MÁSTER UNIVERSITARIO DE INVESTIGACIÓN EN INGENIERÍA DE SOFTWARE Y SISTEMAS INFORMÁTICOS

ITINERARIO INGENIERÍA DEL SOFTWARE (CÓDIGO 31105151)

TRABAJO FIN DE MASTER

UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA

DISPOSITIVO PARA LA SEGURIDAD DE CÁMARAS IP EN ENTORNO DOMÉSTICO

AUTOR: XABIER ALDAMA SÁNCHEZ
DIRECTOR: DR. ISMAEL ABAD CARDIEL

SEPTIEMBRE 2019-2020

MÁSTER UNIVERSITARIO DE INVESTIGACIÓN EN INGENIERÍA DE SOFTWARE Y SISTEMAS INFORMÁTICOS

Itinerario Ingeniería del Software (Código 31105151)

DISPOSITIVO PARA LA SEGURIDAD DE CÁMARAS IP EN ENTORNO DOMÉSTICO

Tipo de trabajo: B

Autor: Xabier Aldama Sánchez
Director: Dr. Ismael Abad Cardiel

DECLARACIÓN JURADA DE AUTORÍA DEL TRABAJO CIENTÍFICO, PARA LA DEFENSA DEL TRABAJO FIN DE MASTER

Fecha: 11/09/2020

Quién suscribe:

Autor(a): Xabier Aldama Sánchez
D.N.I/N.I.E/Pasaporte.: 45817571 F

Hace constar que es la autor(a) del trabajo:

Dispositivo para la Seguridad de cámaras IP en entorno doméstico

En tal sentido, manifiesto la originalidad de la conceptualización del trabajo, interpretación de datos y la elaboración de las conclusiones, dejando establecido que aquellos aportes intelectuales de otros autores, se han referenciado debidamente en el texto de dicho trabajo.

DECLARACIÓN:

- ✓ Garantizo que el trabajo que remito es un documento original y no ha sido publicado, total ni parcialmente por otros autores, en soporte papel ni en formato digital.
- ✓ Certifico que he contribuido directamente al contenido intelectual de este manuscrito, a la génesis y análisis de sus datos, por lo cual estoy en condiciones de hacerme públicamente responsable de él.
- ✓ No he incurrido en fraude científico, plagio o vicios de autoría; en caso contrario, aceptaré las medidas disciplinarias sancionadoras que correspondan.

Fdo.

Impreso TFdM05_AutorPbl. Autorización de publicación y difusión del TFM para fines académicos

Autorización

Autorizo/amos a la Universidad Nacional de Educación a Distancia a difundir y utilizar, con fines académicos, no comerciales y mencionando expresamente a sus autores, tanto la memoria de este Trabajo Fin de Máster, como el código, la documentación y/o el prototipo desarrollado.

Firma del/los Autor/es

Juan del Rosal, 16
28040, Madrid

Tel: 91 398 89 10
Fax: 91 398 89 09
www.issi.uned.es

Resumen

Las cámaras IP son hoy en día accesorios habituales en los hogares. Las video vigilancia y detección de presencia como argumentos para reforzar la seguridad del hogar son los pilares en los que se ha basado la comercialización de estos dispositivos.

Son muchos los beneficios que aportan, pero suponen a su vez un punto de acceso a los datos de los usuarios que cada vez está siendo más explotado por atacantes.

El hecho de que se lance al mercado una gran variedad de estos dispositivos y con gran rapidez, hace que la seguridad ofrecida por los mismos quede cuanto menos en duda.

En este trabajo se plantea la introducción de un dispositivo adicional para aumentar la seguridad de las cámaras IP conectadas en el hogar. Con él se pretende gestionar la seguridad del hogar desde el punto de vista cibernético mediante un análisis y gestión del tráfico relacionado con estos dispositivos y un aprendizaje asociado al mismo. La propuesta pasa por introducir un dispositivo que desempeña las funciones de servidor Wi-Fi para controlar las comunicaciones. La seguridad se aplica gracias a la monitorización y validación del tráfico de la red en la que operan las cámaras.

Palabras clave: seguridad, cámara IP, protección

Índice de contenidos

Lista de figuras

Lista de tablas

1 Introducción

El concepto de internet de las cosas, en adelante IoT de sus siglas en inglés (IoT), fue acuñado en el MIT y en concreto por Kevin Ashton en 1998 para referirse a la interconexión de dispositivos mediante internet [1, 2, 3, 4]. Básicamente, los dispositivos IoT son dispositivos interconectados mediante una red inalámbrica capaces de obtener y procesar datos y ejecutar acciones en base a decisiones programadas o tomadas por humanos. Son utilizados para respuesta a necesidades cotidianas. Las cámaras IP forman parte de este tipo de dispositivos. Se trata de cámaras que son accesibles mediante conexión de red. Gracias a la posibilidad de acceso remoto facilitan las labores de vigilancia.

Las cámaras como parte hardware recuperan imágenes y sonidos y pueden llegar a ejecutar las acciones necesarias mediante accionadores. Para poder procesar la información obtenida, el dispositivo posee un componente software, llamado firmware. Mediante el firmware se realiza toda la gestión la comunicación y datos del dispositivo, ya sea con otros dispositivos o con humanos. Es fácil imaginar la cantidad de datos obtenidos por las cámaras y que son enviados a través de la red de comunicaciones.

Gracias a la facilidad a la hora de instalar y manejar un sistema de seguridad compuesto por cámaras IP, estos dispositivos se han hecho un sitio en las vidas cotidianas de los consumidores. Gartner pronosticó ya en 2017 que la tendencia de adquisición de dispositivos IoT tomaría una curva exponencialmente ascendente [5]. Según sus datos, se pasará de una cifra cercana a los 11 mil millones de dispositivos instalados en 2018 a superar los 20 mil millones.

Año	2016	2017	2018	2020
Consumidores	3,963.0	5,244.3	7,036.3	12,863.0
Industria de innovación cruzada	1,102.1	1,501.0	2,132.6	4,381.4
Soluciones específicas	1,316.6	1,635.4	2,027.7	3,171.0
Total	6,381.8	8,380.6	11,196.6	20,415.4

Tabla 1. Estimación dispositivos IoT instalados. Fuente: Gartner (Enero 2017)

Las instalaciones de dispositivos de seguridad basados en imágenes han aumentado y convertido en algo habitual[6]. En algunos estudios se estima que el número de las instalaciones de cámaras IP o sistemas de video-vigilancia se encuentran en magnitudes próximas a 245 millones [7].

Al tratarse de una práctica muy extendida la instalación de cámaras IP para aumentar la seguridad en el hogar, parece interesante plantear si el sistema de seguridad instalado es en si seguro o no. Se han realizado varios trabajos en los que se demuestra que estos dispositivos tienen varias vulnerabilidades que pueden ser utilizadas por atacantes para obtener imágenes u otros datos privados, comprometer la seguridad de la red o hacer uso de la misma para realizar otros ataques. Desde el año 2011, se está recopilando una lista de vulnerabilidades asociadas a circuitos cerrados de video-vigilancia (CCTV) [8]. De momento, se han recopilado datos de 440000 usuarios de 247 estados diferentes. Como resultado, existe una base de datos de 2526000 vulnerabilidades asociadas a CCTVs.

Un gran problema desde el punto de vista de la ciber-seguridad en las cámaras IP es el uso de firmware genérico para dispositivos de diferentes proveedores. Este punto del nego-

cio hace que la misma vulnerabilidad se encuentre en múltiples dispositivos de diferentes proveedores.

Brian Cusak y Zhuang Tian demuestran lo sencillo que es detectar cámaras Ip en una red y acceder a ellas [9]. Mediante un escáner identifican los dispositivos y haciendo uso de pares usuario/contraseña conocidos acceden a ellos. Se revela el uso de credenciales conocidas en estos dispositivos de seguridad, lo que significa una gran falla de seguridad.

1.1 Alcance y objetivos

En este trabajo se trata de proponer una solución para dotar de seguridad una instalación de cámaras IP en el entorno doméstico. Se trata de un dispositivo a modo de servidor Wi-Fi en el que se vincularán los dispositivos de video-vigilancia y gestionará la seguridad de los mismos. Debido a su destino doméstico, parte del objetivo es que no se requiera de configuración complicada para que un usuario normal sea capaz de manejar la solución. Se trata en definitiva de un dispositivo que permite al usuario añadir la capa de seguridad necesaria a su instalación de video-vigilancia haciendo uso de un servidor Wi-Fi validador en donde dichos dispositivos son conectados.

1.2 Plan de trabajo

La realización de la propuesta se dividió en cinco fases. La fase inicial fue la de revisión bibliográfica, en la que se detectó la necesidad de añadir seguridad a las instalaciones de video-vigilancia, ya que el número de dispositivos crece y a su vez la cantidad y tipo de ataques. No se encontró ninguna solución que cubriera las necesidades planteadas de segu-

ridad desde la propia instalación, con lo que se procedió a implementar un prototipo base a modo de prueba de concepto en la siguiente fase: implementación de prueba de concepto básica. Esta fase fue clave para determinar el enfoque que se adoptó en la solución. Se comprobó que una placa Raspberry Pi puede funcionar de servidor Wi-Fi y que además puede albergar un proceso de monitorización del tráfico de red que fluye por ella permitiendo así su validación. El siguiente paso fue la de búsqueda de tecnologías aplicables en la propuesta. Con los conocimientos adquiridos, en la cuarta fase, se diseñó e implementó la solución que finalmente consistió en la placa Raspberry Pi a modo de servidor Wi-Fi encargado de monitorizar y validar las conexiones a las cámaras IP registradas. En la fase final se evaluó la solución para determinar que efectivamente cubría las necesidades de seguridad.

1.3 Estructura del documento

La documentación de este trabajo se ha planteado en los siguientes apartados.

- **Análisis del problema**: En este punto se expone el concepto de cámara IP y los protocolos de comunicación utilizados. Se explican además los motivos que justifican la relevancia de la seguridad en este tipo de dispositivos.

- **Estado del arte**: Es el apartado en el que se explican las herramientas existentes que han sido empleadas en la solución o que han sido determinantes a la hora de diseñarla.

- **Propuesta**: El trabajo consiste en el planteamiento de una solución para dotar de seguridad a una instalación de cámaras IP en el hogar. Dicha propuesta queda detallada en este punto.

- **Evaluación de la solución**: Es la parte del documento en donde se exponen las pruebas realizadas a la solución propuesta. Se han realizado pruebas funcionales, de rendimiento y de viabilidad de monitorización de tráfico de red. Se han obtenido resultados que cubren las necesidades planteadas que no son otras que proteger contra ataques de tipo DDoS y accesos no autorizados a la instalación de videovigilancia.

- **Conclusiones y trabajos futuros**: Es el apartado en donde se muestran las conclusiones obtenidas tras la implementación y evaluación del dispositivo propuesto. Se plantea también un posible punto de evolución de la propuesta tras analizar los resultados.

2 Análisis del problema

2.1 Cámaras IP y protocolos de comunicación

Una cámara IP es un dispositivo que permite la transmisión de imágenes mediante red de telecomunicaciones. En 1996 Martin Gren desarrolló la primera cámara IP que fue lanzada por la compañía AXIS de la que es cofundador[10].

Para poder comprender la necesidad de seguridad asociada a las cámaras IP, es necesario conocer los protocolos de comunicación empleados: HTTP, TCP/IP y RTP/RTCP. En la breve introducción a los protocolos de transmisión más utilizados que se muestra a continuación, se hace patente la vulnerabilidad de los dispositivos, en este caso cámaras IP, que hacen uso de ellos.

2.1.1 HTTP

Hypertext Transfer Protocol o HTTP de sus siglas en ingles es un protocolo extendido de transferencia de datos a nivel de aplicación mediante la red que ha estado en uso desde los inicios de la WWW (World Wide Web) en el año 1990[11].
Este protocolo hace uso de cabeceras, métodos de peticiones estandarizados o verbos y códigos de error para la gestión de peticiones de transferencia.

El alto nivel de esta especificación facilita la manipulación de las peticiones volviéndolas vulnerables ante ataques de modificación, monitorización y suplantación de identidad.

Muchas de las cámaras IP del mercado hacen uso de este protocolo de comunicaciones.

2.1.2 TCP/IP

Este protocolo es en realidad un conjunto de dos protocolos de comunicaciones TCP (Transmission Control Protocol) e IP (Internet Protocol) [12, 13].

TCP es el encargado de gestionar el intercambio de mensajes entre los puntos comunicantes. Se asegura de que los datos lleguen correctamente. Es el protocolo de transporte más extendido de Internet.

IP se encarga del envío de la información por el canal más eficiente disponible.

El protocolo IP no asegura el correcto envío de los datos. Es por eso que se hace un uso combinado de estos dos protocolos quedando TCP en una capa por encima de IP.

Uno de los ataques más conocidos a este protocolo es el ataque SYN Flood. Este ataque es un tipo de denegación de servicio que aprovecha la necesidad de creación de conexión del protocolo entre el servidor y el cliente para saturarlo [14].

2.1.3 RTP/RTCP

Al igual que en TCP/IP, RTP/RTCP es una combinación de protocolos que se completan entre si [15].

RTP (Real-time Transfer Protocol) es el encargado de la definición del envío de datos de video y audio en tiempo real. Este protocolo se encuentra a nivel de aplicación.

Para el control del envío sobre RTP se establece el protocolo de control RTCP (Real-time Transfer Control Protocol). RTCP se encarga de la supervisión y control de la calidad de la transmisión.

Se han implementado sistemas de tipo sniffers que permiten interceptar y grabar datos transferidos por este protocolo [16]. Por lo tanto este protocolo por defecto queda expuesto frente a ataques.

2.2 La ciberseguridad

Según Kaspersky, la ciberseguridad es la práctica para evitar la ejecución de ataques a equipos electrónicos [17].

En los últimos años, la cantidad de ciber-ataques ha aumentado, suponiendo un gran riesgo para empresas y usuarios.

En lo que a aplicaciones web concierne, se han creado organizaciones para identificar y clasificar estos ataques, o explotación de vulnerabilidades, para poder así aportar conocimientos a los desarrolladores y a los propios usuarios.

Así, OWASP, una de las organizaciones creadas para detectar los errores de ciberseguridad, publicó en el año 2010 una lista con los 10 riesgos más críticos en aplicaciones web. Esta lista ha sido actualizada en el año 2017 [18].

Considerando que las cámaras IP permiten conexión remota a través de aplicaciones web, en la mayoría de las ocasiones, esta clasificación es aplicable.

OWASP Top 10 - 2013		OWASP Top 10 - 2017
A1 – Injection	→	A1:2017-Injection
A2 – Broken Authentication and Session Management	→	A2:2017-Broken Authentication
A3 – Cross-Site Scripting (XSS)	↘	A3:2017-Sensitive Data Exposure
A4 – Insecure Direct Object References [Merged+A7]	U	A4:2017-XML External Entities (XXE) [NEW]
A5 – Security Misconfiguration	↘	A5:2017-Broken Access Control [Merged]
A6 – Sensitive Data Exposure	↗	A6:2017-Security Misconfiguration
A7 – Missing Function Level Access Contr [Merged+A4]	U	A7:2017-Cross-Site Scripting (XSS)
A8 – Cross-Site Request Forgery (CSRF)	☒	A8:2017-Insecure Deserialization [NEW, Community]
A9 – Using Components with Known Vulnerabilities	→	A9:2017-Using Components with Known Vulnerabilities
A10 – Unvalidated Redirects and Forwards	☒	A10:2017-Insufficient Logging&Monitoring [NEW,Comm.]

Figura 1. Clasificación vulnerabilidades aplicaciones web. Fuente: OWASP (Julio 2019)

2.3 La ciberseguridad en cámaras IP

El aumento de instalaciones de cámaras IP junto con el incremento de ataques cibernéticos, hace necesaria una reflexión sobre la seguridad que requiere el mundo en expansión de este tipo de dispositivos.

Según Gartner, "para 2020, más del 25% de los ataques identificados en las empresas involucrarán a el IoT, aunque el presupuesto para la seguridad del IoT representará menos del 10% de la inversión de seguridad de TI" [19].

No parece tener sentido que uno de los mercados con mayor perspectiva de crecimiento, sea el objetivo de una parte tan pequeña de la inversión total de la seguridad de TI. Menos aún si se tiene en cuenta la tendencia creciente de los ataques cibernéticos.

En definitiva, los datos muestran que la funcionalidad tiene mucho mayor peso que la seguridad para los usuarios a la vez que para las empresas desarrolladoras tiene más peso las ventas y lanzamientos de nuevos productos.

Como se menciona en el apartado anterior, las empresas priorizan la velocidad de lanzar al mercado nuevos dispositivos frente a la seguridad de los mismos.

Los usuarios actúan en la línea de las empresas, haciendo caso omiso a la seguridad y valorando únicamente la funcionalidad de los dispositivos IoT en general, las cámaras IP entre ellos.

Debido a esta situación, algunos gobiernos han comenzado a definir ciertos requisitos mínimos de seguridad para IoT, incluidas las cámaras IP. Mientras que en la Unión Europea se está tratando de definir un marco legal, en EEUU, el senado ha aprobado ya en 2017 un proyecto de ley "The IoT Cybersecurity Improvement Act" para mejorar la seguridad en IoT [20]. Según Bruce Schneier de la Universidad de Harvard, es un avance muy modesto, ya que no regula el mercado de estos dispositivos.

Queda clara la preocupación de ciertas instituciones antes esta situación:

- Los ataques a dispositivos IoT aumentan.
- Los usuarios de dispositivos IoT aumentan.
- Las funcionalidades de los dispositivos IoT aumentan.

Las cámaras IP domésticas están conectadas habitualmente mediante una red WIFI. Haciendo uso de esta red, se puede acceder a ellas desde otros dispositivos como teléfonos móviles o tablets, que pueden hacer uso o no de un servidor intermedio.

En esta estructura tan habitual se encuentran numerosos factores de riesgo.

Un atacante puede monitorizar la red y obtener datos suficientes para acceder a un dispositivo, comprometiendo la seguridad de toda la red Wi-Fi del hogar.

También es posible realizar ataques de denegación de servicio DDoS impidiendo el correcto funcionamiento de los dispositivos. Es especialmente ilustrativo el ejemplo que nos concierne de un sistema de video vigilancia. Si se deniega el acceso, es muy posible que no se consiga emitir avisos o que no se puedan ver las imágenes permitiendo acceder a la zona vigilada sin ser detectado.

Existen casos en los que atacantes consiguen modificar el firmware para utilizarlos como botnets y ejecutar peticiones desde ellos.

Como se ha mencionado anteriormente, las cámaras IP manejan datos sensibles de seguridad y de intimidad personal. Estos datos deben tener un tratamiento especial para no ser expuestos. En ocasiones, debido a una configuración deficiente o a falta de implementación en el apartado de la seguridad, no es así.

Los atacantes hacen uso de estas fallas para poder acceder a este tipo de información.

Como se puede apreciar, hay varios puntos en los que la seguridad en dispositivos domésticos debería jugar un papel importante.

