CIBERSEGURIDAD EN INTERNET OF THINGS

Análisis de amenazas, riesgos y vulnerabilidades

Alumno: Daniel Domínguez Margareto

Trabajo Final de Máster: Máster Interuniversitario de la Seguridad en las Tecnologías de la Información y las comunicaciones (MISTIC)

Tutora: Ángela María García Valdés

Directora del Máster: Helena Rifá Pous

Fecha de entrega: 02/06/2020

RESUMEN

El aumento de la popularidad de las IoT ha expuesto de forma evidente las carencias de seguridad de la mayoría de estos sistemas. Gracias a la evolución tecnológica actual, la facilidad para implementar estos sistemas y su extensión, así como su creciente sofisticación y complejidad, hace que los problemas de seguridad se agraven puesto que operan en sistemas cada vez más críticos y cuya operativa e integración con otros subsistemas ha de estar perfectamente implantada y a prueba de ciberataques e intentos de sabotaje.

El objetivo del presente trabajo es poner de manifiesto la importancia de la seguridad en los sistemas IoT, para lo que se expondrá con un caso concreto qué problemas o fallos de seguridad existen en el sistema modelado, y que habrá que intentar contener y mitigar siguiendo ciertas buenas prácticas, usando herramientas adicionales, y si es necesario y lo permiten las circunstancias, cambiando el diseño y la implementación del mismo.

Para ello en primer lugar se procederá a explicar detalladamente el concepto de IoT, el estado del arte y evolución tecnológica, se intentará evidenciar la importancia de la seguridad en estos sistemas, y a continuación se enumerarán los tipos de IoT existentes. Se procederá después con el modelado del sistema IoT, justificando la elección del tipo de IoT y exponiendo una breve evolución de estos sistemas, para completar esta parte con el diseño del sistema en base a los requisitos de un hipotético contratista. Una vez diseñado el sistema IoT se procederá a la realización del estudio de las debilidades, amenazas y riesgos, aclarando en primer lugar los conceptos. Después se indicarán los pasos de la guía de buenas prácticas a aplicar para intentar solventar o mitigar algunos de los problemas de seguridad recogidos relativos al sistema IoT modelado. Por último se enumerarán las vulnerabilidades que afecten a los principales módulos o subsistemas del IoT modelado, encontradas en los motores de búsqueda de los sistemas de referencia.

Para acabar, se aportará una valoración global y se plantearán líneas alternativas y futuras de trabajo.

ABSTRACT

The increasing popularity in IoT has clearly exposed the security leakages in most of these systems. Thanks to the current technological evolution, the easiness to implement these systems and its extension, make these security issues even worse, given that they work in increasingly critical systems, and whose operative and integration with other subsystems must be perfectly executed, error-proof and tamper-proof.

The aim of this dissertation is to show up the importance of security in IoT systems, for which it will be explained in a specific case which problems or security issues the system has and that should be restrained and mitigated by following certain good practices, by means of some additional tools, and if needed and the circumstances permit it, by redesigning and changing or repeating the implementation.

For it, firstly we will explain in detail the concept of IoT, the state of art and technological evolution, try to show up the importance of security in these systems, then all types of IoT will be listed. We will then proceed with the modeling of the IoT system, after that we will justify the election of IoT, and present a brief evolution in these type of IoT; and to finish this part the design of the system will be explained, using the user requirements, established by an hypothetical contractor. Once the design of the IoT system is done, we will proceed with the study of weaknesses, risks and threats, clarifying the concepts in first place. After that, we will present the guide of good practices to follow in order to solve or mitigate some of the security problems of the modeled IoT system. By last, we will list the vulnerabilities affecting the main modules or subsystems of the modeled IoT system, found in the reference systems search engines.

To finish, we will bring out an assessment of the present dissertation and some alternative and future topics to develop.

Contenido

1. PLAN DE TRABAJO: Objetivos, metodología y planificación temporal

1.1. Estado del arte y justificación

Es imposible obviar a día de hoy la proliferación de dispositivos cotidianos con capacidad para conectarse a internet, y a formar parte de un entorno hiperdigitalizado y conectado entre sí. Estos dispositivos o cosas, con una mínima intervención humana, son capaces de realizar acciones y tomar decisiones en base a información adquirida y compartida en estas redes de cosas. Ver [1], [2] y [4].

A medida que se implantan estos entornos, es muy importante evitar fugas de la información compartida y por supuesto garantizar que la toma de decisiones no se vea comprometida por agentes externos maliciosos, cuya intervención podría provocar desde un malfuncionamiento en un electrodoméstico, hasta pérdidas económicas significativas [8].

Más allá del propio entorno IoT, si éste forma parte de un sistema mayor y está poco o mal protegido, puede comprometer la integridad y la seguridad de todo el conjunto. Consultar [9], [10] y [11] para obtener más información sobre los peligros de un entorno IoT poco protegido.

Es por tanto primordial proporcionar las salvaguardas necesarias para proteger los puntos débiles de estos ecosistemas o llegado el caso implementar protocolos de mitigación ante ataques [8].

Actualmente, se están creando multitud de soluciones y protocolos para fortalecer la seguridad en estos sistemas, ya sea para mejorar la seguridad de las comunicaciones entre los objetos de la red [3], mejorar la protección con respecto a ataques externos [12], o bien para detectar de forma eficiente las debilidades del sistema IoT [13].

Cualquier mejora de la protección de estos sistemas, cada vez más habituales en cualquier organización, será más que deseable, y probablemente moverá (ya lo está haciendo) un mercado económico importante. De acuerdo con Rajesh Muru, analista principal de Global Data, "*el avance de la nube en la empresa y la explosión del Internet de las Cosas (IoT, de sus siglas inglesas) fomentarán que haya más adquisiciones*" [14].

1.2. Objetivos

El objetivo principal del presente trabajo es poner de manifiesto la importancia de la seguridad en las redes de cosas, evidenciando el riesgo existente de sufrir ciberataques, fruto de las vulnerabilidades que pueden presentar estos ecosistemas y proponiendo mecanismos para su mitigación.

Para poder cumplir el objetivo marcado, se seguirá una hoja de ruta con los siguientes hitos a cumplir, que servirán para comprender mejor la finalidad de este trabajo:

- Explicar el concepto de IoT, y por qué es importante hoy en día.
- Modelar un sistema IoT real.
- Realizar un estudio de los riesgos y amenazas que pueden sufrir las IoT.
- Elaborar una guía sobre las mejores prácticas para mitigar dichos riesgos y amenazas.
- Realizar un estudio en profundidad de las vulnerabilidades presentes en los sistemas IoT.
- Adquirir una perspectiva global de la importancia de la seguridad en los sistemas IoT.

1.3. Metodología

La realización del presente trabajo servirá a los objetivos establecidos en el apartado anterior, y se dividirá en las siguientes etapas o apartados:

1.3.1. Definición del plan de trabajo

En primer lugar se establece el contexto y se evidencia la importancia de la materia sobre la que se desarrolla el presente trabajo, justificando su realización. Se definen los objetivos a cumplir y la metodología de trabajo a seguir. A continuación se realiza la planificación temporal y se especifican las correspondientes entregas.

1.3.2. Explicación de IoT

Se definirá el concepto de IoT, las posibles tipologías existentes y se demostrará cómo y a qué niveles puede facilitarles la vida a las personas actualmente.

1.3.3. Modelado de sistema IoT

Se establecerá como modelo de referencia para este trabajo un sistema IoT realista y que esté en uso a día de hoy. Se podrán añadir más modelos para ejemplificar tipologías diferentes de sistemas IoT. En base a este sistema se podrá profundizar más fácilmente en las posibles amenazas a las que pueden verse sometidos estos sistemas e incluso realizar una clasificación de las mismas.

1.3.4. Estudio de los posibles riesgos y amenazas inherentes un sistema IoT

Dado un determinado sistema IoT con una tipología determinada, se realizará una revisión o recopilación de todas las posibles amenazas y los riesgos que pueda sufrir dicho sistema y se analizarán las causas. Para tal efecto se podrá utilizar la información contenida en la organización OWASP IoT [6].

1.3.5. Guía de prácticas recomendadas para mitigar los riesgos de un sistema IoT

Se realizará un resumen con todas las prácticas orientadas a mitigar o evitar los riesgos y amenazas recopiladas en la fase anterior.

1.3.6. Estudio detallado de vulnerabilidades asociadas a un IoT determinado

Partiendo del sistema IoT modelado, se especificarán y tipificarán las vulnerabilidades existentes (usando estándares de clasificación de vulnerabilidades) asociadas al mismo, en base a todos los elementos que conformen dicho IoT, su tipología y su finalidad.

1.3.7. Conclusiones

Por último se recogen todas las conclusiones extraídas a lo largo de la realización del trabajo y que servirán para aportar una visión global del problema abordado y también para indicar vías futuras de trabajo en este campo.

1.4. Planificación

A continuación se recoge la planificación temporal del trabajo a realizar mediante un listado de tareas, y su correspondiente diagrama temporal, conocido como diagrama de Gantt.

1.4.1. Listado de tareas

A continuación se desglosan las tareas a desempeñar, necesarias para la realización de este trabajo:

1.4.1.1. Documentación

Se realiza una investigación acerca de la materia tratada en el trabajo, para poder adquirir todos los conocimientos requeridos y recabar la información necesaria, focalizando la investigación en los matices correspondientes al enfoque que se quiera dar. En este caso, se busca información relacionada con la seguridad en las IoT, y más concretamente, la gestión de amenazas, riesgos y vulnerabilidades. Esta será una tarea que se prolongará en el tiempo prácticamente hasta la entrega.

1.4.1.2. Planteamiento de la estrategia

Se concreta el objetivo principal y las ramificaciones y matices que pueda presentar el tema a desarrollar. Esto permitirá establecer la dirección hacia la que se desea orientar el presente trabajo.

1.4.1.3. Plan de trabajo

Estado del arte y justificación: Se expone la importancia de la Seguridad en IoT a día de hoy, tema en el que se basa el trabajo a realizar.

Objetivos: Se explica lo que pretende cumplirse con la realización del trabajo.

Metodología: Se describen los pasos a dar para la realización del trabajo, concretamente lo que podrían ser los capítulos o los apartados de que se compondrá el documento.

Planificación: Se enumeran las tareas a realizar y se referencian a un calendario, con una fecha de inicio y una duración determinada. Por claridad se incluye un diagrama de Gantt.

1.4.1.4. Hito: Entrega 1

Preparación de la entrega 1: Plan de trabajo.

1.4.1.5. Definición de IoT

Concepto: Se explica el concepto de Internet of Things.

Tipología: Se realiza una clasificación de sistemas IoT en base a la finalidad o a la operativa del sistema.

Explicación de la importancia de la seguridad de IoT: Exposición de lo importante que es la seguridad en sistemas IoT. Definición de amenaza, riesgo y vulnerabilidad, y sus efectos sobre estos sistemas.

1.4.1.6. Modelado de IoT

Sistemas IoT específicos: Se enumerarán detalladamente casos concretos de sistemas IoT, y se especificará su finalidad y su composición.

Selección de sistema IoT: Para concretar la búsqueda de amenazas, riesgos y vulnerabilidades se elige uno de los sistemas que se han utilizado a modo de ejemplo.

1.4.1.7. Revisión del trabajo tras corrección Entrega 1

Se revisará el trabajo realizado hasta el momento en base al *feedback* recibido de la corrección de la Entrega 1.

1.4.1.8. Hito: Entrega 2

Descripción detallada del sistema estudiado y conjunto de hipótesis consideradas hasta el momento Se han de sustentar también en los resultados de los experimentos que se hayan realizado

1.4.1.9. Realización de estudio para encontrar amenazas y riesgos

Análisis de características: Se hará un análisis específico del sistema seleccionado, explicando su composición y resaltando aquellos aspectos importantes a considerar a la hora de encontrar amenazas, riesgos y vulnerabilidades.

Elaboración del listado de riesgos y amenazas: De acuerdo al sistema modelado, y a las características de cada elemento que lo conforme, se elabora una lista con los riesgos y amenazas que puedan afectar al sistema descrito.

Análisis de las casuísticas posibles: Completar el listado con la casuística o pasos que permitiría a un atacante aprovechar los riesgos descritos.

1.4.1.10. Elaboración de guía de mejores prácticas para proteger el sistema IoT

Se realiza un listado de las prácticas que permitirían proteger el sistema IoT de un atacante. Se tendrán en cuenta los pasos que realizaría el atacante; las prácticas recogidas estarán orientadas a imposibilitar alguno de estos pasos que tendría que realizar el atacante.

1.4.1.11. Revisión del trabajo tras corrección Entrega 2

Se revisará el trabajo realizado hasta el momento en base al *feedback* recibido de la corrección de la Entrega 2.

1.4.1.12. Hito: Entrega 3

Primera versión del desarrollo completo del trabajo, a falta del estudio de vulnerabilidades.

1.4.1.13. Realización de estudio de vulnerabilidades

Selección de sistema de clasificación de vulnerabilidades: Selección de uno de los sistemas de clasificación existentes (CVE, Bugtraq, etcétera).

Identificación de los elementos de la IoT: Identificar los elementos y/o características de la IoT que puedan presentar vulnerabilidades.

Listado de vulnerabilidades: Asociar las vulnerabilidades encontradas con cada elemento de la IoT.

Análisis del listado obtenido: Se analizarán las vulnerabilidades del listado, exponiendo las conclusiones alcanzadas en base a esta información.

1.4.1.14. Conclusiones y trabajo futuro

Se recompilan todas las impresiones y la perspectiva global de la problemática tratada y se plantean vías de trabajo futuras, en base a la evolución presente en este tipo de tecnología.

1.4.1.15. Revisión completa del trabajo

Se hace una revisión exhaustiva del trabajo realizado hasta el momento, teniendo en cuenta todas las correcciones y sugerencias, y en caso de que fuese necesario, la modificación o replanteo de alguna de las fases que lo conforman.

Si es posible se tendrá en cuenta el *feedback* tras la corrección de la Entrega 3.

1.4.1.16. Hito: Entrega 4 – Memoria final

Se entrega el documento completo, incluyendo todas las revisiones.

1.4.1.17. Preparación del vídeo de presentación

Confección de la presentación: Se elaborará una presentación de diapositivas, que se mostrará durante el vídeo de presentación.

Preparación del guión de la presentación: Se elaborará un guion que permitirá al alumno efectuar la presentación de forma eficiente y que no exceda del tiempo indicado.

Realización del vídeo: Se efectuarán los ensayos necesarios para obtener una presentación con la máxima calidad.

1.4.1.18. Hito: Entrega del vídeo de presentación

Se entrega el vídeo de presentación resultante.

1.4.1.19. Hito: Defensa del Trabajo Final de Máster

Periodo durante el que el tribunal plantea preguntas sobre el trabajo realizado, y el alumno ha de responderlas.

1.4.2. Relación temporal de las tareas y diagrama de Gantt

Se utiliza la aplicación *GanttProject* [7].

1.4.2.1. Relación temporal de las tareas

Nombre	Fecha de inicio	Fecha de fin	Duración
Trabajo Final de Máster	24/2/20	19/6/20	117
Fase de documentación	24/2/20	23/4/20	60
Planteamiento de la estrategia	24/2/20	1/3/20	7
Plan de trabajo	25/2/20	2/3/20	7
Estado del arte y justificación	25/2/20	27/2/20	3
Objetivos	25/2/20	28/2/20	4
Metodología	27/2/20	1/3/20	4
Planificación temporal	29/2/20	2/3/20	3
Hito: Entrega 1	3/3/20	3/3/20	0
Definición de IoT	3/3/20	14/3/20	12
Concepto	3/3/20	7/3/20	5
Importancia de la seguridad en IoT	8/3/20	14/3/20	7
Tipología	10/3/20	14/3/20	5
Modelado de IoT	15/3/20	26/3/20	12
Sistemas IoT específicos	15/3/20	22/3/20	8
Selección de sistema IoT	23/3/20	26/3/20	4
Revisión tras feedback de Entrega 1	27/3/20	30/3/20	4
Hito: Entrega 2	31/3/20	31/3/20	0
Realización de estudio para encontrar riesgos y amenazas	31/3/20	13/4/20	14
Análisis de características de sistema IoT	31/3/20	5/4/20	6
Elaboración de listado de riesgos y amenazas	6/4/20	9/4/20	4
Análisis de las casuísticas posibles	10/4/20	13/4/20	4
Elaboración de guía de mejores prácticas	14/4/20	25/4/20	12
Revisión tras feedback de Entrega 2	26/4/20	27/4/20	2
Hito: Entrega 3	28/4/20	28/4/20	0
Realización de estudio de vulnerabilidades	28/4/20	17/5/20	20
Selección de sistema de clasificación de vulnerabilida...	28/4/20	1/5/20	4
Identificación de los elementos de la IoT	2/5/20	7/5/20	6
Listado de vulnerabilidades	8/5/20	13/5/20	6
Ordenación de las vulnerabilidades encontradas	14/5/20	17/5/20	4
Conclusiones y trabajo futuro	18/5/20	27/5/20	10
Revisión final	28/5/20	1/6/20	5
Entrega TFM	2/6/20	2/6/20	0
Preparación del vídeo de presentación	3/6/20	8/6/20	6
Confección de la presentación de diapositivas	3/6/20	5/6/20	3
Preparación de guión	6/6/20	7/6/20	2
Realización del vídeo	8/6/20	8/6/20	1
Hito: Entrega del vídeo de presentación	9/6/20	9/6/20	0
Defensa del TFM	15/6/20	19/6/20	5

Figura 1: Listado de tareas con sus respectivas fechas de inicio y fin y duración

1.4.2.2. Diagrama de Gantt

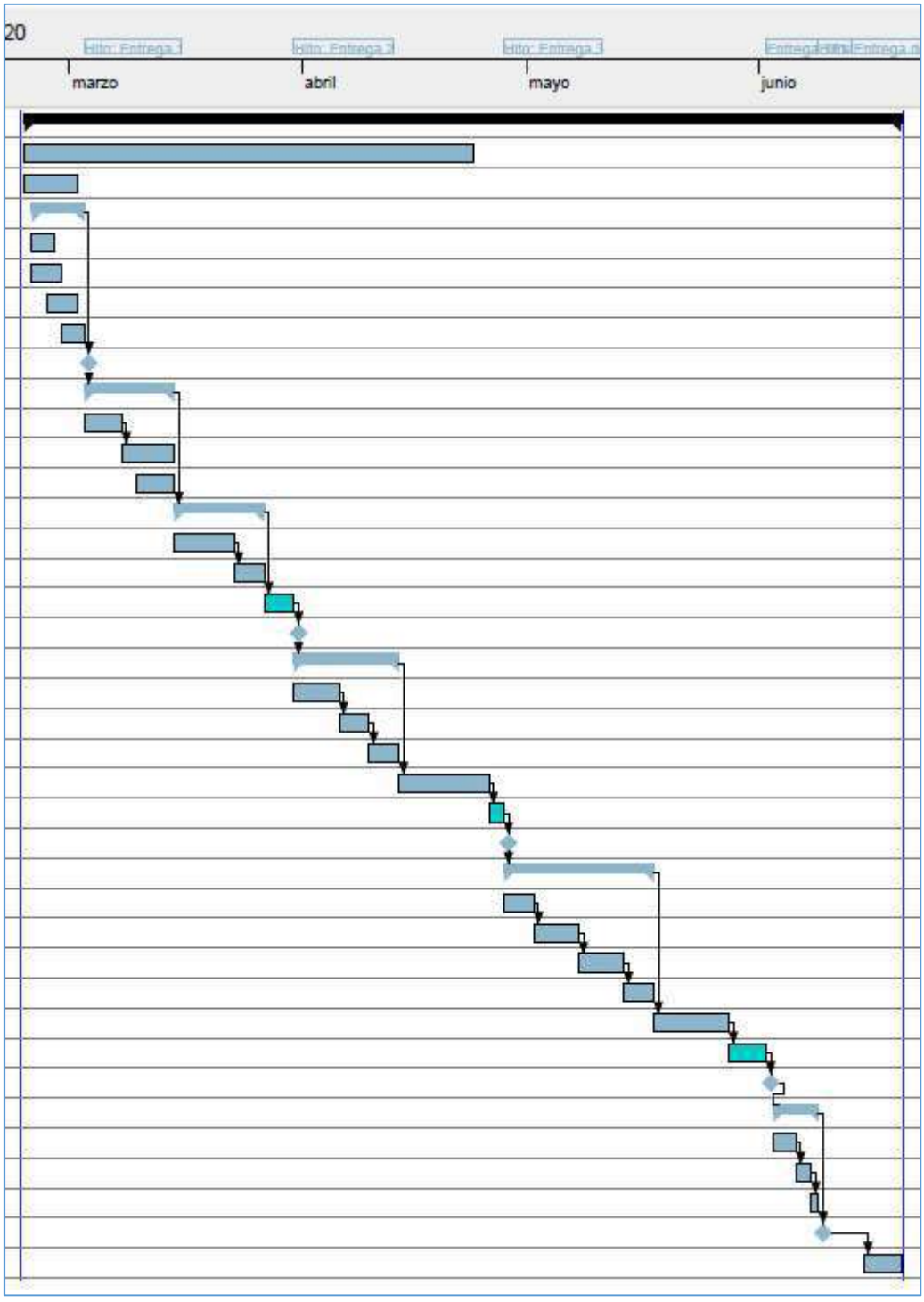

Figura 2: Diagrama de Gantt

Se marcan en verde claro las tareas de revisión del documento.

2. DEFINICIÓN DE IOT

2.1. Concepto de IoT

¿Qué es IoT o "Internet of Things"? De acuerdo con [1] es la interconexión de objetos a través de una red, siendo dicha red la formada por dichos dispositivos dentro de un sistema más complejo, o bien conectada a Internet.

Dentro de una IoT existen dos roles principalmente, los sensores, y los actuadores. Los sensores son aquellos dispositivos que adquieren datos; mientras que los actuadores son aquellos dispositivos que de acuerdo a los datos adquiridos por los sensores, ejecutan una acción o toman una decisión. Puede haber dispositivos que cumplan con ambas operativas.

De acuerdo con la anterior definición, podría considerarse como IoT cualquier sistema formado por dos o más dispositivos con capacidad para comunicarse entre sí o a conectarse a Internet y que desempeñasen una función, desde sensores, *smartwatches*, hasta PCs convencionales o servidores, pasando por lavadoras, termostatos controlables telemáticamente [15] o incluso bombillas inteligentes [16].

2.2. Evolución tecnológica hacia sistemas IoT

2.2.1. Tecnología System on Chip

Gracias a la evolución en la electrónica y a la reducción del tamaño de los circuitos integrados o chips, es posible encontrase con todas o prácticamente todas las características de un PC convencional (a menor escala en varios niveles) en una sola placa (SoC o *System on Chip* [1] y [4]), de modo se hace más sencillo incrustar este tipo de tecnología en dispositivos convencionales cuando hace unos años era impensable [17]**.** Un claro ejemplo serían los *smartwatches* o *smartbands*. Estos dispositivos incorporan una electrónica cuyas dimensiones encajan perfectamente en el hueco destinado antaño al mecanismo de un reloj de pulsera; ahora capaz de conectarse con un *Smartphone* y acceder e informar de las notificaciones de las aplicaciones de mensajería, llamadas, etcétera. Este sería un ejemplo de IoT, ya que lo compondrían dos elementos, el *smartwatch* y el *Smartphone*, del denominado grupo de soluciones IoT *Wearables*, tal y como se podrá contemplar en el apartado 2.5.7.

Figura 3: Ejemplo de IoT con smartphone y smartwatch [18].

2.2.2. Éxito y evolución de los Smartphones

Además, gracias al uso extendido y la evolución de los *smartphones*, multitud de fabricantes están produciendo la tecnología necesaria, concretamente CPUs y GPUs cada vez más pequeñas y eficientes, orientadas a dar un rendimiento cada vez mayor minimizando el consumo [19][20].

2.2.3. Redes inalámbricas

Se ha de tener en cuenta asimismo la evolución de los módulos de comunicación inalámbricos (antenas, módems y módulos Wifi) y el avance de las telecomunicaciones hacia redes de alta velocidad; la red 4G ya está mayoritariamente implantada a nivel mundial [21] pero el 5G es una realidad y existen operadores como Vodafone que ya lo están ofreciendo a sus clientes [22].

2.2.4. Plataformas Do It Yourself

Por último no se debe obviar el éxito en los últimos años de plataformas como *Arduino* y *Raspberry* (por mencionar dos de las más conocidas), que evidencian la importancia de la filosofía DIY (*Do It Yourself*) con una comunidad de usuarios muy extensa, y con infinidad de proyectos que van desde la construcción de un robot rastreador [23], hasta la creación de una estación meteorológica [24].

2.2.5. Herramientas e IoT en la actualidad

En resumidas cuentas, se puede afirmar que actualmente existen herramientas de fácil acceso, a través de las que se podrían crear un amplio abanico de sistemas IoT con dispositivos basados en soluciones *SoC*, cada vez más sofisticados y con funcionalidades más amplias en un formato más compacto, y además a precios económicos. Concretamente, a fecha de la realización de este trabajo, el precio del *Arduino Nano 33 IOT* es de 16.00€, siendo éste uno de los modelos más simples de la familia *Arduino*, equipado con:

- Procesador ARM Cortex de baja potencia: microprocesador ARM de 32 bits a 48 MHz y bus matricial de alta velocidad [25].
- Módulo de radio *u-blox:* módulo de comunicaciones de un fabricante fiable y bastante extendido actualmente, sobre todo en módulos GPS [26] [27].
- Elemento seguro de cripto-autenticación ATECC608A: este módulo permitiría entre otras funcionalidades, gestionar y almacenar las claves de autenticación de una red IOT, cifrar pequeños mensajes, arranque seguro y descarga de datos de forma protegida, y el control anticlonado [28].

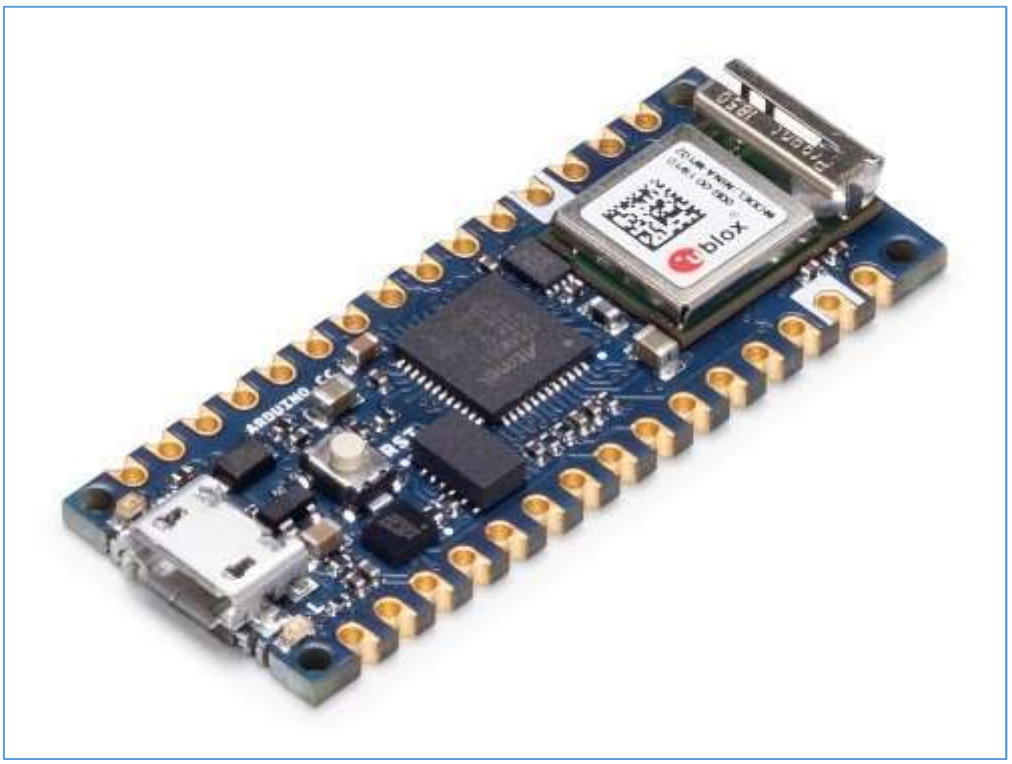

Figura 4: Arduino Nano 33 IOT.

