Universidad de Buenos Aires
Facultades de Ciencias Económicas, Cs. Exactas y Naturales e Ingeniería.

TEMA:

Implementación segura de redes IoT: aspecto fundamental en la prevención de generación de Botnets

---

Carrera de Especialización en Seguridad Informática

Trabajo Final de Especialización

Autor: **Ing. Sebastián A. Fontana**

Tutor de Trabajo Final: **Mg. Ing. Juan A. Devincenzi**

Año de presentación: 2025 - Cohorte del cursante: 2024

Declaración Jurada de origen de los contenidos

Por medio de la presente, el autor manifiesta conocer y aceptar el Reglamento de Trabajos Finales vigente y se hace responsable que la totalidad de los contenidos del presente documento son originales y de su creación exclusiva, o bien pertenecen a terceros u otras fuentes, que han sido adecuadamente referenciados y cuya inclusión no infringe la legislación Nacional e Internacional de Propiedad Intelectual.

Nombre: Sebastián Alejandro Fontana

DNI: 34.566.243

FIRMADO

Resumen

El actual trabajo presenta como objetivo el análisis de la tecnología *Internet of Things* (en adelante IoT), comprendiendo una breve introducción, la descripción del origen, la evolución hasta la fecha y las implementaciones futuras, el detalle de su funcionamiento en base al modelo de referencia de capas o niveles provisto por Cisco, la identificación de las tecnologías y protocolos que forman parte de su estructura, y la definición de las principales ventajas y desventajas de su utilización.

Otro aspecto relevante que se tendrá en consideración, es acerca de la seguridad de estas redes haciendo énfasis en cuáles son las principales vulnerabilidades que se suelen presentar. Con relación a esto, se hará hincapié en que la explotación de tales debilidades puede dar lugar a la generación de las denominadas *Botnet*[1], de las cuáles se brindará una introducción acerca de su funcionamiento, se describirán los diferentes tipos de arquitectura, los principales tipos de ataques y la evolución de las mismas a lo largo del tiempo.

Finalmente, se desarrollará un marco de medidas de protección y buenas prácticas que se podrán implementar con el fin de fortalecer la seguridad de todos aquellos dispositivos que se conectan a internet con el fin de disminuir significativamente la probabilidad de que sean un blanco de ataque vulnerable frente a intentos de acceso no autorizados y otras amenazas que puedan comprometer la integridad de la infraestructura.

Palabras claves: IoT, Botnet.

[1] Red de dispositivos conectados a internet que están infectados con un mismo software malicioso para llevar a cabo una determinada tarea.

Contenido general

Lista de abreviaturas y acrónimos

6LoWPAN: *IPv6 over Low-Power Wireless Personal Area Networks* / IPv6 sobre Redes de Área Personal Inalámbricas de Bajo Poder

ACK: *Acknowledgment* / Acuse de Recibo

AIoT: *Artificial Intelligence of Things*

AMQP: *Advanced Message Queuing Protocol* / Protocolo de Cola de Mensajes Avanzado

API: *Application Programming Interface*

BLE: *Bluetooth Low Energy* / Bluetooth de Baja Energía

C&C: *Command and Control*

CCTV: Circuito Cerrado de Televisión

CoAP: *Constrained Application Protocol* / Protocolo de aplicación restringida

CRM: *Customer Relationship Management* / Gestor de Relación con los Clientes

CRUD: *Create, Read, Update, Delete*

DDoS: *Distributed Denial of Service*

DNS: *Domain Name System*

DoS: *Denial of Service*

DSS: *Data Distribution Service*

EAPOL: *Extensible Authentication Protocol over LAN* / Protocolo de Autenticación Extensible sobre LAN

ERM: *Enterprise Risk Management* / Gestor de Riesgos Empresariales

ERP: *Enterprise Resource Planning* / Planificador de Recursos Empresariales

GPS: *Global Positioning System*

HART: *Highway Addressable Remote Transducer* / Transductor remoto Direccionable de Alta Velocidad

HTTP: *Hypertext Transfer Protocol*

IA: Inteligencia Artificial

ICMP: *Internet Control Message Protocol*

IEEE: *Institute of Electrical and Electronics Engineers* / Instituto de Ingenieros Eléctricos y Electrónicos

IIoT: *Industrial Internet of Things* / Internet de las Cosas Industrial

IoMT: *Internet of Medical Things*

IoT: *Internet of Things* / Internet de las Cosas

IP: *Internet Protocol*

IPv4/IPv6: *Internet Protocol Version 4 / Internet Protocol Version 6*

LAN: *Local Area Network* / Red de Área Local

Li-Fi: *Light-Fidelity* / Fidelidad de la Luz

LoRaWAN: *Long Range Wide Area Network* / Red de Área Amplia de Larga Distancia

LPWAN: *Low Power Wide Area Network* / Red de Área Amplia de Bajo Consumo

M2M: *Machine to Machine*

MQTT: *Message Queuing Telemetry Transport* / Transporte de Telemetría de Cola de Mensajes

NFC: *Near-Field Communication* / Comunicación de Campo Cercano

OCF: *Open Connectivity Foundation*

OSI: *Open Systems Interconnection*

OWASP: *Open Web Application Security Project*

P2P: *Peer to Peer*

PPC: *Pay-Per-Click* / Pago Por Click

RFID: *Radio Frequency Identification*

RTSP: *Real Time Streaming Protocol* / Protocolo de Transmisión en Tiempo Real.

SSID: *Service Set Identifier /* Identificador de Conjunto de Servicios

SPOT: *Smart Personal Object Technology*

TCP/IP: *Transmission Control Protocol / Internet Protocol*

TCP: *Transport Control Protocol* / Protocolo de Control de Transmisión

TFT-LED: *Thin Film Transistor-Liquid Crystal Display* / pantalla de cristal líquido de transistores de película fina

UDP: *User Datagram Protocol* / Protocolo de Datagramas de Usuario

Wi-Fi: *Wireless Fidelity*

WPAN: *Wireless Personal Area Network* / Red de Área Personal Inalámbrica

XMPP: *Extensible Messaging and Presence Protocol* / Protocolo Extensible de Mensajería y Presencia

1. Introducción a Internet de las Cosas

El concepto IoT es actualmente muy utilizado ya que es una tecnología que se encuentra en profundo auge. En tal sentido, existen diversas definiciones dadas por grandes corporaciones las cuáles se encuentran completamente alineadas entre sí. Se citan algunas de ellas a modo de referencia:

IBM: “El internet de las cosas (IoT) se refiere a una red de dispositivos físicos, vehículos, electrodomésticos y otros objetos físicos que están integrados con sensores, *software* y conectividad de red que les permite recopilar y compartir datos.” [1].

Cisco: “El Internet de las Cosas (IoT) conecta objetos ordinarios con otros objetos y aplicaciones en la nube, haciéndolos inteligentes e interactivos. Tales dispositivos “inteligentes” hacen nuestra vida más rica, saludable y ayuda a optimizar el uso de escasos recursos.” [2]

Oracle: “El Internet de las Cosas (IoT) describe la red de objetos físicos ("cosas") que llevan incorporados sensores, *software* y otras tecnologías con el fin de conectarse e intercambiar datos con otros dispositivos y sistemas a través de Internet. Estos dispositivos van desde objetos domésticos comunes hasta herramientas industriales sofisticadas…” [3]

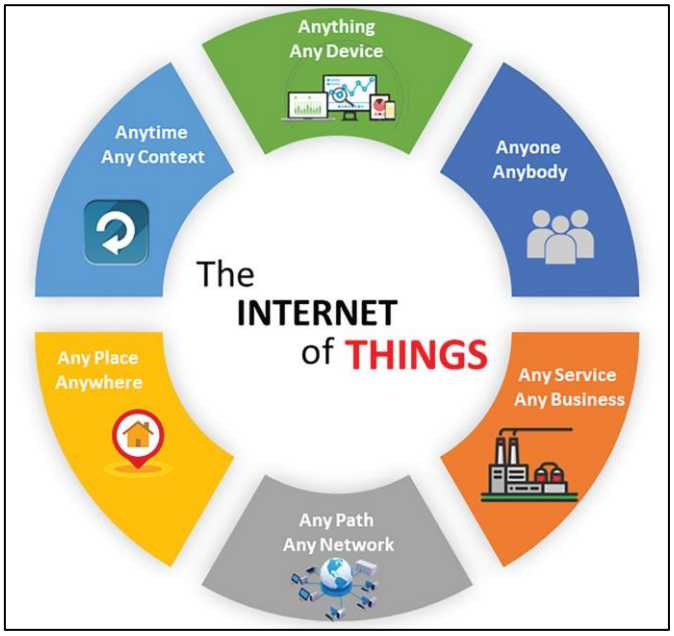

Figura 1. Interoperabilidad de IoT. [4]

Tal como se puede observar en la figura 1, IoT es una tecnología muy adaptable a diferentes entornos que se la puede utilizar:

- En cualquier momento: los dispositivos IoT suelen estar en funcionamiento constante debido a que suelen utilizarse para automatizar procesos.
- En cualquier lugar: es adaptable tanto para hogares como industrias.
- En cualquier red: la configuración de estos equipos permite su inclusión en las redes donde sean requeridos.
- En cualquier organización: su utilización se puede dar en diferentes tipos de empresas, por ejemplo, para empresas petroleras, para cultivos “inteligentes”, optimización de procesos de empresas productoras de diversa índole, etc.
- Con fácil implementación: la tecnología es maleable y puede ser fácilmente instalada en los hogares, como cámaras de seguridad, televisores inteligentes, etc., como así también, en industrias por personal autorizado para tal fin ya que requieren de mayores medidas de seguridad.
- En cualquier dispositivo: existe una gran diversidad de dispositivos IoT, los cuáles ofrecen diferentes servicios. Los mismos serán descriptos en el capítulo 6. Principales escenarios actuales y futuros.

2. Historia del IoT

El concepto de IoT tuvo sus orígenes a fines de la década del 1990 y principios del 2000, cuando investigadores y expertos en tecnología comenzaron a explorar la idea de conectar dispositivos de uso cotidiano a Internet.

Uno de los primeros casos de IoT fue el de una máquina expendedora de Coca-Cola de la Universidad de Carnegie Mellon, la cual se conectó a Internet en el año 1982 y permitió a los usuarios chequear remotamente la disponibilidad del stock de gaseosas antes de acercarse a la misma.

En los próximos años, el desarrollo de las tecnologías inalámbricas y crecimiento de internet permitieron grandes avances en la tecnología IoT. Hasta que en el año 1990 surge la creación de una tostadora con conexión a internet, que permitía su operabilidad a través de la red. [5]

Luego en el año 1993, ingenieros de la Universidad de Cambridge colocaron una cámara la cual capturaba tres fotos por minuto de una cafetera para monitorizar su estado. Esta fue la precursora de las cámaras CCTV[2]. [6]

En el año 1995, se desplegó la primera versión del programa de satélites GPS[3] de larga duración, gestionado por el gobierno de Estados Unidos. Esto fue un gran paso hacia el desarrollo de uno de los componentes más vitales para muchos dispositivos IoT: la ubicación.

El término “Internet de las Cosas” fue pronunciado por primera vez en el año 1999 por Kevin Ashton en presentación para los directivos de su empresa y tiene que ver básicamente con la conexión de todo tipo de objetos que utilizamos en nuestra vida diaria a internet.

Un año después, la empresa LG introdujo al mercado la primera heladera inteligente. La misma contaba con una pantalla táctil TFT-LCD[4] con televisión y puerto LAN[5]. Ofrecía funciones como lápiz electrónico, mensajería de video,

[2] Circuito Cerrado de Televisión
[3] Sistema de Posicionamiento Global
[4] *Thin Film Transistor-Liquid Crystal* Display / Pantalla de Cristal Líquido de Transistores de Película Fina.
[5] Red de Área Local

gestión de horarios y de la temperatura interna, datos nutricionales de alimentos y recetas. Permitía tomar fotos, incluía un reproductor de MP3 y podía programar pedidos automáticos si las tiendas cercanas estaban en línea. [7]

Figura 2. Primera heladera IoT. [8]

Posteriormente, en el año 2004 surge el primer reloj inteligente, lanzado por Microsoft. A tal dispositivo se lo llamó SPOT[6], el cual su finalidad era poder gestionar distintos tipos de pequeños electrodomésticos del hogar.

En el año 2007, sale al mercado el primer iPhone. Las funcionalidades incluían, entre otras, el envío de mensajes de texto, llamadas, navegación por internet, acceso a mapas, posibilidad de tomar fotos y videos, aplicación para escuchar música y acceso al correo electrónico. [9]

Figura 3. iPhone 1 o también llamado 2G. [9]

[6] *Smart Personal Object Technology.*

En el año 2008, la cantidad de dispositivos IoT conectados a Internet superaba en número a la población mundial.

Luego en el año 2009, Google lanza los primeros autos conducidos autónomamente, llamados Waymo. El sistema de percepción del conductor utiliza datos recogidos de su avanzada suite de sensores y descifra lo que lo rodea utilizando inteligencia artificial: peatones, ciclistas, otros vehículos, construcciones, etc. El conductor de Waymo también responde a señales e indicaciones, como los colores de los semáforos y las señales de alto. [10]

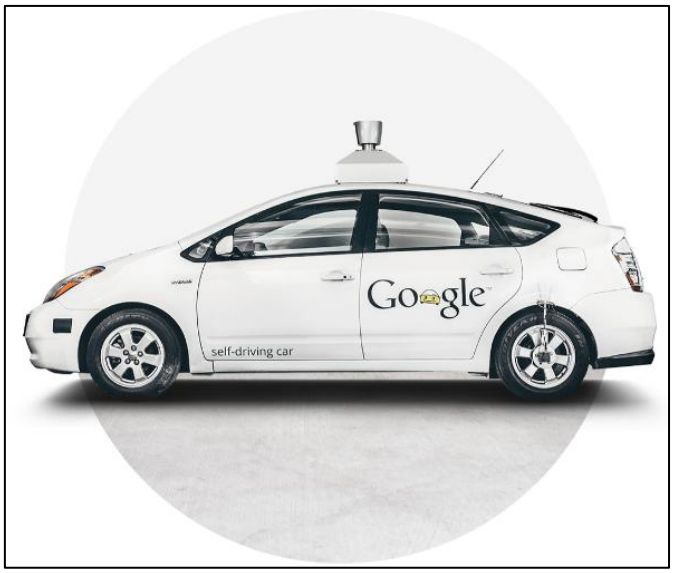

Figura 4. Waymo: primer móvil autónomo de Google en 2009.

En el año 2011, se comienzan a comercializar los primeros televisores inteligentes, y dos años después, los lentes inteligentes Google Lens, seguidos por Echo, un año más tarde.

En el año 2015, Tesla crea un auto con piloto automático, y surge el servicio de Amazon llamado AWS IoT Core. Este es un servicio en la nube que ofrece un conjunto de características para la gestión de dispositivos IoT, el procesamiento de datos y la comunicación entre dispositivos y aplicaciones en la nube.

Actualmente, los usos de esta tecnología son muy amplios y serán descriptos en la sección “6. Principales escenarios actuales y futuros”.

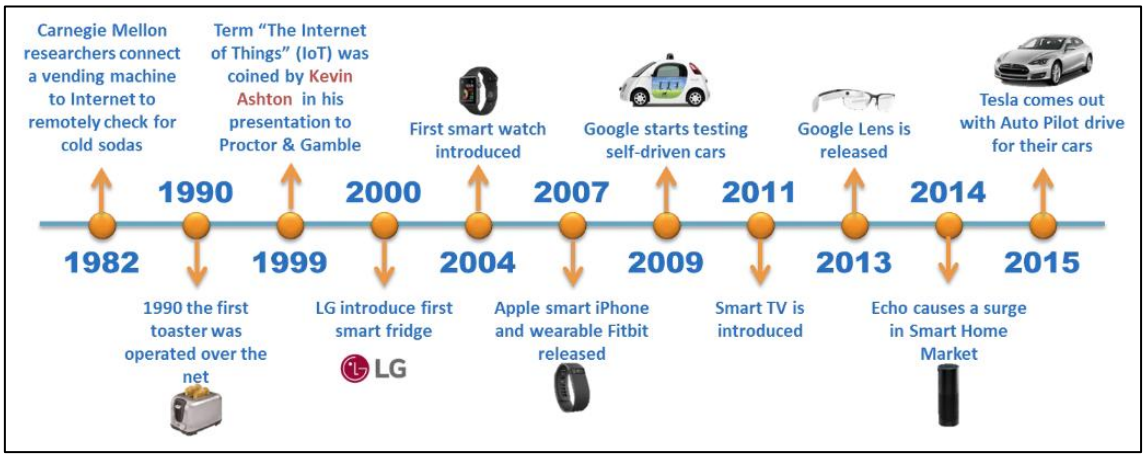

Figura 5. Historia de IoT. [11]

3. Características de las redes IoT.

Tal como se mencionó anteriormente, IoT es una red de objetos físicos que están siendo monitoreados y controlados a través de internet. Estos objetos deben ser capaces de comunicarse entre ellos y compartir información, brindando a los usuarios diferentes tipos de servicios.

Estas redes presentan una serie de características, que las convierten en una tecnología resiliente y que presentan una gran cantidad de beneficios:

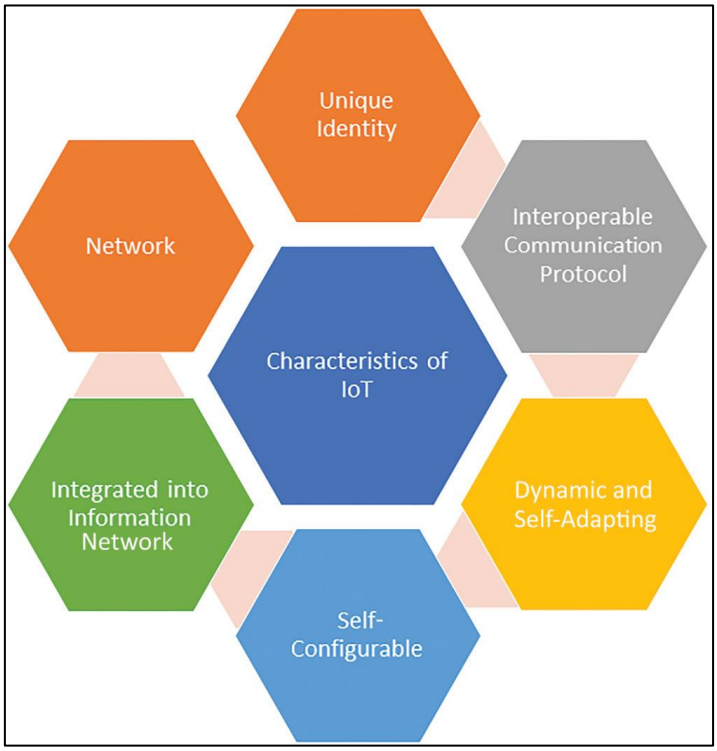

Figura 6. Características de IoT. [4]

- Identidad única: para coordinar de manera efectiva los objetos en las aplicaciones IoT, cada uno recibe una identidad única, lo que permite monitorearlos, actualizarlos y controlarlos de forma remota a través de sus direcciones IP.
- Interoperabilidad de los protocolos de comunicación: es la capacidad que presentan los componentes dentro de la red para intercambiar, compartir información y llevar a cabo en conjunto, una determinada función, considerando que tales componentes han sido producidos por diferentes fabricantes.

- Autoadaptable y dinámico: las aplicaciones y dispositivos IoT pueden percibir cambios en su entorno y adaptarse a ellos según el contexto del usuario. Por ejemplo, en un sistema de vigilancia, las cámaras se ajustan según el entorno entrando en modo de bajo consumo cuando no detectan objetos, cambiando a infrarrojo por la noche y ajustando la claridad según la luz del ambiente.
- Autoconfigurable: hace referencia al conjunto de cambios que un sistema realiza sobre sí mismo en respuesta a eventos en su entorno e internamente. Muchos de estos dispositivos son capaces de obtener actualizaciones de software y configurar redes básicas o realizar verificaciones de estado por sí mismos, o con mínima intervención del usuario.
- Integrado en redes información: los dispositivos IoT están configurados de tal manera que pueden comunicarse entre sí dentro del entorno basado en IoT para crear una red de información. [4]
- Redes: hace referencia a la interconexión de las redes IoT con internet en general. En su funcionamiento, como se verá en la próxima sección, la información suele partir desde la red IoT hacia la nube donde se procesa la información recolectada por los sensores, para que luego la respuesta sea enviada a los actuadores y/o a los sistemas aplicativos administrados por los usuarios finales para la toma de decisiones.

4. Funcionamiento, componentes y arquitectura de una red IoT

4.1. ¿Cómo funciona una red IoT?

El modo de funcionamiento de una red IoT se puede describir de la siguiente manera:

1. Los dispositivos IoT colectan información a través de sus sensores y la transmiten hacia la puerta de enlace (o *gateway*).
2. La puerta de enlace recibe la información de múltiples dispositivos y la envía a la nube para su procesamiento y análisis.
3. En la nube se procesa la información, aplicando algoritmos analíticos, generando información y alertas.
4. Los aplicativos de IoT acceden a la información procesada en la nube, y la presentan a los usuarios de una manera que le aporte valor y permita la toma de decisiones.
5. El sistema puede también enviar comandos o instrucciones a los dispositivos IoT (actuadores) basados en la información ya procesada.
6. El sistema se encuentra continuamente monitorizando y gestionando los dispositivos, conectividad y seguridad para garantizar una operación segura y confiable.

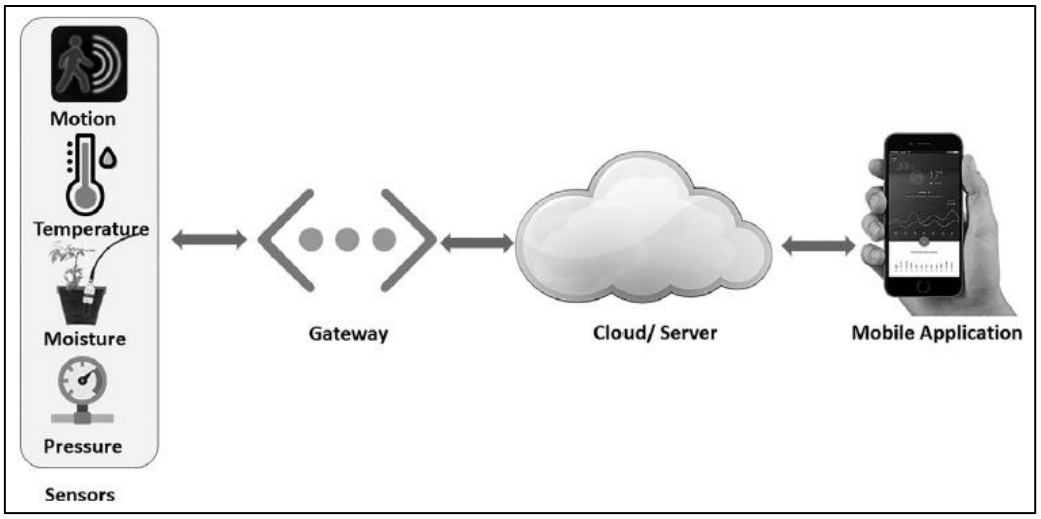

Figura 7. Funcionamiento de IoT. [4]

4.2. Componentes

Tal como se comentó anteriormente, IoT es un sistema que para su funcionamiento se compone de distintos elementos: dispositivos IoT (sensores y actuadores), conectividad, gestor de datos, aplicativo, seguridad e interoperabilidad. Cada componente desempeña un papel crucial en permitir que los dispositivos inteligentes funcionen de manera autónoma e interactiva dentro de la red. Comprender estos componentes proporciona una visión más clara sobre cómo funcionan los sistemas IoT y sus posibles aplicaciones. [12]

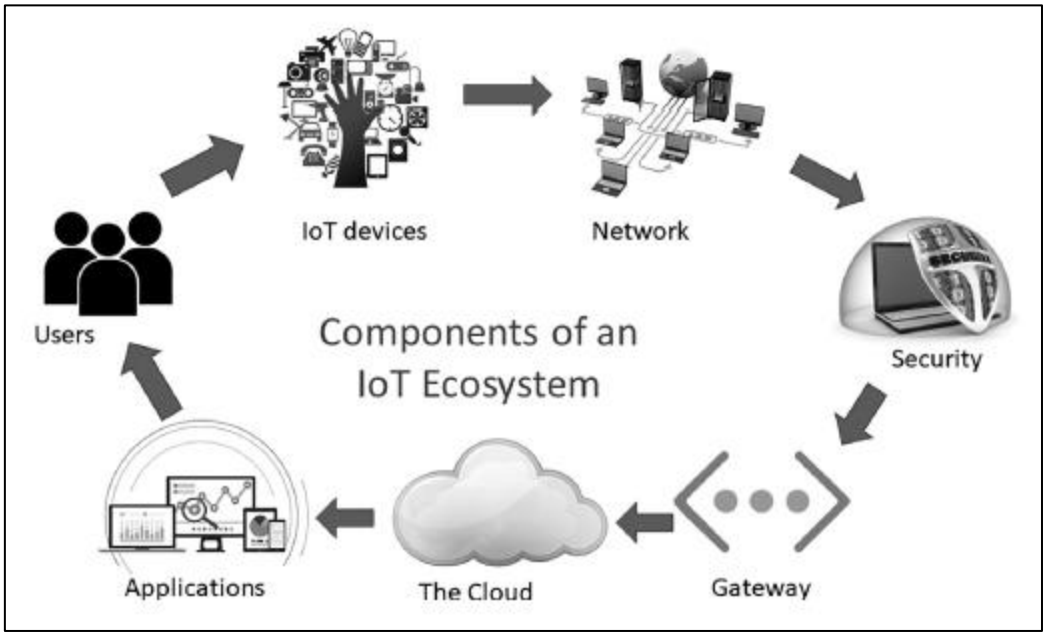

Figura 8. Componentes de un ecosistema de IoT. [4]

4.2.1. Dispositivos IoT

4.2.1.1. Sensores

Los sensores pueden detectar y medir cualquier parámetro físico, entre otros, temperatura, humedad, presión y movimiento. Los sensores convierten estas entradas físicas en datos digitales que pueden ser procesados por computadoras. La versatilidad y capacidad de los sensores han crecido significativamente, y ahora abarcan desde termistores simples para medir la temperatura hasta acelerómetros complejos utilizados en teléfonos inteligentes para detectar orientación y movimiento.

Existen una gran diversidad de tipos de sensores de acuerdo con el elemento que miden. A continuación se describen los principales:

- Sensores de presión: son dispositivos que miden la presión (la fuerza necesaria para impedir la expansión de un fluido) en gases o líquidos. Cuando la presión cambia, el sensor detecta estos cambios y los comunica a los sistemas conectados.
- Sensores de luz: son dispositivos electrónicos que responden al cambio en la intensidad de la luz. Los sensores inteligentes de luz están diseñados para uso industrial y de consumo. Estos dispositivos fotoeléctricos convierten la energía lumínica en señales eléctricas que pueden ser interpretadas y utilizadas por otros dispositivos.
- Sensores de temperatura y humedad: los sensores de temperatura miden la cantidad de energía calórica en una fuente, lo que les permite detectar los cambios de temperatura y convertir estos cambios en datos. Por otro lado, los sensores de humedad detectan el porcentaje de agua presente en el aire, en un material o en cualquier superficie.
- Sensores giroscópicos y de aceleración: son los sensores que permiten a los *smartphones*[7] detectar si los mismos están en posición vertical, en modo horizontal, o si se están moviendo, y se utilizan ampliamente en el diseño de juegos para los mismos. Si bien suelen funcionar juntos, existen diferencias entre ambos. Un giroscopio es un sensor que mide la velocidad de rotación o giro de un objeto y detecta la orientación del dispositivo, mientras que un acelerómetro es un sensor que mide la vibración o aceleración del movimiento de un dispositivo.
- Sensores de proximidad y movimiento: los sensores de proximidad son capaces de detectar la presencia de objetos cercanos sin ningún contacto físico emitiendo un campo electromagnético o un haz de radiación electromagnética (por ejemplo, infrarrojos) y buscando cambios en el campo. Por otro lado, los sensores de movimiento detectan el movimiento

[7] Celulares inteligentes.

en un área específica y pueden generar una señal de salida cuando se detecta movimiento.

