y método seguido

El enfoque seguido para el desarrollo de este TFM es el de una PYME que desea implantar en sus instalaciones un sistema de seguridad contra intrusos de la forma más segura posible a nivel de configuración tanto de los propios dispositivos del sistema como a nivel de red de la empresa. Este sistema de seguridad contará con un sistema de cierre de puerta, alarma ante intrusos y cámaras de seguridad. Podríamos considerar este caso también para un hogar si tenemos en cuenta una PYME con un número de personal reducido, ya que por su tamaño de red son similares.

En cuanto al método seguido para la realización de este proyecto, el proyecto se puede dividir, grosso modo, en tres fases:

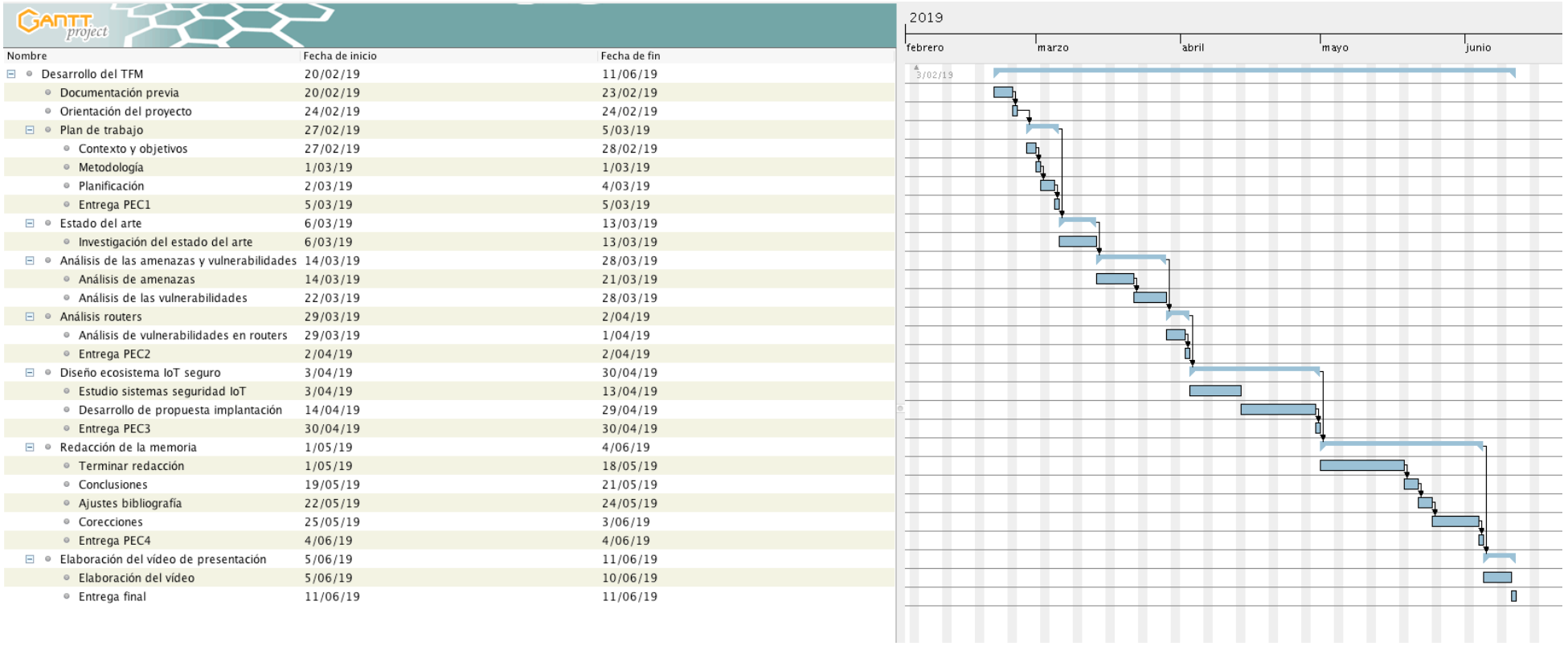
- Una primera fase en la que se llevará a cabo un proceso de investigación y análisis teórico de las vulnerabilidades posibles en un sistema IoT y de los routers, coincidiendo con los dos primeros objetivos descritos en el apartado anterior.

- En una segunda fase se analizarán algunas tecnologías que puedan proteger las comunicaciones de los dispositivos IoT.

- La fase final del proyecto concuerda con el último objetivo, la propuesta de medidas de seguridad del sistema de seguridad IoT y su implantación.

1.4 Planificación del Trabajo

En la planificación temporal de este proyecto se definen 5 fechas destacables que coinciden con las fechas en las que este TFM será revisado por el profesor responsable y su presentación final. Para la visualización de la planificación estipulada, se ha creado el siguiente diagrama de Gantt:



Nombre	Fecha de inicio	Fecha de fin
Desarrollo del TFM	20/02/19	11/06/19
Documentación previa	20/02/19	23/02/19
Orientación del proyecto	24/02/19	24/02/19
Plan de trabajo	27/02/19	5/03/19
Contexto y objetivos	27/02/19	28/02/19
Metodología	1/03/19	1/03/19
Planificación	2/03/19	4/03/19
Entrega PEC1	5/03/19	5/03/19
Estado del arte	6/03/19	13/03/19
Investigación del estado del arte	6/03/19	13/03/19
Análisis de las amenazas y vulnerabilidades	14/03/19	28/03/19
Análisis de amenazas	14/03/19	21/03/19
Análisis de las vulnerabilidades	22/03/19	28/03/19
Análisis routers	29/03/19	2/04/19
Análisis de vulnerabilidades en routers	29/03/19	1/04/19
Entrega PEC2	2/04/19	2/04/19
Diseño ecosistema IoT seguro	3/04/19	30/04/19
Estudio sistemas seguridad IoT	3/04/19	13/04/19
Desarrollo de propuesta implantación	14/04/19	29/04/19
Entrega PEC3	30/04/19	30/04/19
Redacción de la memoria	1/05/19	4/06/19
Terminar redacción	1/05/19	18/05/19
Conclusiones	19/05/19	21/05/19
Ajustes bibliografía	22/05/19	24/05/19
Corecciones	25/05/19	3/06/19
Entrega PEC4	4/06/19	4/06/19
Elaboración del vídeo de presentación	5/06/19	11/06/19
Elaboración del vídeo	5/06/19	10/06/19
Entrega final	11/06/19	11/06/19

Figura 1 Diagrama de Gantt

1.5 Breve sumario de productos obtenidos

Como sumario de productos de este proyecto podemos centrarnos en los que se van a obtener en cada una de las entregas previstas en el plan docente:

- PEC1: Plan de trabajo con el desarrollo del tema y lo objetivos del proyecto y en qué secciones y subsecciones se va a dividir.

- PEC2: Esta segunda entrega se centra en el desarrollo de los conceptos teóricos del proyecto, entre los cuales podemos incluir la investigación de las amenazas y vulnerabilidades más comunes existentes para dispositivos IoT

- PEC3: En esta entrega se va a investigar sobre los routers y algunas de sus vulnerabilidades y sobre algunas tecnologías que permitan securizar las comunicaciones y los datos de los dispositivos IoT.

- PEC4: En esta entrega se va a terminar la redacción de la memoria del proyecto mediante una propuesta de implantación de un ecosistema IoT junto las conclusiones del proyecto y que contendrá todos los puntos entregados anteriormente con el formato requerido para su entrega.

- PEC5: En esta última entrega se llevará a cabo un vídeo en el que se defiende el material elaborado durante el proyecto para que sea visualizado y evaluado por parte del tribunal.

El resultado final de todas las entregas descritas dará como producto una memoria final en la que estará contenido todo el trabajo desarrollado a lo largo de la elaboración del proyecto. Esta memoria contendrá, por tanto, los análisis llevados a cabo en la realización del proyecto, las conclusiones a las que se ha llegado tras su realización y las posibles líneas futuras del proyecto.

1.6 Breve descripción de los otros capítulos de la memoria

Los capítulos contenidos en este TFM, a grandes rasgos, serán:

1. **Introducción:** En este capítulo se introduce el contexto, objetivos, justificación y planificación temporal del proyecto a desarrollar.

2. **Estado del arte:** En este capítulo se llevará a cabo una revisión del estado actual del IoT, los sectores en los que se utiliza, previsiones de futuro sobre su uso y crecimiento y sobre las botnets, que son las principales depredadoras de sistemas IoT.

3. **Análisis de las amenazas y vulnerabilidades más comunes en IoT:** El fin de este capítulo es llevar a cabo una revisión de las amenazas y vulnerabilidades más comunes que se dan en IoT.

4. **Análisis de los routers:** Un dispositivo IoT puede verse vulnerado también por problemas en el dispositivo desde el que es gestionado. En este capítulo se verán algunas vulnerabilidades conocidas para los routers.

5. **Comunicaciones seguras en ecosistemas IoT:** En este capítulo se llevará a cabo una investigación de algunas tecnologías que permiten securizar las comunicaciones y los datos de los dispositivos IoT.

6. **Propuesta de implantación ecosistema de seguridad IoT:** El fin de este capítulo es el de llevar a cabo una propuesta de implantación de un sistema de seguridad en una PYME con la implementación de un ecosistema IoT securizado.

7. **Conclusiones:** En este apartado se plasmarán las conclusiones obtenidas tras la realización del TFM, así como las posibles líneas futuras.

8. **Glosario, Bibliografía y anexos:** En este capítulo final se incluirán las referencias bibliográficas, términos y anexos que complementan los conceptos desarrollados en este TFM.

2. Estado del arte

2.1. Definición de IoT

IoT es un término ampliamente utilizado para un conjunto de tecnologías, sistemas y principios de diseño asociados con la ola emergente de cosas conectadas a Internet. En muchos aspectos, puede parecer lo mismo que la comunicación M2M (machine to machine, máquina a máquina en español) ya que se conectan sensores y otros dispositivos a sistemas de tecnología de la información y la comunicación (ICT) a través de redes cableadas o inalámbricas. Sin embargo, en contraste con M2M, el concepto de IoT también se refiere a la conexión de dichos sistemas y sensores a Internet, así como al uso de tecnologías generales de Internet [1].

Por otro lado, se define un ecosistema IoT como un sistema intercomunicado de dispositivos IoT que capturan información con sus sensores para que ésta sea consumida por otros objetos, entando comunicados entre ellos. La comunicación entre los dispositivos del ecosistema IoT no requiere de la intervención humana, si no que se trata de una comunicación M2M.

Una vez definido lo que es el IoT, se van a analizar algunos sectores en los que hoy en día se hace uso de dispositivos IoT.

2.2 Sectores donde se puede encontrar IoT

En la actualidad hay un flujo contaste de noticias sobre ecosistemas IoT, pero es importante saber en qué sectores tienen presencia las soluciones que se presentan en ellas. Por tanto, a continuación, se enumerarán algunos sectores en los que el IoT está cada vez más a la orden del día.

- **Agricultura y ganadería**

Gracias al uso de IoT en la agricultura se consigue una mejor optimización de los procesos y los tiempos mediante la recolección de numerosos datos tales como la composición del suelo, condiciones climáticas y medioambientales, etc… De esta manera se puede tener un mayor control sobre las cosechas y aumentar la productividad y calidad de los productos obtenidos. Un ejemplo práctico del uso de IoT en la agricultura es el de la implementación de un sistema de riego automatizado [2].

En este entorno una brecha de seguridad en el ecosistema IoT podría provocar un malfuncionamiento de los diferentes procesos de los que cada uno de los dispositivos IoT integrados se encarga, pudiendo conllevar con ello pérdidas para la empresa afectada y que los clientes finales se viesen también perjudicados.

- **Marketing**

Algunos usos del IoT en el marketing son:

- Analizar el hábito de compra del cliente en las plataformas que los clientes utilizan. Una brecha de seguridad

- Obtener una mejor comprensión del proceso de compra y en qué etapa de este se encuentra el cliente.

- Realizar interacciones en tiempo real, notificaciones de punto de venta y, por supuesto, anuncios dirigidos (e incluso completamente contextuales) [3].

Si sucediese un problema de seguridad en este entorno, los datos personales de los clientes finales, así como datos altamente sensibles como tarjetas de crédito, direcciones, etc.. podrían verse vulnerados, conllevando con ello graves consecuencias como suplantaciones de identidad, robos de tarjetas de crédito digitales o malware en forma de anuncios falsos.

- **Smart cities**

Un ejemplo del uso del IoT a nivel urbano es el de un servicio de gestión de espacios de estacionamiento de vehículos que permita monitorear la ocupación de los espacios de estacionamiento para vehículos que se encuentren al aire libre, pudiendo esto derivar en un sistema que guie al usuario hacia zonas en las que pueda estacionar su vehículo [4].

El posible fallo de seguridad de una iniciativa de Smart City podría tener un impacto mucho mayor y consecuencias muy graves. Tomemos por ejemplo las luces y las comunicaciones. Muchas ciudades inteligentes hacen uso de las farolas como la columna vertebral de las redes de área de campo (FAN) de toda la ciudad. Los dispositivos, las puertas de enlace y las redes no seguras son un terreno fértil para los piratas informáticos interesados en causar interrupciones en toda la ciudad y un posible control del sistema. Una vulneración de este dispositivos podría conllevar bloques enteros de luces de la calle apagándose, lo que podría causar estragos en el tráfico, comprometer la seguridad del vecindario e interrumpir las comunicaciones móviles.

- **Industria**

La industria es un sector en que el IoT tiene multitud de usos, como el seguimiento de activos dentro de las fábricas para analizar y mejorar los diferentes procesos empresariales además de mejorar el mantenimiento de los diferentes elementos que intervienen.

Tal y como ocurre en el sector de la agricultuta, si el ecosistema IoT encargado de un proceso industrial se ve vulnerado provocando con ello su malfuncionamiento, la marca y los clientes finales pueden verse altamente perjudicados.

Como se puede ver, hay diversos sectores en los que la implantación de ecosistemas IoT es una realidad, siendo los sectores presentados sólo algunos de todos en los que estas implantaciones son llevadas a cabo. Las implementaciones de los ecosistemas IoT en cada sector disponen de un gran abanico de posibilidades, y con el tiempo las investigaciones e inversiones en IoT van en aumento, tal y como se puede ver en las tendencias que se exponen a continuación.

2.3. Tendencias y estado de los ecosistemas IoT en la actualidad

Con el objetivo de tener un contexto sobre el estado de los ecosistemas IoT, se van a analizar unas cifras con las que se puede ver la tendencia y el estado actual de los ecosistemas IoT así como las cifras que se estiman cara a futuro.

Según IoT Analytics [5], se calculó que en 2018 el número de dispositivos conectados que están en uso en todo el mundo superaba los 17 mil millones, con un número de dispositivos IoT de 7 mil millones (este número no incluye teléfonos inteligentes, tablets, computadoras portátiles o teléfonos de línea fija). Según este análisis, el crecimiento de la conexión global se debe principalmente a los dispositivos IoT, tanto en el lado del consumidor (por ejemplo, las Smart Home) como en el lado de la empresa (por ejemplo, la maquinaria conectada). Se espera que la cantidad de dispositivos IoT que están activos aumente a 10 mil millones para 2020 y 22 mil millones para 2025. Esta cantidad de dispositivos IoT incluye todas las conexiones activas y no tiene en cuenta los dispositivos que se compraron en el pasado pero que ya no se usan.

Esta tendencia de crecimiento en masa se ve respaldada a nivel económico, ya que según el mismo informe se prevé un crecimiento en la inversión desde la cifra de $151B que se invirtió en 2018 a $1,567B en 2025.

Estas cifras demuestran que las empresas han analizado los beneficios que derivan del uso de los ecosistemas IoT en sus modelos de negocio y que están acelerando a un gran ritmo la inversión en éstos. Sin embargo, este enorme crecimiento a pasos tan acelerados nos lleva a plantear la cuestión de si los dispositivos desarrollados son seguros.

2.4. ¿Son seguros los ecosistemas IoT?

Tal y como se ha visto en el apartado anterior, la tecnología IoT tiene que ser desarrollada con mucha agilidad debido a su gran demanda. Un hecho importante es que, si bien parte de estos productos es hardware, el software que se encuentra embebido en estos dispositivos generalmente está realizado a medida para el dispositivo. Es en este punto donde se generan los problemas a la hora de tener un soporte de actualizaciones que acompañe al dispositivo durante su vida útil y a la hora de que el software desarrollado sea seguro. Esto conlleva que, especialmente en el Internet de las Cosas, no se preste atención a diversos aspectos en la configuración de los productos que pueden conllevar una exposición de nuestra red, nuestra privacidad o incluso la integridad de nuestra información.

