

---

# Las redes como medio para obtener datos

---

PID\_00269512

Toni Adame  
Lorenza Giupponi

**Toni Adame**

Ingeniero superior de Telecomunicaciones por la Universidad Politécnica de Cataluña (UPC) desde 2009. Ha trabajado durante varios años en el sector privado como consultor preventa en el ámbito de la integración de sistemas y tecnologías. Desde 2013 pertenece al grupo de investigación Network Technologies and Strategies (NeTS) de la Universidad Pompeu Fabra (UPF), donde ejerce como investigador sénior en proyectos europeos y nacionales que promueven el uso de tecnologías de comunicación inalámbricas (WSN, wifi, LPWAN, redes celulares y RFID) como habilitadoras del *Internet of Things* (IoT). Sus áreas de investigación se focalizan en las comunicaciones de acceso múltiple, los protocolos de acceso al medio y los mecanismos de ahorro energético. Desde 2017 también es profesor asociado en la UPF, habiendo impartido clases en los diferentes grados de Ingeniería de dicha universidad.

**Lorenza Giupponi**

Ingeniera de Telecomunicaciones por la Universidad de Roma «La Sapienza» y doctora por el departamento de Teoría de la Señal y Comunicaciones de la Universidad Politécnica de Cataluña (2007). Se unió en el año 2003 al grupo de comunicaciones móviles de la UPC con una beca del programa Formación Profesorado Universitario. Durante 2006 y 2007 fue profesora ayudante en la UPC. En septiembre de 2007 se unió al Centro Tecnológico de Telecomunicaciones de Cataluña (CTTC), donde actualmente es una investigadora sénior en el departamento de redes móviles. Además, desde 2007 es parte del comité de dirección del CTTC con el cargo de directora de Relaciones Institucionales. Ha recibido premios al mejor artículo de la conferencia en tres ocasiones, entre ellas IEEE CCNC 2010, IEEE WCNC 2018. Dos de sus *Transactions* han sido listados por IEEE Comsoc entre mejores lecturas en el área de la gestión de recursos radio y la radio cognitiva. Desde 2015 es miembro del comité ejecutivo de ns-3, responsable del área Long Term Evolution (LTE) y New Radio (NR). Ha liderado y participado en múltiples proyectos nacionales, europeos y financiados por empresas internacionales. Ha supervisado dos tesis doctorales ya defendidas y su índice h es 27.

El encargo y la creación de este recurso de aprendizaje UOC han sido coordinados por la profesora: Cristina Cano Bastidas (2020).

Primera edición: febrero 2020  
© Toni Adame, Lorenza Giupponi  
Todos los derechos reservados  
© de esta edición, FUOC, 2020  
Av. Tibidabo, 39-43, 08035 Barcelona  
Realización editorial: FUOC

*Ninguna parte de esta publicación, incluido el diseño general y la cubierta, puede ser copiada, reproducida, almacenada o transmitida de ninguna forma, ni por ningún medio, sea este eléctrico, químico, mecánico, óptico, grabación, fotocopia, o cualquier otro, sin la previa autorización escrita de los titulares de los derechos.*

## Índice

<b>Introducción</b>	5
<b>1 Arquitectura de redes de datos</b>	6
1.1 Estructura y elementos de red	7
1.2 Topologías de red	7
1.3 Interconexión entre redes de datos	8
1.4 Requisitos de un sistema de interconexión	9
1.5 Puertos y <i>sockets</i>	9
1.6 Modelos de intercambio de datos	10
<b>2 Obtención de datos de redes externas</b>	11
2.1 Extracción de datos en redes tradicionales: API	11
2.1.1 REST	12
2.1.2 SOAP	12
2.2 Extracción de datos en redes IoT	14
2.2.1 CoAP	14
2.2.2 MQTT	14
<b>Ejercicios de autoevaluación</b>	16
<b>Solucionario</b>	17
<b>Glosario</b>	18
<b>Bibliografía</b>	19

## Introducción

En el “**Reto 1. ¿Cuál es el papel de las redes de computadores en el ciclo de vida de los datos?**” se ofreció una visión general de las redes de computadores en la actualidad y las funciones que desempeñan en las diferentes fases de cualquier aplicación o servicio basado en la transmisión y procesamiento de datos.