- Sensores de flujo y gas: los sensores de flujo son dispositivos que se utilizan para medir el caudal o la cantidad de un líquido o gas en movimiento. Por otro lado, los sensores de gas son dispositivos electrónicos que monitorizan y detectan diferentes tipos de gases, así como los cambios en la calidad de aire debidos a la presencia de gases tóxicos, combustibles o peligrosos.
- Sensores de sonido: los sensores de sonido se definen como un módulo que detecta las ondas sonoras por su intensidad, convirtiéndolas en señales eléctricas.
- Sensores de imagen: un sensor de imagen se compone de un chip formado por componentes sensibles a la luz que al ser expuestos forman una imagen digital. Su capacidad para convertir la luz en señales eléctricas permite a este tipo de sensores registrar y procesar información visual de manera rápida, eficiente y precisa.
- Sensores magnéticos: son dispositivos diseñados para detectar y medir campos magnéticos. En otras palabras, detectan cuándo se atraen o se separan los polos de su imán. Estos campos pueden ser generados por imanes permanentes, electroimanes, corrientes eléctricas o materiales ferromagnéticos.
- Sensores de calidad de aire: son dispositivos diseñados para medir y monitorizar varios parámetros relacionados con la calidad del aire, así como la presencia de contaminantes atmosféricos en un entorno determinado. Estos sensores son fundamentales para tomar medidas que aseguren la calidad del aire y, de esta manera, proteger la salud humana y el medioambiente.
- Sensores de calidad de agua: están diseñados para medir y monitorizar diversos parámetros del agua con el fin de evaluar su calidad y determinar si cumple con ciertos estándares o requisitos específicos. [13]

4.2.1.2. Actuadores

Mientras que los sensores se encargan de recopilar información, los actuadores se centran en tomar acción. Estos son los componentes de un sistema IoT que convierten señales eléctricas en movimiento físico u otras formas de salida. Pueden controlar mecanismos o sistemas, desde ajustar la posición de una válvula hasta regular la velocidad de un motor. [12]

4.2.2. Conectividad

La conectividad es fundamental en los sistemas IoT, ya que vincula sensores y actuadores con unidades de procesamiento e interfaces de usuario. Las redes en IoT pueden ser simples o complejas, locales o globales. Utilizan diversos protocolos de comunicación y tecnologías como Wi-Fi, Bluetooth, Zigbee, redes celulares e incluso LoRaWAN (Red de Área Amplia de Larga Distancia) para comunicaciones de bajo consumo a largas distancias.

La elección de una tecnología de red particular depende de consideraciones como el alcance, el ancho de banda, el consumo de energía y los requisitos generales del sistema. Las redes garantizan que los datos recopilados por los sensores se transmitan a las plataformas en la nube para su procesamiento y que los comandos se envíen de vuelta a los actuadores para realizar las acciones apropiadas. [12]

Se brindarán mayores detalles acerca de las tecnologías y protocolos utilizados en IoT en la sección “5. Protocolos empleados en IoT”.

4.2.3. Gestor de datos

Las plataformas en la nube desempeñan un papel central en la arquitectura IoT al proporcionar grandes capacidades de procesamiento de datos que pueden resultar poco prácticas de manejar a nivel de dispositivo o red local. Estas plataformas reciben datos de diversos dispositivos a través de diferentes redes, los almacenan y los procesan utilizando análisis avanzados y algoritmos de aprendizaje automático. También sirven como la columna vertebral para la gestión de aplicaciones y dispositivos, y a menudo incluyen robustas

características de seguridad para proteger datos sensibles y garantizar la privacidad. Las plataformas en la nube permiten la escalabilidad en los sistemas IoT, facilitando la integración de un número creciente de dispositivos sin necesidad de infraestructura adicional por parte del usuario. [12]

4.2.4. Aplicativo

La capa del aplicativo es la que recibe la información procesada, y la que se encarga de presentar tales datos a los usuarios, y a la vez, permite automatizar acciones sobre los actuadores.

4.2.5. Seguridad

Los dispositivos y sistemas IoT son vulnerables a ciberataques, por lo tanto la seguridad es un componente crucial en el ecosistema IoT. Medidas de seguridad como encriptación, uso de firewalls y controles de acceso permiten proteger información sensitiva y asegurar la privacidad y seguridad de los usuarios.

Se brindará un panorama completo sobre las medidas de seguridad a adoptar para reforzar la seguridad de estas redes en la sección “13. Medidas de seguridad para la implementación de redes IoT seguras”.

4.2.6. Interoperabilidad

Para que el ecosistema de IoT funcione efectivamente, los dispositivos de distintos proveedores deben ser capaces de comunicarse e intercambiar información entre ellos. Los estándares de interoperabilidad, tales como los desarrollados por la Open Connectivity Foundation (OCF) y el AllSeen Alliance, ayudan a asegurar la compatibilidad entre los dispositivos.

4.3. Arquitectura típica de IoT

En la actualidad existen diferentes modelos que detallan la arquitectura de una red IoT, los cuáles se fundamentan en diferentes cantidad de niveles o

capas. En el presente trabajo, se analizará el esquema provisto por Cisco, cuyo modelo de referencia consta de siete capas, las cuales serán descriptas a continuación.

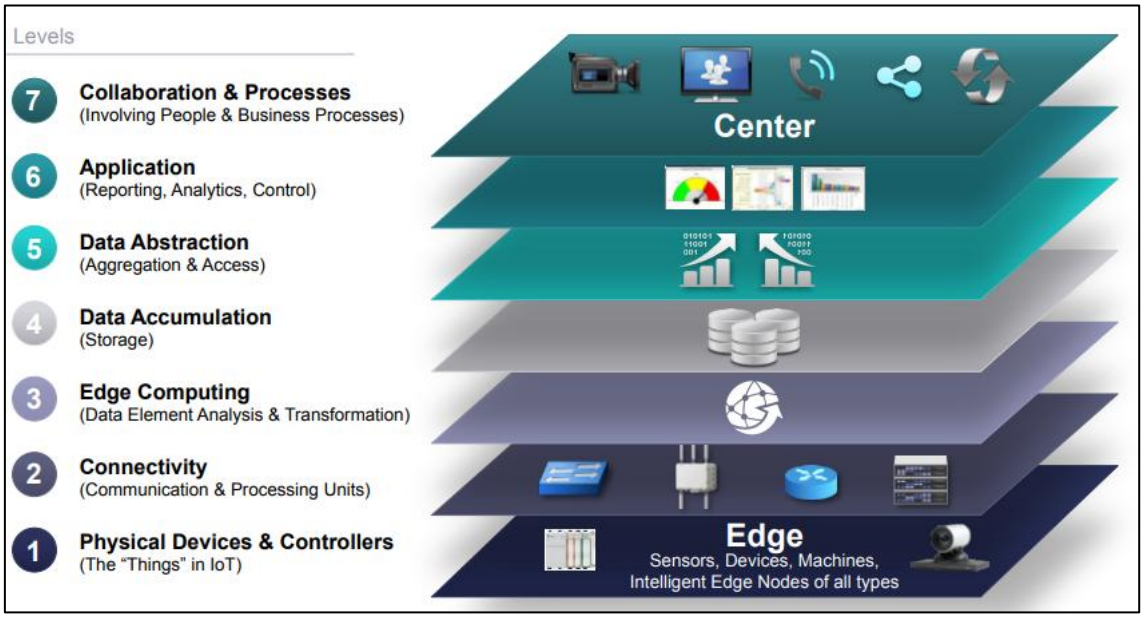

Figura 9. Modelo de referencia de IoT por Cisco. [14]

- Capa 1. Dispositivos físicos y controladores: la presente capa comprende a los dispositivos finales de los sistemas IoT, es decir, los sensores y actuadores. Adicionalmente, forman parte de esta capa todos aquellos dispositivos inteligentes que se encuentran conectados a internet.

- Capa 2. Conectividad: el segundo nivel se encarga de gestionar las comunicaciones entre dispositivos, redes y servicios en la nube que conforman la infraestructura IoT.

  Dada la trascendencia y extensión, este tópico se desarrollará en la sección "5. Protocolos empleados en IoT".

- Capa 3. *Edge Computing* o Computación Frontera: esta capa hace énfasis en la necesidad de procesar y almacenar la información lo más cerca de sus fuentes. Este enfoque permite analizar y transformar grandes volúmenes de datos en tiempo real de manera local, en el límite (o borde) de las redes. De esta manera, se ahorra el tiempo y los recursos que de otro modo serían necesarios para enviar todos los datos a los servicios en

la nube. El resultado es una reducción de la latencia del sistema, lo que lleva a respuestas en tiempo real y mejora del rendimiento.

La computación frontera se produce en servidores locales u otros nodos en el borde dispersos a lo largo de la red. A este nivel, los datos son: evaluados para determinar si requieren un procesamiento adicional en niveles superiores, formateados para un procesamiento posterior, decodificados, filtrados y dirigidos al correspondiente *host* destino. [15]

- Capa 4. Acumulación de Datos: en esta etapa se define si los datos son relevantes para los requisitos del negocio y dónde deben ser almacenados. Los datos se guardan en una amplia gama de soluciones de almacenamiento, desde lagos de datos capaces de contener datos no estructurados como imágenes y flujos de video, hasta almacenes de eventos y bases de datos. El objetivo final es ordenar una gran cantidad de datos diversos y almacenarlos de la manera más eficiente posible.
- Capa 5. Abstracción de Datos: se finaliza la preparación de los datos para que las aplicaciones consumidoras puedan utilizarlos para generar información valiosa. Todo el proceso involucra los siguientes pasos:
    - Combinar datos de diferentes fuentes, tanto IoT como no IoT, incluidos los sistemas ERM[8], ERP[9] y CRM[10];
    - Reconciliar múltiples formatos de datos; y
    - Agregar los datos en un solo lugar o hacerlos accesibles sin importar la ubicación a través de virtualización de datos.

  De manera similar, los datos recopilados en la capa de aplicación se reformatean aquí para enviarlos al nivel físico, de modo que los dispositivos puedan entenderlos.
- Capa 6. Aplicación: la capa de aplicación de una arquitectura de sistema de IoT implica decodificar patrones prometedores en datos de IoT y

[8] Gestor de Riesgos Empresariales.
[9] Planificador de Recursos Empresariales.
[10] Gestor de Relación con los Clientes.

compilarlos en resúmenes que sean fáciles de entender para los humanos, como gráficos y tablas. Los programas para el control y monitoreo de dispositivos, así como el software de control de procesos, son ejemplos típicos de la capa de aplicación de la arquitectura de IoT.

- Capa 7. Colaboración y Procesos: los patrones decodificados a nivel de aplicación se pueden utilizar para extraer más información empresarial, proyectar tendencias futuras e impulsar decisiones operativas que mejoren la eficiencia, la seguridad, la rentabilidad, la experiencia del cliente y otros aspectos importantes de la funcionalidad empresarial.

5. Protocolos empleados en IoT

Las redes IoT basan su funcionamiento en una serie de tecnologías y protocolos, las cuáles se describirán en función a la capa del modelo TCP/IP (acceso a la red, capa de red, transporte y aplicación) a la que corresponden, de acuerdo con la figura que se ilustra a continuación:

Capas TCP/IP	Protocolos / Tecnologías
Capa de aplicación	**AMQP – CoAP – MQTT – XMPP**
Capa de Transporte	**TCP – UDP**
Capa de Internet	**IPv4/IPv6 – 6LoWPAN**
Capa de acceso a la red	**802.15.4 – Bluetooth/BLE – Wi-Fi – Zigbee – Z-Wave – NFC – HART – Li-Fi – Redes móviles (3G/4G/5G) – LoRa y LoRaWAN**

Figura 10. Tecnologías y protocolos en IoT por capa TCP/IP.

5.1. Capa de acceso a la red

Las tecnologías de Redes Personales Inalámbricas (WPAN[11], por sus siglas en inglés) juegan un papel crucial en la conexión de dispositivos IoT dentro de rangos de comunicación cortos. Estas tecnologías permiten que los dispositivos intercambien datos de forma inalámbrica estando diseñadas para comunicaciones de bajo consumo, bajo costo y de corto alcance.

La elección de la tecnología WPAN depende de los requisitos específicos del IoT, como eficiencia energética, tasas de datos y necesidades de interoperabilidad.

A continuación se brindará un mayor detalle sobre tales tecnologías.

- IEEE 802.15.4: publicado por primera vez en 2003 y modificado en 2006, es una tecnología de Red Personal Inalámbrica (WPAN) diseñada para aplicaciones IoT. Se enfoca en proporcionar comunicaciones inalámbricas de bajo consumo y bajo costo, siendo ideal para entornos con recursos limitados en energía, ancho de banda y procesamiento. Este estándar es

[11] *Wireless Personal Area Network*.

especialmente útil para redes de dispositivos en una topología en estrella, donde un dispositivo central coordina la comunicación.

Algunos de los principales usos son en hogares inteligentes, automatización industrial, ciudades inteligentes, salud y agricultura.

- Zigbee: es un estándar de comunicación inalámbrica que se utiliza principalmente para la comunicación bidireccional entre sensores y sistemas de control, y presenta como principales características el bajo consumo de energía, redes en malla (con nodos intermedios que mejoran la resiliencia y confiabilidad), la capacidad para conectar gran cantidad de dispositivos, seguridad robusta (incluye cifrado, autenticación y control de acceso) y garantiza interoperabilidad facilitando la integración de dispositivos de diferentes fabricantes.

  Los usos principales de esta tecnología se producen hogares inteligentes, tecnología verde (parques solares, eólicos y redes de carga de vehículos eléctricos), optimización del control y uso de dispositivos de energía, medicina (monitoreo remoto de la salud de los pacientes mediante sensores) y automatización industrial.

- HART[12]: es un protocolo digital utilizado en aplicaciones de automatización y control industrial. Se destaca en aplicaciones IoT por presentar compatibilidad con sistemas antiguos, facilitar la transmisión de datos entre dispositivos y controladores permitiendo control y monitoreo avanzado, bajo ancho de banda, seguridad (ofrece autenticación y cifrado) e interoperabilidad.

- NFC[13]: es un protocolo de comunicación inalámbrica ideal para aplicaciones IoT que requieren interacción a muy corta distancia. Alguno de los atributos que lo caracterizan son el de ofrecer seguridad (cifrado y autenticación), facilidad de uso (basta con acercar los dispositivos para que se comuniquen), interoperabilidad, y consumo de poca energía.

[12] *Highway Addressable Remote Transducer.*
[13] *Near-Field Communication.*

- Z-Wave: es un protocolo de comunicación inalámbrica diseñado para aplicaciones IoT, especialmente en hogares inteligentes y automatización, que se destaca por su fiabilidad y seguridad. Alguna de las características que presenta es que posee un bajo consumo de energía, admite redes en malla, emplea cifrado, autenticación y presenta interoperabilidad.

- Bluetooth/ BLE (Bluetooth *Low Energy*): tanto la tecnología Bluetooth como Bluetooth *Low Energy* (BLE), son dos protocolos de comunicación inalámbrica utilizados en aplicaciones IoT debido a su bajo consumo de energía, bajo costo y capacidad de comunicación a corta distancia.

  Sus características principales son la interoperabilidad, la seguridad y la compatibilidad con diversos tipos de dispositivos (teléfonos inteligentes, tablets y computadoras) para facilitar el control y conexión.

  Algunas de las aplicaciones de esta tecnología es el registro de actividades deportivas a través de relojes inteligentes, dispositivos médicos y automatización del hogar.

- Li-Fi[14]: la tecnología Li-Fi permite la comunicación inalámbrica transmitiendo datos a través de luz visible, específicamente el parpadeo rápido de LEDs. Esta tecnología es ideal para aplicaciones IoT que requieren alta velocidad, seguridad y baja latencia.

  Las principales aplicaciones de esta tecnología son en hogares inteligentes, en tiendas para transmitir información sobre productos a los teléfonos inteligentes de los clientes, atención sanitaria, automatización industrial y ciudades inteligentes.

- Wi-Fi[15]: es una tecnología de comunicación inalámbrica que utiliza ondas de radio para transmitir datos entre dispositivos. Es fundamental en aplicaciones IoT que demandan conectividad de alta velocidad, como monitoreo en tiempo real, automatización industrial, y plataformas en la nube.

[14] Li-fi: Light Fidelity.
[15] Wireless Fidelity.

Por otro lado, su uso es clave en la interconectividad, la escalabilidad de redes IoT grandes y la transmisión multimedia en aplicaciones de hogares inteligentes.

- Redes móviles (3G/4G/5G): las redes celulares, que incluyen tecnologías que van desde 3G y 4G hasta el cada vez más prevalente 5G, ofrecen una cobertura amplia y son capaces de transmitir mayores cantidades de datos a distancias más largas. Esto las hace adecuadas para dispositivos móviles o aplicaciones remotas donde no hay infraestructura de red local disponible.

  Adicionalmente, la comunicación por satélite se presenta como una opción crucial de conectividad, especialmente en escenarios donde las redes terrestres fallan o no están disponibles, como en entornos marítimos, aeroespaciales o en zonas terrestres extremadamente remotas.

  Este método, aunque a menudo más costoso y con mayor latencia, proporciona enlaces de datos vitales donde otras formas de conectividad no pueden llegar.

- LoRa[16] y LoRaWAN: LoRa es una tecnología de comunicación inalámbrica diseñada para ofrecer un largo alcance y bajo consumo de energía. Es escalable, segura, eficiente y basada en estándares abiertos, lo que la convierte en una opción ideal para diversas aplicaciones IoT en sectores como ciudades inteligentes, agricultura e industria.

  LoRaWAN es protocolo de red que usa la tecnología LoRa, para redes de baja potencia y área amplia (LPWAN, *Low Power Wide Area Network*) empleado para comunicar y administrar dispositivos LoRa.

  La diferencia entre ambas es que LoRa se refiere específicamente a la tecnología de modulación de radio utilizada para la comunicación inalámbrica de largo alcance y LoRaWAN se refiere al protocolo de

[16] Long Range.

comunicación y la arquitectura de red que permite la conectividad de dispositivos LoRa a través de una infraestructura de red gestionada. [16]

A continuación, se presentará un gráfico a modo de resumen con las tecnologías y protocolos mencionados anteriormente, y principales características:

Network	Connectivity	Pros and Cons	Popular use cases
Ethernet	Wired, short-range	☺ High speed ☺ Security ☹ Range limited to wire length ☹ Limited mobility	Stationary IoT: video cameras, game consoles, fixed equipment
WiFi	Wireless, short-range	☺ High speed ☺ Great compatibility ☹ Limited range ☹ High power consumption	Smart home, devices that can be easily recharged
NFC	Wireless, ultra-short-range	☺ Reliability ☺ Low power consumption ☹ Limited range ☹ Lack of availability	Payment systems, smart home
Bluetooth Low-Energy	Wireless, short-range	☺ High speed ☺ Low power consumption ☹ Limited range ☹ Low bandwidth	Small home devices, wearables, beacons
LPWAN	Wireless, long-range	☺ Long range ☺ Low power consumption ☹ Low bandwidth ☹ High latency	Smart home, smart city, smart agriculture (field monitoring)
ZigBee	Wireless, short-range	☺ Low power consumption ☺ Scalability ☹ Limited range ☹ Compliance issues	Home automation, healthcare and industrial sites
Cellular networks	Wireless, long-range	☺ Nearly global coverage ☺ High speed ☺ Reliability ☹ High cost ☹ High power consumption	Drones sending video and images

Figura 11. Resumen por tecnología de IoT: distancias, ventajas y desventajas y principales usos. [15]

5.2. Capa de internet

El conjunto de protocolos de TCP/IP[17] se utiliza para habilitar la comunicación entre dispositivos IoT e Internet. Principalmente, se utiliza IPv6, el cuál proporciona un espacio de direcciones más grande que IPv4, siendo más beneficioso para las aplicaciones IoT en las que puede haber muchos dispositivos que necesitan estar conectados a Internet. Por otro lado, IPv6 también proporciona características de seguridad mejoradas siendo crucial para garantizar la integridad y la confidencialidad de los datos transmitidos entre los dispositivos IoT e Internet.

- IPv4/IPv6: ambos protocolos son los estándar para la capa de Internet, y no es la excepción en el caso de las tecnologías IoT. No obstante, debe mencionarse que existe una variante del IPv6, denominado 6LoWPAN[18], el cual tiene una gran trascendencia en el campo de internet de las cosas y se lo desarrolla a continuación.
- 6LoWPAN: se trata de un protocolo diseñado para la comunicación de dispositivos IoT de bajo consumo y capacidad de ancho de banda limitada. Funciona como una capa de adaptación que permite transmitir paquetes IPv6 sobre redes inalámbricas de bajo consumo, como IEEE 802.15.4.

  A diferencia de protocolos como Bluetooth o Zigbee, 6LoWPAN es un protocolo de red que define mecanismos de compresión y encapsulación de encabezados, permitiendo a dispositivos pequeños con capacidades limitadas de procesamiento enviar información mediante IP.

  Algunas de las características destacadas de esta tecnología son:

  - Compresión de encabezados: reduce el tamaño de los paquetes IPv6, optimizando la transmisión en redes de bajo ancho de banda.

[17] Protocolo de Control de Transmisión/Protocolo de Internet.
[18] IPv6 over Low-Power Wireless Personal Area Networks.

- Fragmentación: permite fragmentar grandes paquetes IPv6 en fragmentos más pequeños para su transmisión eficiente.
- Direccionamiento: utiliza direcciones de 16 bits en lugar de 128 bits para reducir el tamaño de los encabezados y prolongar la vida útil de la batería.
- Redes en malla: soporta redes en malla, donde los dispositivos reenvían paquetes para extender el alcance de la red.
- Bajo consumo de energía: está optimizado para dispositivos de bajo consumo, minimizando el uso de energía para la transmisión de paquetes.
- Seguridad: ofrece mecanismos de encriptación y autenticación para garantizar la privacidad e integridad de los datos.

5.3. Capa de transporte

En el contexto de IoT, tanto el protocolo TCP (Protocolo de Control de Transmisión) como el UDP (Protocolo de Datagrama de Usuario), son dos de los protocolos de transporte más utilizados para definir cómo se transmite la información entre dispositivos. Estos protocolos tienen características diferentes y son adecuados para diferentes casos de uso de IoT, dependiendo de factores como fiabilidad, velocidad y condiciones de la red. A continuación se presentan las características de ambos protocolos y cómo se aplican en IoT:

- TCP: es un protocolo orientado a la conexión que garantiza una entrega confiable y ordenada de los datos entre dispositivos. Establece una conexión entre el emisor y el receptor antes de transmitir los datos y asegura que todos los paquetes de datos se reciban correctamente, en orden y sin duplicados. En definitiva, las principales características de este protocolo son:
    - Confiable: TCP asegura que los datos se entreguen sin pérdidas. Lo hace mediante paquetes de reconocimiento (ACKs) y retransmisiones de paquetes perdidos.

- Orientado a la conexión: se establece una conexión entre el emisor y el receptor antes de que comience la transmisión de datos.
- Entrega ordenada: los datos se reciben en el mismo orden en que fueron enviados.
- Control de flujo: TCP utiliza mecanismos como el *windowing*[19] para controlar el flujo de datos y prevenir la congestión de la red.
- Verificación de errores: cada paquete se verifica para detectar errores, y los paquetes corruptos se retransmiten.

Los casos de uso más tradicionales de TCP en IoT son los siguientes:

- Aplicaciones de hogares inteligentes que requieren comunicación confiable entre dispositivos como termostatos y sistemas de seguridad.
- Salud: donde la precisión y la fiabilidad son críticas, por ejemplo, transmitiendo datos vitales de pacientes a servidores médicos.
- Industrias (IIoT): donde la comunicación entre dispositivos como sensores, actuadores y sistemas de control debe ser altamente confiable.

- UDP: es un protocolo no orientado a la conexión, lo que significa que no establece una sesión entre el emisor y el receptor antes de transmitir los datos. Envía paquetes de datos (denominados datagramas) sin garantizar su llegada ni su orden, lo que lo hace más rápido pero menos confiable que TCP. En resumen, se mencionan algunas de las características que definen a este protocolo:
  - No confiable: UDP no garantiza que los paquetes se entreguen o en el orden correcto. Si se necesita fiabilidad, es responsabilidad de la aplicación gestionarlo.

[19] Dividir un flujo de datos continuo en subconjuntos más pequeños para un procesamiento y análisis más eficientes.

- Sin conexión: no es necesario establecer ni mantener una conexión entre los dispositivos, lo que reduce la sobrecarga y la latencia.
- Transmisión más rápida: UDP tiene una menor sobrecarga en comparación con TCP, lo que lo hace más rápido y eficiente en ciertos casos.
- Sin control de flujo: UDP no tiene control de flujo ni gestión de congestión, lo que significa que no ajusta su tasa de transmisión según las condiciones de la red.

Los principales usos que recibe el protocolo UDP en IoT, son los siguientes:

- Aplicaciones en tiempo real donde la velocidad es más importante que la fiabilidad, como transmisión de datos de sensores, comunicaciones de voz en dispositivos de hogar inteligente o videovigilancia.
- Dispositivos de bajo consumo que necesitan enviar actualizaciones pequeñas y periódicas, como monitoreo ambiental (temperatura, humedad) donde la puntualidad de los datos es más importante que la perfección.
- Transmisión de datos a múltiples dispositivos en una red, como servicios basados en ubicación o comunicación máquina a máquina (M2M) en IoT industrial, donde una gran cantidad de dispositivos necesita recibir la misma información.

5.4. Capa de aplicación

- AMQP (*Advanced Message Queuing Protocol*): es un protocolo estándar abierto de mensajería orientado a *middleware*[20], diseñado para proporcionar una comunicación confiable, segura y eficiente entre dispositivos y aplicaciones en entornos IoT. Este permite que las

[20] *Software* que se sitúa entre un sistema operativo y las aplicaciones que se ejecutan en él.

aplicaciones se comuniquen entre sí de manera asíncrona, lo que significa que no es necesario que estén en línea y disponibles simultáneamente para el intercambio de mensajes. Esta comunicación asíncrona es muy útil en arquitecturas de microservicios, aplicaciones en la nube y sistemas distribuidos en general. De esta manera podemos garantizar que ninguna solicitud se pierda por problemas de disponibilidad o errores que puedan ocurrir.

Cuando una aplicación envía un mensaje a otra a través de AMQP, el mensaje no se envía directamente al destinatario, se coloca en una cola. Esta cola es una estructura de datos que actúa como un intermediario entre el remitente y el destinatario, almacenando temporalmente los mensajes hasta que la aplicación receptora esté lista para procesarlos. [17]

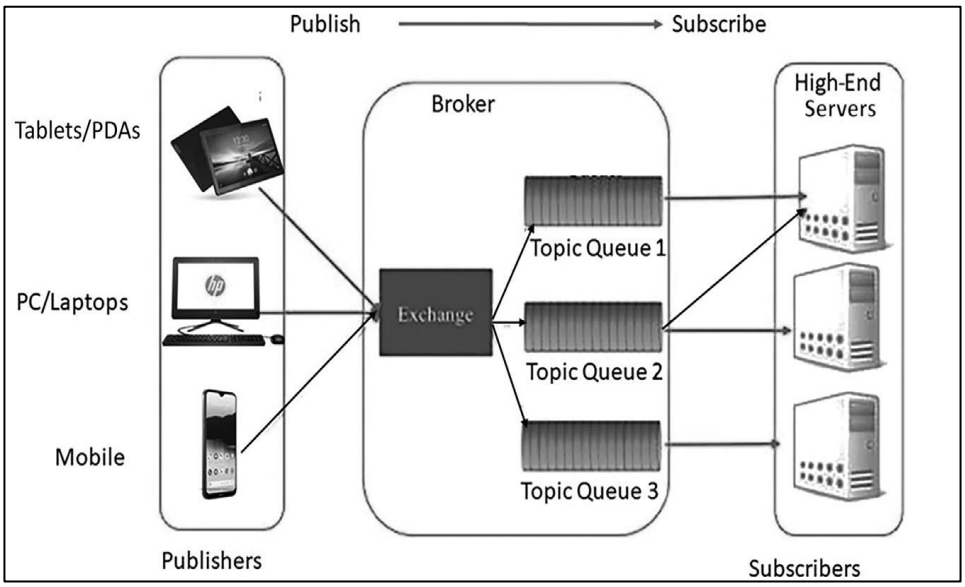

Figura 12. Arquitectura del protocolo AMQP. [4]

Algunas de las principales características que presenta son:

- Comunicación eficiente: permite el intercambio de mensajes entre dispositivos sin la necesidad de una conexión dedicada, facilitando la comunicación eficiente en IoT.

