Análisis de funcionamiento y de vulnerabilidades de Alexa

Sergio Bachiller Rubia

Junio de 2019

Resumen– Tras el crecimiento de los dispositivos que conforman el *Internet of Things*, apareció la necesidad de un método común de acceso y control de éstos. Así es como en 2014 nació, entre otros, Amazon Alexa, un asistente virtual capaz de interactuar con dispositivos del hogar, efectuar compras o consultar las noticias entre otras muchas más posibilidades. Este método centralizado de acceso y control a diferentes dispositivos y a un gran volumen de datos personales, pone en riesgo información como pueden ser el domicilio o el número de tarjeta de crédito, además de dar acceso a dispositivos del hogar, desde lámparas hasta cerraduras. Estos riesgos y la falta de conocimiento sobre cuál es el funcionamiento del sistema motivan la realización de un análisis que aclaren estas posibles inquietudes de los consumidores.

Palabras clave– Amazon, Alexa, Asistente virtual, Domótica, Ciberseguridad, Vulnerabilidad, Internet de las Cosas.

Abstract– After the growth of the devices that make up the *Internet of things*, the need for a common method of access and control of these appeared. This is how in 2014 was born, among others, Amazon Alexa, a virtual assistant capable of interacting with home automation devices, buy online or consult the news among many more possibilities. This centralized method of access and control to different devices and to a large volume of personal data puts at risk information such as domicile or credit card number, in addition to providing access to home automation devices, from lamps to locks. These risks and lack of knowledge about how the whole system works, motivates the realization of an analysis that clarifies these possible concerns of the consumers.

Keywords– Amazon, Alexa, Virtual assistant, Home automation, Cibersecurity, Vulnerability, Internet of Things.

1 Introducción

Internet ha crecido muy rápidamente en los últimos años, y con él todas las tecnologías que son relativas e inherentes a él. Estos avances han permitido hacer la vida cotidiana mucho más fácil, gracias a la digitalización y al desarrollo de dispositivos actuadores, que realizan tareas simples, tediosas y repetitivas para los usuarios. Tras la aparición de estos dispositivos actuadores nació la necesidad de encontrar un método único y compartido que permitiese su control, que hasta ahora se realizaba de forma separada y exclusiva para cada producto y/o fabricante.

Así pues, las grandes empresas tecnológicas vieron una necesidad que saciar en la sociedad, a lo que respondieron con productos como Amazon Alexa, el asistente virtual de la multinacional. Se presentó conjuntamente con los dispositivos Amazon Echo, los altavoces inteligentes del mismo fabricante, que ya llevarían incorporado el software de Alexa. Actualmente, estos altavoces están disponibles hasta en 5 modelos diferentes, incluyendo características como pantalla, conexión Bluetooth y videocámara, lo que permite al usuario realizar video llamadas o vigilar el hogar desde el teléfono móvil. Concretamente, para este estudio, se utiliza un Amazon Echo de 2.ª generación.

Las *skills* o habilidades funcionan de modo similar al que lo hacen las aplicaciones para teléfonos móviles, simplemente es necesario activarlas desde la tienda de habilidades. Fue así como otras compañías como Spotify o Netflix desarrollar *skills* que permitían el uso de sus servicios a través de los dispositivos que utilizaran el

• E-mail de contacto: Sergio.Bachiller@e-campus.uab.cat
• Mención realizada: Tecnologías de la Información
• Trabajo tutorizado por: Juan Carlos Sebastián Pérez (dEIC)
• Curso 2018/19

asistente, como pueden ser los propios Amazon Echo o un teléfono móvil. De igual, la *Application Programming Interface (API)* de Alexa es abierta, lo que significa que cualquier usuario que disponga de una cuenta de desarrollador puede crear sus propias *skills* y compartirla con el resto de usuarios, si procede.

El hecho de que cualquier usuario pueda desarrollar habilidades para el asistente pone en peligro la confidencialidad de los datos del usuario que está utilizando el sistema. Por ejemplo, Alexa conoce el domicilio de los usuarios, puesto que lo necesita con el fin de poder facilitar la predicción meteorológica. Teniendo en cuenta que las *skills* del asistente tienen acceso a este tipo de datos, al programador le resulta relativamente fácil capturar los datos y almacenarlos en una base de datos y utilizarlos con finalidades no autorizadas.