Para ilustrar la relevancia y éxito de estos sistemas, basta acceder a la web de proyectos IOT de *Arduino* para comprobar las soluciones IoT que está creando la comunidad de usuarios [29]. De entre todos, merece la pena comentar, para ejemplificar la rapidez y aparente facilidad con la que estos proyectos pueden crearse y por la utilidad que pueden aportar, el proyecto de monitorización de la información del avance del virus COVID-19 en los distintos países, obtenida en tiempo real de la web https://www.worldometers.info/coronavirus [30]:

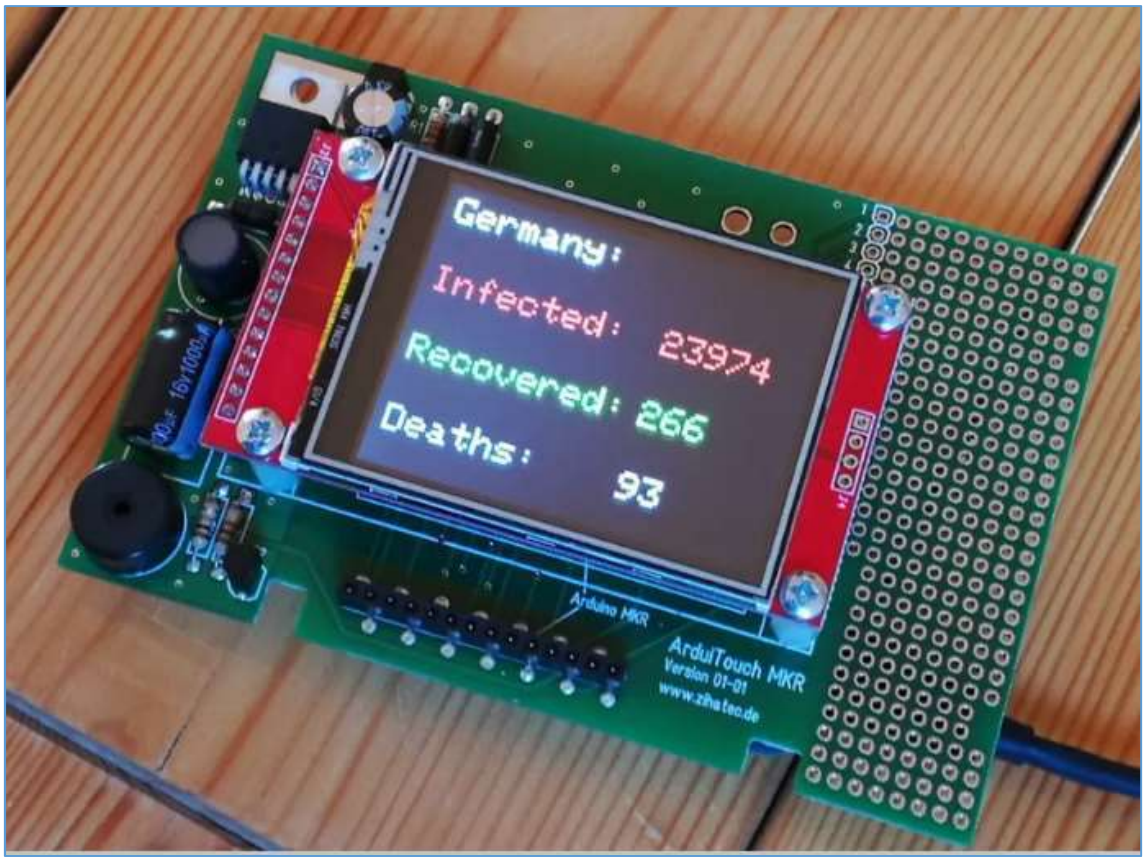

Figura 5: Ejemplo de proyecto IoT con Arduino (23/03/2020).

Por último, cabe mencionar que en los sistemas IoT, puesto que montan dos o más dispositivos, se tenderá a abaratar costes, utilizando para ello equipos más sencillos, más eficientes, y también más baratos, además siempre tendiendo a minimizar la huella de carbono [31]. Esto es y será cada vez más sencillo de realizar, -y además usando equipos con más prestaciones-, gracias a la evolución tecnológica comentada.

2.2.6. Incremento del número de dispositivos IoT

Diversas fuentes ([32] y [33]) dan fe del gran incremento y la tendencia de estos sistemas, y de los dispositivos que la componen, así como del importante mercado económico que mueven.

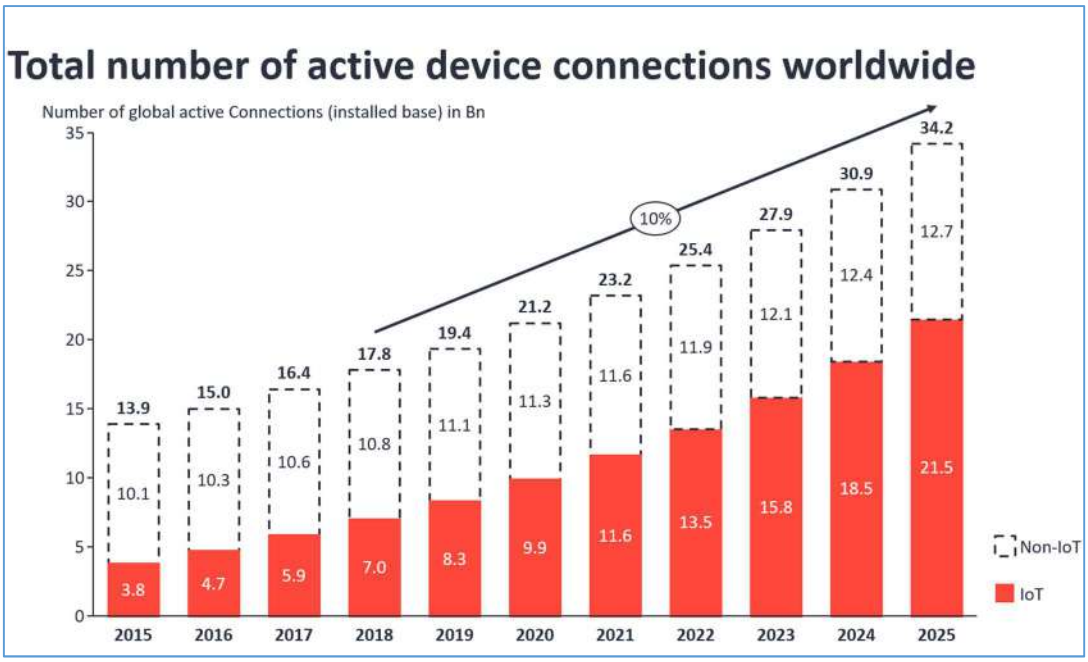

Figura 6: Evolución y previsión del incremento del número de dispositivos IoT.

Tal y como se observa en la gráfica anterior, se contemplaba en 2018 (año al que pertenece este estudio) una cifra en 2020 de 9,9 mil millones de dispositivos IoT en operación. No obstante, en [33], se demuestra que el anterior estudio ha sido quizás demasiado conservador, puesto que actualmente y a lo largo del presente año 2020 se prevé que haya instalados un total de 30 mil millones de dispositivos[1].

2.3. Características principales de un sistema IoT

2.3.1. Comunicación entre nodos

Aunque es deseable que un elemento de un sistema IoT posea propiedades como la eficiencia (para minimizar la huella de carbono principalmente), y por tanto un tamaño reducido, la principal característica de los elementos que forman parte de la IoT, de acuerdo con [1] y [4], es la capacidad para comunicarse con otro elemento, bien sea directamente, o a través de Internet. Es decir, es primordial para que exista red y por tanto IoT, que los nodos puedan comunicarse entre sí.

Para ello, cada nodo debe disponer del hardware necesario (tarjeta de *Ethernet*, tarjeta *WiFi*, antena *Bluetooth*, puerto RS232, etcétera), el *middleware* o *firmware* para poder manejar este *hardware* (el *driver* de sistema operativo o los controladores correspondientes y el servicio o demonio que permita interactuar con el *driver o* controlador) y también el *software* que implemente el método de comunicación o protocolo.

Existen multitud de implementaciones de protocolos de comunicación entre nodos IoT, no existe actualmente un estándar. Aunque entre los más populares se podrían mencionar Sigfox, MQTT, CoAP o ZeroMQ [4] [34].

2.3.2. ID de red

Para que un dispositivo esté dotado de dicha capacidad, es indispensable que también sea posible distinguirlo de entre los demás nodos de la red, es decir, que disponga de un ID o

[1] Si se consulta la bibliografía, ésta especifica las cifras en billones. Al ser la escala estadounidense, se ha de dividir entre 1000 para obtener la magnitud correcta, puesto que 1000 millones = 1 billón en EEUU.

identificador único. De este modo, se podrá ubicar dentro de la red y establecer una comunicación bidireccional con éste. Dependiendo de la tecnología y protocolos de comunicación utilizados, este ID será de una naturaleza o de otra. Por ejemplo, si la IoT se establece entre dispositivos que se comunican a través de conexiones TCP a través de redes inalámbricas (WiFI o datos móviles) o de conexiones Ethernet cableadas, el ID sería la dirección IP de las tarjetas de red de cada elemento; si por el contrario se emplea alguna tecnología basada en Bluetooth, el ID sería el UUID (*Universally Unique Identifier*) del servicio que se desee utilizar [35].

2.3.3. *Gestión de datos*

Por otro lado y también de gran importancia, un elemento de un sistema IoT debe poder tener la capacidad de adquisición y tratamiento de datos, para poder transferirlos al resto de nodos de la red en el formato adecuado. No se está haciendo referencia a la capacidad de cómputo (que debido a la naturaleza de estos dispositivos puede llegar a ser limitada), si no a la obtención de los datos y a su redistribución en la red. Por ejemplo, en un sistema IoT que implemente una estación meteorológica, es crucial poder acceder a los datos de los sensores, puesto que sin ellos no se pueden realizar los diagnósticos correspondientes y la estación no cumpliría su función.

2.3.4. *Capacidad de* procesamiento

Tal y como se ha mencionado hasta este punto los sistemas IoT están formados por elementos que permiten la adquisición de datos y su transmisión en la red. No existe ningún requisito que obligue a estos nodos a disponer de una capacidad de procesamiento determinada, no obstante, es posible que en ciertos sistemas ésta sea necesaria y no pueda obtenerse de elementos más limitados y que haya que contar con un nodo o nodos principales dotados con un hardware más potente, que le permita al sistema realizar operaciones más complicadas con los datos, tomar decisiones en base a dichas operaciones, o distribuir tareas entre los nodos de la red. Es posible por tanto encontrarse con sistemas IoT formados por nodos desiguales en cuanto a su capacidad de computación; normalmente existirá un nodo principal más potente, y los demás, con un hardware más sencillo y por tanto menor capacidad. Dependiendo de la criticidad del sistema, podrá haber un solo nodo principal, o varios para garantizar redundancia. Existen otras alternativas, basadas en el *cloud computing* que consisten básicamente en poder emplear recursos de computación distribuidas [36] (almacenamiento de datos, o capacidad de procesamiento) accesibles en servidores o centros de datos a través de Internet.

Un ejemplo de estos sistemas con nodos desiguales serían las redes neuronales, con nodos sencillos que se encargan de capturar y distribuir muestras de datos por la red, y nodos más potentes, que se encargan de procesar dichas muestras y tomar una decisión o producir un dato de salida [37].

2.4. Importancia de la seguridad en sistemas IoT

Dada la expansión y popularidad de los dispositivos IoT y su conectividad, suponen una víctima potencial de cibercriminales [38], que intentarán aprovechar las vulnerabilidades de estos sistemas para perpetrar sus ataques. Es importante proteger tanto el sistema IoT como cada nodo de forma individual, ya que de lo contrario, un fallo de seguridad en uno de estos dispositivos, podría comprometer la seguridad de la red, o incluso de un sistema mayor en el que pueda estar operando dicha IoT.

Estos serían los vectores de ataque más comunes utilizados por los ciberdelincuentes [38]:

- *Ataques de fuerza bruta*: Se pueden usar para obtener contraseñas, descubrir puertos abiertos (por ejemplo usando la aplicación telnet), o incluso el *login* y *password* de uno de los nodos para acceder a la red.
- *Ataques de denegación de servicio (DDoS)*: Ataques que inhabilitan el sistema y le impiden realizar su función. Pongamos que un atacante conoce o adivina una secuencia de acciones (por ejemplo, una secuencia de voltajes en su módulo de alimentación) en un sensor que provoca que deje de poder tomar muestras; esto va a impedir que el sistema al que pertenece ese sensor pueda realizar su función. Por ejemplo, en un sistema de ayuda a la conducción en un vehículo en el que se toman muestras a través de una cámara que captura las imágenes de señales del tráfico. Si se inhabilita dicha cámara o se provoca un malfuncionamiento, el sistema de ayuda a la conducción no operará correctamente, incluso podría informar de señales con la velocidad máxima incorrecta, pudiendo ocasionar que el conductor incremente la velocidad del vehículo en una zona que está limitada a menor velocidad, llegando a provocar que el conductor cometa una infracción y se vea obligado a pagar una multa por exceso de velocidad.
- *Acceso a uno de los nodos de la red*: El atacante, consciente de que la seguridad desde el exterior es probablemente más robusta que desde el interior, podría encontrar un punto o un nodo más débil con menos protección, acceder a dicho nodo para entrar en la red y una vez dentro realizar el ataque al nodo o al subsistema que desee. En este tipo de ataques, también puede emplearse uno de estos nodos, como una etapa más (trampolín) para cubrir el rastro del atacante, que puede estar perpetrando un ciberataque a otro sistema y necesita ocultar su identidad y evitar que su IP pueda ser rastreada realizando conexiones sucesivas a varios nodos de distintas redes. Normalmente en estos casos el atacante también combinará técnicas de fuerza bruta, para poder irse conectando a los diferentes nodos de la red.
- *Obtención de datos*: El atacante accederá al sistema o al dispositivo IoT para obtener información y revelarla o entregársela a un tercero, bien sea de los datos personales de los usuarios del sistema (*logins*, *passwords*, números de tarjetas de crédito, números de cuenta bancaria, etcétera), de la organización a la que pertenece la IoT o del propio sistema.

Es muy probable que para efectuar uno de estos ataques, el cibercriminal tenga que combinar técnicas, o aprovecharse de varias debilidades a la vez. Es por tanto primordial proteger el sistema, ya que aunque simplemente se esté fortaleciendo uno de dichos puntos débiles, se podrían evitar varias tipologías de ataques con relativamente poco esfuerzo.

En resumidas cuentas, y tal y como se menciona en [8] y [38], es evidente que el éxito de las tecnologías IoT y su uso cada vez más extendido, tiene también sus consecuencias negativas, ya que al adquirir más relevancia, también son más importantes los proyectos para los que se implementan estos sistemas, y por tanto los datos que procesan son más sensibles, pudiendo provocar pérdidas económicas o revelación de datos si no se protegen adecuadamente.

Además, de acuerdo con [4], pese al incremento en el uso y popularidad de los sistemas IoT muchas entidades o administraciones aún son reticentes a instalar estos sistemas precisamente porque si no se establecen mecanismos de seguridad pueden presentar debilidades que podría aprovechar un atacante.

Es por ello de vital importancia proporcionar mecanismos que permitan mitigar los riesgos de seguridad y fortalecer los puntos débiles de estos ecosistemas. Habría que proteger el sistema ante posibles ataques reduciendo el riesgo, y también preparar protocolos de actuación o

cortafuegos para minimizar daños si estos ataques finalmente se producen. Es decir, prevención y mitigación.

2.5. Tipología de Sistemas IoT

Existen multitud de aplicaciones o proyectos en los que se utilizan tecnologías IoT. De acuerdo con la naturaleza de estos proyectos, puede realizarse una clasificación que permita agrupar estos sistemas de acuerdo a un criterio de funcionalidad o finalidad. Partiendo de la tipología establecida en [4] y de las *Key IoT Verticals* de [39], podría establecerse la siguiente clasificación.

Figura 7: Relación entre los tipos de IoT y las tecnologías de comunicación aplicables.

2.5.1. IoT Industrial (IIoT – Industrial Internet of Things)

Son las soluciones IoT que se engloban o que sirven en proyectos destinados a un entorno industrial. Está muy vinculado al desarrollo de la denominada Industria 4.0 [40] ya que ésta se basa en el uso de nuevas tecnologías como la Inteligencia Artificial, *Machine Learning*, gestión de Big Data o Ciberseguridad, entre otras. Muchas de estas tecnologías pueden servirse de sistemas IoT, empleados para controlar y optimizar los procesos productivos de una fábrica de automoción, gestionar el estado de los procesos cuyos datos se obtienen en base a una red de sensores en una depuradora, o monitorizar y controlar el estado y producción energética de manera telemática de una planta eólica, por mencionar algunos ejemplos.

2.5.2. IoT Militar (IoMT – Internet of Military Things)

Son las soluciones IoT empleadas en proyectos de índole militar (ya sea de defensa o inteligencia militar, u orientado a conflictos armados), como es el caso de aviones con armamento no tripulados, misiles controlados telemáticamente, o incluso munición y armaduras inteligentes [74]. Cabe mencionar que muchos de estos proyectos posteriormente son de aplicación industrial o doméstica (muchas de las innovaciones tecnológicas en el ámbito de las telecomunicaciones son de origen militar, como por ejemplo ARPANET, el germen de INTERNET [41]).

En general, en cualquier solución IoT, la seguridad está adquiriendo una importancia determinante, no obstante, en este tipo, la protección de los datos y los procesos que se ven involucrados son de vital importancia, puesto que se podría llegar a ver afectada la integridad física y los derechos y libertades de los ciudadanos. De modo que se ha de minimizar el riesgo ante la pérdida o revelación de datos, potenciar los algoritmos y sistemas de detección de anomalías y proteger con sistemas robustos o incluso experimentales el almacenamiento de datos. Por otro lado, y normalmente con el respaldo de los gobiernos y administraciones, se han de preparar planes de contingencia que permitan minimizar o contrarrestar los daños a estos sistemas si un ciberataque prospera.

2.5.3. Smart meter

Son soluciones IoT que permiten obtener medidas de forma remota, especialmente aquellas relativas a los consumos energéticos, como luz, agua, gas, etcétera [42].

2.5.4. Smart City

Son soluciones IoT orientadas a la gestión de los recursos y servicios de una población. Tales como la gestión eficiente de basuras, la gestión de energía, del agua, de los servicios de transporte, de la gestión del tráfico, servicios sanitarios, etcétera. Es muy complicado encontrarse con casos reales y exitosos en los que se gestionen todos los servicios de una ciudad de este modo; no obstante, sí se están implementando proyectos *Smart city* que permiten gestionar parte de estos recursos y servicios.

Un claro ejemplo sería el proyecto S2CITY (Sistema Inteligente de Servicios al Ciudadano y al Turista) que está actualmente en las primeras fases de implantación en la ciudad española de Valladolid. A través de una tarjeta *Contactless* o de una *app* del *Smartphone* del ciudadano, se podrá gestionar y acceder a diferentes servicios, tales como el pago de la recarga de un vehículo eléctrico, el uso del transporte público (autobuses urbanos), servicio de préstamo de bicicletas, acceso a bibliotecas municipales, o instalaciones deportivas entre otros. También dispone de un módulo destinado a turistas y de un *backend* que da acceso a un portal de atención ciudadana y que además recoge los datos de consumo de estos servicios para retroalimentar el sistema (a través de la gestión inteligente de estos datos con técnicas *Big Data*) y conseguir así la información necesaria para la mejora permanente de los mismos [43] [44].

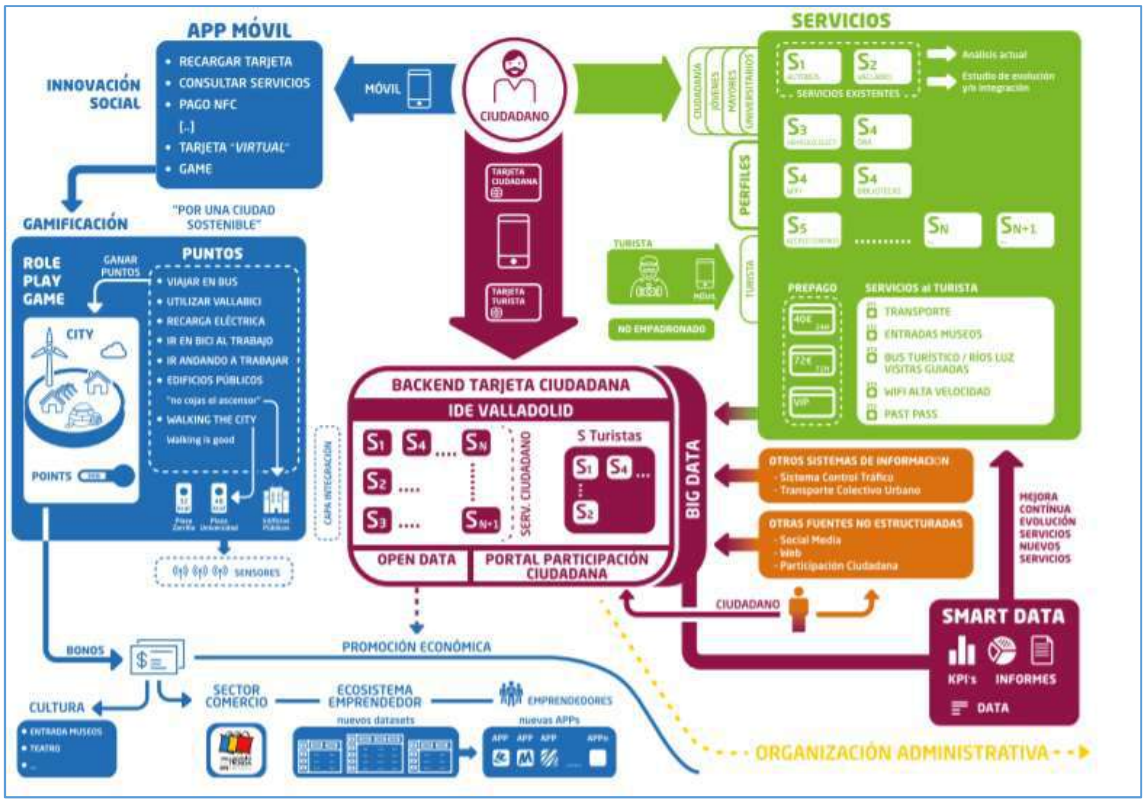

Figura 8: Esquema de implantación del proyecto S2CITY en la ciudad de Valladolid.

2.5.5. Smart building

También conocida como la Inmótica, son soluciones IoT destinadas al equipamiento integral de inmuebles. Permiten gestionar y monitorizar de forma centralizada el funcionamiento de un edificio, concretamente el estado de la estructura, el consumo y eficiencia energéticos, el funcionamiento de ascensores, sistemas de ventilación, calefacción, control de sensores de humos, agua, alarmas de incendios, sensores de movimiento, sistemas de seguridad de acceso al edificio, de video vigilancia, y un largo etcétera. Son quizás de las soluciones IoT más complejas puesto que la integración entre subsistemas tiene que funcionar adecuadamente, y debido a ello el mantenimiento es muy importante, y ha de realizarse regularmente para asegurar el correcto funcionamiento de cada subsistema y del sistema completo [45] [46]. Estas soluciones se pueden encontrar en edificios de administraciones públicas, museos, bibliotecas o instalaciones militares.

2.5.6. Smart Home

También conocida como Domótica, son soluciones IoT destinadas a facilitar tareas domésticas. A través de este tipo de dispositivos IoT, es posible por ejemplo saber qué alimentos faltan en el frigorífico y llegado el caso solicitar la compra de los mismos de forma automática (sin la intervención del inquilino, salvo quizás una autorización para el pago de la compra online). También se puede gestionar y optimizar el consumo de energía; de luz a través de bombillas inteligentes, o enchufes programables, y del gas a través del uso de calderas programables que se conectan a la red WiFi de casa y que se pueden controlar a través de una app del *Smartphone* del inquilino. Son soluciones con gran aceptación actualmente y de uso muy extendido, principalmente por su precio cada vez más económico y porque son fáciles de instalar y de utilizar [47].

2.5.7. Wearables

Son soluciones IoT que permiten el uso de *gadgets* portables con tecnología destinada a facilitar el uso de las comunicaciones o a la realización de tareas cotidianas. El más extendido sin lugar a dudas es el *Smartphone*, y de hecho muchos otros *Wearables* se basan en éste puesto que dispositivos tales como los *smartwatches* o las *smartbands* se conectan mediante *bluetooth* o IrDA (infrarrojos) para obtener y mostrar las notificaciones de llamadas y mensajería recibidas por el *Smartphone*. Existen también otros dispositivos cuya operativa no tiene por qué girar en torno al *Smartphone*, como las gafas de realidad aumentada [48], los relojes GPS, o las *Smart clothes* (o *E-Textiles*) que básicamente son prendas de ropa con tecnología incrustada con diferentes propósitos, entre los cuales, por ejemplo facilitar la mejora del rendimiento de un atleta [49].

2.5.8. Connected Car

Son todas las soluciones IoT instaladas en un vehículo y orientadas a mejorar o facilitar la conducción y a evitar accidentes. Sistemas tales como el *park assist* que le permiten al conductor aparcar el vehículo sin tocar el volante, o los sistemas de aviso como los detectores de fatiga (que emiten un aviso acústico al conductor cuando se detecta que se puede estar quedando dormido), el aviso anticolisión, el detector de velocidad máxima superada o el control de crucero adaptativo son subsistemas que puede llevar instalados un vehículo actualmente y que en la mayoría de los casos tienen como objetivo evitar accidentes. Todo esto se consigue en virtud de multitud de sensores y cámaras que se gestionan desde la centralita del vehículo, cada vez más sofisticada y con más funcionalidades [50].

2.5.9. IoT de Transportes

Derivada de la tipología anterior, se encuentran los sistemas IoT para transportes, es decir, todos aquellos dispositivos orientados a la gestión de tráfico rodado, la gestión de peajes, o la gestión de flotas, por ejemplo de transporte de mercancías, o bien de transporte público ya sea urbano o interurbano.

Respecto a los IoT para gestión de flotas de autobuses, nos podemos encontrar con sistemas embarcados que permiten controlar la localización de cada vehículo y entre otras cosas permiten regular los tiempos de llegada a parada que además se muestran en los paneles situados en las marquesinas de la calle (denominado SAE o Sistema de Ayuda a la Explotación), o permiten la validación y venta de títulos de transporte terrestre (a través por ejemplo del uso de tarjetas *contactless*, del pago en efectivo, códigos QR o incluso del pago con tarjeta bancaria); todo ello respaldado por un sistema de *backoffice* o centro de control que gestiona y centraliza toda esta información y la operativa del servicio. Un ejemplo de ello es el sistema implantado en la ciudad de Rabat (Marruecos) [51]:

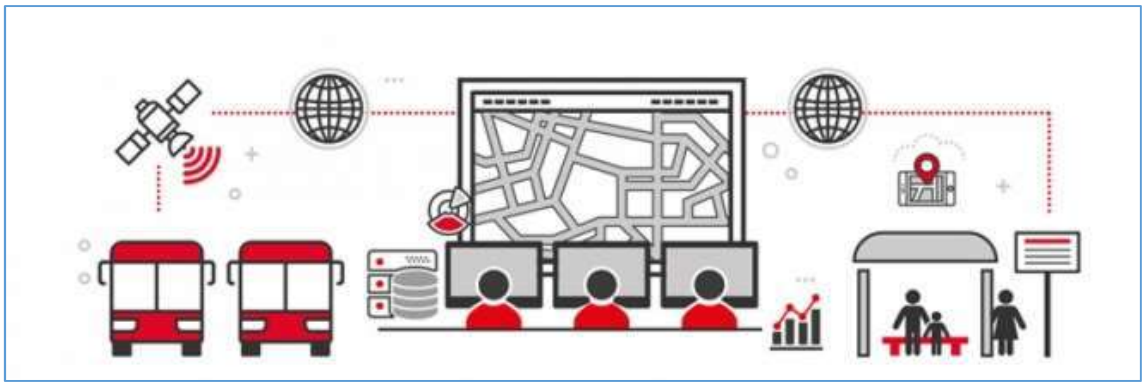

Figura 9: Esquema de funcionamiento de ayuda a la explotación en la ciudad de Rabat (Marruecos).