- Estandarización: como protocolo estandarizado, garantiza que dispositivos y aplicaciones se comuniquen utilizando un protocolo común, lo que simplifica la integración.
- Interoperabilidad: AMQP asegura que dispositivos y aplicaciones de diferentes proveedores puedan comunicarse, mejorando la compatibilidad en sistemas heterogéneos.
- Confiabilidad: proporciona entrega de mensajes garantizada, asegurando que los mensajes se entreguen correctamente y en el orden adecuado.
- Escalabilidad: soporta el manejo de grandes volúmenes de dispositivos y mensajes, con características como almacenamiento en cola y balanceo de carga, lo que facilita la expansión de redes IoT.
- Seguridad: ofrece mecanismos de seguridad como TLS/SSL y autenticación para proteger la comunicación en aplicaciones IoT donde la seguridad es crucial.

Si bien, este protocolo es mayormente utilizado para respaldar transacciones financieras en entidades bancarias, tiene su aplicación en IoT desde casas inteligentes, automatización industrial y monitoreo de salud a distancia. No obstante, cabe mencionar que el protocolo MQTT cuyas características son similares al que se está presentando, brinda mejores prestaciones para ser utilizado en el campo de la IoT.

- MQTT (*Message Queuing Telemetry Transport*): se trata de un protocolo de mensajería ligero y basado en el modelo publicación/subscripción, diseñado para ofrecer una comunicación eficiente, confiable y segura entre dispositivos y aplicaciones, especialmente en entornos IoT con recursos limitados. Este protocolo es ideal para redes restringidas, dispositivos con bajo consumo de energía, y escenarios con ancho de banda limitado o latencia elevada, como en aplicaciones de M2M (Machine-to-Machine).

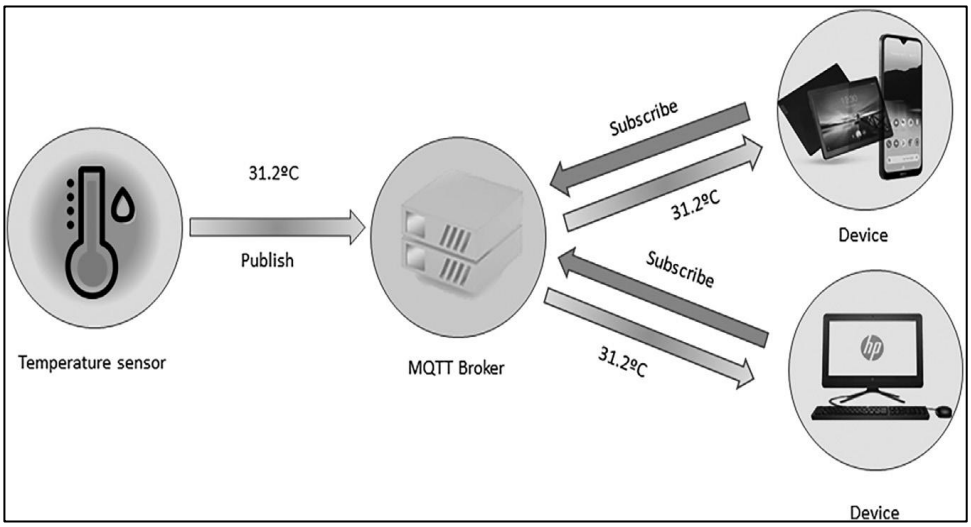

Figura 13. Arquitectura del MQTT. [4]

El protocolo presenta las siguientes características:

- Ligereza y bajo consumo de energía: MQTT es especialmente adecuado para dispositivos IoT con batería limitada, ya que utiliza pequeños paquetes de datos y un protocolo optimizado para redes de bajo ancho de banda.
- Modelo publicación/suscripción: los dispositivos publican mensajes a un broker (intermediario), que los distribuye a los suscriptores interesados en esos mensajes.
- Escalabilidad: MQTT puede conectar grandes cantidades de dispositivos, lo que lo hace ideal para aplicaciones IoT a gran escala.
- Fiabilidad: garantiza la entrega de mensajes incluso en presencia de fallos de red, utilizando diferentes niveles de calidad de servicio (QoS).
- Seguridad: ofrece opciones de seguridad como autenticación, encriptación y control de acceso, asegurando que solo los dispositivos autorizados puedan enviar o recibir mensajes.

No obstante, debe mencionarse que presenta ciertas limitaciones lo que no lo convierte en adecuado para aplicaciones en tiempo real que requieren respuestas inmediatas o transferencia de grandes cantidades

de datos. Asimismo, su principal enfoque es la mensajería ligera, por lo que no es ideal para aplicaciones que manejan grandes volúmenes de información.

Por otro lado, es fundamental hacer mención de la función bróker (intermediario) del protocolo MQTT. Este es el componente central que gestiona las comunicaciones entre los dispositivos y las aplicaciones. Sus funciones principales incluyen las de aceptar conexiones de los publicadores y suscriptores; administrar temas y suscripciones; filtrar, enrutar y entregar mensajes; manejar la autenticación y autorización de clientes; y monitorear el tráfico de mensajes.

Entre las principales aplicaciones de MQTT se pueden mencionar:

- Automatización industrial: comunicación entre sensores, actuadores y sistemas de control para supervisar y controlar procesos y máquinas.
- Automatización de hogar y edificios inteligentes: comunicación entre dispositivos como termostatos inteligentes, sistemas de iluminación y seguridad.
- Salud: transmisión de datos de dispositivos médicos y sistemas de monitoreo de pacientes a aplicaciones de salud para análisis y diagnóstico.
- Transporte: recopilación de datos sobre condiciones de tráfico, rendimiento de vehículos y comportamiento del conductor para optimizar el flujo de tráfico.
- Energía y servicios públicos: monitoreo de consumo de energía mediante medidores inteligentes y sistemas de gestión de energía.
- Agricultura: monitoreo de factores ambientales como la humedad del suelo y la temperatura para optimizar procesos agrícolas como riego y fertilización. [4]

- XMPP (*Extensible Messaging and Presence Protocol*): Es un protocolo de comunicación basado en XML utilizado en IoT para la comunicación en tiempo real entre dispositivos y aplicaciones. Originalmente diseñado para mensajería instantánea, ha sido adaptado para IoT, permitiendo el envío de mensajes, la gestión del estado de los dispositivos, y la comunicación entre diferentes servidores XMPP a través de una asociación (o *friendship*), la cual implica una subscripción mutua para aceptar las actualizaciones en ambos sentidos de la misma.

  XMPP funciona bajo una arquitectura cliente-servidor, donde los mensajes se envían desde el cliente a un servidor, que luego los redirige al destinatario adecuado.

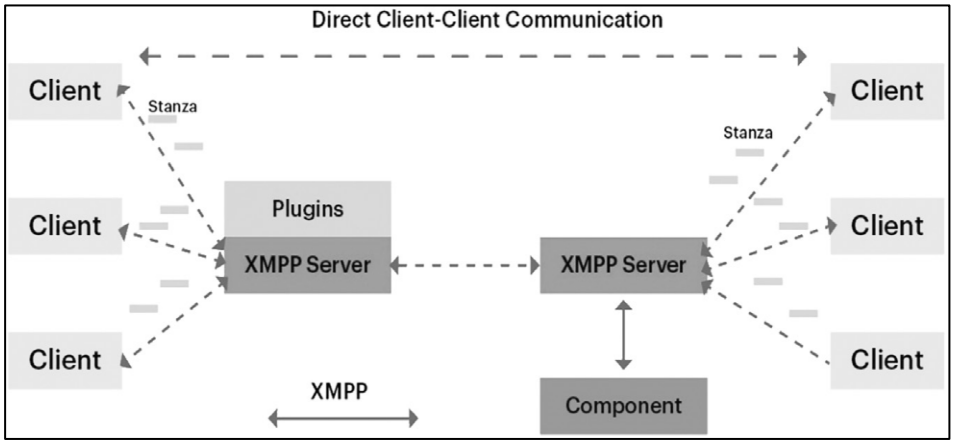

Figura 14. Arquitectura del XMPP. [4]

El protocolo presenta como principales beneficios:

- Comunicación en tiempo real entre dispositivos y aplicaciones.
- Mensajería para el intercambio de información y alertas.
- Presencia, indicando la disponibilidad y estado de los dispositivos.
- Escalabilidad, permitiendo manejar un gran número de dispositivos y mensajes.
- Seguridad, con mecanismos de cifrado y autenticación.
- Estandarización, asegurando la interoperabilidad entre dispositivos y aplicaciones.

- CoAP (*Constrained Application Protocol*): es un protocolo de transferencia web diseñado para dispositivos y redes con recursos limitados, como las que se encuentran en aplicaciones IoT. CoAP es especialmente adecuado para dispositivos con poca memoria y potencia, y se utiliza sobre UDP para minimizar la sobrecarga, lo que lo hace eficiente en cuanto a ancho de banda y consumo de energía.

  Algunas de las características principales de CoAP son:

  - Modelo Cliente-Servidor: los dispositivos pueden actuar como clientes que solicitan recursos o servidores que los proporcionan.
  - Operaciones CRUD: similar a HTTP (*Hypertext Transfer Protocol*), permite crear, leer, actualizar y eliminar recursos.
  - Observación de recursos: los clientes pueden suscribirse a recursos y recibir notificaciones sobre cambios, lo que es útil para aplicaciones en tiempo real.

  Las principales ventajas que ofrece son un alta eficiencia ya que usa UDP para reducir la sobrecarga y permite escalabilidad.

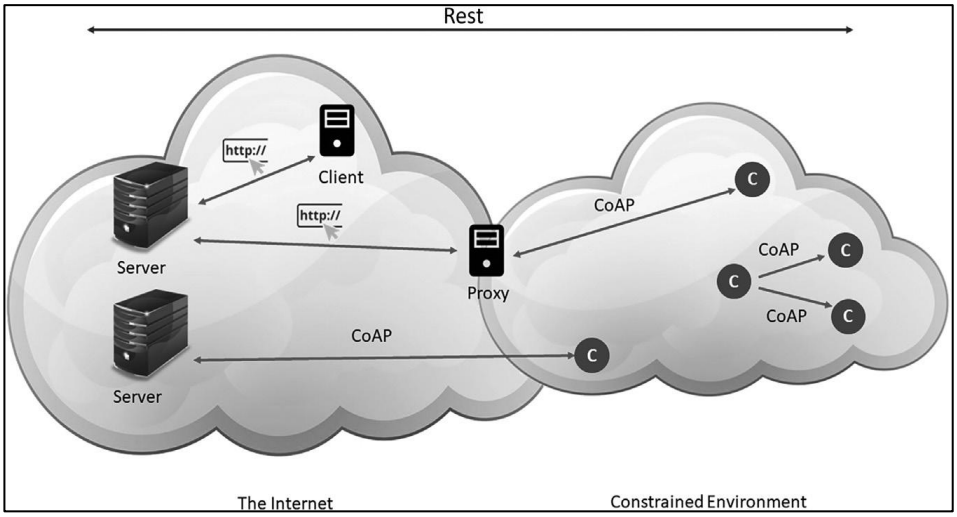

Figura 15. Arquitectura del CoAP. [4]

- DSS (*Data Distribution Service*): es un protocolo de tipo publicación/suscripción creado para sistemas de tiempo real que permite compartir datos con control de calidad de servicio. Las aplicaciones se comunican mediante la publicación y suscripción a temas identificados por

su nombre de tema. Las suscripciones pueden especificar filtros de tiempo y contenido y obtener solo un subconjunto de los datos que se publican en el tema. Los diferentes dominios DDS son completamente independientes entre sí, no compartiéndose datos directamente entre dominios DDS. [18]

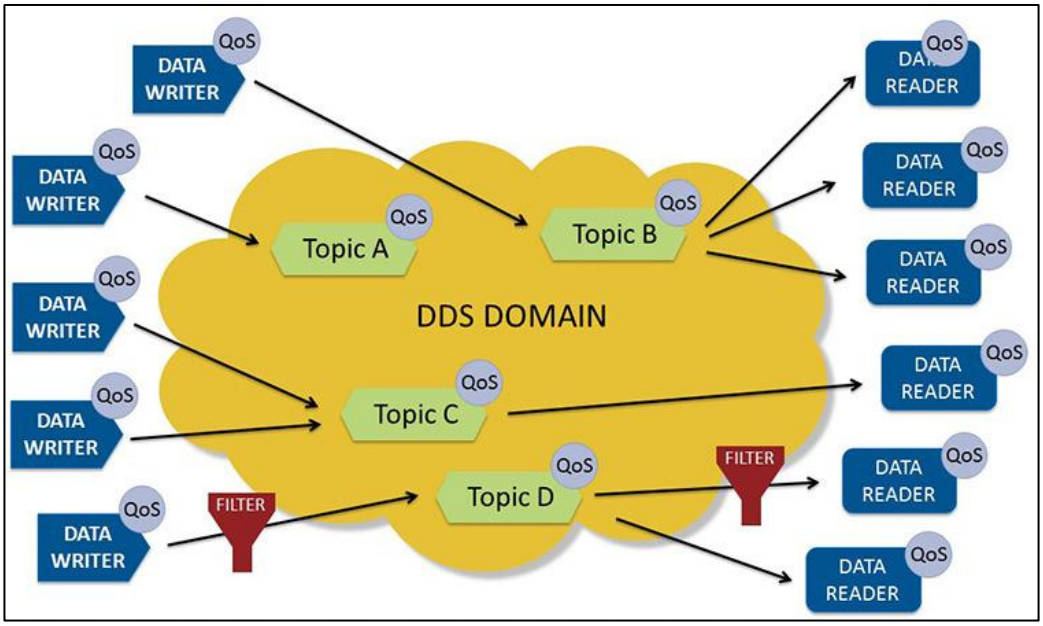

Figura 16. Arquitectura del DDS. [18]

A modo de resumen, a continuación se presentan algunos de los protocolos de la capa de aplicación con sus principales características:

	Transporte	Modelo	Ámbito de aplicación	Conocimiento del contenido	Datos principales	Seguridad	Prioridad de los datos	Tolerancia a fallos
AMQP	TCP/IP	Intercambio de mensajes punto a punto	D2D D2C C2C	Ninguno	Codificados	TLS	Ninguno	Específica de la implementación
CoAP	UDP/IP	Petición/Respuesta (REST)	D2D	Ninguno	Codificados	DTLS	Ninguno	Descentralizado
DDS	UDP/IP (unicast + mcast) TCP/IP	Publicación/Suscripción Petición/Respuesta	D2D D2C C2C	Enrutamiento basado en el contenido, consultas	Declarados codificados	TLS, DTLS, DDS	Prioridades de transporte	Descentralizado
MQTT	TCP/IP	Publicación/Suscripción	D2C	Ninguno	No definidos	TLS	Ninguno	El nodo central (broker) es el punto único de fallo (SPoF)

Figura 17. Tabla comparativa de algunos protocolos de capa de aplicación de IoT. [19]

6. Principales escenarios actuales y futuros

Actualmente la tecnología IoT es ampliamente utilizada tanto en actividades cotidianas como en entornos industriales. Se presentarán los principales usos de acuerdo con el ambiente en el que se desempeñan. Posteriormente, se describirán las actividades para las cuáles será utilizada esta tecnología en un futuro próximo.

6.1. IoT en actividades cotidianas

Dentro del ámbito hogareño y del uso doméstico por parte de los usuarios, las casas inteligentes y los dispositivos portátiles con conexión a internet son los más populares:

- Casas inteligentes: la evolución de la tecnología IoT tiene un gran impacto en cómo las personas interactúan en sus entornos domésticos, transformando las casas convencionales en hogares inteligentes. Esto se produce gracias a la interconectividad de diversos dispositivos y electrodomésticos a través de internet, lo que permite funcionalidades automatizadas y controladas de forma remota que mejoran el confort, la eficiencia energética y la seguridad.

  Uno de los ejemplos más populares del IoT en hogares inteligentes, es el control automático de la temperatura a través de ciclos de calefacción y refrigeración para maximizar el confort y minimizar el consumo de energía. Otra de las características fundamentales que presenta, es que se permite a los usuarios controlar estos dispositivos de forma remota mediante aplicaciones en sus teléfonos, lo que les posibilita ajustar la temperatura de la casa en cualquier momento, ya sea estando dentro de la casa o desde cualquier parte del mundo.

  Por otro lado, los modernos sistemas de seguridad para casas inteligentes incluyen una variedad de dispositivos conectados a internet como cámaras, detectores de movimiento y sensores de puertas. Esta conectividad permite a los propietarios monitorear su

propiedad de forma remota a través de transmisiones de video en vivo, recibir alertas cuando se detecta actividad sospechosa e incluso controlar las cerraduras de las puertas de manera remota. Por ejemplo, si un propietario se da cuenta de que olvidó cerrar la puerta, puede hacerlo desde su teléfono móvil, sin importar su ubicación física. [12]

Otra de las funcionalidades que provee IoT en el marco del hogar, es el control automático de la iluminación. Ya sea desde el encendido y apagado de los leds cuando las personas se encuentran en un determinado ambiente y luego dejan de estarlo, como el manejo automático de la intensidad de la luz en función de la iluminación que proviene del exterior.

Actualmente se está haciendo popular también la posibilidad de contar con un asistente virtual, al que se le conectan todos los dispositivos IoT de la casa, y se permite su control de manera remota a través de comandos de voz. Un ejemplo concreto, es la utilización de Alexa (propiedad de Amazon) para realizar tal acción.

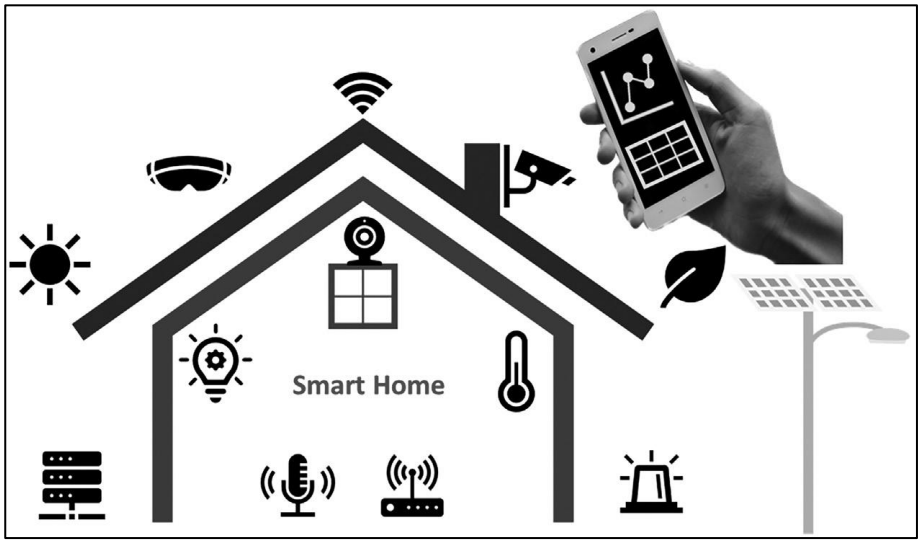

Figura 18. Escenario típico de IoT en los hogares. [4]

- Dispositivos portátiles: La tecnología portátil ha experimentado un aumento en popularidad, impulsada principalmente por aspectos vinculados a la salud y el bienestar. Estos dispositivos, que van desde

rastreadores de actividad deportiva hasta relojes inteligentes, fusionan tecnología de vanguardia con actividades humanas diarias para fomentar estilos de vida más saludables y mejorar la comunicación.

Los rastreadores de salud, por ejemplo, se encuentran entre las tecnologías portátiles más prevalentes. Estos dispositivos monitorean una serie de métricas fisiológicas como la frecuencia cardíaca, los patrones de sueño y los niveles de actividad física. Los modelos avanzados incluso pueden detectar ritmos cardíacos irregulares y otros posibles problemas de salud, lo que motiva a los usuarios a buscar asesoramiento médico cuando se detectan anomalías. Los datos en tiempo real recopilados a menudo se sincronizan con aplicaciones móviles o plataformas en línea, lo que permite a los usuarios seguir su progreso con el tiempo, establecer metas personales y compartir logros con amigos o comunidades en línea.

Por otro lado, los relojes inteligentes funcionan como extensiones del teléfono inteligente, permitiendo a los usuarios gestionar notificaciones, responder mensajes e incluso hacer llamadas directamente desde su muñeca. Además de seguir la salud y el estado físico, estos relojes se están utilizando cada vez más para realizar pagos, navegación GPS e incluso control remoto de dispositivos IoT del hogar.

La ropa inteligente equipada con sensores, por ejemplo, ofrece a los atletas retroalimentación en tiempo real sobre métricas de rendimiento como la postura, la longitud de los pasos y la actividad muscular. Innovaciones como los lentes y los audífonos inteligentes están integrando realidad aumentada y asistentes virtuales para mejorar tanto las actividades laborales como las de ocio, demostrando que el potencial de los dispositivos IoT personales es vasto y variado.

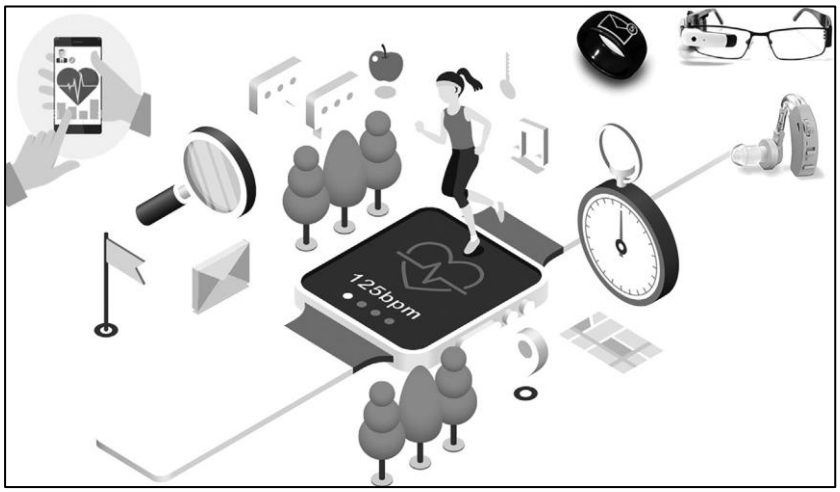

Figura 19. Escenario típico de IoT en dispositivos portátiles. [4]

6.2. IoT en entornos industriales y organizaciones

A continuación se describirán los principales usos de la tecnología IoT en sector industrial:

- Industrial IoT (IIoT): este tecnología representa un gran cambio en la manera en la que operan las industrias, haciendo uso de dispositivos conectados a internet y análisis avanzados para optimizar la producción, aumentar la eficiencia y minimizar los tiempos de inactividad. Esta transformación es especialmente notable en sectores como la manufactura, la logística y la agricultura, donde el IIoT no solo está mejorando los procesos, sino también redefiniendo el potencial de estas industrias.

  En la manufactura, en las instalaciones se integran diversos sensores, máquinas y herramientas de análisis para monitorear y automatizar los procesos de producción. Esto permite tener un mayor grado de personalización, ya que las líneas de producción pueden ajustarse dinámicamente en función de los datos en tiempo real sobre la demanda y suministro favoreciendo operaciones más flexibles y eficientes. Adicionalmente, los sensores recopilan datos directamente de la maquinaria para monitorear su rendimiento y estado salud, permitiendo predecir posibles fallos antes de que ocurran, lo que

permite un mantenimiento predictivo. Esta capacidad reduce significativamente los tiempos de inactividad no planificados y extiende la vida útil de los equipos.

La logística, por su parte, se beneficia de una mayor trazabilidad y eficiencia. Dispositivos como los rastreadores GPS y las etiquetas RFID ayudan a gestionar y monitorear la ubicación y el estado de las mercancías a lo largo de la cadena de suministro. Esto permite a las empresas optimizar las rutas, reducir los costos de transporte y mejorar la velocidad de las entregas. Adicionalmente, el IIoT ayuda en la gestión de inventarios, automatizando los controles de existencias y los procesos de reposición, lo que minimiza el riesgo de contar con *stock*[21] excedente o desabastecimiento.

En el ámbito de la agricultura, el IIoT está revolucionando las prácticas agrícolas tradicionales mediante la agricultura de precisión. Los sensores desplegados en los campos miden diversos parámetros ambientales como la humedad del suelo, la temperatura y la salud de los cultivos. Estos datos se utilizan luego para automatizar y controlar de manera precisa actividades como el riego, la fertilización y la aplicación de pesticidas, asegurando que los cultivos reciban exactamente lo que necesitan para un crecimiento óptimo.

Los drones y vehículos autónomos mejoran aún más estas capacidades al proporcionar imágenes aéreas a gran escala y realizar tareas como la siembra o la cosecha. Este nivel de precisión no solo aumenta los rendimientos y la calidad de los cultivos, sino que también reduce significativamente el desperdicio de recursos, apoyando prácticas agrícolas más sostenibles.

[21] Existencias de productos.

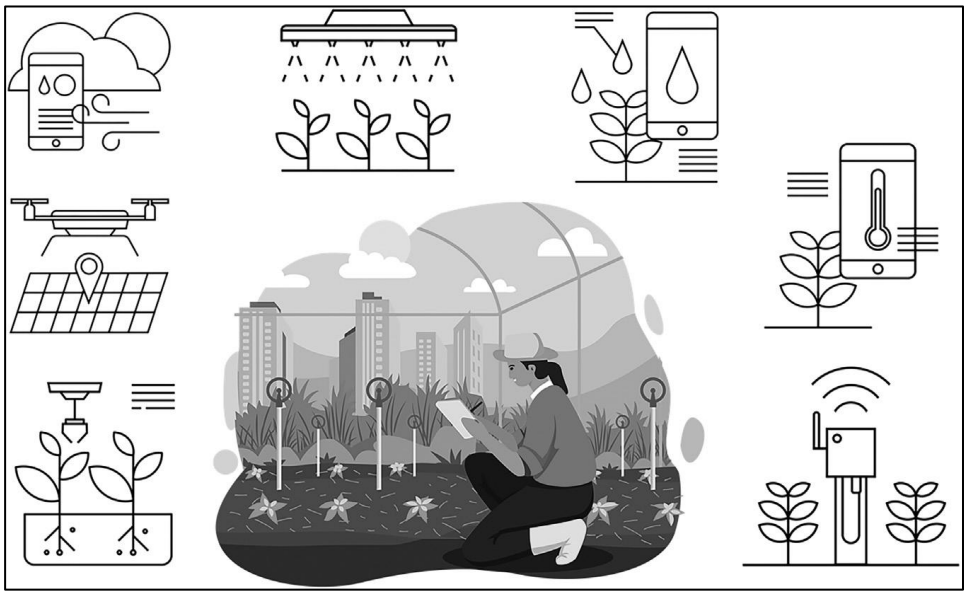

Figura 20. Escenario típico de IoT en la agricultura. [4]

El impacto del IIoT en estos sectores muestra su potencial no solo para agilizar las operaciones, sino también para crear sistemas más inteligentes y receptivos que puedan adaptarse a nuevos desafíos y oportunidades. A medida que las industrias continúan adoptando el IIoT, se espera que la integración de tecnologías inteligentes se profundice, impulsando una mayor eficiencia y generando nuevas posibilidades para la innovación y el crecimiento en el sector industrial. [12]

- IoT en la salud: el impacto de IoT en la atención médica (también llamado IoMT, Internet de las Cosas Médicas) ha sido revolucionario, reformulando cómo se brinda la atención, mejorando los resultados de los pacientes y optimizando las operaciones dentro de las instalaciones de salud. Se incluye una amplia variedad de aplicaciones, como sistemas de monitoreo de pacientes, dispositivos médicos inteligentes y gestión remota de la salud.