Como prueba de que el IoT puede no ser seguro, se va a presentar un caso real de un ataque realizado a un proveedor de Internet norteamericano que conllevó a la paralización de sus sistemas DNS y que fue realizado por una Botnet, uno de los mayores depredadores de los dispositivos IoT con vulnerabilidades. Se definirá a continuación, por tanto, qué es una Botnet y el peligro que representan.

2.5. El peligro de las Botnets y repercusiones de sus ataques en el IoT

El objetivo de este apartado es mostrar como una Botnet puede vulnerar los dispositivos IoT para fines maliciosos. Para ello, en primer lugar, se describirá qué es una Botnet y cuáles son los métodos más comunes mediante los que un atacante infecta un dispositivo, y a continuación se verá un caso real de un ataque producido por una Botnet a un proveedor de Internet Norteamericano a través de dispositivos IoT infectados.

Una Botnet [6] es el nombre genérico utilizado para la denominación de un grupo de dispositivos (no sólo PCs, también dispositivos IoT) que han sido infectados por un malware y son controlados por el atacante de forma remota. A los dispositivos pertenecientes a una botnet se les denomina como “bots” o “zombies”.

En el caso de los dispositivos IoT, es mucho más sencillo que éstos caigan presa de una botnet, ya que los fabricantes dejan de dar soporte a su propio software, en ocasiones el mal diseño del firmware del dispositivo no obliga al usuario a cambiar el usuario y contraseña por defecto, tienen puertos o servicios vulnerables abiertos, etc…

Dos métodos que los cibercriminales utilizan para infectar PCs tradicionales son [6]:

- Ataques drive-by downloads: en este ataque el cibercriminal busca una web en la que pueda explotar una vulnerabilidad para cargar su código malicioso. Este código redirigirá al usuario a otra web controlada por el ciberdelincuente en la que el código del bot es descargado e instalado en el dispositivo.

- Email: en este caso el atacante envía una gran cantidad de spam al usuario en el que adjunta un enlace fraudulento a una página en la que aloja el código que infectará el dispositivo o bien adjunta un archivo Word o PDF que contiene el código malicioso. Una vez el usuario abre cualquiera de estos adjuntos, el código malicioso es descargado e instalado en su equipo, cayendo presa de la botnet.

Para el control de las máquinas infectadas, generalmente los cibercriminales hacen uso de canales IRC o servidores de comando y control a los que todos los dispositivos zombies de la botnet se conectan para buscar nuevas órdenes. Este tipo de jerarquía es una centralizada en la que si los servidores principales de la botnet son desmontados los dispositivos zombies de ésta no podrán

recibir nuevas órdenes. Sin embargo, muchas botnets han evolucionado al uso de las redes P2P (peer-to-peer, o red entre pares en Español), que permiten una infraestructura descentralizada, permitiendo con ello que entre los miembros de la botnet se conecten y comuniquen órdenes mediante conexiones cifradas. Este tipo de infraestructura es descentralizada, haciendo que sean más difíciles de desarticular ya que no existe un solo servidor o grupo de servidores desde los que los zombies reciben las órdenes [12].

Un ejemplo de la gran amenaza que suponen las botnets es el sufrido por el proveedor de Internet Dyn, en el año 2016, en su sistema de dominio de Internet (DNS), que afectó principalmente a la costa Este de Estados Unidos, con una cantidad de trafico registrada de aproximadamente 1,2 Terabytes por segundo [7]. Este ataque permitió a la botnet Mirai [8] anular el servicio de más de 50 webs, siendo algunas de estas webs pertenecientes a grandes empresas de renombre como Amazon, Spotify, Twitter o PayPal.

2.6. Plataformas hardware IoT

Tras el análisis de las repercusiones que puede tener un dispositivo IoT mal securizado y vulnerado, se va a llevar a cabo un análisis de algunos de los dispositivos hardware más comunes a la hora del desarrollo de un proyecto IoT, así como de su finalidad y su uso para cada uno de ellos.

2.6.1. Arduino

Arduino es una plataforma electrónica de código abierto basada en hardware y software de uso sencillo. Una placa Arduino tiene la capacidad de leer entradas (luz en un sensor, un dedo en un botón, etc…) y convertirlas en salidas, como por ejemplo activar un motor o encender un LED. Para la programación del comportamiento de la placa, se pueden cargar un conjunto de instrucciones en el microcontrolador de la tarjeta a través del lenguaje de programación Arduino (basado en Wiring) y el software Arduino (IDE).

Arduino nació en el Instituto de Diseño de Interacción Ivrea como una herramienta de uso sencillo para la creación rápida de prototipos, dirigida a estudiantes sin experiencia en electrónica y programación. Tan pronto como llegó a una comunidad más amplia, la placa Arduino comenzó a cambiar para adaptarse a las nuevas necesidades y desafíos, diferenciando su oferta de tablas simples de 8 bits a productos para aplicaciones IoT, impresión portátil, impresión 3D y entornos integrados. Todas las placas Arduino son completamente de código abierto, lo que permite a los usuarios construirlas de forma independiente y, eventualmente, adaptarlas a sus necesidades particulares. El software también es de código abierto y está creciendo a través de las contribuciones de los usuarios de todo el mundo [11].

Los puntos a favor que pueden determinar el uso de una placa Arduino para un proyecto son los siguientes:

- Precio: las placas Arduino son relativamente económicas en comparación con otras plataformas de microcontroladores. La versión más económica del módulo Arduino se puede ensamblar a mano, e incluso los módulos Arduino pre-montados cuestan menos de $50.

- Es multiplataforma: el software Arduino (IDE) se ejecuta en los sistemas operativos Windows, Macintosh OSX y Linux. Esto supone una ventaja ya que la mayoría de los sistemas de microcontroladores están limitados a Windows.

- El entorno de programación es simple y claro: el software Arduino (IDE) es fácil de usar para los principiantes, pero es lo suficientemente flexible como para que los usuarios avanzados también lo aprovechen.

- El software es de código abierto y extensible: el software Arduino es una herramienta de código abierto, por lo que está disponible para la extensión por parte de programadores experimentados. El lenguaje se puede expandir a través de las bibliotecas de C++, y las personas que deseen comprender los detalles técnicos pueden dar el salto de Arduino al lenguaje de programación AVR C en el que se basa.

- Fuente abierta y hardware extensible: los planos de las placas Arduino se publican bajo una licencia de Creative Commons, por lo que los diseñadores de circuitos experimentados pueden crear su propia versión del módulo, ampliarlo y mejorarlo.

A la hora de adquirir una placa Arduino, en el mercado se puede encontrar una gran oferta de modelos con diferentes características. A continuación, se muestra un cuadro comparativo con las placas Arduino oficiales más populares del mercado [12]:

Características	**UNO**	**Mega**	**Leonardo**	**DUE**
Tipo de microprocesador	Atmega 328	Atmega 2560	Atmega 32U4	AT91SAM3X8E
Velocidad de reloj	16 MHz	16 MHz	16 MHz	84 MHz
Pines digitales de E/S	14	54	20	54
Entradas analógicas	6	16	12	12
Memoria de programa (Flash)	32 Kb	256 Kb	32 Kb	512 Kb
Memoria de datos (SRAM)	2 Kb	8 Kb	2.5 Kb	96 Kb
Memoria auxiliar (EEPROM)	1 Kb	4 Kb	1 Kb	0 Kb

Conclusiones:

Arduino es una buena elección para todo usuario que desee iniciarse en el mundo de los microcontroladores, ya que su bajo coste, su amplia comunidad y su plataforma sencilla la hacen ser una gran opción. Además, las placas Arduino resultan ideales también para realizar prototipos en el mundo del IoT ya que gran parte de las plataformas IoT son compatibles con estas placas hardware.

2.6.2. Waspmote

Waspmote es una plataforma open source creada y fabricada por la empresa Libelium. La finalidad de esta placa es principalmente la de construir redes inalámbricas de bajo consumo.

Esta placa se basa en una arquitectura modular. La idea es integrar solo los módulos necesarios en cada dispositivo, permitiendo que estos módulos se puedan cambiar y ampliar según las necesidades. Las especificaciones técnicas de la placa son las siguientes:

Características	
Microcontrolador	ATmega1281
Frecuencia	14.7456 MHz
SRAM	8 kb
EEPROM	4 kb
FLASH	128 kb
Capacidad máxima de tarjeta SD	16 GB
Peso	20 g
Dimensiones	73.5 x 51 x 13 mm

A continuación, se muestran unas capturas de pantalla de los principales componentes de la Waspmote [13]:

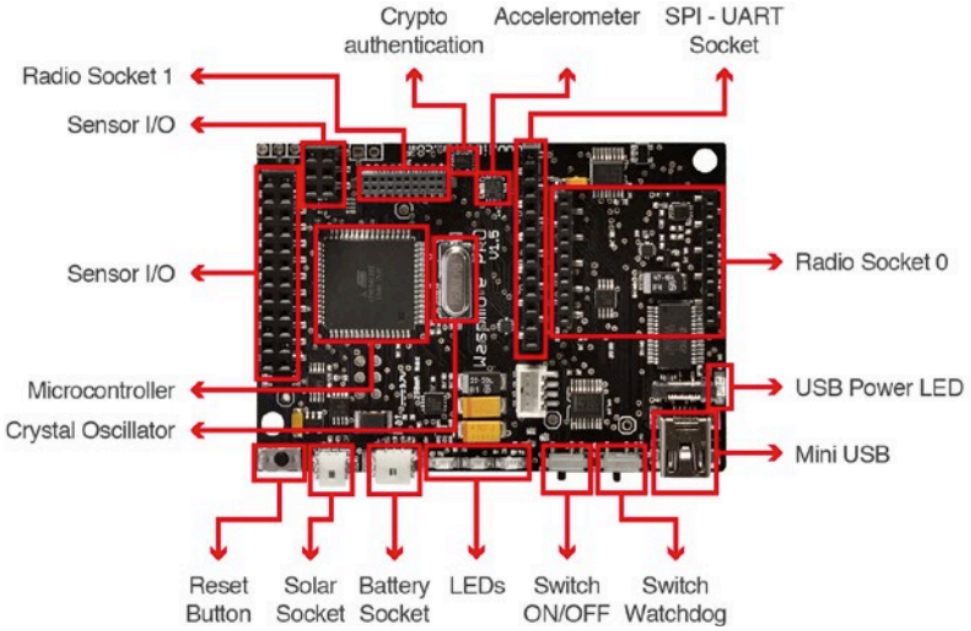


Figura 2 Principales componentes de Waspmote - Parte frontal

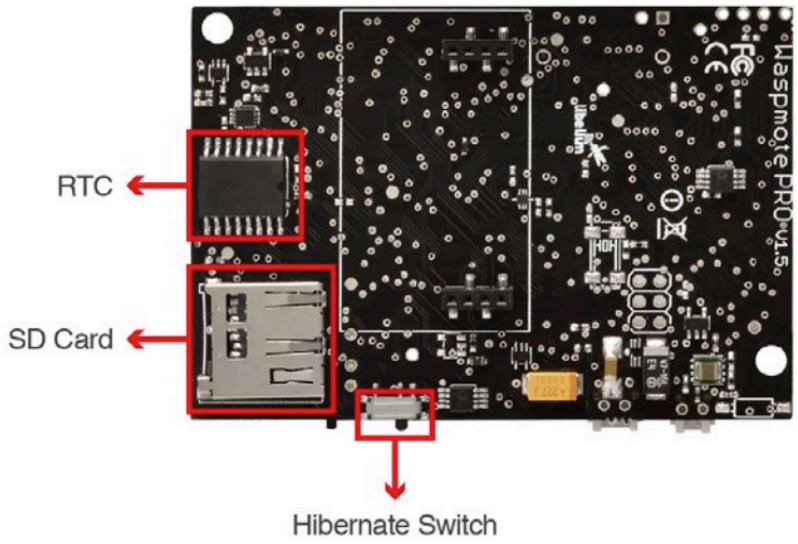


Figura 3 Principales componentes de Waspmote - Parte trasera

A diferencia de la plataforma Arduino, que está más orientada al desarrollo de proyectos educativos o prototipos, Waspmote es una plataforma orientada específicamente a la creación de Redes Sensoriales Inalámbricas que necesitan de una larga duración y cuyo fin es ser desplegadas en entornos reales.

Los ámbitos de aplicación de estas placas son en proyectos de Smart City. Un ejemplo de su uso en entornos reales es en un proyecto llevado a cabo en Castellón a través de una plataforma para el control del uso del agua y la gestión de los residuos en la ciudad [14].

Conclusiones:

Waspmote es una plataforma orientada al desarrollo e implementación de proyectos reales, por lo que, a diferencia de Arduino, su público objetivo son profesionales, no persona que desean iniciarse. Se trata de una plataforma que ya ha sido utilizada en proyectos reales que hoy en día están en funcionamiento y que cuenta con versatilidad al permitir el montaje de diversos sensores sobre la placa base. Comparte el mismo IDE de desarrollo que Arduino, por lo que permite migrar prototipos llevados a cabos en placas Arduino a este tipo de placas para su despliegue en la implantación del mundo real.

2.6.3. Intel Galileo

La placa Intel Galileo es la primera placa Arduino basada en la arquitectura de Intel. Al igual que en las placas Arduino se pueden utilizar módulos que se pueden conectan a la placa, lo que le permite extender su funcionalidad. Al igual que el Arduino Uno, tiene 14 pines de E/S digitales, 6 entradas

analógicas, un puerto serie y un encabezado ICSP para la programación en serie.

La placa cuenta con un procesador basado en el procesador IQuark SoC X1000, que ha sido diseñado específicamente para el Internet de las cosas. Además, sobre la placa se ejecuta un sistema operativo Linux libre que contiene las librerías software de Arduino, permitiendo con ello la posibilidad de reutilizar software ya existente para Arduino.

A parte de la compatibilidad con las placas Arduino, las placas Galileo cuentan con una serie de puertos y características que son estándares en la industria del PC (cuenta, entre otros, con puertos Mini-PCIe y Ethernet).

De esta plataforma cabe destacar que, gracias a su gran conectividad, potencia de procesamiento y el uso de SDKs sencillos ha posible el desarrollo de software capaz de conectar cualquier dispositivo a Internet, siendo su uso en domótica mu interesante sobre todo en proyectos que necesiten de un PC de bajo consumo energético y bajo coste [15].

En la siguiente captura de pantalla de puede apreciar la placa descrita:

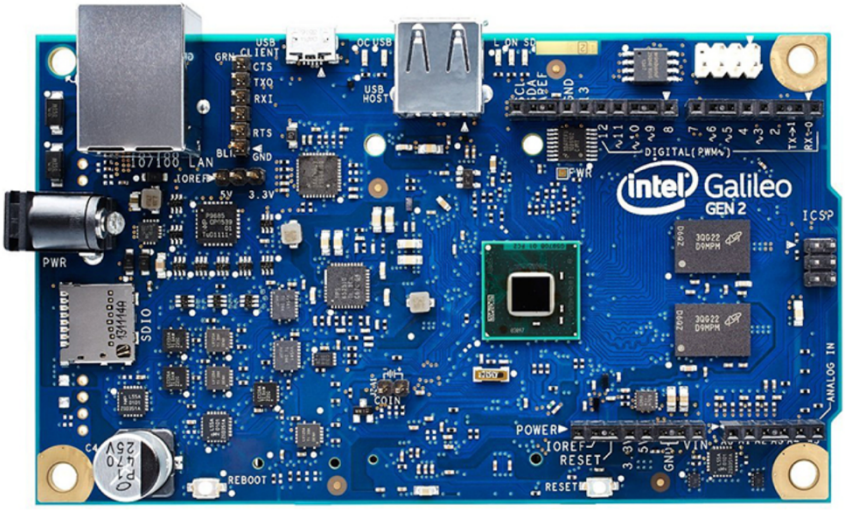


Figura 4 Placa Intel Galileo, segunda generación

Conclusiones:

Las placas Intel Galileo son muy similares a las Arduino UNO, y resultan ideales para iniciarse en el mundo de los microcontroladores. Su precio es inferior a los 70€, resultando tener un coste mayor que Arduino, aunque hay que tener en cuenta que estas placas son más potentes.

Sus usos son similares a los que tienen las placas Arduino: proyectos académicos o prototipos llevados a cabo por personas que se inician en el mundo de los microcontroladores y desean aprender.