2.5.10. Connected Health

Son aquellas soluciones IoT orientadas a la medicina y al cuidado de la salud humana. Este tipo de IoT acompaña y evoluciona en paralelo a los avances en medicina; de hecho en algunas ocasiones, estos avances son posibles gracias a la tecnología [52], como por ejemplo los robots empleados para ayudar a los cirujanos en sus operaciones. Muchas otras de estas soluciones, además de acompañar a los avances en medicina y que en definitiva buscan salvar más vidas, también permiten agilizar la gestión relacionada con la atención al paciente (historial clínico, tratamientos activos y anteriores, pruebas realizadas con su correspondiente resultado, informes de urgencias, etcétera) o también la gestión del equipamiento o el material clínico en hospitales o centros de salud [53].

2.5.11. Smart Retail

Son las soluciones IoT destinadas a facilitar y mejorar las experiencias de compra personalizadas en tiendas inteligentes. Esta tecnología permite recopilar la información necesaria de los hábitos y gustos de los compradores, de modo que estas tiendas se alimentan de esa información para poder aplicar estrategias más personalizadas y eficaces, aumentando las ventas, y fidelizando clientes [54].

2.5.12. Logistics & Asset Tracking

Son soluciones IoT orientadas a la gestión de la logística y el seguimiento de la entrega de bienes, es decir, el seguimiento de la cadena de distribución. Consistiría entre otros en adherir o incrustar tecnología (por ejemplo sensores GPS) que permita hacer un seguimiento del bien enviado en tiempo real y obtener información al momento de si algo ha ido mal durante el envío o cuánto le queda al repartidor para realizar la entrega. El uso de esta tecnología le permite a empresas tan populares como *Amazon* hacer envíos ofreciendo un seguimiento prácticamente en tiempo real [55].

2.5.13. Smart Agriculture

Son las soluciones IoT destinadas a mejorar los procesos que tienen lugar en el sector primario, especialmente en la agricultura. Se hace uso de la información proporcionada por sensores de humedad, calidad del agua, composición del suelo, temperatura ambiente, para la toma de decisiones minimizando riesgos (por ejemplo, sembrar según qué cultivos, en qué suelo y en qué cantidades para evitar pérdidas económicas o para maximizar beneficios) [56] [57]. Estas decisiones, dependiendo de la precisión del sistema y de las implicaciones en caso de error, podrían llegar a ser automáticas, sin la intervención del agricultor (por ejemplo, la activación del riego automático dependiendo de la humedad del suelo, la hora del día y la temperatura).

3. MODELADO DE SISTEMA IOT

En los siguientes apartados se analizarán las vulnerabilidades y riesgos inherentes a los sistemas IoT. No obstante, dado el número de diferentes sistemas IoT se ha de seleccionar un tipo de sistema IoT y modelarlo o diseñarlo para poder especificar una lista coherente y no demasiado generalista de riesgos, amenazas y vulnerabilidades.

3.1. Justificación e importancia de la selección y modelado de un sistema IoT

Dada la cantidad ingente existente de sistemas IoT, y sus correspondientes tipologías, para poder hacer un estudio pormenorizado de riesgos, amenazas y vulnerabilidades sin generalizar demasiado, se ha de escoger un tipo concreto de IoT y modelar un sistema específico de forma detallada para poder particularizar las vulnerabilidades asociadas a ese sistema IoT y no a otro.

Se ha de tener en cuenta no solamente la finalidad del IoT seleccionado o su ámbito de operación sino también los elementos que lo componen, y dentro de éstos, qué características tienen, tales como su sistema operativo y la versión de éste, las aplicaciones que corren en él, su operativa normal, etcétera. Se han de conocer todos los detalles posibles para poder determinar de forma precisa las amenazas de seguridad que pudieran materializarse, a nivel particular en cada elemento del sistema y a nivel global, en el sistema completo y los supra sistemas a los que pertenece.

3.2. Selección de tipo de IoT: Sistemas IoT de Transporte

Se seleccionará el tipo de IoT destinado a transportes, por la interesante evolución tecnológica en este sector, y más concretamente en el sector del transporte de personas. No sólo por su evolución, sino por la tendencia de las administraciones a potenciar e incentivar cada vez más el uso del mismo, mientras que en parte debido a la contaminación provocada por el tráfico rodado, se aumentan las restricciones en los vehículos particulares, así como la presión fiscal especialmente en los vehículos diésel, que se ampliará a todos los vehículos de combustión fósil y de aquí a unos años dejará de estar permitida su circulación (entre 2035 y 2050, según el país o la administración [58]). Esto por tanto evidencia la importancia que están adquiriendo estos sistemas, que no solamente pretenden mejorar la calidad del servicio de las empresas de transporte sino también ofrecer una experiencia de usuario que permita cambiar y mejorar el concepto de ir de A á B.

3.3. Importancia y evolución de los sistemas IoT de transporte

3.3.1. Sistemas de Información al Usuario (SIUs)

En pocos años, los Sistemas de Información al Usuario (comúnmente conocidos por SIUs), han pasado de informar de la parada siguiente en un panel de LEDs o dots de plástico, a mostrar en un TFT anuncios de la compañía de transportes o de la administración, intercalando animaciones con audio para anunciar la siguiente parada, y el itinerario que se está haciendo en un sinóptico o incluso en algunos casos en un mapa del trayecto realizado en tiempo real.

En los vehículos de transporte interurbano (largas distancias y por tanto más tiempo sentado en el autobús) este sistema se ha mejorado de manera mucho más evidente. Se ha pasado de instalar uno o dos televisores por vehículo en los que solo el conductor podía elegir el contenido, a disponer cada viajero de una pantalla individual (adosado al asiento delantero) con contenido multimedia a elegir de distinta naturaleza: películas, series, videojuegos, documentales, etcétera.

3.3.2. Sistemas de Ayuda a la Explotación (SAEs)

Respecto a los Sistemas de Ayuda a la Explotación (SAEs [59]), son los que permiten gestionar la flota de transporte y aúnan diferentes tipos de tecnologías, que permiten desde geolocalizar el vehículo y situarlo en un mapa que puede monitorizarse desde un centro de control, hasta gestionar los tiempos estimados de llegada (ETAs, *Estimated Arrival Times*) y el contenido multimedia si es posible, en paneles instalados en las marquesinas de las paradas. En este caso, la evolución ha dependido principalmente de la deriva que han llevado las tecnologías de comunicaciones móviles y su expansión; de modo que hace 20-30 años la tecnología para gestionar las flotas era la radio y los vehículos se identificaban a través del concepto de TDMA (*Time Division Multiple Access* [60]), posteriormente se comenzó a utilizar tecnologías de datos móviles como GPRS y GSM para las llamadas de voz, y actualmente se emplean las redes móviles de alta velocidad (sobre todo 4G, y se comienza a experimentar con el 5G), a través de las que es posible también efectuar llamadas VoIP.

3.3.3. Sistemas de Validación y Venta (SVV)

Por otro lado, y particularizando en los equipos de validación y venta de títulos de transporte, se ha pasado a un siguiente nivel, partiendo desde los sistemas de validación de tarjetas *contactless*. Actualmente se siguen implantando estos sistemas, pero fortaleciendo la seguridad en estos soportes para evitar el fraude, dada la relativa facilidad demostrada para descubrir las claves de una tarjeta Mifare Classic [61], que es una tecnología muy ampliamente utilizada a escala mundial. De hecho, existen en Internet, al alcance de cualquier usuario intermedio, guías muy completas para llevar a cabo este proceso de *pirateo* en cuestión de horas con un PC promedio [62] software gratuito o *crackeado* y un lector NFC de tarjetas *contactless*. Es por esto que se están centrando los esfuerzos en la utilización de soportes más robustos al fraude (Mifare Desfire, Mifare Plus, Calypso o incluso emuladores de tarjetas en el chip NFC de un *Smartphone* [63] y [64]) o en los que se tiene un control muy exhaustivo de la caducidad asumiendo un pequeño margen de fraude (código de barras, QR) o a través de la integración con plataformas y pasarelas de pago o TPV Virtuales (como Redsys [65]) para permitir el pago a bordo con tarjetas *contactless* bancarias a través de la denominada tecnología EMV (lectores *contactless* sin *pinpad*, puesto que el pago del pasaje es inferior a 20 €) [66].

3.3.4. Servicios adicionales

Cabe destacar también que en un intento por mejorar la experiencia del viajero, se ofrecen servicios de diversa índole, como tomas de corriente, tomas de USB o WiFi al pasaje o lo que es lo mismo, WiFi gratuito para que los viajeros se conecten con sus *Smartphones* y puedan navegar por Internet o utilizar las RRSS [67].

3.3.5. Integración entre servicios y sistemas

Tal y como se puede observar, son varios subsistemas, operando a la vez y en paralelo, y con altos niveles de integración entre los mismos. Se ha de tener en cuenta que cada uno también puede corresponder a un proveedor o industrial diferente, de modo que la dificultad de la integración aumenta sensiblemente.

Sistemas tan complejos como éstos tienden a la modularidad para poder cumplir tareas muy definidas pero dado que unos sistemas dependen de otros, las interfaces entre ellos han de estar muy bien diseñadas y ser capaces de ofrecer una comunicación fluida, robusta y segura entre los mismos, pudiendo además ser escalable a otros subsistemas que puedan añadirse a posteriori.

3.4. Diseño de la IoT de Transporte

Para poder diseñar o seleccionar las propiedades y componentes de cada subsistema, se enumerarán los requisitos de alto nivel de nuestro Sistema de Transporte, y a continuación se definirán las características más importantes de cada subsistema, de la forma más detallada posible. De este modo se podrán sentar las bases que permitirán detectar con mayor precisión los riesgos, amenazas y vulnerabilidades del sistema.

3.4.1. Requisitos del IoT de Transporte

Se establecen una serie de requisitos de alto nivel, o lo que es lo mismo, requisitos de usuario que permitirán definir las características del sistema. Solamente se tienen en cuenta aquellos requisitos que permitirán definir a grandes rasgos nuestro sistema, sin profundizar demasiado en aspectos técnicos no relacionados con el tema que nos ocupa.

- RU1. El sistema se implementará para una flota de 100 autobuses.
- RU2. Los autobuses realizarán líneas urbanas, y están equipados para la realización de trayectos cortos.
- RU3. Las cocheras dispondrán de red WiFi para la descarga de actualizaciones del sistema.
- RU4. En cada autobús se tendrá que implementar el siguiente equipamiento:
    - RU4.A. Sistema de información al usuario (SIU) a bordo a través de un TFT cuyo contenido pueda configurarse y controlarse remotamente desde un Centro de Control.
    - RU4.B. Se controlarán los paneles exteriores del autobús que permiten informar a los usuarios que van a viajar de qué trayecto se está realizando.
    - RU4.C. Sistema de megafonía, para el anuncio de paradas y avisos al pasaje.
    - RU4.D. Sistema de ayuda a la explotación (SAE), a través de:
        - RU4.D.1. Un equipo embarcado que permita controlar en tiempo real la posición del vehículo desde un Centro de Control.
        - RU4.D.2. Comunicación entre el Centro de Control y el vehículo a través de llamadas y mensajería.
    - RU4.E. Sistema de Validación y Venta (SVV). Permitirá:
        - RU4.E.1. La emisión de billetes de transporte con pago efectivo en soporte papel con un QR reutilizable.
        - RU4.E.2. Validación y venta de títulos de transporte en soporte *Mifare Classic* claves fijas (tarjeta actual) que podrán ser parametrizables.
        - RU4.E.3. Validación y venta de títulos de transporte en soporte *Mifare Desfire* tipo SAM AV2 sobre soporte físico y también virtualizado en *Android HCE* (tarjeta a implementar).
        - RU4.E.4. Validación de códigos QR tanto de billetes en soporte papel como de eventos especiales, que la empresa de transportes enviará a los usuarios interesados a sus *Smartphones*.
        - RU4.E.5. Envío de las transacciones realizadas en tiempo real al Centro de Control.
    - RU4.F. Consola de conductor con pantalla táctil. Permitirá:
        - RU4.F.1. Iniciar megafonías o anuncios al pasaje.
        - RU4.F.2. Interactuar con el Centro de Control a través del SAE para la solicitud de llamadas, envío de mensajería, etcétera.
        - RU4.F.3. Controlar el SVV (apertura de servicio, emisión de billetes en efectivo, anulaciones, consultas de movimientos, etcétera).

    - RU4.G. WiFi al pasaje.
- RU5. El SIU encargado de mostrar información en los paneles instalados en marquesinas, recibirán la información de tiempo estimado de llegada tanto a través del Centro de Control como de una baliza bluetooth instalada en cada vehículo y también en la marquesina.
- RU6. Las comunicaciones entre los principales sistemas a bordo serán en base a protocolos abiertos basados en estándar para garantizar la integración si se añade o se sustituye un subsistema por otro de otro fabricante.
- RU7. Se implementará una red móvil 4G cuyas IPs son públicas y fijas, asignándose una para cada vehículo.
- RU8. Desde el centro de control se podrá:
    - RU8.A. Establecer comunicación directa con cada vehículo (llamadas, envío de mensajería al conductor) y recibir la información con las posiciones GPS en tiempo real, representándolas en un mapa.
    - RU8.B. Generar la parametrización de cada subsistema y enviarla a cada vehículo, de forma particular o a toda la flota.
    - RU8.C. La parametrización generada podrá ser de carácter ordinario o de carácter urgente. Si es de carácter ordinario, se podrá actualizar una vez finalice el servicio, ya en cocheras a través de WiFi. Si es de carácter urgente, se podrá enviar a los sistemas a bordo a través de la red móvil.

Tal y como se observa, no hay requisitos específicos sobre la seguridad del sistema. No obstante, se añade el siguiente:

- RU9. Se deberán tomar las medidas oportunas para la prevención y mitigación de ataques, así como del control anti fraude del sistema, siempre que sean posibles y compatibles con el resto de requisitos.

3.4.2. Diseño del sistema e interacción entre cada subsistema

Se establecen en primer lugar una serie de características comunes a todo el sistema, o que afectan a todos los subsistemas.

Para identificar cada requisito se utilizará el nemónico RSGEN, requisitos de sistema generales.

REQUISITO DE USUARIO	REQUISITO DE SISTEMA
RU3, RU4.A, RU4.D1, RU4.E.5, RU7, RU8.A, RU8.B, RU8.C	RSGEN1.A. Los sistemas a bordo han de poder conectarse con el Centro de Control.
	RSGEN1.B. Se creará por tanto una red embarcada punto a punto Ethernet, en la que todos los subsistemas tendrán acceso al exterior a través del router, y en la que todos los subsistemas podrán interactuar.
	RSGEN1.C. Se utilizará un router en el que irá instalada la SIM 4G, y que estará conectada a dicha red Ethernet embarcada.
RU3	RSGEN2. El router además tendrá antena Wifi, y se conectará a la red de las cocheras cuando se detecte la ESSID correspondiente configurada.

RU3, RU9	RSGEN3. La red Wifi de cocheras deberá tener método de seguridad y cifrado WPA2+AES con filtrado de MAC [68].
RU3, RU8.C, RU9	RSGEN4. Para la actualización de versiones en los subsistemas se utilizará el protocolo estándar SFTP (*Secure File Transfer Protocol*).
RU4.D.1, RU4.D.3, RU5, RU6	RSGEN5. Para la comunicación tanto entre equipos en la red embarcada dentro del vehículo como con el Centro de Control se utilizará el protocolo abierto basado en el modelo de datos *Transmodel CEN* [69].
RU4.E.5, RU9	RSGEN6. El envío de transacciones en tiempo real se efectuará a través de un Web Service sobre HTTPS, entre el cliente (SVV) y el servidor (Centro de Control).
RU4.G	RSGEN7. El router se configurará como un punto de acceso WiFi que dará acceso a Internet a los viajeros. Se podrá limitar la velocidad de navegación para evitar el incremento del consumo de datos.
RU4.G, RU9	RSGEN8.A. El router se configurará como punto de acceso cuando WiFi no esté en cocheras, restringiendo el acceso a la red embarcada, y a la red que se conecta con el Centro de Control y el resto de los equipos.
	RSGEN8.B. Se configurarán reglas en dicho router para evitar accesos no permitidos.
RU5, RU9	RSGEN9.A.El SIU de las paradas utilizará un router más simple dado solo necesita el uso de la red móvil 4G. Se podría utilizar el modelo LR77 v2 de Advantech B+B Smartworx [78].
	RSGEN9.B. Este router permitirá también la creación de reglas para la configuración de un firewall.
RU6,RU9	RSGEN10. Para añadir seguridad al protocolo abierto, se utilizará un algoritmo de cifrado simétrico 3DES y se realizará la comprobación del mensaje con un MD5.
RU7, RU9	RSGEN11.A. Para evitar ciberataques, se configurarán reglas en todos los routers y/o puntos de acceso al exterior.
	RSGEN11.B. Se seleccionará una lista determinada de IPs y servicios o puertos permitidos desde los cuales sí se podrá acceder (entre las que evidentemente figurará el Centro de Control).

	RSGEN11.C. Dicho firewall se configurará también en el Centro de control.
	RSGEN11.D. En los equipos embarcados, se protegerán los puertos bien conocidos con contraseña, o si no es posible se restringirán.
	RSGEN11.E. Se realizará una traducción NAT de puertos conocidos a puertos no estándar.
	RSGEN11.F. De ser posible, se controlarán los intentos de conexión a puertos no configurados y se rechazarán.
	RSGEN11.G. Los equipos de un autobús no tendrán acceso a equipos de otros autobuses, ni tampoco a los paneles de las marquesinas. Solamente se tendrá acceso a otros equipos dentro de la red embarcada y también con el Centro de Control.

3.4.3. Consideraciones generales sobre los equipos del Sistema IoT de Transporte

Para ahorrar el coste de la licencia del Sistema Operativo, todos los equipos embarcados operarán con GNU Linux, con versiones de *kernel* que varían entre el 2.6.35.2 para el equipo más simple que es el que gestiona los paneles de las paradas y tiene un hardware más sencillo y antiguo, y el 4.9.218 (*long term*) [70] para los equipos que implementan los demás subsistemas, el SIU, el SAE y el SVV. Respecto a la arquitectura, por eficiencia y costes, se opta por soluciones basadas en ARM.

El router será comercial, como el Cisco 880G [71], o el RUT950 de Teltonika [72], que además dispone de antena GNSS/GPS y permitirá la realización de llamadas GSM. En este equipo es donde se configurará el firewall que protege las redes embarcadas de los vehículos.

Respecto al Centro de Control, se instalará Windows Server 2019 con Software de gestión de seguridad con un *Suite* completo que incluya protección antivirus, antimalware, protección contra intrusos y también gestión de firewall (por ejemplo el de Kaspersky [73]).

3.4.4. Diseño del SIU

Existen dos partes dentro del SIU, el que está a bordo del vehículo y se encarga de mostrar la información al usuario en el autobús (en este caso en el TFT), y el SIU de los paneles de las paradas, que mostrará los tiempos estimados de llegada.

SIU del vehículo

Se utilizará el nemónico RSSIE, requisitos de sistema del SIU Embarcado.

REQUISITO DE USUARIO	REQUISITO DE SISTEMA
RU2, RU4.A	RSSIE1. No se utilizarán pantallas individuales por asiento. Se usará una TFT o pantalla de plasma industrial.

RU4.A	RSSIE2.A. El equipo[2] controlará el TFT a través de una conexión HDMI o mini HDMI.
	RSSIE2.B. El equipo permitirá el anuncio de paradas mostrando la información en el TFT. Se mostrará tanto la llegada a parada como la parada siguiente.
	RSSIE2.C. El equipo también mostrará la información multimedia (fotos, vídeos y audio) generada y actualizada por el Centro de Control.
	RSSIE2.D. El equipo ha de disponer de un dispositivo de almacenamiento lo suficientemente grande para poder almacenar la información multimedia actual y anterior. Se instalará un disco adicional de 100 GB.
RU4.B	RSSIE3. El equipo controlará los paneles exteriores del autobús para mostrar el trayecto que está realizándose, y cualquier otro mensaje que se configure desde el Centro de Control.
RU4.C	RSSIE4.A. El equipo controla los altavoces y permite abrir un canal de audio con el micrófono que está instalado en el autobús para que el conductor pueda realizar un aviso al pasaje.
	RSSIE4.B. El equipo emite por esos altavoces aviso de siguiente parada y llegada a parada, con ficheros de audio (locuciones).
RU4.F.1	RSSIE5. La consola del conductor permitirá solicitar al SIU el inicio de megafonías y avisos al pasaje.

SIU de la parada

Se utilizará el nemónico RSSIP, requisitos de sistema del SIU de parada.

REQUISITOS DE USUARIO	REQUISITOS DE SISTEMA
RU5	RSSIP1.A.El equipo controla un panel LED (a través de un protocolo propietario) en el que se mostrará la información de los tiempos estimados de llegada que envía el software de gestión del Centro de Control.

[2] Se ha de tener en cuenta que cada subsistema está gestionado por un equipo embarcado de perfil industrial y con sistema operativo GNU Linux.

	RSSIP1.B. El software de gestión del Centro de Control realiza los cálculos correspondientes para obtener los tiempos de llegada.
	RSSIP1.C. El equipo controla la baliza bluetooth de modo que si recibe la señal correspondiente (detección de campo de antena bluetooth del vehículo y número de bus correspondiente), se dejarán de mostrar tiempos para mostrar el aviso de llegada inminente.

3.4.5. Diseño del SAE

Se utilizará el nemotécnico RSSAE, requisitos de sistema del SAE.

REQUISITOS DE USUARIO	REQUISITOS DE SISTEMA
RU4.D.1	RSSAE1.A. El equipo tomará la información GPS/GNSS del router para detectar las paradas del itinerario, aunque dispone de su propia antena GPS.
	RSSAE1.B. Cada parada se asocia a unas coordenadas GPS, y esa relación se configura en el Centro de Control y se envía al equipo.
	RSSAE1.C. Ante la detección de una nueva parada, el equipo informará al resto de subsistemas de la red embarcada del evento.
	RSSAE1.D. De forma síncrona cada 5 segundos, y ante eventos determinados (como por ejemplo llegada y salida de parada) se envían mensajes al Centro de Control con la posición GPS, e información del vehículo y del servicio realizado.
	RSSAE1.E. La información recibida por estos mensajes se procesa en el Centro de Control y se muestra en un mapa, donde se podrá ver cómo avanza el vehículo realizando su recorrido.
RU4.D.2	RSSAE2.A. Las llamadas serán VoIP sobre SIP a través de la red móvil, o en caso de tener mala calidad de señal, serán GSM.
	RSSAE2.B. Desde el Centro de control se pueden iniciar llamadas, o recibir la petición realizada desde el vehículo y confirmarla para abrir el canal.
	RSSAE2.C. Desde el Centro de Control se podrán enviar mensajes.

	RSSAE2.D. Se podrán enviar mensajes cortos o predeterminados (así se evitan a los conductores distracciones al volante) desde el vehículo que se recibirán en el Centro de Control.
RU4.F.2	RSSAE3. La consola de conductor permitirá al SAE que envíe al Centro de Control una solicitud de llamada, y el envío de mensajes predeterminados.

3.4.6. Diseño del SVV

Se utilizará el nemónico RSSVV, requisitos de sistema del SVV

REQUISITOS DE USUARIO	REQUISITOS DE SISTEMA
RU4.E.1	RSSVV1. El equipo embarcado estará equipado o podrá controlar una impresora de papel térmico para imprimir los tickets correspondientes a los billetes de viaje y los QRs.
RU4.E.2,RU4.E.3	RSSVV2.A. El equipo estará equipado con un lector contactless que permita operar con tarjetas Mifare Classic, Mifare Desfire y tarjetas emuladas en dispositivos Android (Android HCE).
	RSSVV2.B. Las claves fijas de la tarjeta Mifare Classic podrán ser parametrizables, de modo que se tendrán que distinguir las versiones de claves a utilizar dentro de la tarjeta, conteniendo en la parametrización las distintas versiones de claves fijas existentes.
RU4.E.2, RU9	RSSVV3. Las claves en la parametrización del SVV han de venir cifradas u ofuscadas, para evitar ser fácilmente descubiertas si se interceptan.
RU4.E.3, RU9	RSSVV4.A. El equipo dispondrá de una o varios slots para la inserción de módulos de acceso seguros o SAMs que permitirán la autenticación con la tarjeta Mifare Desfire.
	RSSVV4.B. Ha de existir una entidad confiable que proporcione estos SAMs y los gestione para evitar fraude.
	RSSVV4.C. Con cada transacción que el equipo SVV envíe, se informará del SAM que lleva instalado.
RU4.E.4	RSSVV5. El equipo controlará un lector QR que permitirá leer QRs en formato papel y también en la pantalla de un Smartphone.

RU4.E.4, RU9	RSSVV6.A. El QR podrá usarse durante un tiempo parametrizable, controlado por el Centro de Control.
	RSSVV6.B. El equipo, al validar un QR en primer lugar comprobará si ya se ha caducado el tiempo de uso.
	RSSVV6.C. Luego comprobará si ya se ha validado en ese equipo.
	RSSVV6.D. Por último realizará una petición por Web Service al Centro de Control sobre HTTPS para comprobar si ese QR ya se ha validado en otro vehículo.

4. ESTUDIO DE AMENAZAS Y RIESGOS

4.1. Introducción al análisis y definiciones

A continuación se presenta el estudio que llevado a cabo para analizar las amenazas y riesgos inherentes al sistema IoT modelado.

Se tendrán en cuenta no solamente las características definidas a más bajo nivel del sistema, sino también la arquitectura global, y por otro lado los malos usos que puedan dar del mismo tanto los usuarios de la empresa de transportes, como los usuarios del servicio. También se tendrán en cuenta posibles eventos fortuitos que pongan al descubierto las debilidades del sistema.

En primer lugar antes de encontrar agujeros de seguridad en el sistema es importante definir el significado de debilidad, amenaza y riesgo [75].

4.1.1. ¿Qué es una debilidad?

Es un agujero del sistema que puede ser aprovechada por un atacante para degradar la integridad de los datos intercambiados, afectar el servicio o poner en entredicho la privacidad de los usuarios del mismo.

En el presente trabajo se habla de dos tipos de vulnerabilidades, las vulnerabilidades cuya definición coincide con la mencionada (se denominarán agujeros de seguridad o simplemente debilidades), y las vulnerabilidades que se estudian en la sección 6 y que se corresponden con los sistemas de clasificación utilizados por organizaciones y herramientas de seguridad, como *Bugtraq* o CVE.

4.1.2. ¿Qué es una amenaza?

Es toda acción que aprovecha una vulnerabilidad del sistema para atacar la seguridad de un sistema de información. Es interesante comentar que las amenazas pueden venir originadas por ataques, sucesos físicos fortuitos (apagones, terremotos, etcétera) o negligencias y malas políticas de seguridad o mal uso de las mismas.