Según la Fundación Telefónica "el catálogo de amenazas que acechan en las redes al usuario es tan grande como la imaginación de los ciberdelincuentes"[21]. Entre todas, destaca 6 de las amenazas existentes:

- Introducción de programas maliciosos
- Adquisición de control de equipo para generar botnets
- Suplantación de identidad
- Robos económicos
- Robos lúdicos
- Robos de imagen

Como ejemplo de uno de estos grandes ataques producidos en los que cámaras IP se han visto involucradas, se encuentra el realizado en Octubre de 2016. En este ataque se hizo uso de cámaras IP y routers domésticos para mediante multitud de peticiones conseguir una denegación de servicio (DDoS) contra la empresa proveedora Dyn. Como resultado, grandes empresas como Amazon, Netflix y Paypal no pudieron ofrecer sus servicios.

3 Estado del arte

3.1 Herramientas de seguridad

Para la contención de los ataques cibernéticos se han creado una serie de herramientas que analizan y gestionan el tráfico de datos. Cada una de estas herramientas hace uso de diferentes metodologías para desempeñar su labor. Para el desarrollo de la propuesta de este trabajo se han utilizado ideas que parten de las herramientas que se mencionan. Para ello, es preciso conocer el funcionamiento de estos sistemas de seguridad y el aporte que hacen a la neutralización de las fallas de seguridad. Según Hdiv [22], estas herramientas pueden clasificarse en tres grandes grupos: WAF, AST y RASP.

3.1.1 WAF (Web Application Firewall)

Es el principal representante en lo referente a protección de aplicaciones. Las herramientas WAF basan su funcionamiento en el análisis del tráfico existente entre los clientes y los servidores web para determinar si la petición se trata de un ataque o no. Dentro del mercado existen fundamentalmente dos tipos de soluciones WAF:

- **Basadas en blacklist:** las peticiones recibidas se comparan con un lista predeterminada de posibles ataques para detectar si se trata de un ataque.

- **Basadas en whitelist:** se trata de la estrategia contraria. Son soluciones que definen en primer lugar cuales son los tipos de datos aceptados y posteriormente validan las peticiones entrantes sobre los valores obtenidos. A pesar de la evidente

superioridad de esta solución frente a la anterior, la complejidad reside en el método de obtención de dichas whitelists. Existen dos técnicas de obtención de whitelist, por un lado procesos de aprendizaje y por otro el parseo de contenido de salida.

Los WAF poseen ciertas limitaciones a tener en cuenta:

- **Vulnerabilidad:** sobre todo los WAF basados en blacklist presentan un riesgo importante dado que pueden generarse nuevos tipos de riesgos no contemplados por los blacklist. Los basados también en whitelist no son capaces de detectar el origen de los problemas de seguridad en el código (por ejemplo sql injection o SQLi), no pudiendo ser tan efectivos de forma externa.

- **Rendimiento:** sobre todo los WAF que combinan ambas técnicas (whitelist + blacklist) y que están obligados a analizar las respuestas que salen del servidor presentan problemas de rendimiento cuando la carga de las aplicaciones es importante debido al alto costo computacional que implica este análisis.

- **Falsos positivos:** probablemente uno de los principales problemas de las soluciones WAF. La causa de esta realidad radica fundamentalmente en la complejidad de las soluciones y los problemas de adaptación a los cambios en la funcionalidad de la aplicación que habitualmente es constante. Por otro lado, las soluciones WAF se aplican habitualmente en los entornos de producción y preproducción no detectándose los problemas de integración con las aplicaciones en la fase de desarrollo.

- **Experiencia de usuario:** cuando un WAF detecta un error, al estar fuera del aplicativo, no tiene otra forma de reportar el problema que no sea redirigir a una página de error concreta definida por la aplicación. Este tipo de respuestas rompen por completo la usabilidad del usuario, dado que es posible que haya añadido caracteres no permitidos de forma no maliciosa y se encuentra con un mensaje de error genérico.

3.1.2 AST (Application Security Testing)

Se trata de soluciones de detección de vulnerabilidades de seguridad que pueden ser utilizadas ya sean dentro del entorno de desarrollo o en los entornos de producción en forma de scanner que analiza un sitio web de forma externa. Este tipo de soluciones requieren en todos los casos que los problemas detectados sean solventados de forma manual por los programadores.

Por otro lado, es importante tener en cuenta que este tipo de soluciones únicamente son capaces de detectar problemas de seguridad comunes que siguen un patrón concreto, ya sea en código o en el comportamiento en ejecución que permite su detección. Sin embargo es importante destacar que las soluciones AST no son capaces de detectar las vulnerabilidades de negocio o aquellas relacionadas con la lógica de negocio o los procesos de negocio de la aplicación. Una solución AST no es capaz de detectar si podemos acceder a una zona autenticada sin los roles suficientes o poder leer un dato que no nos corresponde.

3.1.3 RASP (Runtime Application Self Protection)

Se trata de soluciones que intentan superar las limitaciones de las soluciones WAF centrándose en el análisis de la ejecución de las peticiones dentro del servidor de aplicaciones. Estas soluciones son una evolución de las soluciones AST, pasando a realizar acciones proactivas de protección una vez que han detectado un problema de seguridad. Las soluciones englobadas en la categoría RASP están basadas en técnicas de análisis del código ejecutado en el servidor a nivel de bytecode.

A pesar de la mejora frente a lo ofrecido por las soluciones AST, mantienen una de las principales limitaciones, protegen únicamente una parte de los riesgos, dejando sin protección alguna las vulnerabilidades de negocio o aquellas que son propias de la aplicación y no siguen un patrón concreto que puede ser detectable mediante herramientas.

Estas herramientas son capaces de proteger de forma más efectiva de gran parte de los riesgos comunes, debido al control total de la ejecución en el servidor.

3.1.4 Firewall

Además de las herramientas citadas en los puntos anteriores, existen otras que permiten controlar la comunicación por red entre dispositivos. Los firewalls son capaces de gestionar las autorizaciones para acceder a un sistema informático. Así, mediante reglas definidas, permitirán o no el acceso. Kenneth Ingham y Stephanie Forrest[23] explican la necesidad de firewalls desde que se creó el concepto de intercambio de información y colabora-

ción en red. La tecnología firewall existe desde el año 1987 y se ha aplicado en múltiples instalaciones de redes desde entonces para proteger el acceso a redes internas.

3.1.5 Impacto en la propuesta de las herramientas de seguridad

Si bien las tecnologías anteriormente explicadas ofrecen soluciones para múltiples casos, resultan insuficientes por sí mismas para afrontar y dar solución a la seguridad de las cámaras IP. Es por esto que la solución que se plantea junta ideas de estas herramientas y consigue de esta manera mejorar la seguridad de una instalación de cámaras IP en un entorno doméstico.

Al igual que en los WAF, se realizará un análisis estático de la red, es decir, se monitorizará el intercambio de datos desde el punto de vista de comunicaciones red.

Las herramientas AST y RASP son de gran utilidad cuando se tiene acceso al software y/o al código fuente de la cámara. En este aspecto no parecen de utilidad para dar respuesta a las necesidades planteadas. Por otro lado, a pesar de no poder acceder al código del firmware de la cámara IP, los AST muestran un concepto interesante: análisis del funcionamiento del sistema. Al existir una petición malintencionada, se detectará la funcionalidad inusual.

Para poder ofrecer seguridad ante situaciones detectadas, se realizarán acciones de denegación de acceso en función del resultado del análisis. Para ello se hará uso de un firewall. Mediante reglas configuradas, se permitirá o no la comunicación con las cámaras IP.

3.2 Acceso remoto a cámaras IP

La ventaja de las cámaras IP es la funcionalidad de visualización de imágenes en remoto. Para realizar esta función se hace uso de dos protocolos muy diferenciados: Peer-to-peer en adelante P2P y Real Time Streaming Protocol (RTSP de sus siglas) junto con HTTP o HTTPS.

3.2.1 Acceso directo

Este tipo de cámaras permiten acceso directo desde un navegador. Se accede a la cámara mediante peticiones HTTP/HTTPS o RTSP y se envían directrices para que esta realice funciones tales como cambio de la zona capturada o envío de imágenes.

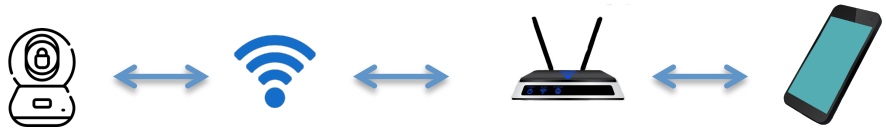

Figura 2. Arquitectura acceso directo

3.2.2 P2P

Las cámaras IP que hacen uso del protocolo Peer-to-peer, en adelante P2P, para la transmisión de imágenes, aportan la ventaja de no requerir puertos abiertos en el router. La conexión entre el dispositivo en donde se visualizarán las imágenes y la propia cámara IP se realiza a través de un servidor intermedio que se encarga de gestionar la petición. Esta situación añade la seguridad implementada en el servidor P2P que puede variar desde muy estricta a ninguna.

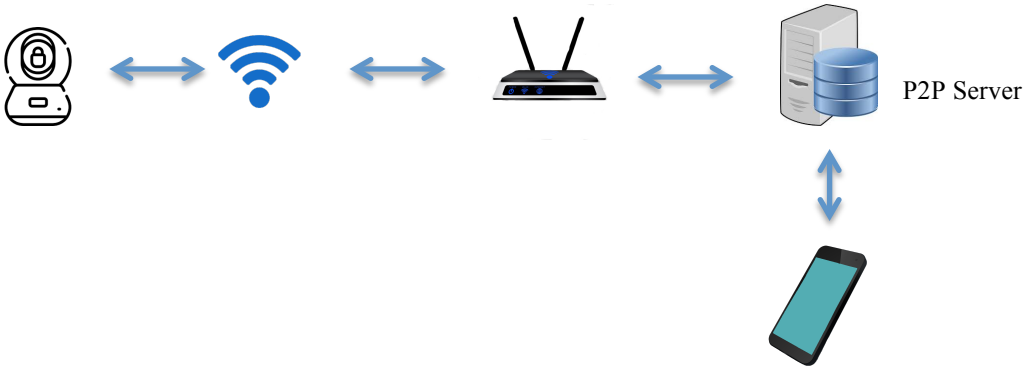

Figura 3. Arquitectura P2P

3.3 Seguridad en dispositivos IoT

Como ya se ha comentado, la seguridad en este tipo de dispositivos es algo en lo que se debe avanzar. En varios trabajos, la seguridad aplicada a las cámaras IP radica en la securización del canal de conexión. Así, Francisco Javier García Mata [24], explica la protección aplicada a las cámaras IP como securización de la red en la que está implantada. Se enumeran los siguientes aspectos de seguridad:

1. Autenticación mediante nombre de usuario
2. IEEE802.1x
3. HTTPS o SSL/TLS
4. VPN
5. Redes WIFI y sus configuraciones de seguridad

Si bien estos puntos son necesarios, resultan insuficientes para considerar segura una instalación de cámara IP.

Garg Hittu y Dave Mayank [25], proponen hacer uso de una plataforma middleware expuesta en la nube a la que se conectan los dispositivos IoT mediate REST API. Es solución gestionaría las peticiones de los dispositivos y controlaría el tráfico entre ellos. El problema de esta solución es que no se detectarían accesos lícitos de usuarios no autorizados, en caso de acceder a un dispositivo se seguiría comprometiendo la red WIFI. Tampoco solventaría el problema de monitorización de red WIFI doméstica, aunque si sería muy útil para solucionar el problema de monitorización de redes públicas.

Desde el punto de vista de aplicaciones web, existen productos como Hdiv o Contrast[26] que ya han demostrado su eficacia a la hora de detectar vulnerabilidades y bloquear ataques. Estos productos podrían utilizarse como complemento de la solución anteriormente planteada o como sistema de seguridad de los casos en los que se haga uso de un servidor web intermedio. En cualquier caso, siguen sin resolver los problemas existentes en la seguridad de las cámaras IP.

Otros autores han hecho varios esfuerzos por crear taxonomías de los ataques relacionados con el mundo IoT.

Eyal Ronen y Adi Shamir [27], en su trabajo de casos de ataques a dispositivos de luces inteligentes, realizan una serie de ataques para demostrar las vulnerabilidades contra estos dispositivos. Se demuestra que los dispositivos son vulnerables y se extiende la conclusión al resto de dispositivos IoT. Para tratar de mitigar estos puntos vulnerables, se presenta una lista de puntos en los que se debe poner especial atención a la hora de desarrollar dispositivos inteligentes.

- Hacer uso de protocolos de comunicación seguros como TLS junto a certificados firmados. Es interesante el uso de contraseñas individuales por dispositivo que no muestren patrones de generación para evitar descubrir su valor. Las conexiones a los dispositivos deben contener los identificadores únicos de ese dispositivo.

- Las APIs de acceso deben tener los puntos de acceso imprescindibles. Cada uno de estos puntos es una puerta en el dispositivo. Conviene limitar y controlar este aspecto.

- La implementación relacionada con los dispositivos debe ser probada y auditada desde el punto de vista de la seguridad.

- Las redes en donde los dispositivos se conectan deben estar separadas del resto de redes para así protegerlas.

S. Rizvi, A. Kurtz, J. Pfeffer y M. Rizvi [28] plantean una clasificación en base a cuatro dominios de seguridad con subdivisiones.

Top-Level Security Domain	Sub-Domains
Architecture	Perception Layer
	Application Layer
	Network Layer
Threat Vector	Communication Attacks
	Physical Attacks
	Application/Software Attacks
Trust	Privacy
	Availability
	Reliability
Compliance	Policy Control
	Government Oversight
	Non-Government Oversight

Tabla 2. Ataques específicos en IoT. Fuente: Securing the Internet of Things (IoT): A Security Taxonomy for IoT (2018)

En el trabajo realizado por M. Nawir, A. Amir, N. Yaakob and O. B. Lynn [29], se realiza un análisis de diferentes ataques explicando su objetivo, debilidades y técnicas de seguridad.

En lo referente a securización de redes domésticas con cámaras IP no se han encontrado soluciones específicas. Sin embargo, existen productos como concentradores de dispositivos que de una forma indirecta ofrecen capas de seguridad en una instalación de dispositivos IoT y por lo tanto también de cámaras IP.

3.4 Sistemas de notificaciones a dispositivos móviles

Las notificaciones son una funcionalidad habitual en el mundo de las aplicaciones móviles. Estas permiten desencadenar acciones en el dispositivo móvil sin requerir interacción por parte del usuario. En ocasiones, lo que se desea es conseguir una acción por parte de este. En la solución de validación que se plantea en este trabajo se requieres de un sistema de notificación para realizar una de las comprobaciones centrales, “Double check”. Para el envío y gestión de las notificaciones los métodos más habituales son el uso de peticiones periódicas a un servicio externo y el uso de herramientas de notificaciones de tipo push como es Firebase Cloud Messaging.

3.4.1 Sistema de notificaciones por consulta periódica

Un sistema de notificación por consulta periódica se basa en realizar peticiones cada espacio de tiempo predefinido. El cliente que desea ser notificado pregunta al sistema encar-

gado de enviar las notificaciones por envíos pendientes. En caso de ser así, la notificación forma parte de la respuesta a la solicitud. Esta metodología permite una ejecución controlada, ya que es habitual no depender de servicios externos para su gestión y el tiempo entre solicitudes está definido. Con este sistema es sencillo asegurar que la totalidad de notificaciones llegan a todos sus destinatarios.

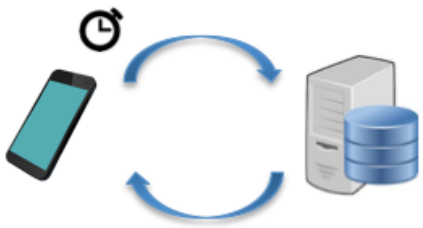

Figura 4. Ejemplo de petición periódica

El inconveniente de esta metodología es la ejecución de solicitudes sin respuesta práctica. Es decir, se realizan múltiple peticiones desde el dispositivo cliente que no son necesarias ya que no existen notificaciones pendientes. Desde el cliente no hay forma de controlar esto, ya que no se tiene la información de si hay notificaciones pendientes. Si a esta casuística se le añade la necesitad de la pronta disponibilidad de la información de la notificación que exige realizar peticiones cada periodo de tiempo reducido, se obtiene un coste elevado tanto de procesador como de red.

3.4.2 Firebase Cloud Messaging (FCM)

Firebase Cloud Messaging, en adelante FCM, es un sistema de notificaciones push implementado por Google. Esta herramienta ofrece el servicio de notificaciones a dispositivos Android, iOS y aplicaciones Web [30]. Se trata de la evolución de la herramienta Google Cloud Messaging (GCM) implementada por la misma compañía que dejó de dar servicio el

10 de abril de 2018 [31] y que se limitaba a servicio de notificación a dispositivos Android. FCM es capaz de enviar mensajes de 4KB.

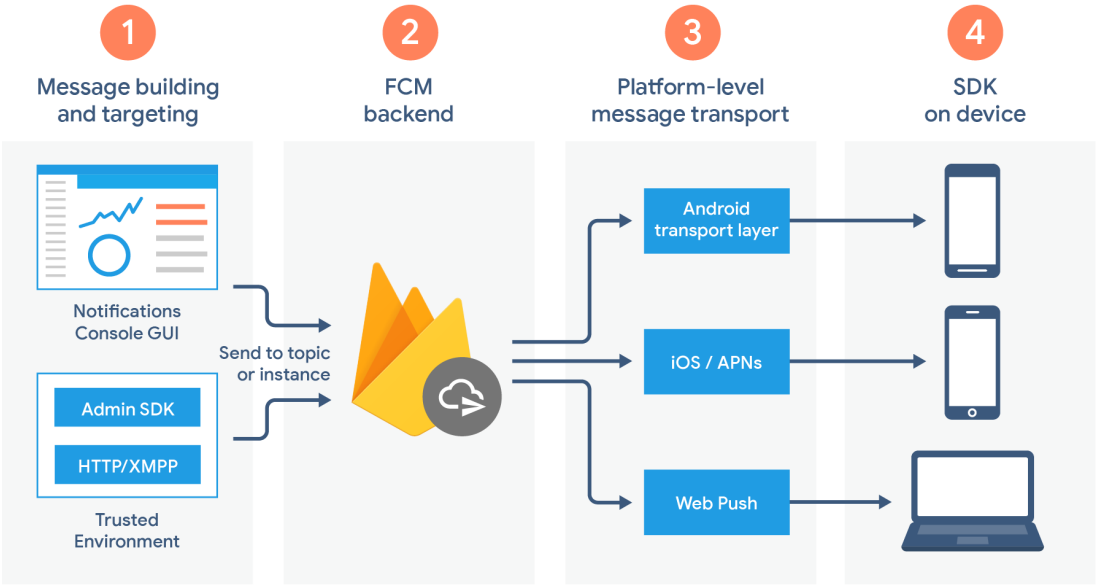

Figura 5. Descripción general de la arquitectura de FCM. Fuente: https://firebase.google.com/docs/cloud-messaging/fcm-architecture

Para el envío de notificaciones, cada dispositivo debe registrar la aplicación en el servidor del que se obtiene un identificador único para la instancia de la aplicación cliente. El servidor que pretende el envío realiza una solicitud de notificación al servidor de FCM y este se encarga del envío a los dispositivos suscritos. Para recibir la notificación, el dispositivo debe tener conectividad a internet.