A continuación, en el “**Reto 2. Las redes de computadores como generadoras de datos**”, se profundizó en la descripción de aquellas tecnologías que habilitan la captación y transmisión de la información desde el mismo lugar en que esta fue generada hasta el primer elemento de la red que permita su procesamiento. Con respecto al modelo OSI, dicho reto se centró en las funciones desempeñadas por las capas física, de enlace de datos y de red como habilitadoras de las comunicaciones de red.

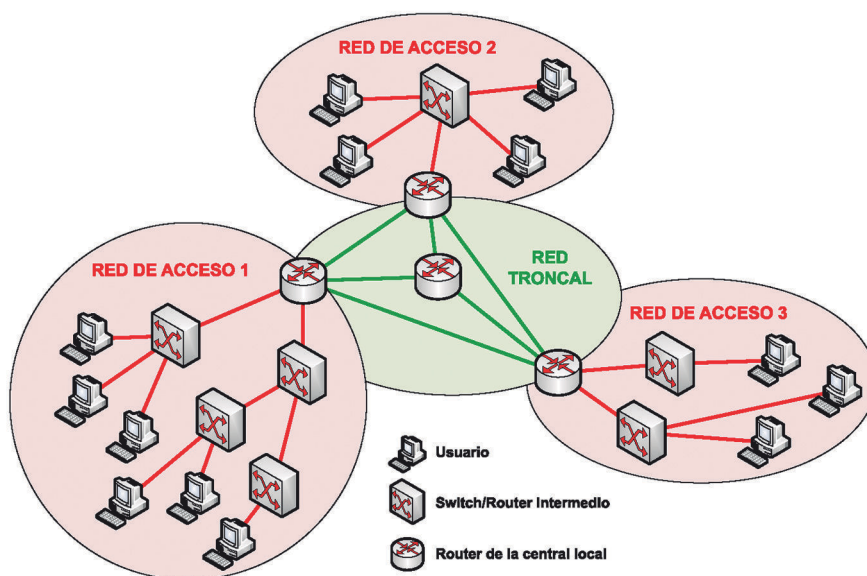
El objetivo principal de este actual “**Reto 3. Las redes como medio para obtener datos**”, consiste en presentar las redes de comunicaciones como una herramienta para extraer datos de redes externas, ya sea a través de APIs o protocolos específicamente diseñados para tal fin. Además, en este reto se estudiará el rol que juegan las capas de transporte y de aplicación del modelo OSI en dicho proceso de extracción de datos.

En el presente documento se hace especial hincapié en la descripción de la arquitectura, la topología y los principales elementos de una red de datos que permiten la interconexión con otros tipos de redes, así como la extracción de datos de estas. Además, también se profundiza en el concepto de APIs (el principal elemento software que habilita el intercambio de datos a nivel de aplicación) y se describen dos de los protocolos más usados para la extracción de datos en redes IoT: CoAP y MQTT.

## 1. Arquitectura de redes de datos

Se denomina red de datos al conjunto de sistemas de transmisión y, cuando proceda, equipos de conmutación y demás recursos que permitan el intercambio de datos entre dos o más dispositivos de dicha red. La arquitectura de una red de datos y, de forma general, de cualquier red de telecomunicaciones, puede dividirse en una **red de acceso** y una **red troncal**, tal y como puede observarse en la figura 1.

Figura 1. Diagrama de red de acceso y red troncal



- **Red de acceso:** Es el conjunto de elementos que permiten establecer una conexión permanente a cada abonado con la central local de la que depende. Aunque existe gran cantidad de posibles tecnologías; en la práctica se usa principalmente el acceso por cable físico (cable de cobre o fibra óptica) y el acceso inalámbrico (celular o wifi).
- **Red troncal (o de transporte):** Es la red de transmisión que interconecta las diferentes centrales locales entre sí. Está constituida por sistemas de transmisión de diferentes tecnologías, tales como pares de cobre, fibra óptica (el más común actualmente), cable coaxial, microondas o satélite.