2.6.4. Raspberry Pi

Raspberry Pi es una plataforma de código abierto que se ha vuelto muy popular, sobre todo en el mundo educativo, en los últimos años. Se trata de una computadora de bajo coste, del tamaño de una tarjeta de crédito, que se conecta a un monitor de computadora o televisor mediante conexión HDMI, y que usa un teclado y ratón estándar. Puede ejecutar una gran cantidad de sistemas operativos, como Raspbian (basado en Debian Linux), Android, Windows 10, IoT Core, etc… [16]

A continuación, se muestra una tabla comparativa con los diferentes modelos de Raspberry Pi del mercado junto a sus principales características diferenciadoras [17]:

Características	Raspberry Pi 2 B	Raspberry Pi 3 B	Raspberry Pi Zero	Raspberry Pi Zero W
CPU	Cortex-A7	Cortex-A53 64-bit	ARM1176JZF-S	ARM1176JZF-S
Nº Cores	4	4	1	1
CPU Clock	900 MHz	1.2 GHz	1 GHz	1 GHz
RAM	1 GB	1 GB	512 MB	512 MB
Memoria	Micro SD	Micro SD	Micro SD	Micro SD
USB	4	4	1 microUSB	1 microUSB
Ethernet	SI	SI	No	No
Wi-Fi	No	SI	No	SI
Bluetooth	No	SI	No	SI
HDMI	Si	Si	Mini	Mini
GPIO	17	17	17	17
Altura	85.6 mm	85.6 mm	65 mm	65 mm
Ancho	56.5 mm	56.5 mm	30 mm	30 mm
Profundidad	17 mm	17 mm	5 mm	5 mm
Peso	45 g	45 g	9 g	9 g
Consumo	820 mA	1400 mA	350 mA	350 mA

En las siguientes figuras podemos apreciar, respectivamente, los modelos 3B y Zero de Raspberry Pi:

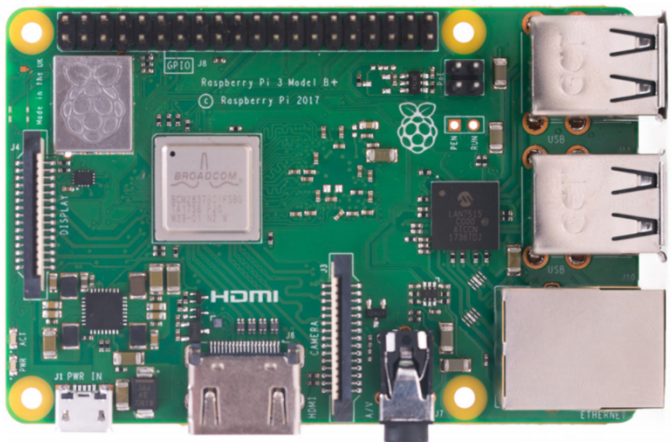


Figura 5 Raspberry Pi 3B

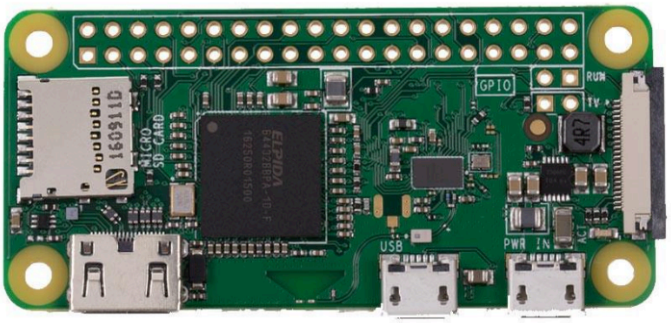


Figura 6 Raspberry Pi Zero W

Conclusiones:

La potente CPU junto con la red LAN inalámbrica y Bluetooth 4.1 convierte a Rapsberry Pi (en sus modelos 3 y Zero), en un candidato ideal para proyectos de IoT, ya que se pueden conectar múltiples sensores simultáneamente. Además, la Raspberry Pi tiene un conector GPIO (E/S de propósito general) de 40 pines para la interfaz con sensores externos.

La Raspberry Pi Zero es la Raspberry Pi más pequeña jamás fabricada y, aunque no tiene un procesador que sea tan poderoso como el Pi 3, su tamaño pequeño es especialmente adecuado para proyectos integrados (como wearables, etc.), donde el tamaño es una propiedad muy a tener en cuenta.

2.7. Plataformas software IoT

Una vez analizadas algunas de las plataformas hardware más comunes, en este punto se va a llevar a cabo un estudio de las principales plataformas software que hay actualmente en el mercado. Algunas de las plataformas que se analizarán son Open Source mientras que otras son propietarias, y en algunas de ellas la propia plataforma ofrece software, hardware y todo lo necesario para el desarrollo de un proyecto.

2.7.1. ThingSpeak

ThingSpeak es un servicio de plataforma open source de análisis de IoT que permite agregar, visualizar y analizar flujos de datos en vivo en la nube. El usuario puede enviar datos a ThingSpeak desde sus dispositivos, crear visualizaciones instantáneas de datos en vivo y enviar alertas utilizando servicios web como Twitter. Para usuarios más avanzados, mediante la integración del software MATLAB analytics con ThingSpeak, se puede escribir y ejecutar el código MATLAB para realizar el preprocesamiento, las visualizaciones y los análisis. ThingSpeak permite a los ingenieros y científicos

crear prototipos y construir sistemas de IoT sin tener que configurar servidores o desarrollar software web [18].

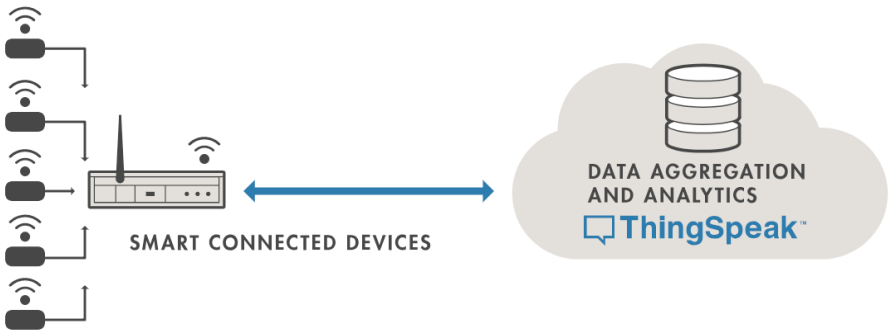


Figura 7 Plataforma ThingSpeak

Características principales

A continuación, se enumeran algunas de las características más relevantes de esta plataforma [18]:

- **API:** ThingSpeak dispone de una API disponible en GitHub (URL de acceso: https://github.com/iobridge/thingspeak) para su descarga y uso de forma gratuita. Esta librería es totalmente open source, por lo que la comunidad de usuarios puede modificar el código fuente original de forma totalmente libre y contribuir al proyecto si así lo desea incluyendo nuevas funcionalidades o mejorando las ya existentes.

- **Canales:** En esta plataforma, la vía a través de la cual se almacenan y publican los datos es a través de unas vías de comunicación denominadas “Chanels” (Canales).

- **Plugins:** Para la extensión de la funcionalidad básica de la plataforma, Thingspeak permite tanto el desarrollo de plugins propios como la instalación de plugins ya existentes desarrollados por otras personas. Esta funcionalidad constituye una poderosa herramienta que permite al usuario crear aplicaciones de forma nativa en la propia plataforma.

- **Integración:** ThingSpeak permite una amplia integración con plataformas tanto hardware como software. Las plataformas con las que ThingSpeak permite su integración son las siguientes:

    - Arduino
    - Raspberry Pi
    - IOBridge
    - Electric Imp
    - Móviles y aplicaciones web
    - Redes Sociales
    - Análisis de datos con MATLAB

- Aplicaciones disponibles: con la plataforma ThingSpeak, el usuario dispone de una serie de aplicaciones para IoT muy interesantes. A continuación, se describirán las más relevantes:

    - **ThingTweet:** Esta aplicación permite vincular una cuenta de Twitter a una cuenta de ThingSpeak. Con esta vinculación, se posibilita el envío de alertas a través de Twitter usando ThingTweet. Por ejemplo, se puede hacer que un dispositivo le envíe un tweet al usuario cuando la batería de un dispositivo se esté acabando o cuando algún proceso como el preparar un café de una cafetera se haya terminado.

    - **TalkBack:** Esta aplicación permite que cualquier dispositivo actúe sobre comandos en cola. Por ejemplo, si se tiene una puerta equipada con Wi-Fi y un sensor de movimiento, se puede poner en cola los comandos para abrir y cerrar la puerta. Cuando la puerta detecte a alguien cerca, se ejecutará el comando que la abrirá. Después de un tiempo especificado, se ejecutará el comando que la cerrará. Si no hay más comandos en la cola, la puerta no se abrirá cuando la siguiente persona se acerque.

    - **ThingHTTP:** La aplicación ThingHTTP permite la comunicación entre dispositivos, sitios web y servicios web sin tener que implementar el protocolo en el nivel del dispositivo. Esta aplicación permite al usuario crear sus propias acciones a través del uso de otras aplicaciones como TimeControl y React, que serán descritas a continuación.

    - **TimeControl:** TimeControl trabaja con otras aplicaciones ThingSpeak para realizar una acción en un momento específico o en un horario regular. Se puede usar TimeControl con:

        - ThingHTTP para comunicarse con dispositivos, sitios web o servicios web

        - ThingTweet para enviar alertas a través de Twitter

        - TalkBack para poner en cola los comandos de un dispositivo

      Un ejemplo de su uso es realizar una solicitud ThingHTTP que se conecte a un termostato que acepte solicitudes http para su control a distancia.

    - **React:** React funciona con las aplicaciones ThingHTTP, ThingTweet y MATLAB Analysis para realizar acciones cuando los datos de un determinado canal cumplen una determinada condición. Por ejemplo, se puede hacer que una aplicación móvil informe su latitud y longitud a un canal ThingSpeak y que cuando

su posición esté a cierta distancia del hogar del usuario, ThingHTTP encienda las luces de su sala de estar.

Ámbitos de aplicación

ThingSpeak se utiliza a menudo para la creación de prototipos y los sistemas de prueba de concepto IoT que requieren análisis [18].

Conclusiones

ThingSpeak se trata de una plataforma muy versátil que permite de forma relativamente sencilla visualizar y analizar flujos de datos en vivo en la nube. Algunas de las capacidades clave de ThingSpeak que la hacen una plataforma interesante son:

- Configurar fácilmente los dispositivos para enviar datos a ThingSpeak.
- Visualizar los datos de su sensor en tiempo real.
- Aprovechar el potencial del software MATLAB para el tratamiento de los datos recolectados por los dispositivos IoT.
- Ejecutar análisis de IoT automáticamente en base a horarios o eventos.
- Desarrollo de prototipos y sistemas IoT sin configurar servidores ni desarrollar software web.
- Actuar automáticamente sobre los datos y llevar a cabo comunicaciones utilizando servicios de terceros como Twitter.

2.7.2. Electric Imp

Electric Imp es una plataforma que se caracteriza por proveer soluciones finales con integración hardware, software, APIs, servicios en la nube, con total seguridad y rapidez.

Esta plataforma ofrece todo un ecosistema de productos, del que se analizarán a continuación los productos más interesantes.

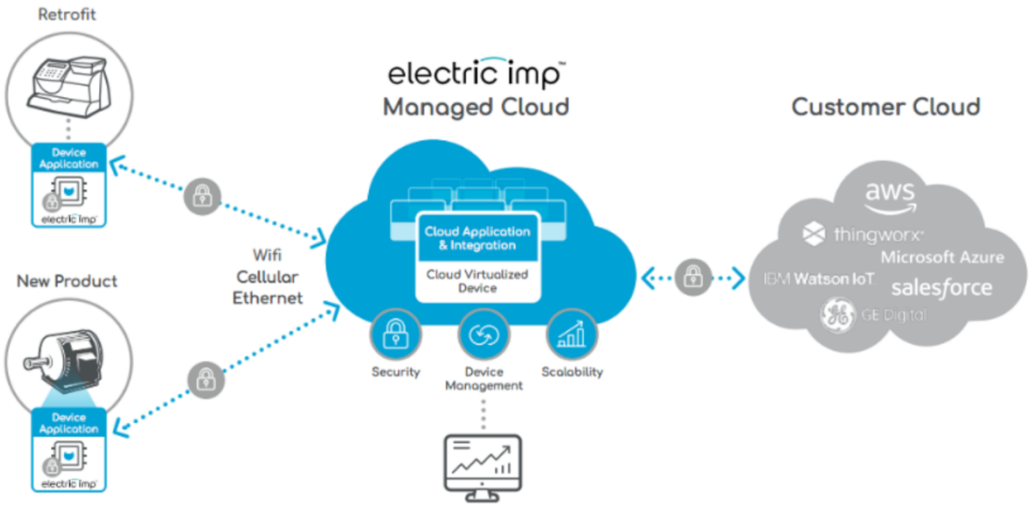


Figura 8 Ecosistema de la plataforma Electric Imp

Características principales

La plataforma cuenta con el sistema ImpOs como núcleo desde el que el dispositivo y su agente online proveen los mecanismos necesarios para satisfacer las necesidades del usuario en el desarrollo de cualquier proyecto.

Dispone de una API abierta que nos permite gestionar la conectividad , energía, seguridad, notificaciones, etc... El código de la aplicación desarrollada se será ejecutado en una máquina virtual que ImpOs ofrece.

Por otro lado, impCloud es la pieza central de la plataforma a través de la cual se conectan los dispositivos a través de internet y se les identifica mediante un agente online único. La plataforma impCloud ofrece: un servicio SaaS, almacenamiento de datos y visualización de reportes. En esta plataforma es el agente quien gestiona este tipo de comunicaciones todo ello bajo el protocolo HTTPS, securizando con ello las comunicaciones y haciendo que el usuario no tenga que preocuparse de hacerlo él mismo [20].

Ámbitos de aplicación

La plataforma Electic Imp cuenta con una oferta diversificada de soluciones en el mundo IoT, que van desde el Smart Home, hasta soluciones industriales y de análisis de gran volumen de datos.

Conclusiones

Electric Imp es una plataforma que nos ofrece un amplio ecosistema de servicios, que se caracteriza por ser escalable, teniendo en cuenta que aparte de usar sus módulos y su agente podemos usar cualquier tipo de dispositivo.

Además, pone a disposición del usuario diferentes herramientas para adaptar sus dispositivos a las necesidades cambiantes del proyecto, encargándose de todas las comunicaciones y su securización. Esto es un punto muy a tener en

cuenta a la hora de realizar cualquier proyecto, ya que proporciona un entorno de trabajo en el que el usuario se puede centrar en el desarrollo y estar seguro de que las comunicaciones realizadas entre los dispositivos implantados se llevan a cabo de forma segura y controlada.

2.7.3. Amazon Web Services IoT

Amazon Web Services IoT (AWS IoT, de ahora en adelante), proporciona al usuario software de dispositivos, servicios de control y servicios de datos. El software del dispositivo permite conectar dispositivos, recabar datos y tomar acciones inteligentes a nivel local de manera segura, incluso sin conexión a Internet. Los servicios de control permiten controlar, administrar y asegurar los dispositivos. Por último, los servicios de datos ayudan en la extracción del valor de los datos recogidos de los dispositivos IoT [21].

Características principales

A continuación, se enumeran algunas de las características más relevantes de esta plataforma [21]:

- **SDK para dispositivos con AWS IoT:** AWS IoT cuenta con un SDK para dispositivos que permite conectar de forma sencilla el dispositivo hardware con a AWS IoT Core, del que se hablará más adelante. El SDK para dispositivos con AWS IoT permite conectar, autenticar e intercambiar mensajes con AWS IoT Core mediante protocolos MQTT, HTTP o WebSockets. Además, admite como lenguajes de programación C, JavaScript y Arduino, e incluye bibliotecas cliente, guía para desarrolladores y guía de puertos para fabricantes. También le permite al usuario la opción de utilizar un código abierto alternativo o escribir su propio SDK.

- **Gateway para dispositivos:** La gateway para dispositivos funciona como punto de entrada para los dispositivos compatibles con IoT que se conectan a AWS. La gateway para dispositivos se encarga de administrar todas las conexiones de dispositivos activas e implementa semántica para varios protocolos con el objetivo de garantizar que los dispositivos puedan comunicarse de manera segura y eficiente con AWS IoT Core. La gateway para dispositivos está completamente administrada y ajusta su escala automáticamente para admitir más de mil millones de dispositivos sin la necesidad de tener que administrar infraestructuras.

- **Agente de mensajes:** AWS IoT cuenta con un agente de mensajes de publicación/suscripción que transmite mensajes de manera segura hacia y desde aplicaciones y dispositivos compatibles con IoT con baja latencia.