En el análisis realizado por Yavuz Selim, Bahadir Ismail y Murat Demirbas [32] sobre la funcionalidad de GCM, antecesor de FCM, se detalla que aunque el envío de notificaciones es inestable cuando existe un gran número de receptores. Aun así, gran parte de sus clientes recibieron los mensajes en una franja de tiempo inferior a 10 segundos.

		Experiments	
		Offline	Online
Connection Type	WiFi	1656	199
	Cellular	1299	165
	Unidentified	163	18
Device Type	Smartphone	2586	334
	Tablet	532	48
Total		3118	382

Tabla 3. Distribución de dispositivos clientes para notificaciones. Fuente: *Google cloud messaging (GCM): An evaluation*

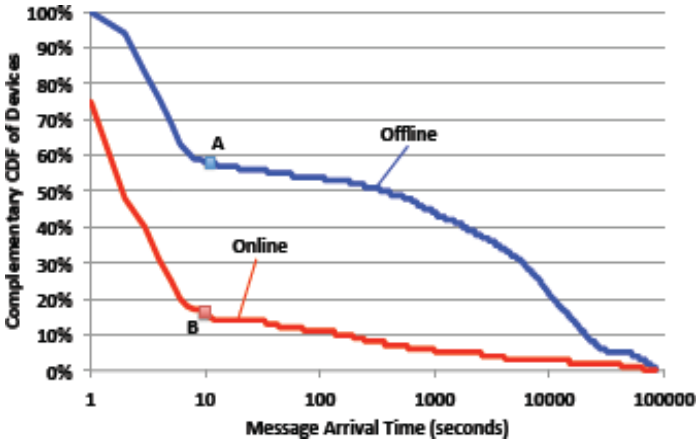

Figura 6. Porcentaje de dispositivos a la espera de mensajes por tiempo transcurrido. Fuente: *Google cloud messaging (GCM): An evaluation*

La figura 1 muestra los dispositivos tanto online como offline utilizados para realizar las pruebas de envío de notificaciones, mientras que la figura 1 muestra el número de dispositivos que permanecen a la espera de recepcionar la notificación por tiempo de espera.

En el caso de aplicar esta tecnología con los datos mostrados sería viable, ya que el número de dispositivos empleados para visionar una cámara es muy reducido. Se puede suponer, que las notificaciones se recibirían en menos de 10 segundos. Google afirma que estos números han mejorado con la implementación de FCM.

4 Propuesta

4.1 Descripción general de la solución

Se propone la utilización de un dispositivo que realice las funciones de servidor Wi-FI para gestionar las conexiones de las cámaras IP y así dar respuesta a las cuestiones de seguridad planteadas en los apartados anteriores y que el usuario doméstico sea consciente de las mismas. La solución plantea un sistema para evitar accesos no autorizados a las cámaras IP y de esta manera mantener la seguridad de los dispositivos.

El servidor Wi-FI será el encargado de analizar el tráfico de la red a los dispositivos y detectar y proteger así frente a varios de los errores de seguridad existentes. Aplicando este enfoque no es necesario conocer las vulnerabilidades existentes en el firmware interno de las cámaras, ya que se detectará y protegerá el dispositivo de forma perimetral. Este funcionamiento se puede asemejar a los WAFs, ya que se plantea un "dispositivo enlace" que se sitúa entre cliente y servidor para proteger a este.

Una vez vinculadas las cámaras IP al servidor Wi-Fi que se expone, este analizará el tráfico de red para determinar si las conexiones son lícitas o no. Para ello se requiere de un proceso de monitorización de red que obtenga los datos para ser analizados y un servicio de validación que detecte los accesos no autorizados a las cámaras. El servicio de validación realiza comprobaciones de tipo listas blancas, listas negras, DDoS y control de autorización de accesos.

La solución incluye un servicio que funciona como apoyo de la validación y que corre fuera del servidor Wi-Fi explicado. Este servicio se trata de una aplicación móvil. Su labor

es detectar y notificar al servicio de validación del servidor Wi-Fi el uso de una aplicación móvil para acceder a una de las cámaras IP registradas. Esta funcionalidad es necesaria para determinar que un usuario registrado ha sido el que ha iniciado la comunicación con la cámara IP, y que por lo tanto es una conexión lícita. De esta manera se controla la autorización de accesos a los dispositivos vinculados en el sistema de validación.

Se analizará el flujo de red al completo, es decir, en ambas direcciones. Para ello, es necesario que el sistema identifique los dispositivos vinculados al servidor Wi-Fi. No se permite la conexión entre estos dispositivos. De esta manera se evita que un dispositivo que pueda estar comprometido funcione como puerta de acceso y comprometer a otros dispositivos. Como se ha dicho, la protección se aplica tanto a las conexiones hacia las cámaras IP como a las salientes desde estas. Únicamente usuarios lícitos pueden mantener la comunicación y únicamente hacia usuarios lícitos se permite la comunicación. Para conocer los usuarios lícitos, se ha implementado un proceso de creación de lista blanca dinámica. Se establece un tiempo finito determinado como "periodo seguro" en el que las conexiones son registradas por cámara IP y el origen, si es entrante, o destino, en caso de ser saliente, es añadido a esta lista. Las conexiones que superan la validación son asimismo añadidas la lista de confianza por el hecho de haber sido ya validadas.

Para permitir o bloquear los accesos se hace uso de una herramienta Firewall. La configuración del Firewall se realiza mediante reglas añadidas y eliminadas desde el proceso de validación anteriormente mencionado. De esta manera se controla y asegura toda conexión entrante y saliente de las cámaras IP vinculadas al servidor Wi-Fi.

El servicio de validación no hará funciones de puente entre el servidor Wi-Fi en el que corre y la cámara IP. Para monitorizar y validar el flujo de red estará a la escucha de una forma paralela. Se ha comprobado que la validación en su punto más duradero se realiza en cuestión de pocos segundos, menos de 8 segundos, siendo la duración de una validación completa habitual de entre 2 y 4 segundos desde el intento de conexión con la cámara IP. Este tiempo de respuesta parece adecuado para proteger una instalación domestica de cámaras IP.

El diseño de la solución en módulos de comprobación individuales por cámara IP, hace que la protección sea totalmente configurable. Es decir, cada dispositivo puede ser configurado de forma autónoma activando o desactivando los comprobadores deseados. Como parte de la configuración permitida, se permite elegir el tipo de protección entre notificar la detección de un ataque (email y notificación mediante la aplicación móvil de apoyo) o bloquear la comunicación. La configuración predeterminada será de bloquear los ataques detectados.

4.2 Arquitectura de la solución

La propuesta consta de dos funciones generales que funcionarán dentro de un dispositivo Raspberry para dar un servicio de seguridad completo. La Raspberry Pi será un servidor WI-FI en donde correrán los siguientes servicios:

- Interceptador y controlador de las comunicaciones
- Servicio de validación de peticiones

Como parte del servicio de validación de peticiones se ha implementado una aplicación Android que hará la función de apoyo a la comprobación de autorización de acceso a las cámaras IP registradas en el sistema. Este módulo es la parte de la propuesta que funciona de forma externa al dispositivo Raspberry Pi. Se encargará de verificar que un usuario registrado en el sistema es el que ha iniciado la comunicación con la cámara IP.

La funcionalidad de la Raspberry Pi y de los dos componentes mencionados se explicará a continuación. En la figura 7 se muestra la estructura general del sistema propuesto:

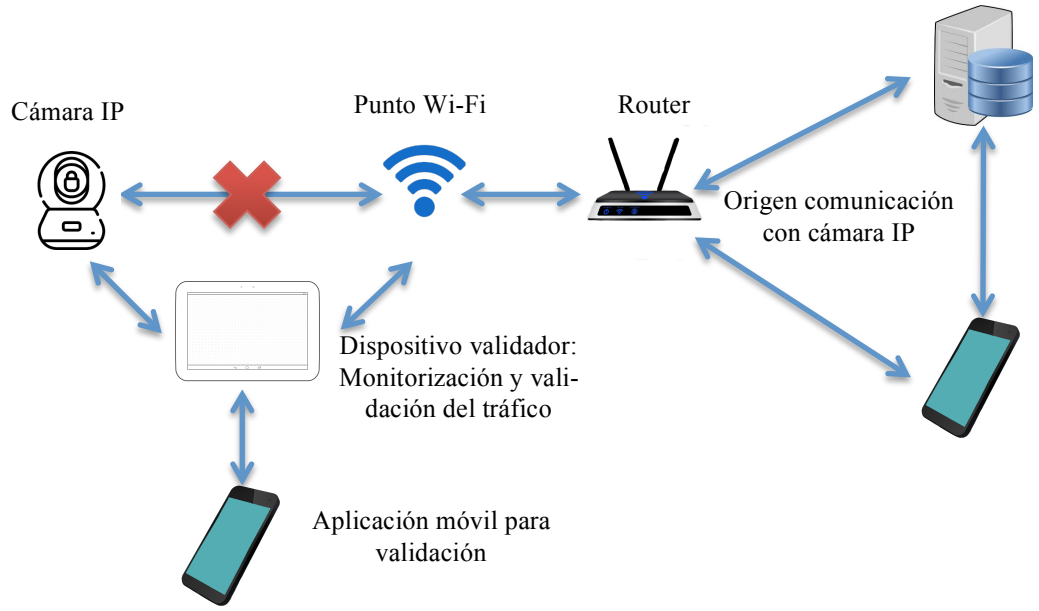

Figura 7. Arquitectura general de la solución

4.3 Dispositivo Raspberry Pi

Como se ha dicho en el apartado anterior, se implantará un servidor WI-FI en la red doméstica a el que se vincularán las cámaras IP. Debido a su reducido tamaño, potencia y bajo consumo, este dispositivo será una Raspberry Pi.

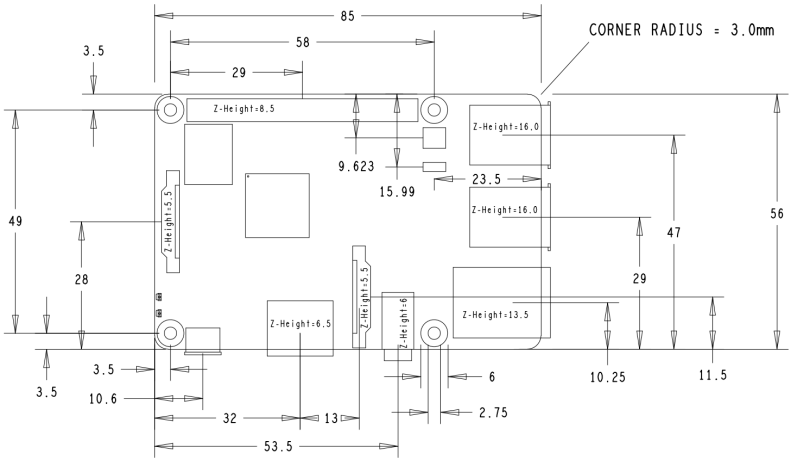

Figura 8. Esquema Raspberry Pi 3B+. Fuente: raspberrypi.org (Julio 2019)

Gracias a la compatibilidad con el estándar IEEE 802.11, la Raspberry Pi se vinculará con la red WI-FI doméstica. A su vez, hará los servicios de router WI-FI para generar una subred y permitir la vinculación de las cámaras IP en ella.

Se deberá configurar la placa Raspberry Pi para funcionar como servidor WI-FI. Para conseguirlo se debe instalar y configurar el servidor DHCP mediante isc-dhcp-server y el servicio de creación de puntos de acceso mediante hostapd.

Los componentes que se detallan a continuación están instalados, configurados e implantados dentro de la Raspberry Pi.

4.3.1 Interceptador y controlador de las comunicaciones

Como se ha comentado anteriormente en este trabajo, el primer punto para validar las comunicaciones entre usuarios y cámaras IP precisamente conseguir los datos de las comunicaciones. Con esta información se consigue gestionar el flujo de red.

4.3.1.1 Monitorización de la red

Las peticiones de conexión a las cámaras IP deben ser monitorizadas y analizadas para añadir la capa de seguridad que compete en este trabajo. El uso de un proxy inverso permite capturar y reenviar dichas peticiones a un servicio de validación en el que se aplicarán diferentes reglas. Básicamente, su función en este caso radica en redirigir las peticiones cuyo destino son las cámaras IP instaladas en la subred anteriormente citada para que otro componente intermedio certifique su legitimidad.

El problema de este enfoque, es que estas peticiones deberían ser redirigidas al servicio de validación. Al actuar de esta manera, la salida desde este servicio perdería visión del destinatario final imposibilitando la ejecución correcta de la funcionalidad original. Se detectó a su vez, que las señales que requieren ser analizadas no consisten en su totalidad, ya que para la conexión con una cámara IP requiere de varias peticiones de red y en la mayoría de los casos sobre diferentes protocolos. Este hecho complica la tarea de redireccionar y analizar todo el tráfico permitiendo a su vez el envío de dicha peticiones a su destinatario real.

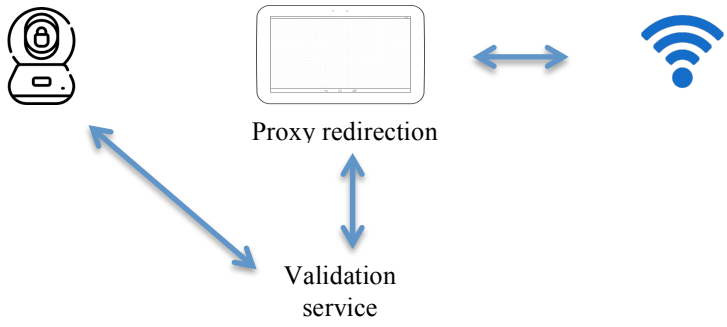

Figura 9. Enfoque para interceptación de la señal mediante proxy inverso

Para solucionar esta problemática se ha modificado el enfoque de captura de comunicaciones. Haciendo uso de librerías de monitorización de tráfico (tcpdump [33]), se detecta en tiempo real el tráfico existente en la interfaz de red asignada al concentrador. Entre los datos proporcionados por esta librería, se encuentra la IP de origen, IP de destino, puerto de destino y protocolo empleado para la comunicación. Haciendo uso de estos datos es posible realizar la labor de validación de las peticiones a los dispositivos conectados al concentrador. Es preciso puntualizar que en el concentrador también se tiene la información de las IPs y direcciones MAC de los dispositivos conectados, en este caso cámaras IP. De esta manera se puede trazar un mapa con el origen y el dispositivo destino. El servicio de validación hará uso de la información proporcionada por la librería de monitorización de red y en función del resultado del análisis de estos datos modificará o no las reglas de acceso de comunicaciones del controlador o punto de acceso a las cámaras IP. Si bien esta validación no se encuentra estrictamente en medio de las comunicaciones a modo de puente (ya que es un análisis en paralelo al flujo de red), el coste en milisegundos del análisis proporciona un resultado similar y efectivo.

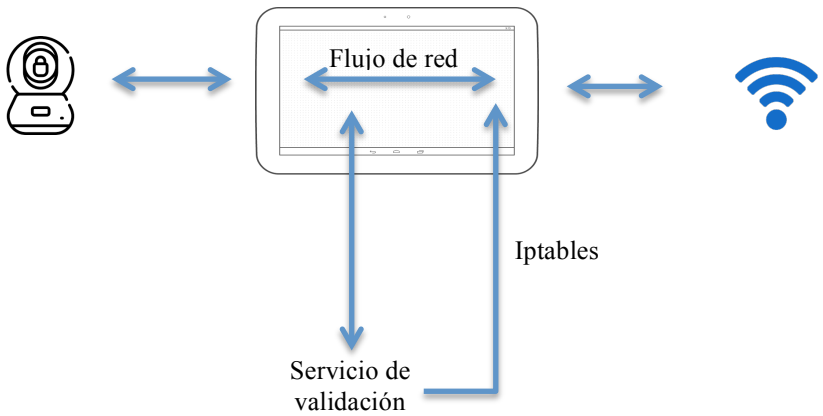

Figura 10. Enfoque para interceptación de la señal mediante monitorización (tcpdump)

4.3.1.2 Gestión de accesos red a los dispositivos

Las reglas aplicadas mediante Iptables realizarán la labor de firewall. Como se ha comentado en líneas anteriores, estas reglas serán modificadas por el servicio de validación para permitir o bloquear la comunicación de red con los dispositivos.

Por tanto, el módulo de monitorización de red es un sistema compuesto por la librería tcpdump e iptables que pueden colaborar gracias al paso centralizado en el concentrador del flujo de red.

4.3.2 Servicio de validación de peticiones

El servicio de validación de peticiones se trata de un servicio web implementado en JAVA y ejecutado dentro de un contenedor de servlets Tomcat. Este servicio hace uso de la información proporcionada por el sistema de monitorización para obtener la información relativa al tráfico de red.

Se ejecutan una serie de validaciones en cadena, de tal modo que si en una de ellas el resultado es erróneo se considerará que la petición es ilegítima, y por lo tanto se identificará como ataque. Las validaciones que se realizan desde este módulo son las siguientes:

1. Comparación de IP peticionaria con lista de IPs permitidas.
2. Comparación de IP peticionaria con lista de IPs maliciosas
3. Control del número de peticiones periódicas desde una IP
4. Control mediante “Double check” de la legitimidad de la petición.

En el apartado “Software de validación” se detalla el diseño y funcionalidad del sistema software al completo.

4.3.3 Aplicación móvil de control de accesos

Como se ha mencionado en puntos anteriores, existe una parte de la solución que funciona fuera del servidor Wi-Fi de la propuesta. En el punto “Servicio de validación de peticiones” se menciona la validación de control mediante “Double check” de la legitimidad de la petición. Esta validación se realiza mediante colaboración entre el servicio de validación del servidor Wi-Fi y la aplicación móvil que se explica en este punto.

Al detectar comunicación hacia una cámara IP, entre las comprobaciones que son ejecutadas está la validación “Double check”. Aquí, el servicio de validación envía una notificación a la aplicación o aplicaciones instaladas y vinculadas para asegurar que desde uno de estos móviles es desde donde se ha iniciado la comunicación con la cámara IP. En caso ser así, la aplicación notificará al servicio validador de que efectivamente se ha iniciado desde un dispositivo móvil registrado y que por lo tanto la conexión es lícita.