## 1.1 Estructura y elementos de red

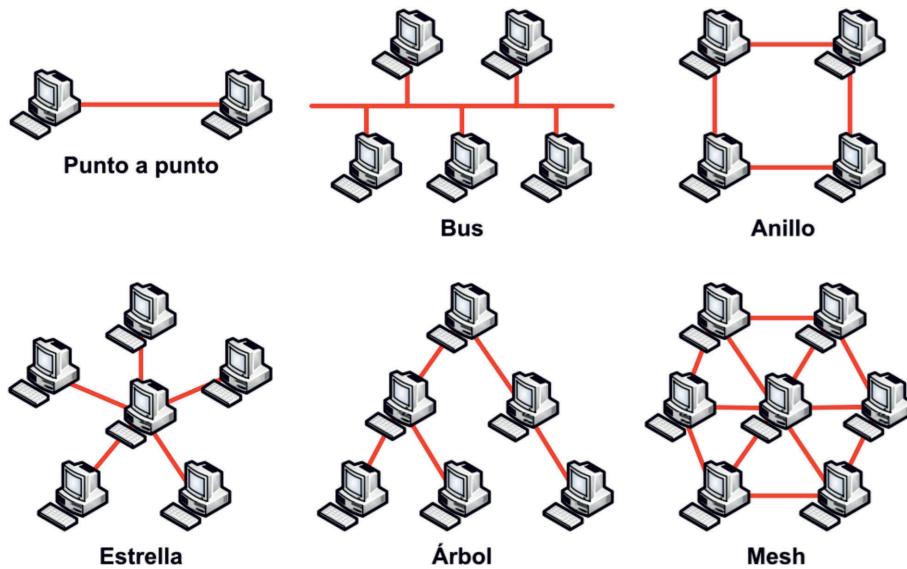
En general, una red de telecomunicación contiene los siguientes elementos:

- **Terminal:** Es el elemento de origen o destinación de una comunicación.
- **Enlace:** Es el elemento de transmisión y tiene la capacidad de transportar la información entre los diferentes elementos de la red. Está formado por circuitos por donde viaja la información de una comunicación completa. Existen dos tipos de enlaces:
  - Enlace de acceso: Es utilizado por uno o varios usuarios, normalmente es de baja velocidad y es la parte que requiere de más infraestructura.
  - Enlace troncal: Es utilizado de forma simultánea por muchos usuarios, es de alta velocidad y suele utilizar la tecnología más moderna.
- **Nodo:** Es el elemento principal de interconexión de la red troncal. Entre sus principales funciones destacan:
  - Conmutación: Establecimiento de caminos entre un origen y una destinación.
  - Repetición: Regeneración de señales.
  - Supervisión y control.

## 1.2 Topologías de red

En el contexto de una red de comunicaciones, el término *topología* se refiere a la forma en que se interconectan entre sí los dispositivos pertenecientes a dicha red. Existen múltiples topologías de red, compiladas en la figura 2 y descritas a continuación.

Figura 2. Diagrama de las diferentes topologías de red



- **Punto a punto:** Conectan únicamente dos dispositivos, situados en cada extremo de la red. Cada dispositivo puede tomar el rol de emisor o la función de receptor.
- **Bus:** Se caracteriza por tener un único canal de comunicaciones compartido al cual se conectan los diferentes dispositivos.
- **Anillo:** La red consta de un conjunto de **repetidores** unidos por enlaces punto a punto formando un bucle cerrado. Solo podrá iniciar una comunicación aquel dispositivo que tenga permiso para hacerlo mediante el uso de un *token*.
- **Estrella:** Todos los dispositivos están conectados directamente a un punto central, de modo que cualquier comunicación ha de realizarse obligatoriamente a través de dicho punto.
- **Árbol:** Los dispositivos se organizan en forma de árbol; es decir, como una serie de redes en estrella pero sin un punto central.
- **Mesh:** En una red *mesh* (o mallada), todos los dispositivos están conectados (directa o indirectamente) entre ellos. Así, es posible llevar los mensajes de un dispositivo a otro por distintos caminos.