  Una de las aplicaciones más importantes es el monitoreo de pacientes, donde a través de equipamiento con sensores que poseen los mismos, se puede recopilar de manera continua datos vitales de salud, como la frecuencia cardíaca, la presión arterial, los niveles de glucosa y la saturación de oxígeno de los pacientes. Estos datos

pueden ser transmitidos en tiempo real a los proveedores de atención médica, quienes pueden monitorear las condiciones de los pacientes de manera remota, permitiendo intervenciones oportunas antes de que las condiciones empeoren.

La integración del IoT en la atención médica también plantea desafíos, especialmente en lo que respecta a la seguridad de los datos y la privacidad. El enorme volumen de datos sensibles de salud que se transmiten y almacenan aumenta el riesgo de brechas y accesos no autorizados. Por lo tanto, son esenciales medidas de ciberseguridad robustas para proteger los datos de los pacientes y garantizar el cumplimiento de los requisitos regulatorios.

- IoT en comercios minoristas: el sector minorista está paulatinamente incrementado la inclusión de IoT para transformar diversos aspectos de sus operaciones, especialmente en la gestión de inventarios y la mejora de la experiencia del cliente. Las tecnologías IoT, a través de la integración de sensores, etiquetas RFID y análisis avanzados de datos, están proporcionando a los minoristas niveles de eficiencia e información sin precedentes.

  En la gestión de inventarios, los dispositivos IoT juegan un papel crucial en la optimización de la cadena de suministro y la garantía de disponibilidad de productos. Las etiquetas RFID y los sensores, por ejemplo, se utilizan para rastrear productos a lo largo de la cadena de suministro en tiempo real. Esto permite a los minoristas reducir los casos de desabastecimiento y excesos de stock, situaciones que son costosas y perjudiciales para las operaciones comerciales. Al saber exactamente dónde se encuentran los artículos en la cadena de suministro, los minoristas pueden optimizar sus niveles de inventario e incluso automatizar los procesos de reposición.

  Adicionalmente, los datos de IoT permiten el análisis predictivo, lo que puede prever patrones de demanda futuros basados en datos

históricos, tendencias actuales del mercado y otros factores externos como el clima o las condiciones económicas.

Más allá del seguimiento de inventarios, los dispositivos IoT también mejoran la experiencia en la tienda para los clientes. Las estanterías inteligentes equipadas con sensores de peso y tecnología RFID pueden notificar a los empleados de la tienda cuando los productos se están agotando o cuando una estantería necesita ser reorganizada. Esto asegura que los productos estén siempre disponibles para los clientes y exhibidos de manera ordenada, mejorando la experiencia general de compra.

- IoT en los vehículos: en el sector automotriz, el IoT ha tomado una gran relevancia sobre todo en el caso de los coches autónomos. Los mismos se conectan a internet para disponer de un mapa de navegación actualizado, contar con novedades en la ruta (desvíos, colisiones, alertas, etc.) y siempre disponer de la ruta más eficiente para llegar a destino.

  Por otro lado, para su autoconducción, cuenta con una gran cantidad de sensores y cámaras que les permite identificar correctamente la pista por donde deben circular, discernir cuando deben girar, frenar y acelerar. Actualmente, ya se pueden observar también taxis de este tipo, que ya no cuentan con un conductor, sino que el auto se autoconduce para llegar al destino que el pasajero solicita.

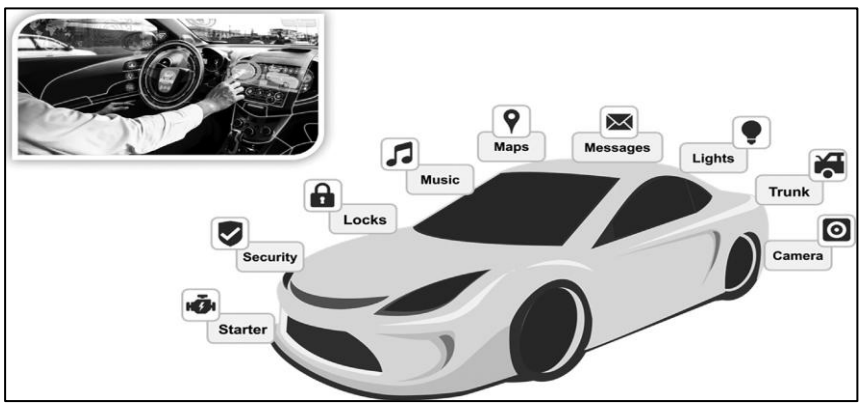

Figura 21. Escenario típico de IoT en sector automovilista. [4]

6.3. Usos futuros de IoT

Tal como se mencionó en secciones previas, la utilización de IoT está creciendo exponencialmente, de lo que se puede inferir que las aplicaciones en un futuro próximo podrían cubrir una vasta cantidad de posibilidades. No obstante, a continuación se mencionarán aquellas implementaciones que ya se encuentran en proceso de desarrollo y que en los próximos años será posible verlas en funcionamiento:

- Ciudades inteligentes: las ciudades inteligentes podrían generarse a través de la utilización de tecnologías avanzadas como sensores IoT, la computación en la nube, el análisis de datos y el aprendizaje automático para mejorar la infraestructura y los servicios urbanos, haciéndolos más eficientes, sostenibles y habitables. El objetivo principal de las ciudades inteligentes es mejorar la calidad de vida de los ciudadanos proporcionando mejores servicios, reduciendo costos y optimizando los recursos. A continuación, se presentan algunos de los ejemplos sobre como IoT podría emplearse en este contexto:
    - Gestión del Tráfico: los sensores pueden monitorear el flujo de tráfico en tiempo real, proporcionando datos a los sistemas de gestión de tráfico para optimizar las rutas, reducir la congestión y mejorar la seguridad. Los semáforos inteligentes podrían ajustar el tiempo de las señales según el volumen de tráfico, y los sistemas de estacionamiento inteligentes pueden guiar a los conductores a los espacios de estacionamiento disponibles, reduciendo la congestión vehicular.
    - Gestión de Residuos: los sensores podrían monitorear los niveles de llenado de los contenedores de basura, optimizando el proceso de recolección y reduciendo el costo de la recolección de residuos.
    - Gestión Energética: los sensores podrían monitorear el consumo de energía en edificios y otras infraestructuras urbanas,

proporcionando datos en tiempo real para optimizar el consumo de energía, reducir costos y disminuir las emisiones de carbono.

- Seguridad Pública: los sensores podrían monitorear espacios públicos, detectando posibles peligros de seguridad, como incendios, inundaciones y delitos.
- Monitoreo Ambiental: los sensores podrían monitorear factores ambientales como la calidad del aire, la calidad del agua y los niveles de ruido, proporcionando datos en tiempo real a las autoridades de la ciudad.

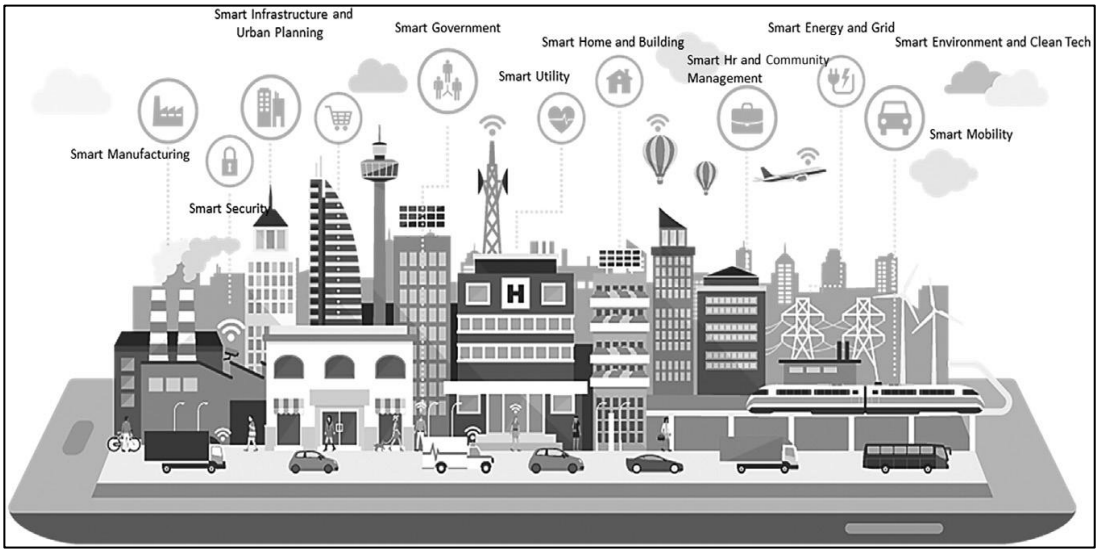

Figura 22. Escenario típico de IoT en ciudades inteligentes. [4]

En general, las ciudades inteligentes representan una oportunidad significativa para crear entornos urbanos más sostenibles, eficientes y habitables.

- Integración de IA en IoT: la inteligencia artificial (en adelante, IA) está preparada para mejorar significativamente las capacidades de los sistemas IoT, transformándolos de simples recolectores de datos en poderosas herramientas de toma de decisiones con funciones cognitivas avanzadas. A esta fusión de la IA con IoT, se la denominó como *Artificial Intelligence of Things* (AIoT), y se presume que a medida que la IA continúe evolucionando, su integración en las redes IoT redefinirá cómo

estos dispositivos van a interactuar con su entorno y como cada vez tomarán decisiones de una manera más autónoma.

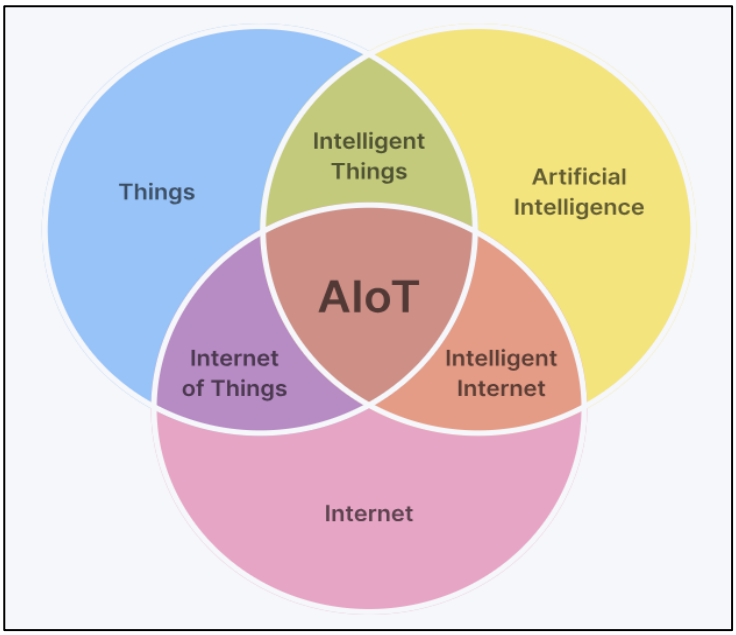

Figura 23. Conformación de la AIoT. [20]

En el sector minorista, la combinación de IA e IoT podría generar nuevas posibilidades para experiencias de compra personalizadas. Los dispositivos IoT equipados con capacidades de IA pueden recopilar y analizar datos de las interacciones de los consumidores y de los sensores ambientales dentro de una tienda en tiempo real.

En las futuras ciudades inteligentes, la IA podría ayudar a analizar los datos de varios sensores para gestionar el flujo de tráfico, reducir el consumo de energía y mejorar la seguridad pública.

En el sector de la salud, los dispositivos IoT impulsados por IA podrían monitorear las condiciones de los pacientes en tiempo real y a través de un análisis avanzado de patrones, ajustar los tratamientos según sea necesario sin intervención humana, incluso, permitiendo predecir posibles complicaciones futuras para poder anticiparse y evitarlas.

- IoT en blockchain: una de las principales ventajas de integrar blockchain con IoT es la capacidad de crear un registro transparente y auditado de

las transacciones de datos. Cada transacción de datos registrada en blockchain estará vinculada criptográficamente a transacciones anteriores, formando una cadena de bloques que no puede ser alterada ni eliminada sin el consenso de los participantes en la red. Esto asegura que los datos recopilados de los dispositivos IoT permanezcan seguros e inviolables, mitigando el riesgo de acceso no autorizado o manipulación.

La integración de la tecnología blockchain con IoT tiene el potencial de revolucionar la forma en que se recopilan, gestionan y comparten los datos en diversas industrias, particularmente en la gestión de la cadena de suministro. Al proporcionar una plataforma segura, transparente y descentralizada para registrar y verificar las transacciones de datos de IoT, blockchain mejorará la confianza, la responsabilidad y la eficiencia en todo el ecosistema de la cadena de suministro.

- Monitoreo del medio ambiente: tal como se mencionó en el desarrollo de ciudades inteligentes, el IoT también podría utilizarse para el monitoreo ambiental de todo el planeta, tomando información a través de los distintos sensores, que van desde la calidad del aire hasta los niveles de agua en ríos y embalses. Estas aplicaciones pueden ayudar a combatir la degradación ambiental y proporcionar los datos necesarios para abordar la contaminación y el cambio climático. Con los avances en los sensores y dispositivos IoT, el monitoreo ambiental en tiempo real puede volverse más generalizado, ofreciendo datos precisos para informar las políticas públicas y las prácticas corporativas.

- IoT y 5G: el despliegue de la tecnología 5G está preparado para potenciar las capacidades de IoT con una mayor velocidad, menor latencia y mayor conectividad. El 5G permitirá una nueva generación de aplicaciones de IoT, incluyendo infraestructuras de ciudades inteligentes más sofisticadas y confiables, aplicaciones mejoradas de IoT en la agricultura para monitorear y gestionar cultivos, y mejoras en la telemedicina, donde la transmisión instantánea de datos puede salvar vidas. [12]

- Robots domésticos: Los robots funcionarán como asistentes personales, capaces de gestionar horarios, proporcionar recordatorios e incluso ofrecer compañía. Ayudarán con una variedad de tareas, desde la planificación diaria hasta situaciones de emergencia y desempeñarán un papel importante en el monitoreo de la salud y el bienestar de los residentes, proporcionando recordatorios para la medicación, ayudando con ejercicios y alertando a los cuidadores en caso de emergencias.

  Por otro lado, se encargarán de una amplia gama de tareas domésticas, desde la limpieza y la lavandería hasta la jardinería y el mantenimiento del hogar, reduciendo la carga para los propietarios. La integración con electrodomésticos inteligentes permitirá que los robots gestionen tareas como la cocina, el lavado de platos y la gestión de inventarios en la cocina.

  En cuanto a la seguridad, los robots equipados con sensores avanzados y cámaras proporcionarán vigilancia mejorada, siendo capaces de patrullar el hogar, identificar intrusos y notificar a los propietarios o a las autoridades. [21]

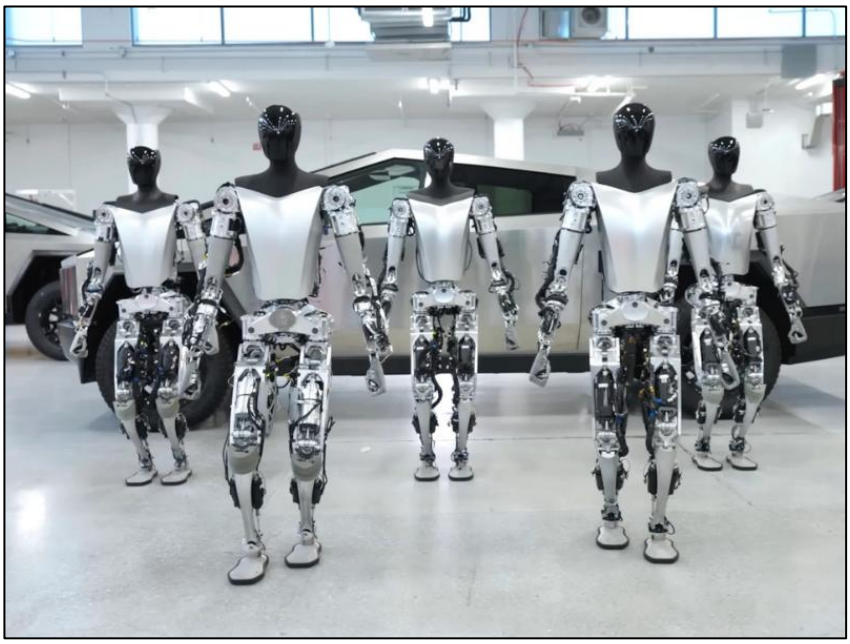

Figura 24. Robots Optimus (Tesla). [22]

7. Ventajas y desventajas

La utilización de la tecnología IoT, si bien presenta una serie de beneficios, al mismo tiempo presenta algunas características que pueden impactar negativamente en el ámbito organizacional y en la sociedad, sobre todo, en lo que respecta a la seguridad de la información.

En cuanto a las ventajas se pueden mencionar las siguientes:

- Tecnologías más avanzadas: la tecnología IoT ha evolucionado significativamente en los últimos años, esto implica que las empresas pueden hacer uso de una tecnología que posee: un procesamiento de mayor potencia para manejar mayores necesidades de datos y ancho de banda, mayores capacidades de almacenamiento, baterías más pequeñas y de mayor duración, y análisis más avanzados (utilizando IA).
- Menores costos operativos: una flota de dispositivos IoT puede ser rentable porque ayuda a las empresas a optimizar sus flujos de trabajo y reducir los costos operativos al proporcionar información en tiempo real. Los dispositivos pueden avisar proactivamente sobre su estado para que el personal pueda programar el mantenimiento antes de que afecte la producción. También pueden integrarse en sistemas más grandes para optimizar la eficiencia operativa y ayudar a reducir costos.
- Mayor productividad: los dispositivos IoT pueden gestionar, monitorear y alertar al personal sobre cambios en los procesos o en la productividad, ayudándoles a tomar decisiones más inteligentes sobre el trabajo.
- Perspectivas comerciales más avanzadas: los dispositivos IoT ayudan a las organizaciones a recopilar datos para identificar información valiosa sobre su negocio, tanto interna como externa.
- Mejores experiencias para el cliente: los dispositivos IoT ayudan a las empresas a rastrear, monitorear, descubrir y analizar los datos de los clientes más rápido que antes. Las empresas pueden predecir cambios o tendencias en el comportamiento del cliente antes de que ocurran.

- En el ámbito personal y hogareño, se presenta un serie de beneficios como el tracking de los parámetros de salud y la mejora en el confort y la seguridad.

Por otro lado, presenta las siguientes desventajas:

- Desafíos de seguridad y privacidad de datos: aunque la ciberseguridad es una prioridad alta, los dispositivos IoT no siempre se incluyen en la estrategia de seguridad. Estos deben ser protegidos contra manipulaciones físicas, ataques de software basados en Internet, ataques a la red y ataques basados en hardware. La privacidad de los datos es otra preocupación, especialmente porque los dispositivos IoT se están utilizando en industrias más sensibles, como la salud y las finanzas. Integrar medidas de encriptación y seguridad con dispositivos IoT puede ser difícil cuando se trata de una flota grande de dispositivos.
- Tecnología oculta: implementar dispositivos IoT en cualquier empresa puede tener una curva de aprendizaje considerable. Tiene sentido desarrollar una estrategia sobre cómo y por qué implementarlos antes de su adopción. De esa manera, las empresas pueden asegurarse de que estén funcionando como se espera y que sean fácilmente compatibles.
- Desafíos de gestión del cambio en IoT: las empresas tienden a considerar el IoT solo como una tecnología y no tienen en cuenta las implicaciones que tiene sobre los procesos empresariales. Esto significa que a menudo dejan la implementación y el mantenimiento exclusivamente a los equipos de TI, sin considerar los efectos legales, industriales o internos.
- Dependencia de conectividad y energía: muchos dispositivos dependen de una conexión continua a la energía o a internet para funcionar correctamente. Cuando cualquiera de estos se interrumpe, los dispositivos y todo lo que está conectado a ellos también dejan de funcionar. Por lo tanto, las empresas deben comprender cómo los cortes de energía o de conectividad afectarán a sus dispositivos para planificar un escenario de contingencia y poder seguir operando.

- Desafíos de interoperabilidad: actualmente no hay un consenso respecto a los protocolos y estándares del IoT, por lo que los dispositivos fabricados por diferentes proveedores podrían no funcionar con la pila tecnológica existente. Cada uno puede requerir configuraciones y conexiones de hardware diferentes, lo que hace que el despliegue eficiente sea difícil. [23]

8. Principales amenazas y vulnerabilidades

En el ámbito de IoT, existen muchas vulnerabilidades debido a su estructura en la que intervienen varios niveles, subniveles de hardware y software en aplicaciones y servicios.

El Open Web Application Security Project (OWASP) ha abordado los problemas de seguridad asociados con el IoT con el propósito de ayudar a desarrolladores, fabricantes y consumidores. OWASP es de código abierto y tiene una política de licencias gratuita. El proyecto es una iniciativa de desarrollo de software basada en el modelo comunitario. Un modelo comunitario implica esfuerzos e iniciativas colectivas por parte de universidades, organizaciones e instituciones en un proyecto de código abierto.

OWASP ha emprendido una serie de subproyectos relacionados con la seguridad, como los destinados a definir las "Principales Vulnerabilidades", "Áreas de Superficie de Ataque" y "Guías de Pruebas".

En tal sentido, OWASP ha identificado las diez vulnerabilidades principales en aplicaciones/servicios IoT, las cuales se presentan a continuación, partiendo de la de mayor riesgo a la de menor:

- Contraseñas débiles, fáciles de adivinar o embebidas en el código: hace referencia a la utilización de contraseñas que se pueden romper fácilmente a través de ataques de fuerza bruta, que se encuentran disponibles públicamente (por ejemplo, en diccionarios de contraseñas comunes), o bien contraseñas que no se modifican periódicamente.
- Servicios de red inseguros: son aquellos servicios de red o puertos inseguros que se encuentran funcionando en los dispositivos, especialmente aquellos expuestos a internet, que comprometen la confidencialidad, integridad y/o disponibilidad de la información y permite control remoto sin autorización.
- Ecosistema de interfaces inseguras: hace referencia a la utilización de interfaces inseguras, a modo de ejemplo, interfaces web, de la nube, APIs, y de los dispositivos en sí, que hace que los mismos queden

comprometidos frente a diferentes tipos de ataques. Algunos de los conflictos más conocidos son la falta de autenticación/autorización, ausencia o debilidad en la encriptación, y falta filtros tanto para la salida o entrada de los datos.

- Ausencia de un mecanismo seguro para la actualización: falta de un proceso seguro para actualizar los dispositivos. Esto también incluye la falta de validación del firmware en el dispositivo, falta de encriptación de la información en tránsito, ausencia de mecanismos de vuelta atrás en caso de actualizaciones fallidas o que generan incompatibilidades, y falta de notificaciones de seguridad respecto a cambios en la configuración derivadas de las actualizaciones.

- Utilización de componentes inseguros o desactualizados: hace referencia a la utilización de componentes que ya no cuentan con soporte por parte del proveedor, o bien, aun teniendo soporte, no se le han aplicado los últimos parches de seguridad. Esto puede ocurrir en equipos propios, como de proveedores directos o pertenecientes a la cadena de suministro.

- Insuficiente protección de privacidad: hace referencia a la información personal de los usuarios que está almacenada en los diferentes dispositivos o en el mismo ecosistema IoT, que es usada de manera insegura, inapropiada o sin permiso.

- Transferencia y almacenamiento de datos inseguro: es la falta de encriptación o control de acceso respecto a la información sensible dentro de todo el ecosistema IoT, estando la misma en reposo, en tránsito o en procesamiento.

- Deficiente gestión de dispositivos: la falta de soporte respecto a la seguridad de los dispositivos desplegados en producción, incluida la gestión de activos y actualizaciones, monitoreo de sistemas y capacidades de respuesta.

- Configuraciones predeterminadas inseguras: se refiere a la utilización de dispositivos sin modificarles la configuración que poseen de fábrica. Un

ejemplo habitual, es el de las cámaras de videovigilancia, las cuales suelen tener nombres de usuario y contraseñas por defecto. Lo mismo ocurre con los routers, que muchas veces facilita el acceso en modo administrador en las redes públicas.

- Debilidades en la seguridad física: la ausencia de medidas de control sobre la seguridad física, favorece a potenciales atacantes tener acceso a información sensible que luego puede ser utilizada en futuros remotos ataques o tomar control local sobre el dispositivo.

8.1. Explotando la debilidad *top* 1 de OWASP en IoT (cámara IP)

Tal como se pudo observar en la sección “6.1. IoT en actividades cotidianas”, hoy en día se encuentra en auge el desarrollo de las casas inteligentes. Esto se debe a la incorporación de dispositivos que se conectan a la red Wi-Fi, desde la cual se configuran, utilizan y administran de manera local y también remota. Algunos de ellos son cámaras de videovigilancia, lavarropas, heladeras, dispositivos móviles (celulares, equipos portátiles, relojes inteligentes, etc.), entre otros.

Cabe destacar que, todos estos dispositivos que se encuentran dentro de la red hogareña, están conectados a un router que les permite tener acceso a internet. De esto se desprende que, es de vital importancia que tal dispositivo se encuentre configurado de acuerdo con buenas prácticas, ya que frente a una brecha de seguridad, un intruso podría infiltrarse en la red y comenzar a realizar distintos tipos de actividades. Por ejemplo, detectar los dispositivos que se encuentran conectados a dicha red, e identificar los servicios y protocolos que corren con el fin de hallar diferentes vulnerabilidades. A partir de esta acción, el ciberdelincuente podría llegar a explotar alguna de ellas, ingresando al dispositivo e instalando algún malware que lo convierta en un *bot* más de la *botnet* que gestiona.

Por otro lado, el acceso a los diferentes dispositivos IoT, le permitiría al atacante contar con información confidencial de la víctima, ya sean documentos almacenados, acceso a la cámara IP, etc.

Considerando lo anteriormente mencionado, y en conjunción a la primera vulnerabilidad publicada en el OWASP *Top* 10 de IoT que hace mención a la utilización de contraseñas débiles o fáciles de adivinar, se procederá a efectuar una actividad de laboratorio, que consistirá en intentar lograr tener acceso a la transmisión de una cámara IP que se encuentra funcionando dentro de una red privada.

Para ello, la actividad se compondrá de dos etapas. En la primera de ellas, se realizará un ataque al router de tal red, para obtener la clave del Wi-Fi que permita lograr la infiltración, haciendo uso de las distintas herramientas contenidas en el sistema operativo Kali Linux, las cuáles se mencionarán en la próxima sección. Una vez obtenida la misma, y lograda la infiltración, se procede a efectuar la segunda etapa. Esta radica en escanear la red con el fin de detectar los *hosts* que se encuentran activos e identificar la dirección IP que corresponda a la cámara. Obtenida la misma, se analizarán los servicios que se encuentran habilitados y se atacará al protocolo “RTSP” mediante fuerza bruta, para obtener las credenciales de acceso a la transmisión de tal dispositivo.

8.1.1. Herramientas a utilizar

Para el desarrollo del laboratorio se utilizarán los siguientes equipos y aplicaciones:

Equipos:

- Router TP LINK TL-WR840N (300M Wireless N Router).
- Notebook Lenovo Legion 7i.
- Cámara IP Gadnic.

Aplicaciones:

- Suite Aircrack-ng: Airmon-ng, Airodump-ng, Aireplay-ng, Aircrack-ng.
- VLC Media Player.
- NMAP.
- Hydra.
- Wireshark.