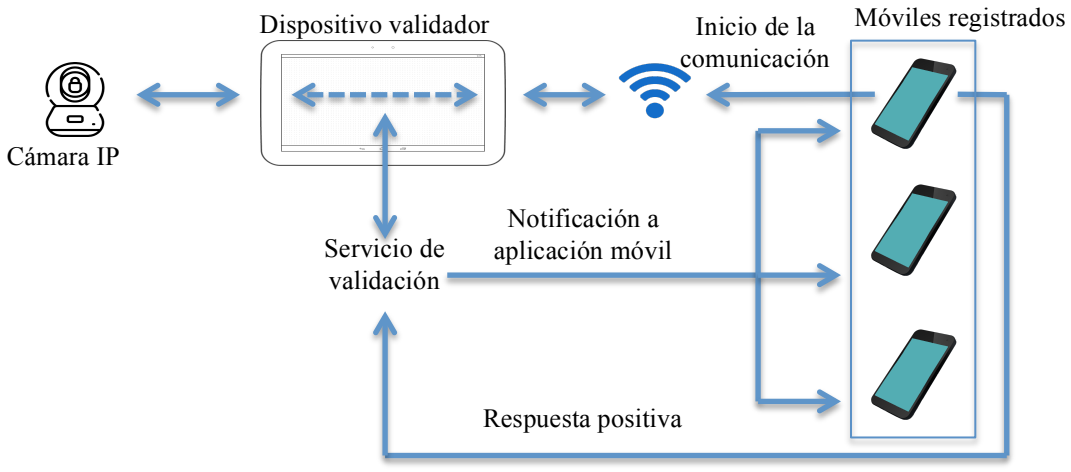

Figura 11. Validación Double check

4.4 Software de validación

En apartados anteriores se ha realizado una breve explicación del sistema de validación. En este punto se detallará el sistema software al completo. El software de validación consta de dos aplicaciones claramente separadas debido en primer lugar al dispositivo en el que se instalan/ejecutan y en segundo lugar a la funcionalidad que ofrecen:

1. Servicio de validación de peticiones
2. Aplicación Android de control de acceso

El servicio de validación es una aplicación que queda desplegada dentro del dispositivo concentrador. Dispone de recursos software como base de datos y lógica de negocio implementada en JAVA y gestión de dependencias mediante maven.

El despliegue y funcionamiento de esta aplicación se realiza sobre Spring Boot 2.1.0. Para ello se hace uso de la dependencia spring-boot-starter-parent, encargada de gestionar todos los requerimientos de este servidor embebido.

Dependencia	Versión
Hibernate	5.1
Micrometer	1.1
Reactor Californium	Californium-SR2
Spring Data Lovelace	Lovelace-SR2
Spring Framework	5.1
Tomcat	9
Undertow	2

Tabla 4. Dependencias actualizadas Spring Boot. Fuente: spring.io - Third-party library upgrades (Julio 2020)

Son especialmente relevantes las dependencias de Spring Framework y Tomcat para el sistema planteado.

Como tecnología de persistencia, se ha optado por HSQLDB, ya que posee una gran eficiencia a la hora de gestionar datos en forma de objetos al hacer uso de tratamiento de bloques de datos en memoria [34, 35]. Este es un requerimiento del servicio de validación implementado.

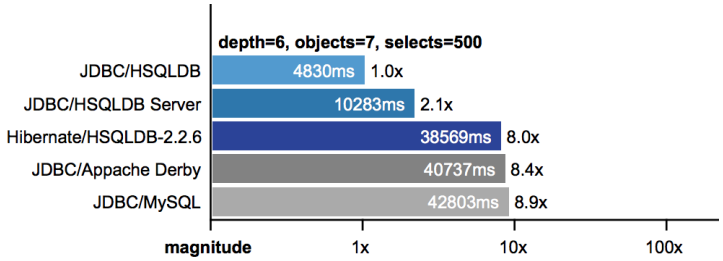

Figura 12. Resultados test escritura. Fuente: Results from running the Poleposition open source database benchmark

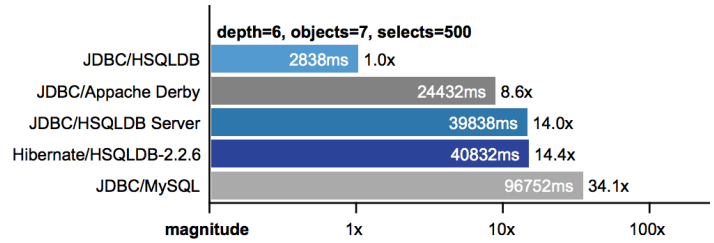

Figura 13. Resultados test lectura. Fuente: *Results from running the Poleposition open source database benchmark*

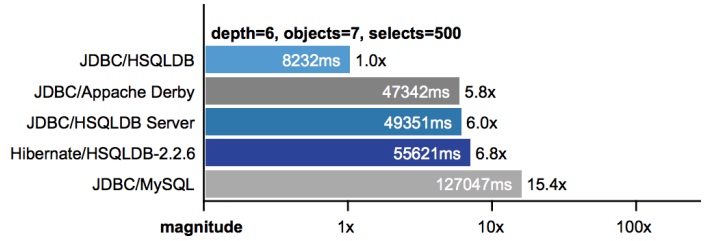

Figura 14. Resultados test actualización. Fuente: *Results from running the Poleposition open source database benchmark*

Las figuras 12, 13 y 14 muestran los datos obtenidos mediante el proyecto poleposition de código abierto diseñado para realizar pruebas no sesgadas de este tipo de herramientas de persistencia de datos. Los resultados muestran que HSQLDB es la herramienta más eficiente tanto para lectura como escritura y actualización de los datos.

4.4.1 Diseño del servicio de validación de peticiones

El módulo software central del sistema propuesto es el servicio de validación de peticiones. A continuación se muestra el diseño de los submódulos implementados que componen dicho servicio. El diseño se muestra en forma de diagramas de flujo para facilitar la comprensión.

4.4.1.1 Obtención de IPs maliciosas

Una de las partes requeridas para la validación del flujo de red es la identificación de IPs clientes como IPs ya identificadas como maliciosas. En la figura 15 se detalla el flujo para obtener esta información.

Como fuente de información se consultan los proveedores HoneyPot[36] y Tor[37]. Ambos son proyectos con largo recorrido en el análisis de IPs causantes de accesos malintencionados.

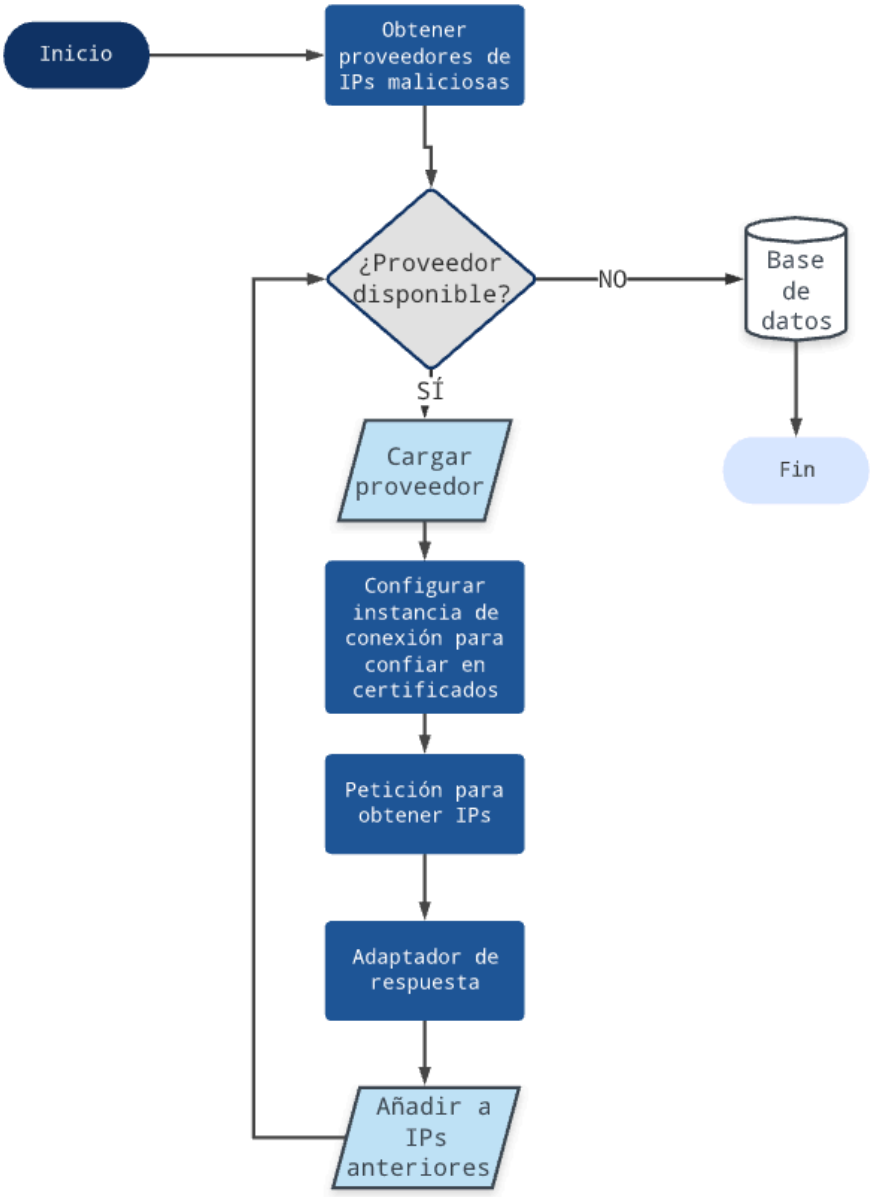

Figura 15. Diagrama de flujo de obtención de IPs maliciosas

La obtención de esta información se realiza de forma periódica. Se ha implementado un sistema de tareas que gestiona este tipo de requerimientos. Las tareas suscritas serán ejecutadas periódicamente en base a la configuración individual de cada una de ellas. En el caso de la tarea de búsqueda de IPs conocidas y registradas como malintencionadas, esta tarea será ejecutada cada 6 horas. Al ser una lista de lenta modificación, se ha creído conveniente no añadir mayor peso de procesamiento al sistema configurando una periodicidad me-

nor. La información obtenida será almacenada en la base de datos anteriormente descrita para poder disponer de esta información en todo momento.

4.4.1.2 Detección de nuevos dispositivos

La validación que se quiere realizar no sería posible sin la identificación de los dispositivos conectados. Para ello se obtiene la IP del dispositivo concentrador. Si se tiene en cuenta que el dispositivo mencionado es el punto de acceso a el que cada una de las cámaras IP se conecta, se puede considerar que las IPs de dichas cámaras IP forman parte de la subred proporcionada por el dispositivo concentrador. El servicio de validación está en funcionamiento sobre este dispositivo, con lo que al obtener la IP del servicio, es posible obtener la IP base de la subred. Haciendo uso de esta información, se modifica la parte final de la IP para realizar sondeos en forma de peticiones ICMP. Si se detecta respuesta, se deduce que hay un dispositivo conectado. Con este procedimiento se obtiene la IP de las cámaras conectadas. El problema que existe en este punto, es que esta IP puede cambiar al desconectarse del dispositivo y volverse a conectar ya sea de forma intencionada o por una pérdida de conexión involuntaria. En el servicio, existe información relacionada con cada cámara que no se debe perder por esta situación. Para solucionarlo, se ha decidido hacer uso de la dirección MAC, única para cada cámara. El modo de obtener esta MAC es la ejecución del comando “arp” junto con la IP obtenida del método anterior. Una vez obtenida esta información se comprueba si el dispositivo ya se encontraba registrado en la base de datos del servicio. Es caso de ser así, se actualizan sus datos. De lo contrario se añadirá al registro de dispositivos y se crearán los tokens identificadores del mismo. Los cambios detectados en lo referente a dispositivos serán notificados a los módulos que han solicitado esta información. De esta manera la información del servicio será robusta y coherente.

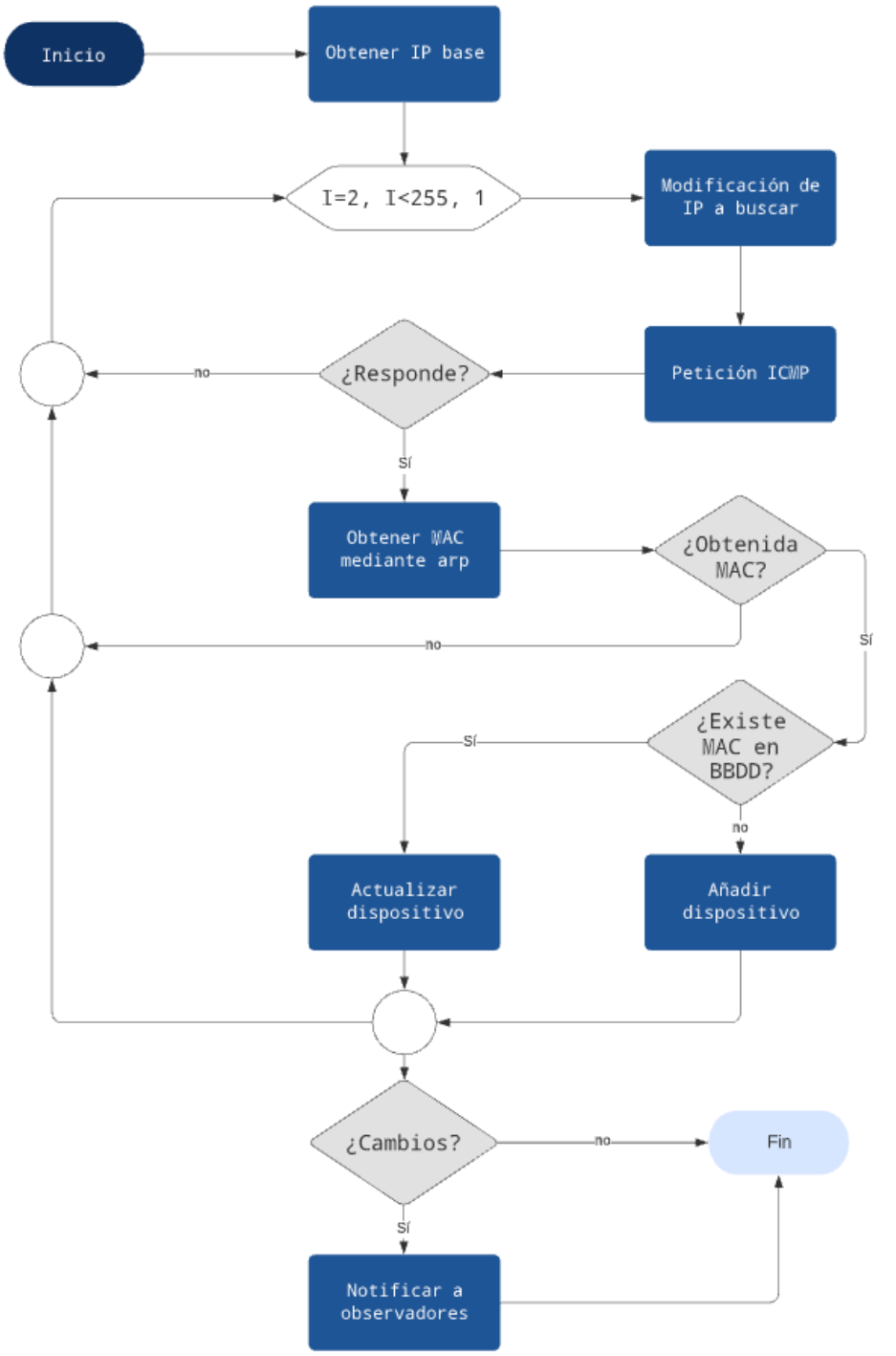

Figura 16. Diagrama de flujo de detección de cámaras IP

La figura 16 detalla el flujo que se sigue para realizar el proceso explicado.

4.4.1.3 Monitorización de datos

El siguiente punto detalla el proceso de monitorización de red, necesario para la validación del flujo de datos entre clientes y cámaras IP registradas.

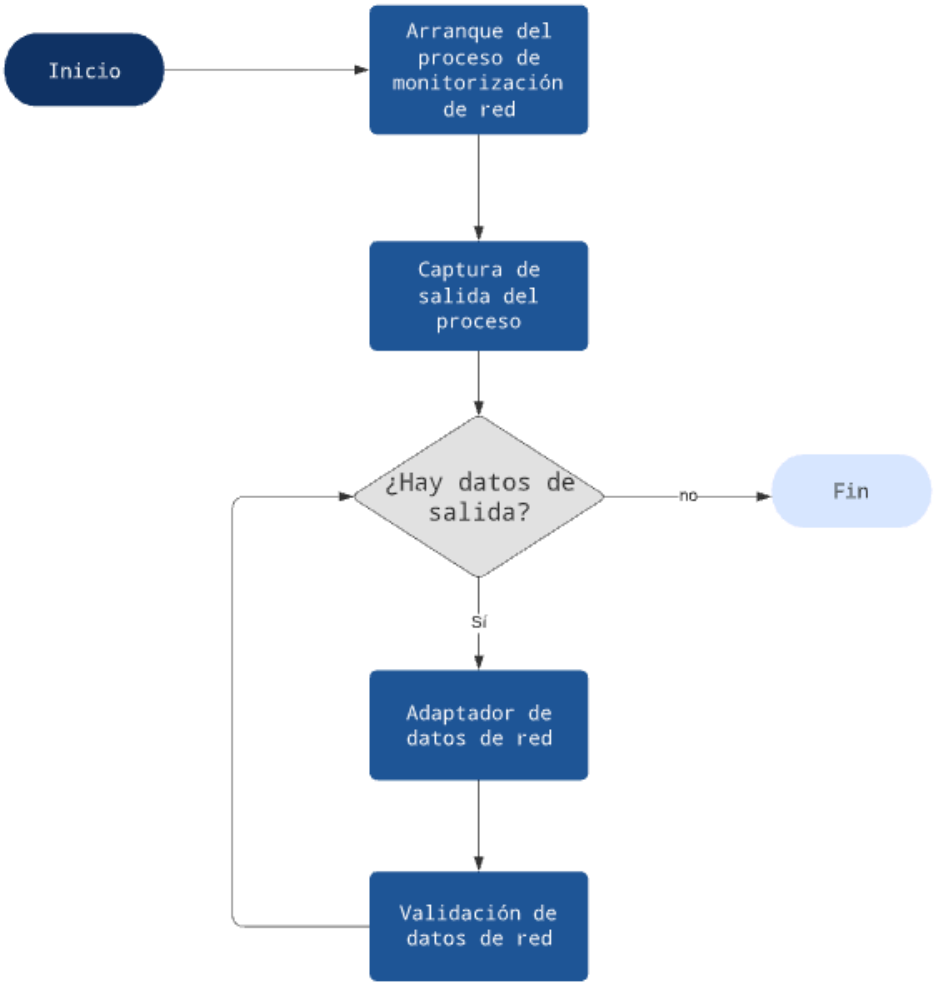

Figura 17. Diagrama de flujo de monitorización de datos

La monitorización de red comienza con el arranque del servicio JAVA que se está detallando. En la figura 17 se detalla el algoritmo empleado para esta labor. Como se ha mencionado en puntos anteriores de este documento, para poder capturar el flujo de red que circula por el dispositivo concentrador de la solución propuesta, se hace uso de la herra-

mienta “tcpdump”. La llamada ha este proceso se hace desde JAVA y se captura la salida del mismo obteniendo así los siguientes datos:

- IP que realiza la comunicación.
- Puerto desde donde se realiza la comunicación.
- IP de destino de la comunicación.
- Puerto de destino de la comunicación.
- Protocolo de la comunicación.