### 1.3 Interconexión entre redes de datos

La interconexión entre redes de datos permite a sus usuarios tener acceso a los recursos existentes más allá de los que se disponen en su propia red. Debido a la heterogeneidad de redes de datos, no es práctico tratar de agrupar todos los recursos en una única red, sino que lo prioritario es posibilitar la interconexión entre varias redes para que dos estaciones cualesquiera se puedan comunicar independientemente de su red de origen.

Desde el punto de vista de un usuario, un conjunto de redes interconectadas puede considerarse simplemente como una red más grande. Sin embargo, si cada una de las redes constituyentes retiene su identidad y son necesarios un conjunto de protocolos para la comunicación a través de múltiples redes, entonces a la configuración entera se la conoce como conjunto de redes.

Internet puede considerarse como el conjunto de redes interconectadas más grande e importante de nuestro planeta, ya que está formada por millones de redes domésticas, académicas, comerciales y gubernamentales. Gracias a la interconexión de redes existente en internet, un usuario de un dispositivo electrónico puede acceder a todo un conjunto de servicios (y datos) desde cualquier parte del mundo.

## 1.4 Requisitos de un sistema de interconexión


Los requisitos globales para un sistema de interconexión entre redes, independientemente de su tamaño, son los que siguen a continuación:

- 1) Proporcionar un enlace entre redes. Como mínimo, se necesita una conexión física y de control del enlace.
- 2) Proporcionar el encaminamiento y entrega de datos entre procesos en diferentes redes.
- 3) Proporcionar un servicio de monitorización que realice un seguimiento de la utilización de las diferentes redes y dispositivos de encaminamiento (*routers*) y mantenga información de estado.
- 4) Proporcionar los servicios mencionados de forma que no se requiera la modificación de la arquitectura de red de cualquiera de las redes interconectadas. Esto significa que el sistema de interconexión entre redes se debe acomodar a las diferencias existentes entre las distintas redes.

## 1.5 Puertos y *sockets*

Para intercambiar datos en la aplicación entre dos dispositivos conectados a internet no solo es necesario conocer la dirección IP origen y destino (elementos de la **capa de red**), sino también el puerto origen y destino (elementos de la **capa de transporte**).

Al conjunto formado por una dirección IP y un número de puerto de red se le denomina *socket*, y constituye el elemento que utili-



Los conceptos de puerto y *socket* son fundamentales en internet. Podéis encontrar más información sobre su uso y, en general, sobre TCP/IP, en los apartados 1 y 3 del documento *El nivel de transporte* asociado a este reto.



za un proceso (o aplicación) para pedir servicios de red al sistema operativo.

## 1.6 Modelos de intercambio de datos

Cuando se establece un sistema de interconexión entre redes de datos, los dispositivos de diferentes redes son capaces de intercambiar datos como si se alojaran en la misma red. Por tanto, podrán aplicar cualquiera de los siguientes modelos de intercambio de datos: cliente/servidor y *peer-to-peer* (P2P).




Podéis encontrar más información sobre los modelos de intercambio de datos en el apartado 1 del documento *Las capas superiores del modelo OSI: sesión, presentación y aplicación* asociado a este reto.

## 2. Obtención de datos de redes externas

Tal y como se ha descrito en el apartado anterior, es muy habitual que las organizaciones y los usuarios particulares no solo utilicen información alojada en sus propias redes, sino que también necesiten extraer datos de redes externas para alimentar sus propias aplicaciones y servicios.