Así mismo, al usuario le pueden surgir otras dudas, como qué podría hacer alguien que consiguiese acceso a la red donde está conectada Alexa o si alguien podría usar el Amazon Echo a través de una ventana. Para dar respuesta a estas preguntas es necesario estudiar cual es el funcionamiento de todo el sistema de Amazon Alexa y contemplar todas las situaciones en las que un usuario pueda llegar a pensar.

2 Objetivos

Conociendo estas inquietudes de los consumidores, hay que tener en cuenta que pueden afectar negativamente a las ventas de Amazon o incluso el rechazo al producto. Así pues, los siguientes objetivos se plantean con la intención de resolver estas dudas.

En una primera instancia, es necesario estudiar cual es la arquitectura del sistema que hay detrás de Alexa; una vez se conoce cual es la estructura resulta más fácil el analizar y entender su funcionamiento.

En este punto, se diseñan dos diagramas de red local, uno en el que se analice el tráfico sin conectar el Amazon Echo y otro habiéndolo conectado. El análisis del tráfico se realizará mediante un proxy instalado entre la red local y el asistente, de tal modo que se podrá identificar el tráfico que genera Alexa y conocer su contenido.

A sabiendas de cuál es el funcionamiento concreto del sistema, es posible realizar un análisis de vulnerabilidades. Se tendrán en cuenta las vulnerabilidades a nivel de aplicación, de *API* y de red.

No hay que olvidar la estrecha relación que guarda Alexa con los diferentes dispositivos del hogar, de modo que también resulta interesante saber de su funcionamiento y analizar si existe una brecha de seguridad en relación con Alexa.

Finalmente, se estudia la documentación de los desarrolladores y la política de publicación de *skills*, con tal de verificar la política de privacidad y seguridad.

Tras conseguir estos objetivos, los beneficios esperados son los siguientes:

- Conocer el funcionamiento del sistema inherente a Amazon Alexa.
- Ganar conocimiento y experiencia en penetración de vulnerabilidades.
- Verificar o refutar la política de privacidad y seguridad de Amazon relativa a su asistente virtual.
- Proporcionar documentación técnica sobre el funcionamiento y las vulnerabilidades del ecosistema Amazon Alexa, con tal de que se puedan corregir el mal funcionamiento y las vulnerabilidades.

De este modo, se busca resolver las inquietudes que puedan surgir entre los potenciales consumidores y los consumidores que actualmente utilizan este servicio pero no depositan la totalidad de su confianza en él, lo que propicia que no hagan uso del todo el abanico de posibilidades que se ofrece. Este último caso resulta importante porque, al fin y al cabo, son clientes que pueden perecer al no fomentar el uso, y por ende, la creación de nuevas posibilidades debido a esta falta de confianza en el producto.

3 Metodología

En un estudio como éste, en el que los resultados que encontremos puede diferir totalmente en la hipótesis inicial, es necesario utilizar una metodología que permita reajustar las tareas a realizar. Kanban es una metodología ágil que justamente permite esto, la revisión continua de las tareas, además también se adapta a los posibles cambios que pueda aparecer durante el estudio, pues en cualquier momento puede aparecer una nueva vulnerabilidad o se puede producir un cambio en el funcionamiento del sistema.

Esta metodología se basa en definir las tareas, para posteriormente escribirlas en tarjetas. Estas tarjetas se dividirán posteriormente en cuatro columnas diferentes: por hacer, en proceso, a revisar y finalizadas. Además, se fijan dos puntos temporales de re planificación, justo antes de los hitos, de esta forma se pueden modificar las tareas de cara a la siguiente etapa del estudio.

4 Estado del arte

Al tratarse de un proyecto tan joven, es difícil encontrar información relativa al análisis que se realiza. Aun así, se dispone de información de calidad, como el artículo *Security Analysis of the Amazon Echo* [1], dónde en 2017 ya se realizó un estudio muy similar al que se pretende hacer dos años después. De todos modos, en el susodicho artículo no se hace referencia a los dispositivos de hogar digital, por lo que sigue resultando interesante considerarlos objeto de estudio.