- **Autenticación y autorización:** AWS IoT Core ofrece autenticación mutua y cifrado en todos los puntos de conexión para que los datos

nunca se intercambien entre dispositivos y AWS IoT Core sin una identidad comprobada. AWS IoT Core admite el método de autenticación de AWS (llamado "SigV4"), la autenticación basada en el certificado X.509 y la autenticación basada en token creado por el cliente (a través de autorizadores personalizados). El usuario puede crear, implementar y administrar certificados y políticas para los dispositivos en la consola o con la API. Estos certificados de dispositivos se pueden aprovisionar, activar y asociar con las políticas de IoT configuradas con AWS IoT Core, con lo que si así lo desea puede anular instantáneamente el acceso de un dispositivo individual.

- **Motor de reglas:** El motor de reglas permite crear aplicaciones de IoT que unan, procesen, analicen y actúen sobre datos generados por dispositivos conectados a escala global sin tener que administrar ninguna infraestructura.

A parte de estas características, AWS IoT ofrece los siguientes servicios:

- **Amazon FreeRTOS:** es un sistema operativo para microcontroladores que facilita la programación, implementación, protección, conexión y administración de los dispositivos de borde pequeños y de poca potencia.

- **AWS IoT Greengrass:** es un software que le permite ejecutar capacidades de computación local, mensajería, almacenamiento de datos en caché, sincronización e inferencias de aprendizaje automático en dispositivos conectados de manera segura.

- **AWS IoT Core:** permite que los dispositivos conectados interactúen de manera sencilla y segura con las aplicaciones en la nube y otros dispositivos.

- **AWS IoT Device Management:** facilita la incorporación, la organización, la monitorización y la administración remota de dispositivos IoT a escala y de forma segura.

- **AWS IoT Device Defender:** monitoriza y audita continuamente sus configuraciones de IoT para garantizar que no se aparten de las prácticas recomendadas de seguridad.

- **AWS IoT Things Graph:** facilita la conexión de diferentes dispositivos y servicios en la nube para crear aplicaciones IoT.

- **AWS IoT Analytics:** facilita la ejecución de análisis sofisticados en volúmenes masivos de datos de IoT.

- **AWS IoT SiteWise:** facilita la recopilación, estructuración y búsqueda de datos de IoT que provienen de bases de datos de instalaciones

industriales, que luego utiliza para analizar el rendimiento de los equipos y procesos.

- **AWS IoT Events:** facilita las tareas de detección y respuesta a eventos que provienen de grandes cantidades de aplicaciones y sensores compatibles con IoT.

En la siguiente figura se pueden apreciar de forma gráfica todos los servicios descritos:

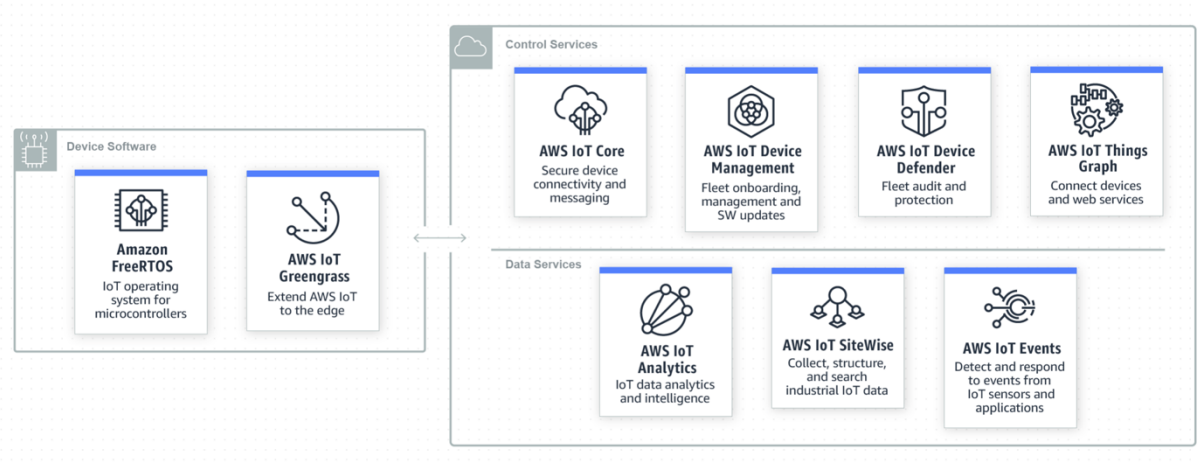


Figura 9 Servicios de AWS IoT

Ámbitos de aplicación

Los principales ámbitos de aplicación de esta plataforma y sus servicios son:

- **Industria:** la plataforma permite el desarrollo de aplicaciones industriales de IoT para poder realizar un mantenimiento predictivo y garantizar la calidad, así como para vigilar a distancia las operaciones.

- **Smart Home:** en proyectos de domótica, seguridad, vigilancia y redes domésticas.

- **Proyectos comerciales:** en aplicaciones comerciales para la monitorización de tráfico y de estado, y para la seguridad pública.

Conclusiones

AWS IoT es una plataforma con una gran variedad de servicios muy potentes que ofrecen al usuario una interesante variedad de soluciones. Otros puntos destacables de esta plataforma son su escalabilidad y la posibilidad que brinda al usuario de desarrollar sus propias aplicaciones conectadas con los servicios de AWS IoT para la gestión de sus propios dispositivos IoT y los datos obtenidos por éstos.

2.7.4. Google Cloud IoT

La plataforma Google Cloud IoT ofrece un conjunto completo de herramientas para conectar, procesar, almacenar y analizar datos tanto en el perímetro como en la nube. La plataforma se compone de servicios en la nube escalables y totalmente administrados, una pila de software integrada con funciones de aprendizaje automático y el sistema operativo administrado Android Things desarrollado para el Internet de las cosas.

Características principales

A continuación, se enumeran algunas de las características más relevantes de esta plataforma [22]:

- **Sistema operativo Android Things:** es una plataforma de sistema operativo integrado basada en Android diseñada para ser utilizada con dispositivos IoT de baja potencia y con memoria limitada.

- **Sencillez:** Google pretende hacer más sencillo el uso de los dispositivos IoT y el tratamiento de los datos que éstos recogen proporcionando la infraestructura y los servicios que el usuario necesite para administrar esos datos, utilizando los servicios de software de Google.

En la implementación de una solución con Google Iot Core se pueden diferenciar los siguientes bloques principales:

- Un Administrador de Dispositivos encargado registrar cada uno de los dispositivos IoT de los que se van a recopilar los datos. Este proceso puede realizarse de forma manual a través de una consola o mediante programación, para registrar los dispositivos de una manera más automatizada, enfocado a escenarios que implican miles o incluso decenas de miles de dispositivos. El administrador de dispositivos establece la identidad de un dispositivo y proporciona un mecanismo para autenticarlo cuando se conecta a la nube, al tiempo que mantiene una configuración para cada dispositivo que ayuda a que Google Cloud le reconozca.

- Un Puente de Protocolo, que le proporciona a los dispositivos IoT una forma de comunicarse utilizando protocolos estándar con el servicio Google Cloud. Este servicio incluye soporte nativo para la conexión segura a través de MQTT, un protocolo estándar de IoT.

Ámbitos de aplicación

Algunos ámbitos de aplicación de esta plataforma son:

- **Mantenimiento predictivo:** Consiste en la predicción de forma automatizada de cuándo se debe realizar el mantenimiento de los

equipos, optimización de su rendimiento en tiempo real, anticipación de los periodos inactivos, detección de anomalías y monitorización del estado y la ubicación de los dispositivos.

- **Monitorización de recursos en tiempo real:** La plataforma permite monitorizar los recursos valiosos en tiempo real, ejecutar análisis complejos y procesos de aprendizaje automático en los datos recopilados y mostrar el estado de una empresa para aportar métricas útiles.

Conclusiones

La plataforma de Google es una herramienta potente y versátil que al usuario ofrece administración de dispositivos, escala de infraestructura, redes y una variedad de productos de análisis y almacenamiento de datos que el usuario puede utilizar de forma relativamente sencilla para aprovechar al máximo los datos generados por sus dispositivos.

3. Análisis de las amenazas y vulnerabilidades más comunes en IoT

Un ecosistema IoT, como cualquier objeto que se encuentra conectado a Internet y por lo tanto es accesible desde cualquier lugar, es por lo tanto susceptible de sufrir una intrusión por parte de un atacante externo, ya sea mediante la explotación de alguna vulnerabilidad en el código fuente, la captura de las comunicaciones o simplemente por simples fallos de configuración en algún dispositivo.

Debido al gran crecimiento que, como se pudo ver en el apartado anterior, está viviéndose en torno al IoT, los cibercriminales están centrando sus esfuerzos en comprometer el mayor número de dispositivos posibles con el fin de utilizarlos para sus fines. El abanico de las posibles amenazas sobre un ecosistema IoT es muy amplio, pero en esta memoria se va a realizar un análisis de las más comunes.

3.1 Análisis de amenazas en IoT

3.1.1 Ataques DDoS

Uno de los mayores problemas que se está dando actualmente en los ecosistemas IoT son los ataques de denegación de servicio distribuidos (DDoS). Hoy en día existen numerosas botnets compuestas por multitud de "objetos" que se encuentran conectados a internet que por sí mismos no cuentan con un alto poder de computación, pero que con su capacidad de poder generar peticiones TCP/UDP un ciberdelincuente tiene la posibilidad de colapsar los recursos de cualquier empresa e incluso los servidores DNS públicos.

El mayor problema para este tipo de ataques es que es difícil prevenirse contra ellos, ya que, aunque existan empresas especializadas en la protección contra este tipo de ataques como la empresa Neustrar [9], su efectividad no es absoluta. Además, las direcciones IP de los dispositivos que se ven afectados por una botnet suelen cambiar a menudo (IPs dinámicas), los dueños no suelen saben que sus dispositivos han sido capturados y en la gran mayoría de los casos nunca surge un parche de seguridad por parte de los fabricantes que solucione el problema.

3.1.2 Espionaje y vigilancia

Muchos de los aparatos IoT implantados son cámaras de seguridad y monitorización que se utilizan con el objetivo de vigilar un área de interés para el usuario que los implanta. Aunque el objetivo de este tipo de dispositivos es video vigilar el área deseada, los dueños de estos dispositivos acaban fácilmente siendo víctimas de chantaje o en el mejor de los casos sufren simplemente un ataque no malintencionado de una tercera persona que toma el control de los dispositivos.

Normalmente los vectores de ataque para estos casos suelen provenir de una configuración errónea o insuficiente por parte del usuario que utiliza este tipo de dispositivos del usuario y contraseña, dejando las credenciales por defecto del dispositivo.

3.1.3 Ransomware

El secuestro de ordenadores de sobremesa y portátiles por este tipo de ataques es algo habitual para los usuarios medios como empresas, ya que en estos dispositivos se suele almacenar información valiosa para el usuario. Sin embargo, los dispositivos IoT no almacenan información sensible, por lo que este tipo de ataque no suele ser el más común en este tipo de tecnología. Sin embargo, un ataque de este tipo sobre un ecosistema IoT puede ser útil para el atacante cuyo fin sea inutilizar cierto sistema pidiendo una retribución económica a cambio de su liberación.

En el año 2017 en Victoria, Canada, se dio un caso en el que 55 cámaras de velocidad se vieron infectadas con el ransomware WannaCry, quedando totalmente inutilizadas. Si bien la infección dada en este caso fue simplemente un error por parte de un técnico que difundió el malware de una máquina a otra sin querer, se demostró el tipo de consecuencias que tal tipo de ataque podrían tener en el caso de que un atacante malintencionado paralizara todo un ecosistema IoT para posteriormente pedir algo a cambio de liberar los dispositivos capturados [10].

3.1.4 Movimientos laterales

En este tipo de ataque los dispositivos IoT no son el objetivo real del ataque, si no que son utilizados como un punto de entrada que le permite al cibercriminal moverse de forma lateral a otras áreas de la red interna en la que el dispositivo IoT atacado se encuentra.

Un ejemplo de este tipo de ataque es el de la vulneración de una impresora en el que un atacante interno reemplaza un controlador de impresora con un archivo malicioso. Cuando un usuario se conecta a esa impresora, el código malicioso es entregado y ejecutado sin controles a nivel del sistema, repitiéndose este proceso se con usuarios adicionales que se conecten a la impresora [23].

3.2 Análisis de vulnerabilidades en IoT

Tal y como ocurría con las amenazas, en IoT existen diversas amenazas debido a la gran variedad de tecnologías disponibles en los ecosistemas IoT. A continuación, se va a llevar a cabo un análisis de las más comunes según la guía OWASP [24]. Para el análisis se expondrá, para cada vulnerabilidad, los agentes de la amenaza, los vectores de ataque, la o las vulnerabilidades que se explotan, el impacto técnico y el impacto a nivel de negocio.

3.2.1 Interfaces web no seguras

- **Agentes de la amenaza**

Se considera como agente en este caso a cualquier persona que tiene acceso a la interfaz web del dispositivo, incluyendo tanto a usuarios internos como externos.

- **Vectores de ataque**

El atacante hace uso de credenciales débiles o captura las credenciales en texto plano y consigue acceso a la interfaz web del dispositivo. Este ataque puede verse realizado tanto por un usuario interno como por uno externo.

- **Vulnerabilidades de seguridad**

Podemos encontrar una interfaz web insegura cuando se presentan problemas como la ausencia de bloqueos de cuentas por intentos de acceso fallidos o credenciales de cuentas débiles. Las interfaces web inseguras prevalecen ya que la intención de éstas es estar únicamente expuestas en redes internas, sin embargo, las amenazas por parte de un usuario interno pueden ser tan importantes como las amenazas de un usuario externo. Los problemas con la interfaz web son fáciles de descubrir al examinar la interfaz manualmente junto con herramientas automatizadas que ayuden a identificar problemas como el cross-site scripting.

- **Impactos técnicos**

Una interfaz web insegura puede provocar la pérdida o la corrupción de datos, denegación de acceso y puede conllevar a la toma del control del dispositivo por parte del atacante.

- **Impactos en el negocio**

Hay que considerar el impacto en el negocio de las interfaces web mal aseguradas que podrían llevar a dispositivos comprometidos junto con clientes comprometidos. Una interfaz web no segura podría conllevar que los clientes se viesen perjudicados y/o que la marca de viese dañada.

3.2.2 Autenticación/autorización insuficiente

- **Agentes de la amenaza**

Cualquier persona que tenga acceso a la interfaz web, la interfaz móvil o la interfaz en la nube, incluidos los usuarios internos y externos.

- **Vectores de ataque**

El atacante hace uso de credenciales débiles, mecanismos de recuperación de contraseña inseguros, credenciales mal protegidas o falta de control de acceso granular para acceder a una interfaz en particular. Este ataque puede verse realizado tanto por un usuario interno como por uno externo.

- **Vulnerabilidades de seguridad**

La autenticación puede no ser suficiente cuando se usan contraseñas débiles o si estas están mal protegidas. La autenticación/autorización insuficiente prevalece ya que se supone que las interfaces solo estarán expuestas a los usuarios en redes internas y no a usuarios externos en otras redes. Las deficiencias se encuentran a menudo presentes en todas las interfaces. Muchos de los problemas con la autenticación/autorización son fáciles de descubrir cuando se examina la interfaz manualmente, y también se pueden descubrir a través de pruebas automatizadas.

- **Impactos técnicos**

La autenticación/autorización insuficiente puede provocar la pérdida o corrupción de datos, denegación de acceso y puede comprometer completamente el dispositivo y/o las cuentas de usuario.

- **Impactos en el negocio**

Consideremos el impacto en el negocio de las cuentas de usuario comprometidas y, posiblemente, de los dispositivos. Todos los datos pueden ser robados, modificados o eliminados. Todo esto podría conllevar que los clientes se viesen perjudicados.

3.2.3 Servicios de red inseguros

- **Agentes de la amenaza**

Cualquier persona que tenga acceso al dispositivo a través de una conexión de red, incluidos los usuarios externos e internos.

- **Vectores de ataque**

El atacante utiliza servicios de red vulnerables para atacar el dispositivo en sí o para realizar ataques laterales desde el dispositivo. El ataque podría venir de usuarios externos o internos.

- **Vulnerabilidades de seguridad**

Los servicios de red inseguros pueden ser susceptibles a ataques de desbordamiento de búfer o ataques que crean una condición de denegación de servicio que deja el dispositivo inaccesible para el usuario. Los ataques de denegación de servicio contra otros usuarios también pueden facilitarse cuando hay servicios de red inseguros disponibles. Los servicios de red inseguros a

menudo pueden ser detectados por herramientas automatizadas, como escáneres de puertos y fuzzers.

- **Impactos técnicos**

Los servicios de red inseguros pueden provocar la pérdida o corrupción de datos, la denegación de servicio o la facilitación de ataques en otros dispositivos.