4.1.3. ¿Qué es un riesgo?

Es la probabilidad de que se produzca un incidente de seguridad, materializándose una amenaza que finalmente provoca daños materiales y/o económicos.

4.2. Análisis de las características del sistema IoT

A continuación y de forma breve se enumeran las características del sistema que merece la pena tener en cuenta para el presente análisis.

4.2.1. Características comunes

- Los equipos embarcados funcionan con sistema operativo GNU/Linux.
- Los subsistemas embarcados funcionan con un router que permite la configuración de reglas para la creación de un firewall.
- Este router supone uno de los puntos de acceso desde el exterior al sistema.
- La conexión entre equipos dentro de cada subsistema se realizará por Ethernet.
- La red de datos móviles no es VPN por requisitos de usuario, los equipos con tarjeta SIM (los routers) tienen IP pública y fija.
- La descarga de ficheros se realizará a través de SFTP.
- Las comunicaciones entre subsistemas (a través de la red 4G) se harán a través de un protocolo basado en el modelo de datos *Transmodel CEN* [69].
- A dicho protocolo se añadirá seguridad cifrando el contenido de los mensajes (3DES) y un MD5 para verificar la integridad de los mismos.
- Las comunicaciones solo se permitirán entre Centro de Control y subsistemas en autobuses y paradas (SIUP), no entre los demás subsistemas entre sí.
- Los equipos de cada subsistema, pueden comunicarse entre sí a través de la red embarcada con protocolos propietarios.

4.2.2. Características de los routers utilizados

- Se utilizarán routers industriales basados en Linux, con soporte a largo plazo que permita a través de las actualizaciones solucionar vulnerabilidades.
- A bordo de los autobuses, podrá usarse el modelo RUT950 de Teltonika [72].
- En los SIU de las paradas, se podrá usar un router más sencillo, como el LR77 v2 de Advantech B+B Smartworx [78].
- Se configura un firewall que permita registrar los intentos de acceso tanto exitosos como frustrados, y se podrán descargar y consultar los logs correspondientes.
- Se ha de establecer una lista blanca de IPs permitidas que se corresponderán con los servidores del Centro de Control, y también los servidores y pasarelas del proveedor, para permitir las conexiones por motivos de atención de incidencias y mantenimiento.
- Los servicios con puertos conocidos se traducen en el router (traducción NAT) de modo que la conexión desde el exterior utilice un puerto no reconocido o estándar. Es decir, para acceder al SSH de un equipo en un autobús o en una parada, no se usará el puerto 22; se usará la IP de la SIM del router y un puerto que se traducirá al puerto 22 de la IP que tenga asignada el equipo al que se quiere acceder en la red embarcada.
- La herramienta utilizada para crear estas reglas sería el comando iptables, o similar.
- Es posible generar reglas para construir un firewall a través de herramientas como Firewall Builder [81].

4.2.3. Características del SIU

4.2.3.1. SIU del vehículo

- Equipo con GNU/Linux kernel 4.9.218.
- TFT o plasma industrial conectado por HDMI con el equipo.
- El equipo monta un disco adicional de 100GB donde almacenar todo el contenido multimedia, que recibe del Centro de Control a través del protocolo SFTP.

- Control de paneles exteriores del autobús a través de conexión cableada RS232 o RS485 y protocolo propietario.
- Control de los altavoces instalados en el autobús para emitir avisos, megafonías, etcétera.

4.2.3.2. SIU de la parada

- Equipo con GNU/Linux kernel 2.6.35.2.
- Control de un panel de LED a través de conexión cableada RS232 y/o RS485 y protocolo propietario.
- Recibe los tiempos que se han de pintar del Centro de Control a través de su conexión con el exterior, el router y protocolo basado en el modelo de datos *Transmodel CEN* [69].
- Control de antena bluetooth que recibe la señal de la baliza bluetooth instalada en el autobús, para mostrar el aviso inminente de llegada.

4.2.4. Características del SAE

- Equipo con GNU/Linux kernel 4.9.218.
- Se comunica con el SVV (a través de un protocolo propietario sobre Ethernet) y más concretamente la consola de conductor o HMI, para solicitar el envío de mensajería, solicitar fonías, o cambiar la parametrización vinculada al vehículo.
- Se comunica con el Centro de control a través del protocolo basado en el modelo de datos *Transmodel CEN* [69] para enviar las posiciones GPS, la información de los servicios realizados, mensajería, etcétera.

4.2.5. Características del SVV

- Equipo con GNU/Linux kernel 4.9.218.
- Incorpora (no menos de) impresora de papel térmico, lector de tarjetas contactless, pantalla táctil y/o teclado, lector de códigos de barras y QRs y slot(s) para módulos SAM.
- Se accede a dos Web Services sobre HTTPS en el Centro de Control (servidor):
    - Envío de transacciones en tiempo real
    - Consulta para saber si un QR se ha utilizado ya o no.
- Es el punto de entrada de dinero al sistema y por tanto se considera operacionalmente crítico.

4.3. Listado de debilidades amenazas y riesgos

Se enumeran las debilidades, amenazas y riesgos que afectan de forma general al sistema IoT, y posteriormente aquellas que afectan a cada subsistema. Se enumeran las vulnerabilidades y se recoge para cada vulnerabilidad la amenaza que puede materializarse, y el riesgo asociado, indicando los pasos que han de cumplirse para que la amenaza tenga éxito.

4.3.1. Análisis de Sistema IoT

Las debilidades estructurales del sistema IoT se derivan de la debilidad principal, que es el uso de una red 4G con IPs públicas. No obstante, pese a estar relacionadas, se tratarán y explicarán de forma independiente.

4.3.1.1. Red 4G con IPs públicas

Esta debilidad es realmente grave, y expone el sistema a ataques desde cualquier punto de Internet; no es necesario acceder a una VPN en primer lugar, lo que supondría una barrera para cualquier atacante, que para acceder al sistema en primer lugar tendría que sortear o hackear

la seguridad de la VPN, que aunque no es inexpugnable si permitiría mejorar la seguridad y privacidad del sistema.

Amenazas y riesgos asociados:

AMENAZA	DESCRIPCIÓN	RIESGO
Ataque de denegación de servicio de conexión a la red 4G.	El ataque consistiría en realizar intentos de conexión (no importa si fructuosos o no) a las IPs de las SIM de la red 4G, de modo que en cada intento, se consumen ciertos datos, y si se ha contratado con el operador de telefonía una tarifa con datos limitados, es posible tras muchos reintentos (hablamos de cientos de miles de intentos) llegar a ese umbral, y provocar que la velocidad de conexión de ese subsistema (vehículo o parada) se limite o incluso se bloquee.	Probabilidad: Alta .- Un atacante muy probablemente encuentre la IP pública de uno de los vehículos o los SIUs de parada. .- Es muy probable que la empresa de transporte haya contratado una tarifa limitada, puesto que son más baratas. .- Es poco probable que un router sea configurado para su desconexión de la red. Se utilizarán reglas del firewall configurado para proteger el subsistema de atacantes externos, pero no deshabilitando la conexión. Impacto: Alto/Muy Alto .- La conexión del subsistema empeoraría o se vería muy comprometida, teniendo que funcionar en degradado. .- El funcionamiento en degradado está pensado para un corto espacio de tiempo en el que se termina el servicio y se buscan piezas o equipos para sustituir las partes dañadas. .- Estaría afectando al envío de posiciones GPS, comunicaciones con el Centro de Control, descarga de actualizaciones, envío de transacciones en tiempo real y envío de tiempos en el caso del SIU de las paradas (como mínimo). Esto último provocaría que este subsistema dejara de funcionar adecuadamente, puesto que los tiempos recibidos en el subsistema

		estarían obsoletos al representarlos en el panel.
Descubrimiento de servicios / puertos en los equipos embarcados	A través de un ataque de fuerza bruta, el atacante descubriría los puertos que el router traduce (NAT) a puertos conocidos correspondientes con servicios en la red embarcada.	Probabilidad: Baja/Media .- Un atacante muy probablemente encuentre la IP pública de uno de los vehículos o los SIUs de parada. .- Dado que el router implementa un firewall que filtra aquellas peticiones procedentes de IPs no conocidas, el atacante deberá descubrir en primer lugar alguna de estas IPs. .- Tras descubrir una de estas IPs permitidas, el atacante deberá poder camuflar su propia IP y realizar un ataque IP spoofing suplantando la IP permitida. .- El atacante debería ser capaz de además de suplantar la IP permitida, realizar todas las peticiones necesarias e interceptar las correspondientes respuestas. Impacto: Alto .- El atacante con esta información podría organizar estrategias de ataque más exhaustivas.
Inicio de sesión (SSH/SFTP/FTP) con los equipos embarcados	El atacante, suplantando una de las IPs permitidas, iniciaría la conexión con uno de los equipos de los subsistemas (en los autobuses, o los SIUs de las paradas). Deberá partirse de la situación de que el atacante ha sido capaz de realizar de forma exitosa el ataque anterior, y ha podido descubrir los puertos utilizados.	Probabilidad: Baja .- El atacante, suplantando la IP permitida deberá iniciar una conexión, para lo que tendrá que adivinar en primer lugar usuario y password. Si no se usan nombres conocidos de usuarios (root, admin, etcétera), será menos probable que el atacante descubra las credenciales. Impacto: Alto/Muy Alto .- Si un atacante consigue iniciar una sesión con uno de los equipos, tendría acceso

		completo a la red embarcada y podría provocar ataques de denegación de servicio a varios niveles, obtener y revelar información esencial sobre el sistema, y manipular los datos de operación.
Secuestro de sesión (SSH/SFTP/FTP) con los equipos embarcados	Similar al anterior, en este ataque se estaría secuestrando una sesión ya iniciada por un usuario legítimo.	Probabilidad: Baja .- El atacante debería ser capaz de detectar sesiones ya iniciadas, suplantar la IP que ha iniciado la sesión, e interceptar la sesión ya iniciada. Impacto: Alto/Muy Alto .- Las implicaciones son las mismas que en el caso del ataque anterior.
Inicio o secuestro de sesión de equipos embarcados desde otro equipo embarcado	El atacante realizaría un ataque de inicio o secuestro de sesión con uno de los equipos embarcados de los subsistemas, desde otro de estos equipos embarcados. Se partiría por tanto de la situación en la que el atacante ha sido capaz de realizar de forma exitosa el ataque de inicio o secuestro de sesión.	Probabilidad: Muy baja/Baja .- El atacante tendría que ser capaz de suplantar IPs en dos tramos. En primer lugar se suplantará la IP permitida para poder acceder a la sesión con el equipo embarcado, y una vez que el atacante controla dicha sesión, volver a suplantar la IP permitida, aunque proceda de un subsistema, puesto que los routers filtran las conexiones procedentes de otros subsistemas; solo permiten conexiones procedentes de las IPs permitidas. Impacto: Alto/Muy Alto .- Las implicaciones son las mismas que en el caso del ataque anterior.

4.3.1.2. Uso de Protocolos abiertos basado en estándar

Se utiliza un protocolo abierto entre los vehículos y los SIUP con el Centro de Control. Se recuerda que se añade seguridad cifrando el contenido de cada mensaje, con un hash para garantizar la integridad de los datos.

Amenazas y riesgos asociados:

AMENAZA	DESCRIPCIÓN	RIESGO
Sniffing	El ataque consistiría en interceptar el tráfico intercambiado y procesarlo o descifrarlo en este protocolo entre los dos extremos de la comunicación.	Probabilidad: Baja .- El atacante con probabilidad media será capaz de interceptar tráfico intercambiado en este protocolo. .- El atacante debería ser capaz de adivinar el algoritmo de cifrado y las claves utilizadas, para acceder al contenido en claro del mensaje. Impacto: Medio .- El atacante obtendrá información relevante del protocolo e incluso relativa a la operativa del subsistema.
Phishing con reenvío de mensaje	El atacante, reconociendo el protocolo, interceptaría mensajes y los reenviaría sin modificarlos para alterar la secuenciación del protocolo, provocando que el otro extremo reenviara una y otra vez el mismo mensaje, o reenviando el ACK correspondiente a un mensaje que ya ha recibido.	Probabilidad: Media .- El atacante debería en primer lugar ser capaz de reconocer que protocolo se está usando. .- El atacante tendrá que ser capaz de capturar o escuchar (ataque man-in-the-middle) la comunicación entre dos extremos de la red IoT. .- Pese a que el contenido está cifrado y firmado, el atacante interceptará mensajes y las reinsertará en la sesión de comunicación (suplantando su IP por una de las permitidas). Impacto: Medio/Alto .- Si el atacante es capaz de reinsertar mensajes, alteraría la secuenciación de los mensajes intercambiados, provocando la ralentización del servicio, o incluso el reinicio de sesión de comunicación entre los dos extremos.
Phising con hash	El atacante interceptaría el mensaje intercambiado y modificaría parte del mensaje utilizando	Probabilidad: Baja .- El atacante debería en primer lugar ser capaz de reconocer que protocolo se

	fragmentos del propio mensaje, recalculando la firma del mismo y reinsertándolo en el enlace de la comunicación entre ambos extremos.	está usando y de detectar que se está cifrando y firmando el contenido del mensaje. .- El atacante tendrá que ser capaz de capturar o escuchar (ataque man-in-the-middle) la comunicación entre dos extremos de la red IoT. .- El atacante interceptará mensajes, los editará y los reinsertará en la sesión de comunicación (suplantando su IP por una de las permitidas). .- El atacante debería ser capaz de adivinar el algoritmo de firma y ser capaz de recalcular las firmas del mensaje modificado. Impacto: Alto .- Si el atacante tiene éxito, la comunicación entre los extremos no solamente se verá afectada como en el caso descrito anterior, si no que cabe la posibilidad de que se alteren datos que se intercambian y que el otro extremo los de por buenos.
Phishing con cifrado y hash	El atacante interceptaría el mensaje intercambiado y modificaría todo o parte del mensaje, editando los campos que correspondan, recalculando la firma del mismo y reinsertándolo en el enlace de la comunicación entre ambos extremos.	Probabilidad: Muy Baja/Baja .- El atacante debería en primer lugar ser capaz de reconocer que protocolo se está usando y de detectar que se está cifrando y firmando el contenido del mensaje. .- El atacante tendrá que ser capaz de capturar o escuchar (ataque man-in-the-middle) la comunicación entre dos extremos de la red IoT. .- El atacante interceptará mensajes, los editará y los reinsertará en la sesión de comunicación (suplantando su IP por una de las permitidas).

		.- El atacante debería ser capaz de adivinar el algoritmo de cifrado, las claves utilizadas, y el algoritmo de firma y ser capaz de recalcular las firmas del mensaje modificado. Impacto: Alto/Muy Alto .- Si el atacante tiene éxito, la comunicación entre los extremos no solamente se verá afectada como en el caso descrito anterior, si no que los datos intercambiados serán los que el atacante desee modificar, por tanto podría afectar tanto a la operativa percibida por uno de los extremos, como a la operativa real del sistema.

4.3.1.3. Uso de Web Services sobre HTTPS

Se utilizan Web Services sobre HTTPS, es decir, se utilizará cifrado asimétrico y certificados (HTTP sobre TLS/SSL). Se tienen en cuenta posibles debilidades de los Servidores Web [79]:

- Configuración por defecto: No se modifica la configuración que se autogenera por defecto en el servidor Web.
- Configuración errónea de sistemas operativos y redes: Ciertas parametrizaciones permitirían la ejecución de comandos en el servidor a cuya sesión un atacante podría acceder si se configura una contraseña débil.
- Errores en los sistemas operativos y Web Servers: Ciertos bugs en los servidores Web y/o sistemas operativos pueden explotarse para que un atacante sea capaz de adquirir acceso no autorizado al sistema.

Amenazas y riesgos asociados:

AMENAZA	DESCRIPCIÓN	RIESGO
Ataque de directorio transversal	El atacante se aprovecha de bugs en el Servidor Web para obtener acceso no permitido a ficheros y directorios que no están en el dominio público.	Probabilidad: Baja .- El atacante tiene que ser capaz de sortear el firewall del servidor y acceder al Web Server. .- El atacante deberá encontrar la ruta y/o combinación del directorio que esté fuera del alcance del dominio público del Servidor. Impacto: Alto

		.- Una vez que el atacante ha conseguido acceder, puede instalar software malicioso, obtener información sensible o incluso ejecutar comandos.
Ataque de denegación de servicio	El ataque consistiría en provocar malfuncionamientos del servidor Web, o fallos que provoquen una parada en la operativa, de modo que deje de estar disponible para los usuarios legítimos.	Probabilidad: Baja .- Han de existir bugs que el ataque sea capaz de explotar que permitan una denegación de servicio. Impacto: Muy Alto .- El servicio deja de estar disponible.
Secuestro de DNS	El ataque consiste en modificar la configuración del DNS para que apunte al Web Server del atacante, de modo que recibe todas las respuestas.	Probabilidad: Baja .- El atacante ha de poder acceder al servidor DNS y modificar su configuración. Impacto: Alto .- Todas las respuestas del Web Service correspondiente se dirigirán al atacante.
Sniffing	El ataque consistiría en interceptar mensajes que si consigue descifrar le permitirían al atacante obtener acceso no autorizado al servicio web.	Probabilidad: Baja .- El atacante con probabilidad media / alta podrá capturar tráfico de la comunicación del web service. .- El atacante debería ser capaz de adivinar el algoritmo de cifrado y las claves utilizadas para poder acceder al contenido en claro. Impacto: Alto .- Si el ataque prospera podrá
Phishing	El atacante suplanta a uno de los extremos (cliente/servidor) y modifica y reinserta mensajes, forzando respuestas.	Probabilidad: Baja .- El atacante deberá de ser capaz de adivinar el algoritmo de cifrado y las claves utilizadas. .- El atacante deberá ser capaz de suplantar uno de los dos extremos de la comunicación o bien secuestrará el DNS. .- El atacante deberá ser capaz de interceptar

		mensajes, descifrarlos, modificarlos y reinsertarlos en el canal de comunicación. Impacto: Alto .- El atacante podría forzar el envío de peticiones cuya respuesta podría llegar a incluir información sensible que el atacante sustraería.

4.3.2. Análisis del SIU

4.3.2.1. Análisis del SIU del vehículo

4.3.2.1.1. Uso de dispositivo de almacenamiento masivo o disco duro

Es muy común que estos dispositivos no estén debidamente protegidos ni se configuren como una partición fantasma o cifrada.

AMENAZA	DESCRIPCIÓN	RIESGO
Denegación de Servicio de información de usuario	El ataque consistiría en que, una vez que se ha obtenido acceso a la sesión, el atacante podría acceder a la unidad en la que se monta el dispositivo de almacenamiento y borrar su contenido.	Probabilidad: Baja .- Existe una baja probabilidad (o muy baja) de que el atacante secuestre o inicie una sesión con el equipo SIUV (SIU del vehículo). Impacto: Alto/Muy Alto .- Si se elimina el contenido, el SIUV no podrá mostrar información al usuario en el autobús. Ni reproducir audios ni mostrar vídeos.
Phishing	El ataque consistiría en que, una vez que se ha obtenido acceso a la sesión, el atacante podría acceder a la unidad en la que se monta el dispositivo de almacenamiento y modificar su contenido sustituyendo los ficheros de media con los que incruste el propio atacante.	Probabilidad: Baja .- Existe una baja probabilidad (o muy baja) de que el atacante secuestre o inicie una sesión con el equipo SIUV (SIU del vehículo). Impacto: Alto/Muy Alto .- Se podría llegar a afectar subrepticiamente la información al usuario, de modo que no sería evidente a priori para el conductor o el controlador, lo que implica que al no ser tan evidente la incidencia se tardará más en solventar.

4.3.2.2. Análisis del SIU de la parada

Las debilidades y amenazas que afectan al protocolo entre el subsistema y el centro de control y a los términos generales de un ataque de secuestro de sesión ya se recogen ya en el apartado 4.3.1.2. Servicios que quedarían afectados con ataques aprovechando las debilidades comentadas:

- Visualización correcta de tiempos de llegada de una precisión determinada en el panel.
- Actualizaciones.

4.3.3. Análisis del SAE

Las debilidades y amenazas que afectan al protocolo entre el subsistema y el centro de control y a los términos generales de un ataque de secuestro de sesión ya se recogen ya en el apartado 4.3.1.2. Servicios que quedarían afectados con ataques aprovechando las debilidades comentadas:

- Envío de posiciones GPS al Centro de Control.
- Solicitud o gestión de llamadas telefónicas entre el Centro de Control y el conductor. No obstante, se podrán seguir realizando llamadas directamente, sin pasar por el sistema.
- Actualizaciones.

4.3.4. Análisis del SVV

Las debilidades y amenazas que afectan al protocolo entre el subsistema y el centro de control y a los términos generales de un ataque de secuestro de sesión ya se recogen ya en los apartados 4.3.1.2 y 4.3.1.3. Servicios que quedarían afectados con ataques aprovechando las debilidades comentadas:

- Envío de transacciones en tiempo real al Centro de Control (Web Service de transacciones en tiempo real).
- Envío de petición al WS de antifraude de QR.
- Actualizaciones.

4.3.4.1. Tecnología Mifare Classic

Amenazas y riesgos asociados:

AMENAZA	DESCRIPCIÓN	RIESGO
Descubrimiento de claves y modificación del contenido de la tarjeta	El ataque, muy frecuentemente por fuerza bruta, consistiría en descubrir las claves que permitirían acceder al contenido en claro de la tarjeta y llegar a modificarlo.	Probabilidad: Media/Alta .- El atacante, muy probablemente podría descubrir las claves. .- El atacante una vez que acceda al contenido de la tarjeta, deberá descubrir qué campos se refieren al saldo o la caducidad de la tarjeta. .- El atacante deberá crear una aplicación o usar una herramienta que le permita editar la imagen de la tarjeta, con el efecto de una recarga (de saldo o de caducidad).

		Impacto: Medio/Alto .- Si las claves son fijas e iguales para todas las tarjetas, el atacante podría divulgar estas claves de modo que los usuarios podrían llegar a modificar la imagen de sus tarjetas y viajar gratuitamente. Se generalizaría el fraude. .- En caso de que las claves no sean iguales para todas las tarjetas, la afectación no sería la misma, puesto que el atacante sólo descubriría las claves de su tarjeta, y solo podría modificar la imagen de su tarjeta. Solo el atacante podría llegar a cometer fraude [86].

4.3.4.2. Ventas en efectivo con soporte papel

Este tipo de medio de pago, permite el fraude por parte de los conductores, que podrían reutilizar vendiendo el soporte físico (ticket) para su propio beneficio.

Amenazas y riesgos asociados:

AMENAZA	DESCRIPCIÓN	RIESGO
Reutilización de tickets	Este tipo de ataque (más bien fraude) lo llevaría a cabo el conductor, que es quien realiza la venta del ticket al usuario. El conductor realiza ventas y las anula antes de que los viajeros accedan al vehículo, entregando a éstos el ticket del billete vendido. Los viajeros le pagan al conductor un dinero por el por un servicio que piensan que están disfrutando, cuando no es así, ya que el conductor se queda con el importe.	Probabilidad: Baja/Media .- El atacante (el conductor en este caso) debe realizar ventas y anularlas sin que eso llame la atención del centro de control. Dichos tickets los entrega al viajero. .- El viajero al recibir el ticket con la venta realizada minutos, horas o incluso días antes de la fecha actual, no se tendría que fijar en la comentada fecha para que el ataque surtiera efecto. .- En caso de que revisase la fecha y hora del ticket, el viajero obviaría esta situación o no le daría importancia. .- La empresa de transportes no tendría que realizar inspecciones sorpresa para

		disuadir al conductor de realizar estos intentos de fraude. Impacto: Medio .- El perjuicio producido por el conductor a la empresa de transportes depende de la cantidad de billetes vendidos fraudulentamente durante un tiempo determinado. Suelen ser cantidades bajas en comparación con lo que ingresa la compañía.

4.3.4.3. Ventas en soporte QR

Debido a la operativa de este soporte de medio de pago, se está aceptando una ventana de fraude, que se traduce en un tiempo determinado durante el que el viajero puede reutilizar el QR.

AMENAZA	DESCRIPCIÓN	RIESGO
Reutilización de soporte QR	El atacante (en este caso el viajero) copia u obtiene por otros medios un ticket QR que ya se haya utilizado para disfrutar del servicio de forma fraudulenta.	Probabilidad: Media .- El sistema es probable que realice algún tipo de control antifraude sobre el QR, pero podrá existir una ventana de tiempo durante la que el usuario puede reutilizar el QR. Impacto: Bajo/Medio .- El perjuicio provocado a la empresa de transportes es bajo comparado con los ingresos de la compañía.

4.3.4.4. Tecnología EMV

Cuando se realiza un pago con tarjeta, el SVV, a través de un túnel virtual que levantan los propios dispositivos EMV, se conectan a la red bancaria para validar la transacción y sustraer la cantidad correspondiente al viaje del número de cuenta asociado a la tarjeta. Si en el momento de efectuar la validación, no es posible realizar esta conexión con la red bancaria, se da de paso la transacción y se considera el pago realizado, aunque no se hubiera podido verificar.

AMENAZA	DESCRIPCIÓN	RIESGO
Pago del servicio con tarjetas sin fondos	El atacante (viajero) utiliza para el pago tarjetas que no tienen crédito o que están asociadas a cuentas bancarias sin fondos.	Probabilidad: Media .- El atacante debe ser capaz de detectar los puntos y situaciones en las que hay poca o nula cobertura de la

		red de datos, necesaria para validar el pago con la tarjeta. .- En caso contrario, el atacante debería ser capaz de poder forzar la pérdida de cobertura (a través de ataques de denegación de servicio, o similar). Impacto: Bajo .- El atacante tendrá un tiempo limitado para poder realizar este fraude de manera impune, puesto que en cuanto el sistema sea capaz de enviar y validar la transacción, es muy probable que su tarjeta o tarjetas queden inutilizadas.

4.4. Análisis de casuísticas posibles

En el listado de amenazas y riesgos asociados a cada vulnerabilidad se ha recogido, concretamente en la columna del riesgo, los pasos que tiene que conseguir llevar a cabo el atacante para conseguir que su ataque prospere, y cómo de fácil o de difícil es que se cumplan todos los condicionantes.

Cabe destacar la casuística que tiene que darse para que las amenazas relativas a las principales debilidades del sistema prosperen:

4.4.1. Ataques a la red 4G

Tal y como se ha mencionado, la debilidad más grave del sistema es la red 4G con IPs públicas. Esto expone el sistema a las inclemencias de los atacantes, que pueden efectuar sus ataques desde cualquier punto de Internet.

No obstante, el operador móvil tiene cierta responsabilidad con respecto a la seguridad de estos equipos y por tanto del sistema. Es beneficioso para el operador (y también para el resto de implicados en la implementación y el uso del sistema) ejercer cierto tipo de control y por tanto aplicar mecanismos para detectar estos ataques y prevenirlos en la medida de lo posible. Si se controla el volumen de tráfico en la red, se podrán detectar picos de tráfico o nodos que reciben o transmiten un volumen inusual de datos, pudiendo de este modo controlar o evitar la saturación de la red, que le impediría al operador dar un servicio al resto de sus usuarios y clientes.

Por otro lado, sería muy deseable modificar la topología de la red manteniendo el cumplimiento de los requisitos establecidos por el contratista. La instalación y uso de una red virtual privada (VPN) sería la solución más práctica puesto que establecería un primer obstáculo (no exento de problemas) a los atacantes. En caso de que el contratista no acepte esta solución y quiera seguir manteniendo las IPs públicas, habría que idear otros mecanismos, como la instalación de servidores proxy que permitan abstraer la red frente al exterior del sistema, de modo que desde

fuera de la red solamente pueda verse la IP pública de la interfaz externa del proxy, pudiendo añadir excepciones para la lista de IPs permitidas (servidores del contratista y del proveedor).