Una vez conectadas físicamente la red origen y la red externa, es necesario establecer un modelo de comunicación para habilitar el intercambio de datos entre ellas. Como se ha descrito en retos anteriores, si esta interconexión se realiza a través de internet, la comunicación estará basada en la arquitectura de protocolos TCP/IP.

No obstante, por encima de TCP/IP es necesario definir la capa de aplicación más adecuada que permita acceder a los servicios del resto de capas y defina los protocolos que utilizan las aplicaciones para intercambiar datos. A continuación veremos algunos ejemplos de capas de aplicación utilizadas en la actualidad, ya sean APIs o protocolos específicos de las redes IoT.



Podéis encontrar más información sobre la capa de aplicación y algunos de sus principales protocolos genéricos en la introducción y en el apartado 2, respectivamente, del documento *Las capas superiores del modelo OSI: sesión, presentación y aplicación* asociado a este reto.

### 2.1 Extracción de datos en redes tradicionales: API

Una API, abreviatura del término en inglés *application programming interface* (interfaz de programación de aplicaciones) es una especificación formal sobre cómo un módulo de un software se comunica o interactúa con otro. Una API puede definirse para un sistema web, para un sistema operativo, para una base de datos, un hardware específico o una librería de software.

Técnicamente, una API define un conjunto de comandos, funciones y protocolos informáticos que permiten a los desarrolladores crear programas específicos para ciertos sistemas. Gracias a las APIs, en definitiva, se pueden intercambiar datos de otras aplicaciones o sitios web mediante mecanismos que aportan un mayor grado de abstracción sobre el software.

Las web APIs son un tipo particular de API basadas en el protocolo de aplicación HTTP que ejecutan los servidores y navegadores web. Existe un gran cantidad de sitios web y redes sociales que proveen

#### Enlace recomendado

Directorio de proveedores de APIs en sitios web:  
[www.programmableweb.com/apis/directory](http://www.programmableweb.com/apis/directory)

de web APIs para interactuar con ellos, como por ejemplo Twitter, Google, Facebook, YouTube, WordPress o Amazon, entre otros.

Las web APIs constituyen un concepto y no una tecnología específica. De hecho, las web APIs pueden construirse a partir de una gran variedad de tecnologías, como por ejemplo las dos que se analizarán en detalle en el presente documento: REST y SOAP.

### 2.1.1 REST

REST (REpresentational State Transfer) es una arquitectura de desarrollo web que se apoya totalmente en el estándar HTTP para el intercambio de datos. REST se compone de una lista de reglas que debe cumplir el diseño de la arquitectura de una API para que esta se considere como **RESTful**.

REST normaliza el uso de **hipermedios** (término que se refiere al conjunto de procedimientos para crear contenidos web que contengan texto, imagen, vídeo, audio y otros métodos de información) para permitir al usuario navegar por los distintos recursos de una API a través de enlaces HTML.

Las operaciones más importantes relacionadas con los datos en cualquier sistema REST y la especificación HTTP son cuatro:

- POST (crear),
- GET (leer y consultar),
- PUT (editar) y
- DELETE (eliminar).

La arquitectura REST permite a los proveedores de APIs entregar datos en múltiples formatos, como texto plano, HTML, XML y JSON, entre otros. De todos ellos, el formato de intercambio de datos más utilizado en REST es JSON (JavaScript Object Notation), dado su formato ligero y de fácil lectura, así como su rapidez y sencillez a la hora de intercambiar datos.

#### Enlace recomendado

Reglas de diseño RESTful:  
<https://restfulapi.net/rest-architecturalconstraints/>

#### Enlace recomendado

Descripción del formato de intercambio de datos JSON:  
[www.json.org/](http://www.json.org/)

### 2.1.2 SOAP

SOAP (Simple Object Access Protocol) es un protocolo que permite la comunicación entre aplicaciones a través de mensajes por medio

Figura 3. Ejemplo de mensaje JSON

```
{
  "nombre": "DEVICE_1",
  "valor1": 1535,
  "sensors": [
    {
      "name": "Sensor_hum",
      "description": "Mide Humedad",
      "sensor-type": "Humedad"
    },
    {
      "name": "Sensor_temp",
      "description": "Mide Temperatura",
      "sensor-type": "Temperatura"
    }
  ]
}
```

de internet. Únicamente utiliza XML (eXtensible Markup Language) como formato de intercambio de datos y normalmente funciona sobre HTTP para invocar un *web service*. Sin embargo, no está limitado a este protocolo, sino que también puede utilizar FTP, POP3, TCP o SMTP.