8.1.2. Configuración del router

La configuración del router es la siguiente:

- SSID[22] = MSI-UBA-TP-ESP
- Contraseña = Administrator

8.1.3. Etapa 1: Ataque de fuerza bruta a Wi-Fi de red IoT hogareña

8.1.3.1. Monitoreo de la red

En primer lugar se procede a colocar a la placa de red en modo monitor a través del comando "# airmon-ng start wlan0":

```
┌──(root㉿kali)-[~]
└─# airmon-ng start wlan0

Found 2 processes that could cause trouble.
Kill them using 'airmon-ng check kill' before putting
the card in monitor mode, they will interfere by changing channels
and sometimes putting the interface back in managed mode

    PID Name
   1653 NetworkManager
   1717 wpa_supplicant

PHY     Interface       Driver          Chipset

phy0    wlan0           iwlwifi         Intel Corporation Comet Lake PCH CNVi WiFi
                (mac80211 monitor mode vif enabled for [phy0]wlan0 on [phy0]wlan0mon)
                (mac80211 station mode vif disabled for [phy0]wlan0)
```

Figura 25. Se coloca interface de red en modo monitor.

Se puede observar a través del comando "# ifconfig" que la interface de red se encuentra monitoreando:

```
┌──(root㉿kali)-[~]
└─# ifconfig
eth0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether 7c:8a:e1:55:a0:11  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 20  bytes 1344 (1.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 20  bytes 1344 (1.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlan0mon: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        unspec 3C-9C-0F-6D-FB-C3-00-40-00-00-00-00-00-00-00-00  txqueuelen 1000  (UNSPEC)
        RX packets 318  bytes 98137 (95.8 KiB)
        RX errors 0  dropped 318  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Figura 26. Wlan0mon activa.

[22] Identificador de Conjunto de Servicios

Posteriormente, se procede a ingresar el comando "# airodump-ng wlan0mon" para poder observar que redes se encuentran alcanzadas por nuestra placa de red, siendo "MSI-UBA-TP-ESP" a la que queremos lanzar el ataque, que posee la cámara IP conectada.

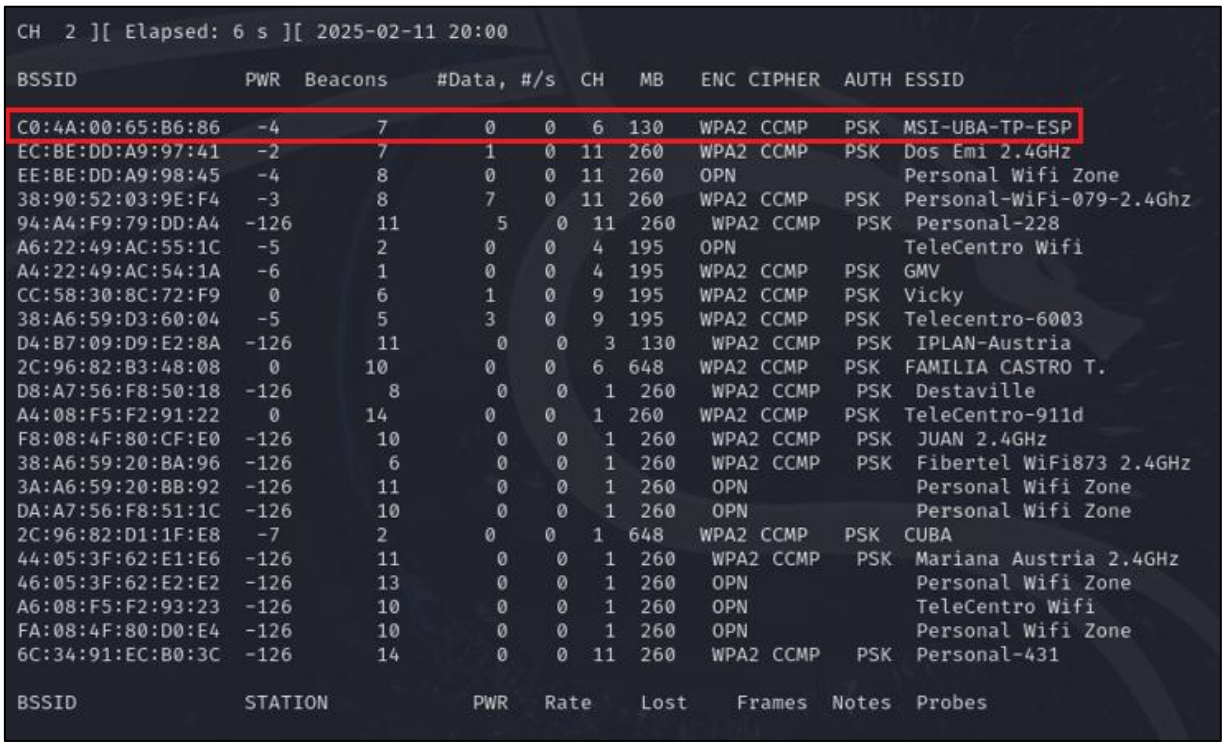

Figura 27. Redes alcanzadas por la placa de red.

8.1.3.2. Captura de tráfico y ataque de desautenticación

Identificada la red de la cual queremos obtener las credenciales, se comienzan a capturar los paquetes de la misma a través del comando "# airodump-ng -c6 -w capture -d C0:4A:00:65:B6:86 wlan0mon", con el objetivo de determinar que *hosts* se encuentran conectados a ella:

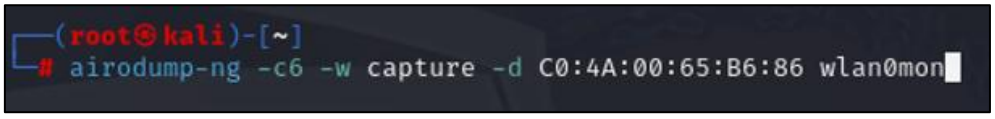

Figura 28. Uso de airodump-ng para capturar el tráfico de red.

Los parámetros del mismo se detallan a continuación:

- -c6: se establece que se capturen paquetes en el canal 6, que es donde opera la red a atacar.

- -w capture: se indica que se guarde la captura en un archivo llamado "capture".
- -d C0:4A:00:65:B6:86: se indica la dirección MAC-ADDRESS[23] del router de la red a atacar.

Se puede observar que existen dos dispositivos conectados a tal router:

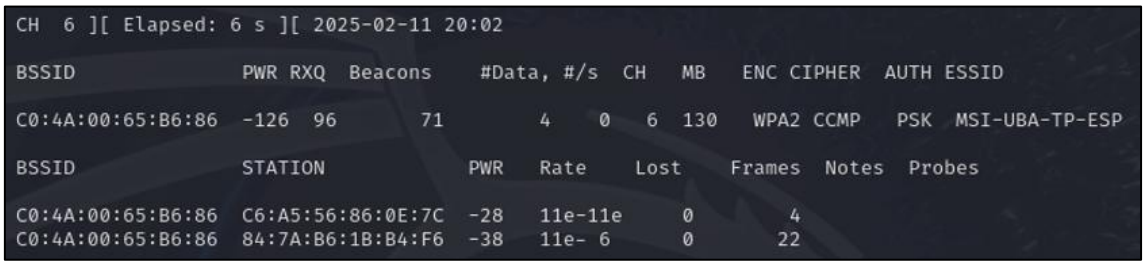

Figura 29. *Hosts* conectados al router.

Posteriormente, se inicia un ataque de desautenticación a uno de los *hosts* que se encuentran conectados, con el fin de que automáticamente se vuelva a conectar a la red, permitiendo capturar el hash de la contraseña. Para ello, se ingresa el comando "# aireplay-ng -0 0 -a C0:4A:00:65:B6:86 -c C6:A5:56:86:0E:7C wlan0mon":

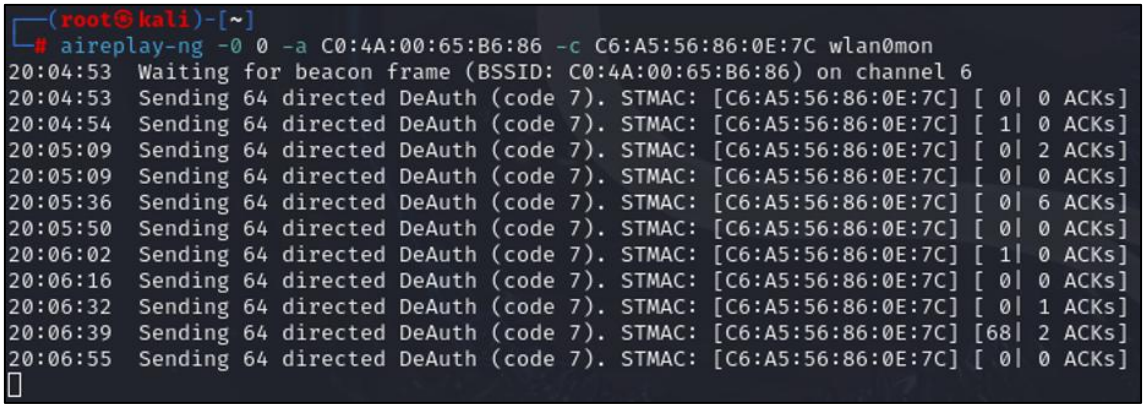

Figura 30. Envío de paquetes de desautenticación.

Los parámetros del mismo se detallan a continuación:

- -0 0: el primer 0 indica que es un ataque de desautenticación. El segundo menciona el envío continuo de paquetes para desautenticar.
- -a: C0:4A:00:65:B6:86: se indica la MAC-ADDRESS del router.

[23] Identificador único de un dispositivo o interfaz de red.

- -c 08:38:E6:8D:25:3D wlan0mon: se indica la MAC-ADDRESS del equipo a desautenticar y a través de que interface se enviará el paquete.

Cuando se muestra el mensaje “PMKID found”, significa que ya capturamos el hash de la contraseña, y ya se puede dejar de enviar los paquetes de desautenticación a la víctima:

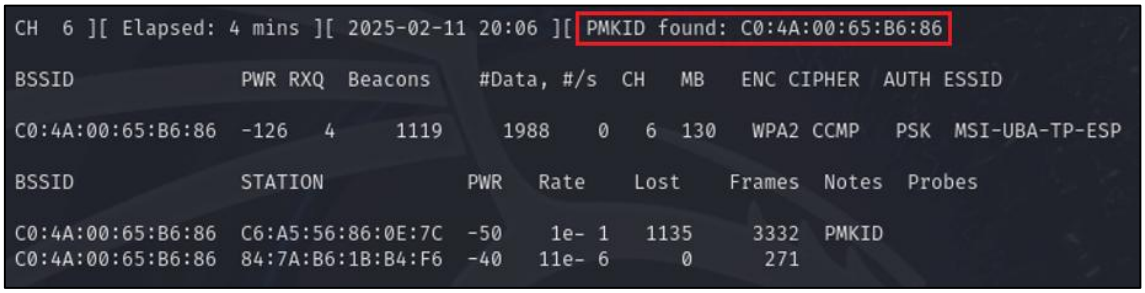

Figura 31. Aviso de credencial capturada.

Se procede a modificar el modo de monitoreo de la placa de red, para que vuelva a operar normalmente:

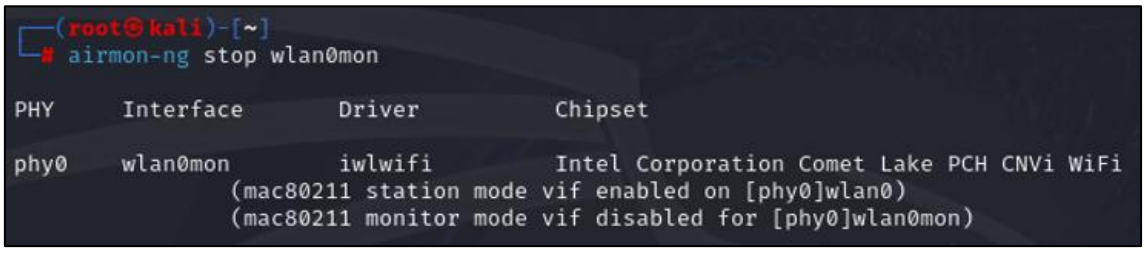

Figura 32. Detención del modo monitor.

8.1.3.3. Ataque de fuerza bruta sobre el hash de la clave

Realizado el proceso previo, se cuenta con el hash de la contraseña en el archivo “capture”, que es donde se encuentra la captura del tráfico del dispositivo víctima. Para chequear que realmente se dispone de tal información, se procede a abrir el archivo “capture.cap” en la herramienta Wireshark para verificar que exista el paquete de autenticación EAPOL[24]:

[24] Protocolo de Autenticación Extensible sobre LAN (EAPoL), es un protocolo de autenticación de puertos de red utilizado en IEEE 802.1x o control de acceso a la red basado en puerto.

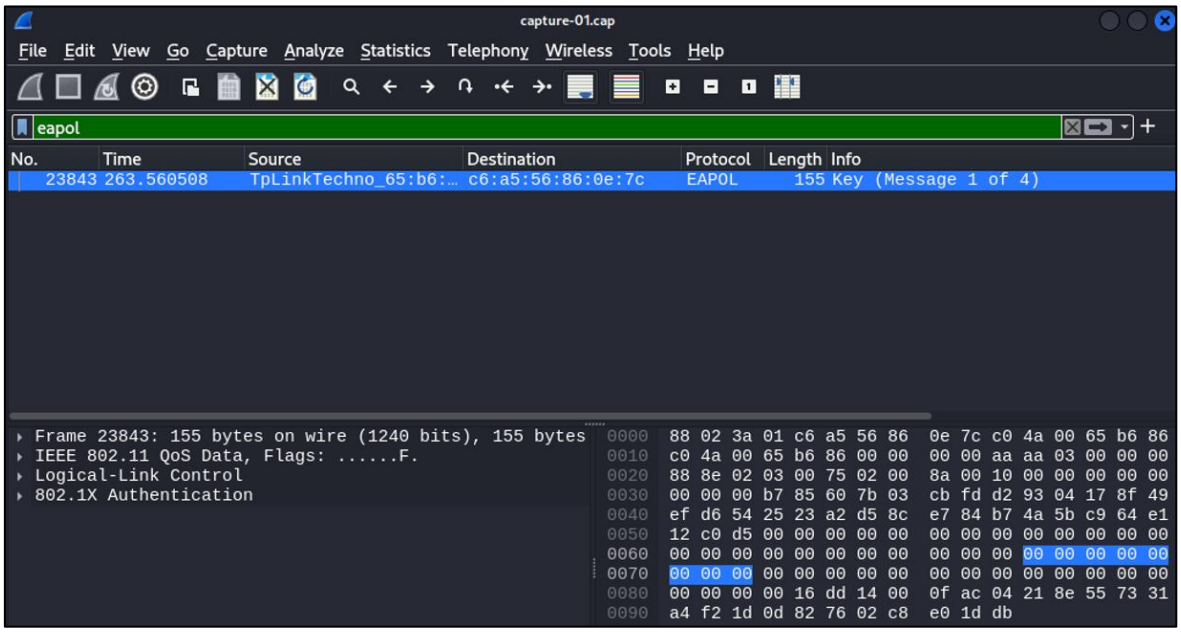

Figura 33. Paquete de autenticación EAPOL con las credenciales.

Una vez que se cuenta con la captura de tráfico que incluye el hash de la contraseña, se procede a utilizar la herramienta “aircrack-ng” y el diccionario de contraseñas “rockyou” para efectuar el ataque de fuerza bruta. Para ello, se ingresa el siguiente comando:

“# aircrack-ng -w /usr/share/wordlists/.rockyou.txt /home/Kali/Desktop/capture-01.cap”:

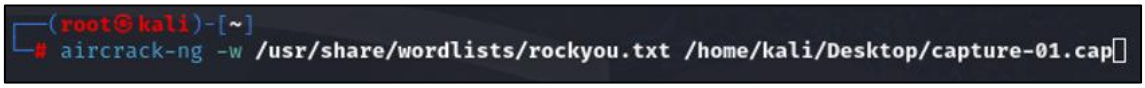

Figura 34. Utilización de aircrack-ng con el diccionario rockyou.txt.

Los parámetros se detallan a continuación:

- -w /usr/share/wordlists/rockyou.txt: -w indica que se utilizará un archivo de entrada (diccionario de contraseñas) para hacer el ataque de fuerza bruta y a continuación se detalla la ruta donde se encuentra guardado el mismo.
- /home/kali/Desktop/capture01.cap: es el archivo que contiene la captura con las credenciales a descifrar.

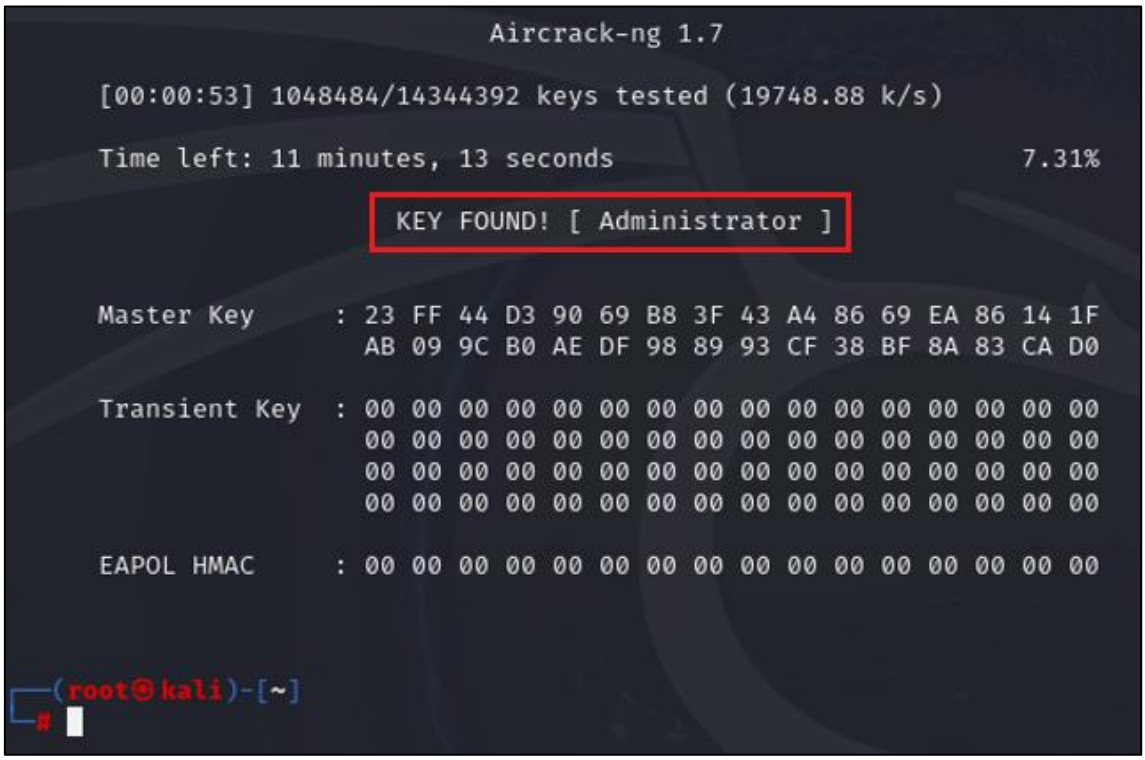

Figura 35. Se encuentra el texto plano del hash de la clave (Administrator).

Ejecutado el mismo, se puede observar que se encontró la contraseña de la red “MSI-UBA-TP-ESP”, y la misma es “Administrator”.

8.1.4. Etapa 2: Ataque de fuerza bruta a cámara IP

8.1.4.1. Escaneo de la red y detección de la dirección IP de la cámara

Obtenida la clave del router para poder acceder a la red, se ingresa y se identifica cual es la dirección de red y máscara de subred, con el propósito de realizar un escaneo que permita identificar los equipos que se encuentran allí conectados:

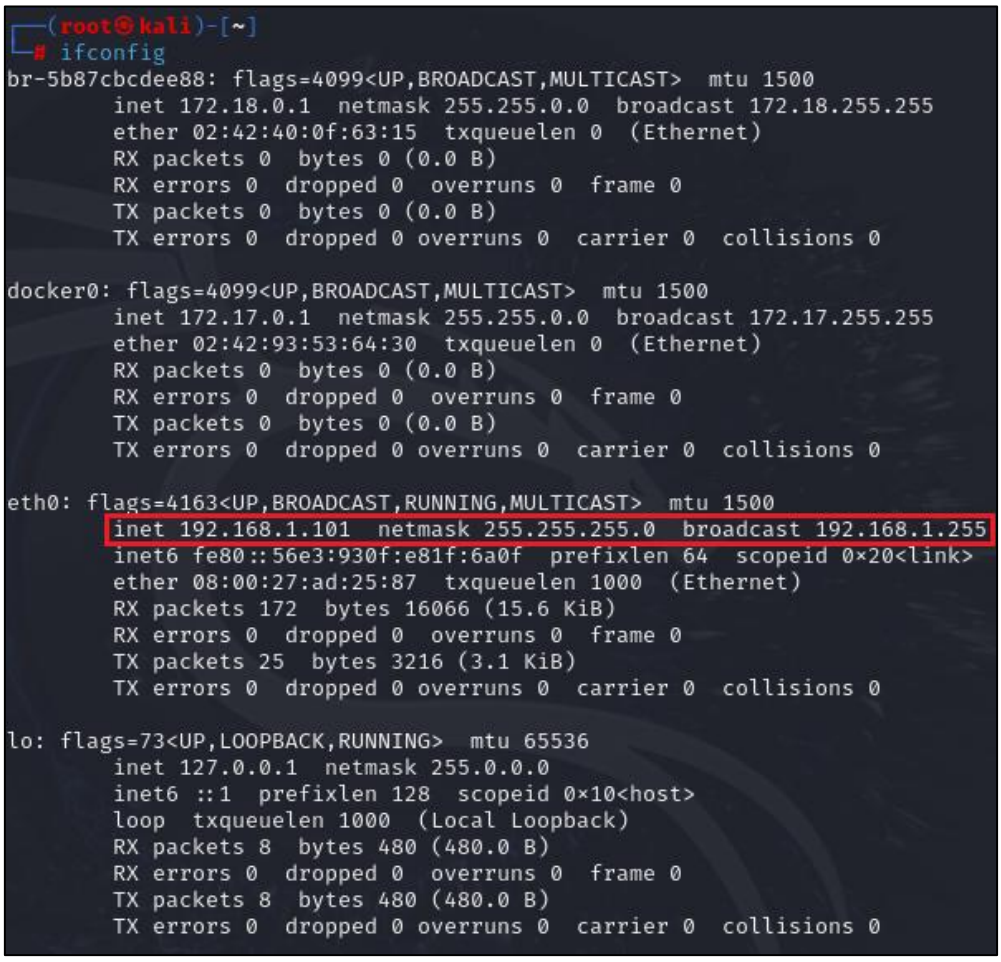

Figura 36. Se puede observar que la red privada tiene el rango 192.168.1.0/24.

Tal como se puede observar en la figura previa, la red a escanear será "192.168.1.0/24". Para ello se ingresa el comando "#nmap -sP 192.168.1.0/24":

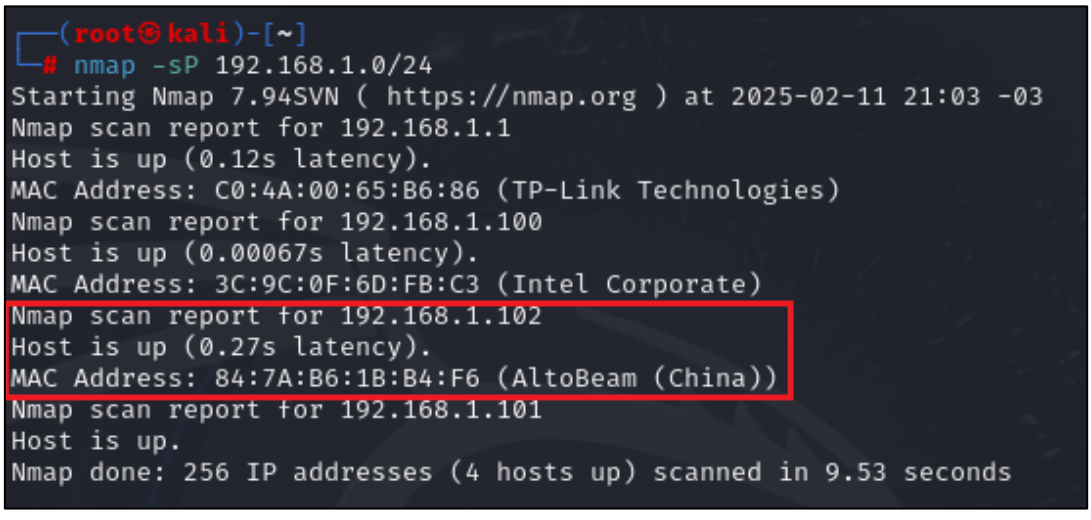

Figura 37. *Hosts* activos en la red.

En base a los *hosts* detectados, se puede inferir que aquel que posee asignada la IP “192.168.1.102” es la que corresponde a la cámara IP, debido a que se menciona el fabricante llamado “AltoBeam”. Esta empresa es reconocida por fabricar y comercializar dispositivos electrónicos que se conectan al Wi-Fi, siendo gran parte de ellos, dispositivos IoT.

8.1.4.2. Escaneo de los protocolos abiertos de la cámara IP.

Posteriormente, se procede a escanear todos los servicios y protocolos que se encuentran en funcionamiento de dicho equipo, a través del comando “#nmap -sV 192.168.1.102”:

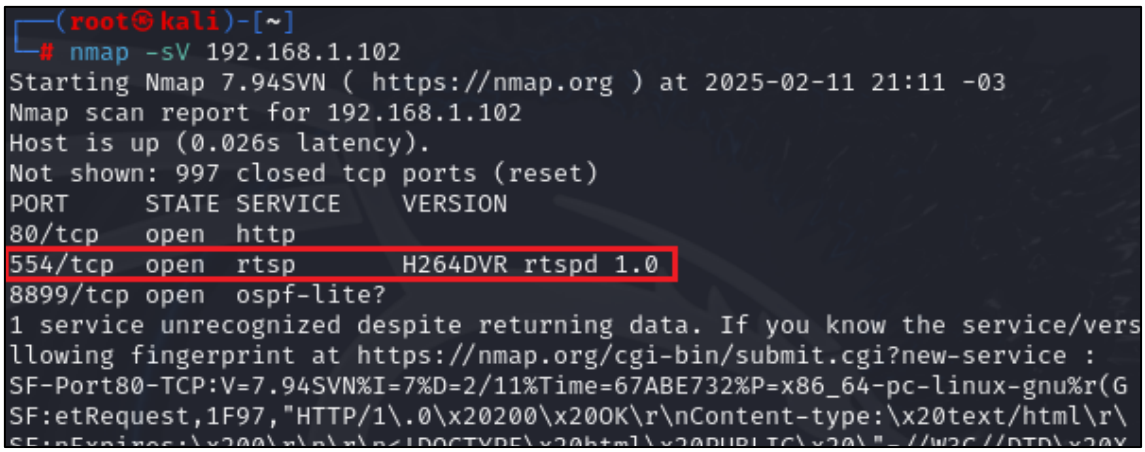

Figura 38. Protocolos activos en el *host*.

Se puede observar activo el protocolo “RTSP” (Real Time Streaming Protocol), el cual permite la transmisión de audio y video en tiempo real, de lo que podemos inferir que efectivamente se trata de la cámara IP.

8.1.4.3. Ataque de fuerza bruta al protocolo RTSP

Se inicia la recolección de información sobre el protocolo RTSP, mediante el comando de NMAP “# nmap --script rtsp-url-brute -p 554 192.168.1.102” [24], con la intención de obtener algún dato que pueda ser útil para realizar el ataque:

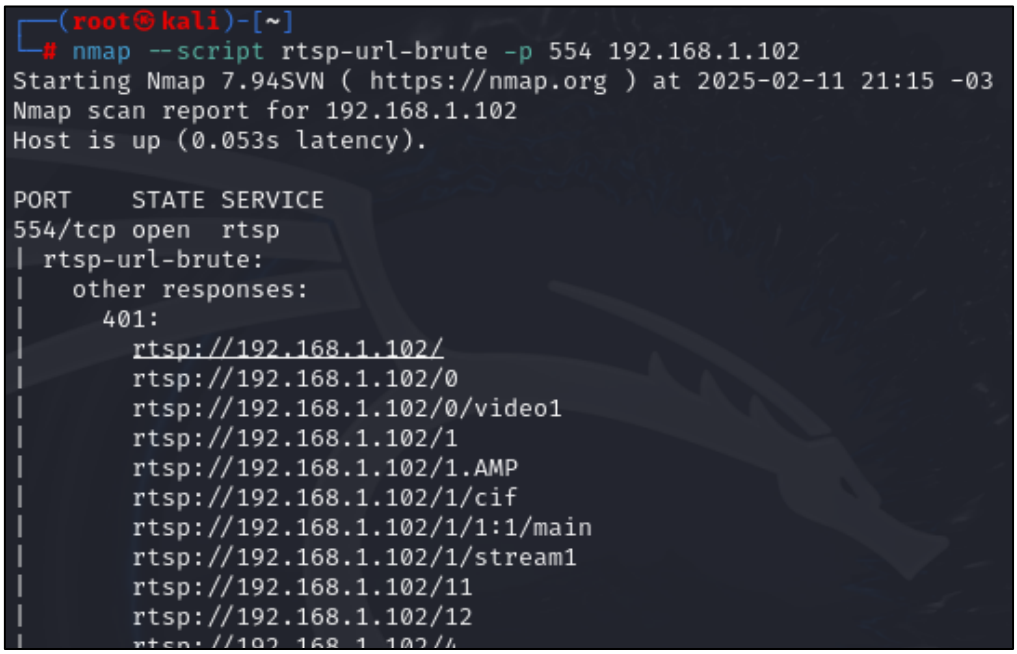

Figura 39. Parte de la salida del ataque de fuerza bruta al protocolo RTSP.

La salida del comando produce gran cantidad de información de directorios y archivos contenidos, y al final de la misma se menciona el usuario “admin” y contraseña “tlJwpbo6”:

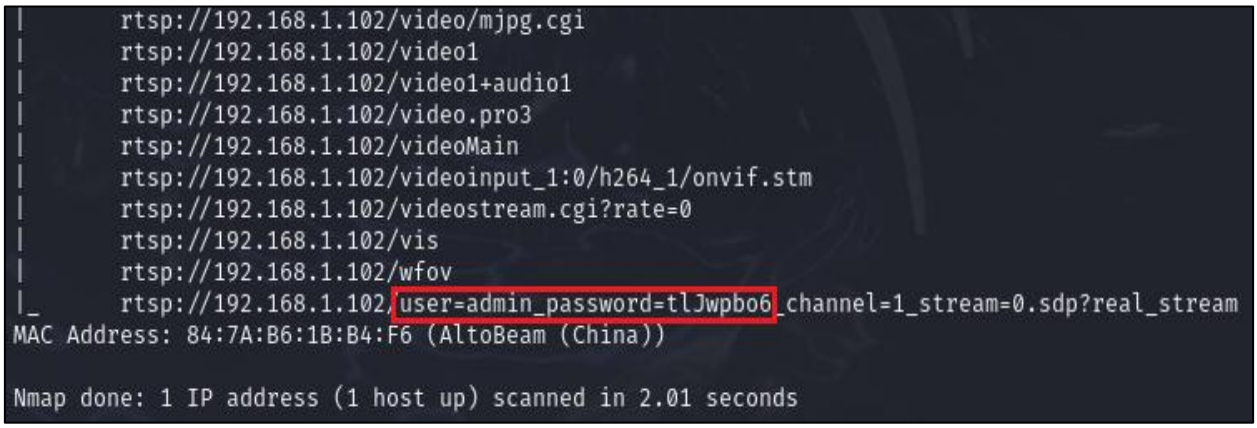

Figura 40. Detección de posible usuario y clave para acceder a la transmisión de la cámara.

Para poder corroborar estas credenciales e intentar acceder a la cámara, se utiliza el aplicativo “VLC Media Player”, donde se ingresa la dirección IP de la cámara, y a través de que protocolo y número de puerto queremos acceder:

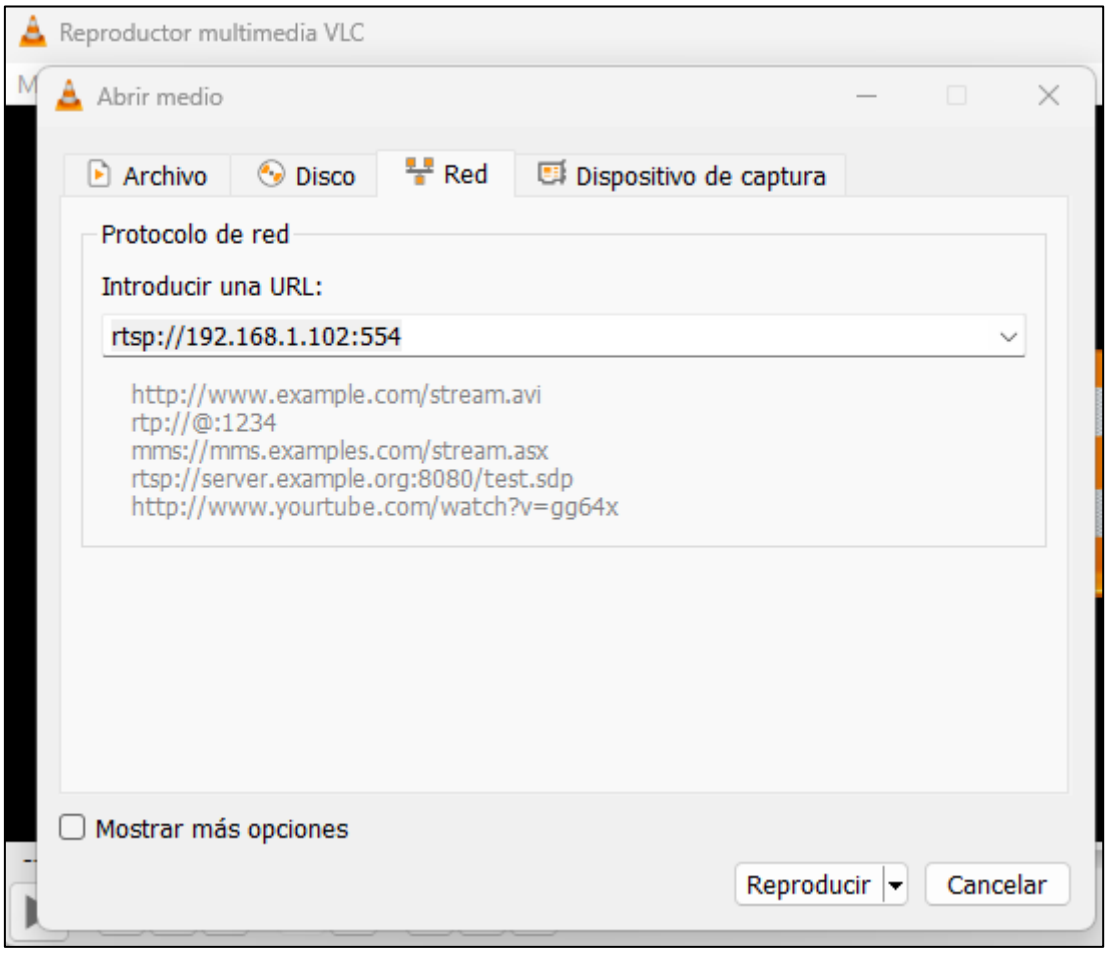

Figura 41. Se ingresa en VLC el protocolo, dir. IP y puerto para conectarse a la cámara.

Se ingresa como usuario “admin” y contraseña “tlJwpbo6”:

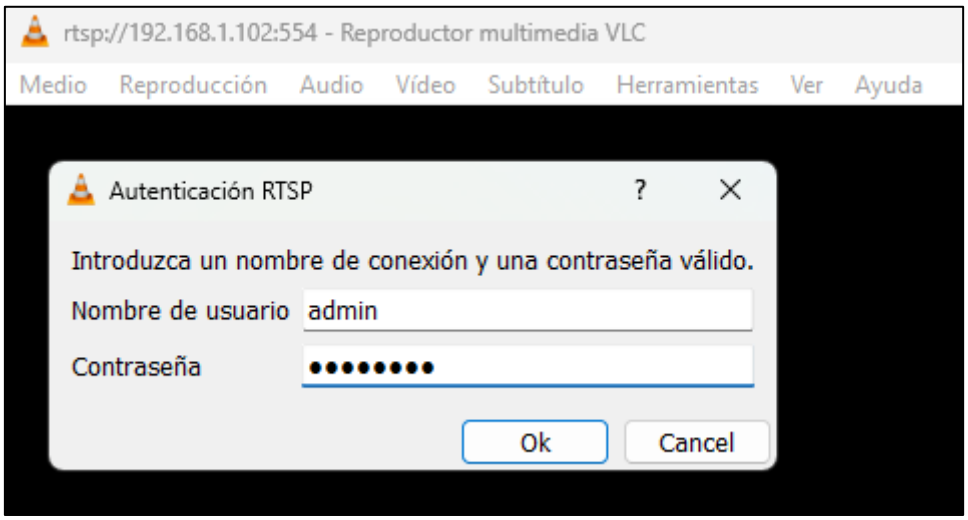

Figura 42. Primer intento sin éxito de login.

Con tal contraseña no se pudo lograr acceder, por lo que luego se intentó a través de la técnica de "*password-guessing*", ingresar con la clave "admin" que suele encontrarse configurada por defecto en esa clase de dispositivos IoT:

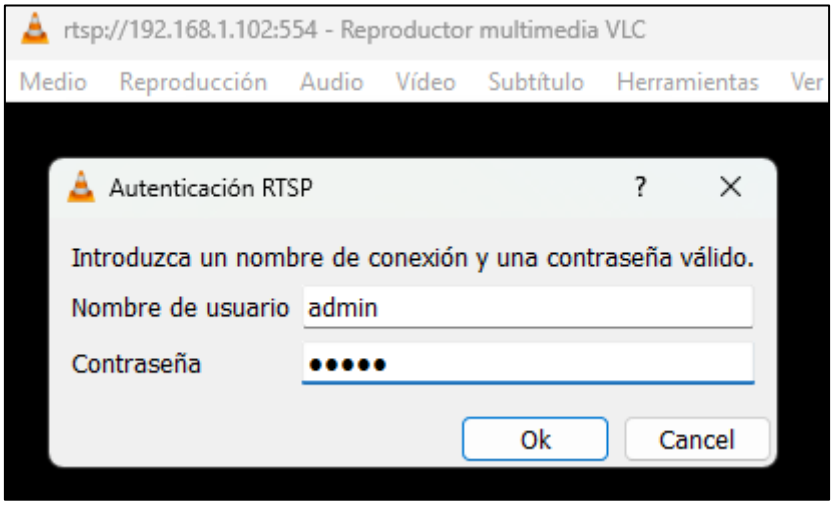

Figura 43. Segundo intento sin éxito de login.

De igual manera, la clave "admin" no era la correcta. Por tal razón, se procedió a hacer uso de la herramienta Hydra, bajo el entorno de Kali Linux, para realizar un ataque de fuerza bruta mediante la utilización del diccionario de contraseñas "rockyou".

Para ello se ejecuta el comando "# hydra -l admin -P /usr/share/wordlists/rockyou.txt 192.168.1.102 rtsp", cuyos parámetros se detallan a continuación:

- -l admin: se especifica que se quiere lanzar el ataque de fuerza bruta con el usuario "admin".
- -P /usr/share/wordlists/rockyou.txt: se indica la ruta de donde se encuentra el diccionario de contraseñas con el que se realizará el ataque.
- 192.168.1.102 rtsp: se detalla la dirección IP y protocolo por el que se hará el ataque.