En este documento se desarrolla una política de seguridad ideal para el Amazon Echo. Esta política ideal define cuatro tipos de usuarios, primario (dueño del dispositivo), secundario (consumidores del servicio), otros (visitantes) y afiliados de Amazon (con acceso a información de la que Amazon es propietaria). De este modo, se garantiza la confidencialidad, la integridad y la disponibilidad de los datos, considerando vulnerabilidad cualquier comportamiento que se escape de estas directrices. Amazon actualmente aplica una política de seguridad muy similar, aunque no tan concreta. Se utiliza un sistema de perfiles de voz, con tal de mejorar la experiencia individual de cada usuario, pero no permite cambiar los permisos a los que cada usuario tiene acceso.

En el mismo artículo se realizan diferentes pruebas con Alexa como capturar el tráfico enviado por el Amazon Echo gracias a un proxy que lo separa del router de la red local. Este tráfico es, principalmente, peticiones HTTPS las cuales están encriptadas, aunque se puede encontrar alguna que otra petición HTTP que no, como los binarios de actualizaciones. También se conoce información como la descripción de los componentes hardware del dispositivo o que el sistema operativo está basado en Android (lo que implica que también podría estar sujetos a sus vulnerabilidades).

Siguiendo en la línea de la conexión a la red, hay que destacar que se puede observar que se envía tráfico desde el Amazon Echo aunque no se esté utilizando. Comentan que pueden ser actualizaciones de estado o actualizaciones del sistema. También se intenta realizar un ataque de reproducción de paquetes sin éxito, lo que indica que es una señal de que Amazon se ha preocupado por este aspecto.

Teniendo en cuenta que el principal modo de uso de Alexa es mediante la voz, los autores realizaron ensayos con diferentes acentos anglosajones sin éxito, lo que implica una pérdida de clientes en ciertos países. También se intenta dar instrucciones mediante voz generada por ordenador y grabaciones con éxito, lo que podría implicar una posible vulnerabilidad. Es posible restringir acciones de Alexa con un PIN de 4 dígitos, en este mismo estudio se detalla como desarrollar un programa que realice un ataque de fuerza bruta mediante generación de voz; se calcula que tardaría unas 41 horas y 40 minutos.

Otras pruebas relativas a ataques de sonido, como *Skill Squatting* o *Voice Masquerading*, se han realizado en estudios como [2] con una tasa de éxito superior al 50 %. En el mismo artículo se desarrollan diferentes métodos de protección ante estas vulnerabilidades que presentan los VPA (*Virtual Personal Assistant)*.

5 ARQUITECTURA Y FUNCIONAMIENTO

En primera instancia, antes de realizar un análisis sobre las brechas de seguridad que presenta el sistema, es necesario saber cual es su arquitectura y su *modus operandi*.

En la documentación de Amazon[3] se dice que cuando el usuario interactúa con Alexa se envía la grabación a la nube de Amazon, donde el audio es transcrito y procesado. Esto es algo que tiene muchas ventajas, como por ejemplo, que el modelo de transcripción resulta fácil de mejorar y actualizar al estar en los propios servidores de Amazon, sin hacer que esa tarea recaiga sobre los mismos dispositivos Amazon Echo.

A sabiendas de que la transcripción se realiza en los servidores de Amazon, es posible hacernos una idea de cual es la arquitectura y funcionamiento del sistema al completo.

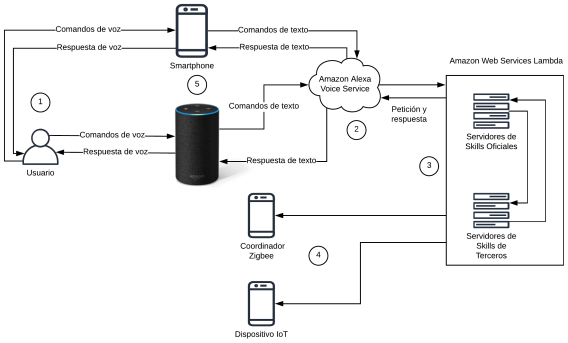

Fig. 1: Arquitectura del sistema Amazon Alexa

Siguiendo la Figura 1, podemos ver el proceso que se sigue cuando un usuario utiliza Alexa, tanto desde el smartphone como desde un dispositivo Amazon Echo. El siguiente paso que se realiza es enviar la locución a *Amazon Voice Service*, dónde se transcribe con el objetivo de analizar la petición utilizando aprendizaje computacional. Durante este análisis, se identifica el nombre de la *skill* que se quiere ejecutar y los parámetros a enviar al servicio, que se envían de forma estructurada dependiendo del servicio. De hecho, existen países como Rumanía o Costa Rica, donde hay empleados que transcriben audios que no han podido ser transcritos por la inteligencia artificial con tal de mejorar el reconocimiento[4].