- **Impactos en el negocio**

Consideremos el impacto en el negocio de los dispositivos que se han vuelto inútiles debido a un ataque de denegación de servicio o de un dispositivo que se utiliza para facilitar los ataques contra otros dispositivos y redes. Estas consecuencias podrían conllevar que los clientes u otros usuarios se viesen perjudicados.

3.2.4 Ausencia de cifrado en las comunicaciones

- **Agentes de la amenaza**

Cualquier persona que tenga acceso a la red a la que está conectado el dispositivo, incluidos los usuarios externos e internos.

- **Vectores de ataque**

El atacante se aprovecha de la ausencia de cifrado de las comunicaciones para ver los datos que son transferidos a través de la red. El ataque podría venir de usuarios externos o internos.

- **Vulnerabilidades de seguridad**

La ausencia de encriptación de las comunicaciones permite ver los datos mientras viajan a través de redes locales o de Internet. La ausencia de encriptación de las comunicaciones prevalece en las redes locales, ya que es fácil suponer que el tráfico de la red local no será ampliamente visible; sin embargo, en el caso de una red inalámbrica local, la mala configuración de la red inalámbrica puede hacer que el tráfico sea visible para cualquier persona dentro del alcance de esa red. Muchos problemas con el cifrado son fáciles de descubrir simplemente al ver el tráfico de la red y buscar datos legibles. Las herramientas automatizadas también pueden buscar la implementación adecuada del cifrado de transporte común, como SSL y TLS.

- **Impactos técnicos**

La ausencia de encriptación de las comunicaciones puede provocar la pérdida de datos y, dependiendo de los datos expuestos, podría comprometer completamente el dispositivo o las cuentas de usuario.

- **Impactos en el negocio**

Considere el impacto en el negocio de los datos expuestos mientras viajan a través de varias redes. Los datos pueden ser robados o modificados, conllevando con ello que los usuarios cuyos datos han sido expuestos podrían verse perjudicados.

3.2.5 Interfaz en la nube insegura

- **Agentes de la amenaza**

Cualquier persona con acceso a Internet.

- **Vectores de ataque**

El atacante utiliza varios vectores, como la autenticación insuficiente, la falta de encriptación de las comunicaciones y la enumeración de cuentas para acceder a los datos o controles a través del sitio web de la nube. El ataque probablemente vendrá de internet.

- **Vulnerabilidades de seguridad**

Una interfaz de nube insegura está presente cuando se usan credenciales fáciles de adivinar o es posible la enumeración de la cuenta. Las interfaces de nube inseguras son fáciles de descubrir simplemente revisando la conexión a la interfaz de nube e identificando si SSL está en uso o utilizando el mecanismo de restablecimiento de contraseña para identificar cuentas válidas que pueden conducir a la enumeración de cuentas.

- **Impactos técnicos**

Una interfaz de nube insegura podría comprometer los datos del usuario y el control del dispositivo.

- **Impactos en el negocio**

Consideremos el impacto empresarial de una interfaz de nube insegura. Los datos pueden ser robados o modificados y se asume el control de los dispositivos. Esta situación podría conllevar que los clientes se viesen perjudicados e incluso que la marca se vea dañada.

3.2.6 Interfaz móvil insegura

- **Agentes de la amenaza**

Cualquier persona que tenga acceso a la aplicación móvil.

- **Vectores de ataque**

El atacante utiliza varios vectores, como la autenticación insuficiente, la falta de cifrado de las comunicaciones y la enumeración de cuentas para acceder a los datos o controles a través de la interfaz móvil.

- **Vulnerabilidades de seguridad**

Una interfaz móvil insegura está presente cuando se usan credenciales fáciles de adivinar o es posible la enumeración de la cuenta. Las interfaces móviles inseguras son fáciles de descubrir simplemente revisando la conexión a las redes inalámbricas e identificando si SSL está en uso o utilizando el mecanismo de restablecimiento de contraseña para identificar cuentas válidas que pueden conducir a la enumeración de cuentas.

- **Impactos técnicos**

Una interfaz móvil insegura podría comprometer los datos del usuario y el control del dispositivo.

- **Impactos en el negocio**

Consideremos el impacto en el negocio de una interfaz móvil insegura. Los datos pueden ser robados o modificados y se asume el control de los dispositivos, conllevando con ello que los clientes puedan verse perjudicados y/o que la marca se vea dañada.

3.2.7 Configurabilidad de seguridad insuficiente

- **Agentes de la amenaza**

Cualquier persona con acceso al dispositivo

- **Vectores de ataque**

El atacante utiliza la ausencia de permisos granulares para acceder a datos o controles en el dispositivo. El atacante también podría utilizar la ausencia de opciones de cifrado y la falta de opciones de contraseña para realizar otros ataques que puedan comprometer el dispositivo y/o los datos. El ataque podría provenir de cualquier usuario del dispositivo, ya sea de forma intencional o accidental.

- **Vulnerabilidades de seguridad**

La configuración de seguridad insuficiente está presente cuando los usuarios del dispositivo tienen una capacidad limitada o nula para alterar sus controles de seguridad. Una configuración de seguridad insuficiente es evidente cuando la interfaz web del dispositivo no tiene opciones para crear permisos de usuario granulares o, por ejemplo, forzar el uso de contraseñas seguras. La revisión manual de la interfaz web y sus opciones disponibles revelarán estas deficiencias.

- **Impactos técnicos**

Una configuración de seguridad insuficiente podría comprometer el dispositivo, ya sea de forma intencional o accidental y/o pérdida de datos

- **Impactos en el negocio**

Consideremos el impacto en el negocio si los datos pueden ser robados o modificados y se asume el control del dispositivo. Por ello, debido a una configurabilidad de seguridad insuficiente, cabe la posibilidad de que los clientes se vean perjudicados.

3.2.8 Software/Firmware inseguro

- **Agentes de la amenaza**

Cualquier persona con acceso al dispositivo y/o a la red a la que el dispositivo se encuentra conectado. También se puede considerar como un agente a cualquier persona que pueda acceder al servidor de actualizaciones.

- **Vectores de ataque**

El atacante utiliza varios vectores, como la captura de archivos de actualización a través de conexiones no cifradas, el archivo de actualización en sí no está encriptado o puede realizar su propia actualización maliciosa a través del secuestro de DNS. Según el método de actualización y la configuración del dispositivo, el ataque podría provenir de la red local o de Internet.

- **Vulnerabilidades de seguridad**

La falta de capacidad para actualizar un dispositivo presenta una debilidad de seguridad en sí misma. Los dispositivos deben poder actualizarse cuando se descubren vulnerabilidades y las actualizaciones de software/firmware pueden ser inseguras cuando los archivos actualizados y la conexión de red a la que se entregan no están protegidos. El software/firmware también puede ser inseguro si contiene datos confidenciales codificados, como las credenciales. Los problemas de seguridad con el software/firmware son relativamente fáciles de descubrir simplemente inspeccionando el tráfico de la red durante la actualización para verificar el cifrado o utilizando un editor hexadecimal para inspeccionar el archivo de actualización en busca de información interesante.

- **Impactos técnicos**

El software/firmware inseguro podría comprometer los datos del usuario, el control del dispositivo y los ataques contra otros dispositivos.

- **Impactos en el negocio**

Consideremos el impacto en el negocio si se pueden robar o modificar los datos y tomar el control de los dispositivos con el propósito de atacar a otros dispositivos, pudiendo conllevar con ello que los clientes se vean perjudicados e incluso que otros usuarios se viesen también perjudicados.

3.2.9 Seguridad física insuficiente

- **Agentes de la amenaza**

Cualquier persona que tenga acceso físico al dispositivo.

- **Vectores de ataque**

El atacante utiliza vectores como puertos USB, tarjetas SD u otros medios de almacenamiento para acceder al sistema operativo y, potencialmente, a cualquier dato almacenado en el dispositivo.

- **Vulnerabilidades de seguridad**

Las debilidades de seguridad física están presentes cuando un atacante puede desarmar un dispositivo para acceder fácilmente al medio de almacenamiento y cualquier información almacenada en ese medio. Las debilidades también están presentes cuando los puertos USB u otros puertos externos se pueden usar para acceder al dispositivo utilizando funciones destinadas a la configuración o el mantenimiento.

- **Impactos técnicos**

La ausencia de seguridad física podría comprometer el propio dispositivo y cualquier dato almacenado en él.

- **Impactos en el negocio**

Los datos podrían ser robados o modificados y podría tomarse el dispositivo para fines distintos a los que se pretendía originalmente, conllevando con ello que los clientes se viesen perjudicados y/o que la marca se viese dañada.

Como se ha podido ver, el número de amenazas y vulnerabilidades en IoT es bastante amplio.

A continuación, se realizará un análisis los routers, que suponen otro factor a través del cual un ecosistema IoT puede verse vulnerado, bien sea debido a una mala configuración, vulnerabilidades en el software, etc…

4. Análisis de la seguridad y las amenazas de los routers

Los routers actúan como la barrera entre Internet y nuestra red local privada, por lo que es importante que un router se encuentre configurado de una forma minuciosa y cuidadosa ya que es la “cerradura” de nuestra empresa para un tercero no deseado que desea entrar en nuestra red. Si bien es importante la seguridad cara a las redes externas, también es importante considerar las medidas de seguridad a nivel interno, ya que unas malas políticas de seguridad internas pueden conllevar a la apertura de una puerta por la que un atacante puede acceder a nuestra red.

Uno de los primeros problemas que se puede encontrar en los routers es la ausencia de actualizaciones por parte del fabricante. Muchos de los routers instalados por parte de los Proveedores de Servicios de Internet (ISP, Internet Service Provider en inglés), no suelen ser actualizados con frecuencia, ya que estas actualizaciones deben ser lanzadas por el proveedor del servicio que lo ha instalado y esto no suele ocurrir.

Un router se puede considerar como un dispositivo IoT en el momento en el que éste cuenta con puertos USBs y se conectan en ellos discos duros accesibles desde Internet, nubes personalizadas, etc…, por lo que su vulneración supone que tanto nuestra propia red como los datos de los dispositivos que se encuentran conectados al router pueden verse en peligro.

A continuación, se describirán dos vulnerabilidades de los routers de las que un cibercriminal puede aprovecharse pudiendo con ello además comprometer los dispositivos que se encuentran conectados al/a los router/s vulnerado/s. Para el análisis se expondrá, para cada vulnerabilidad, los agentes de la amenaza, los vectores de ataque, la o las vulnerabilidades que se explotan, el impacto técnico y el impacto a nivel de negocio.

4.1. Vulnerabilidad Filet-O-Firewall

- **Agentes de la amenaza**

Cualquier persona con acceso a Internet.

- **Vectores de ataque**

Un atacante con un sitio web especialmente diseñado puede hacer que un usuario que esté ejecutando los navegadores Google Chrome o Firefox con JavaScript habilitado pueda realizar solicitudes de UPnP arbitrarias a su firewall, lo que abre su red a Internet.

Hoy en día, la mayoría de los routers domésticos que realizan la traducción de direcciones de red (NAT) también actúan como un servidor de seguridad que evita que los dispositivos en Internet inicien la comunicación con los dispositivos que se encuentran detrás del router, actuando por tanto como un

firewall. Este sería el funcionamiento normal de un router antes de haber sufrido el ataque a la vulnerabilidad Filet-O-Firewall [26]. En el siguiente diagrama se puede ver el funcionamiento descrito:

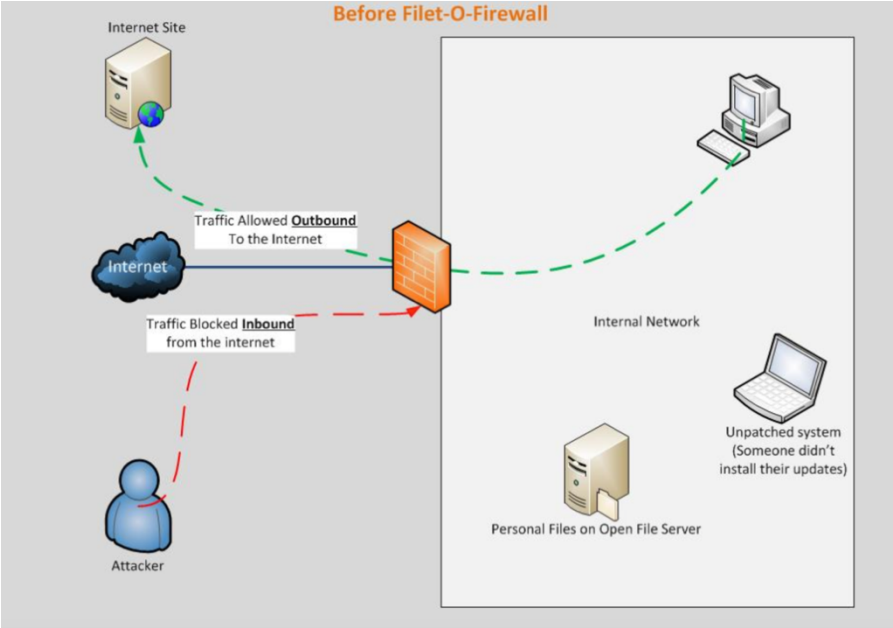


Figura 10 Funcionamiento normal de router actuando como firewall

- **Vulnerabilidades de seguridad**

Un atacante que explota la vulnerabilidad de Filet-O-Firewall podría exponer cualquiera o todos los dispositivos detrás del firewall de un usuario directamente a Internet. El proceso se puede hacer de forma casi transparente para el usuario final sin que el usuario instale o ejecute ninguna aplicación. El usuario simplemente debe navegar al sitio web del atacante usando un navegador afectado con JavaScript habilitado [26]. El resultado de la explotación de esta vulnerabilidad se puede visualizar en el siguiente diagrama:

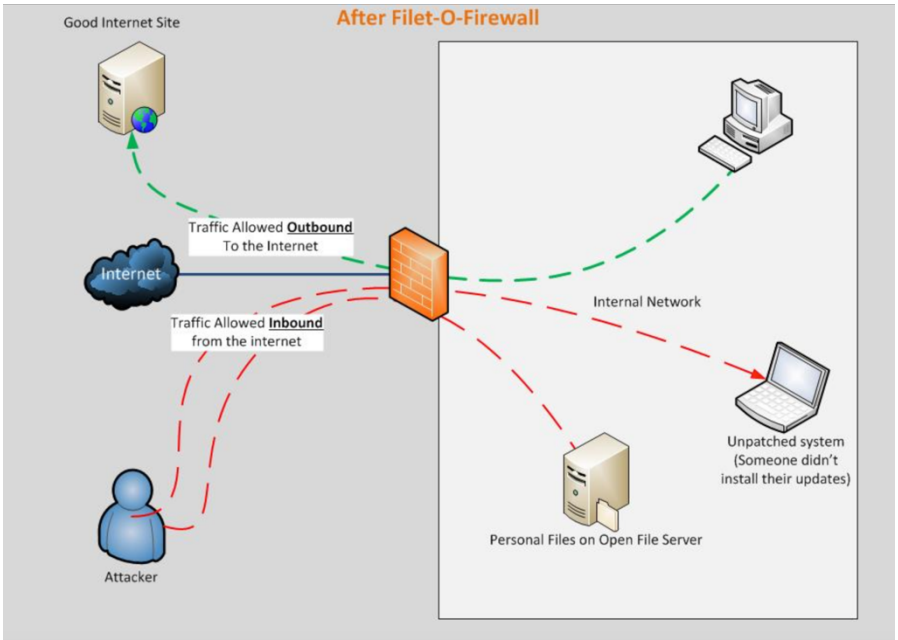


Figura 11 Explotación de la vulnerabilidad Filet-O-Firewall

- **Impactos técnicos**

La explotación de esta vulnerabilidad podría comprometer a todos los dispositivos conectados a la red. Ser víctima de este tipo de ataque puede tener graves consecuencias. Imaginemos un escenario de una PYME que cae víctima de este tipo de ataque, brindándole la posibilidad al atacante de detectar los dispositivos conectados a la red, ver que *IOTs* pueden ser objetivo de un ataque explotando sus vulnerabilidades y abrir una serie de puertos a esos *IOTs*.

- **Impactos en el negocio**

Ser víctima de la explotación de esta vulnerabilidad podría conllevar a que datos de la empresa puedan verse expuestos, hecho que podría perjudicar a los clientes y a la propia marca.

Para evitar esta vulnerabilidad el usuario debe desactivar en la configuración de su router la opción de uso del protocolo de comunicación uPnP, a menos que sea absolutamente necesario [27].

4.2. Ataque DNS Changer

- **Agentes de la amenaza**

Cualquier persona que tenga acceso a internet.