```
┌──(root㉿kali)-[~]
└─# hydra -l admin -P /usr/share/wordlists/rockyou.txt 192.168.1.102 rtsp
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret ser
vice organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics a
nyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-02-11 21:26:37
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525 t
ries per task
[DATA] attacking rtsp://192.168.1.102:554/
[STATUS] 2423.00 tries/min, 2423 tries in 00:01h, 14341976 to do in 98:40h, 16 active
[STATUS] 2905.67 tries/min, 8717 tries in 00:03h, 14335682 to do in 82:14h, 16 active
[554][rtsp] host: 192.168.1.102   login: admin   password: camera123
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-02-11 21:29:42
```

Figura 44. Detección de la clave (camera123) del usuario “admin”.

Tras finalizar la ejecución, se puede observar que se encontró la contraseña “camera123” para el usuario “admin”. Para verificarlo, se ingresa nuevamente a VLC, en esta ocasión empleando las credenciales mencionadas anteriormente:

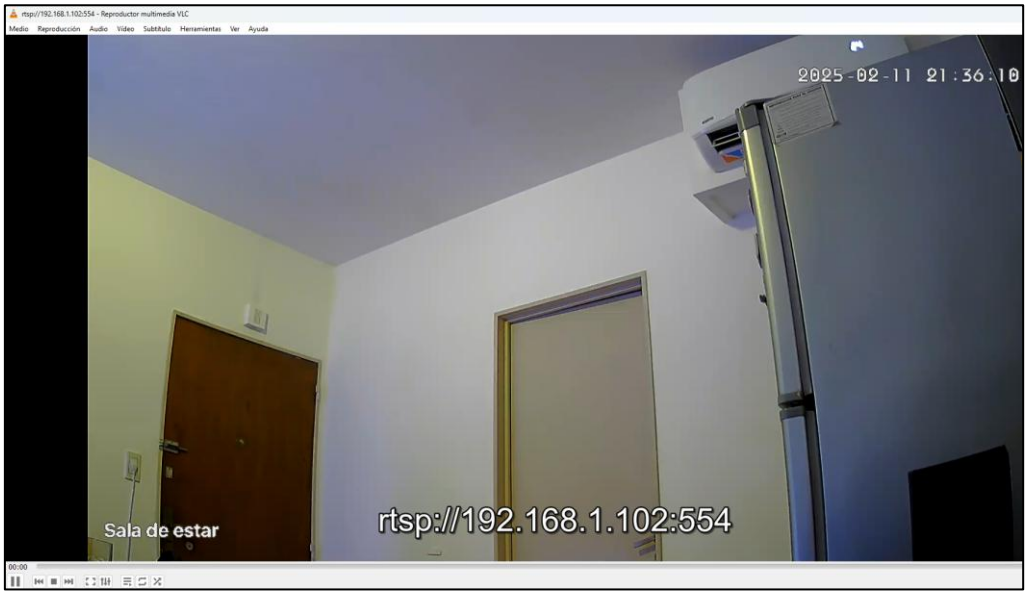

Figura 45. Acceso a la transmisión de la cámara IP.

Finalmente, se puede observar que se pudo acceder a la transmisión en vivo de la cámara IP, resultando exitoso el ataque.

8.1.5. Conclusiones del resultado del ataque realizado

En función de las actividades efectuadas anteriormente, se pudo corroborar la trascendencia que presenta la utilización de claves robustas tanto

para los dispositivos IoT, como para el router, que es la pieza central de una casa inteligente.

La utilización de claves fáciles de adivinar, o bien, las que presentan por defecto los distintos equipos configuradas por sus fabricantes, permite que puedan ser quebradas sin mucha dificultad a través de métodos de fuerza bruta, ya que en la web existen gran cantidad de diccionarios de contraseñas que presentan aquellas que son utilizadas con más frecuencia, y a la vez, aquellas que vienen por defecto en los equipos.

El empleo de contraseñas complejas que reúnan un mínimo de 10 caracteres, combinación de letras mayúsculas, minúsculas, dígitos y símbolos especiales, garantizan una gran seguridad ya que para poder quebrarlas se requiere de gran capacidad computacional, y aun así, el tiempo necesario para poder lograr ese cometido puede llegar a ser décadas e incluso siglos.

Si bien es sencillo mitigar este tipo de brechas de seguridad, ya que solo consiste en establecer contraseñas seguras, por lo general no es tenido en consideración y los equipos se mantienen con las configuraciones provistas por los fabricantes. Es por esta razón que, esta vulnerabilidad es la número uno dentro del *top* 10 de vulnerabilidades de OWASP, sobre dispositivos IoT,

9. Explotación de vulnerabilidades y generación de *Botnets*

Los desafíos de privacidad y seguridad son algunas de las preocupaciones más críticas en el ámbito de IoT. A medida que los dispositivos IoT se proliferan en hogares, empresas y espacios públicos, estos generan y transmiten grandes cantidades de datos, algunos de los cuales pueden ser altamente sensibles. La naturaleza interconectada de estos dispositivos significa que las vulnerabilidades en uno de ellos pueden comprometer potencialmente la seguridad de toda la red.

Los dispositivos IoT suelen estar diseñados para permitir un buen rendimiento con una gran facilidad de uso en lugar de la seguridad, lo que los hace susceptibles a varios ataques. Por ejemplo, muchos dispositivos vienen con contraseñas predeterminadas que rara vez son cambiadas por los usuarios, dejándolos expuestos a accesos no autorizados. Los atacantes pueden explotar estas vulnerabilidades para tomar el control de los dispositivos, robar información personal o incluso usar dispositivos comprometidos como parte de *botnets* en ciberataques más grandes. [12]

Una *botnet* es una red de *bots* (del inglés, forma abreviada de decir robot) que ejecutan el mismo software diseñado para realizar una tarea específica. El atacante a través de un centro de comando y control coordina la actividad de los *bots* individuales.

El tamaño de la *botnet* puede variar desde unos pocos dispositivos infectados hasta millones de ellos. El sujeto malintencionado que se encuentra detrás del control de una *botnet*, es llamado *bot-herder* o *botmaster*.

Cabe aclarar que, no todas las *botnets* poseen un fin malintencionado. Algunas de ellas son claves para que internet funcione correctamente. Por ejemplo, compañías como Google o Bing que corren motores de búsqueda web dependen de las *botnets* conocidas como Googlebot o Bingbot (respectivamente), para rastrear Internet, indexar su contenido y descubrir nuevos sitios.

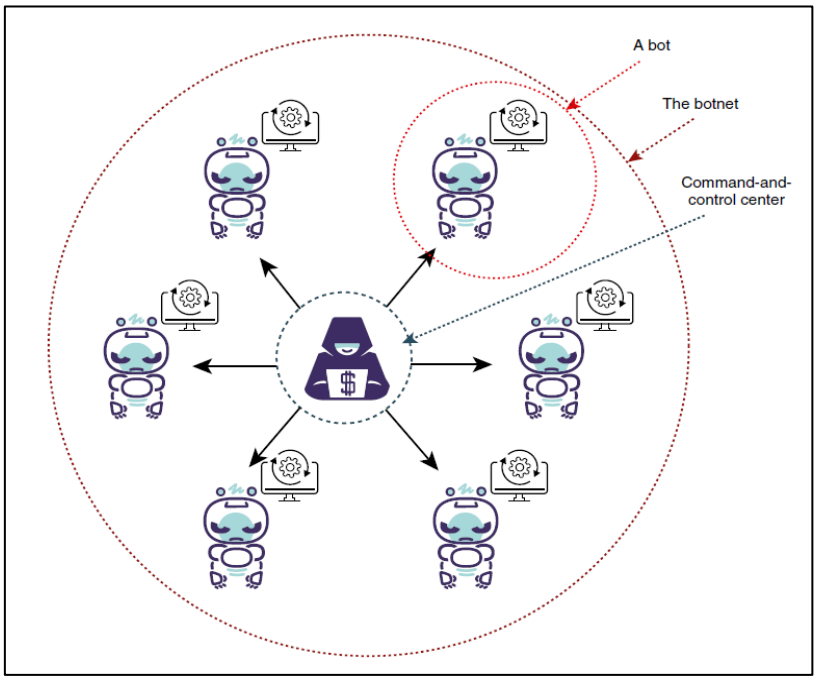

Figura 46. Arquitectura básica de una *botnet*. [25]

10. Funcionamiento y tipos de *Botnets*

Básicamente, las *botnets* dependen de dos elementos para funcionar: una gran red de dispositivos infectados para llevar a cabo el ataque, o *bots* tal como se mencionó anteriormente, y alguien que se encargue de controlar remotamente tales dispositivos.

En cuanto al funcionamiento, se pueden distinguir tres etapas en el proceso de ataque mediante *botnets*:

- Preparación y exposición: esta etapa comienza cuando los ciberdelincuentes encuentran una vulnerabilidad en un sitio web, aplicación o comportamiento humano. El objetivo es preparar al usuario para que quede expuesto sin saberlo a una infección de *malware* (*software* malicioso). Comúnmente, explotan problemas de seguridad en software o sitios web o envían el *malware* a través de correos electrónicos y otros mensajes en línea. También es común encontrarlos en troyanos[25] que se encuentran embebidos en *keygens*[26] o *cracks*[27] de programas. Por otro lado, las mismas *botnets* pueden presentar vectores de propagación similares a los *worms*[28], los cuáles permiten que se infecten y añadan nuevos nodos a la red de manera autónoma.
- Infección: el usuario se infecta con el *malware* de la *botnet* al realizar una acción que compromete su dispositivo, mediante alguno de los métodos mencionados anteriormente.
- Activación: el ciberdelincuente comienza a tomar el control de cada computadora y las organiza en una red de *bots* que puede gestionar de

[25] Programa o código malicioso que simula ser un programa legítimo para infiltrarse en un ordenador.
[26] Es un programa informático que al ejecutarse genera un código para que un determinado programa de *software* pago en su versión de prueba pueda ofrecer los contenidos completos del programa ilegalmente y sin conocimiento del desarrollador.
[27] Parche creado sin autorización del desarrollador del programa al que modifica cuya finalidad es la de modificar el comportamiento del *software* original.
[28] Programa informático malicioso que se replica para propagarse a otras computadoras.

forma remota. En lo posible, el ciberdelincuente buscará infectar y controlar miles y hasta millones de computadoras. [26]

Para poder controlar remotamente a estos nodos, el atacante utiliza un canal de comunicación llamando C&C (*Command and Control*), el cual consiste en uno o varios servidores (dependiendo el modelo), a los que les ingresa las instrucciones (*push*), las cuales luego serán descargas por los nodos infectados (*pull*). Esta administración puede realizarse principalmente adoptando dos modelos diferentes: centralizado (cliente/servidor) o descentralizado (P2P[29]).

10.1. Modelo Centralizado

La forma convencional de configurar una *botnet* es utilizar el modelo cliente-servidor, en el cual los *bots* reciben sus instrucciones y actualizaciones desde una sola ubicación, típicamente un sitio web o servidor compartido. Aunque fue efectivo en sus comienzos, una *botnet* cliente-servidor es fácil de detener simplemente cerrando la ubicación del servidor.

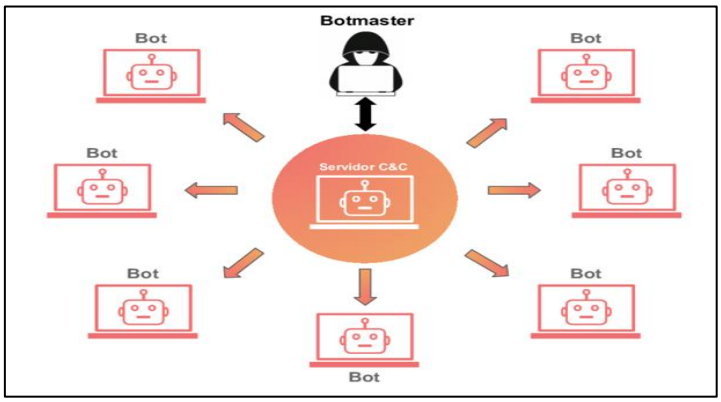

Figura 47. Modelo centralizado de *botnet*. [27]

10.2. Modelo Descentralizado

En esta clase de redes P2P no existe un servidor central, sino que todas las solicitudes son gestionadas por los nodos dentro de la red. En arquitecturas puramente descentralizadas, cada nodo se comporta de la misma manera,

[29] Red *peer-to-peer*, o red entre pares, es una red de ordenadores que funcionan sin clientes ni servidores fijos, sino que los mismos adoptan ambas funciones.

actuando como servidores cuando procesan una consulta de búsqueda de archivos y como clientes cuando solicitan un archivo. No obstante, existen arquitecturas parcialmente descentralizadas, donde los nodos con mejores recursos computacionales y ancho de banda de red tienen la oportunidad de ser promovidos y convertidos en "supernodos", desempeñando un papel más importante en la red.

Los nodos normales (o nodos hoja) se conectan a al menos un supernodo y solo pueden enviar consultas a los mismos. Un supernodo mantiene una tabla de valores hash de los archivos que están disponibles en sus nodos hoja locales, y solo ellos pueden reenviar los mensajes de los nodos hoja, cumpliendo la función de servidores locales.

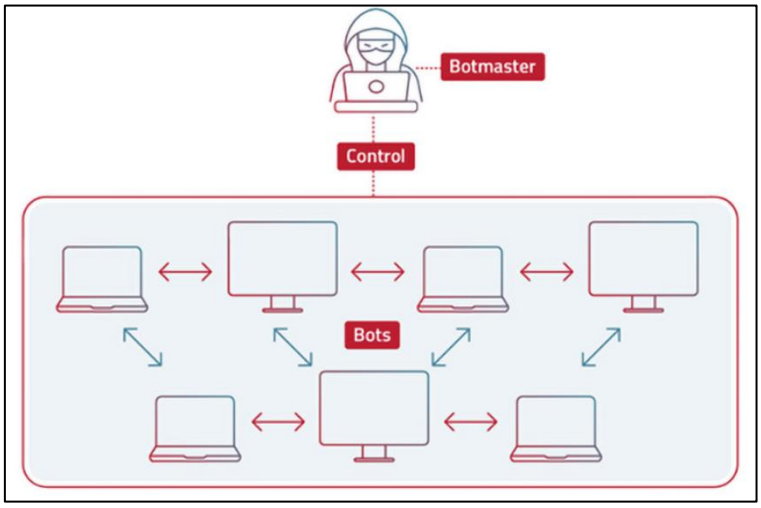

Figura 48. Modelo descentralizado (P2P) de *botnet*. [28]

Para dar órdenes a los *bots*, el *botmaster* coloca los nuevos comandos en los supernodos a través de un *push*, y estos últimos se encargan de diseminarlo a sus supernodos vecinos. Finalmente, los nodos hoja, obtienen los comandos de los supernodos a los que están enlazados mediante un *pull*, para luego ejecutarlos. [29]

11. Principales tipos de ataque de las *botnets*

En la actualidad existen diferentes tipos de ataques realizados con *botnets*. En la presente sección se detallarán los principales haciendo especial énfasis en DDoS[30], siendo este uno de los más populares y que mayores pérdidas económicas genera.

11.1. Ataques DDoS:

11.1.1. Descripción del ataque DDoS.

Los recursos de red tienen un límite finito de solicitudes que pueden atender al mismo tiempo. Además del límite de capacidad del servidor, el canal que conecta el servidor a Internet tiene un ancho de banda limitado. Cuando la cantidad de solicitudes sobrepasa los límites de capacidad de cualquiera de los componentes de la infraestructura, el nivel de servicio probablemente se vea afectado de alguna de las siguientes maneras: lentitud en la respuesta de las solicitudes de los cliente, o bien, no se trata directamente la solicitud del cliente. [30]

Para lanzar un ataque DDoS, los atacantes utilizan *malware* para crear una red de *bots*, a su vez, cada dispositivo infectado es capaz de propagar el *malware* a otros dispositivos para amplificar las dimensiones de un ataque. Una vez que se ha creado la *botnet*, se envían instrucciones remotas a la misma, indicándole que se envíen solicitudes y tráfico a un servidor, sitio web, aplicación web, API o recurso de red objetivo. Esto crea una cantidad excesiva de tráfico que provoca una denegación de servicio, impidiendo que el tráfico legítimo llegue a su destino previsto.

Un ejemplo concreto de un ataque DDoS puede ser el de impedir a un usuario que acceda a un sitio web, compre un producto o servicio, vea un vídeo o interactúe en redes sociales. Incluso en el ámbito organizacional, no poder disponer de los sistemas funcionando correctamente, puede provocar la

[30] *Distributed Denial of Service*: Denegación de servicio distribuida.

disrupción de la continuidad operativa con las consecuencias económicas y reputacionales que ello conlleva.

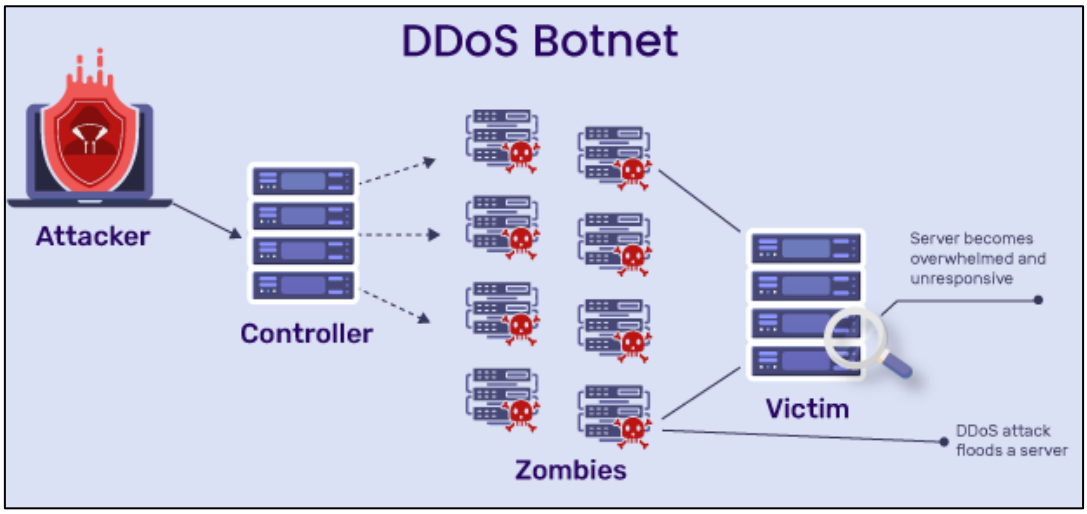

Figura 49. Ataque DDoS. [31]

11.1.2. Finalidad del ataque DDoS

Los ataques DDoS se pueden lanzar por diferentes motivos:

- Hacktivismo: los atacantes pueden dirigir un ataque DDoS contra empresas o sitios web con los que tienen desacuerdos filosóficos o ideológicos.
- Guerra cibernética: los gobiernos pueden utilizar ciberamenazas como los ataques DDoS para perjudicar la infraestructura crítica de un estado enemigo.
- Extorsión: los atacantes suelen utilizar amenazas DDoS para extorsionar a las empresas a cambio de dinero.
- Entretenimiento: muchos ataques son lanzados por ciberdelincuentes que simplemente quieren entretenerse causando conflictos, y a la vez, para hacerse conocido en el ambiente bajo diferentes pseudónimos.
- Competencia empresarial: una empresa puede lanzar un ataque DDoS a otra para obtener una ventaja competitiva. [32]

11.1.3. Tipos de ataques DDoS

Cabe mencionar que principalmente existen cuatro diferentes tipos de ataque DDoS:

1) Ataques a la capa de aplicación: los ataques DDoS a la capa de aplicación (capa 7 del modelo OSI[31]) se dirigen a vulnerabilidades específicas de las aplicaciones web con el fin de evitar que la aplicación funcione según lo previsto. Estos ataques DDoS suelen tener como objetivo los protocolos de comunicación implicados en el intercambio de datos entre dos aplicaciones a través de Internet. Aunque son difíciles de prevenir y mitigar, se encuentran entre los ataques DDoS más fáciles de lanzar. Los ataques más comunes de esta clase son:
    - Inundaciones HTTP: explotan el protocolo HTTP de Internet que se utiliza para cargar páginas web o enviar contenido a través de Internet. Las inundaciones HTTP provocan que un servidor, un sitio web o una aplicación web se ralenticen o se bloqueen saturándolos con un gran número de solicitudes HTTP GET o POST.
    - Slowloris: un ataque DDoS de Slowloris está diseñado para saturar un servidor web al abrir y mantener muchas conexiones HTTP simultáneas con un servidor de destino. Slowloris agota los recursos del servidor con solicitudes que parecen más lentas de lo habitual, pero que de otro modo parecen ser tráfico estándar. Los atacantes aprovechan una función exclusiva del protocolo HTTP: la capacidad de los clientes de dividir las solicitudes GET o POST en varios paquetes. Un ataque Slowloris compromete el servidor web objetivo abriendo varias conexiones y manteniéndolas abiertas todo el tiempo posible. Esto se consigue enviando solicitudes HTTP parciales que nunca se completan.

[31] *Open System Interconnection Model,* Modelo de Interconexión de Sistemas Abiertos.

2) Ataques de protocolo: los ataques de protocolo explotan las debilidades y las vulnerabilidades de los protocolos de comunicaciones de Internet de las capas 3 y 4 del modelo de OSI. Estos ataques intentan consumir y agotar la capacidad informática de diversos recursos de infraestructura de red, como servidores o firewalls, a través de solicitudes de conexión malintencionadas que vulneran el protocolo de control de transmisión (TCP) o los protocolos del Protocolo de control de mensajes de Internet (ICMP). Los dos casos más comunes son los siguientes:

   - Inundación SYN: una de las principales maneras en las que las personas se conectan a las aplicaciones de Internet es a través del TCP. Esta conexión requiere un protocolo de negociación en tres pasos desde un servicio TCP, como un servidor web. Los pasos incluyen enviar un paquete SYN (sincronización) desde el lugar en que el usuario se conecta al servidor, después devolver un paquete SYN-ACK (confirmación de sincronización), y finalmente recibir un mensaje ACK (confirmación) final como respuesta para completar el protocolo de negociación de TCP. Durante un ataque por inundación SYN, un cliente malintencionado envía un gran volumen de paquetes SYN (primer paso de una negociación normal), pero nunca llega a enviar la confirmación para completar la negociación. Esto deja al servidor esperando una respuesta a estas conexiones TCP semiabiertas. Finalmente, el servidor se queda sin capacidad para aceptar nuevas conexiones para los servicios que realizan un seguimiento de los estados de conexión.

   - Ataque DDoS Smurf: una gran cantidad de paquetes de protocolo de ICMP con la IP de origen falsificada de un destino se transmiten a una red informática mediante una dirección IP de difusión. De forma predeterminada, la mayoría de los dispositivos de una red responderán enviando una respuesta a la dirección IP de origen.

Según el número de equipos de la red, puede que el ordenador de la víctima se ralentice en extremo debido a la inundación de tráfico.

3) Ataques por amplificación/reflexión de DNS[32]: el atacante aprovecha los servidores DNS vulnerables para enviar grandes volúmenes de tráfico legítimo al servidor objetivo. Uno de los principales beneficios de este método de ataque, es que el atacante se esconde detrás de una dirección IP legítima.

4) Ataques volumétricos: los ataques DDoS basados en el volumen se dirigen a las capas 3 y 4 de OSI, saturando al objetivo con una inundación de tráfico procedente de diferentes orígenes y, finalmente, consumiendo todo el ancho de banda disponible del objetivo, lo que hace que este se ralentice o se bloquee. Los ataques volumétricos suelen utilizarse para desviar la atención de otros tipos de ataques DDoS o ciberataques más peligrosos.

    - Inundaciones UDP: se suelen elegir para ataques DDoS que consumen un mayor ancho de banda. Los atacantes intentan sobrecargar los puertos en el *host* objetivo con paquetes IP que contienen el protocolo UDP. A continuación, el *host* objetivo busca aplicaciones que estén asociadas con los paquetes UDP y, cuando no las encuentra, envía el mensaje "destino inalcanzable" al remitente. Las direcciones IP a menudo se falsifican para que el atacante permanezca anónimo y, una vez que el *host* objetivo se inunda con el tráfico del ataque, el sistema deja de responder y de estar disponible para usuarios legítimos.
    - Inundaciones ICMP: se utiliza principalmente para los mensajes de error y, por lo general, no intercambia datos entre sistemas. Una inundación ICMP es un método de ataque DDoS con una infraestructura de capa 3 que utiliza mensajes ICMP para sobrecargar el ancho de banda de la red objetivo. [32]

[32] Domain Name System: Sistema de Nombres de Dominio