SOAP es un sistema de mensajería de una dirección sin estado, que puede ser utilizado para formar protocolos más complejos y completos según las necesidades de las aplicaciones. La estructura básica de un mensaje SOAP debe contener un **envelope**, para describir qué información contiene un mensaje y cómo procesarlo, las **reglas de codificación** para expresar instancias de tipos de datos y un **estilo de comunicación** a elegir entre RPC (Remote Procedure Call), donde se invoca una operación que devuelve un resultado, o *documento*, que está orientado a mensajes.

Figura 4. Ejemplo de mensaje SOAP

```
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <!--Optional header information goes here.-->
    <To>Scott</To>
    <From>Suzanne</From>
  </soap:Header>
  <soap:Body>
    <!--Message goes here.-->
    Please pick up some milk on your way home from work.
  </soap:Body>
</soap:Envelope>
```

## 2.2 Extracción de datos en redes IoT

En general, las redes IoT están compuestas por dispositivos que tienen claras limitaciones en términos de procesado, memoria, energía y capacidades de comunicación. Si bien es cierto que algunas redes IoT son compatibles con las web APIs, las limitaciones mencionadas anteriormente han motivado el desarrollo de mecanismos alternativos y con un mayor nivel de simplicidad para la extracción de datos en este tipo de redes.

De entre las múltiples opciones existentes, en las siguientes líneas se hará especial énfasis en dos tecnologías concretas: CoAP y MQTT.

### 2.2.1 CoAP

Constrained Application Protocol (CoAP) es un protocolo de la capa de aplicación que permite la transferencia de datos entre dispositivos de recursos limitados en el ámbito del IoT; por ejemplo, en el interior de una red de sensores inalámbricos (WSN).

CoAP implementa una versión simplificada del modelo REST de HTTP (con las operaciones GET, POST, PUT y DELETE), utiliza UDP como protocolo de transporte y sigue un modelo de intercambio de datos cliente/servidor.

#### Enlace recomendado

Especificación de CoAP:  
<https://coap.technology/>

Podéis encontrar más información sobre UDP en el apartado 2 del documento *El nivel de transporte* asociado a este reto.



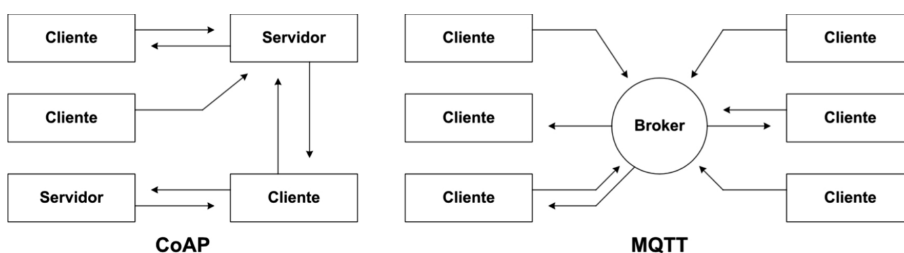
### 2.2.2 MQTT

Al igual que CoAP, Message Queuing Telemetry Transport (MQTT) es un protocolo de la capa de aplicación orientado al intercambio de datos en el IoT. A diferencia de CoAP, MQTT usa TCP como protocolo de transporte y sigue un modelo de intercambio de datos *publisher/subscriber*.

#### Enlace recomendado

Especificación de MQTT:  
<http://mqtt.org/>

Figura 5. Esquema de comunicaciones en CoAP y MQTT



Típicamente, MQTT sigue una topología de estrella, con un nodo central que hace de servidor (o *broker*). El *broker* es el encargado de

gestionar la red, mantener activo el canal y transmitir los mensajes entre un origen y un destino.