A continuación, la *skill* recibe la petición y la procesa. En caso de que se trate de una habilidad de terceros y sea necesario la conexión a un servidor del desarrollador se realiza la consulta mediante un servicio REST.

Finalmente, en caso de que sea necesario, desde *Amazon Web Services Lambda* se envía la orden de actuar a los dispositivos del hogar y se envía un audio al dispositivo que ejecuta Alexa para informar al usuario que la tarea ha sido completada. Los detalles de las diferencias entre dispositivos Zigbee y dispositivos IoT comunes se explica más adelante. Se puede observar un diagrama de flujo de lo arriba explicado en la Figura 2. En la mitad superior del diagrama podemos observar un ejemplo de dar una orden de encendido o apagado a un dispositivo de hogar digital, por otro lado, en la mitad inferior, se puede ver una petición de información como la predicción meteorológica o de reproducción de multimedia bajo demanda.

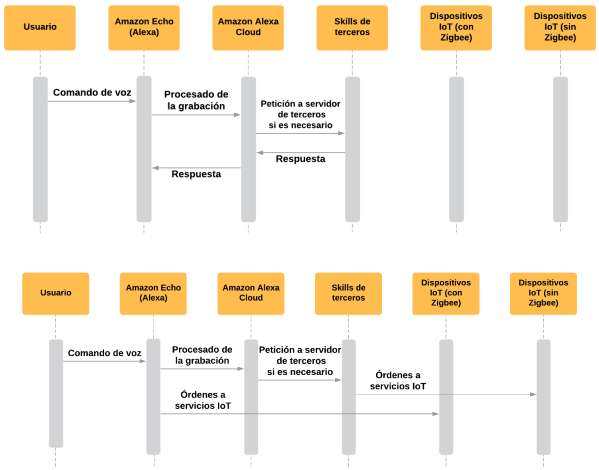

Fig. 2: Diagrama de flujo de uso de Alexa

6 Análisis de vulnerabilidades

Aún conociendo el resultado de algunas de las pruebas gracias a estudios anteriores, hay que tener en cuenta que están algo desactualizados, y a sabiendas que el desarrollo es muy rápido, es posible que un cambio producido por una actualización haya abierto una nueva brecha de seguridad.

6.1. Vulnerabilidades registradas en CVE

CVE (*Common Vulnerabilities and Exposures)* nace en 1999 con el objetivo de generar una lista con identificadores comunes y únicos para las diferentes vulnerabilidades de ciberseguridad. Cada entrada en CVE está asignada a una vulnerabilidad con su correspondiente descripción, estableciendo así un estándar público, gratuito y común que facilita el intercambio de datos relacionados con estas brechas de seguridad[5].

Las entradas CVE incluyen una métrica de impacto y explotabilidad llamada CVSS (*Common Vulnerability Scoring System)*. CVSS proporciona una forma de determinar las principales características de una vulnerabilidad y calcular un valor numérico que refleja su severidad. Se puede encontrar una lista con estas características en el Apéndice X.

Gracias a la estructura de repositorio de CVE es posible realizar una búsqueda del objeto de estudio; en este caso se ha utilizado *Alexa* y *Echo* como palabras clave.

6.1.1. Vulnerabilidad CVE-2018-11567

Esta vulnerabilidad parte del hecho de que Alexa, cuando no entiende una instrucción o al utilizar la *wake word* [1] y no recibir una instrucción, pide al usuario que repita la orden. Esta funcionalidad puede ser explotada a partir de una skill desarrollada para ello[8].