Cabe destacar que actualmente, en el mercado se encuentran distintas soluciones que permiten prevenir con gran eficacia este tipo de ataques. Algunas de las más populares, por ejemplo, son Prolexic (Akamai) o FortiDDoS (Fortinet). Su funcionamiento radica en redirigir el tráfico entrante a la solución contratada, allí se lo inspecciona en búsqueda de posibles paquetes provenientes de diferentes *botnets*, separando dicho tráfico del legítimo. Realizada esta tarea, se reenvía el tráfico seguro a la red de la empresa y/o cliente que contrató la solución. [33]

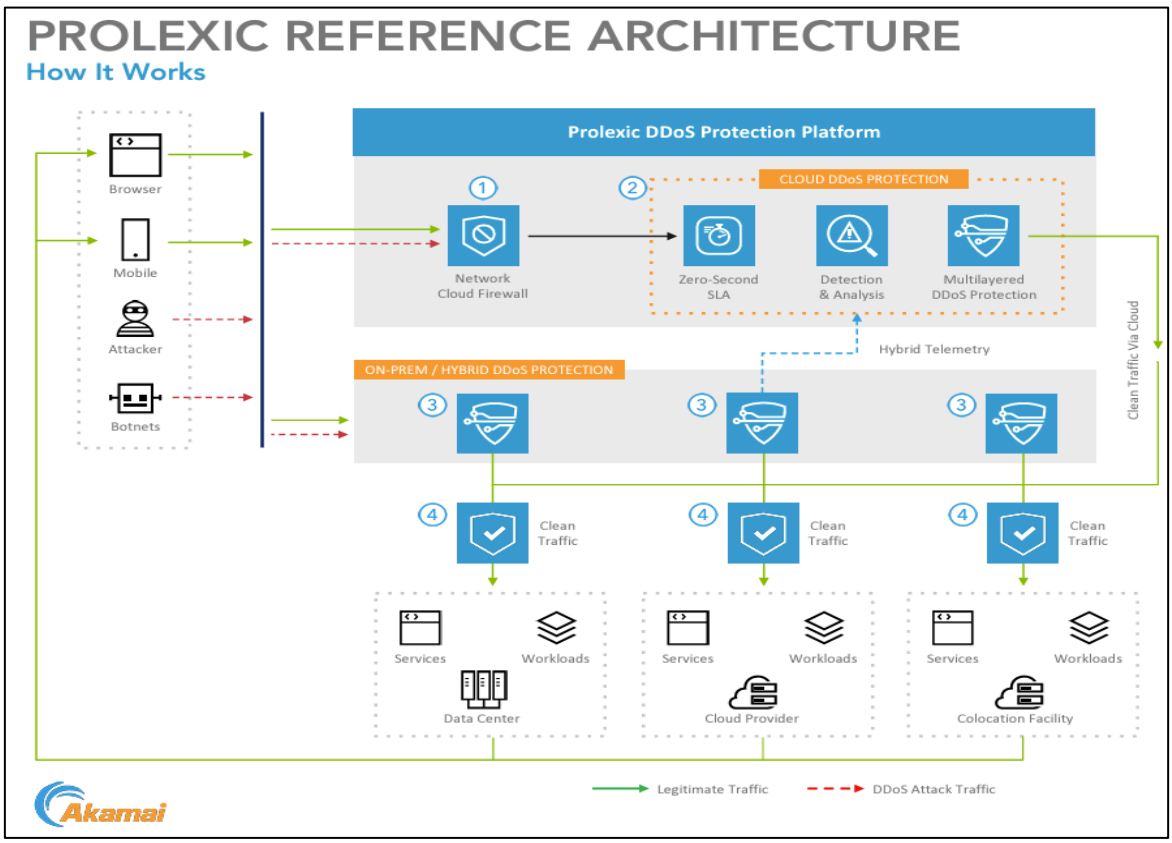

Figura 50. Prevención de DDoS con Prolexic (Akamai). [34]

11.2. Spam y Phishing

Los nodos de una *botnet* también pueden ser utilizados para lanzar campañas masivas de spam y phishing por correo electrónico. Estos correos electrónicos pueden contener enlaces o archivos adjuntos maliciosos que instalan el software de la *botnet*, propagándose aún más y ampliando su alcance. También pueden engañar a los usuarios para que revelen información personal

o credenciales de inicio de sesión. Las *botnets* también pueden difundir mensajes de spam a través de otros métodos, como publicaciones en foros de Internet o comentarios en blogs.

11.3. Minería de criptomonedas

Por último, algunos atacantes utilizan *botnets* para obtener ganancias financieras, por ejemplo a través de campañas de minería de criptomonedas. Las criptomonedas como Bitcoin requieren una gran capacidad computacional para crear nuevas unidades, un proceso conocido como minería. Los atacantes pueden usar una *botnet* para aprovechar la potencia de procesamiento de las máquinas bajo su control, generando nuevas monedas para sí mismos, mientras que los propietarios de las máquinas pagan el costo en forma de mayor consumo de electricidad. [35]

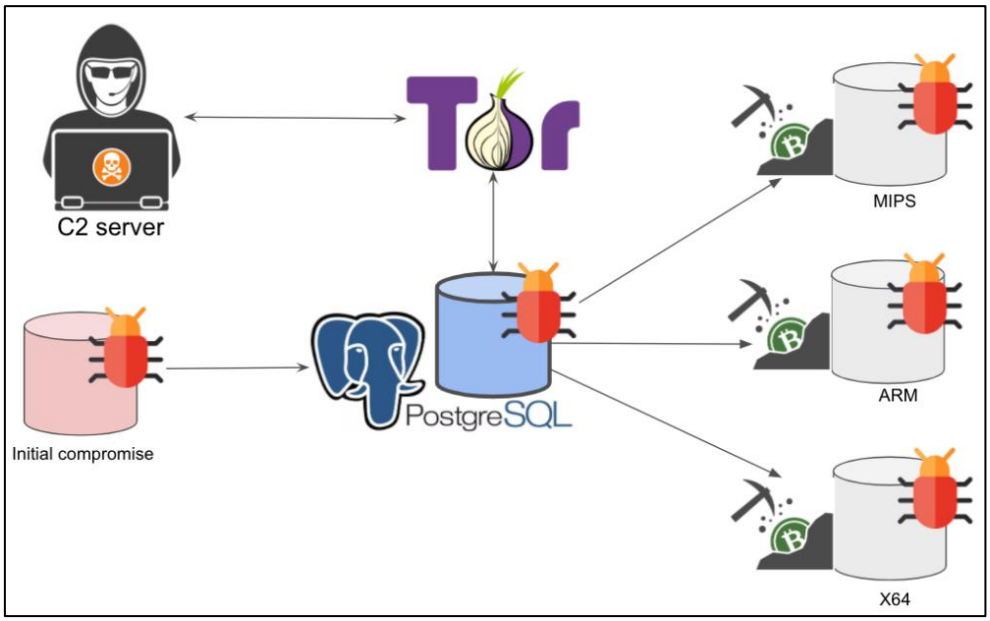

Figura 51. Ataque de *Cryptocurrency mining* con *botnet*. [36]

11.4. Fraude publicitario

Los atacantes pueden utilizar las máquinas de una *botnet* para simular de manera maliciosa la actividad de usuarios reales. Por ejemplo, una *botnet* puede perpetuar el denominado "fraude de clicks", en el que las máquinas de la *botnet* hacen clic repetidamente en los enlaces o botones de una campaña publicitaria.

Dado que los anunciantes pagan por cada usuario que hace clic en un anuncio (modelo de pago conocido como *pay-per-click* o PPC), este tipo de ataque puede ser utilizado para dañar significativamente los presupuestos publicitarios de los competidores. Las *botnets* también pueden usarse para aumentar artificialmente la popularidad de cierto contenido en un sitio web, otorgándole vistas, *likes* o votos positivos.

12. Ataques significativos de botnets a lo largo de la historia

La utilización de *botnets* se remonta hacia el año 2.000, cuando surgieron las primeras redes de este tipo. A continuación, se presentará el detalle de los hitos más destacados desde su surgimiento, ya sea por su trascendencia en términos de la magnitud de daños causados o cantidad de *bots* reclutados, o bien, por disponer de nuevas características que lo hacen superior respecto a sus predecesores.

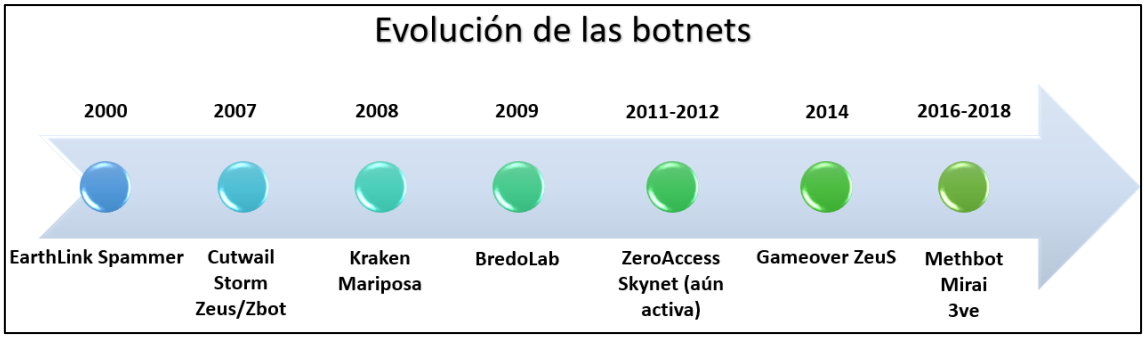

Figura 52. Evolución de las *botnets*.

- EarthLink Spammer: fue uno de los primeros ataques de *botnets*. Los atacantes se involucraron en el *phishing*, un elemento crucial de los ataques de *botnets*, y enviaron correos electrónicos que pretendían ser de sitios web conocidos. Estos ataques de phishing engañaron a los usuarios para que entregaran información sensible, incluidos nombres de usuario, contraseñas y números de tarjetas de crédito.

- Cutwail: involucró a sistemas Windows, utilizándolos para enviar el troyano Pushdo, que convertía a las computadoras en *spambots*[33]. En su pico máximo, llegó a enviar más de 74 mil millones de correos electrónicos al día.

- Storm: se originó mediante la diseminación del gusano Storm, un *malware* de tipo troyano distribuido por medio de correos electrónicos infectados. Se estimó que en septiembre de 2007 esta *botnet* estaría constituida por millones de computadoras. Fue identificada por primera

[33] Programa informático diseñado para ayudar en el envío de spam.

vez en enero de 2007, en un punto abarcó el 8% de todo el *malware* de las computadoras bajo el sistema operativo Microsoft Windows.

- Zeus/Zbot: obtuvo acceso a la información bancaria de los usuarios. En un momento, Zeus fue responsable del 90% de todo el fraude relacionado con bancos en línea y causó pérdidas a sus víctimas por más de 120 millones de dólares. Al igual que muchos ataques de *botnets*, Zeus ha evolucionado desde su lanzamiento inicial y sigue activo hoy en día.

- Kraken: obtuvo acceso a cientos de miles de computadoras, incluyendo al menos 50 de las 500 empresas más grandes del mundo. En su punto máximo, cada *bot* enviaba hasta 500.000 correos electrónicos de spam por día, lo que lo convirtió en el más grande del mundo en ese momento.

- Mariposa: estuvo involucrada en el robo de información personal y en ataques DDoS. Aunque finalmente capturó más de 12 millones de direcciones IP e infectó a más de un millón de computadoras, pudo ser desactivada en diciembre de 2009.

- BredoLab: poseía más de 30 millones de equipos infectados al mes de julio del 2009. Dichos equipos habían sido infectados con el troyano “Win32/TrojanDownloader.Bredolab” por lo cual reportaban a alguno de los 143 servidores maliciosos que conformaban la *botnet* en cuestión, recibiendo instrucciones directamente de los usuarios maliciosos por medio de estos. Una particularidad de la *botnet* Bredolab es que era utilizada para brindar servicios de "*pay per install*" (pago por instalación), que significa que los *botmasters* alquilaban la *botnet* a desarrolladores de *malware* que tuvieran problemas diseminando sus propias amenazas para luego cobrarles por cada instalación realizada.

- ZeroAccess: afectaba a los sistemas operativos Microsoft Windows, teniendo capacidades de rootkit[34]. Se estima que la *botnet* contó con más de dos millones de nodos, los cuáles eran principalmente utilizados para minar criptomonedas y fraude en campañas online.

- Skynet: es una variante de Zeus empleada para ataques DDoS y minería de bitcoin, que cuenta con servidores C&C centralizados ocultos detrás de un dominio .onion dentro de la red TOR. Todo el tráfico se enruta hacia esta red mediante un proxy SOCKS instalado localmente en cada una de las víctimas. Actualmente se encuentra activa, y fue utilizada recientemente (a mediados de 2023) para realizar ataques a Microsoft, X y OpenAI. [37]

- Gameover ZeuS: adicionalmente al componente original de Zeus, pensado para robar cuentas financieras, GameOver Zeus es una variante avanzada con un componente de *ransomware*. Por lo tanto, no solo es un *malware* que roba credenciales bancarias mediante una función de *keylogger* y códigos de inyección web, sino también, que se lo utilizó para distribuir el *ransomware* CryptoLocker. Esta *botnet* funcionaba de manera descentralizada con protocolos P2P.

- Methbot: se llegaron a reproducir entre 200 y 400 millones de visualizaciones de anuncios de video por día utilizando más de 2.000 servidores diferentes con más de 650.000 direcciones IP residenciales que simulaban ser usuarios finales reales. Se estima que el esquema generaba entre 3 y 5 millones de dólares al día en ingresos por publicidad para sus operadores. [38]

- Mirai: aprovecha los problemas de seguridad de los dispositivos IoT, y tiene el potencial de convertir el poder colectivo de millones de dispositivos IoT en *botnets*, y lanzar ataques. Los creadores de dicha *botnet* fueron apresados, acusados de alquilar su *botnet* para ataques DDoS y fraudes de clics. [39]

[34] *Software* malicioso diseñado para darle a un atacante la capacidad de introducirse en un dispositivo con permiso administrador.

- 3ve: utilizaba los paquetes de *malware* Boaxxe y Kovter para infectar las computadoras. Se difundieron a través de correos electrónicos y descargas de *software* ilegal y, una vez que infectaban los equipos, los *bots* generaban clicks falsos en anuncios en línea. Los *bots* podían imitar el comportamiento de un usuario humano (por ejemplo, movimiento irregular del cursor) para evadir la detección. En su momento más importante, la *botnet* controlaba más de un millón de direcciones IP residenciales y corporativas, principalmente en Europa y América del Norte. [40]

13. Métodos de detección de *bots*

Respecto a los métodos de detección de *bots*, en primer lugar, surgieron los denominados interactivos con desafíos CAPTCHA[35], y luego, los métodos de detección transparentes. Estos últimos difieren de los primeros, no requiriendo interacción del usuario y se enfocan en identificar el sistema que realiza la solicitud y detectar anomalías generalmente asociadas con tráfico malicioso. [41]

13.1. Métodos interactivos

Los métodos de detección interactivos, más conocidos como CAPTCHAs, están diseñados para presentar un acertijo simple con el que el usuario debe interactuar y resolver para demostrar que es humano. A lo largo de los años se han inventado diferentes tipos de acertijos, los cuáles se describirán a continuación:

- *Word Puzzles* (o acertijos de palabras): Fueron inventados por primera vez en 1997 y progresivamente adoptados por empresas como PayPal a principios de la década de 2000 para protegerse contra el fraude. Uno de los despliegues más extendidos fue reCAPTCHA, lanzado en 2007 y adquirido por Google en 2009. reCAPTCHA ayudó a proteger sitios web contra la automatización y, al mismo tiempo, ayudó a digitalizar libros del dominio público. A los usuarios se les presentaba una serie de caracteres alfabéticos que debían escribir en un campo para acceder a los recursos protegidos.

  El sistema se basaba en la tecnología de reconocimiento óptico de caracteres (OCR) para validar la respuesta del usuario. Sin embargo, a medida que la tecnología OCR mejoraba, los estafadores también la utilizaban para automatizar la resolución de acertijos.

  Para compensar esto, las soluciones CAPTCHA fueron mejoradas añadiendo varios tipos de distorsiones, esquemas de colores y fondos

[35] *Completely Automated Public Turing test to tell Computers and Humans Apart*: test de Turing público y automático para distinguir a los ordenadores de los humanos.

ruidosos a la cadena de caracteres para dificultar el reconocimiento por parte de las máquinas. El problema es que esto también hizo que fuera significativamente más difícil y frustrante para los humanos reconocer los caracteres.

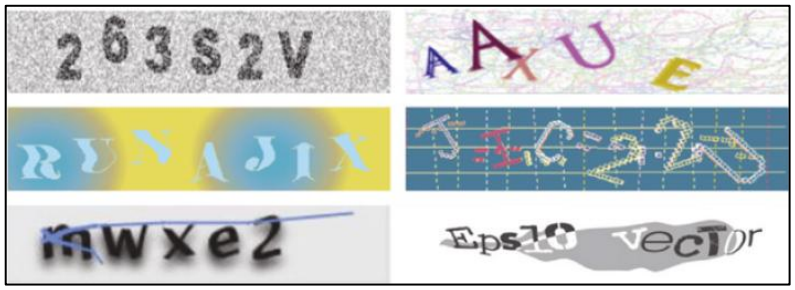

Figura 53. Ejemplos de CAPTCHAs de acertijos de palabras. [41]

- *Image Puzzles* (acertijo de imágenes)*:* Fueron introducidos para compensar la evolución de la tecnología OCR y hacer que los CAPTCHAs fueran más amigables para los humanos. reCAPTCHA introdujo la versión 2 con acertijos de imágenes en 2012. A los usuarios se les presentaba una descripción y un conjunto de imágenes y estos debían seleccionar las que correspondían a la descripción para completar el acertijo.

  La introducción de esta nueva variante tomó por sorpresa a los operadores de *bots*, ya que sus *botnets* equipadas con OCR no podían manejar los acertijos de imágenes. Sin embargo, la visión por computadora mejoró drásticamente con modelos preentrenados como ResNet, que reconocen diversos objetos, puntos de referencia o animales de forma inmediata. Esto hizo que el reconocimiento de imágenes fuera más accesible para todos, incluyendo las nuevas *botnets* herramientas para completar correctamente estas pruebas.

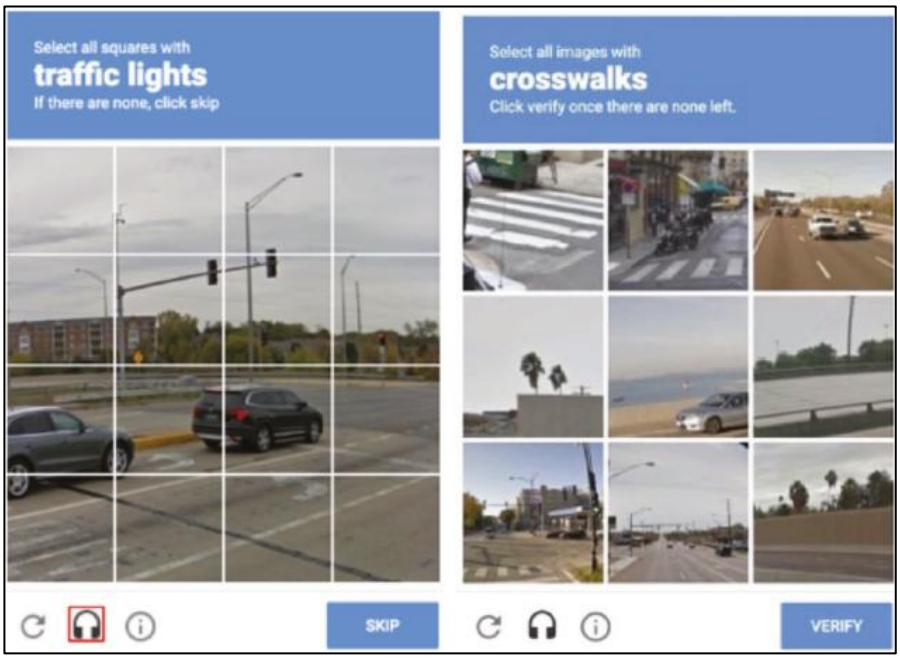

Figura 54. Ejemplos de CAPTCHAs de acertijos de imágenes. [41]

- Mini-juegos*:* Al mismo tiempo en que aparecieron los acertijos de imágenes, empresas como "*Are You a Human*" introdujeron una alternativa diferente a los acertijos CAPTCHA con mini-juegos. Estos pueden ser una interacción divertida para los usuarios, pero también pueden generar molestias si estos tienen que pasar por el proceso varias veces durante una sesión mientras interactúan con un sitio.

Figura 55. Ejemplos de CAPTCHAs con mini juegos. [41]

- Desafíos de comportamiento: La novedad de los mini-juegos y los acertijos de imágenes perdió fuerza con el tiempo, ya que últimamente no estaban garantizando alta seguridad por los avances en las *botnets*, y a la vez, eran molestos para los usuarios. Por otro lado, aunque eran

populares en sitios de juegos, no eran ideales para sitios de banca financiera o comercio electrónico.

Los propietarios de sitios web buscaban soluciones que requirieran la menor interacción posible para limitar la fricción, surgiendo en el 2014 la solución de Google llamada “reCAPTCHA No CAPTCHA”, que solo requería que el usuario marcara una casilla. Durante esta simple interacción, el *software* recopila información de telemetría del lado del cliente para evaluar el riesgo según la huella digital del dispositivo y cómo el usuario interactúa con el desafío. Se tienen en cuenta la huella digital, el movimiento del *mouse*, los eventos táctiles o las pulsaciones de teclas, y en caso de detectar anomalías, el usuario puede ser desafiado con un acertijo de imágenes.

Figura 56. Ejemplos de reCAPTCHA No CAPTCHA. [41]

13.2. Métodos transparentes

Los métodos de detección transparentes hacen uso de diversos datos recopilados del cliente a través de JavaScript o un SDK (Kit de Desarrollo de *Software*) o aplicación móvil, así como información sobre cómo el cliente se comunica con el servidor, principalmente a través de los protocolos HTTP, TCP, IP y TLS. Estos métodos de detección no requieren interacción del usuario.

Dentro de los métodos transparentes se pueden mencionar los siguientes, de acuerdo con el orden cronológico en el que fueron surgiendo:

- *Web Application Firewall* (WAF)*:* Alrededor de 2010, los ataques provenían principalmente de pequeñas *botnets*, que consistían en unos

pocos nodos, y los *scripts*[36] que ejecutaban eran simples. La tasa de solicitudes de la *botnet* solía ser muy alta, por lo que detectarlas y bloquearlas era relativamente sencillo. Lo único que se necesitaba en este escenario era un firewall de aplicaciones web (WAF) con capacidades de bloqueo de IP y limitación de tasa, así como la capacidad de crear reglas personalizadas.

Debido a que el número de nodos dentro de la *botnet* era pequeño, los profesionales de la seguridad web podían simplemente agregar direcciones IP individuales a una lista de bloqueo. Incluso podían implementar una limitación de tasa con un umbral bajo en los recursos críticos, de manera de detectar automáticamente y mitigar los clientes que enviaban solicitudes a alta velocidad.

Finalmente, debido a que los *scripts* eran tan simples, la firma del encabezado HTTP se veía muy diferente de lo que se espera de una solicitud regular de navegadores legítimos como Chrome o Firefox. Desarrollar y agregar una regla personalizada en la política del WAF era lo suficientemente fácil como para bloquear la actividad de los *bots*.

- Gestión de *bots*: Alrededor del año 2012, a medida que los WAFs se implementaban comúnmente para proteger sitios web, los operadores de *bots* evolucionaron sus *botnets*. Estas se hicieron más grandes y sus *scripts* más avanzados. En lugar de unos pocos nodos, una *botnet* consistía en varios cientos de nodos. De esta manera, el ciberdelincuente podía distribuir el tráfico, reducir la velocidad de solicitudes de cada nodo, derrotar cualquier intento de control de tasa y hacer ineficaz la estrategia de bloqueo de IP.

  La firma del encabezado también se mezclaba más con las solicitudes de navegadores reales y se volvía más compleja. Los operadores de *bots* diseñaron sus *botnets* para aleatorizar características específicas del dispositivo, como el valor del encabezado HTTP User-Agent.

[36] Conjunto de instrucciones escritas en un lenguaje de programación.

Este cambio en la estrategia de ataque obligó a los defensores a actualizar continuamente sus reglas personalizadas de WAF para mantenerse al día con la evolución de los ataques. En este punto, se requerían productos diseñados específicamente para detectar *bots*. Si bien las soluciones de gestión de *bots* necesitaban los métodos básicos de detección de un WAF, también nuevos técnicas diseñadas específicamente para detectar *bots*, incluyendo la capacidad de calcular la reputación de la dirección IP, evaluar el soporte para JavaScript, recopilar una huella digital y detectar anomalías típicas del tráfico de *bots*.

La empresa Distil Networks fue pionera en este campo e introdujo su producto de gestión de *bots* en 2011. Tras el fuerte interés de empresas de comercio electrónico, viajes y hostelería, otros proveedores de seguridad web como Akamai Technologies y Cloudflare siguieron su ejemplo.

- Gestión avanzada de *bots*: En el año 2016, las soluciones de gestión de *bots* se convirtieron en el producto preferido para proteger sitios web contra *bots*. Esto obligó a los operadores de *bots* a mejorar sus *scripts* para derrotar los nuevos métodos de detección. Para superar la reputación de IP, aumentaron el tamaño de sus *botnets* a miles de nodos, lo que hizo que la reputación de IP fuera mucho menos efectiva.

  Los operadores de *bots* también encontraron formas de imitar la ejecución del JavaScript del administrador de *bots* al obtener huellas digitales válidas de sistemas legítimos y reproducirlas desde su *botnet*. Los operadores de *bots* más avanzados entendieron que las huellas digitales se utilizaban para identificar a los usuarios de forma única. Para derrotar estas técnicas, los atacantes aleatorizaron los puntos de datos de la huella digital para hacer que cada solicitud pareciera provenir de un usuario diferente.

  Adicionalmente a las herramientas que existían al momento, se requerían soluciones de gestión de *bots* más avanzadas, que incluyeran otros métodos adicionales, como la recolección de datos más avanzada