Los datos se obtienen como resultado de monitorizar la interfaz de red que proporciona el dispositivo concentrador que forma parte de la solución y de aplicar un filtro por tamaño de paquetes (inferior a 61). Cada vez que se obtiene un dato de la monitorización éste es analizado para ser validado por el proceso de validación de red.

Para comprobar que el proceso de monitorización sigue activo, se ha implementado una tarea específica. Esta tarea se ejecuta cada 60 segundos. Como ya se ha dicho es la responsable de comprobar el funcionamiento del proceso de monitorización. Como función de salud del sistema, al detectar que el proceso sigue activo, lo detiene por un periodo de tiempo de 5 segundos. De esta manera se previenen problemas derivados de posibles saturaciones de servicio. El la figura 16 se muestra cómo al obtener datos de salida del proceso de monitorización, estos pasan por un adaptador que los convierte al formato necesario por el sistema de validación. Después de este proceso, se ejecuta la validación de datos de red. Este proceso se detalla a continuación en un punto aparte por resultar de gran importancia para el funcionamiento de la propuesta.

4.4.1.4 Validación de datos de red

En el punto anterior se ha detallado el proceso para obtener los datos referentes al flujo de red mediante monitorización de la misma. El siguiente punto trata del análisis que se realiza sobre estos datos de red para determinar de esta manera si la comunicación es lícita o no. La figura 18 muestra el algoritmo en forma de diagrama de flujo diseñado para realizar esta tarea.

Al comiendo del proceso de validación de tráfico de red, se debe identificar el sentido de este. Para ello se comprueba que el destinatario, IP destinataria, forma parte de las cámaras conectadas. Esta información ha sido obtenida mediante el proceso ya detallado de detección de dispositivos. En caso de ser así, se entiende que la comunicación es entrante, y se ejecutarán las validaciones específicas para este sentido del tráfico de la red. En caso contrario se realizarán las comprobaciones para en tráfico saliente.

El tráfico entrante, deberá pasar por 4 validaciones diferentes que serán detallados en puntos independientes:

- Comprobación de listas blancas.
- Comprobación de IPs maliciosas.
- Validación de denegación de servicio (DDoS).
- Validación de origen de la petición.

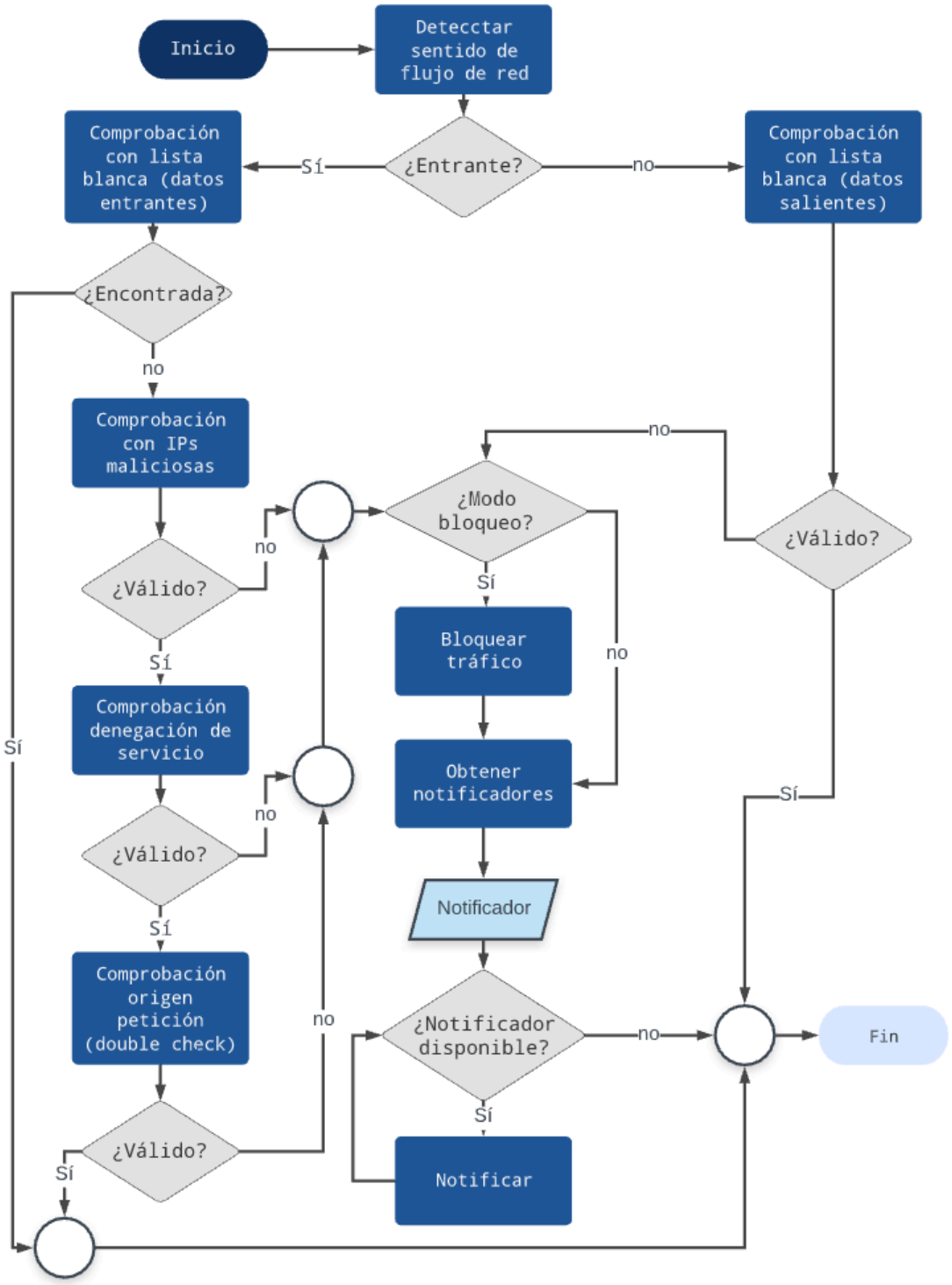

Figura 18. Diagrama de flujo del proceso de validación de comunicación red

Cada uno de los validadores tiene un estado independiente por cada cámara IP. Es decir, La relación cámara-validador tiene un estado que determina las acciones a ejecutar. Los estados posibles son:

- Inactivo
- Monitorización
- Bloqueo

Si un validador para una determinada cámara se encuentra inactivo, no realizará comprobación alguna y se interpretará como válido el análisis de la red realizado por este validador. Si el validador se encuentra en modo monitorización, al detectar tráfico ilícito (no ha superado la validación) se notificará a los usuarios registrados en el sistema mediante los notificadores. Estos notificadores hacen uso de los canales email y la herramienta Firebase Cloud Messaging. Un validador en modo monitorización no reclama el bloqueo de la red para la IP infractora, sólo notifica la infracción. Si el validador se encontrase en modo bloqueo, sí se reclamaría la denegación de red para la IP del tráfico analizado. Si los datos a analizar son interpretados como tráfico saliente, la validación a realizar es la comprobación en listas blancas de la IP destinataria.

4.4.1.5 Validador listas blancas

La comprobación de la IP entrante o saliente (dependerá del sentido del tráfico detectado) contra una lista blanca generada por el sistema es la primera validación que se realiza por el validador. Si el validador está activo para la cámara involucrada en la comunicación, se comprobará que la IP analizada exista entre las registradas como permitidas (registrada en la lista blanca). En caso de encontrarse en la lista, el este validador avisa al proceso de validación de que es una IP permitida, y por lo tanto exenta de otras validaciones. Al ser una IP de confianza, se infiere que no realizará acciones ilícitas y de esta manera se ahora

en procesamiento. Si no se encuentra en la lista de confianza, las demás validaciones son ejecutadas.

Este validador, además de la comprobación contra la lista de confianza, realiza labores de aprendizaje. El proceso de aprendizaje consiste en registrar el tráfico existente durante un periodo determinado de tiempo. Toda IP registrada en el proceso de aprendizaje se añade a la lista de confianza. El periodo de aprendizaje comprende la primera semana de cada cámara detectada en el sistema.

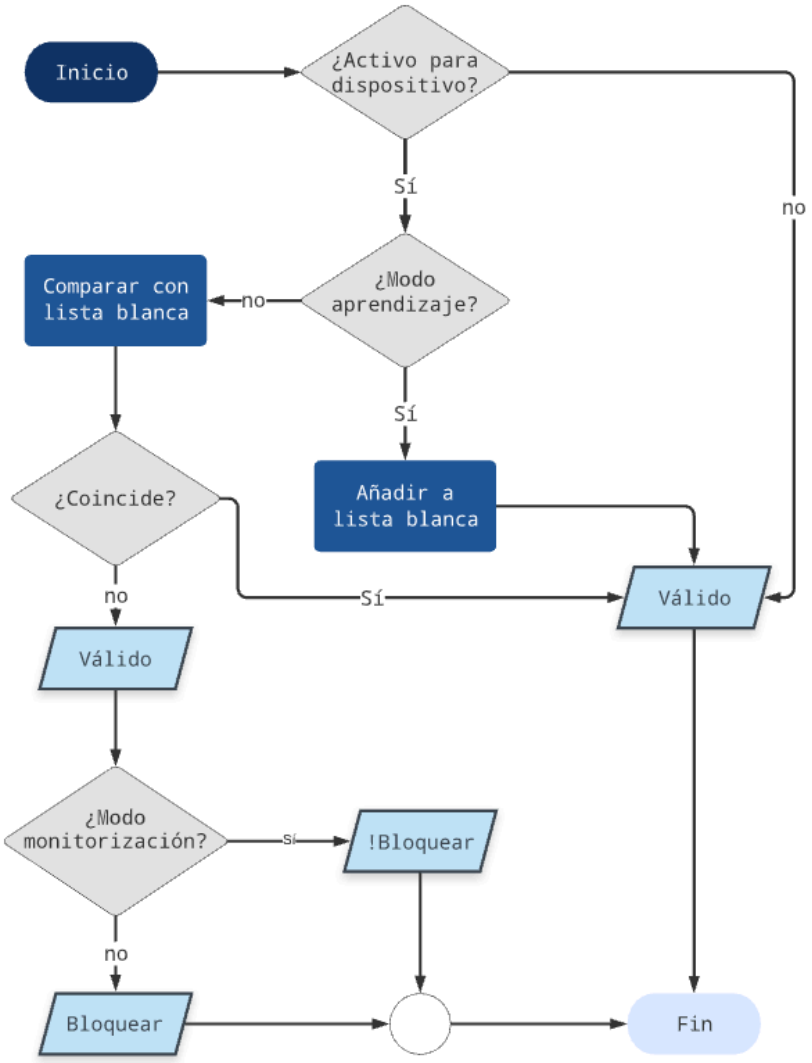

Figura 19. Diagrama de flujo del validador por lista blanca.

4.4.1.6 Validador IPs maliciosas

Como se ha comentado anteriormente, existen bases de datos que contienen IPs que son utilizadas para realizar ataques. El sistema maneja dos proveedores conocidos para obtener esta información. En la figura 20 se muestra el algoritmo para impedir que el tráfico relacionado con estas IPs reconocidas como maliciosas no se permita en el sistema.

Al igual que en el resto de validadores, para proceder a la comprobación del flujo de red en el que una cámara IP registrada está involucrada, es necesario que el validador esté activo. En caso de no estarlo se devolverá como resultado del análisis la indicación de IP válida. Al disponer ya de la información necesaria para esta validación, la comprobación radica en ver si la IP que realiza la petición existe dentro de la lista de IPs maliciosas. En caso de no existir, el validador aceptará la comunicación. En caso contrario, la petición se considera ilícita. Una petición ilícita será siempre notificada. Si el estado del validador para la cámara involucrada es de bloqueo, el validador solicitará el bloqueo de la IP entrante además de la notificación.

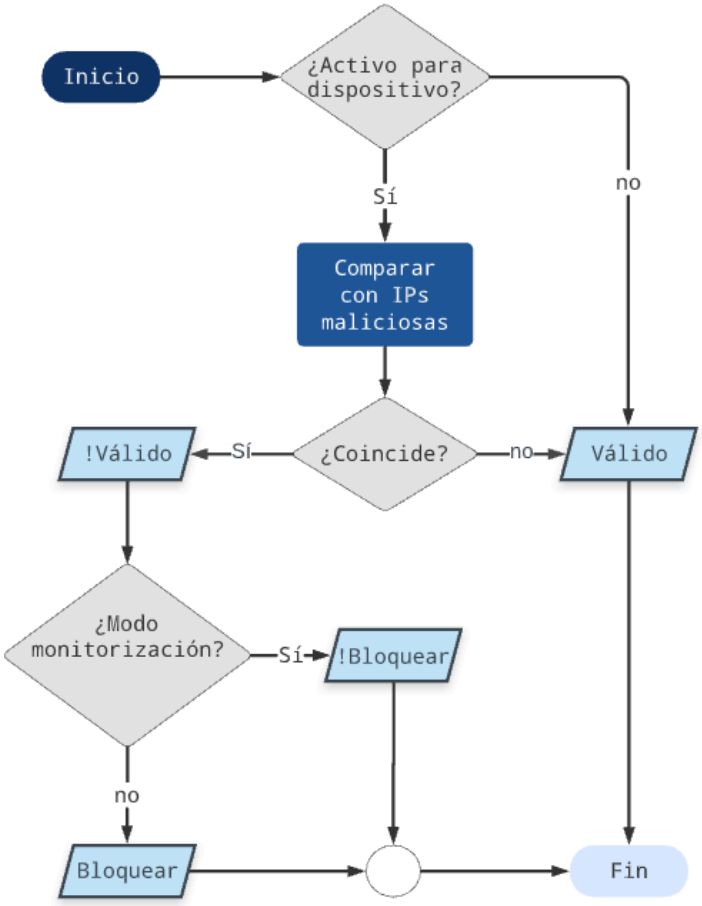

Figura 20. Diagrama de flujo del validador de IPs maliciosas

4.4.1.7 Denegación de servicio por saturación

Como se ha mencionado anteriormente, uno de los ataques que se realizan a los sistemas en general y entre ellos a cámaras IP es el de denegación de servicio (DDoS). Para hacer frente a este ataque, se monitorizará la actividad de red. En la figura x se detalla la metodología empleada para la validación de denegación de servicio.

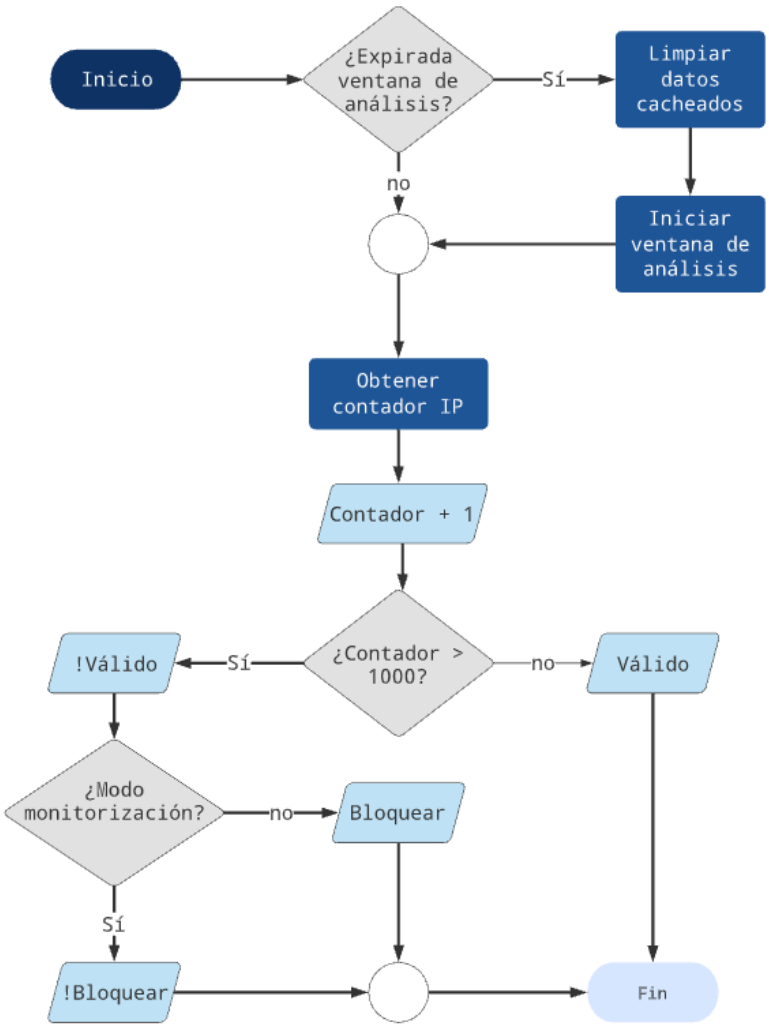

Figura 21. Diagrama de flujo de validador de DDoS.

Al requerirse la comprobación por el validador de denegación de servicio, siempre que este esté activo para la cámara relacionada con el tráfico de red detectado, se analiza el número de peticiones registradas en un periodo de tiempo determinado. Se ha considerado que las cámaras IP no deben recibir más de 1000 peticiones por minuto. Con estos datos, el concentrador mantiene un recuento de las peticiones por destinatario en una ventana de tiempo de un minuto. Si se sobrepasa el valor de 1000 peticiones en esa ventana de tiempo, se considerará la conexión como ilícita. En caso de estar en modo bloqueo para la cámara relacionada con la comunicación, el validador solicita el bloqueo de la comunicación para

la IP que genera el tráfico entrante. Tanto si está en modo bloqueo como en modo monitorización, se notificará de la situación a los usuarios registrados mediante email y notificación móvil.

4.4.1.8 Double check validator

El siguiente validador es el más complejo de la solución, ya que requiere de dispositivos externos para su funcionamiento. La figura 22 detalla la lógica diseñada para realizar esta comprobación. El sistema hace uso de Firebase Cloud Messaging de Google para el envío de notificaciones y una aplicación móvil para responder a esta notificación.

La doble comprobación de solicitudes se basa en la confirmación de que se ha realizado la petición desde uno de los dispositivos móviles, clientes de las cámaras IP, registrados en el sistema. El dispositivo concentrador, al recibir una petición que no se encuentre entre las autorizadas ni entre las maliciosas conocidas, enviará una notificación a los dispositivos clientes. En un principio, para esta labor se han considerado dos medios diferentes: APIs y Firebase Cloud Messaging (FCM).