Para explotar esta vulnerabilidad es necesario reproducir esta característica de *reprompt*, en los 8 siguientes segundos Alexa esperará, por segunda y última vez, que el usuario repita la instrucción. Durante este lapso de tiempo y teniendo instalada una *skill* maliciosa, se podría conseguir una transcripción de lo ocurrido durante ese tiempo.

Amazon asegura haber implementado mitigaciones para detectar este tipo de comportamientos y haber rechazado o suprimido las *skills* que persigan reproducir este fin. Parece ser que efectivamente, esta vulnerabilidad ha sido solucionada, pues está calificada como vulnerabilidad con un impacto bajo con una puntuación de 3.3 sobre 10 y con un vector CVSS de AV:L/AC:L/PR:N/UI:R/S:U/C:L/I:N/A:N.

Al no poder disponer de una *skill* para emular el ataque, no es posible realizar la prueba.

6.1.2. Vulnerabilidades CVE-2017-1000251 y CVE-2017-1000250

Estas vulnerabilidades afectan a BlueZ, la pila del protocolo Bluetooth que utiliza el *kernel* de Linux desde la versión 2.6.32 hasta 4.13.1[9], ambas inclusive, sobre el cual están desarrollados algunos de los modelos de los diferentes dispositivos Echo.

El vector CVSS de la vulnerabilidad CVE-2017-1000250 es AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N y AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H para la vulnerabilidad CVE-2017-1000251, lo que representa un potencial impacto medio (6.5) y alto (8.5), respectivamente.

La vulnerabilidad CVE-2017-1000251 permitiría ejecutar código en remoto mientras que la CVE-2017-1000250 produce una filtración del servidor SDP *Service Discovery Protocol*, ligado a la conexión Bluetooth. Aprovechando ambas vulnerabilidades, se permitiría, gracias a un *stack overflow* o desbordamiento de pila, dar al atacante acceso a una consola, pudiendo así, por ejemplo re programar las respuestas de Alexa desde otro dispositivo conectado vía Bluetooth.

Fundamentalmente, este problema surge de que el Amazon Echo está constantemente escuchando posibles conexiones Bluetooth. Teniendo en cuenta la limitada interfaz de usuario no hay forma de apagar la conexión ni la de instalar algo similar a un antivirus. En resumidas cuentas, representa una brecha de seguridad importante en el asistente, algo de lo que se aprovechó BlueBorne[10].

El software desarrollado por el laboratorio, los atacantes pueden tomar el control de un dispositivo vulnerable y usarlo para un sinfín de fines maliciosos, aun así, Armis mantiene la distribución de este programa controlada bajo un programa de demos, al cual se ha solicitado acceso, aunque sin contestación, con tal de reproducir el ataque. Tras comprobar la efectividad del ataque, Armis informó a Amazon, quién lanzó la versión v.591448720 de su software, que, teóricamente, solventaba esta brecha.

Al no poder disponer del software para emular el ataque, no es posible realizar la prueba.

[1] Palabra clave a la que espera Alexa para empezar la escucha.

6.2. *Metasploit Framework*

Metasploit es un *framework* de *pentesting*, que ofrece al usuario un seguido de ataques que explotan diferentes vulnerabilidades de forma sencilla y rápida. Se realiza una búsqueda con palabras clave *Alexa*, *Echo* y *Amazon* y no se encuentra ningún *exploit*.

6.3. Ataques basados en *API*

Como se ha comentado anteriormente, las *Application Programming Interface* ofrecen diferentes métodos y funciones de un servicio para facilitar el desarrollo de complementos del servicio (en esta caso *skills*), actuando así como capa de abstracción.

El problema aparece cuando el desarrollador de la, en esta caso Amazon, tiene que encontrar un método para encontrar el equilibrio entre las posibilidades que puede ofrecer y la integridad, confidencialidad y disponibilidad de sus servicios, y por ende, de la información de sus usuarios. De este modo, diferentes desarrolladores han creado sus propias *skills* que, utilizando la *API* que ofrece Amazon, consiguen datos de sus usuarios, como podemos ver en [11].

Aunque existan *APIs* que puedan filtrar datos a terceros, sigue siendo necesario activar la *skill* y usar una cuenta de Amazon, por lo que si no se utilizan funcionalidades de dudoso origen no debería ser posible realizar este tipo de ataques. Además, para hacer públicas estas *skills* es necesario que pasen el filtro requisitos de privacidad de Amazon.