Los ataques relacionados con la debilidad de la red, como por ejemplo el acceso a los equipos, el hackeo del protocolo entre subsistemas y Centro de Control y el hackeo de los servicios web, reduciendo el riesgo y más concretamente la probabilidad de que éstos sucedan, si se mejora la seguridad de la red de la manera descrita.

4.4.2. Ataques al sistema SVV

Estos ataques son realmente intentos de fraude, principalmente llevados a cabo por los usuarios finales para poder acceder al servicio de forma gratuita, o para incluso sacar un provecho económico (vendiendo la aplicación, vendiendo a mitad de precio las recargas, etcétera). Es responsabilidad del contratista, pero sobre todo del proveedor instalar un sistema robusto, suficientemente probado y fiable, y también diseñado en base a la seguridad para dificultar el fraude. El uso de tecnologías cada vez más sofisticadas y enfocadas no solo a la facilidad de uso sino también a la seguridad (como es el caso de las tarjetas Desfire con el uso de SAMs), en conjunción con la evolución equipos embarcados (como ya se ha mencionado en apartados anteriores), permite mejorar y facilitar la implementación de estos sistemas, que tienen la dura misión de ser el punto de entrada del dinero al sistema.

5. GUÍA DE MEJORES PRÁCTICAS PARA PROTEGER EL SISTEMA IOT

A continuación se recogen una serie de procedimientos o pasos a seguir que servirán para proteger el sistema IoT y prevenir ataques, minimizar riesgos y mitigar los efectos de un ataque en caso de que se produzca.

Dado que se está diseñando e implementando un sistema desde cero, lo ideal sería tener en cuenta no solo las medidas para prevenir ataques del sistema ya terminado, sino también los requisitos de seguridad en todas las fases del SDLC (*Systems Development Cycle Life*), es decir durante todas las fases del ciclo de vida del desarrollo e implantación de un sistema [76]: (i) análisis y diseño, (ii) desarrollo e implementación, (iii) *testing* y aceptación, (iv) despliegue e integración y (v) mantenimiento y entrega. Por simplicidad se asumirá que la implementación del sistema completo la realizará un único proveedor.

Por otro lado, para organizar los procedimientos a seguir sobre seguridad, en atención al origen de los riesgos que comportan, podría hacerse la siguiente distinción de dominios de seguridad (4.2 de [76]):

1. *Personas*: Todas aquellas medidas de seguridad que afectan a las partes implicadas, desde los desarrolladores del sistema, hasta los usuarios finales, pasando por los clientes, que son quienes utilizarán y sacarán provecho del sistema.
2. *Procesos*: Se ha de considerar la seguridad como requisito de diseño (aunque no se especifique de forma explícita en los requisitos de usuario) y se ha de tener en cuenta todos los procedimientos de seguridad necesarios durante las fases de ciclo de vida del desarrollo de un sistema IoT, incluyendo desarrollo e implementación, pruebas, procedimientos de integración continua, auditorías de seguridad, etcétera.
3. *Tecnologías*: Se han de tomar todas las medidas necesarias para eliminar las debilidades del sistema y facilitar los procesos de seguridad a llevar a cabo durante las fases de la implantación del sistema.

5.1. Personas

En este dominio se atajarán los problemas relacionados con el desarrollo del sistema por parte de los proveedores, así como del personal que proporcionará el servicio de transportes a través del sistema y finalmente los viajeros o usuarios finales del sistema.

5.1.1. Formación

Es importante que las partes implicadas adquieran conocimientos que les permitan hacer su trabajo con seguridad.

5.1.1.1. Formación del equipo de desarrolladores del proveedor del sistema

4. PR01. Potenciar los conocimientos del personal en cuanto al desarrollo de aplicaciones seguras.
    a. PR01.A. Se considerará como requisito principal la seguridad en las fases SDLC de cada aplicación del sistema.
    b. PR01.B. Se evitará el uso de funciones vulnerables, en atención al lenguaje o lenguajes de programación utilizados.
    c. PR01.C. Se utilizarán herramientas de análisis estático (por ejemplo *CppCheck* si el lenguaje de programación es C/C++) y dinámico (por ejemplo *Valgrind*) que permitirán asegurar que el código fuente no presenta vulnerabilidades.
5. PR02. Potenciar los conocimientos del personal en cuanto a las herramientas de realización de controles de validación y *testing* de las aplicaciones desarrolladas.
    a. PR02.A. Familiarización con herramientas como Google Test o CUnit para la ejecución de pruebas unitarias para la verificación de cada función implementada.
    b. PR02.B. Familiarización con herramientas de integración continua (como Jenkins), que permitirán la ejecución de pruebas automáticas para verificar la robustez de las aplicaciones desarrolladas.
6. PR03. Potenciar los conocimientos del personal en cuanto a la aplicación del RGPD (Reglamento General de Protección de Datos).
    a. PR03.A. Tener en consideración los términos del contrato de protección de datos firmado con la otra parte (con la empresa de transportes):
        i. PR03.A.I. Tipo y duración del tratamiento de datos.
        ii. PR03.A.II. Datos tratados.
        iii. PR03.A.III. Quién será y qué funciones tendrá el responsable del tratamiento.
        iv. PR03.A.IV. Como responder a las peticiones para la aplicación de los derechos de la otra parte (rectificación, eliminación, etcétera).
    b. PR03.B. Tener en cuenta mecanismos y herramientas para evitar el tratamiento de datos no recogidos en el contrato, bien a través de seudonimización o cualquier otro mecanismo recogido en la RGPD.

5.1.1.2. Formación del personal de la empresa de transportes

1. PR04. Potenciar o efectuar campañas de formación a los grupos de empleados de la empresa de transportes:
    a. PR04.A. Formación de conductores:
        i. PR04.A.I. Operativa: Qué secuencia de acciones se han de llevar a cabo y cuales no para comenzar el servicio, su realización y su finalización. Muy importante también la gestión del uso de login y password de cada conductor para poder iniciar un servicio. Importante también restringir

el acceso durante las ausencias temporales (recesos) a otros conductores o usuarios malintencionados (bloqueo del HMI de la consola de conductor, etcétera).

ii. PR04.A.II. Control del fraude: Atención del viajero o usuario final frente a situaciones conflictivas (si un usuario no paga, o la validación de la tarjeta de transportes es incorrecta, etcétera).

iii. PR04.A.III. Hábitos para la mejora de la seguridad: Evitar durante las llamadas con el centro de control la mención a datos personales o bancarios, evitar dejar notas escritas en el salpicadero o alrededor del asiento con datos sensibles, como números de teléfono personales, números de cuenta, pines, contraseñas, etcétera.

b. PR04.B Formación de inspectores y/o supervisores:

i. PR04.B.I. Operativa: Qué acciones tienen que llevar a cabo los inspectores para verificar si los viajeros han pagado su viaje.

ii. PR04.B.II. Control de fraude e imposición de sanciones: Qué acciones tiene que realizar el inspector para tratar con el viajero final que ha incumplido o no ha realizado el pago, y asignarle la sanción correspondiente que puede ser económica, o a través del bloqueo de su tarjeta de transporte.

c. PR04.C. Formación de controladores o personal del centro de control:

i. PR04.C.I. Operativa: Será importante mejorar los conocimientos y hábitos de los controladores en cuanto al acceso a la sesión de control a través de login y password.

ii. PR04.C.II. Hábitos para la mejora de la seguridad: Se han de tener en cuenta estos accesos con login y password también durante los recesos, para evitar el acceso a la sesión a usuarios malintencionados. Se podría evitar bloqueando la sesión del Sistema Operativo (en Windows, usar el atajo ⊞+L). También se ha de evitar que el lugar de trabajo se puedan encontrar anotaciones con datos personales o bancarios ya sea del controlador o de cualquier otro trabajador (conductores, inspectores, etcétera).

iii. PR04.C.III. Contacto con los conductores / inspectores / supervisores: Tener en cuenta la confidencialidad que se ha de mantener durante una llamada telefónica o VoIP con un conductor, ya que podría ser oída por un usuario malintencionado. Se han de evitar menciones a datos relevantes personales, o bancarios como números de pin, contraseñas, números de tarjetas bancarias, números de cuenta, etcétera.

5.1.1.3. Campaña de información al usuario final

PR05. Se ha de informar al usuario final de los servicios que se ofrecerán junto con el nuevo sistema, advirtiendo también de los usos indebidos y sus consecuencias. En un intento de advertir y disuadir a los posibles usuarios malintencionados.

PR06. Junto con la información tarifaria que se aplique, se advertirá de las sanciones que se podrán imponer a los usuarios que viajen sin haber pagado su título de transporte.

5.1.2. Asignación de roles

PR07. Para cada grupo de usuarios dentro del sistema, se asignarán determinados roles que permitirán gestionar de manera eficiente los permisos o el grupo de acciones que serán permitidos dentro del ámbito correspondiente del sistema.

5.1.2.1. Roles de usuarios del sistema embarcado

Son usuarios con los que podrán efectuarse operaciones en el sistema a bordo del autobús, a través de la consola de conductor o el HMI de SVV.

CONDUCTORES: PR08. Controlan a nivel usuario el sistema embarcado. Podrán acceder al HMI del SVV e interactuar a través de dicho HMI con el SAE y el SIU embarcados. Podrán solicitar llamadas desde el centro de control y bloquear y desbloquear el HMI con su login y password.

SUPERVISORES: PR09. Podrán desbloquear un HMI bloqueado con cualquier conductor con su login y password.

INSPECTORES: PR010. Podrán solicitar la impresión de listados (o en todo caso consultar la información) con el detalle de las ventas realizadas en el último trayecto (cantidad, y detalle con su respectivo número de serie de la operación de venta) o la información con la recaudación y viajeros acumulados.

MANTENEDORES: PR011.Estos usuarios podrán acceder a las opciones de mantenimiento del sistema embarcado y poder cambiar parámetros relativos a la operativa, como por ejemplo el número de autobús, tipo de periféricos conectados, consultar el número de serie y las versiones de los equipos, consultar información de coberturas y estados de los diferentes elementos de la red embarcada, y poder enviar un informe a un servidor a través de un cliente de Web Service.

5.1.2.2. Roles de usuarios del centro de control de SAE/SIU

CONTROLADOR: PR012. A través de este usuario se podrá iniciar sesión en el puesto del Centro de Control, y monitorizar los autobuses de las líneas controladas, pudiendo iniciar llamadas telefónicas, envío y recepción de mensajería y consulta de información en tiempo real del avance del autobús en el trayecto, y los tiempos de paso por parada. También se podrá consultar la información de los paneles de las paradas, pudiendo modificar la información adicional y el formato de la misma (tamaño y tipo de letra, logo, información en scroll, animaciones, etcétera).

ADMINISTRADOR: PR013. El administrador además de efectuar las operaciones que realiza el controlador, modificar los permisos del controlador y también podrá refrescar la información de los paneles con datos precargados, reiniciarlos remotamente, o recoger los logs generados en los equipos para monitorizar su actividad.

5.1.2.3. Roles de usuarios del centro de control de SVV

CONTROLADOR: PR014. A través de este usuario se podrán crear parametrizaciones para el SVV embarcado, modificando tarifas, calendarios de aplicación, títulos de transporte, diseño de cabeceras y pies, o añadiendo, modificando (cambiando contraseña o datos asociados) o eliminando conductores, inspectores, supervisores y mantenedores. También se podrán consultar los datos de ventas, y generar y visualizar los reportes de recaudación del sistema.

ADMINISTRADOR: PR015. Además de las acciones asignadas al controlador, a través de este usuario se podrá crear otros usuarios administradores, o modificar los permisos asignados a cada rol de usuarios del SVV embarcado (conductores, inspectores, supervisores y mantenedores).

5.2. Procedimientos

En este dominio se atajarán los problemas que pueden prevenirse estableciendo procedimientos de seguridad en las entidades intervinientes en la implementación del sistema (proveedores, contratistas, y terceras partes). Se consideran en este apartado los procedimientos a implantar en las fases SLDC, y los procedimientos a implantar a nivel corporativo, entre ellas las políticas internas de seguridad.

5.2.1. Procedimientos de Gestión de operaciones

5.2.1.1. Plan de gestión de incidencias

PC01. Se ha de elaborar un plan de gestión de incidencias para controlar las vulnerabilidades del sistema y las acciones a llevar a cabo en caso de que se produzca un ataque, aprovechando vulnerabilidades que se hayan tenido en cuenta y también las que no.

5.2.1.2. Plan de gestión de cambios

PC02. Se ha de definir un plan de gestión de cambios durante la fase de desarrollo del sistema u otras fases SDLC, puesto que dichos cambios pueden tener un impacto en la integridad del sistema, su robustez y desempeño, y también afectar al presupuesto del proyecto, calendario, alcance, comunicación y recursos. Por todo ello cualquier cambio se ha de gestionar correctamente.

5.2.1.3. Implementación de gestión de vulnerabilidades y parches

PC03. Se ha de definir un plan que permita gestionar las vulnerabilidades del sistema y las actualizaciones y los parches necesarios para solventarlos. Se ha de valorar el impacto que puede suponer y los beneficios de seguridad que se obtendrían de su aplicación.

5.2.1.4. Implementación de gestión de la configuración

PC04. Se ha de tener en cuenta que existe una parte asociada al software y procesos del sistema que es la configuración. En este caso existe parte de esa configuración que queda bajo el control del contratista, puesto que forma parte de los requisitos del sistema, y es muy complicado impedir que cualquier cambio no deseado se aplique en el sistema. No obstante, dado que dicha configuración se genera a partir de herramientas integradas con la solución, se establecerán procedimientos para asegurar que las configuraciones generadas no contendrán elementos que puedan provocar errores, malfuncionamientos o vulnerabilidades en el sistema.

PC05. Se podría establecer que antes de recargar parámetros de configuración en el sistema, se han de comprobar en primer lugar, no solamente su formato para evitar cargar ficheros corruptos o mal formados, si no también si los valores parametrizados están dentro de rangos esperados y que van a gestionarse correctamente en el software de cada subsistema.

5.2.2. Procedimientos de seguridad en las fases SLDC

PC06. Estos procedimientos se implantarán en la empresa proveedora, y podrían ser exigibles a terceras partes, si intervienen en alguna de las fases mencionadas.

5.2.2.1. Análisis y diseño

Durante las fases de análisis y diseño del sistema y de cada uno de sus componentes, se tendrá en cuenta el requisito de seguridad, aunque no figure entre los requisitos de usuario. La seguridad ya formará parte del sistema ya desde el diseño.

1. PC08. Establecer un marco de referencia en el que se pueda establecer la seguridad como la base de diseño en todo el ciclo de vida de la solución implementada.

2. PC09. Aplicar el principio de menor privilegio, por el que se asignarán privilegios o permisos a los roles de usuarios del sistema mínimos indispensables para realizar su función.
3. PC010. Verificar que los controles de seguridad establecidos son fiables, reutilizables, fáciles de gestionar y están bajo el control del personal correspondiente.
4. PC011. Realizar revisiones del diseño de forma regular, para encontrar posibles vectores de ataque.
5. PC012. Establecer los requisitos de seguridad del sistema.
6. PC013. Realizar evaluación de riesgos del sistema.

5.2.2.2. Desarrollo e implementación

Durante el desarrollo e implementación del sistema, se establecerán los siguientes procedimientos:

1. PC014. Cada vez que se consoliden cambios relativos a cualquier aplicación del sistema, se realizarán una serie de operaciones sobre el código fuente:
    a. PC014.A. Se comprueba si el código repositado compila correctamente.
    b. PC014.B. Se comprueba si las pruebas unitarias asociadas al código modificado compilan y se ejecutan correctamente.
    c. PC014.C. Se comprueba si el código presenta *warnings* o posibles *bugs* que han pasado desapercibidos al compilador.
    d. PC014.D. Se realizan una serie de pruebas automáticas, destinadas a comprobar diferentes aspectos de la aplicación del sistema. En este caso se comprueba que no existen vulnerabilidades y si las hay, se intentan explotar (se generan y ejecutan *exploits*).
    e. PC014.E. Se analizan las posibles dependencias con otros elementos del sistema y se comprueba si los cambios generados son compatibles o entran en algún conflicto con las versiones actuales instaladas.
2. PC015. Si todas las pruebas son satisfactorias, se ejecuta una lista de pruebas sobre un simulador o mejor sobre un equipo de pruebas, verificando el cumplimiento de todos y cada uno de los requisitos de usuario y sistema, además de los requisitos de seguridad establecidos por el proveedor. Es importante añadir a dicha lista de pruebas, las correspondientes a los cambios realizados.
3. PC016. Se comprueba (si no se ha incluido en el paso anterior) si se dejan sin utilizar ficheros en el equipo que eran necesarios en versiones anteriores y en las sucesivas actualizaciones ya no lo son. Si se da el caso, durante el proceso de actualización se añadirá un paso adicional para borrar estos ficheros.
4. PC017. Se comprueba (si no se ha comprobado en el paso 2) qué ocurre si se apaga de forma abrupta un equipo durante el punto crítico de una actualización de software, que es cuando se están sobrescribiendo los ficheros correspondientes a la versión anterior.
5. PC018. Si el equipo queda inutilizado tras la ejecución de esa prueba, y siempre que sea posible, proporcionar mecanismos para poder volver a una versión segura que permita la actualización remota.

En caso de que el proveedor subcontrate los desarrollos, el subcontratista se tendrá que encargar de entregar, junto con el software, todos los informes y certificados que permitan atestiguar que se han efectuado todas las pruebas pertinentes.

5.2.2.3. Testing y aceptación

En esta fase se pone el foco en las pruebas y en conseguir que el contratista dé el visto bueno al sistema, en consonancia con el hito o hitos establecidos en el contrato o acordados entre ambas partes:

1. PC019. Se definen de forma conjunta con el contratista una lista de pruebas que permitirán comprobar que el sistema se ajusta y cumple con los requisitos de usuario.
2. PC020. El proveedor, normalmente en presencia del contratista, procederá a la realización de las pruebas definidas. El alcance de las pruebas, y si las pruebas podrán o no cambiar durante la misma sesión, se acordará previamente.
3. PC021. El proveedor, además de la realización de las pruebas definidas, podrá aportar resultados de pruebas definidas y realizadas de manera adicional, como valor añadido.
4. PC022. Entre las pruebas definidas deberían tenerse en cuenta pruebas E2E o Extremo a Extremo para verificar la operativa del sistema.
5. PC023. Entre las pruebas definidas, sería deseable incluir pruebas que permitan comprobar el nivel de seguridad del sistema. Debería en otras palabras efectuarse una auditoría de seguridad, incluyéndose pruebas con ataques simulados o exploits de las partes más vulnerables del sistema.
6. PC024. Deberían incluirse por tanto pruebas de seguridad o exploits de (al menos):
    a. PC024.A. Acceso a la red WiFi de las cocheras.
    b. PC024.B. Acceso a red embarcada del vehículo con ataques a través de la interfaz de red móvil del router (red de datos móviles).
    c. PC024.C. Acceso a la red embarcada del vehículo con ataques desde la conexión WiFi al pasaje.
    d. PC024.D. Acceso a red embarcada del SIU en las paradas de las marquesinas.
    e. PC024.E. Acceso a la red del Centro de Control.

5.2.2.4. Despliegue e integración

Esta fase es la más crítica, puesto que se comienza a desplegar el software del sistema, y es cuando se pueden manifestar errores o vulnerabilidades no consideradas hasta el momento. Además, en caso de que se hayan cometido errores en las fases de validación, pruebas y monitorización de vulnerabilidades de las versiones, podría instalarse software con errores importantes de seguridad que permitan el ataque de hackers, malfuncionamientos que permitan el fraude de manera subrepticia, o bien errores que generen reclamaciones de los usuarios o incluso demandas judiciales, o en todo caso, fallos graves que puedan desembocar en pérdidas económicas para el contratista y que bien podrían suponer la aplicación de penalizaciones para el proveedor. En casos muy graves, podría suponer la anulación del proyecto por incumplimiento de contrato.

Una vez que se da el visto bueno para el software desarrollado, el proveedor se ha de encargar de desplegarlo de la mejor manera posible, esto es, haciendo lo que esté en su mano para evitar errores o en caso de que ocurran, actuando a la mayor brevedad con contramedidas o acciones mitigadoras.

Se definen una serie de pasos cuyo seguimiento se recomienda:

1. PC025. Se ha de preparar dos entornos en el Centro de Control. El entorno de Producción y el de Preproducción, que preferiblemente estará en el mismo cluster o en la misma red de ordenadores, en un servidor de iguales o similares características al de Producción, ya que lo ideal es que fuese una réplica del entorno de Producción.

2. PC026. Comenzar el despliegue de forma escalonada, no todos o varios subsistemas a la vez.
3. PC027. Instalar la actualización en un solo equipo y verificar si la operativa es correcta; in situ si es necesario. Lo preferible sería realizar las pruebas en un bus virtual o en un bus real pero sin servicio real al usuario.
4. PC028. En caso de actualizar un equipo del Centro de Control, las pruebas pueden realizarse más cómodamente en Preproducción y si todas las pruebas son satisfactorias, se procederá a instalarlo en el entorno de Producción.
5. PC029. Probar la integración con todos los subsistemas, tanto a nivel embarcado como a nivel del Centro de Control.
6. PC030. Si es posible, dejar instalado el nuevo software en un solo elemento del subsistema correspondiente durante varios días, realizando un seguimiento para comprobar que la operativa es correcta, y que responde correctamente a situaciones que se salgan de la norma (denominados *corner cases*).
7. PC031. Si se observan errores, se procederá a la resolución de los mismos, repitiendo los pasos enumerados.
8. PC032. Si no se observan errores, se podrá actualizar otro elemento del subsistema, o actualizar más equipos, hasta tener una muestra significativa sobre la que poder hacer un seguimiento y cubrir todas las posibles casuísticas.
9. PC033. Tras un tiempo prudencial que podrá variar en base a la envergadura y complejidad del subsistema actualizado (desde unas horas para una aplicación del Centro de Control, a varias semanas en el caso del SVV embarcado), si todo se ha desarrollado correctamente, se podrá proceder al despliegue en toda la flota o la actualización en Producción (en caso de actualización de Software para el Centro de Control).
10. PC034. Tras la actualización en toda la flota, se realizará un seguimiento exhaustivo diario y proactivo, sin esperar a que el contratista notifique incidencias. Se revisarán logs, la actividad de los subsistemas y los equipos, reportes de los diferentes subsistemas, y si los hubiera, logs que genere el propio sistema operativo para comprobar que el equipo responde correctamente, no solamente a nivel software pero también a nivel de sistema operativo, e integración con el hardware si éste fuese nuevo o hubiese sufrido modificaciones para su funcionamiento en el sistema.

5.2.2.5. Mantenimiento y entrega

En la fase final, se entregará el sistema completo al contratista, con todos sus componentes y subsistemas operativos y funcionando en base a los requisitos acordados.

Aunque el sistema esté operando normalmente, no está exento de incidencias y vulnerabilidades que hay que vigilar, detectar y subsanar. Es decir, se ha de realizar un mantenimiento del sistema, independientemente de si lo hace el proveedor o el contratista.

1. PC035. Se ha de llevar un registro de las incidencias, para poder catalogarlas según la criticidad o el riesgo que suponga para la integridad del sistema, y el nivel de dificultad que supone.
2. PC036. Se ha de poder llevar un seguimiento de estas incidencias y actualizar su estado de forma regular, según su nivel de criticidad.
3. PC037. Se ha de llevar el control y monitorización de lo que se haya identificado como KPI (Key Performance Indicator); de modo que ante cualquier anomalía en sus valores, se procederá a revisar el subsistema afectado y a identificar la incidencia.

4. PC038. Se han de realizar auditorías al sistema para verificar su operativa correcta y su robustez, para evitar en la medida de lo posible la aparición de errores que pudieran trascender a los usuarios finales.
5. PC039. Dichas auditorías han de incluir un listado con pruebas para verificar la robustez del sistema ante ataques de seguridad.
6. PC040. Se han de mantener todos los equipos actualizados, incluyendo la instalación de parches sencillos ya que pueden ayudar a evitar ataques o pueden contribuir a mejorar la robustez del sistema.
7. PC041. En todo momento se han de realizar tareas que permitan un mantenimiento proactivo (es decir, sin esperar las notificaciones del contratista), hasta alcanzar un nivel de incidencias manejable o incluso despreciable (nunca será 0%).
8. PC042. Se han de detectar los elementos más vulnerables del sistema para establecer tareas de mantenimiento preventivo, que eviten incidencias cuya solución será más costosa. Por ejemplo, monitorizar el espacio de disco disponible en los servidores, los errores de disco en los arranques de los equipos, errores de conexión o los logs de la actividad de los firewalls, así como los registros de actividad del software de protección antivirus de los equipos del Centro de Control.

5.2.3. Procedimientos corporativos de seguridad

PC043. Se han de establecer políticas de seguridad corporativas en todas las entidades implicadas (proveedores, subcontratas y contratistas).

5.2.3.1. Plan de comunicación de medidas de seguridad

PC044. Se ha de establecer un plan que permita distribuir a todo el personal de forma eficiente, los cambios en las medidas relativas a la seguridad. Tanto las políticas establecidas, como cualquier cambio de las mismas o actualización.

PC045. Es tremendamente importante, no solo establecer medidas de seguridad adecuadas, si no también poder transmitirlas de manera eficiente a todo el personal; y esto es especialmente complicado cuando se trabaja con grupos de personas muy diferentes.

5.2.3.2. Control para prevenir la revelación de información

Se han de establecer controles adecuados para evitar fugas de información. Se pueden establecer varios mecanismos:

1. PC046. La empresa, previo aviso a sus trabajadores y si así figura en el contrato del trabajador, podrá llevar a cabo controles razonables para controlar esta pérdida o fuga de información. Llegado el caso, si estos controles no figurasen como posible mecanismo de control legítimo, la empresa podría promover la firma de adendas al contrato de cada trabajador para legitimar estos controles. Los cuales podrán ser:
    a. PC046.A. Búsqueda de cabecera de correos en el buzón corporativo.
    b. PC046.B. Consulta aleatoria de correos completos en el buzón corporativo, sobre todo aquellos destinados a personal externo.
    c. PC046.C. Monitorización de actividad a través de los mecanismos al alcance de la empresa (video vigilancia, control de las sesiones a las cuentas de cada trabajador, control de actividad de las aplicaciones utilizadas, etcétera).
    d. PC046.D. Control o escucha de llamadas telefónicas y mensajería.
    e. PC046.E. Control de chats corporativos.
2. PC047. Cambiar contraseñas de forma sistemática. Por ejemplo cada 45 días, no pudiendo repetir las 10 anteriores.

3. PC048. Durante los recesos bloquear la sesión o el HMI.
4. PC049. Evitar anotar datos sensibles y dejarlos a la vista en el puesto de trabajo (números de teléfono, contraseñas, números de tarjeta, pines, etcétera).
5. PC050. Evitar dejar a la vista en el puesto de trabajo documentos con información sensible o datos personales de clientes o proveedores.
6. PC051. En una llamada telefónica, si no se puede evitar que terceras personas puedan oír la conversación, se ha de evitar hacer referencia (al menos de manera evidente) a cualquier dato personal o bancario.
7. PC052. Destruir los documentos antes de arrojarlos a la papelera.

5.2.3.3. Disponibilidad de documentación y circulares

PC053. La empresa ha de asegurar la disponibilidad de los documentos oficiales y las circulares, además de transmitirle al personal cómo podrá acceder a dichos documentos.

PC054. En caso de que fallen los recursos telemáticos, se han de proporcionar métodos de respaldo para permitir a todo el personal acceder a la documentación requerida.

5.3. Tecnologías

En este dominio se establecerán prácticas recomendadas que consistirán en el establecimiento de estrategias que harán uso de la tecnología al alcance.

5.3.1. Control de Acceso

Se han de establecer estrategias que permitan controlar el acceso al personal autorizado, y así evitar accesos no permitidos.