- **Vectores de ataque**

DNS Changer se trata de un troyano cuya función es la de modificar los servidores DNS de la conexión del usuario para apuntar a los sitios web del atacante. Es decir, que cuando el usuario trata de acceder a un sitio web normal, con los servidores DNS modificados se redirige al usuario a portales web de phishing. De esta forma el atacante puede conseguir que, creyendo el usuario que está accediendo por ejemplo a 'facebook.com', realmente se le redirige a un portal que utiliza su misma interfaz para robar sus credenciales. Pero esta anterior no es la única de sus posibilidades, sino que también se dedica a inyectar publicidad en los dispositivos conectados [28].

Los ataques de DNSChanger comienzan cuando un cibercriminal compra y coloca anuncios en sitios web convencionales. Esos anuncios contienen código JavaScript malicioso que puede revelar la dirección IP local de un usuario al activar lo que se llama una solicitud WebRTC a un servidor STUN de Mozilla (stun.services.mozilla [.] com).

Una vez que el atacante establece la dirección IP local de un objetivo, intenta determinar si vale la pena atacar al objetivo. Si no, a la víctima se le muestra un anuncio benigno. Los objetivos deseables reciben un anuncio falso en forma de una imagen PNG que contiene el malware.

A continuación, DNSChanger usa Chrome para cargar múltiples funciones, incluida una clave AES oculta con estenografía en una imagen pequeña. La clave AES se utiliza para ocultar el tráfico y descifrar los fingerprints del router para determinar si un objetivo está utilizando un modelo de router vulnerable.

- **Vulnerabilidades de seguridad**

En los casos en que el enrutador no es vulnerable, el atacante usará DNSChanger para intentar usar las credenciales predeterminadas para cambiar las entradas de DNS. Si la vulnerabilidad está presente, el atacante utilizará las vulnerabilidades conocidas del router para modificar las entradas de DNS en el router y también intentará que los puertos de administración estén disponibles desde direcciones externas para ataques adicionales.

Algunas recomendaciones para defenderse ante este tipo de ataques son tener actualizado el router a su última versión, cambiar el rango de IP local predeterminado en los routers, deshabilitar las funciones de administración remota en los routers y usar complementos del navegador que bloquean los anuncios [29].

- **Impactos técnicos**

La explotación de esta vulnerabilidad podría comprometer a todos los dispositivos conectados a la red. Ser víctima de este tipo de ataque provoca que todas las comunicaciones de los dispositivos implantadas sean redirigidas a los servidores del atacante.

- **Impactos en el negocio**

Ser víctima de la explotación de esta vulnerabilidad podría conllevar a que datos de la empresa puedan verse expuestos, hecho que podría perjudicar a los clientes y a la propia marca.

5. Comunicaciones seguras en ecosistemas IoT

Esta sección proporciona una investigación de algunas tecnologías que se pueden utilizar para implementar un mecanismo de seguridad adecuado para la protección de las comunicaciones y los datos de los dispositivos de un ecosistema IoT.

5.1. DTLS

Datagram Transport Layer Security (DTLS) es una extensión de TLS que funciona en la capa de sesión del modelo OSI para proporcionar autenticidad, integridad y confidencialidad en las comunicaciones.

En un dispositivo IoT no es factible el uso de TLS ya que usa TCP como protocolo subyacente, mientras que los dispositivos IoT utilizan el protocolo UDP. Por lo tanto, DTLS, que sí utiliza UDP, puede ser utilizado para garantizar la seguridad de extremo a extremo en las redes multi-hop [29].

Este protocolo admite el uso de claves públicas previamente compartidas y sin formato, un método que es más liviano (en el sentido de cómputo) que otros protocolos de establecimiento de claves, pero tiene como desventaja que no es escalable cuando se trata de una red con gran cantidad de dispositivos. Para resolver esta carencia, se puede hacer uso de un sistema de gestión de claves simétricas que sea escalable según el número de dispositivos del ecosistema IoT en el que se desea implantar [30].

5.2. ECC

La criptografía de curvas elípticas (Eliptical Curve Criptrography, ECC de ahora en adelante), es una técnica de cifrado de clave pública que puede utilizarse en sistemas integrados en tiempo real con recursos limitados en IoT.

Se basa en una estructura algebraica de curvas elípticas sobre campos finitos. Demuestra ser una técnica criptográfica poderosa, ya que se requiere menos tiempo de computación y proporciona el mismo nivel de seguridad. El tamaño de la clave en el caso de ECC es pequeño y rápido en comparación a RSA, lo que lo convierte en una opción popular para proporcionar autenticación en redes inalámbricas [31].

Además de las características anteriormente descritas, ECC también se puede utilizar con RFID para proporcionar seguridad [32] y puede extenderse a ECDSA, ECDH para proporcionar aún más características de seguridad a la red.

5.3. HMAC

El código de autenticación de mensaje (Message Authentication Code, o MAC) se usa para autenticar un mensaje y es generado a partir del mensaje y una clave secreta.

HMAC, por su parte, es también un MAC que utiliza una función hash. La idea de esta solución es la concatenación del mensaje y la clave para posteriormente hashear la cadena resultante de esa concatenación. Se utiliza para proporcionar autenticación e integridad al mensaje que se está transfiriendo, pero no puede hacerlo por si sólo, por lo que debe combinarse con un protocolo. HMAC se puede usar con RFID para mejorar la seguridad y proteger la red contra varias amenazas de seguridad como DoS, escuchas ilegales y ataques de repetición [33].

5.4. Blockchain

La tecnología Blockchain fue creada inicialmente con el objetivo de resolver el problema del doble gasto en Bitcoin, pero debido al gran uso de la criptografía, también se puede usar en otras áreas.

Blockchain utiliza un enfoque descentralizado donde se obtiene la misma funcionalidad y la misma cantidad de certeza sin la autoridad central. La tecnología Blockchain también se puede utilizar para proporcionar seguridad a la red IoT [34]. Los enfoques Blockchain y Peer to Peer también pueden aprovecharse para un ecosistema IoT de diseño privado donde los datos producidos por los dispositivos no se confían a compañías centralizadas, sino que son propiedad del propietario de los dispositivos, que puede decidir qué datos se compartirán y con quién [35].

5.5 VPN

Una VPN (Virtual Private Network, en Inglés) es una tecnología que permite ampliar de forma segura una red LAN a una red pública o no controlada como puede ser Internet [36].

Con una VPN se lleva a cabo una comunicación transparente entre un dispositivo o red fuera de la LAN y que virtualmente aparecerá conectada a ella mediante un túnel cifrado a través del que viajará el tráfico. En este túnel se lleva a cabo el cifrado o el descifrado de los paquetes de información de la comunicación al entrar o salir de un punto del túnel, ocurriendo lo mismo al otro extremo del túnel. Esta será la tecnología que se propondrá utilizar para asegurar las comunicaciones desde una red externa hacia la red de la empresa. El motivo por el cual se ha elegido esta tecnología es por la sencillez de su implantación y porque resulta ser una solución económica, lo que resulta ideal para una PYME.

6. Propuestas de medidas de seguridad y su implantación para un ecosistema IoT

Tras haber llevado a cabo los análisis de las posibles vulnerabilidades y amenazas a las que un ecosistema IoT se puede ver expuesto, se va a realizar una propuesta de implementación segura del ecosistema IoT. Tal y como se especificó en la introducción de este proyecto, el objetivo final de este es el de implementar de forma segura un ecosistema IoT de seguridad para una Pyme.

Para la propuesta, supongamos que la PYME cuenta con uno o varios dispositivos de control de los dispositivos IoT (un dispositivo de control puede ser un móvil o un ordenador), y como dispositivos IoT pertenecientes al ecosistema cámaras WiFi, alarmas y una cerradura inteligente controlada mediante una aplicación móvil.

Como se ha visto a lo largo del desarrollo de esta memoria, un ecosistema IoT puede sufrir problemas de seguridad por los siguientes motivos:

- Una mala gestión de los permisos de los usuarios con acceso al ecosistema y del control de los accesos que se lleva a cabo sobre estos.

- Malas configuraciones en los routers y dispositivos de gestión de la red a la que el ecosistema se encuentra conectado, conllevando con ello la posibilidad de exponer sin quererlo vulnerabilidades tanto de los dispositivos componentes del ecosistema como del propio router/dispositivo de gestión.

- Una seguridad física insuficiente de los diferentes dispositivos implantados ante un posible intruso presencial.

- Configuración insuficiente y/o inadecuada de los dispositivos del ecosistema, como pueden ser contraseñas débiles y por defecto o firmwares de los dispositivos sin actualizar.

A continuación, se desglosarán en los diferentes apartados las propuestas, tanto a nivel de infraestructura, como a nivel de comunicaciones y tecnológico.

6.1. Propuestas para securizar los accesos y los datos transmitidos

La autenticación juega numerosos roles dentro del ecosistema de IoT. Cuando un usuario inicia sesión en un sistema, se está autenticando en ese sistema. Como parte de este proceso, la autenticación proporciona controles de acceso que determinan lo que el usuario autenticado puede hacer en el sistema de destino. Los métodos de autenticación, como los inicios de sesión de los usuarios, se basan en un secreto compartido, algo que tanto el usuario como el sistema conocen. En este modelo, se tiene un nombre de usuario y una contraseña, y se debe proporcionar correctamente ambos para poder autenticarse correctamente. El proceso es similar en los modelos de máquina a máquina (M2M), donde un dispositivo de IoT puede autenticarse y conectarse a

una puerta de enlace para transferir datos o actualizar el firmware o la configuración.

Si bien esta forma de autenticar los usuarios es eficaz, las contraseñas estáticas pueden ser problemáticas, y no se recomiendan para los sistemas de producción (aunque son predominantes en los sistemas de consumo como las cámaras web, por ejemplo). Las contraseñas débiles o predeterminadas de los dispositivos presentan una vulnerabilidad real que permite a los usuarios malintencionados acceder con relativa facilidad a los dispositivos.

Como solución a la problemática que pueden suponer las contraseñas, se pueden implantar políticas para la gestión de los accesos rigurosas y bien establecidas, y como sistema de autenticación una infraestructura de clave pública (PKI, Public Key Infrastructure en inglés, de ahora en adelante).

6.1.1. Políticas para los accesos

Tal y como se comentaba anteriormente, un punto importante para la securización del ecosistema es tener bien controlado el acceso a los dispositivos, de modo que ninguna persona no deseada tenga acceso a los dispositivos implantados y, en caso de que aún con las políticas dicho tercero consiguiese acceder al sistema, poder tener una capacidad de respuesta lo suficientemente sólida, rápida y eficaz como para denegarle el acceso.

Las políticas que nos pueden permitir conseguir esto son:

- Limitar el número de intentos de acceso a las interfaces web de los dispositivos a un máximo de 3 veces, teniendo que recurrir al servicio técnico en caso de querer reactivar una cuenta bloqueada. Con esta medida se pueden evitar los ataques de fuerza bruta que tengan como objetivo adivinar la contraseña de un usuario.

- En caso de que los accesos se lleven a cabo mediante contraseñas (en el próximo apartado se propondrá una alternativa a este tipo de acceso), establecer como longitud mínima de la contraseña 8 caracteres y que estos caracteres sean alfanuméricos. Con esta medida evitamos de nuevo un ataque por fuerza bruta y por rainbow tables.

- Establecer un tiempo máximo de validez de las contraseñas/certificados que permiten el acceso al sistema, obligando al usuario a renovarlos cuando ese plazo se vea cumplido. En el caso de las contraseñas, no se debería permitir que la nueva contraseña sea la misma que al menos las 4 anteriormente utilizadas , y en el caso del uso de certificados el usuario se verá obligado a solicitar una renovación. El período de revocación será de 3 meses, y con ello se consigue una mayor robustez del sistema, ya que en caso de una filtración de las credenciales, éstas acabarían caducando en caso de no haber detectado su sustracción.

- Cerrar automáticamente las sesiones de los usuarios tras 10 minutos de inactividad. Con esta medida, se puede impedir que si una persona deja

su sesión abierta y abandona el puesto en el que se encuentra el dispositivo de control desde el que se está accediendo, su sesión quede eternamente abierta y un tercero no deseado pueda aprovechar esta situación.

Con estas políticas tendríamos un sistema de accesos robusto. Una forma de aumentar aún más la robustez de los accesos es la implementación de un sistema de autenticación de doble factor para ciertos accesos, como los accesos a algunos servicios críticos de la empresa como el área de administración de los sistemas.

6.1.2. Autenticación de doble factor

La autenticación de doble factor consiste en la adición de una capa extra de seguridad en los accesos de manera que en el momento de la autenticación se pueda demostrar que el usuario es quien dice ser (cumpliendo con ello el principio de confidencialidad). Esta capa consiste en, que cuando el usuario ingresa la clave de acceso de su cuenta para el servicio al que desea acceder, debe certificar su identidad con diferente métodos o medidas de seguridad [37]:

- Algún objeto físico en posesión del usuario, como una memoria USB con un identificador único, una tarjeta de coordenadas (este método se utiliza en muchas entidades bancarias), aplicaciones de autentificación, tokens de seguridad con o sin conexión… En este caso el método de autentificación más habitual es la generación de un código único y temporal que sirve para acreditar nuestra identidad.

- Algún secreto conocido por el usuario, como una contraseña extra, un código pin, una pregunta de control que se activa una vez que la contraseña es correcta.

- Mediante la biométrica propia del usuario (la huella dactilar, reconocimiento de iris o de voz, reconocimiento facial) que sirve para desbloquear el acceso.

La solución que se propone para utilizar en los accesos a los servicios web del ecosistema implantado es la de un código autogenerado que se mande a un dispositivo físico que la empresa proporciona a los empleados autorizados a acceder a los susodichos servicios. Si bien el uso de un dispositivo físico puede suponer otros problemas de seguridad derivados, como la filtración de información por una vulnerabilidad del dispositivo o por le robo de este, si se protegen de forma adecuada los mensajes y se lleva un buen control de los dispositivos como no permitir que se puedan sacar de la oficina, que se tenga un registro de cada una de las personas que lo utilizan y que se pueda invalidar el funcionamiento de estos a merced, por ejemplo, ofrece los siguientes beneficios:

- Aunque un cibercriminal consiga obtener el nombre de usuario de uno de los usuarios del sistema, no podrá acceder con la cuenta porque no dispone del código generado en el dispositivo físico del usuario.

- Hace imposible un acceso al sistema mediante ataques de fuerza bruta.
- Dificulta en gran medida las intrusiones de personas no deseadas en el sistema.

Pese a sus beneficios, la autenticación de doble factor no resulta ser un sistema infalible, ya que existe malware como Hesperbot que está especializado en burlar este tipo de sistema engañando a los usuarios para que descarguen una aplicación falsa en lugar de la real a través de la cual el cibercriminal consigue obtener el código de acceso generado para el usuario. Sin embargo, pese a esta problemática con unas buenas políticas de seguridad empresariales resulta ser un mecanismo robusto y seguro de controlar y defender de cibercriminales los accesos de los usuarios.

A continuación se propondrá el uso de un sistema de autenticación basado en PKI.

6.1.3. Infraestructura de clave pública

Una PKI se trata de un sistema criptográfico en el que existen un par de claves para llevar a cabo las funciones criptográficas. En la siguiente figura podemos observar el funcionamiento de estas claves:

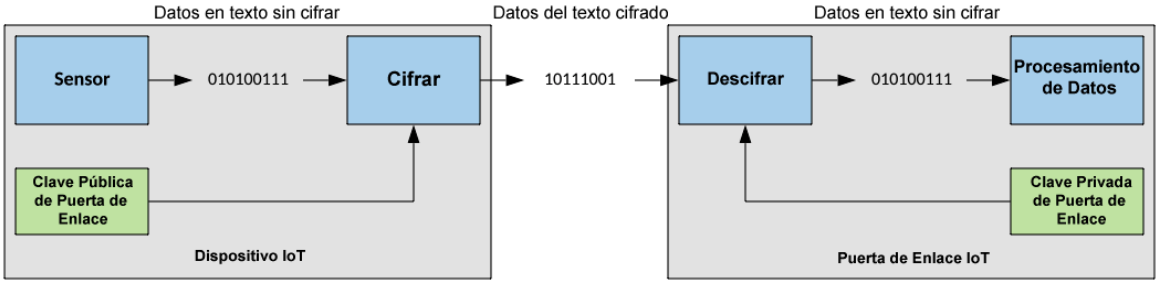


Figura 12 Funcionamiento de la criptografía de clave pública

La clave pública, como su nombre indica, no es un secreto y puede ser por tanto utilizada por cualquier dispositivo para cifrar los datos (en el ejemplo mostrado en la figura 13, la puerta de enlace). Por otro lado, la clave privada es un secreto que debe ser conocido únicamente por el dispositivo receptor de la información y se puede utilizar para descifrar los datos que han sido cifrados con la clave pública pareja de la susodicha clave privada. Al llevar a cabo un cifrado de los datos antes de la comunicación, ningún tercero que capture esos datos mediante un ataque man in the middle podrá obtener información que le sea útil. Con esto conseguiríamos que si un dispositivo malicioso intentara falsear su identidad como si fuese una puerta de enlace para los dispositivos IoT del ecosistema, podría recibir los datos sin problema pero no podría hacer uso de ellos al no tener la clave privada pareja de la clave pública que los cifró.