Al no disponer de una *skill* verificada por la multinacional que haya sido desarrollada con este fin, no es posible reproducir este tipo de ataque.

6.4. Ataques basados en sonido

La forma en la que se inicia la interacción con Alexa es mediante la *wake word*, se plantea entonces la posibilidad de utilizar una *wake word* o incluo dar una orden utilizando una voz emulada, y comprobar, si sigue siendo posible utilizar el asistente mediante un servicio de *text-to-speech*.

Aunque sea posible lo planteado anteriormente, Alexa está diseñada para no revelar información del usuario de forma verbal, aun así, sigue siendo posible ejecutar acciones tal que comprar en Amazon.

Según [1], es posible utilizar Alexa a través de barreras, como por ejemplo una puerta o desde una distancia considerable. En cuanto a experiencia de usuario esto puede resultar algo útil y bueno, aunque acaba representando una vulnerabilidad importante, que podrían usar nuestro asistente de forma maliciosa sin siquiera esta mínimamente cerca.

Dado que si alguien intenta tomar acceso de Alexa a través de una barrera o a una larga distancia el dueño del dispositivo puede percatarse, se trata de conseguir un audio distorsionado, que no pueda ser fácilmente entendido por una persona aunque sí para el Echo.

6.4.1. Interacción con Alexa usando un simulador de voz, voz distorsionada o incluso ininteligible para el oído humano

Existen muchos simuladores de voz, aunque pocos proporcionan una naturalidad del habla suficiente como para ser comprendido por Alexa, de modo que se utiliza una demo de *Watson*[12], una herramienta que ofrece un servicio *text-to-speech* de la mano de IBM, el cual sí ofrece un audio y narrado de calidad.

Las pruebas se realizan con una instrucción sencilla, como por ejemplo, "*Alexa, ¿cuál es la capital de Estados Unidos?*". Tal y como se menciona anteriormente, podemos comparar el espectrograma de la voz emulada con el de una voz humana y comprobar que efectivamente, es muy similar mostrando las pausas naturales del lenguaje y con un nivel de voz uniforme, como se puede observar en la Figura 3.

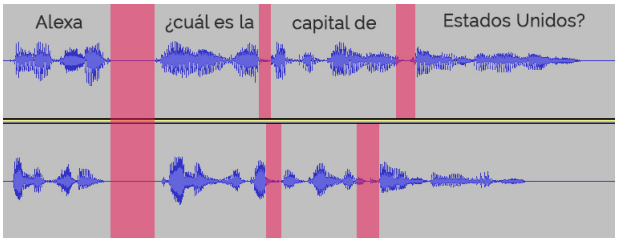

Fig. 3: Comparación entre voz emulada (superior) y voz humana (inferior)

Tras diferentes intentos para comprobar la efectividad de la locución, se comprueba que siempre se detecta y responde a la voz simulada. Una vez conociendo que es funcional, se intenta modificar el audio todo lo posible con el objetivo de que no sea fácilmente entendible por el oído humano (recordemos que el objetivo es evitar descubierto). La máxima distorsión que se consigue realizar es una modificación del *pitch* o tono aumentado en un 62 % y un aumento de la velocidad del 5 %, a partir de esos parámetros el Echo deja de entender la locución. Estas modificaciones no son suficientes para distorsionar el audio de tal forma que sea ininteligible, de manera que no se puede utilizar como un método de ataque, por ejemplo, intercalando la locución distorsionada en una canción.

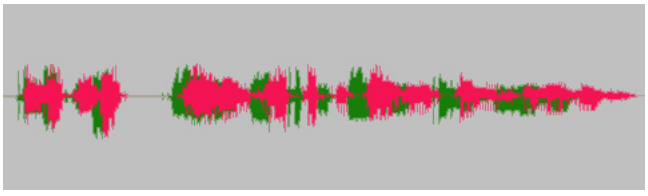

Fig. 4: Espectrograma de voz modificada (verde) y voz funcional (rosa)

En lo que a sonidos inaudibles respecta, es necesario hacer que el audio supere los 20KHz (límite superior de la escucha humana) para conseguirlos. Además de un equipo especializado para ello, no se disponen de las herramientas

para reproducir el ataque, aun así se dispone de anteriores estudios que verifican su funcionalidad. El primer paso es aplicar un filtro pasa bajos a 8KHz, para eliminar las frecuencias altas, seguido de un proceso de *upsampling* para aumentarla frecuencia de muestreo y llegar a los +20KHz gracias a la modulación de ultrasonidos. Finalmente es necesario regresar a frecuencias audibles para que las pueda captar el micrófono del dispositivo, aunque esta vez no es audible para la escucha humana [13].