del lado del cliente, la tecnología de validación de huellas digitales, así como técnicas avanzadas como la prueba de trabajo para forzar la ejecución de JavaScript.

- Detección avanzada de abusos y fraudes: Los atacantes encontraron eficiencia aprovechando el número cada vez mayor de servicios de proxies[37] anónimos o redes privadas virtuales (VPN) baratas que surgían en todo el mundo.

Para superar los métodos de detección de prueba de trabajo e inteligencia del dispositivo, algunos de los operadores de *bots* más avanzados equiparon sus *botnets* con un motor de ejecución de JavaScript mínimo como Js2Py, que convierte el código JavaScript en código Python. Otros actualizaron sus *botnets* con tecnología de navegador sin interfaz de usuario para ejecutar JavaScript de forma nativa, reduciendo así la eficacia de los métodos de prueba de trabajo y de reputación del dispositivo.

Debido a esto, se requerían nuevamente actualizaciones más avanzadas en las herramientas, incluyendo la detección de comportamiento y de navegadores sin interfaz gráfica. Se necesitaban métodos más complejos de ofuscación del código JavaScript y de la carga útil de la huella digital para forzar al cliente a ejecutar JavaScript. Para detectar mejor los navegadores sin interfaz gráfica, se desarrollaron métodos avanzados de huella digital del cliente para detectar las *botnets* más complejas. La detección biométrica de comportamiento, procesando los movimientos del ratón, los eventos de pulsación de teclas del teclado o los sensores de movimiento de dispositivos móviles, ayudaba a verificar si un humano estaba interactuando con la máquina.

Los equipos de seguridad web no pueden basarse en métodos de detección simples, asumiendo que solo serán atacados por *bots* simples. Los *bots* pueden evolucionar rápidamente su estrategia cuando su

[37] Un servidor proxy es un dispositivo o programa que actúa como intermediario entre un ordenador y la red, permitiendo la conexión a internet de forma indirecta.

actividad es mitigada. Para hacer frente a los mismos, es requerida la combinación de las técnicas de detección descriptas anteriormente, tal como se muestra en la siguiente figura.

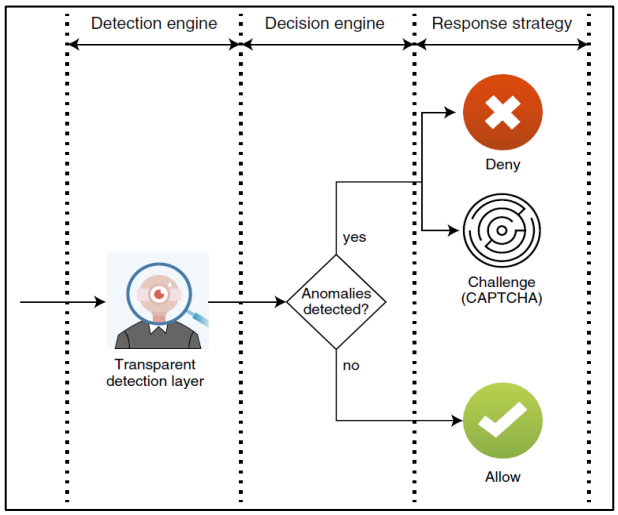

Figura 57. Diagrama de detección de *botnets*. [41]

A modo de resumen, se presenta a continuación un esquema que permite observar las técnicas de detección de *bots* que tienen los métodos mencionados:

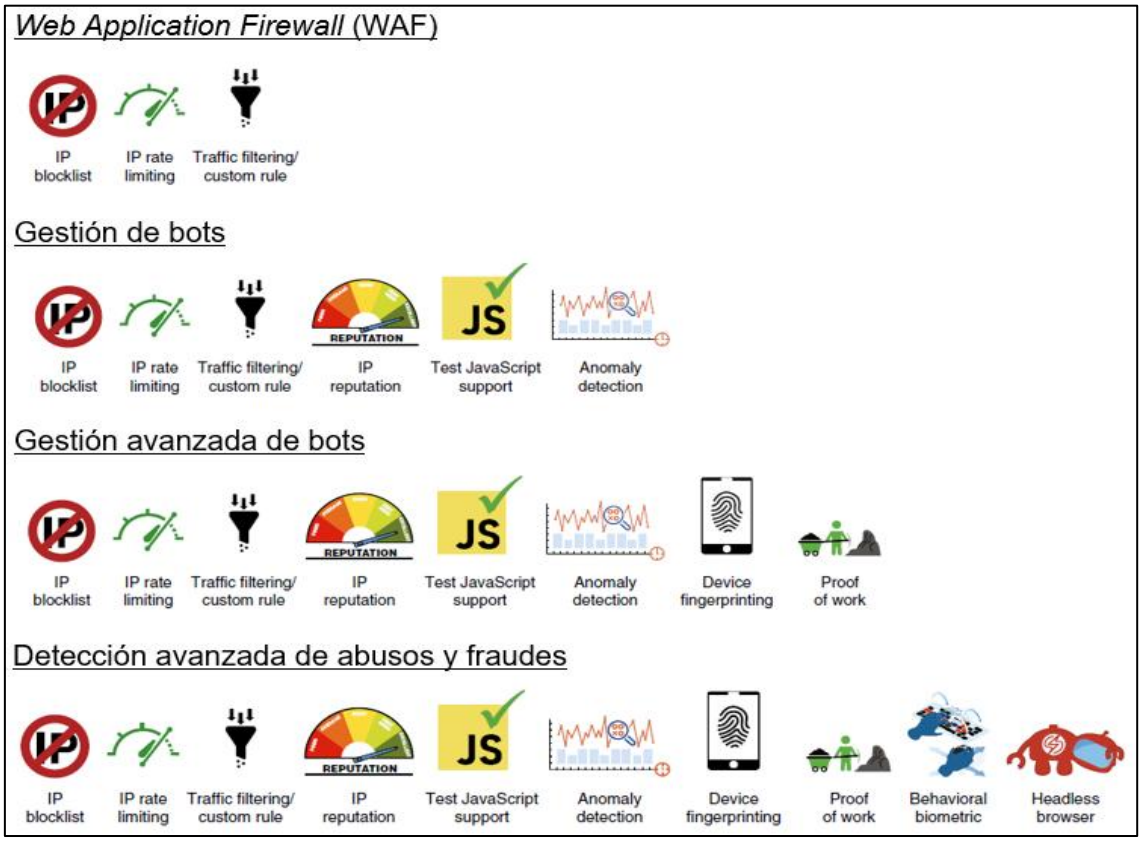

Figura 58. Técnicas de detección de *bots* por método empleado. [41]

14. Medidas de seguridad para la implementación de redes IoT seguras

En función de las vulnerabilidades en IoT identificadas por OWASP, mencionadas en la sección “8. Principales amenazas y vulnerabilidades”, se procede a mencionar medidas y buenas prácticas de seguridad que permitan eliminar o reducir los riesgos inherentes de los dispositivos a un nivel aceptable:

- Contraseñas débiles, fáciles de adivinar o embebidas en el código: utilizar contraseñas que reúnan una serie de requisitos de complejidad tales como: un mínimo de 15 caracteres de longitud, empleando palabras o frases difíciles de adivinar y que no deriven de datos personales ni actividades o cosas vinculadas al usuario y nunca reutilizar contraseñas en varias cuentas. La contraseña debe incluir letras minúsculas y mayúsculas, símbolos y números. [42]

  Por otro lado, los fabricantes de los dispositivos IoT, deberían adoptar medidas de seguridad para asignar por defecto contraseñas seguras y aleatorias, de manera que no existan dos dispositivos de una misma marca y modelo que posean la misma clave. Adicionalmente, se debería incentivar al cliente que adquiere el dispositivo, a cambiar la contraseña que tiene por defecto previo al primer uso. [43]

- Servicios de red inseguros: efectuar un escaneo y análisis de los puertos que posee el dispositivo, haciendo especial énfasis en los que se encuentran activos. Verificar si tales puertos son realmente requeridos para el normal funcionamiento, caso contrario, deben deshabilitarse. Se debe hacer especial énfasis en aquellos puertos que permiten servicios de acceso remoto, que tal vez no son necesarios.

- Ecosistema de interfaces inseguras: algunas de las medidas que se pueden considerar para subsanar las debilidades referidas al presente aspecto son el parcheo constante de las APIs, la implementación de controles de acceso estrictos para limitar el acceso a las mismas, el

despliegue de canales de comunicación seguros entre los diversos componentes del ecosistema IoT y la adopción de métodos de cifrado.

- Ausencia de un mecanismo seguro para la actualización: se debe definir un proceso para la actualización de los dispositivos, que incluya la verificación periódica de la versión de *firmware* instalada, ciclo de vida y parches desplegados por el fabricante para su implementación, mecanismo de *roll-back* o vuelta atrás en caso que la actualización del sistema operativo produzca errores o incompatibilidades y notificaciones a los administradores en caso que un cambio de versión del *firmware* modifique configuraciones de seguridad.
- Utilización de componentes inseguros o desactualizados: estrechamente vinculado al punto anterior, los administradores de los dispositivos IoT deben verificar periódicamente el estado de las versiones del *software* empleado en todo el ecosistema IoT. Debe darse prioridad a aquellos parches que subsanan vulnerabilidades de seguridad de criticidad alta y media. Adicionalmente, se debe mantener una adecuada concientización y estar informado sobre casos de ataques a IoT que puedan estar ocurriendo en otras organizaciones con el fin de tomar las medidas necesarias para reforzar la seguridad y prevenir posibles ataques.
- Insuficiente protección de privacidad: para hacer frente a esta vulnerabilidad, se debe utilizar cifrado para proteger los datos sensibles en almacenamiento y transmisión, y solicitar el consentimiento del usuario para la recopilación o el uso de datos.
- Transferencia y almacenamiento de datos inseguro: tanto para el almacenamiento de información como para la transferencia de la misma, se deben incorporar métodos de cifrado que garanticen la integridad y confidencialidad de la información.
- Deficiente gestión de dispositivos: además de una adecuada gestión del ciclo de vida y parcheado de los dispositivos (como se mencionó anteriormente), también se debe llevar a cabo una adecuada administración de los usuarios y permisos asignados, sistema de alertas

ante cambios en la configuración de seguridad, realización de mantenimiento periódico preventivo y análisis de capacidades de respuesta en base a las necesidades de la operatoria.

- Configuraciones predeterminadas inseguras: previo a la puesta en producción de los diferentes dispositivos que conforman el ecosistema IoT, se debe hacer el correspondiente *hardening*[38] con el fin de configurarlo en base a las mejores prácticas de seguridad, a la vez, permitiendo eliminar cualquier parametrización insegura que provenga con el dispositivo de fábrica. Esto incluye también, la modificación de los usuarios que vienen cargados por defecto con su correspondiente contraseña.

- Debilidades en la seguridad física: algunas medidas que se pueden adoptar para reforzar la seguridad de los dispositivos incluyen deshabilitar o aislar puertos de depuración, utilizarlos para la validación de firmware, adoptar mecanismos de detección de manipulación y no almacenar datos sensibles en tarjetas de memoria extraíbles. A su vez, en la medida de lo posible, colocar los dispositivos IoT en lugares que no tengan público acceso e idealmente que se encuentren siendo monitoreados por cámaras de videovigilancia.

Adicionalmente a las medidas detalladas anteriormente basadas en el *top* 10 de OWASP de ataques de dispositivos IoT, debe mencionarse una de las más importantes ya que es transversal a todas, y se trata de la capacitación y concientización. Esta aporta los siguientes beneficios:

- Previenen infecciones: los usuarios capacitados tienen la capacidad de identificar correos, enlaces y sitios web maliciosos que podrían llegar a introducir *malware* en sus dispositivos.

- Mejoran las prácticas de seguridad: son conscientes de tener que elaborar contraseñas fuertes, mantener actualizado el *software* y disponer de un *software* antivirus.

[38] Proceso de asegurar un sistema mediante la reducción de vulnerabilidades en el mismo.

- Detectan señales de ataques: los usuarios capacitados pueden reconocer comportamientos inusuales en sus dispositivos, como lentitud o uso excesivo de datos.
- Fomentan la colaboración con TI: facilitan la identificación y reporte rápido de amenazas.

15. Conclusiones

Si bien la incorporación de la tecnología de IoT está presentando grandes avances en el sector hogareño como en el industrial, incluso se está planteando su utilización para la creación de ciudades inteligentes y su posible combinación con la tecnología blockchain e inteligencia artificial, esta presenta una serie de desventajas que la hace vulnerable y propensa a ataques de ciberdelincuentes.

Algunos aspectos que deben considerarse al momento de la adquisición y puesta en producción de los dispositivos IoT, es que deben configurarse inicialmente para que cumplan con ciertos requisitos de seguridad basado en buenas prácticas, y entre otros aspectos, deshabilitando puertos y servicios innecesarios o inseguros, y modificando los usuarios y contraseñas que vienen por defecto.

Una inadecuada configuración en cualquiera de los componentes de un ecosistema IoT, puede generar brechas de seguridad que podrían ser explotadas no solo dando la posibilidad de tener acceso a información confidencial, sino también, en el entorno industrial podría dar lugar a consecuencias catastróficas, ya que podrían tener acceso a manipular información de sensores y actuadores los cuáles son vitales para gestionar los diferentes procesos en los que intervienen productos químicos, gases, manejo de presiones, cargas eléctricas, etc.

En adición a las consecuencias mencionadas anteriormente, los atacantes podrían convertir tales dispositivos vulnerables en componentes de una *botnet*. Esto no solo hace que el rendimiento de los equipos infectados sea muy inferior en base a las características de *hardware* que presenta, sino que también, permite al ciberdelincuente utilizarlos para realizar ataques más complejos. Entre otros, minería de criptomonedas, spam/phishing y ataques de denegación de servicios distribuida provocando la disrupción de la continuidad operativa de una empresa.

En conclusión, si bien la tecnología IoT permite la mejora de la eficiencia en el entorno industrial a través de la automatización de diferentes procesos y

mejora la calidad de vida de las personas a través de su inclusión en el ámbito hogareño y de la salud, es indispensable gestionar a estos dispositivos correctamente, configurándolos y actualizándolos periódicamente con el fin de disminuir significativamente las vulnerabilidades que van surgiendo. Cabe mencionar que, la capacitación y concientización de los usuarios es un elemento crucial para preservar la seguridad de estas redes, ya que son los que trabajan con ellas directamente y ante distintas anomalías pueden notificar con mayor celeridad a los técnicos o áreas que correspondan.

16. Referencias

[1] IBM, «¿Qué es el Internet de las cosas (IoT)?,» [En línea]. Available: https://www.ibm.com/es-es/topics/internet-of-things.

[2] Cisco, «What Is IoT (Internet of Things)?,» [En línea]. Available: https://www.cisco.com/c/en/us/solutions/internet-of-things/what-is-iot.html.

[3] Oracle, «¿Qué es el IoT?,» [En línea]. Available: https://www.oracle.com/ar/internet-of-things/.

[4] G. J. L. Agrawal, Internet of Things - Theory to Practice, CRC Press, 2024.

[5] BBC, «The toaster that changed the world,» [En línea]. Available: https://www.bbc.com/storyworks/future/the-new-wave/innovation.

[6] A. Braun, «History of IoT: A Timeline of Development,» [En línea]. Available: https://www.iottechtrends.com/history-of-iot/.

[7] ITWeb, «No need for PCs with intelligent fridges,» [En línea]. Available: https://www.itweb.co.za/article/no-need-for-pcs-with-intelligent-fridges/KA3WwqdlozkqrydZ.

[8] «The Sidney Morning Herald,» [En línea]. Available: https://www.smh.com.au/technology/towards-a-consensual-hallucination-20120524-1z7d7.html.

[9] IPhone, «Apple Fandom,» [En línea]. Available: https://apple.fandom.com/es/wiki/IPhone_2G.

[10] «Waymo,» [En línea]. Available: https://waymo.com/intl/es/waymo-driver/.

[11] S. Coding, «Internet Of Things (IoT),» [En línea]. Available: https://simplycoding.in/internet-of-things/.

[12] S. Abrams, Internet of Things for Beginners - Comprehensive Guide to a Connected World, 2024.

[13] «AlfaIoT - Sensores IoT populares,» [En línea]. Available: https://alfaiot.com/iot/sensores-iot-populares-tipos-y-funciones/.

[14] Cisco, «Reference Model for IoT,» [En línea]. Available: https://www.cisco.com/c/dam/global/en_ph/assets/ciscoconnect/pdf/bigdata/jim_green_cisco_connect.pdf. [Último acceso: 10 12 2024].

[15] «Altexsoft - IoT Architecture Layers Components,» [En línea]. Available: https://www.altexsoft.com/blog/iot-architecture-layers-components/. [Último acceso: 11 12 2024].

[16] «AlfaIoT - LoRa,» 10 1 2025. [En línea]. Available: https://alfaiot.com/iot/lora-una-tecnologia-lpwan-ideal-para-el-internet-de-las-cosas/.

[17] «SomosPNT - Colas de Mensajería Protocolo AMQP,» [En línea]. Available: https://somospnt.com/blog/325-colas-de-mensajeria-protocolo-amqp#:~:text=%C2%BFC%C3%B3mo%20funciona%20el%20protocolo%20AMQP,se%20coloca%20en%20una%20cola.. [Último acceso: 28 12 2024].

[18] «DDS - What is DDS,» [En línea]. Available: https://www.dds-foundation.org/what-is-dds-3/. [Último acceso: 28 12 2024].

[19] «INCIBE - IoT Protocolos de Comunicación - Ataques y Recomendaciones,» [En línea]. Available: https://www.incibe.es/incibe-cert/blog/iot-protocolos-comunicacion-ataques-y-recomendaciones. [Último acceso: 17 12 2024].

[20] «KaaIoT - IoT knowledge base,» 29 12 2024. [En línea]. Available: https://www.kaaiot.com/iot-knowledge-base/how-does-aiot-differ-from-traditional-iot.

[21] «Medium - The Future of Robotics in Smart Home Technology,» 12 1 2025. [En línea]. Available: https://tcaflisch.medium.com/the-future-of-robotics-in-smart-home-technology-89db93116a90.

[22] «INELTEC - Tesla Muestra Avances Humanoide Optimus,» [En línea]. Available: https://www.ineltec.es/noticias/tecnologia/tesla-muestra-avances-humanoide-optimus.

[23] «TechTarget - Top Advantages and Disadvantages of IoT,» [En línea]. Available: https://www.techtarget.com/iotagenda/tip/Top-advantages-and-disadvantages-of-IoT-in-business.

[24] «NMAP - RTSP URL Brute,» [En línea]. Available: https://nmap.org/nsedoc/scripts/rtsp-url-brute.html. [Último acceso: 11 02 2025].

[25] S. D., The Reign of Botnets. Defending Against Abuses, Bots and Fraud, 2024.

[26] «Kaspersky - Botnet Attacks,» [En línea]. Available: https://www.kaspersky.com/resource-center/threats/botnet-attacks.

[27] «Researchgate - Red de Bots con Arq. Estrella o Centralizada,» [En línea]. Available: https://www.researchgate.net/figure/Figura-2-Red-de-Bots-con-arquitectura-estrella-o-centralizada_fig1_356958002.

[28] «Radware - Bot Management,» 01 01 2025. [En línea]. Available: https://www.radware.com/cyberpedia/bot-management/botnet/.

[29] «University of Central Florida - Peer to Peer Botnets,» 2 1 2025. [En línea]. Available: https://www.cs.ucf.edu/~czou/research/P2PBotnets-bookChapter.pdf.

[30] «Kaspersky - DDoS Attacks,» 3 1 2025. [En línea]. Available: https://latam.kaspersky.com/resource-center/threats/ddos-attacks?srsltid=AfmBOorgauK3CYgGH6zQc-NMrPsQWw8g3vCNBl0EjawTa1xjh55JjoF3.

[31] «Indusface - What is a DDoS botnet,» 3 12 2025. [En línea]. Available: https://www.indusface.com/learning/what-is-a-ddos-botnet/.

[32] «Akamai - What is DDoS,» [En línea]. Available: https://www.akamai.com/es/glossary/what-is-ddos.

[33] «AKAMAI - Prolexic Solutions,» 12 1 2025. [En línea]. Available: https://www.akamai.com/es/products/prolexic-solutions.

[34] «Akamai - Prolexic DDoS Protection,» 12 1 2025. [En línea]. Available: https://www.akamai.com/es/resources/reference-architecture/prolexic-ddos-protection.

[35] «ECCOUNCIL - Botnet Attack Prevention,» 3 12 2025. [En línea]. Available: https://www.eccouncil.org/cybersecurity-exchange/penetration-testing/botnet-attack-prevention/.

[36] «Palo Alto Networks - Cryptocurrency mining botnet,» 3 1 2025. [En línea]. Available: https://unit42.paloaltonetworks.com/pgminer-postgresql-cryptocurrency-mining-botnet/.

[37] «Radware - DDoS,» 4 1 2025. [En línea]. Available: https://www.radware.com/security/threat-advisories-and-attack-reports/infrashutdown-anonymous-sudan-partners-with-ddos-for-hire-operator/.

[38] «Fraud0 - Methbot,» 4 1 2025. [En línea]. Available: https://www.fraud0.com/resources/methbot/.

[39] «Cloudflare - Mirai Botnet,» [En línea]. Available: https://www.cloudflare.com/es-es/learning/ddos/glossary/mirai-botnet/.

[40] «ZDNET - FBI Dismantles Gigantic Ad Fraud Scheme Operating Across Over One Million- IPs,» 4 1 2025. [En línea]. Available: https://www.zdnet.com/article/fbi-dismantles-gigantic-ad-fraud-scheme-operating-across-over-one-million-ips/.

[41] D. Sénécal, The Reign Of Botnets - Defending Against Abuses, Bots and Fraud on the Internet, Wiley, 2024.

[42] «AVG - Cómo crear una contraseña segura,» 4 1 2025. [En línea]. Available: https://www.avg.com/es/signal/how-to-create-a-strong-password-that-you-wont-forget#:~:text=La%20complejidad%20se%20utiliza%20a,as%C3%AD%20como%20n%C3%BAmeros%20y%20s%C3%ADmbolos..

[43] «Wattlecorp - OWASP IoT Top10 Risks,» 4 1 2025. [En línea]. Available: https://www.wattlecorp.com/owasp-iot-top-10/#1_Weak_Guessable_or_Hardcoded_Passwords.