La configuración estática de claves en los dispositivos de un ecosistema puede ser problemática, además de poder crear problemas de seguridad, sobre todo en ecosistemas con un gran número de dispositivos. Para evitar los problemas

de seguridad que se puedan ocasionar mediante la creación y asignación de las claves, una PKI puede ayudar automatizando la construcción, difusión y revocación de las claves.

Existen una gran variedad de PKIs, pero a un nivel alto, una PKI se encarga de vincular claves públicas a los dispositivos (como los dispositivos IoT del ecosistema) a través de una autoridad de certificación (CA, de ahora en adelante). Este enlace tiene dos propósitos:

1. Permite que un dispositivo IoT cifre los datos que va a mandar para que estos estén protegidos mientras son transferidos, lo que es altamente deseable, sobre todo si esos datos van a ser transferidos por Internet.

2. El proceso de enlace permite la autenticación de un dispositivo a la puerta de enlace IoT, ya que cada clave pública esta vinculada a ciertos dispositivos específicos (todo esto a través de un proceso seguro que involucra una clave pública para la CA).

Gracias al uso de una PKI, a medida que el ecosistema IoT implantado escala en tamaño, aumenta la seguridad de este y limita el esfuerzo humano requerido para la administración de su seguridad. Como se puede observar mediante el análisis llevado a cabo, la PKI no sólo favorece la autenticación de los dispositivos (e incluso de los usuarios, si el acceso externo a los diferentes componentes del ecosistema se lleva a cabo también mediante el par de claves), si no que asegura los datos transferidos en caso de que estos sean filtrados y obtenidos por un tercero no deseado.

Por otro lado, si bien este sistema resulta seguro y eficaz, ante una filtración de la clave privada de nada serviría el cifrado de la información por parte de los dispositivos emisores, ya que el receptor maligno podrá descifrarla al tener la clave privada en su poder. Por eso, es importante seguir buenas prácticas en el almacenamiento de la clave privada, como pueden ser:

- Control de accesos a las puertas de enlace que almacenan las claves privadas mediante un archivo de log de accesos.

- Proteger el fichero en el que se almacene la clave privada mediante contraseña y permisos de usuario.

- Enlazando con el punto anterior, establecer un sistema de roles y permisos para los usuarios del ecosistema para que sólo ciertas personas con cierto rol asociado puedan acceder a la información sensible.

En el ecosistema propuesto, se pueden combinar el uso de factores de doble autenticación para los accesos a los servicios críticos de la empresa y por otro lado la infraestructura PKI para la securización de los datos que son transmitidos entre todos los dispositivos. Con ambas medidas, se tendrá un sistema robusto en el que a priori la autenticación de los usuarios está controlada de forma segura y los datos transmitidos se encuentran protegidos

ante un tercero no deseado, ya que se tiene control sobre quién manda la información y quién puede leerla.

Una vez planteadas las propuestas de la securización de los accesos y los datos, se propondrá una posible solución para la securización de las comunicaciones del ecosistema, sobre todo desde el punto de vista de un acceso externo a la LAN de la empresa.

6.2. Propuesta de securización de las comunicaciones en el ecosistema

Una primera aproximación para tener comunicaciones totalmente cifradas entre los dispositivos de control y los dispositivos IoT podría ser el uso del protocolo HTTPS, generando por ejemplo nuestra propia entidad de confianza e instalando los certificados en nuestros dispositivos IoT o instalando certificados de una entidad certificadora ya existente. Una herramienta que nos permitiría llevar a cabo esta tarea de forma sencilla es Let's Encrypt [38], una Autoridad Certificadora totalmente fiable, de uso gratuito y de código libre, que automatiza los procesos de instalación de los certificados. Con la implementación de esta solución podríamos asegurar las comunicaciones de manera individual a cada dispositivo IoT, ya sea desde la propia LAN de la empresa, como desde Internet mediante un acceso externo, especialmente si las comunicaciones se llevan a cabo estando el dispositivo de control conectado a una red no segura (como lo puede ser la red WiFi de un hotel o un aeropuerto).

Aunque la implementación de HTTPS debería de ser un requisito general para los dispositivos IoT, no resulta práctico abrir el acceso a los dispositivos desde fuera de la LAN mediante la apertura de puertos en los dispositivos IoT, ya que esto podría conllevar a que cualquiera pudiese acceder a ellos desde Internet para intentar explotar cualquier vulnerabilidad, además de que muchos dispositivos no admiten HTTPS.

Por estos motivos, otra solución para la securización de las comunicaciones en nuestro ecosistema es el uso combinado de sistemas de cifrado adicionales junto a HTTPS, haciendo con esto que el usuario pueda conectarse a la red de la empresa para manejar los dispositivos IoT sin que otros usuarios de Internet puedan siquiera conocer su existencia a priori. Es en este contexto donde es una buena opción la implementación de una red privada virtual (VPN, de ahora en adelante).

6.2.1. Uso de una VPN para securizar los accesos externos

Con una VPN se lleva a cabo una comunicación transparente entre un dispositivo o red fuera de la LAN y que virtualmente aparecerá conectada a ella mediante un túnel cifrado a través del que viajará el tráfico. En este túnel se lleva a cabo el cifrado o el descifrado de los paquetes de información de la comunicación al entrar o salir de un punto del túnel, ocurriendo lo mismo al otro extremo del túnel.

En el siguiente diagrama se puede apreciar de manera gráfica el funcionamiento de una VPN:

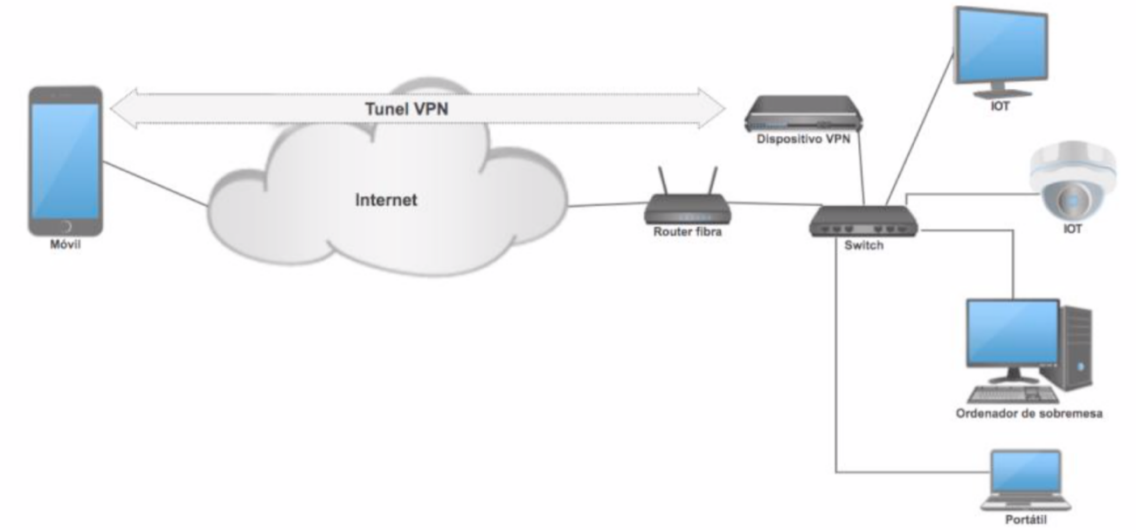


Figura 13 Ejemplo de una conexión VPN a través de un móvil utilizado como dispositivo de control

Desde el punto de vista técnico, el túnel creado para la comunicación es una encapsulación de un protocolo sobre otro, en que la información se cifra y cuando llega al destino se desencapsula el protocolo que encapsuló y cifró el paquete original y librea el paquete en la red de destino para que pueda ser leído.

Gracias a las VPNs podemos acceder desde nuestro dispositivo de control a nuestra red LAN de la empresa sin necesidad de estar conectado a ella como si estuviésemos físicamente conectados. Además, si llevamos a cabo la conexión desde cualquier red no controlada (Internet en general) donde un usuario malicioso estuviese esnifando el tráfico de la red, obtendrá únicamente tráfico ilegible. En el anexo 9.1, se analiza cómo se puede llevar a cabo el montaje de una VPN propia en la PYME además de profundizar en mayor medida en los beneficios a nivel de seguridad que ofrece para las comunicaciones.

Desde el punto de vista de la encapsulación, existen diversos protocolos de encapsulación que se pueden utilizar para la tunelización de la VPN, entre los que cabe destacar los protocolos PPPTP, SSTP y L2TP. Cada uno de estos protocolos exige ciertos requisitos del sistema y tiene sus propias fortalezas y debilidades, como se verá a continuación [39].

6.2.1.1. Protocolo PPPTP

El protocolo de túnel punto a punto (Poin-toPoint Tunneling Protocol, PPPTP), admite multiprotocolo para ser cifrado y encapsula el tráfico dentro de paquetes IP. PPPTP hace uso de la misma autenticación que PPP (Protocolo punto a punto). En la siguiente figura se puede ver el diagrama de representación de un paquete encapsulado con PPPTP:

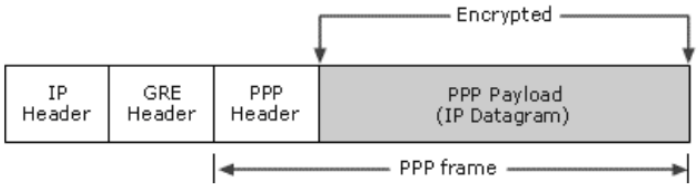


Figura 14 Paquete encapsulado con PPPTP

Su fuerza depende de la fuerza de la contraseña que se utiliza para la autenticación para proporcionar seguridad, y solo tiene la capacidad de cifrar datos a lo largo de una clave de 128 bits, por lo que no garantiza una buena seguridad, por lo que no será el protocolo que se usará en la VPN a implantar en el ecosistema propuesto.

6.2.1.2. Protocolo SSTP

El protocolo Secure Socket Tunneling Protocol se trata de un protocolo de túnel que utiliza el protocolo HTTPS (HTTP Secured) a través del puerto 443 de la conexión TCP al tráfico de tránsito a través de proxies y firewalls que pueden bloquear el tráfico PPTP y L2TP. Facilita el mecanismo para encerrar el tráfico PPP a través del protocolo SSL (Secure Socket Layer) que admite TLS (Transport Layer Security) integrado con el intercambio de claves improvisado, el cifrado, la confidencialidad de los datos, la integridad de los datos y la autenticación. Fue creado para admitir clientes remotos, por lo general no tiene la capacidad de admitir túneles VPN de sitio a sitio. Por lo general, el rendimiento de este protocolo es justo hasta que se presenta un exceso de ancho de banda suficiente en la red subyacente, en cuyo caso no resulta una buena opción.

6.2.1.3. Protocolo L2TP

El protocolo de túnel de capa 2 (Layer 2 Tunneling Protocol, L2TP de ahora en adelante) integra las características del protocolo PPPTP con el protocolo L2F (reenvío de capa 2, Layer 2 Forwanding en inglés), desarrollado por CISCO Systems. La tunelización se realiza mediante el uso de múltiples niveles de encapsulación, que son L2TP, UDP (User Datagram Protocol), IPSec (IP Security), IP (Internet Protocol) y Data-Link, en los que IPSec sirve para el cifrado de los túneles L2TP. En este protocolo se lleva a cabo la encapsulación de los paquetes en dos capas:

1. **Primera capa: Encapsulación L2TP**

Consiste en un marco PPP encerrado con el encabezado de L2TP y UDP. En la siguiente figura podemos ver el resultado de un paquete encapsulado en esta capa:

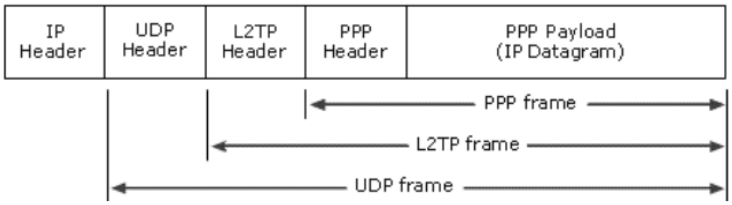


Figura 15 Paquete encapsulado en la primera capa de L2TP

2. Segunda capa: Encapsulación IPSec

En esta capa el mensaje resultante de la capa anterior se adjunta además con un encabezado IPsec. IPsec está diseñado para especificar la seguridad entre el canal de comunicación de dos dispositivos de comunicación, tales como puertas de enlace, routres y cortafuegos. IPsec es un protocolo desarrollado para proteger las comunicaciones de protocolo IP mediante el cifrado y la autenticación de cada paquete IP durante la sesión. La arquitectura tradicional de IPv4 no está diseñada para IPSec, mientras que es una característica integrada de IPv6. IPSec proporciona dos protocolos de seguridad [40]:

1. El encabezado de autenticación (AH, Authentication Header) que asegura las direcciones de origen y destino del encabezado IP mediante el uso de una función de hash con una clave secreta.

2. La carga útil de seguridad encapsulada (ESP, Encapsulated Security Payload) que proporciona integridad, confidencialidad y autenticación, y permite el cifrado del payload, lo que garantiza la integridad de los datos y la confidencialidad de los mismos.

En la siguiente figura se puede apreciar un paquete encapsulado por esta capa:

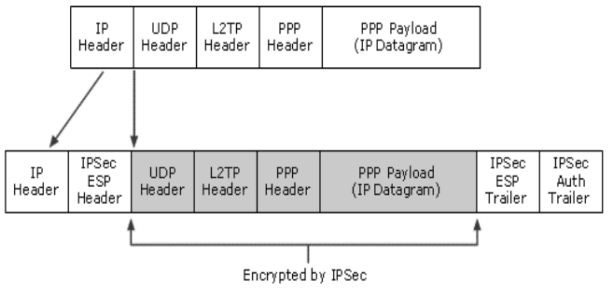


Figura 16 Paquete completamente encapsulado de L2TP

Este protocolo presenta como principal desventaja el coste computacional de la doble encapsulación de los datos, que puede llegar a provocar una velocidad de conexión relativamente baja. A pesar de esta desventaja, es el protocolo que se propone utilizar para la VPN del ecosistema, ya que la seguridad que proporciona es incomparable a cualquier otro protocolo VPN, algo altamente deseable en la transmisión de datos con información sensible en su interior.

7. Conclusiones y trabajo futuro

A través del desarrollo de este TFM se han obtenido las siguientes conclusiones:

1. Se ha obtenido conocimiento del contexto actual del IoT y los pronósticos de uso y crecimiento en los años venideros. Tal y como se vio en la introducción de este proyecto el IoT está en auge y a cada año el número de dispositivos IoT en funcionamiento aumenta a pasos agigantados.

2. Los dispositivos y ecosistemas IoT que se utilizan e implantan pueden presentar graves problemas de seguridad, a veces propiciados por malas prácticas del propio usuario como por parte de la empresa que suministra los dispositivos o servicios, al no ofrecerles un soporte continuado o no haber desarrollado la tecnología con el objetivo de que sea segura.

3. Se ha aprendido cuáles son las amenazas y vulnerabilidades más comunes de un ecosistema IoT. A través de los diferentes análisis llevados a cabo en el desarrollo del proyecto se ha aprendido a cómo se pueden presentar estos peligros y las formas de solventarlos o de minimizar los problemas derivados de ellos en caso de no poder solventarlos.

4. Los dispositivos IoT de nuestro ecosistema no son el único punto de entrada para las acciones de los cibercriminales. Los propios routers pueden presentar una puerta de entrada a la LAN de nuestra empresa debido a malas configuraciones, puertos abiertos sin protección y en ocasiones vulnerabilidades presentes en el propio software interno del router. Por ello, es importante no tener abiertos puertos que no sean necesarios y mantener siempre actualizado el software del/de los router.

5. Con el establecimiento de unas buenas políticas de accesos y el uso de una infraestructura PKI se puede conseguir de forma relativamente sencilla y económica la securización y el control de los accesos de los usuarios y los datos que son transferidos en nuestro ecosistema IoT.