6.4.2. Interacción con Alexa a través de barreras físicas

Según la documentación de estudios anteriores, debería ser posible la interacción con el dispositivo Amazon Echo a través de barreras físicas como una pared, puerta o incluso una larga distancia. Se propone entonces, el reproducir este comportamiento haciendo uso de un tono de voz moderado y natural de un uso común del dispositivo a través de una puerta cerrada, una pared y a una distancia considerable.

En el caso de la puerta resulta posible la interacción, teniendo en cuenta un grosor de puerta de madera de 5cm de grosor y subiendo ligeramente el tono del habla. Aunque similar, la prueba con la pared no resulta exitosa, la cual atenúa y/o distorsiona el sonido como para no reproducir este comportamiento con Alexa, muy posiblemente debido al grosor y material de la misma. Finalmente, se resuelve con éxito el hablar a 5 metros de distancia y ser respondido por Alexa subiendo ligeramente el tono del habla, del mismo modo que la puerta.

6.4.3. Skill Squatting

La transcripción del audio escuchado por el Echo (que recordemos, se realiza en los servidores de Amazon), resulta una caja negra a ojos del usuario, y como hemos visto con en pruebas anteriores, puede funcionar incluso con voces emuladas por un ordenador. Sin embargo, hay palabras, sobre todo anglosajonas, que debido a su fonética, se puede confundir con otras muy similares, algo de lo que se aprovecha el *Skill Squatting*.

Esta técnica se basa en la semejanza entre dos fonemas, como por ejemplo, en las palabras *coal* y *call* [14]. Aunque parezca algo absurdo, esto abre una gran brecha de seguridad si, sin intención alguna, un usuario ejecuta una *skill* maliciosa, que por ejemplo, vulnere su privacidad haciendo uso de los datos disponibles en la API.

Skill objetivo	*Squatted* Skill	Éxito
Coal	Call	100.0 %
Heal	He'll	96.4 %
Dull	Doll	80.8 %
Sweeten	Sweden	57.4 %
Bean	Been	17.8 %

Tabla 1: Palabras susceptibles a ser malinterpretadas. Extraídas de [14].

Aunque sea posible cambiar el idioma de Alexa, no es posible emular este comportamiento, dado que los acentos son algo que el reconocimiento de Alexa tiene en cuenta, por lo que se necesitaría de un Echo de un país de habla inglesa. Además, tampoco se dispone de un método para comprobar la transcripción de las instrucciones.

6.5. Ataques basados en red

Para las siguientes pruebas se parte de una red doméstica como la mostrada en la Figura 5, que dispone de un rango de direcciones IP de 192.168.1.0/24, asignadas mediante protocolo DHCP (*Dynamic Host Configuration Protocol*), lo que implica que las direcciones IP reflejadas en la imagen pueden variar a lo largo de las pruebas.

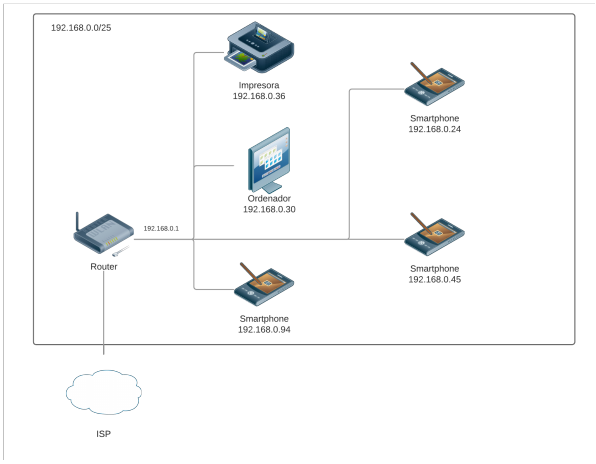

Fig. 5: Diagrama de red sin Amazon Echo

De tal modo, se realiza la instalación del dispositivo que ejecuta Alexa, en este caso un dispositivo Amazon Echo, siguiendo el asistente de instalación. La red, entonces, quedaría como la que se muestra en la Figura 6. A destacar, la dirección MAC del Echo, que será de utilidad próximamente para filtrar paquetes e intentar suplantar la identidad del dispositivo.