5.3.1.1. Robustez de credenciales

Las credenciales robustas dificultan los ataques por fuerza bruta.

1. TC01. Los ID de usuario asignados a los controladores (Centro de Control), normalmente son alfanuméricos y siguen una lógica predecible (por ejemplo primera letra del nombre, y luego tres letras de cada apellido: *ddommar*).
2. TC02. Debido a lo anterior, las contraseñas elegidas deberán cumplir una serie de condiciones:
    a. TC02.A. Mínimo de 8 caracteres
    b. TC02.B. Se utilizarán minúsculas
    c. TC02.C. Se utilizarán mayúsculas
    d. TC02.D. Se utilizarán números
    e. TC02.E. Se utilizarán símbolos
    f. TC02.F. No se podrá repetir ninguna contraseña ya utilizada.
    g. TC02.G. Se evitarán palabras localizables en diccionarios.
3. TC03. Los login y password de usuario (conductores, inspectores, supervisores y mantenedores) para el acceso al sistema embarcado son numéricos. Por tanto, han de tener una longitud mínima y una dificultad mínima suficientes para que un ataque manual (usuario malintencionado que aproveche la ausencia del usuario en el autobús) tarde lo bastante como para ser disuasorio.
4. TC04. De lo anterior se establece que ID y password tendrán 7 dígitos de longitud.
5. TC05. Del punto 3 se deduce que no se podrán utilizar login y password con poca complejidad, no se podrán utilizar combinaciones numéricas sencillas como por ejemplo ‘1111111’.

6. TC06. Desde los servidores se entiende que se podrá acceder a los equipos de los subsistemas a través de sesiones SSH o SFTP.
7. TC07. Siempre que sea posible, se utilizará el sistema de claves asimétricas (clave privada y pública, en uso de algoritmos como RSA) para el acceso a los equipos por SSH o SFTP, siempre que dichas claves ya se encuentren alojadas en la memoria de los equipos.
8. TC08. Se ha de evitar la transferencia de claves privada / pública a los equipos embarcados.
9. TC09. Si no se puede garantizar la entrega de los equipos con las claves ya alojadas en su memoria, se usarán credenciales a través de user y password.
10. TC010. Se creará un usuario específico para evitar el uso de user conocidos, como root, admin, etcétera. Por ejemplo, "*uati0t*" (User Administrator Transport IoT).
11. TC011. Se elegirá una contraseña siguiendo si es posible los condicionantes establecidos en el paso 2. Por ejemplo *"_^Was1x)9oP*-"*.

5.3.1.2. Almacenamiento seguro de credenciales

Utilizar mecanismos de almacenamiento seguros para guardar las credenciales de los usuarios.

1. TC012. Almacenar el hash de la contraseña en lugar de la contraseña en claro en las bases de datos.
2. TC013. Para poder visualizar las contraseñas y modificarlas, solamente el administrador o un usuario con privilegios podrá consultar y modificar esta información.
3. TC014. Se puede plantear el uso de unidades fantasma o cifradas para almacenar la información de las credenciales del sistema.

5.3.1.3. Control de acceso a recursos críticos

TC015. Para el acceso a algunos recursos, equipos, o instalaciones, se habrá de hacer un control más exhaustivo, estableciendo controles biométricos, o varias etapas de autenticación (usuario, contraseña, además de pin y/o número de verificación). Fortalecer estos mecanismos con firewalls o reglas de enrutado siempre que sea posible. Incluir también reglas anti-spoofing en los firewalls.

TC016. Se podría establecer este control para acceder al centro de control y sortear el firewall ya que es uno de los puntos de entrada al sistema.

5.3.2. Software de terceros

5.3.2.1. Actualización y parches de seguridad

Se han de utilizar librerías o paquetes de software de terceros que hayan recibido parches de las últimas vulnerabilidades conocidas.

1. TC017. Se han de establecer estrategias de actualización que permitan la actualización de estos paquetes y librerías de terceros a la mayor brevedad tras la liberación de cada release oficial.
2. TC018. El estado del Centro de Control es más fácilmente controlable, puesto que está conectado a internet, y puede gestionarse mejor la actualización de estos paquetes y librerías.
3. TC019. En lo que respecta a los equipos embarcados, el uso del gestor de paquetes rpm podría solucionar este problema, ya que es una herramienta potente y muy fácil de usar.

4. TC020. No obstante, dado que existe diversidad en las versiones del kernel de Linux (que como se ha comentado es el sistema operativo usado para los equipos embarcados), será complicado abordar dos problemas:
    a. TC020.A. Usar la misma versión de estos parches y paquetes para todo el sistema.
    b. TC020.B. Decidir entre usar la última versión en aquellos equipos donde sea compatible y usar una versión anterior en aquellos con un sistema operativo más viejo, o bien usar la versión más vieja compatible a todos los equipos, pero con posibles agujeros de seguridad o vulnerabilidades sin resolver.
    c. TC020.C. Lo ideal, por facilitar el mantenimiento sería utilizar la misma versión en todos los equipos. Se harán excepciones en aquellos casos en los que las versiones más nuevas resuelvan vulnerabilidades más graves.

5.3.2.2. Uso de plataformas de terceros con soporte a largo plazo

Se ha de priorizar el uso e instalación de software de terceros que reciba un soporte a largo plazo.

5.3.3. Comunicaciones seguras

TC021. Se han de establecer estrategias de diseño para asegurar comunicaciones seguras, tanto a nivel embarcado, como entre cada subsistema, también con el Centro de Control.

5.3.3.1. Topologías de red seguras

TC022. Se han de utilizar topologías de red que permitan proteger el sistema de forma sencilla. Se deberán evitar el uso de sistemas de red abierta con IP pública. En este caso, se favorecerá el uso de redes virtuales privadas o VPNs, en las que para acceder hay que disponer del certificado necesario o conocer las credenciales de acceso, además de la IP y el puerto.

TC023. En caso de que no sea posible utilizar las VPNs, se emplearán mecanismos que permitan camuflar la topología de red al exterior, como sistemas de servidores proxy o DNS. De este modo, desde el exterior de la red, solamente se podrá ver la IP pública externa del proxy, no de los equipos que operan dentro del sistema.

5.3.3.2. Uso de protocolos seguros

TC024. Se han de utilizar protocolos seguros para las comunicaciones. Un protocolo seguro es aquel que permite la transmisión de información cifrada, garantizando también la integridad de los datos transferidos y la propiedad de autenticación entre dos extremos.

TC025. Aunque suponga ralentizar las comunicaciones, se empleará un protocolo (abierto basado en Transmodel [[RSGEN5]]) añadiendo cifrado de datos + hash.

5.3.3.3. Uso de algoritmos de cifrado conocidos

TC026. Se han de usar algoritmos de cifrado y hash conocidos y efectivos, preferiblemente aquellos que ofrezcan mayor robustez de cifrado en compromiso con el esfuerzo computacional necesario, considerando los equipos en los que se va a emplear. Como ejemplos de algoritmos conocidos de cifrado, se podrían emplear TDES [82] o Blowfish [83]; y ejemplos de algoritmos conocidos de hash o cálculo de firmas, MD5 o SHA-512 [84].

5.3.3.4. Implementación de interfaces Web Seguras

TC027. Los servicios Web del sistema han de implementarse sobre HTTPS, y además se usará seguridad adicional, autenticación requerida y autorización.

TC028. Es decir, se requerirá, además del certificado necesario para acceder al servicio Web sobre HTTPS, autenticación a través de clave asimétrica o bien con user y password, verificándose también los privilegios del usuario que realiza la petición al servicio.

5.3.4. Programación de código seguro

Durante la fase de implementación y desarrollo del sistema, el proveedor llevará a cabo una serie de prácticas recomendadas.

5.3.4.1. Prácticas de implementación de código seguro

1. TC029. Se han de emplear herramientas de gestión de tareas, para poder controlar todas las incidencias del sistema, de modo que cada incidencia permita realizar un seguimiento y pueda llevar asociado un parche o desarrollo vinculado. Se utilizarán herramientas como REDMINE o JIRA.
2. TC030. Cualquier input que proceda del exterior, ya sea por interacción con el usuario a bordo, usuario final, controlador o incluso personal técnico de la empresa proveedora, pasará por un proceso de validación antes de aceptarlo y procesarlo en el módulo software del subsistema que corresponda.
3. TC031. Las queries de base de datos que se realicen desde una aplicación se parametrizarán y escaparán correctamente para evitar problemas de inyección de código (SQL, XSS [77]).

5.3.4.2. Capacidad de auditar la operativa

TC032. Cualquier actividad que se realice en cualquier subsistema susceptible de ser auditada, se ha de logar o registrar en logs, para poder consultar la información de dicha actividad, que ayudará a localizar y solventar los errores que pudieran producirse.

5.3.4.3. Principios de seguridad desde el diseño y por defecto

TC033. Se han de seguir los principios de seguridad por defecto y ya desde el diseño.

5.3.4.4. Implementar técnicas de desarrollo software

TC034. Desarrollar las aplicaciones pensando no solamente en la operativa, sino también en las pruebas y el mantenimiento a realizar de dicho código.

5.3.4.5. Verificar la generación de código

TC035. Realizar los desarrollos y las pruebas en entornos lo más parecidos posibles al entorno de producción. Se utilizarán testbenches o bancos de pruebas que repliquen el hardware y la instalación realizada en un autobús; por otro lado, se usarán entornos de validación y pruebas, y entornos de preproducción simulando el Centro de Control antes de consolidar los cambios en el sistema en operación.

5.3.4.6. Asegurar la instalación de parches y actualizaciones.

1. TC036. Cada vez que se consoliden cambios de código, se han de realizar los siguientes pasos:
    a. TC036.A. El entorno de integración continua se descargará los cambios y compilará todos los paquetes software.
    b. TC036.B. El entorno de integración continua ejecutará conjuntos de pruebas automáticas predefinidas con el objeto de verificar si las funcionalidades ya operativas no se ven afectadas.
    c. TC036.C. El entorno de integración continua ejecutará conjuntos de pruebas que intenten verificar la seguridad y la robustez del sistema y si se han añadido nuevas vulnerabilidades.

2. TC037. Asegurar y comprobar los parches y las actualizaciones antes de aplicarlas. Verificar no solo que no se está afectando a funcionalidades ya operativas, si no que no se ven afectados otros componentes del sistema u otros subsistemas.

5.3.4.7. Implementar medidas contra la inserción de código malicioso y control anti fraude

TC038. Establecer planes de contingencia en caso de que los cambios introducidos generen incidencias o problemas al sistema, y también en casos en los que se haya inyectado un payload malicioso que infecte el equipo. Se podrá optar por deshabilitar o apagar ciertas funcionalidades de los subsistemas, para evitar problemas más graves. Por ejemplo, si por error se despliega una versión de parámetros con precios a 0€ en todas las líneas y para todos los medios de pago, el sistema podría restringir estas ventas, y no permitir ninguna venta de títulos de transporte, restaurando la versión de configuración anterior.

5.3.4.8. Implementar controles para evitar la modificación de código y parametrizaciones

TC039. Establecer mecanismos para evitar manipulaciones no deseadas de software o parametrizaciones. Se añadirá a cada fichero de configuración una firma o CRC, de modo que si un ataque modifica un fichero de parámetros o éste fichero llega corrupto al equipo, el software podrá detectarlo y no cargarlo o regenerar un fichero con parámetros por defecto.

5.3.5. Control Anti fraude

Se han de establecer medidas para controlar los intentos premeditados y no premeditados de usuarios del sistema para aprovechar las vulnerabilidades del sistema para su propio beneficio, y más concretamente del subsistema de SVV, que es el punto de entrada del dinero para la empresa de transporte y por tanto un punto operacional crítico. Estos ataques se reducen a que el atacante pretende disfrutar del servicio sin pagar y de forma subrepticia, o incluso hacer negocio explotando una vulnerabilidad del sistema.

Los medios de pago más susceptibles en este caso serían las tarjetas Mifare, el pago a través de QRs, aunque también el pago en efectivo. La tecnología de las tarjetas Desfire con SAM se considera suficientemente robusta como para que de momento sea disuasoria para atacantes, más todavía si se emplea SAM [63].

5.3.5.1. Control de transacciones de tarjetas

TC040. Se ha de controlar la trazabilidad de las transacciones, es decir, cada transacción debe tener una transacción predecesora y una sucesora, sin saltos que no puedan explicarse. Si esto ocurre (por ejemplo, una diferencia de saldo entre dos operaciones de validación de tarjeta sin una operación de recarga entre medias), podría deberse a un error en el algoritmo de validación, de recuperación de la tarjeta, o a un intento de fraude. Se añadirán controles en el Centro de Control, para poder detectar estas situaciones.

5.3.5.2. Bloqueo o inutilización de tarjetas

TC041. Si se detecta un caso de fraude, o simplemente un usuario pierde una tarjeta, el sistema ha de permitir la inserción de los UIDs de dichas tarjetas en una lista de tarjetas “sospechosas”. De este modo, no se permitirá a ningún usuario utilizar estas tarjetas para disfrutar del servicio del sistema, ni tampoco al usuario legítimo en caso de que haya encontrado la tarjeta.

5.3.5.3. Cambio de claves

TC042. Para dificultar la tarea del atacante, el controlador del sistema podrá modificar las claves a utilizar en las tarjetas. No obstante, si éste ha sido capaz de piratear la tarjeta reventando las claves, podrá hacerlo de nuevo [62].

5.3.5.4. Diversificación de claves

TC043. Sería posible no utilizar claves fijas A y B iguales para todas las tarjetas, sino utilizar una fórmula matemática que permitiría emplear claves diferentes por tarjeta. La ventaja de este método es que se obliga al atacante a piratear la tarjeta y descubrir las claves cada vez que éste tenga que usar una tarjeta diferente (con UID diferente, por tanto), mientras que con claves fijas, una vez que las descubra, podrá tener acceso a todas las tarjetas. Para disuadir al atacante, se ha de combinar con el uso de listas negras de tarjetas.

5.3.5.5. Control de fraude de las ventas en efectivo

Este tipo de fraude consiste en que el conductor reutiliza los tickets ya usados de otros viajeros o utiliza cualquier artimaña para conseguir que los viajeros le paguen un dinero que no va a la empresa de transportes, si no directamente a su bolsillo, por lo que merece la pena establecer ciertos controles o precauciones:

1. TC044. Añadir sistema CCTV a bordo.
2. TC045. Realizar inspecciones más a menudo.
3. TC046. Instalar sensores de contaje independientes del sistema SVV y gestionar la información de dichos sensores desde el Centro de Control.
4. TC047. Controlar desde el centro de control los viajeros subidos y bajados en cada parada usando los sensores mencionados.
5. TC048. Mostrar por pantalla del HMI del SVV la ocupación de viajeros para que un inspector pueda consultar y verificar la información a simple vista.
6. TC049. Realizar campañas de información al viajero para que reclame su billete y compruebe que todos los datos son coherentes, incluyendo la hora de emisión del mismo.

5.3.5.6. Control de fraude de ventas con código QR

Este tipo de fraude lo lleva a cabo el viajero, que intenta reutilizar el mayor número de veces un QR que se ha pagado una sola vez, o incluso del que se han podido distribuir copias. También se tiene en cuenta la posibilidad de que el QR se haya podido manipular o regenerar con datos más convenientes para el viajero malintencionado.

1. TC050. Establecer caducidades relativamente cortas, admitiendo cierto margen para el fraude. Se podrá establecer por tanto desde su emisión, una caducidad para el billete de QR de varios minutos o pocas horas.
2. TC051. Habilitar un Web Service en el Centro de control para que los sistemas SVV puedan consultar si un QR con un código determinado ya se ha utilizado en otro SVV de la flota. Esto sería posible como en este caso siempre que se envíen las transacciones en tiempo real al Servidor y se consoliden en Base de Datos.
3. TC052. Cifrar y firmar los datos del QR y añadir dicha firma al propio QR para evitar o dificultar que pueda ser manipulado, regenerado y redistribuido.

5.3.5.7. Control de fraude de ventas con tarjeta bancaria (tecnología EMV)

Este tipo de fraude se basa en la operativa de la infraestructura de red bancaria necesaria para el funcionamiento del pago con tarjetas bancarias a bordo del vehículo a través de tecnología EMV [85].

Si en el momento de efectuar la validación, no es posible realizar esta conexión con la red bancaria, se da de paso la transacción y se considera el pago realizado, aunque no se hubiera podido verificar.

1. TC053. Establecer un control de cobertura de datos móviles, para determinar los puntos ciegos en los recorridos de los autobuses.
2. TC054. En caso de que haya fallos reiterados de cobertura, se podrá optar por desactivar o inhabilitar el lector de tarjetas bancarias para evitar el fraude.
3. TC055. Realizar un control de transacciones de las tarjetas bancarias, en los puntos y momentos en los que se pierda la cobertura de datos móviles.
4. TC056. Determinar si hay usuarios que han usado su tarjeta bancaria sin tener dinero en la cuenta o sin tener dinero disponible a crédito, para poder añadirlas a una lista negra que también se comparta con la infraestructura bancaria, para bloquear estas tarjetas e impedir su utilización, al menos para acceder al servicio proporcionado por el sistema.

5.3.6. Revisiones de seguridad

5.3.6.1. Revisiones de código

TC056. Asegurar que durante las fases SLDC, el código se revisa en cuanto a términos, requisitos y operativa de seguridad antes de aceptarlo.

5.3.6.2. Realizar análisis de superficie de ataque

TC057. Asegurar que durante las fases SLDC se realizan análisis sobre el conjunto de posibles vectores de ataque y se documenta debidamente.

5.3.6.3. Realizar tests SLDC IoT

TC058. Asegurar que durante las fases de SLDC se realizan tests de penetración cuando se completa el software. Se deberán llevar a cabo tests adicionales en base a las necesidades funcionales y a las evaluaciones de riesgos realizadas.

5.3.6.4. Diseño de plan de contingencia

TC059. Realizar el diseño de un plan de contingencia que esté alineado con los desarrollos software, y que impida que se interrumpa el desarrollo normal de las fases SLDC.

5.3.6.5. Seguimiento de los requisitos

TC060. Se ha de revisar y monitorizar los requisitos establecidos cada cierto tiempo para asegurar el éxito de las fases de implantación del proyecto.

5.3.7. Seguridad en todas las fases del proyecto

5.3.7.1. Monitorización y registro seguro de logs

TC061. A lo largo de todas las fases SDLC de implantación del proyecto se ha de asegurar que los logs generados por los módulos de todos los subsistemas se almacenan de forma segura y permiten su recuperación de forma sencilla. Se podrán guardar en ficheros .zip con contraseña, siguiendo las prácticas recomendadas en TC02.

TC062. Se deben registrar trazas en los logs que sean inteligibles y que permitan una monitorización eficaz de la actividad del sistema.

5.3.7.2. Sistemas de detección de condiciones físicas

Se deben implementar sistemas de detección de condiciones físicas (sensores de humedad, temperatura, humos, etcétera) en las instalaciones en las que se almacenen los equipos que alojen datos importantes y que por lo general están desatendidos.

1. TC063. En los habitáculos en los que se almacenan los equipos de las redes embarcadas (marquesinas y autobuses), sería adecuado instalar sensores de humedad y temperatura, y poder monitorizar estos valores desde el Centro de Control.

2. TC064. En el Centro de Control, sería adecuado instalar sensores de temperatura, humedad y humos en la sala de servidores, donde se almacenarían todos los equipos principales.

5.3.7.3. Plan de mitigación por daños físicos

Se ha de establecer una estrategia que permita dar continuidad al servicio aunque se produzcan daños físicos en cualquier elemento físico que permita dar aplicación a cualquier fase del proyecto.

1. TC065. Los entornos de validación y verificación de la empresa proveedora han de poder replicarse y poder tener respaldo en caso de que los primeros fallen.
2. TC066. Se ha de disponer de suficiente stock de los equipos del sistema para compensar incidencias o roturas de hardware.
3. TC067. En el Centro de Control, los servidores empleados han de tener un sistema de respaldo y de balanceo de carga.

5.3.7.4. Auditar el acceso a la infraestructura SDLC

Implementar controles para los accesos al sistema en cada fase del proyecto. Todos los accesos generarán un log que podrá descargarse y consultarse. Esto implicará:

1. TC068. Todas las reglas de los firewalls configurados en los equipos generarán logs que podrán consultarse.
2. TC069. Los intentos de acceso al sistema embarcado se almacenarán en el equipo y enviarán como una transacción en tiempo real al centro de control.
3. TC070. Los accesos o intentos de inicio de sesión en los puestos del operador se registrarán en el servidor y se podrá consultar la información.

5.3.7.5. Implementar protocolos de identificación

TC071. Tanto en las instalaciones del proveedor, como en las del contratista, se han de establecer protocolos de identificación para que el personal perteneciente a la empresa sea fácilmente detectable. Se usarán tarjetas personales que permitirán acceder a los recintos y distinguir al personal corporativo de los que son visitantes.

6. ESTUDIO DE VULNERABILIDADES

A continuación se recoge el estudio de las vulnerabilidades del sistema IoT de transporte en base a los sistemas de referencia o de clasificación de vulnerabilidades que se utilicen.

Se enumerarán las vulnerabilidades inherentes al sistema completo, también particularizando con los subsistemas, y las características más importantes de los elementos que los componen.

Se especificarán dichas vulnerabilidades en base al sistema de referencia comentado, para posteriormente analizarlas y estudiar los resultados.

6.1. Sistemas de clasificación de vulnerabilidades

Existen varios sistemas de clasificación de vulnerabilidades, pero sin duda los más populares son CVE [87] y Bugtraq [88]. CVE por su parte es una lista de vulnerabilidades comunes, que se apoya en otras listas como CWE (Common Weakness Enumeration, o lista de debilidades comunes), y en el momento de realización del presente trabajo, acumula un total de 136067 registros; Bugtraq por su parte, es una lista de distribución de correo, aunque también presenta un

buscador en el que se puede recuperar información sobre vulnerabilidades de su correspondiente base de datos.

6.1.1. Explicación de Sistemas de clasificación de vulnerabilidades o sistemas de referencia

6.1.1.1. Definición

Son sistemas que establecen un método de referencia para vulnerabilidades de seguridad informática bien conocidas, de modo que en una lista accesible a todos los usuarios de Internet se publica información de cada vulnerabilidad, pudiendo consultar el proveedor del paquete de software vulnerable, el producto software concreto y la versión afectada y por supuesto en qué consiste la vulnerabilidad, y dependiendo del sistema de clasificación se mostrará información adicional (como en el caso de Bugtraq), como por ejemplo si existe exploit, si el bug está ya solucionado, y en tal caso, qué proveedor, a partir de qué versión y a partir de cuándo, y por último una serie de referencias que permiten recopilar información asociada al paquete de software vulnerable, al exploit, a la solución o a otras vulnerabilidades o debilidades asociadas con dicho paquete de software. Se muestra a continuación un ejemplo [90]:

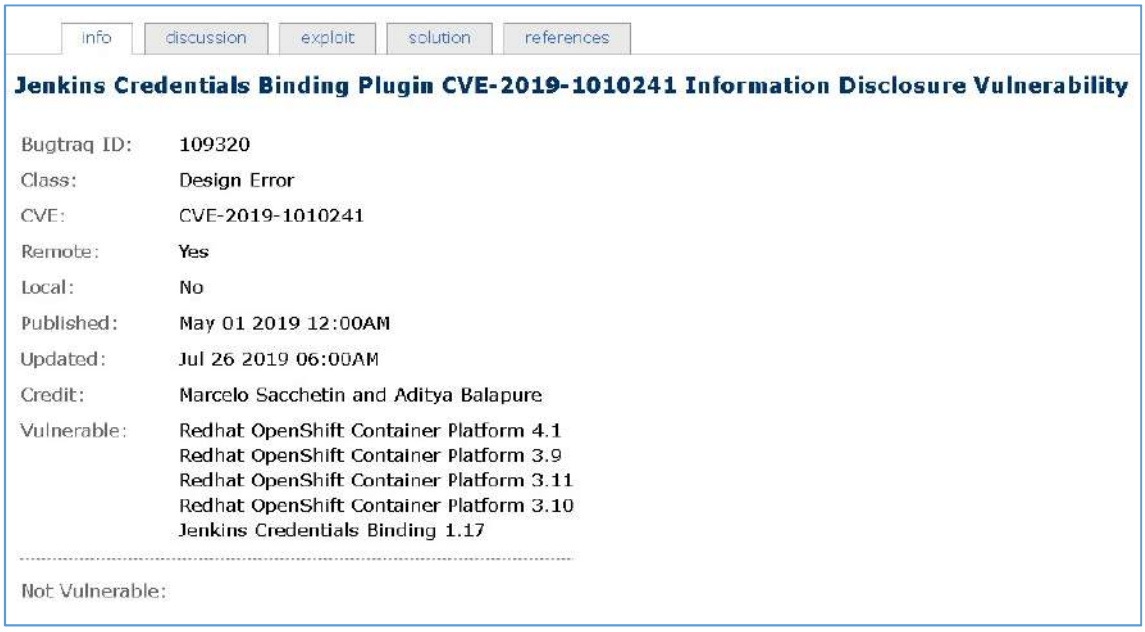

Figura 10: Captura de la información mostrada para la vulnerabilidad CVE-2019-1010241/BID-109320 en la web de Security Focus.

También existen entidades que integran toda la información y permiten visualizar la información comentada, y además la puntuación de la vulnerabilidad, es decir, una escala para ilustrar la peligrosidad de la vulnerabilidad. Es el caso del NVD (National Vulnerability Database), una organización gubernamental Norteamericana que dispone de un repositorio de información de vulnerabilidades, donde se muestra la información de cada vulnerabilidad, su relación con otras vulnerabilidades de otros sistemas de referencia (básicamente CVE con respecto a Bugtraq, u otros sistemas de referencia, como el de Microsoft, si se hace referencia a una vulnerabilidad de alguno de sus productos), y además muestra esta puntuación, en base al sistema CVSS (Common Vulnerability Scoring System) [91]:

Figura 11: Captura de parte de la información de la vulnerabilidad CVE-2019-1010241 / BID-109320, incluyendo la puntuación CVSS.

Se utilizarán todas las herramientas al alcance para obtener más información, los sistemas de referencia comentados (además, comentar que es frecuente que exista una relación entre ambos sistemas de referencia, es decir, que la misma vulnerabilidad pueda encontrarse en ambos sistemas o bases de datos), y las utilidades que permitan visualizar toda la información lo más convenientemente posible, como NIST NVD.

6.1.1.2. Formato del Identificador

Para diferenciar una vulnerabilidad de otra dentro del sistema de referencia, registro, lista o base de datos, se utiliza un identificador único. Dependiendo del sistema, este identificador tiene un formato u otro.

En CVE, el identificador es CVE-AAAA-XXXXX, donde AAAA es el año de inclusión en el registro, y XXXXX es un número arbitrario que no se haya repetido para ese año.

Respecto a Bugtrag, simplemente se utiliza un formato numérico BID-NNNNN, empleando numeración creciente sin repetir.

6.1.1.3. Sistema de puntuación o evaluación de la vulnerabilidad

Existe un sistema de puntuación o scoring, denominado CVSS (Common Vulnerability Scoring System), creada y gestionada por FIRST Org. Inc., que es el que se emplea en herramientas como la comentada, NVD para clasificar la vulnerabilidad en base a varios criterios, a los que se asignan un peso determinado o un porcentaje respecto del total [92]:

Métricas base

- Vector de acceso (AV): Describe cómo una vulnerabilidad puede explotarse.
- Complejidad de ataque (AC): Describe cómo de fácil o difícil es la vulnerabilidad de ser explotada.
- Autenticación (Au): Indica el número de veces que el atacante se ha de autenticar para realizar el ataque.

Métricas de impacto

- Confidencialidad (C): Describe el impacto en la confidencialidad de los datos procesados por el sistema.