Al hacer uso de una API para la comunicación entre los dispositivos es necesario conocer la dirección de los mismos. Esto supone un aumento de la complejidad debido al mantenimiento necesario de las direcciones IP. A su favor, la petición está controlada al carecer de servicios intermediarios y la respuesta es prácticamente inmediata.

Mediante FCM, los dispositivos subscritos podrán recibir notificaciones privadas. Como se ha comentado en puntos anteriores, esta solución permite la comunicación sin conocer la

dirección IP del dispositivo móvil. En la herramienta antecesora de FCM, Google Cloud Messaging, el punto negativo era que la funcionalidad quedaba limitada a dispositivos Android y el tiempo de envío de la notificación era variable. Estos factores han sido solucionados por el proveedor del servicio. Es por esto que la solución adoptada ha sido FCM.

Al requerirse una validación de este tipo, el validador para la cámara involucrada está activo, se accede a una de las claves generadas para esta cámara. Es importante anotar que cada cámara dispone de 100 claves y que la elección de la accedida es totalmente aleatoria. El índice de esta clave se envía a los dispositivos móviles clientes para que respondan con su valor, validando así la petición, junto con la confirmación de que se ha realizado la petición desde el dispositivo. De esta manera se certifica que se debe permitir la comunicación detectada. Una vez validada la comunicación, se mantendrá su validez durante el tiempo establecido de 15 minutos.

Es posible que no se obtenga respuesta por parte de ningún cliente o que esta se demore. Para la demora se establece un periodo de 15 segundos en los que la comunicación sigue permitida. En caso de no tener respuesta en este tiempo, se identificará la comunicación de red como ilícita. En este escenario, si el estado del validador para la cámara relacionada es de monitorización, se enviarán las notificaciones pertinentes. En caso de que el validador se encuentre en modo bloqueo, se procederá a inhabilitar el acceso a la red para la IP que ha realizado la petición.

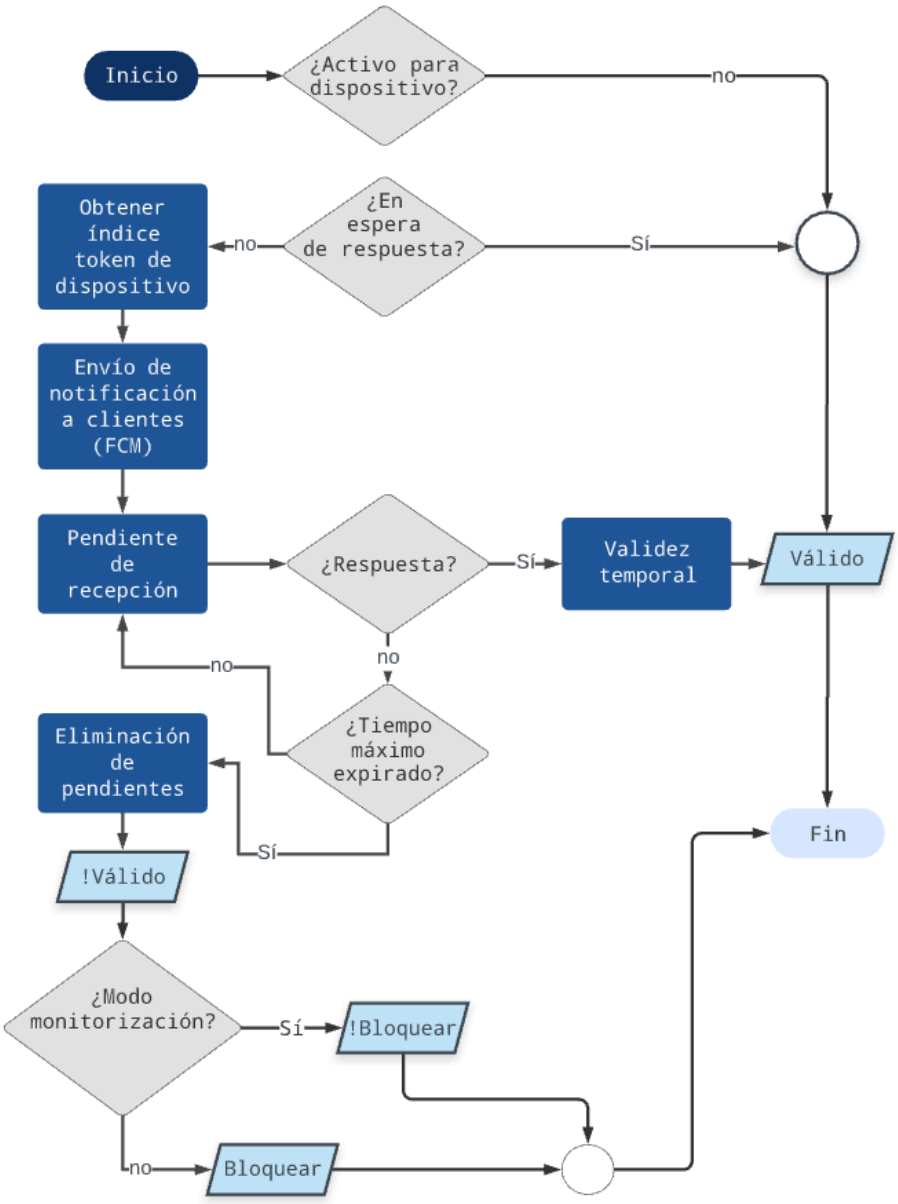

Figura 22. Diagrama de flujo del validador por doble petición

4.4.2 API de acceso

Para la configuración del servicio de validación y para interactuar con él, se han implementado una serie de servicios a modo de API Rest. A continuación se detallan estos servicios expuestos por el servicio JAVA.

4.4.2.1 Servicio para la respuesta de double check

POST /check/use

Este servicio rest es el encargado de atender las peticiones enviadas desde de los dispositivos móviles suscritos al sistema como respuesta a la notificación del validador por doble check. El servicio atiende a peticiones de tipo POST. Como parte de la cabecera debe viajar el token de identificación del emisor de la petición. El nombre de la cabecera es "camip-token". En un escenario normal, este token es enviado desde el servicio de validación al sincronizar el dispositivo móvil con el sistema.

Los datos enviados en el cuerpo de la petición representan los datos requeridos por la solicitud del validador.

4.4.2.2 Servicio para la sincronización de dispositivos móviles

GET /device/sync

Este servicio registra en el sistema de validación los dispositivos que formarán parte del mismo. Al atender esta petición, el dispositivo solicitante queda registrado con un código de identificación. Este código se envía como parte de la respuesta para ser utilizado como clave de autorización añadiéndola a la cabecera de las siguientes peticiones. Para completar la respuesta, se enviarán los identificadores de las cámaras IP registradas y sus claves de identificación que serán solicitadas en el servicio de doble comprobación. En caso de estar ya registrado y se solicite una actualización de los datos, el identificador existirá en la cabecera "camip-token" de la petición y no se registrará de nuevo. Esta petición se ejecuta desde la aplicación móvil proporcionada. Si la petición se realiza sin la información de la

cabecera de identificación, se realiza una comprobación de origen de la misma. En este caso solo se permite esta solicitud desde la propia red del dispositivo concentrador.

4.4.2.3 Servicio de dispositivos registrados

GET /net/devices

Servicio para obtener los datos básicos de las cámaras IP registradas. Para acceder a este servicio el solicitante debe estar conectado a la red proporcionada por el dispositivo concentrador. La respuesta se compone del identificador de cada cámara, ip en la que atiende, su dirección mac y nombre asignado.

4.4.2.4 Servicio de actualización del nombre de la cámara

PUT /net/devices/{id}

Servicio de tipo PUT para actualizar el nombre para identificar la cámara. Para acceder a este servicio el solicitante debe estar conectado a la red proporcionada por el dispositivo concentrador. En la solicitud se indica el identificador de la cámara a actualizar y en el cuerpo de la misma el nombre dentro del parámetro “name”.

4.4.2.5 Servicio para obtener el estado de los validadores

GET /devices/{id}/status

Este servicio de tipo GET es el encargado de proporcional la información del estado de cada validador asociado a una cámara IP. Para acceder a este servicio el solicitante debe estar conectado a la red proporcionada por el dispositivo concentrador. Como parte de la solicitud se envía el identificador anteriormente obtenido de la cámara a consultar. La respuesta identificará a cada uno de los 4 validadores (WHITELIST, MALICIOUSIP, DDOS y DOUBLECHECK) e informará de su estado (STOP, MONITOR, BLOCK).

4.4.2.6 Servicio de actualización de estado de validadores

PUT /devices/{id}/status/{validator}

Servicio de tipo PUT para actualizar el estado de una validador asociado a una cámara. Para acceder a este servicio el solicitante debe estar conectado a la red proporcionada por el dispositivo concentrador. En la solicitud se indica el identificador de la cámara y nombre del validador a actualizar. El estado que se aplica se envía en el cuerpo de la misma en el parámetro “status”.

4.4.2.7 Servicio para monitorizar comunicaciones

GET /net/flow

Este servicio de tipo GET devuelve las IPs que han realizado comunicación con alguna de las cámaras las últimas 48 horas. Al igual que en otras peticiones de administración, para acceder a este servicio el solicitante debe estar conectado a la red proporcionada por el dispositivo concentrador.

4.4.2.8 Servicio de obtención de reglas

GET net/rules

El servicio de tipo GET net/rules es el punto de acceso para conocer las reglas de acceso que se han aplicado como resultado del análisis de las comunicaciones. Debido a su naturaleza de administración, para acceder a este servicio el solicitante debe estar conectado a la red proporcionada por el dispositivo concentrador. La respuesta es una lista en la que se identifica a cada regla con su código identificador, IP a la que afecta y sentido de la comunicación afectada (entrante o saliente).

4.4.2.9 Servicio para eliminar reglas

DELETE net/rules/{id}

Servicio de tipo DELETE. Este servicio permite eliminar las reglas de control de comunicaciones creadas por el sistema propuesto. El servido comprueba que el solicitante se encuentre conectado a la red proporcionada por el dispositivo concentrador como método de gestión de la seguridad.

4.4.3 Aplicación móvil de control de accesos

Para apoyar al servicio de validación en el proceso de dotar de seguridad la instalación de cámaras IP domésticas, se ha implementado una aplicación móvil Android. Esta aplica-

ción desempeña una labor crucial a la hora de ejecutar la validación “Double check” para comprobar que el acceso a la cámara se ha realizado desde uno de los dispositivos móviles autorizados. A continuación se detalla el diseño de implementación de esta aplicación móvil.

La aplicación móvil requiere mantener los datos cuando esta está cerrada, ya que maneja información de claves para la conexión e identificación del propio dispositivo móvil y de las cámaras IP registradas. Para llevar a cobo esta tarea, se ha optado por la tecnología de persistencia embebida SQLite. El uso esta herramienta de es muy recomendado por su naturaleza embebida y excelente respuesta frente a las necesidades funcionales requeridas por las aplicaciones.

A continuación se detallan las funciones que desempeña la aplicación móvil para cumplir con su labor en la comprobación “Double check”.

4.4.3.1 Sincronización con servicio de validación

El sistema propuesto realiza una serie de peticiones entre el servicio de validación y la aplicación móvil que se está detallando en este apartado. Para poder confiar el en intercambio entre estos dos módulos la comunicación debe mantener indicadores de seguridad. Estos indicadores son claves o tokens de seguridad generados por el servicio de validación JAVA. La aplicación web debe conocer estos datos de seguridad. Para ello se ha diseñado e implementado el algoritmo mostrado en la figura 24. Como puntualización importante y algo que se ha comentado en puntos anteriores, para poder realizar la tarea de sincroniza-

ción, el dispositivo móvil en el que la aplicación está instalada debe de estar conectado a la red proporcionada por el dispositivo concentrador.

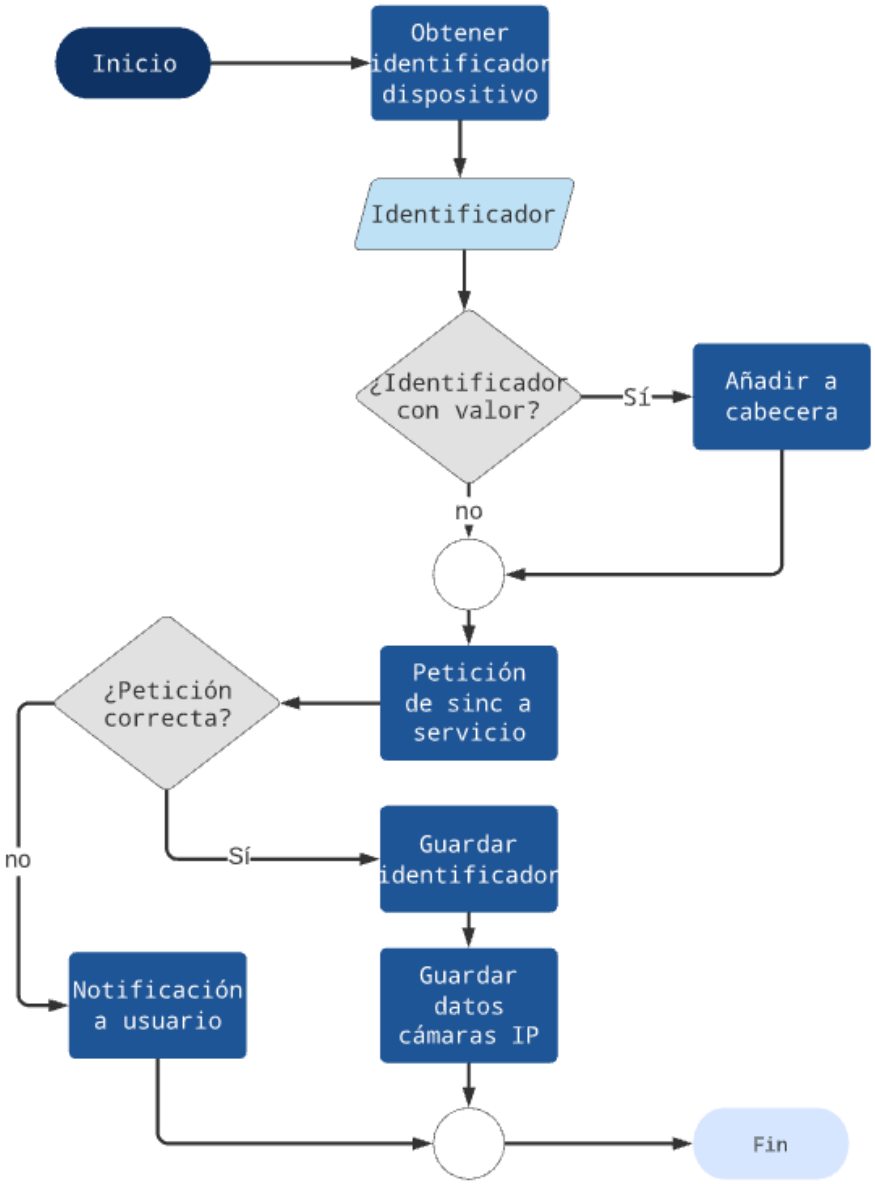

Figura 23. Diagrama de flujo de sincronización con servicio de validación.

Para comenzar con la sincronización, el usuario pulsa el botón de la aplicación móvil destinado a esta acción. Esto envía una petición de sincronización al servicio de validación. El servicio genera un código único que identifica a el dispositivo móvil y será enviado co-

mo respuesta. Este identificador es el que se utilizará en futuras peticiones. Además del identificador del dispositivo móvil, el servicio devuelve los datos de las cámaras IP registradas en el sistema. Estos datos incluyen la identificación del canal de notificaciones, la dirección mac para identificar cada una de las cámaras, el nombre identificativo y las cien claves asociadas a cada cámara. Estas claves son imprescindibles para la validación "double check".

Para obtener los datos de cámaras registradas después de la primera sincronización, basta con realizar la misma acción. En este caso se añade el identificador o token del dispositivo móvil a la cabecera de la petición. El servicio de validación enviará la misma respuesta que en el caso inicial, pero no realizará registro alguno de nuevo dispositivo móvil.

4.4.3.2 Registro de aplicaciones clientes de cámaras IP

Un dispositivo móvil puede tener multitud de aplicaciones para manejar las cámaras IP instaladas en el hogar. Es preciso que la aplicación de apoyo a la validación conozca cuáles son estas aplicaciones. Esto se debe a que el servicio de validación, al enviar la notificación para la comprobación "Double check", quiere certificar que la comunicación con la cámara IP se ha iniciado desde el dispositivo móvil en donde se ha instalado la aplicación. Para identificar las aplicaciones relacionadas con cámaras IP se necesita la interacción del usuario para que este identifique cada una de ellas. Para facilitar esta tarea, en una de las vistas de la aplicación se muestra una lista con todas las aplicaciones instaladas en el dispositivo móvil. Para obtener esta lista se hace uso del manejador de paquetes (PackageManager) propio del sistema operativo Android. Esta clase es la contenedora de la información necesaria para esta tarea. De aquí se obtiene la lista de aplicaciones instaladas. Esta

lista se cruza con la lista de aplicaciones ya registradas (en un inicio será vacía). Cada aplicación que coincida se marca como ya registrada para que el usuario conozca este dato. Una vez mostrada la lista, el usuario registra o elimina del registro la aplicación sobre la que pulse. La figura 25 muestra este algoritmo en detalle para facilitar su comprensión.