6. Una VPN es una buena solución para la securización desde el punto de vista del acceso desde un dispositivo de control que se encuentra conectado a una red externa (Internet en general) a nuestra LAN. Gracias a su uso junto al protocolo L2TP presenta una gran opción para cifrar las comunicaciones, ya que el cifrado de doble capa que se aplica con L2TP, si bien puede acarrear un mayor coste computacional para la conexión, resulta ser un método muy seguro que impide un tercero que intercepte la conexión VPN entre el dispositivo de control y nuestra red pueda obtener información útil de los paquetes que haya conseguido capturar.

7. Se ha obtenido un punto de vista profesional y analítico de la proposición, planteamiento y resolución de problemas de ciberseguridad y la importancia que tiene proteger nuestros dispositivos y datos de personas no deseadas.

Como trabajo futuro se podría implementar físicamente el ecosistema planteado así como las medidas y protocolos propuestos para su securización, lo que conllevaría con el desarrollo de un presupuesto así como de un análisis de qué dispositivos IoT en concreto (en lo que a marcas, modelos, etc… se refiere) se van a implantar y por qué.

8. Glosario

OWASP: Es una organización benéfica sin fines de lucro centrada en mejorar la seguridad del software. Su misión es hacer visible la seguridad del software, para que las personas y las organizaciones puedan tomar decisiones informadas.

Cross-site scripting: Es una técnica que consiste en ser capaz de inyectar código JavaScript en una aplicación web con el objetivo de llevar a cabo acciones maliciosas en la misma mediante la ejecución por parte del cliente de ese código [41].

Fuzzers: Un fuzzer es un programa que inyecta automáticamente datos semi-aleatorios en un programa y detecta errores [42].

NAT o Network Address Translation: es un mecanismo utilizado por routers IP para intercambiar paquetes entre dos redes que asignan mutuamente direcciones incompatibles. Consiste en convertir, en tiempo real, las direcciones utilizadas en los paquetes transportados.

DNS o Domain Name System: Es un sistema que permite traducir nombres por IPs con el fin de evitar recordar complejos números. Los servidores DNS se encargan de traducir un nombre de dominio en una IP.

Phising: es un ataque de ingeniería social cuyo objetivo final es el de obtener datos de una víctima con una finalidad fraudulenta. Para conseguirlo, el atacante suplanta la identidad de una entidad de confianza y convence a un usuario para realizar una acción que le permite captar datos de interés.

Modelo OSI: modelo de interconexión de sistemas abiertos (ISO/IEC 7498-1), más conocido como “modelo OSI”, (en inglés, Open System Interconnection) es un modelo de referencia para los protocolos de la red (no es una arquitectura de red), creado en el año 1980 por la Organización Internacional de Normalización (ISO).

Hardening: es un término utilizado en seguridad informática que se refiera al proceso de asegurar un sistema mediante la reducción de vulnerabilidades en el mismo. Esto se logra eliminando software, servicios, usuarios, etc; innecesarios en el sistema; así como cerrando puertos que tampoco estén en uso además de muchos otros métodos.

9. Bibliografía

[1] Höller, Jan & Tsiatsis, Vlasios & Mulligan, Catherine & Karnouskos, Stamatis & Avesand, Stefan & Boyle, David. (2014). From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence. 10.1016/B978-0-12-407684-6.00001-2.

[2] Clemares, L. (2019). La tecnología IoT irrumpe con fuerza en la agricultura y ganadería. Recuperado de:

https://www.tendencias21.net/telefonica/La-tecnologia-IoT-irrumpe-con-fuerza-en-la-agricultura-y-ganaderia_a2105.html

[3] How the Internet of Things impacts marketing. (2019). Recuperado de:

https://www.i-scoop.eu/how-the-internet-of-things-impacts-marketing/

[4] Sanchez, L. et al. (2014). SmartSantander: IoT experimentation over a smart city testbed. *Computer Networks*, *61*, 217–238. DOI: https://doi.org/10.1016/j.bjp.2013.12.020. Recuperado de:

https://repositorio.unican.es/xmlui/bitstream/handle/10902/9926/SmartSantanderIoT.pdf;jsessionid=7044B7DAC2EF5B0D2AA98AF3B6DCA1A3?sequence=3

[5] State of the IoT 2018: Number of IoT devices now at 7B – Market accelerating. (2019). Recuperado de:

https://iot-analytics.com/state-of-the-iot-update-q1-q2-2018-number-of-iot-devices-now-7b/

[6] ¿Qué es un botnet? – Kaspersky Daily. Recuperado de:

https://www.kaspersky.es/blog/que-es-un-botnet/755/

[7] Mucientes, E. (2019). Así se gestó el ciberataque más grave de los últimos 10 años. Recuperado de:

https://www.elmundo.es/tecnologia/2016/10/22/580b10e5268e3e06158b45e0.html

[8] Fruhlinger J., 2018. *The Mirai botnet explained: How teen scammers and CCTV cameras almost brought down the internet.* Recuperado de:

https://www.csoonline.com/article/3258748/the-mirai-botnet-explained-how-teen-scammers-and-cctv-cameras-almost-brought-down-the-internet.html

[9] DDoS Attack Protection Solutions & Services | Neustar. (2019). Recuperado de:

https://www.home.neustar/ddos-protection?_ga=2.235505605.598927519.1546113049-2028372576.1546113049v

[10] Howe, S. (2018). Why IoT could be the next ransomware target. Recuperado de:

https://www.cso.com.au/article/645755/why-iot-could-next-ransomware-target/

[11] Arduino - Introduction. (2019). Recuperado de:

https://www.arduino.cc/en/Guide/Introduction

[12] Tabla comparativa de Arduinos. (2019). Recuperado de:

http://sabetecnologia.blogspot.com/2013/03/tabla-comparativa-de-arduinos.html

[13] Waspmote Technical Guide. (2019). Recuperado de:

http://www.libelium.com/downloads/documentation/waspmote_technical_guide.pdf

[14] Smart City project in Castellón: a platform to control water usage and waste management | Libelium. (2019). Recuperado de:

http://www.libelium.com/smart-city-project-in-castellon-a-platform-to-control-water-usage-and-waste-management/

[15] Introducción a las Placas Intel® Galileo. (2019). Recuperado de:

https://www.intel.es/content/www/es/es/support/articles/000005912/boards-and-kits/intel-galileo-boards.html

[16] Raspberry Pi. Recuperado de:

https://www.raspberrypi.org/

[17] Modelos y características de Raspberry Pi. (2019). Recuperado de:

https://www.luisllamas.es/modelos-de-raspberry-pi/

[18] ThingSpeakDocumentation- MathWorks España. (2019). Recuperado de:

https://www.mathworks.com/help/thingspeak/

[19] Electric Imp Secure IoT Connectivity Platform. (2019). Recuperado de:

https://www.electricimp.com/

[20] Platform Overview | Dev Center. (2019). Recuperado de:

https://developer.electricimp.com/platform-overview

[21] Internet de las cosas | Plataforma como servicio | AWS IoT. (2019). Recuperado de:

https://aws.amazon.com/en/iot/

[22] Google Cloud IoT - Fully managed IoT services | Google Cloud. (2019). Recuperado de:

https://cloud.google.com/solutions/iot/

[23] The hidden Role of IoT in Cyber Attacks, n.d. Recuperado de:

https://info.vectranetworks.com/hubfs/Vectra_Networks_IoT_Webinar_Slides.pdf

[24] Top 10 IoT Vulnerabilities (2014) - OWASP. Recuperado de:

https://www.owasp.org/index.php/Top_10_IoT_Vulnerabilities_(2014)

[25] Filet-O-Firewall exposes millions of home routers to attacks. (2019). Recuperado de:

http://web.archive.org/web/20160208233326/http:/www.filet-o-firewall.com/

[26] Filet-o-Firewall: new vulnerabilities in UPnP expose the whole network. (2019). Recuperado de:

https://www.kaspersky.com/blog/filet-o-firewall/4533/

[27] DNSChanger vuelve, así funciona este malware para routers. (2019). Recuperado de:

https://www.adslzone.net/2016/12/17/dnschanger-vuelve-asi-funciona-este-malware-routers/

[28] DNSChanger Exploit Kit Hijacks Routers, Not Browsers. (2019). Recuperado de:

https://threatpost.com/dnschanger-exploit-kit-hijacks-routers-not-browsers/122539/

[29] Rene Roepke, Timo Thraem, Johannes Wagener, and Alex Wiesmaier (2017), ”A Survey on Protocols securing the Internet of Things: DTLS, IPSec and IEEE 802.11i”.

[30] Shahid Raza, Ludwig Seitz, Denis Sitenkov, and Göran Selander, (2015) “S3K: Scalable Security with Symmetric Keys DTLS Key Establishment for the Internet of Things”, IEEE Transactions on Automation Science and Engineering.

[31] Sumanth Koppula, Jayabhaskar Muthukuru,”Secure Digital Signature Scheme Based on Elliptic Curves for Internet of Things” International Journal of Electrical and Computer Engineering (IJECE) Vol. 6, No. 3, June 2016, pp. 1002 ~ 1010 ISSN: 2088-8708, DOI: 10.11591/ijece.v6i3.9420.

[32] Himja Agrawal, Prof. P.R. Badadapure, ”A Survey Paper On Elliptic Curve Cryptography”, International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056 Volume: 03 Issue: 04 | Apr-2016 www.irjet.net p-ISSN: 2395-0072 © 2016.

[33] Abdolmaleki, B., Baghery, K., and Emadi, M. J., 2016, “HMAC-Based Authentication Protocol: Attacks and Improvements”. Amirkabir International Journal of Electrical and Electronics Engineering, 48(2), pp.71-79. DOI:10.22060/eej.2016.817.

[34] Konstantinos Christidis, ”Blockchains and Smart Contracts for the Internet of Things (2016)” Special Section on the Plethora of Research in Internet of Things (IoT).

[35] Marco Conoscenti, Antonio Vetr`o, Juan Carlos De Martin, ”Blockchain for the Internet of Things: a Systematic Literature Review (2016)”.

[36] OpenVPN, OpenVPN How To, S.F. Recuperado de:

https://openvpn.net/community-resources/how-to/

[37] Dos mejor que uno: doble factor para acceder a servicios críticos. (2019). Recuperado de:

https://www.incibe.es/protege-tu-empresa/blog/dos-mejor-uno-doble-factor-acceder-servicios-criticos

[38] Let's Encrypt - Free SSL/TLS Certificates. (2019). Recuperado de:

https://letsencrypt.org/

[39] Singh, Kuwar Kuldeep Veer Vikram & Gupta, Himanshu. (2016). A New Approach for the Security of VPN. 1-5. 10.1145/2905055.2905219.

[40] Wafaa Bou Diab, Samir Tohme, Carole Bassi, VPN Analysis and New Perspective for Securing Voice over VPN Network, Networking and Services, Fourth International Conference, 16-21 March, 2008, 73-78.

[41] Alonso J., Guzmán A., Laguna P., Martín A. (S.F.). *Ataques a aplicaciones web.* Recurso educativo del Máster interuniversitario en seguridad de las tecnologías de la información y las comunicaciones.

[42] Fuzzing - OWASP. Recuperado de:

https://www.owasp.org/index.php/Fuzzing

10. Anexos

10.1 Montaje de una VPN propia

Para el montaje de una VPN es necesario tener un dispositivo que actúe como servidor VPN. En el mercado existen opciones muy económicas, como la Raspberry Pi (que en parte se trata también de un dispositivo IoT de nuestro ecosistema) que se analizó en el capítulo 2.6.4 de esta memoria. Si bien esta plataforma está algo limitada en especificaciones hardware, para el caso propuesto de una PYME mediana sus capacidades son más que suficientes.

Como primer paso en la Raspberry Pi se puede instalar el sistema operativo Raspbian, que se trata de una distribución basada en el sistema operativo Linux Debian.

Una buena opción de software para VPN que resulta ser práctico y seguro es OpenVPN. Este software puede ser implementado en las capas 2 o 3 del modelo OSI, y entre algunas de sus funcionalidades está la posibilidad de scripting, diseño modular y la creación de interfaces virtuales para poder implementar reglas de firewall específicas, por ejemplo, con iptables.

Como método de securización de las transmisiones de los datos se hará uso del sistema de clave pública descrito en el apartado 6.1 de esta memoria, mediante el uso de un par de claves para cada uno de los usuarios con acceso a la red. De esta forma, tras y como se describió en las características de este tipo de cifrado, usando la clave privada del propio usuario se puede desencriptar todo el tráfico entrante y con la clave pública del otro extremo del túnel puede encriptar información que a su vez el destino podrá desencriptar con su propia clave privada.

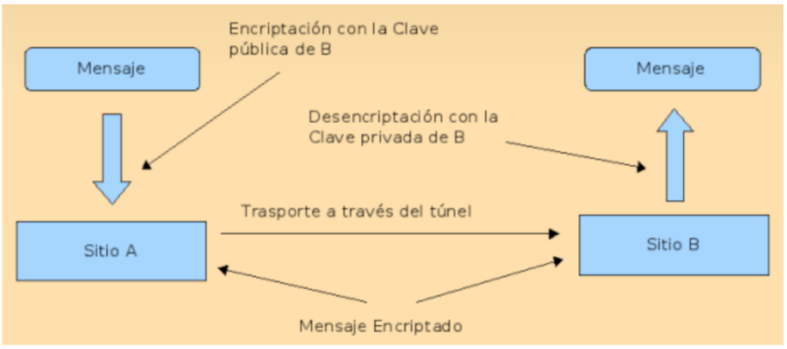


Figura 17 Diagrama de cifrado asimétrico SSL/TLS en OpenVPN

Con estas premisas, la configuración de OpenVPN en modo cifrado asimétrico con SSL/TLS necesitará:

- Una autoridad de certificación (CA, de ahora en adelante) que se encargará de emitir los certificados de los clientes que puedan conectarse a la VPN y cuya clave pública del CA podrá ser validada por todos los certificados de los clientes.

- Uso de las claves privadas para los clientes o usuarios, emitidas por la CA. Es deseable que el par de claves se protejan con contraseña para permitir su uso.

- Un sistema de revocación, con el cual se pueda dar de baja la clave privada de los clientes, aunque ésta no haya caducado. De esta forma, si se sufre un robo o filtración de la clave privada, o si simplemente no se desea que esta clave siga estando en vigor, se podrá revocar desde el propio OpenVPN.

Si llevamos a cabo un proceso de hardening sobre OpenVPN, según se explica en la propia documentación oficial de este software [34], se puede implementar una directiva tls-auth cuyo fin es añadir una firma HMAC adicional a todos los paquetes handshake SSL/TLS para integrar la verificación. De esta forma, cualquier paquete UDP que no contenga la firma correcta HMAC será eliminada para evitar su procesamiento y el consumo de recursos que ello conllevaría. Las ventajas de esta implementación son:

- Evitar los ataques de denegación de servicio (DoS) o la inundación de puertos UDP del servicio.

- Evita la exploración de puertos que pueda determinar qué puertos UDP del servidor se encuentran a la escucha.

- Evitar vulnerabilidades de desbordamiento de búfer en la implementación SSL/TLS.

- La posibilidad de denegar rápidamente las iniciaciones de protocolo de enlace SSL/TLS desde una máquina no autorizada.

Aún llevando a cabo el hardening, se debe dejar un puerto UDP abierto en el router de la empresa para que se pueda acceder a él desde nuestros dispositivos de control de forma externa.

Una vez configurada la VPN en OpenVPN, se pueden generar las claves para un cliente, como por ejemplo un teléfono móvil. Tal y como se comentó anteriormente, es altamente deseable asignarle una contraseña a la clave privada en el momento de la creación del par de claves por si esta se ve sustraída por un tercero no deseado. OpenVPN empaquetará la configuración de las claves en un archivo ovpn que incluye:

- Parámetros de configuración: la IP, puerto de conexión y otros parámetros como si todo el tráfico es redirigido al túnel OpenVPN. Estos parámetros son necesarios para poder establecer la comunicación entre el cliente y OpenVPN.

- La clave pública de la CA que emitió el certificado para el cliente.

- La clave pública del cliente.

- La clave privada del cliente.

- La clave del tls-auth que se utilizará para distinguir la conexión como una conexión legítima en el servidor OpenVPN.

Tras obtener el archivo ovpn, debe ser abierto en el dispositivo de control en el que se desea cargar la configuración con OpenVPN, y tras introducir la contraseña con la que se protegió el archivo, el dispositivo quedaría configurado de forma prácticamente automática para conectarse al servidor OpenVPN (la Raspberry Pi), permitiendo con esto una conexión segura desde una red externa a la LAN de la empresa y a todos los dispositivos IoT conectados a ella.

Una buena guía para llevar a cabo la configuración de una Raspberry Pi como un servidor VPN se puede encontrar en el siguiente enlace:

https://www.fwhibbit.es/solucion-de-vpn-basada-en-raspberry-pi