- Integridad (I): Describe el impacto en la integridad del sistema.
- Disponibilidad (A): Describe el impacto sobre la disponibilidad del sistema. Esto describiría por tanto la posibilidad de realizar un ataque de denegación de servicio (DDoS).

Métricas temporales

- Explotabilidad (E): Describe el estado actual de las técnicas de explotación utilizadas.
- Nivel de remediación (RL): Permite que la puntuación de las métricas temporales de una vulnerabilidad bajen según se despliegan actualizaciones y nuevas versiones.
- Informe de confianza (RC): Mide el nivel de confianza de la propia existencia de la vulnerabilidad y de los detalles técnicos de la misma.

Métricas del entorno

- Daño colateral potencial (CDP): Mide el potencial impacto ya sea en elementos físicos o el impacto económico en la organización afectada si se explota la vulnerabilidad.
- Distribución del objetivo (TD): Mide la proporción de sistemas vulnerables en el entorno.
- Modificación del impacto de la subpuntuación: Tres métricas adicionales evalúan los requisitos específicos de seguridad para confidencialidad (CR), integridad (IR) y disponibilidad (AR), permitiendo el ajuste fino de la puntuación en atención al entorno de los usuarios.

La combinación de todos los parámetros que permiten obtener la puntuación de la vulnerabilidad se denomina vector CVSS.

Es interesante comentar que existe una herramienta Calculadora (existen realmente dos versiones de esta herramienta, la v2 y la v3) accesible en la web de NVD [93] que permite obtener las puntuaciones de las métricas comentadas, en base al vector CVSS de la vulnerabilidad. Así pues, si disponemos del siguiente vector CVSS [94]:

AV:L/AC:H/Au:S/C:P/I:P/A:P/E:POC/RL:TF/RC:UC/CDP:L/TD:L/CR:M/IR:M/AR:M

La puntuación con respecto a las métricas correspondientes es la siguiente:

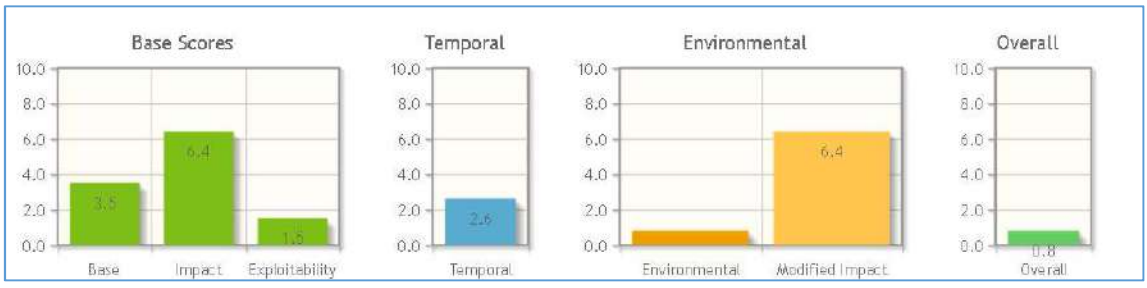

Figura 12: Resultado de las puntuaciones de cada métrica para el vector CVSS mencionado.

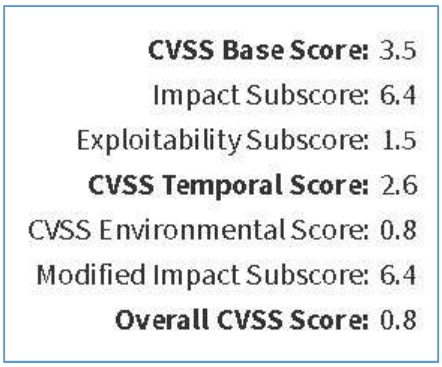

Figura 13: Resultado numérico de las puntuaciones para cada métrica con respecto al vector CVSS mencionado.

6.1.1.4. Coordinación entre los sistemas de clasificación existentes

Tal y como se ha comentado, una misma vulnerabilidad puede figurar en varios sistemas de referencia o bases de datos de vulnerabilidades. Existen sistemas de referencia en los que se muestra un mapeo de una base de datos a otra, como Security Focus / Bugtraq, tal y como se ha visto en el ejemplo mencionado en la Figura 10 del apartado 6.1.1.1.

Esta coordinación no es inmediata, y puede demorar varios días o meses, de modo que el hecho de que no se muestre esta relación o mapeo, no significa que la vulnerabilidad en cuestión no esté registrada en otros sistemas de referencia.

Por este motivo se tendrá especial cuidado al tratar con vulnerabilidades que se hayan registrado recientemente, puesto que puede ser que solamente se encuentren en un sistema de referencia, o en todos pero con el mapeo entre los sistemas de referencia aún pendiente.

6.2. Identificación de los elementos vulnerables de la IoT

A continuación se enumeran los elementos del sistema IoT de transportes modelado susceptibles de presentar alguna vulnerabilidad recogida en los sistemas de referencia mencionados.

6.2.1. Redes de comunicaciones

- Red abierta 4G con IPs públicas.

6.2.2. Sistema Operativo

- Los equipos embarcados trabajan con GNU/Linux, con kernel 2.6.35.2 para el SIUP y 4.9.218 para los demás.
- En el Centro de Control, los servidores trabajan con Windows Server 2019.
- Las estaciones de trabajo de la red del Centro de Control trabajan con Windows 10.

6.2.3. Firmware de los Routers

Estos son los posibles modelos de router a utilizar en el sistema:

- 880G de Cisco.
- RUT950 de Teltonika.
- LR77v2 de Advantech.

6.2.4. Herramientas software de seguridad

- Herramientas para los firewall de los routers (GNU/Linux):
  - iptables
  - firewalld ó firewall-cmd
- Herramientas para crear firewall con iptables:

- Firewall Builder
- Herramientas o suites antivirus:
  - Kaspersky Internet Security

6.2.5. Protocolos y Servicios

- Comunicación a través de la red móvil con el protocolo basado en el modelo de datos *Transmodel CEN* [69] con cifrado 3DES y hash MD5 de los datos.
- SFTP para la transferencia de ficheros.
- SSH para conexión remota (únicamente desde los servidores permitidos). Se utiliza la versión 2.
- Telnet
- Web Services sobre HTTPS

6.2.6. Dispositivos de almacenamiento

- Uso de disco duro externo para el SIUV, se asumirá que es un disco SSD, con firmware Intel Pro 5400s 2.5".

6.2.7. Tecnologías AFC (Automatic Fare Collection)

Serían las tecnologías empleadas para el cobro de los tickets de transporte en el SVV.

- Tarjetas Mifare Classic 1K.
- Tarjetas Mifare Desfire SAM AV2.
- Tarjeta emulada HCE Android sobre chip NFC.
- Códigos QR.

6.3. Listado de vulnerabilidades

Se utilizan los buscadores NVD NIST [95] y CVE Details [96] principalmente y se recogen las vulnerabilidades que podrían afectar al sistema. Para ciertos casos, no afecta directamente si no que habría que instalar o configurar el elemento al que se hace referencia; se menciona igualmente ya que resulta positivo tenerlo en cuenta para o bien evitar instalar elemento afectado, o bien instalarlo pero con los parches o los fixes aplicados:

ID ELEMENTO VULNERABLE	ID VULNERABILIDAD	CVSS SCORE V3.0	ELEMENTO AFECTADO	FIX EXISTENTE
6.2.1. Redes de comunicaciones	CVE-2017-11435	9.8 CRITICAL	Wi-Fi Router model HG100R-* 2.0.6	N/D
	CVE-2017-7405	9.8 CRITICAL	D-Link DIR-615 (antes de la versión v20.12PTb04)	Versión > v20.12PTb04
	CVE-2019-1003036	4.3 MEDIUM	Jenkins Azure VM Agents Plugin 0.8.0	Versión > 0.8.0
6.2.2. Sistema Operativo – GNU/Linux *kernel* 2.6.35.2 [97][3]	CVE-2018-20961	10.0 CRITICAL	Error en la función f_midi_set_alt de drivers/ usb/ gadget/ function/f_midi.c que permite DoS (denial of service)	Kernel > 4.16.4
	CVE-2019-11811	10.0 CRITICAL	Intento de acceso tras free a /proc/ioports.	Kernel > 5.0.4

[3] De acuerdo con [96] existen un total de **230** vulnerabilidades para el kernel 2.6.35.2. Se recogerán en la lista las 10 más graves (ordenadas por CVSS Score).

	CVE-2019-15292	10.0 CRITICAL	Intento de acceso tras free en atalk_ proc_ exit	Kernel > 5.0.9
	CVE-2019-15926	9.4 CRITICAL	Intento de acceso fuera de los límites en las funciones ath6kl_ wmi_ pstream_ timeout_ event_rx y ath6kl_ wmi_ cac_ event_rx in the file drivers/ net/ wireless/ ath/ ath6kl/ wmi.c.	Kernel > 5.2.3
	CVE-2018-20836	9.3 CRITICAL	Race condition que lleva a uso tras free en las funciones smp_ task_ timedout y smp_ task_ done en drivers/ scsi/ libsas/ sas_expander.c.	Kernel > 4.20
	CVE-2019-11815	9.3 CRITICAL	Race condition que lleva a uso tras free relativo a la limpieza del espacio de nombres de red.	Kernel > 5.0.8
	CVE-2011-2497	8.3 HIGH	Integer overflow en la función l2cap_config_req en net/ bluetooth/ l2cap_core.c provoca DoS y corrupción de memoria.	Kernel > 3.0
	CVE-2017-1000251	8.3 HIGH	La pila Bluetooth nativa es vulnerable a stack overflow.	Kernel > 4.13.1
	CVE-2011-0709	7.8 HIGH	Error en la función br_mdb_ip_get en net/ bridge/ br_multicast permite DoS.	Kernel > 2.6.35.5
	CVE-2011-1076	7.8 HIGH	Error en net/ dns_resolver/ dns_key.c permite que los servidores DNS causen DoS	Kernel > 2.6.38
6.2.2. Sistema Operativo – GNU/Linux *kernel* 4.9.218 [98][4]	CVE-2019-15926	9.4 CRITICAL	Intento de acceso fuera de los límites en las funciones ath6kl_ wmi_ pstream_ timeout_ event_rx y ath6kl_ wmi_cac_event_rx in the file drivers/ net/	Kernel > 5.2.3

[4] En CVE Details no se recogen las vulnerabilidades de la versión de kernel 4.9.218, de modo que se considerará la versión más cercana, la 4.9.190. Esta versión presenta un total de 68 vulnerabilidades, de las que se recogerán las 10 más graves (ordenadas por CVSS Score). Al menos, la vulnerabilidad CVE-2019-15926 es común a ambos kernels.

			wireless/ ath/ ath6kl/ wmi.c.	
	CVE-2019-15666	7.8 HIGH	Intento de acceso fuera de límites en __xfrm_policy_unlink que provoca DoS.	Kernel > 5.0.19
	CVE-2019-15807	7.8 HIGH	Pérdida de memoria (memory leak) en drivers/ scsi/ libsas/ sas_expander.c. Puede provocar DoS.	Kernel > 5.1.13
	CVE-2019-15916	7.8 HIGH	Memory leak en la función register_queue_kobjects en net/ core/ net-sysfs.c que provoca un DoS.	Kernel > 5.0.1
	CVE-2019-16994	7.8 HIGH	Memory leak en sit_init_net en net/ ipv6/ sit.c, que provoca un DoS.	Kernel > 5.0
	CVE-2019-16995	7.8 HIGH	Memory leak en hsr_dev_finalize en net/ hsr/ hsr_device.c, que provoca un DoS.	Kernel > 5.0.3
	CVE-2017-7482	7.2 HIGH	Uso incorrecto de las claves de RXRPC puede provocar error de corrupción de memoria y posible escalado de privilegios.	Kernel > 4.12
	CVE-2019-16939	7.2 HIGH	La implementación de volcado de datos de XFRM en net/ xfrm/ xfrm_user.c puede llegar a permitir escalado de privilegios y DoS.	Kernel > 4.13.11
	CVE-2017-17806	7.2 HIGH	La implementación HMAC (crypto/hmac.c) puede provocar overflow.	Kernel > 4.14.8
	CVE-2017-18075	7.2 HIGH	crypto/pcrypt.c no libera correctamente instancias, pudiendo provocar DoS.	Kernel > 4.14.13
6.2.2. Sistema Operativo – Windows	CVE-2018-8476	10.0 CRITICAL	Windows Deployment Services TFTP, puede provocar la ejecución de código arbitrario de forma remota.	N/D

Server 2019 [99][5]	CVE-2018-8626	10.0 CRITICAL	Los Servidores Windows Domain Name System presentan un problema por el que se puede código arbitrario de forma remota.	
	CVE-2019-1181 CVE-2019-1182 CVE-2019-1222 CVE-2019-1226	10.0 CRITICAL	Existe un error en la aplicación Remote Desktop Services por el que se puede ejecutar código arbitrario de forma remota.	N/D
	CVE-2018-8256	9.3 CRITICAL	Existe un error en el terminal PowerShell cuando gestiona ficheros cocinados específicamente por el que se puede ejecutar código arbitrario de forma remota.	N/D
	CVE-2018-8413	9.3 CRITICAL	Existe un error en Windows Theme API al no descomprimir correctamente algunos ficheros, por el que se puede ejecutar código arbitrario de forma remota.	N/D
	CVE-2018-8423	9.3 CRITICAL	Existe un error en Microsoft JET Database Engine, por el que se puede ejecutar código arbitrario de forma remota.	N/D
	CVE-2018-8432	9.3 CRITICAL	Existe un error en Microsoft Graphics Components, por el que se puede ejecutar código arbitrario de forma remota.	N/D
6.2.2. Sistema Operativo – Windows 10 [100][6]	CVE-2016-3236	10.0 CRITICAL	La implementación del protocolo Web Proxy Auto Discovery (WPAD) presenta un error que provoca que atacantes remotos redirijan el	N/D

[5] De acuerdo con CVE Details, Windows Server 2019 presenta un total de **405** vulnerabilidades. Se recogerán las 10 más graves (ordenadas por CVSS Score).

[6] De acuerdo con CVE Details, Windows 10 presenta un total de **1111** vulnerabilidades. Se recogerán las 10 más graves (ordenadas por CVSS Score). Al menos, las vulnerabilidades CVE-2018-8626, CVE-2019-1181, CVE-2019-1182 y CVE-2019-1222 son comunes a Windows Server y Windows 10.

			tráfico de red a través de vectores inespecíficos.	
	CVE-2016-3266	10.0 CRITICAL	Los drivers del modo kernel permiten a los usuarios locales la escalada de privilegios a través de una aplicación creada para tal efecto.	N/D
	CVE-2016-3270	10.0 CRITICAL	El componente Graphics en el kernel permite a los usuarios locales la escalada de privilegios a través de una aplicación creada para tal efecto.	N/D
	CVE-2016-7182	10.0 CRITICAL	El componente Graphics en el kernel permite la ejecución de código arbitrario a través de una fuente modificada True Type.	N/D
	CVE-2017-8543	10.0 CRITICAL	Permite que un atacante tome el control del sistema afectado cuando Windows Search falla.	N/D
	CVE-2017-8589	10.0 CRITICAL	Permite la ejecución de código arbitrario cuando Windows Search falla.	N/D
	CVE-2017-11771	10.0 CRITICAL	El componente Windows Search permite la ejecución de un código arbitrario cuando falla en gestionar correctamente las respuestas DNS.	N/D
	CVE-2018-8626	10.0 CRITICAL	Los Servidores Windows Domain Name System presentan un problema por el que se puede código arbitrario de forma remota.	N/D
	CVE-2019-1181 CVE-2019-1182 CVE-2019-1222 CVE-2019-1226	10.0 CRITICAL	Existe un error en la aplicación Remote Desktop Services por el que se puede ejecutar código arbitrario de forma remota.	N/D
6.2.3. Firmware de	CVE-2013-1241	6.3 MEDIUM	El módulo ISM no gestiona correctamente las cabeceras de	N/D

los Routers - Cisco 880G			autenticación de los paquetes, lo que permite a los usuarios autenticados remotamente causar DoS.	
6.2.3. Firmware de los Routers – Teltonika RUT950	CVE-2017-8116	10.0 CRITICAL	La interfaz de gestión permite a los atacantes remotos ejecutar comandos arbitrarios con permisos root.	Firmware Version > 00.03.265
	CVE-2018-19878	6.8 MEDIUM	La aplicación permite a un usuario realizar login sin limitación. Por cada login correcto, la aplicación almacena una sesión. Un usuario puede realizar re-login sin realizar logout, provocando que la aplicación almacene la sesión en memoria, lo que consume recursos del dispositivo.	Device NOT in (RTU950 R_31.04.59)
	CVE-2018-19879	5.0 MEDIUM	Problema en cgi-bin/luci, la funcionalidad de autenticación no está protegida en contra de herramientas automáticas que pueden realizar varios intentos de autenticación a la aplicación. Esto puede permitir que se lleguen a crackear unas credenciales.	Device R_31.04.89 > R_00.05.00 .5
6.2.3. Firmware de los Routers – Advantech LR77v2 (Advantech B+B Snartworx)	CVE-2017-7909	7.5 HIGH	La interfaz web utiliza JavaScript para comprobar la autenticación del cliente y redirigir a los usuarios no autorizados. Los atacantes pueden interceptar estas respuestas y saltarse la autenticación para acceder a páginas web restringidas.	Version > 1.5.2

6.2.3. Firmware de los Routers Estos son los posibles modelos de router a utilizar en el sistema: - 880G de Cisco. - RUT950 de Teltonika. - LR77v2 de Advantech. Herramientas software de seguridad - **iptables**	CVE-2019-11360	4.3 MEDIUM	Buffer overflow en iptables-restore en la herramienta iptables de netfilter permitirá a un atacante provocar un crash en el programa o ganar el control de la ejecución a través de un fichero de iptables-save especialmente modificado.	IPtables > 1.8.2
6.2.3. Firmware de los Routers Estos son los posibles modelos de router a utilizar en el sistema: - 880G de Cisco. - RUT950 de Teltonika. - LR77v2 de Advantech. Herramientas software de seguridad – **firewalld, firewall-cmd**	CVE-2016-5410	2.1 LOW	El script firewalld.py permite a los usuarios locales saltarse la autenticación y modificar las configuraciones del firewall a través de métodos de la API de D-Bus.	Firewalld > 0.4.3.3

6.2.3. Firmware de los Routers Estos son los posibles modelos de router a utilizar en el sistema: - 880G de Cisco. - RUT950 de Teltonika. - LR77v2 de Advantech. Herramientas software de seguridad – **Firewall Builder**	CVE-2009-4664	3.3 LOW	Al correr en Linux permite a los usuarios locales escalar privilegios a través de un ataque de enlace simbólico en un fichero temporal sin especificar, que se crea a través del script de iptables.	Firewall Builder Version NOT IN (3.0.4, 3.0.5, 3.0.6)
6.2.3. Herramientas software de seguridad – Kaspersky Internet Security	CVE-2006-3074	5.0 MEDIUM	En el fichero klif.sys no se está validando algunos parámetros de llamadas al sistema, lo que permite a los atacantes realizar ataques DoS (reboot).	Kaspersky Internet Security > 7.0
	CVE-2009-2647	5.0 MEDIUM	Vulnerabilidad sin especificar permite a los atacantes deshabilitar la aplicación.	Kaspersky Internet Security Critical Fix > 9.0.0.463
	CVE-2009-2966	4.3 MEDIUM	El ejecutable avp.exe permite a los atacantes remotos provocar una denegación de servicio o DoS.	Kaspersky Internet Security > 9.0.0.459 Kaspersky Antivirus > 9.0.0.463
6.2.5. Protocolos y Servicios - SFTP	CVE-2018-16792	6.4 MEDIUM	SolarWinds SFTP Server en 2018-09-10 es vulnerable a inyección XXE (XML External Entity).	SFTP Server de distinto Vendor (vsFTPd o Mozilla SFTP Server).

	CVE-2018-16791	5.0 MEDIUM	SolarWinds SFTP Server en 2018-09-10 almacena las contraseñas de los usaurios de manera poco segura, permitiendo a un atacante determinar contraseñas para potenciales cuentas privilegiadas. Además permite al atacante la capacidad de un acceso por la puerta de atrás al servidor.	SFTP Server de distinto Vendor (vsFTPd o Mozilla SFTP Server).
6.2.5. Protocolos y Servicios – SSH v2	CVE-2002-1645	10.0 CRITICAL	Error de buffer overflow en la característica de capturador de URL permite a los atacantes remotos ejecutar código arbitrario a través de una URL larga.	SSH Client for Workstations > 3.2.0
	CVE-1999-1029	7.5 HIGH	El SSH Server no registra en los logs correctamente los intentos de login si la conexión se cierra antes del número máximo de intentos, permitiendo a un atacante remoto adivinar la contraseña sin que se recoja en los logs.	SSH Version > 2.0.12
	CVE-2002-1644	7.2 HIGH	SSH para Servidores y Workstations, al correr sin PTY, acaba provocando que los atacantes obtengan escalada de privilegios.	SSH Client for Server & Workstations > 3.2.1
	CVE-2002-1715	7.2 HIGH	SSH 1 a 3, permite a los usuarios locales saltarse shells restringidos para luego obtener acceso normal al shell.	SSH Version > 3
	CVE-2000-0217	5.1 MEDIUM	La configuración por defecto de SSH permite X forwarding, lo que permitiría a un atacante remoto obtener el control de las sesiones X de un cliente a través de	Modificar configuración por defecto de SSH con respecto a X forwarding.

			un programa malicioso xauth.	
	CVE-1999-1231	5.0 MEDIUM	SSH permite a los usuarios con nombres válidos insertar la contraseña múltiples intentos, pero sólo muestra nombre inválido para una contraseña una vez, lo que permite a los atacantes remotos determinar los nombres de cuentas válidos en el servidor.	SSH Version > 2.0.12
	CVE-2001-0364	5.0 MEDIUM	SSH Communications Security sshd para Windows permite a los atacantes remotos crear un ataque DoS.	SSH Server Version > 2.4
	CVE-1999-0398	4.6 MEDIUM	Para algunas instancias de SSH en Sistemas Linux, SSH permitirá el login a los usuarios con cuentas expiradas.	SSH Version NOT in (1.2.27, 2.0.11)
	CVE-1999-1159	4.6 MEDIUM	SSH permite a los usuarios locales solicitar el reenvío remoto de puertos desde puertos privilegiados sin ser root.	SSH Version > 2.0.11
6.2.5. Protocolos y Servicios – Telnet (NCSA)	CVE-1999-1090	7.5 HIGH	La configuración por defecto del paquete NCSA Telnet para Macintosh y PC habilita FTP, lo que permite a los atacantes leer y modificar ficheros arbitrarios.	Modificar configuración por defecto
	CVE-2005-0468	7.5 HIGH	Error de heap buffer overflow en la función env_opt_add en telnet.c permite a los atacantes remotos ejecutar código arbitrario.	N/D
	CVE-2005-0469	7.5 HIGH	Error de buffer overflow en la función slc_add_reply al manejar las subopciones LINEMODE, permite a los atacantes remotos	N/D

			ejecutar código arbitrario.	
6.2.5. Protocolos y Servicios – Telnetd (Linux)	CVE-2004-0998	7.5 HIGH	Vulnerabilidad format string en telnetd-ssl permite a los atacantes remotos ejecutar código arbitrario.	telnetd-ssl versión > 0.17
	CVE-2005-2040	5.0 MEDIUM	Múltiples errores buffer overflow en la función getterminal en telnetd para Heimdal permiten a los atacantes remotos ejecutar código arbitrario.	telnetd for Heimdal > 0.6.5
6.2.5. Protocolos y Servicios – Microsoft Telnet Client	CVE-2005-0488	5.0 MEDIUM	Ciertos clientes Telnet basados en BSD, permiten a servidores Telnet maliciosos leer variables de entorno sensibles.	N/D
6.2.5. Protocolos y Servicios – Web Services (REST Project, Restws)	CVE-2012-5556	6.8 MEDIUM	Múltiples vulnerabilidades cross-site request forgery (CSRF) en los módulos RESTful Web Service (RESTWS) permite a los atacantes el secuestro de la autenticación de usuarios arbitrarios.	RESTWS versions NOT in (7.x-1.x before 7.x-1.1 abd 7.x-2.x before 7.x-2.0-alpha3)
	CVE-2013-0205	6.8 MEDIUM	Vulnerabilidad CSRF en RESTWS permite a los atacantes remotos secuestrar la autenticación de usuarios arbitrarios.	RESTWS versions NOT in (7.x-1.x before 7.x-1.2 and 7.x-2.x before 7.x-2.0-alpha4)
	CVE-2013-1946	4.3 MEDIUM	RESTWS cuando se habilita el caché de página y a usuarios anónimos se les asignan permisos RESTWS, se les permite a los atacantes remotos provocar DoS.	RESTWS versions NOT in (7.x-1.x before 7.x-1.3 and 7.x-2.x before 7.x-2.0-alpha5)
6.2.6. Dispositivos de almacenamie	CVE-2017-5695	2.1 LOW	Vulnerabilidad de corrupción de datos permite a los usuarios locales provocar DoS.	N/D

nto – SSD (Pro 5400s 2.5" Firmware)				

Comentar que no se han hallado vulnerabilidades en los sistemas de referencia utilizados relativos a las tecnologías AFC (se han buscado los términos "Mifare", "Desfire", "SAM", "HCE", las ISOs "14443", "7816"y "QR Code", sin resultado relevante). No obstante, se conocen algunas debilidades en estas tecnologías, concretamente en el caso de Mifare Classic, que permite a los atacantes descubrir las claves de la tarjeta [62]. Esto implica que o bien las búsquedas realizadas no son las indicadas, o bien los sistemas de referencia y sus buscadores no son infalibles (se asume por tanto que las vulnerabilidades existen pero los buscadores no son capaces de mostrarlas, lo cual es muy poco probable), o bien no existen vulnerabilidades recogidas en estos sistemas de referencia relativas a las tecnologías indicadas.

6.4. Análisis del listado obtenido

A continuación, tras observar la lista de vulnerabilidades elaborada, se extraen una serie afirmaciones.

6.4.1. Consideraciones iniciales

Se especifican las vulnerabilidades encontradas para cada módulo, ordenadas por puntuación decreciente CVSS.

Para aquellos elementos del sistema con más vulnerabilidades, se muestran solamente las 10 vulnerabilidades más graves.

Para el caso del dispositivo de almacenamiento para el SIUV, se considera un SSD con firmware Intel Pro 5400s 2.5" para particularizar un caso concreto de firmware de disco SSD, recogido en CVE Details.

6.4.2. Tipología de vulnerabilidades recogidas

Los tipos de vulnerabilidades recogidas abarcan un amplio abanico (denegación de servicio, ejecución remota de código arbitrario, buffer overflow, bypass de autenticación, escalado de privilegios, toma de control del sistema, entre otros), especificando el elemento afectado, las versiones y cómo se produce la misma, y también y principalmente qué es lo que permite hacer a los atacantes. Importante también es la puntuación de la vulnerabilidad en base a las métricas de CVSS, de modo que cuanto más peligrosa o grave sea, más puntuación tendrá en una escala del 1 al 10 (0-3.9 LOW, 4.0-6.9 MEDIUM, 7-9 HIGH, 9-10 CRITICAL).

De entre las recogidas, las que más se repiten son la de denegación del servicio (DoS), y la ejecución remota de código arbitrario.