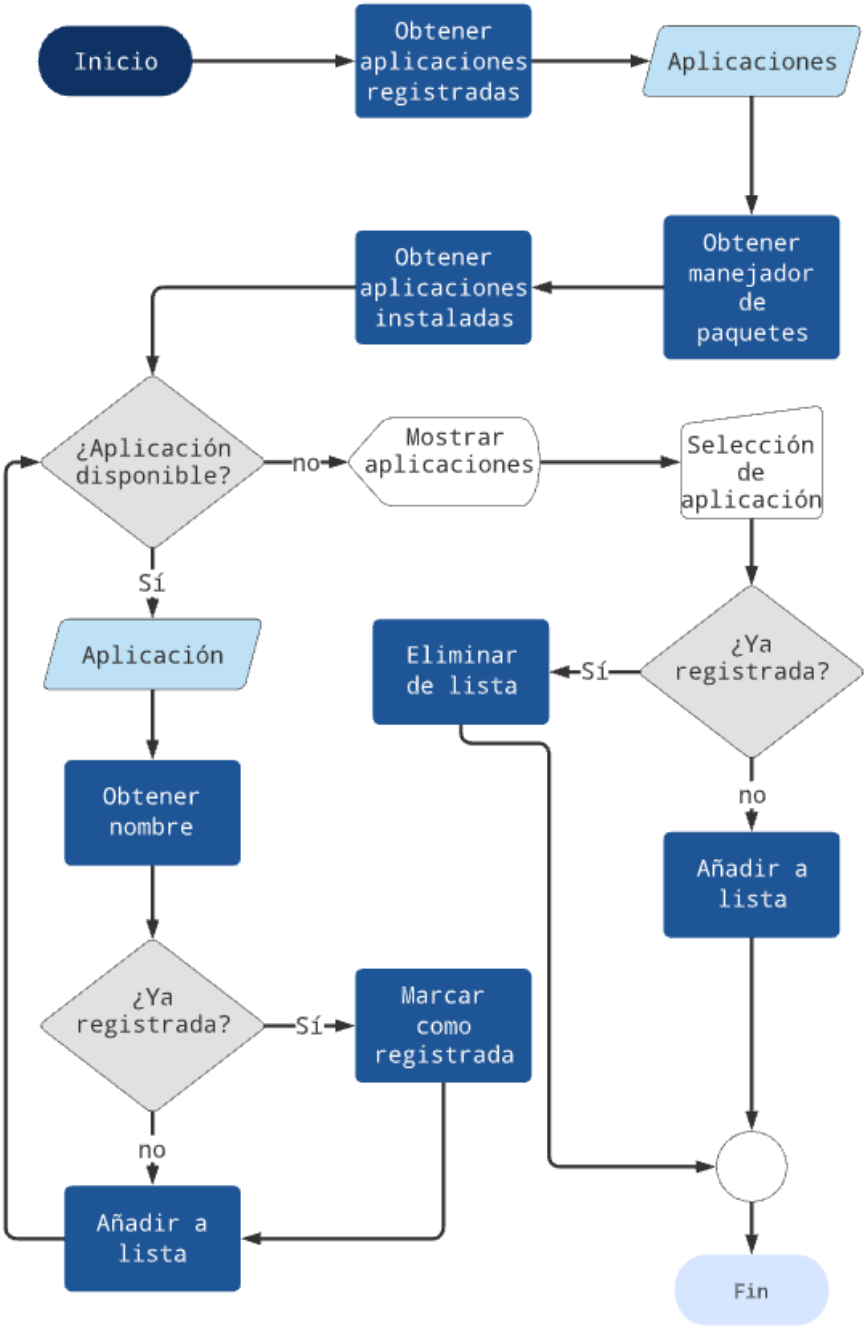

Figura 24. Diagrama de flujo de registro de aplicaciones clientes de cámaras IP

4.4.3.3 Validación Double check

La validación “Double check” es la tarea principal de la aplicación móvil. Las funciones anteriormente detalladas de la aplicación móvil son instrumentos para obtener datos necesarios para poder ejecutar esta validación. En la figura 26 se muestra en detalle la metodología seguida para dar respuesta a esta necesidad.

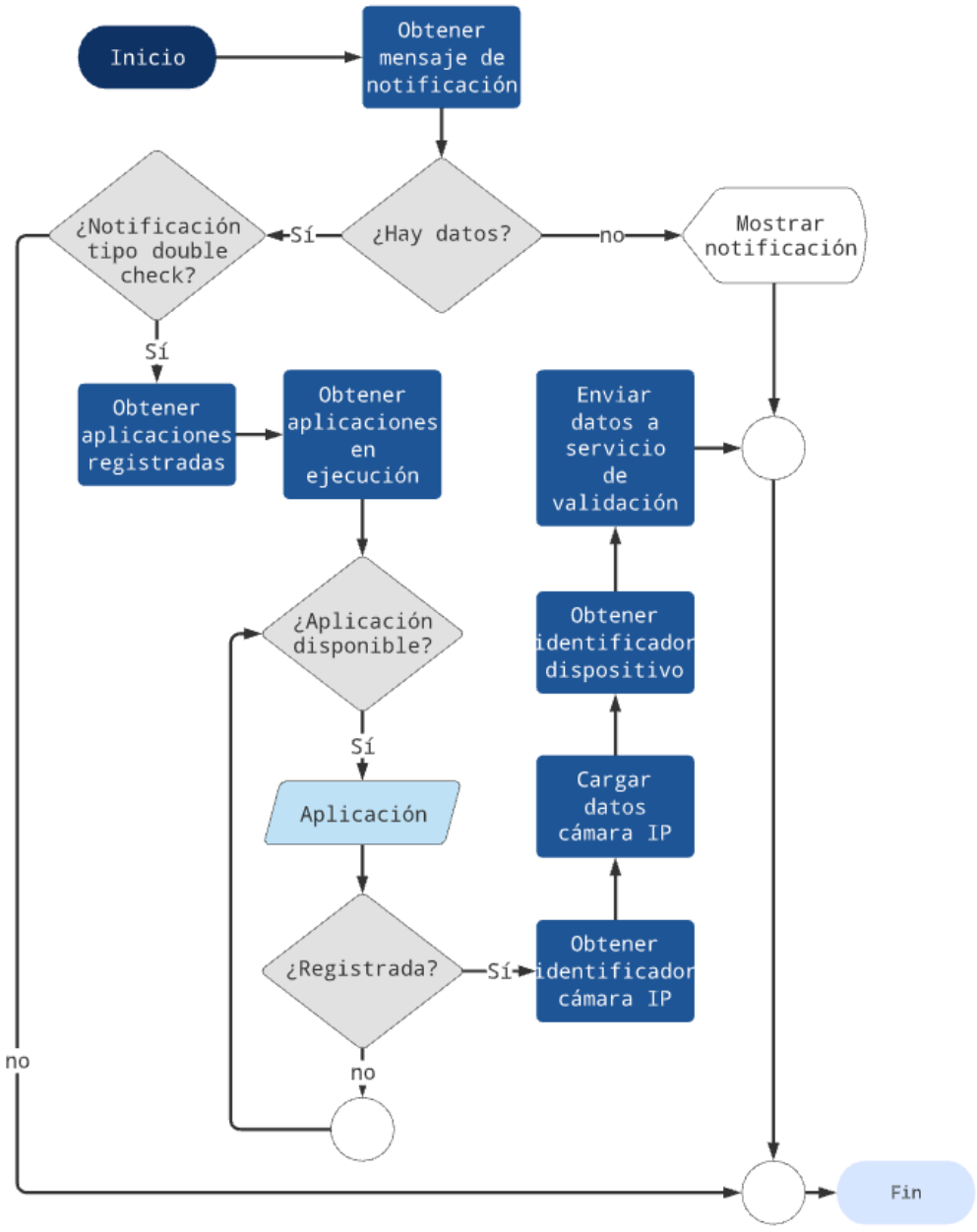

Figura 25. Diagrama de flujo de validación Double check

La validación comienza desde el servicio de validación. Este envía una notificación mediante Firebase Cloud Messaging que será atendida por todos los móviles suscritos al canal "camip-{mac-dispositivo-concentrador}". Todos los móviles con la aplicación Android instalada y sincronizados con el servicio de validación están suscritos a este canal. Cuando una notificación es recibida en la aplicación, se verifica que es una notificación de "double check". Esto significa que dicha notificación tiene datos (no es únicamente título y mensaje) y que uno de estos datos identifica el tipo correcto de notificación. En caso de no cumplir estos requisitos, la información recibida se gestionará como una notificación común. El sistema también maneja este tipo de notificaciones para alertar de posibles ataques detectados. Una vez identificada la notificación de "double check" esta comienza a ser procesada. Con la ayuda del manejador de paquetes se obtiene una lista de aplicaciones, esta vez las que se encuentran en ejecución (ya sea en primer o en segundo plano). Estas aplicaciones son comparadas con las aplicaciones que han sido previamente identificadas por el usuario y registradas en la aplicación para poder determinar que una de ellas está en uso, y que por lo tanto es muy probable que la comunicación con la cámara IP se haya iniciado desde el terminal móvil. En este punto se comienza a responder al servicio de validación con los datos requeridos. Estos datos son los siguientes:

- Identificador del dispositivo móvil
- Identificador de la cámara solicitado por el servicio de validación
- Ip desde donde se realizó la comunicación
- Ip de destino de la comunicación
- Dirección mac de la cámara IP

El identificador del dispositivo móvil se ha obtenido como respuesta a la sincronización, y es requerido por el servicio para identificar al terminar que envía la respuesta.

El identificador de la cámara se obtiene de la lista de identificadores asociados a cada cámara que son cargados en la sincronización. Como ya se ha comentado en puntos anteriores, cada cámara tiene cien identificadores asociados. Para evitar que se suplante la identidad del usuario al responder a esta validación, cada vez se solicitará un identificador diferente de entre estos cien por parte del servicio de validación. La solicitud de esta información se lleva a cabo mediante el código que identifica a el propio identificador y es respondida con su valor.

La IP de origen y destino de la comunicación y la dirección mac de la cámara involucrada son datos necesarios para identificar la solicitud del proceso de validación por parte del servicio.

4.5 Funciones adicionales

4.5.1 Definición de subred

Uno de los métodos de protección contra accesos no deseados a las cámaras IP es la separación entre la conexión de estas y el resto de dispositivos de la red de comunicaciones en general. El dispositivo concentrador propuesto crea una subred en la que se conectarán las cámaras IP. Esta subred quedará definida por un rango. Como parte del control al analizar la red, ninguna transmisión con destino local desde las cámaras podrá realizarse fuera de esta subred o rango de IPs. Este control se ejecuta mediante reglas definidas para iptables que funciona a modo de firewall.

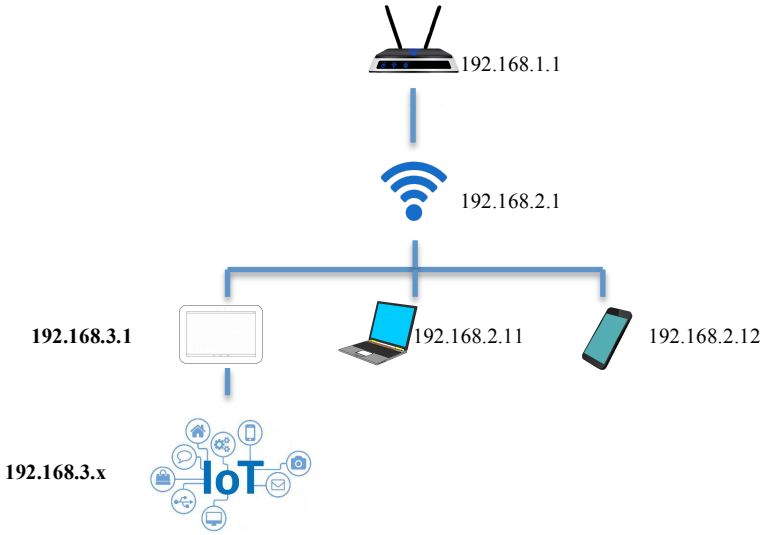

Figura 26. Esquema generación subred.

Gracias a la creación de la subred, se evita que toda la red doméstica quede expuesta en caso de que un atacante acceda a uno de los dispositivos registrados en el sistema propuesto.

4.5.2 Monitorización de accesos

En cualquier sistema informático se recomienda la monitorización de actividad. El dispositivo propuesto realiza esta labor gracias al proceso de monitorización de la red. El servicio de validación de red registra y guarda en base de datos todas las IPs que se han comunicado con el dispositivo las últimas 48 horas. Este dato es útil para conocer la realidad de las comunicaciones entre clientes y cámaras IP

5 Evaluación de la solución

A continuación se detalla la evaluación planteada y ejecutada para la solución propuesta en este trabajo.

5.1 Contextualización

Para demostrar y evaluar la solución propuesta se han planteado una serie de pruebas. Estas pruebas pretenden demostrar la viabilidad del dispositivo de validación abordando los puntos relevantes del mismo.

Las pruebas se han separado en tres grupos:

- **Pruebas funcionales**: En ellas se prueba la funcionalidad de la solución, que no es otra que validar las conexiones permitiendo o denegando el acceso a las cámaras IP.

- **Pruebas de rendimiento**: Estas prueba se han enfocado a la evaluación del sistema externo utilizado para las notificaciones y validación "double check". Es por lo tanto la demostración de la viabilidad del uso de Firebase Cloud Messaging como herramienta de notificaciones.

- **Monitorización de red**: Con estas pruebas se pretende detectar si la monitorización de red es correcta para la solución propuesta. Se desea determinar si la estrategia adoptada permite conseguir el objetivo, que no es otro que securizar la instalación de cámaras IP en el entorno doméstico.

Para esta evaluación se ha implementado prototipo de dispositivo concentrador, servicio de validación y aplicación Android.

Para ejecutar dicha evaluación se ha utilizado una Raspberry pi 3 modelo B+ que creará un punto de acceso para la conexión de las cámaras IP. Para la configuración del punto de acceso se ha configurado el dispositivo siguientes las directrices del propio fabricante[38]. Como resultado de esta configuración se obtiene una red representada en el siguiente esquema (figura 28). La Raspberry Pi queda representada como “RPi”

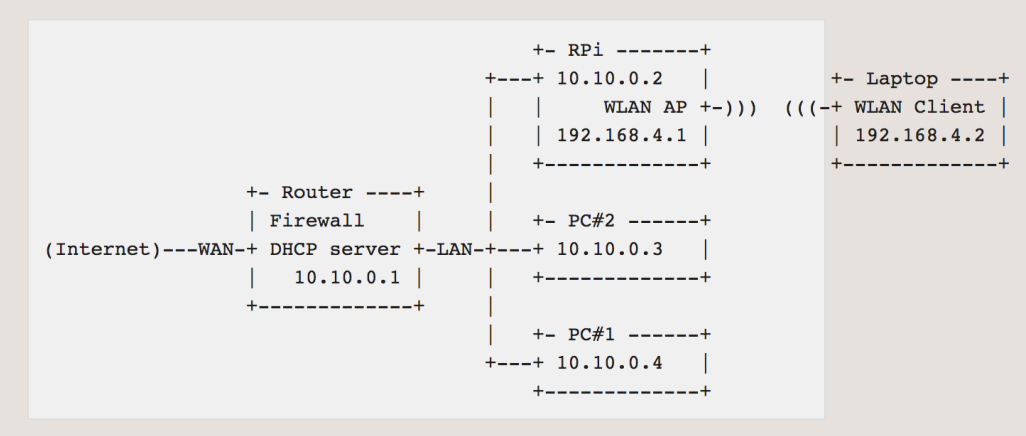

Figura 27. Esquema de red tras configuración de Raspberry Pi. Fuente: www.raspberrypi.org

A su vez, la Raspberry Pi será el dispositivo concentrador encargado del análisis y gestión de los accesos a las cámaras IP. Se ha instalado el sistema operativo raspbian para la implantación de los módulos requeridos por la solución y explicados en el apartado propuesta.

La cámara utilizada para llevar a cabo las pruebas ha sido Mi Home Security Camera 360º de Xiaomi.

Figura 28. Mi Home Security Camera 360º. Fuente: www.mi.com

5.2 Pruebas funcionales

5.2.1 Descripción

Para comprobar la efectividad de la propuesta ante el problema de seguridad planteado en una instalación doméstica de cámaras IP, se han ejecutado una serie de pruebas de la solución planteada. En dichas pruebas se ha evaluado el comportamiento del servidor Wi-Fi de validación frente a cuatro casos diferentes. A continuación se explican los casos de prueba:

5.2.1.1 Dispositivo registrado efectúa una solicitud

El primer caso de uso se refiere a la utilización normal de la solución para acceder a una cámara IP. Para realizar esta prueba se ha vinculado la cámara IP de Xiaomi comentada en el punto anterior. Para poder realizar la comprobación “Double check” se ha vinculado un dispositivo móvil en el sistema haciendo uso de la aplicación móvil Android. En esta aplicación móvil se ha seleccionado la aplicación del proveedor de la cámara IP, en este caso

“Xiaomi Home”, que realiza la conexión con dicha cámara para detectar su funcionamiento. Este escenario sería suficiente para comenzar con el funcionamiento normal del sistema, pero para evitar el proceso de aprendizaje y que directamente se añada el dispositivo móvil a la lista de confianza, se ha desactivado el comprobador de listas blancas. Ahora la conexión pasará por el resto de comprobaciones de seguridad.

Como resultado de esta prueba, la conexión debe ser permitida.

5.2.1.2 Dispositivo no registrado efectúa una solicitud legítima

Este caso de uso es similar al caso de uso 1. Este escenario de ejecución también contempla la vinculación de la cámara IP al sistema planteado, pero no se vincula ningún dispositivo móvil mediante la aplicación proporcionada. Al estar activada (en modo bloqueo) la comprobación “Double check”, la cual solicita confirmación de inicio de comunicación desde uno de los dispositivos móviles configurados, se espera que la comunicación sea bloqueada. Al no existir aplicación móvil vinculada, el sistema no recibirá la confirmación requerida.

Como ya se ha dicho, el resultado de esta prueba debe ser el bloqueo de la conexión.

5.2.1.3 Dispositivo registrado sin configurar efectúa una solicitud legítima

El caso de uso número 3 se refiere a una configuración similar a la del caso de uso número uno. La diferencia entre estos dos casos de prueba radica en que la aplicación móvil estando vinculada a el sistema, no ha sido configurada y por lo tanto no se ha seleccionado la aplicación móvil del proveedor de la cámara IP que inicia la comunicación, en este caso

"Xiaomi Home". Al estar la comprobación "Double check" activa en modo bloqueo, se espera que como resultado de esta prueba, la comunicación quede bloqueada. La acción correcta frente a este caso es el rechazo o bloqueo de la petición de conexión.

5.2.1.4 Ataque DDoS desde un dispositivo registrado que efectúa múltiples solicitudes legítimas

Este caso de uso se refiere a tratar de generar una denegación de servicio mediante múltiples peticiones a la cámara IP. Este escenario podría bloquear el sistema de aviso de la cámara e impediría la conexión a ella para visionar las imágenes. Para realizar esta prueba se ha vinculado la cámara IP de Xiaomi "My Home Security Camera 360º" al servidor Wi-Fi de la propuesta. Para evitar el proceso de aprendizaje y que directamente se añada el dispositivo móvil a la lista de confianza, se ha desactivado el comprobador de listas blancas. Para recibir la notificación de bloqueo de petición se ha vinculado la aplicación móvil que forma parte de la solución. Para esta prueba también es necesaria la identificación de la aplicación del proveedor de la cámara IP ya que la comprobación "Double check" debe ser correcta. Para facilitar reproducir un ataque DDoS, se han modificado los parámetros de la comprobación. No se permitirán más de diez conexiones en dos minutos. Con la configuración original, la comprobación bloquearía la las peticiones que superen la mil conexiones en un minuto. La prueba consiste por lo tanto de tratar de acceder a la cámara IP en más de diez ocasiones en dos minutos.

El resultado de esta prueba debe ser el bloqueo de la petición por DDoS.

5.2.1.5 Dispositivo reconocido como malicioso realiza una solicitud legítima

En el caso de uso número cinco se pretende reproducir el acceso a la cámara IP desde un dispositivo identificado como malicioso. Para ello es necesario vincular la cámara IP de Xiaomi al dispositivo Raspberry Pi descrito. Para evitar el proceso de aprendizaje y que directamente se añada el dispositivo móvil a la lista de confianza, se ha desactivado el comprobador de listas blancas encargado de esta tarea. Para recibir la notificación de bloqueo de petición, se ha vinculado al sistema la aplicación móvil que forma parte de la solución. Para poder reproducir el acceso de un dispositivo reconocido como malicioso, se ha añadido el dispositivo desde donde se realizará la conexión a la cámara IP a la lista de desconfianza.