## Ejercicios de autoevaluación

1. ¿En qué dos redes puede dividirse cualquier red de telecomunicaciones?
  - a) Red de datos y red de intercambio de ficheros.
  - b) Red de acceso y red troncal.
  - c) Red inalámbrica y red cableada.
  - d) Red de transmisión de audio y red de transmisión de vídeo.
2. ¿Cuáles son las principales características de un enlace troncal?
  - a) Es utilizado por uno o varios usuarios, normalmente es de baja velocidad y es la parte que requiere de más infraestructura.
  - b) Es utilizado por pocos usuarios, es de alta velocidad y utiliza medios inalámbricos.
  - c) Es utilizado de forma simultánea por muchos usuarios, es de alta velocidad y suele utilizar la tecnología más moderna.
  - d) Es utilizado de forma simultánea por muchos usuarios y utiliza bajas velocidades de transmisión de datos.
3. ¿Cuál de las siguientes opciones NO es una topología de red?
  - a) Estrella.
  - b) Árbol.
  - c) Anillo.
  - d) Rombo.
4. ¿Cuál de las siguientes opciones NO puede considerarse como un requisito para un sistema de interconexión entre redes?
  - a) Proporcionar un sistema de monitorización de las redes y los dispositivos de encaminamiento.
  - b) Proporcionar un servicio de almacenamiento de datos.
  - c) Proporcionar el encaminamiento y entrega de datos entre procesos en diferentes redes.
  - d) Proporcionar un enlace entre redes.
5. ¿Cómo se denomina el conjunto formado por una dirección IP y un número de puerto?
  - a) Dirección de transporte.
  - b) Telnet.
  - c) Dirección de red.
  - d) *Socket*.
6. ¿En qué protocolo de aplicación se basan las web APIs?
  - a) HTTP.
  - b) FTP.
  - c) TCP.
  - d) SMTP.
7. ¿Cuál de las siguientes operaciones NO está definida en un sistema REST?
  - a) DELETE.
  - b) POST.
  - c) SWITCH.
  - d) GET.
8. ¿Qué modelo de intercambio de datos utiliza MQTT?
  - a) *Peer-to-peer*.
  - b) *Publisher/subscriber*.
  - c) Cliente/servidor.
  - d) Ninguno de los anteriores.

## **Solucionario**

1. b

2. c

3. d

4. b

5. d

6. a

7. c

8. b



## Glosario

**API** Application Programming Interface

**CoAP** Constrained Application Protocol

**FTP** File Transfer Protocol

**HTML** HyperText Markup Language

**HTTP** HyperText Transfer Protocol

**IoT** *Internet of Things*

**IP** Internet Protocol

**JSON** JavaScript Object Notation

**MQTT** Message Queuing Telemetry Transport

**OSI** Open Systems Interconnection

**P2P** *Peer-to-Peer*

**POP3** Post Office Protocol version 3

**REST** REpresentational State Transfer

**RPC** Remote Procedure Call

**SMTP** Simple Mail Transfer Protocol

**SOAP** Simple Object Access Protocol

**TCP** Transmission Control Protocol

**UDP** User Datagram Protocol

**WSN** Wireless Sensor Network

**XML** eXtensible Markup Language

## Bibliografía

**Comer, D. E.** (2009). *Computer networks and internets* (5.<sup>a</sup>ed.). Nueva Jersey: Prentice-Hall, Inc.

**Halili, F.; Ramadani, E.** (2018). *Web services: a comparison of SOAP and REST services*. Modern Applied Science (vol. 12, núm. 3, pág. 175).

**Naik, N.** (2017). *Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP*. IEEE international systems engineering symposium (ISSE) (pág. 1-7).

**Stallings, W.** (2004). *Comunicaciones y Redes de Computadores* (7.<sup>a</sup>ed.). Madrid: Pearson Educación, S. A.