6.4.3. Elementos del sistema con más vulnerabilidades

Resulta interesante cuanto menos que un Sistema Operativo de un vendor con tanta experiencia y tan popular como Microsoft y en este caso concreto Windows 10 sea el que más vulnerabilidades presenta, que, de acuerdo a lo que se recoge en CVE Details [100] son 1111. Lo normal aunque si bien es cierto no deseable en un producto nuevo es que haya un número determinado de errores o debilidades, hasta que se realizan las correcciones y ajustes correspondientes a través de las actualizaciones; no obstante, 1111 son quizás demasiadas, y muchas de ellas muy graves. De hecho las mostradas en la lista del apartado anterior, son las 10 más graves, todas con puntuación CVSS catalogada como CRITICAL, y lo peor es que no tiene o

no se informa del fix o parche correspondiente. Es por tanto un punto importante a tener en cuenta, se han de proteger las estaciones de trabajo del Centro de Control con otras técnicas complementarias, como una topología de red segura, cortafuegos, redundancia, software de protección, firewalls, y una política eficiente de seguridad, conocida y aplicada por todos los empleados.

Por debajo de Windows 10, se observa que Windows Server 2019 presenta 405 vulnerabilidades, que aunque menos de la mitad de Windows 10, también suponen un número ciertamente elevado, y quizás éstas sean más problemáticas puesto que se corresponden con un sistema crítico como son los Servidores. En este caso, también como en el caso anterior, habría que apuntalar el sistema de los servidores con otras herramientas, como las comentadas en el párrafo anterior.

Con respecto a otros sistemas operativos del sistema IoT, como los GNU/Linux que operan en los sistemas embarcados, presentan menos vulnerabilidades, aunque también son graves, y están expuestas a las inclemencias de los atacantes en la red abierta 4G, obstaculizándolos únicamente el firewall de los routers de cada subsistema. Tanto el firewall de los routers como las propias herramientas utilizadas en los routers para tal efecto, presentan vulnerabilidades (las de las herramientas como iptables o firewalld quizá no tantas y no sean problemáticas, además de que pueden tener solución a partir de cierta versión); con respecto a las vulnerabilidades del firmware de los routers, quizá sea mejor utilizar el RUT950 de Teltonika y el LR77v2 de Advantech (para este caso del SIUP solamente se había propuesto este modelo) puesto que está mejor especificada la versión de firmware a partir de la cual se pueden considerar solucionadas las vulnerabilidades encontradas.

6.4.4. Conclusiones

El sistema presenta una serie de vulnerabilidades que afectan en mayor o menor medida a sus componentes más importantes, siendo el más evidente el sistema operativo de los puesto de operador del Centro de Control, seguido muy de cerca por el sistema operativo elegido para los Servidores, Windows 10 y Windows Server 2019 respectivamente. En estos casos, no resulta tan sencillo cambiar de sistema operativo puesto que hace falta adquirir licencias (que no son gratuitas), y por las que además de la instalación de dichos sistemas, se garantiza la actualización de los mismos a los últimos parches de seguridad.

Respecto a las demás vulnerabilidades, en su práctica totalidad, se especifica la versión o versiones que son vulnerables, de modo que es relativamente sencillo evitarlas, actualizando a una versión posterior, o utilizando otro producto que presente menos vulnerabilidades. Esta actualización puede que no sea siempre posible, por incompatibilidades con otros paquetes software del sistema, o porque se inhabiliten componentes software que son cruciales para la operativa de dicho componente en el subsistema que corresponda.

Como observación final, es evidente que el ejercicio de consulta de vulnerabilidades del sistema diseñado es tremendamente positivo para poder establecer los puntos más débiles, y fortalecerlos o bien utilizando otras versiones de dichos productos, o bien aumentando o apuntalando la seguridad de dicho punto con otras herramientas que permitan mitigar los riesgos.

7. CONCLUSIONES Y TRABAJO FUTURO

7.1. Conclusiones

Queda patente la importancia y la creciente expansión de los sistemas IoT en todos los ámbitos. Esto como se ha observado es gracias a varios factores, entre los que se destaca la evolución tecnológica actual (SoCs, dispositivos handheld y embarcados, redes inalámbricas, plataformas DIY), pudiéndose añadir la tendencia actual del mercado, o el interés del consumidor por equipar su entorno (laboral o domiciliario entre otros) de gadgets que automatizan procesos o facilitan tareas allá donde funcionen, también dependiendo del sistema o sistemas con los que interactúen. Además de la comentada evolución, cabe destacar la sofisticación y mejora en los procesos en los que intervienen estos sistemas, siendo parte esencial de sistemas mayores y más complejos, como estaciones meteorológicas, sistemas de seguridad en edificios, vehículos autónomos (perteneciente a la tipología Connected Car) y los sistemas IoT de Transporte, cuya tipología se ha estudiado más profundamente en este trabajo y del que se ha presentado un diseño IoT de Transporte grosso modo para particularizar el estudio de debilidades, riesgos, amenazas y vulnerabilidades.

En definitiva los sistemas IoT es tecnología en plena expansión y por tanto de una relevancia considerable. Tanto o más importante, es la propia seguridad de estos sistemas, que, debido a muchas de las limitaciones de estos sistemas, es un punto que se ha de mejorar y que es en parte objeto de este trabajo.

Cualquier sistema, sea IoT o no, no está exento de debilidades o agujeros de seguridad, pero ciertas prácticas o premisas, si se aplican, permitirán mejorar la seguridad del sistema, minimizando los riesgos de que estas debilidades se conviertan en amenazas reales, dificultando la tarea a los ciberdelincuentes. De entre estas buenas prácticas, una de las más importantes sería la seguridad desde el diseño, es decir, considerar parámetros de seguridad, tanto integrales de todo el sistema, como de cada componente, en todas las fases SLDC del proyecto.

No se debe olvidar, que gran parte de los sistemas IoT se implementan en base a un contrato que establece entre el contratista y proveedor el cumplimiento de unos requisitos de usuario que pueden derivar inevitablemente en vulnerabilidades cuya solución es más compleja (como la mencionada en relación a la red móvil 4G con IPs públicas) ya que no se deben ignorar los requisitos de usuario, si no se estaría incumpliendo con el contrato. En este punto, sería deseable negociar el cambio de ciertas condiciones técnicas con el contratista, para evitar o dejar cubiertas estas debilidades del sistema, alcanzando una solución de compromiso (ciertas condiciones del contrato pueden esconder un requisito de usuario que puede cumplirse de otro modo y sin dejar comprometida la seguridad del sistema).

En estos sistemas además, una vez que los componentes del mismo han tomado forma y están diseñados o con un diseño previo (es decir, antes de la implementación), resulta tremendamente positivo realizar un estudio de vulnerabilidades para identificar el elemento más débil y tomar las precauciones debidas a tiempo, pudiendo contemplar también un cambio en el diseño. A tal efecto, son muy útiles los sitios web que detallan y permiten ordenar y explotar la información de las vulnerabilidades, como CVE Details [96].

7.2. Trabajo futuro

En futuras iteraciones o lo que podría suponer la ampliación del presente trabajo, podría considerarse el análisis de amenazas, riesgos y vulnerabilidades en otras tipologías de IoT también muy en auge actualmente, como las Smart Homes (son los propios usuarios los que adquieren, instalan y ponen en marcha el sistema, con los consiguientes problemas de

seguridad, muchos de ellos debidos a una configuración incorrecta o insuficiente realizada por usuarios con conocimientos informáticos pero no expertos en la materia), Connected Car (sistema que se está implantando en el vehículo autónomo de Tesla Inc.), sistemas IoT de Logistics & Asset Tracking (analizando por ejemplo el efectivo sistema utilizado por Amazon para la logística y el seguimiento de envíos), o incluso por su interés y por permitir mejorar la eficiencia de la producción agrícola, los sistemas Smart Agriculture. Todos estos sistemas presentan vulnerabilidades que bien hubiesen podido aportar aspectos importantes y de interés al trabajo, pudiendo observar y estudiar diferentes vectores de amenazas de seguridad de acuerdo con la naturaleza de la tipología del IoT correspondiente.

Por otro lado, habría sido interesante el desarrollo o configuración (en herramientas como Metasploit, para generar el exploit basta concatenar una serie de comandos parametrizando los valores correspondientes, como la IP, el puerto de conexión, etcétera) de exploits o aplicaciones de pentesting que permitieran explotar las vulnerabilidades más peligrosas del sistema. No solo documentar el proceso de creación del mismo, si no el análisis de los resultados y las consecuencias de infectar con dicho exploit los elementos vulnerables del sistema. Tras lo cual, se podría determinar cuál es la mejor forma de paliar o solucionar esa debilidad, independientemente de las buenas prácticas aplicadas.

Por último, y tal vez con más recursos y tiempo, sería muy positivo e interesante poder implementar un sistema IoT propio (una Smart Home por ejemplo) y realizar la búsqueda empírica de vulnerabilidades, amenazas y riesgos del sistema, pudiendo crear o encontrar los exploits correspondientes a cada vulnerabilidad, observando los efectos del exploit en el sistema, y estableciendo los cambios pertinentes acordes a la guía de buenas prácticas, comprobando si efectivamente dichas prácticas permiten mejorar la seguridad del sistema y ayudan a mitigar el riesgo de cada agujero de seguridad o si por el contrario se demuestra que son ineficaces y que el diseño del sistema se ha de revisar. Sería deseable conseguir un sistema IoT' (prima) que solucione las vulnerabilidades del sistema IoT original; en este punto se podría iterar, y comprobar las vulnerabilidades, riesgos y amenazas de dicho sistema IoT', y establecer las medidas para conseguir un sistema IoT'' más inmune aún a ciberataques. Se entraría no obstante en una dinámica con tendencia asintótica (donde el número de incidentes o agujeros de seguridad tiende a 0) al implementar mejoras infinitesimales cuyo esfuerzo o coste computacional para conseguirlas podrían cumplir con el principio de Pareto (regla del 80-20, donde se establece que el 80% de los resultados se obtienen con el 20% del esfuerzo, y el 20% restante, con el 80% del esfuerzo, siendo estos porcentajes no estrictos [101]) y por tanto una vez superado ese umbral del 80% de mejoras (muy probablemente tras la aplicación de las primeras medidas obteniendo el sistema IoT'), no serían ya modificaciones eficientes y no sería realista aplicarlas, no obstante podrían resultar de interés académico.

Referencias

[1] Web de Deloitte sobre IoT. <https://www2.deloitte.com/es/es/pages/technology/articles/IoT-internet-of-things.html>. Última visita 26/02/2020.

[2] Web de Domo Desk sobre IoT. <https://www.domodesk.com/221-a-fondo-que-es-iot-el-internet-de-las-cosas.html>. Última visita 26/02/2020.

[3] Web de Sigfox, solución de comunicación entre dispositivos IoT. <https://www.sigfox.com/en/sigfox-story>. Última visita 27/02/2020.

[4] Wikipedia sobre IoT. <https://en.wikipedia.org/wiki/Internet_of_things>. Última visita 01/03/2020.

[5] Web de Oracle sobre IoT. <https://www.oracle.com/es/internet-of-things/what-is-iot.html>. Última visita 29/02/2020.

[6] OWASP IoT. <https://owasp.org/www-project-internet-of-things/>. Última visita 28/02/2020.

[7] Web de GanttProject. <https://www.ganttproject.biz/>. Última visita 02/03/2020.

[8] Web de INCIBE sobre la importancia de la seguridad en IoT. <https://www.incibe-cert.es/blog/importancia-seguridad-iot-principales-amenazas>. Última visita 02/03/2020.

[9] Blog de computing.es sobre las amenazas de IoT. <https://www.computing.es/movilidad/noticias/1116891046501/amenazas-iot-seguridad-de-red.1.html>. Última visita 02/03/2020.

[10] Blog de noticias de redeszone.net sobre la amenaza de seguridad relativa a IoT. <https://www.redeszone.net/noticias/seguridad/dispositivos-iot-amenaza-seguridad/ >. Última visita 02/03/2020.

[11] Blog de noticias de ituser.es sobre los ataques sufridos en organizaciones a través de IoT. < https://www.ituser.es/seguridad/2020/02/siete-de-cada-diez-organizaciones-sufren-ciberataques-a-traves-de-dispositivos-iot>. Última visita 02/03/2020.

[12] Blog de noticias de dealerworld.es sobre la solución de Cisco para fortalecer seguridad en IoT. <https://www.dealerworld.es/seguridad/cisco-fortalece-la-seguridad-iot>. Última visita 03/03/2020.

[13] Blog de noticias de redeszone.net sobre la solución *Honware*. <https://www.redeszone.net/noticias/seguridad/honware-detectar-vulnerabilidades-dispositivos-iot/>. Última visita 03/03/2020.

[14] Blog de noticias de computerworld.es sobre la importancia económica de la seguridad en IoT. <https://cso.computerworld.es/tendencias/la-industria-apuesta-por-comprar-proveedores-de-seguridad-en-la-nube>. Última visita 03/03/2020.

[15] Web de Netatmo, termostatos inteligentes. <https://www.netatmo.com/es-es/energy/thermostat>. Última visita 05/04/2020.

[16] Web de la bombilla inteligente de Xiaomi. <https://www.mi.com/es/mi-led-smart-bulb/>. Última visita 05/04/2020.

[17] Youtube, explicación SoC. < https://www.youtube.com/watch?v=L4XemL7t6hg>. Última visita 05/04/2020.

[18] Ejemplo de IoT con Smartphone y smartwatch. < https://uidesign.gbtcdn.com/gb_blog/1501-1800/1591/connect-NO.1-D5-smart-watch-to-iPhone-K5.jpg>. Última visita 05/04/2020.

[19] Web Qualcomm, < https://www.qualcomm.com/products>. Última visita 05/04/2020.

[20] Web Samsung, familia de procesadores de Exynos. < https://www.samsung.com/semiconductor/minisite/exynos/products/all-processors/>. Última visita 05/04/2020.

[21] Blog 4G mobile data usage. < https://tefficient.com/category/analysis/4g-lte/page/2/>. Última visita 05/04/2020.

[22] ADSLZone sobre oferta comercial 5G de Vodafone. < https://www.adslzone.net/noticias/operadores/vodafone-yu-nuevas-tarifas-fibra-5g/>. Última visita 05/04/2020.

[23] Youtube, construcción de robot rastreador usando Raspberry Pi y Arduino. < https://www.youtube.com/watch?v=BrMRJuwiP8o>. Última visita 05/04/2020.

[24] Raspberry pi org, proyecto de estación meteorológica. < https://projects.raspberrypi.org/en/projects/build-your-own-weather-station>. Última visita 05/04/2020.

[25] Microchip Technology, Datasheet SAM D21 Family. Low-Power, 32-bit Cortex-M0+ MCU with Advanced Analog and PWM. DS40001882D.

[26] U-blox, Datasheet NINA-W10 series. Stand-alone multiradio modules. UBX-17065507-R05.

[27] U-blox, Web sobre productos de posicionamiento (GPSs). <https://www.u-blox.com/en/positioning-chips-and-modules>. Última visita 05/04/2020.

[28] Arduino, web sobre el producto Nano 33 IOT. <https://store.arduino.cc/arduino-nano-33-iot>. Última visita 05/04/2020.

[29] Arduino, proyectos de la comunidad de usuarios. <https://create.arduino.cc/projecthub/projects/tags/iot?page=1>. Última visita 05/04/2020.

[30] Arduino, proyecto con display que muestra información del avance del COVID-19. <https://create.arduino.cc/projecthub/hwhardsoft/covid19-realtime-monitor-5f6920?ref=tag&ref_id=iot&offset=0>. Última visita 23/03/2020.

[31] Forbes, influencia de las IoT en la huella de carbono. <https://www.forbes.com/sites/simonchandler/2019/11/05/how-the-internet-of-things-will-help-fight-climate-change/>. Última visita 05/04/2020.

[32] IoT Analytics, actualización del estado de las IoT en 2018, <https://iot-analytics.com/state-of-the-iot-update-q1-q2-2018-number-of-iot-devices-now-7b/>. Última visita 05/04/2020.

[33] Security Today, predicciones sobre implantación, riesgos, estadísticas y soluciones. <https://securitytoday.com/Articles/2020/01/13/The-IoT-Rundown-for-2020.aspx?Page=2>. Última visita 05/04/2020.

[34] 14Core, "The IOT Protocols The base of Internet of Things Ecosystem". <https://www.14core.com/the-iot-protocols-the-base-of-internet-of-things-ecosystem/>. Última visita 05/04/2020.

[35] Bluetooth.com, procedimiento para el descubrimiento de servicios. <https://www.bluetooth.com/specifications/assigned-numbers/service-discovery/>. Última visita 05/04/2020.

[36] Wikipedia, definición de Cloud Computing. <https://en.wikipedia.org/wiki/Cloud_computing>. Última visita 05/04/2020.

[37] Wikipedia, Redes neuronales Artificiales. <https://en.wikipedia.org/wiki/Artificial_neural_network>. Última visita 05/04/2020.

[38] Web de INCIBE sobre los protocolos de comunicación de redes IoT, ataques y recomendaciones. <https://www.incibe-cert.es/blog/iot-protocolos-comunicacion-ataques-y-recomendaciones>. Última visita 30/04/2020.

[39] Behr Tech Blog, "6 Leading Types of IoT Wireless Tech and Their Best Use Cases". <https://behrtech.com/blog/6-leading-types-of-iot-wireless-tech-and-their-best-use-cases/>. Última visita 05/04/2020.

[40] M. Aabid A Majeed, Thashika D Rupasinghe, "Internet of Things (IoT) Embedded Future Supply Chains for Industry 4.0: An Assessment from an ERP-based Fashion Apparel and Footwear Industry". Sheffield Hallam University, United Kingdom, Vol. 6, No. 1, March 2017.

[41] FIB, "Historia de Internet". <https://www.fib.upc.edu/retro-informatica/historia/internet.html>. Última visita 05/04/2020.

[42] iotfactory.eu. "Smart metering". <https://iotfactory.eu/products/smart-metering/>. Última visita 05/04/2020.

[43] red.es, Ciudades e Islas Inteligentes. Agente Digital para España: Valladolid. "S2CITY-SISTEMA INTELIGENTE DE SERVICIOS AL CIUDADANO Y AL TURISTA", Dosier informativo. Ministerio de Energía, Turismo y Agenda Digital (Gobierno de España).

[44] Zona movilidad, noticia "Valladolid, más cerca de ser una Smart city". <https://www.zonamovilidad.es/valladolid-mas-cerca-de-ser-una-smart-city>. Última visita 05/04/2020.

[45] Comfy, "Top 8 Smart Buildings from around the world". <https://www.comfyapp.com/blog/top-8-smart-buildings-from-around-the-world/>. Última visita 05/04/2020.

[46] Wikipedia sobre Inmótica. <https://es.wikipedia.org/wiki/Inm%C3%B3tica>. Última visita 05/04/2020.

[47] Sitio web para Smart Homes en Amazon. <https://www.amazon.com/smart-home-devices/>. Última visita 05/04/2020.

[48] Epson, especificaciones gafas de realidad aumentada. <https://www.epson.es/products/see-through-mobile-viewer/gafas-moverio-bt-300/Datos-tecnicos>. Última visita 05/04/2020.

[49] Wikipedia, E-Textiles. <https://en.wikipedia.org/wiki/E-textiles>. Última visita 05/04/2020.

[50] BMW, "Connected Car". <https://www.bmw.com/es/innovation/connected-car.html>. Última visita 05/04/2020.

[51] Revista viajeros. "GMV Moderniza el transporte urbano de Rabat". <https://www.revistaviajeros.com/noticia/12216/gmv-moderniza-el-transporte-urbano-de-rabat->. Última visita 05/04/2020.

[52] Farahani, Bahar, et al. "Towards fog-driven IoT eHealth: Promises and challenges of IoT in medicine and healthcare." Future Generation Computer Systems 78 (2018): 659-676.

[53] Ubuntupit, ejemplos de elementos IoT healthcare. <https://www.ubuntupit.com/iot-in-healthcare-20-examples-thatll-make-you-feel-better/>. Última visita 05/04/2020.

[54] onTheSpot, Blog. "¿Qué es Smart retail?". <https://empresas.blogthinkbig.com/que-es-smart-retail/>. Última visita 05/04/2020.

[55] AWS Amazon, "Asset Tracking That Works Out of the Box from AWS IoT, Verizon, and Domo". <https://aws.amazon.com/es/blogs/apn/asset-tracking-that-works-out-of-the-box-from-aws-iot-verizon-and-domo/>. Última visita 05/04/2020.

[56] Youtube, Explicación Smart Agriculture. <https://www.youtube.com/watch?v=j4HBlOf5ZDA>. Última visita 05/04/2020.

[57] IOT Solutions World Congress, "IOT TRANSFORMING THE FUTURE OF AGRICULTURE". <https://www.iotsworldcongress.com/iot-transforming-the-future-of-agriculture/>. Última visita 05/04/2020.

[58] El Horizonte noticias (México), "Europa se quedará sin autos de combustión". <https://d.elhorizonte.mx/estilo/europa-se-quedara-sin-autos-de-combustion/2815192>. Última visita 05/04/2020.

[59] Wikipedia, SAE. <https://es.wikipedia.org/wiki/Sistema_de_ayuda_a_la_explotaci%C3%B3n>. Última visita 05/04/2020.

[60] Wikipedia, TDMA. <https://es.wikipedia.org/wiki/Acceso_m%C3%BAltiple_por_divisi%C3%B3n_de_tiempo>. Última visita 05/04/2020.

[61] NXP Semiconductors. MF1S70YYX_V1 MIFARE Classic EV1 4K - Mainstream contactless smart card IC for fast and easy solution development, Rev. 3.2 — 23 November 2017 Product data sheet 279332 COMPANY PUBLIC.

[62] Web sobre pirateo de tarjetas Mifare Classic. < https://firefart.at/post/how-to-crackmifare-classic-cards/>. Última visita 05/04/2020.

[63] IDESCO, "Mifare DESFire is the most secure RFID technology". <https://idesco.fi/desfire/mifare-desfire-technology/>. Última visita 05/04/2020.

[64] Calypso Net, "Ticketing Transaction Layers". <https://www.calypsonet-asso.org/content/ticketing-transaction-layers>. Última visita 05/04/2020.

[65] Redsys.es, Movilidad y pagos. <http://www.redsys.es/movilidad.html>. Última visita 05/04/2020.

[66] Web EMVCo, pago contactless. <https://www.emvco.com/emv-technologies/contactless/>. Última visita 05/04/2020.

[67] Flixbus, ejemplos de servicios adicionales en autobuses. <https://www.flixbus.es/servicio/en-autobuses>. Última visita 05/04/2020.

[68] NetSpot, "Protocolos de seguridad inalámbrica: WEP, WPA, WPA2 y WPA3". <https://www.netspotapp.com/es/wifi-encryption-and-security.html>. Última visita 05/04/2020.

[69] Transmodel-CEN. Modelos de datos y protocolos abiertos. <http://www.transmodel-cen.eu/>. Última visita 05/04/2020.

[70] The Kernel.org archive. <https://www.kernel.org/>. Última visita 05/04/2020.

[71] Routers Cisco, series 880G y 890G. <https://www.cisco.com/c/en/us/products/collateral/routers/800-series-routers/datasheet_c78-732744.html>. Última visita 05/04/2020.

[72] Router Teltonika RUT950. <https://teltonika-networks.com/es/product/rut950/>. Última visita 05/04/2020.

[73] Kaspersky, Total Security. <https://www.kaspersky.es/total-security>. Última visita 05/04/2020.

[74] Wikipedia, Internet of Military Things. <https://en.wikipedia.org/wiki/Internet_of_Military_Things>. Última visita 05/04/2020.

[75] Blog INCIBE, Amenaza vs Vulnerabilidad, ¿sabes en qué se diferencian?. < https://www.incibe.es/protege-tu-empresa/blog/amenaza-vs-vulnerabilidad-sabes-se-diferencian>. Última visita 20/04/2020.

[76] ENISA "Good practices for security of IoT. Secure Software Development Lifecycle, NOVEMBER 2019.

[77] OWASP, Code Injection. <https://owasp.org/www-community/attacks/Code_Injection>. Última visita 26/04/2020.

[78] Advantech B+B Smartworx, LR77 v2. < http://advantech-bb.com/product/4g-lte-router-lr77-v2/>. Última visita 26/04/2020.

[79] Guru99, How to Hack a Web Server. <https://www.guru99.com/how-to-hack-web-server.html>. Última visita 29/04/2020.

[80] Wikipedia, HTTPS. <https://en.wikipedia.org/wiki/HTTPS>. Última visita 29/04/2020.

[81] Firewall Builder site, Sourceforge. < http://fwbuilder.sourceforge.net/>. Última visita 01/05/2020.

[82] Wikipedia, TDES. < https://en.wikipedia.org/wiki/Triple_DES>. Última visita 01/05/2020.

[83] Wikipedia, Blowfish. < https://en.wikipedia.org/wiki/Blowfish_(cipher)>. Última visita 01/05/2020.

[84] Wikipedia, Secure Hash Algorithms. < https://en.wikipedia.org/wiki/Secure_Hash_Algorithms>. Última visita 01/05/2020.

[85] EMV Terminal certification. < https://blog.payjunction.com/emv-terminal-certification-explained>. Última visita 01/05/2020.

[86] Heraldo.es, "Detectados 300 posibles casos de estafa en las recargas de bus y tranvía de Zaragoza". < https://www.heraldo.es/noticias/aragon/zaragoza/2019/05/17/estafa-recargas-tarjetas-bus-tranvia-zaragoza-1315274.html>. Última visita 01/05/2020.

[87] Web sobre CVE de Mitre. <https://cve.mitre.org/>. Última visita 19/05/2020.

[88] Web de Security Focus sobre Bugtraq. <https://www.securityfocus.com/>. Última visita 19/05/2020.

[89] Web sobre CWE de Mitre. <https://cwe.mitre.org/>. Última visita 20/05/2020.

[90] Ejemplo de vulnerabilidad en la base de datos de BugTraq / SecurityFocus. <https://www.securityfocus.com/bid/109320>. Última visita 20/05/2020.

[91] Ejemplo de vulnerabilidad en la base de datos de NVD. <https://nvd.nist.gov/vuln/detail/CVE-2019-1010241>. Última visita 20/05/2020.

[92] Wikipedia sobre CVSS. < https://en.wikipedia.org/wiki/Common_Vulnerability_Scoring_System>. Última visita 20/05/2020.

[93] NVD Calculator V2. <https://nvd.nist.gov/vuln-metrics/cvss/v2-calculator>. Última visita 21/05/2020.

[94] Métricas obtenidas para ejemplo de vector CVSS. <https://nvd.nist.gov/vuln-metrics/cvss/v2-calculator?vector=(AV:L/AC:H/Au:S/C:P/I:P/A:P/E:POC/RL:TF/RC:UC/CDP:L/TD:L/CR:M/IR:M/AR:M)>. Última visita 21/05/2020.

[95] Buscador de vulnerabilidades NVD. <https://nvd.nist.gov/vuln/search>. Última visita 24/05/2020.

[96] Buscador de vulnerabilidades CVE Details. <https://www.cvedetails.com/>. Última visita 25/05/2020.

[97] CVE Details, Linux Kernel 2.6.35 RC2 Security Vulnerabilities. <https://www.cvedetails.com/vulnerability-list/vendor_id-33/product_id-47/version_id-123860/Linux-Linux-Kernel-2.6.35.html>. Última visita 25/05/2020.

[98] CVE Details, Linux Kernel 4.9.190 Security Vulnerabilities. < https://www.cvedetails.com/vulnerability-list/vendor_id-33/product_id-47/version_id-334730/Linux-Linux-Kernel-4.9.190.html>. Última visita 25/05/2020.

[99] CVE Details, Windows Server 2019 Security Vulnerabilities. < https://www.cvedetails.com/vulnerability-list/vendor_id-26/product_id-50662/Microsoft-Windows-Server-2019.html>. Última visita 25/05/2020.

[100] CVE Details, Windows 10 Security Vulnerabilities. <https://www.cvedetails.com/vulnerability-list/vendor_id-26/product_id-32238/Microsoft-Windows-10.html>. Última visita 25/05/2020.

[101] Wikipedia, Principio de Pareto. <https://es.wikipedia.org/wiki/Principio_de_Pareto>. Última visita 29/05/2020.