Como resultado de esta prueba, la conexión no debe ser permitida.

5.2.2 Resultados obtenidos

Los resultados obtenidos de las pruebas mencionadas se muestran siguiente en la tabla de resultados:

Caso de uso	Petición aceptada	Petición rechazada	Acción correcta
1	X		X
2		X	X
3		X	X
4		X	X
5		X	X

Tabla 5. Definición y resultados de casos de prueba

Por lo tanto, las pruebas realizadas demuestran que funcionalmente la propuesta cubre las necesidades planteadas en este trabajo.

5.3 Pruebas de rendimiento

5.3.1 Descripción

Se han ejecutado pruebas para comprobar el tiempo de análisis requerido por dispositivo de validación. La importancia de esta prueba radica en que el resultado obtenido es crítico, ya que será el tiempo en el que la comunicación esté permitida aún sin ser legítima.

Para probar esta funcionalidad se ha planteado la ejecución de pruebas basadas en las realizadas en el estudio de Yavuz Selim, Bahadir Ismail y Murat Demirbas ya mencionado en este trabajo. En este caso la herramienta utilizada será FCM en lugar de GCM.

La prueba consta en medir el tiempo de respuesta en la situación de validación "Doublé check". El valor registrado representa el tiempo transcurrido desde que comienza la validación mencionada hasta que el propio servicio de validación obtiene la respuesta desde la aplicación móvil.

Para obtener resultados de diversos escenarios de ejecución, se han planteado tres variables:

- **N. notificaciones:** Número de notificaciones que se envían desde el servicio de validación.

- **N. móviles:** Número de dispositivos móviles vinculados con la solución y por lo tanto, a los que se enviará cada una de las notificaciones.

- **N. repeticiones:** Número de veces que se repetirá en envío de las notificaciones correspondientes al caso de prueba a el número de móviles fija en el mismo.

Para la confección de los casos de uso se han fijado los siguientes valores de las variables anteriormente citadas:

Caso	N. notificaciones	N. móviles	N. repeticiones
1	1	1	10
2	10	1	10
3	10	10	10

Tabla 6. Definición de escenarios test notificaciones

Los casos se ejecutarán por duplicado ya que el estado de la aplicación puede influir en el resultado obtenido. Los dos estados de aplicación a tener en cuenta son: aplicación en ejecución y aplicación en segundo plano. Para analizar mejor el resultado obtenido, se muestran los valores en términos de duración mínima, duración media y duración máxima.

5.3.2 Resultados obtenidos

La siguiente tabla muestra los resultados obtenidos de la ejecución de las pruebas definidas. Se muestran dos tablas, la primera con los resultados obtenidos al ejecutar las pruebas con la aplicación móvil en primer plano y la segunda con la aplicación en segundo plano.

Resultados de ejecución con aplicación en primer plano.

Caso	Mínimo(ms)	Máximo (ms)	Media (ms)
1	1457	4262	2468
2	1864	6324	4231
3	1793	8122	5212

Tabla 7. Resultados test notificaciones primer plano

Resultados de ejecución con aplicación en primer plano.

Caso	Mínimo(ms)	Máximo (ms)	Media (ms)
1	1322	3561	2237
2	1683	5427	3986
3	1868	7853	5294

Tabla 8. Resultados test notificaciones segundo plano.

Los resultados muestran que la herramienta seleccionada como sistema de notificaciones es adecuada, ya que el tiempo de envío requerido para la entrega de los datos es completamente aceptable.

En los casos más habituales, en los que existirá un único móvil para gestionar las cámaras IP, el tiempo medio máximo se sitúa cercano a los 4 segundos.

5.4 Monitorización de red

Se ha diseñado un experimento para determinar si la monitorización de la red es correcta. Para ello se ha utilizado la cámara IP anteriormente comentada y se ha accedido a ella a través de la aplicación oficial del proveedor. Esta operación se ha repetido 10 veces, cantidad que se entiende suficiente para demostrar un patrón. Con este muestreo se trata de determinar si el enfoque empleado es correcto o no. En definitiva se trata de diferenciar entre

las peticiones de red para iniciar la comunicación y las peticiones de envío de datos. Esta puntualización es importante para poder mantener el control sobre posibles ataque de de-negación de servicio.

Al diferenciar entre estos dos grupos de mensajes se consigue a su vez disminuir la cantidad de tráfico procesado, ya que para mantener la seguridad del sistema es suficiente con evaluar el tráfico de inicio de conexión.

Como ya se ha comentado en el punto dedicado a la propuesta, la herramienta utilizada en el proceso de monitorización es tcpdump. Para detectar el tráfico de red que fluye por el servidor Wi-Fi y teniendo en cuenta que la interfaz de red configurada es “wlan0”, se utilizará el siguiente comando:

```
tcpdump –i wlan0
```

```
15:01:51.917777 IP 8.211.48.22.10001 > 192.168.2.3.37854: UDP, length 44
15:01:51.920308 IP 192.168.2.3.37854 > 8.211.48.22.10001: UDP, length 48
15:01:51.925760 IP 47.91.78.150.10001 > 192.168.2.3.37854: UDP, length 44
15:01:51.925807 IP 47.254.135.53.10001 > 192.168.2.3.37854: UDP, length 44
15:01:51.932141 IP 192.168.2.3.37854 > 47.91.78.150.10001: UDP, length 48
15:01:51.932602 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 48
15:01:51.984868 IP 47.254.135.53.10001 > 192.168.2.3.37854: UDP, length 44
15:01:51.986940 IP 47.91.78.150.10001 > 192.168.2.3.37854: UDP, length 44
15:01:51.987179 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 48
15:01:51.988769 IP 192.168.2.3.37854 > 47.91.78.150.10001: UDP, length 48
15:01:52.052827 IP 47.254.135.53.10001 > 192.168.2.3.37854: UDP, length 104
15:01:52.344188 IP 47.254.135.53.10001 > 192.168.2.3.37854: UDP, length 590
15:01:52.348607 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 84
15:01:52.452995 IP 47.254.135.53.10001 > 192.168.2.3.37854: UDP, length 90
15:01:52.456588 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 52
15:01:52.466978 IP ec2-3-127-24-63.eu-central-1.compute.amazonaws.com.8053 >
192.168.2.3.52446: UDP, length 96
15:01:52.480665 IP 192.168.2.3.52446 > ec2-3-127-24-63.eu-central-
1.compute.amazonaws.com.8053: UDP, length 80
15:01:52.499186 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 404
15:01:52.518520 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 1076
15:01:52.519218 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 1076
15:01:52.519299 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 1076
15:01:52.520489 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 156
15:01:52.537787 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 404
15:01:52.576001 IP 47.254.135.53.10001 > 192.168.2.3.37854: UDP, length 69
15:01:52.577552 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 52
15:01:52.578853 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 1076
15:01:52.579067 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 114
15:01:52.580408 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 404
15:01:52.585132 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 404
15:01:52.626610 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 1076
15:01:52.626955 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 753
15:01:52.645708 IP 47.254.135.53.10001 > 192.168.2.3.37854: UDP, length 60
15:01:52.647410 IP 192.168.2.3.37854 > 47.254.135.53.10001: UDP, length 60
```

Figura 29. Muestreo de la monitorización de red con cámara IP (192.168.2.3)

Para explicar los datos obtenidos de la salida del comando tcpdump se tomará como ejemplo la siguiente línea del muestreo:

```
15:01:51.917777 IP 8.211.48.22.10001 > 192.168.2.3.37854: UDP, length 44
```

- **15:01:51.917777:** Tiempo en el que se ha detectado el tráfico de red.
- **8.211.48.22:** IP origen del tráfico.
- **10001:** Puesto origen del tráfico.
- **192.168.2.3:** IP destino.
- **37854:** Puesto de destino.
- **UDP:** Protocolo de la comunicación.
- **Length 44:** Longitud del paquete enviado.

Como resultado de estas pruebas se ha detectado que la comunicación con la cámara mantiene un patrón que será útil para la función comentada.
Las conclusiones obtenidas son las siguientes:

- Se ha detectado que todas las peticiones de comunicación se realizan sobre el protocolo UDP.

- Al inicio de las comunicaciones, la longitud de los mensajes es inferior a 50. Esta longitud aumenta cuando la conexión está establecida.

- Se han detectado múltiples paquetes de longitud 60. Este valor corresponde a la longitud del paquete vacío TCP ACK. Este valor puede ser establecido como punto de inflexión para el análisis del flujo de red.

Tras estas comprobaciones, se ha determinado que el comando que se utilizará en el proceso de monitorización de red será:

```
tcpdump –i wlan0 less 61
```

6 Conclusiones y Trabajos futuros

6.1 Conclusiones

La seguridad en las cámaras IP es ciertamente un camino en el que queda mucho recorrido. En el momento en el que nos encontramos, es preciso valorar todas las aportaciones al ámbito de la seguridad para este tipo de dispositivos.

En el trabajo realizado se ha comprobado que mantener un proceso de monitorización de red es de vital importancia para conseguir implementar un sistema de protección perimetral. En un inicio se pensó en utilizar un enfoque de proxy inverso por el que pasara toda comunicación y así poder permitir o no el acceso, a modo de semáforo. En la fase de prueba de concepto se desechó esta idea, ya que el manejo de las redirecciones en cuanto a la comunicación entre el cliente y las cámaras se refiere suponían un grave problema. Se planteó entonces cambiar y usar un enfoque de monitorización en paralelo. De esta forma no se altera la comunicación original, se reducen los posibles problemas y no se añade tiempo de respuesta a las peticiones. En las pruebas de monitorización de red realizadas se ha concluido que existe gran cantidad de envío de paquetes entre cliente y cámara IP. Este hecho dificulta el análisis de la comunicación. Para agilizar la validación, se ha optado por filtrar la monitorización por tamaño de paquetes. Según los datos obtenidos, en general, en el inicio de las comunicaciones se envían paquetes de tamaño inferior al resto de la comunicación. Este dato unido a que la validación planteada no requiere de analizar el contenido del flujo de red sino la comunicación en si, muestra que la monitorización de paquetes con tamaño inferior a 61 es suficiente para la solución descrita.

El tiempo necesario para bloquear una conexión ilícita ha resultado importante, ya que

en este espacio un atacante puede llegar a conseguir su objetivo. En la solución implementada, la validación “Double check” es el punto más sensible en este aspecto ya que se depende de sistemas de notificaciones externos. En las pruebas de rendimiento se ha comprobado que el tiempo medio de la respuesta de esta validación en un escenario habitual de un único dispositivo móvil vinculado es ligeramente superior a los 2 segundos. Parece un tiempo aceptable para bloquear una petición ilícita.

En la búsqueda de datos sobre ataques en los que cámaras IP se han visto involucradas, se ha detectado que el acceso ilícito a estos dispositivos es muy recurrente, ya sea para acceder a los datos personales como para modificar su comportamiento. En el dispositivo de validación detallado se ha decido utilizar 4 comprobaciones para evitar este tipo de ataques: Identificación de accesos desde IPs maliciosas, listas de confianza y comprobación “Double check” para certificar el uso lícito de la cámara. Para reforzar la seguridad y evitar ataques de tipo DDoS se ha añadido la comprobación de número de peticiones por minuto. Según los resultados obtenidos de las pruebas funcionales realizadas, se ha conseguido el objetivo de proteger frente a este tipo de acciones.

Con estos resultados se puede concluir que se han cumplido los objetivos planteados en el trabajo.

6.2 Trabajos futuros

El dispositivo Wi-Fi de validación cumple con los objetivos de seguridad que se han planteado en el inicio del trabajo. Aun así, la aportación de seguridad por parte de los fabricantes es de vital importancia para cubrir todas las necesidades de seguridad que surjan para estos dispositivos. Una de las razones por la que se ha detectado la necesidad de im-

plementar esta solución es la falta de capas de seguridad desde el proveedor del servicio-cámara IP. Este hecho queda patente al detectar la gran cantidad de vulnerabilidades y ataques que se han producido en sistemas de video-vigilancia de este tipo. Parece de gran importancia aportar soluciones de seguridad desde el sitio del fabricante.

Por otra parte, los objetivos de este trabajo de proteger instalaciones de cámaras IP han sido cumplidos. Queda pendiente comprobar la efectividad de la propuesta al extenderla a los dispositivos IoT en general. Al tratarse de una solución de protección perimetral parece viable la aplicación de esta metodología para conseguir la protección en el universo IoT en el hogar. En este trabajo, no queda demostrado su funcionamiento para los dispositivos IoT en general al considerarse fuera del ámbito del mismo.

Bibliografía

1- D. Airehrour, J. Gutierrez, S. K. Ray, "*A testbed implementation of a trust-aware RPL routing protocol*", Telecommunication Networks and Applications Conference (ITNAC) 2017 27th International, pp. 1-6, 2017

2- M. Endler, A. Silva, R. A. M. S. Cruz, "*An approach for secure edge computing in the Internet of Things*", Cyber Security in Networking Conference (CSNet) 2017 1st, pp. 1-8, 2017

3- D. Maevsky, A. Bojko, E. Maevskaya, O. Vinakov, L. Shapa, "*Internet of Things: Hierarhy of smart systems*", Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS) 2017 9th IEEE International Conference on, vol. 2, pp. 821-827, 2017

4- I. Bhardwaj, A. Kumar, M. Bansal, "*A review on lightweight cryptography algorithms for data security and authentication in IoTs*", Signal Processing Computing and Control (ISPCC) 2017 4th International Conference on, pp. 504-509, 2017

5- "*Gartner Says 8.4 Billion Connected "Things" Will Be in Use in 2017, Up 31 Percent From 2016*". En línea. Disponible en: https://www.gartner.com/en/newsroom/press-releases/2017-02-07-gartner-says-8-billion-connected-things-will-be-in-use-in-2017-up-31-percent-from-2016. Accedido: 2020

6- A. Costin, *"Security of CCTV and video surveillance systems; Threats, vulnerabilities, attacks, and mitigations*", The 6th International Workshop on Trustworthy Embedded Devices (pp. 45-54). Vienna, Austria: ACM, 2016

7- N. Jenkins. 245 million video surveillance cameras installed globally in 2014. June 2015

8- *"CCTV Calculator. Vulnerability database*". En línea. Disponible en: https://www.cctvcalculator.net/en/knowledges/vulnerability-database/. Accedido: 2020

9- B. Cusack and Z. Tian, *"Evaluating IP surveillance camera vulnerabilities*", 2017

10- *"Changing the face of surveillance: The brains behind the first network camera ".* En línea. Disponible en: https://www.axis.com/newsroom/article/first-network-camera. Accedido: 2020

11- "*Hypertext Transfer Protocol -- HTTP/1.1*". En línea. Disponible en: https://www.ietf.org/rfc/rfc2616.txt. Accedido: 2020

12- F. Baker, "*Core Protocols in the Internet Protocol Suite*", 2009

13- W. Du, *Computer Security: A Hands-on Approach*, 1ª edición. CreateSpace, 2017

14- Carnegie Mellon University, "*1996 CERT Advisories*", pp. 119-127, 2017

15- "*RTP: A Transport Protocol for Real-Time Applications*", En línea. Disponible en: https://www.rfc-editor.org/rfc/rfc3550.txt. Accedido: 2020

16- J. Gómez, M. A. de Castro, P. Guillén, *Hackers. Aprende a atacar y defenderte*. 2ª edición. Madrid: Ra-Ma, 2014, pp. 96-102

17- *"What is cibersecurity?"*. En línea. Disponible en https://www.kaspersky.com/resource-center/definitions/what-is-cyber-security. Accedido: 2020

18- "*OWASP Top Ten Project*". En línea. Disponible en: https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project. Accedido: 2020

19- "*Leading the IoT*". En línea. Disponible en: https://www.gartner.com/imagesrv/books/iot/iotEbook_digital.pdf. Accedido: 2020

20- B. Schneier, "*IoT Security: What's Plan B?*", in IEEE Security & Privacy, vol. 15, no. 05, pp. 96-96, 2017

21- P. Rodriguez, "*Ciberseguridad: Protegiendo la información vulnerable*" Fundación Telefónica, Madrid, 2019

22- "*Hdiv security*". En línea. Disponible en: https://hdivsecurity.com/. Accedido: 2019

23- K. Ingham and S. Forrest, "*A History and Survey of Network Firewalls*", The University of New Mexico Computer Science Department Technical Report 2002-37, 2002, pp. 2-9

24- J. García, *Videovigilancia: CCTV usando videos IP*. Málaga: Vértice, 2011, pp. 132-136

25- G. Hittu and D. Mayank, "*Securing IoT Devices and SecurelyConnecting the Dots Using REST API and Middleware*", 2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU), Ghaziabad, India, 2019, pp. 1-6

26- "*Contrast security*". En línea. Disponible en: https://www.contrastsecurity.com/. Accedido: 2019

27- E. Ronen and A. Shamir, "*Extended Functionality Attacks on IoT Devices: The Case of Smart Lights*", 2016 IEEE European Symposium on Security and Privacy (EuroS&P), Saarbrucken, 2016, pp. 3-12

28- S. Rizvi, A. Kurtz, J. Pfeffer and M. Rizvi, "*Securing the Internet of Things (IoT): A Security Taxonomy for IoT*", 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), New York, NY, 2018, pp. 163-168

29- M. Nawir, A. Amir, N. Yaakob and O. B. Lynn, "*Internet of Things (IoT): Taxonomy of security attacks*", 2016 3rd International Conference on Electronic Design (ICED), Phuket, 2016, pp. 321-326

30- "*Firebase Cloud Messaging*". En línea. Disponible en: https://firebase.google.com/docs/cloud-messaging/. Accedido: 2020

31- "*Google Cloud Messaging*". En línea. Disponible en: https://developers.google.com/cloud-messaging. Accedido: 2019

32- Y. S. Yilmaz, B. I. Aydin and M. Demirbas, "*Google cloud messaging (GCM): An evaluation*", 2014 IEEE Global Communications Conference, Austin, TX, 2014, pp. 2807-2812

33- "*TCPDUMP & LIBPCAP*". En línea. Disponible en: http://www.tcpdump.org/. Accedido: 2020.

34- "*PolePosition Results. Results from running the Poleposition open source database benchmark*". En línea. Disponible en: http://hsqldb.org/PolePosition.pdf. Accedido:2020

35- "*PolePosition. The open source database benchmark*". En línea. Disponible en: http://www.polepos.org/. Accedido: 2020

36- "*Project Honey Pot*". En línea. Disponible en: https://www.projecthoneypot.org. Accedido: 2020

37- "Tor". En línea. Disponible en: https://www.torproject.org/. Accedido: 2020

38- "*Setting up a Raspberry Pi as a routed wireless access point"*. En línea. Disponible en: https://www.raspberrypi.org/documentation/configuration/wireless/access-point-routed.md. Accedido: 2020

39- "*Spring Boot 2.1.0*". En línea. Disponible en: Spring Boot 2.1.0. Accedido: 2020