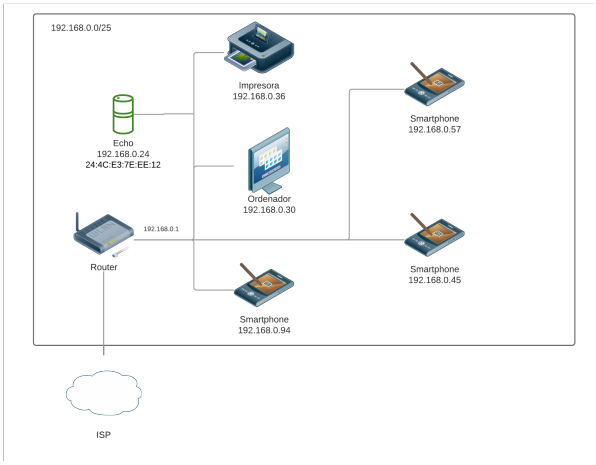

Fig. 6: Diagrama de red con Amazon Echo

Con el fin de interceptar el tráfico enviado y recibido por el Amazon Echo usaremos Wireshark[15], una herramienta de software libre que permite analizar el tráfico en diferentes

protocolos de comunicación. Esta herramienta se estará ejecutando en el único ordenador de la red. Además, se utiliza el modo promiscuo, que permite capturar todo el tráfico de la red, incluyendo los que no van dirigidos al *host*. Tras realizar el análisis y filtrar el tráfico del Echo, no aparece ningún tipo de tráfico. Esto es debido a que no existe un medio compartido (cable) y el router actúa como *switch*.

Viendo que la arquitectura anterior no resulta útil a la hora de analizar las comunicaciones entre el Echo y el router de la red, es necesario diseñar un nuevo modelo que permita ver este tráfico. Para solucionar esto, se monta un *proxy* en la red, lo que acaba actuando como un ataque *Man-in-the-Middle*. Para esta tarea se usa Zentyal[16], una distribución de Linux Debian, con tal de ejecutar un servidor que actúe de *firewall*, tal y como se muestra en la Figura 7.

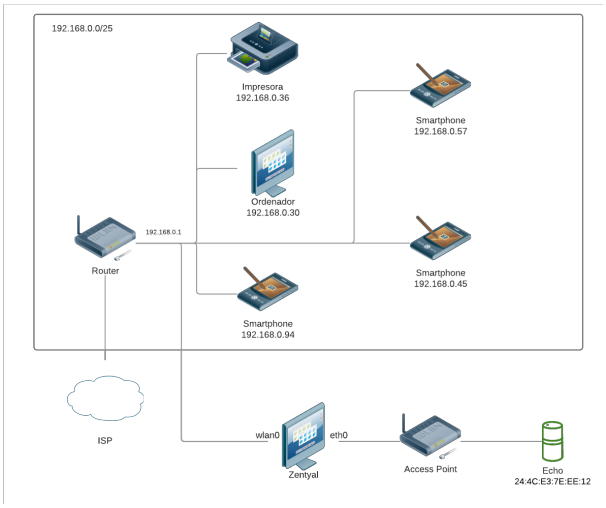

Fig. 7: Diagrama de red con proxy

Con tal de conseguir este fin se utiliza un router Linksys. Primeramente, se debe cambiar el modo de uso del router a *Access Point* o punto de acceso; el router debe recibir el tráfico de la red inalámbrica y reenviarlo a través de la red cableada hasta el servidor de Zentyal, es decir, debe usar el ordenador que ejecuta el proxy como *gateway*. Para este cometido, se debe conocer la IP que se le asigna al servidor en la red interna para poder especificarla en la configuración en el campo de puerta de enlace. Hay que tener en cuenta que las redes creadas por los dos routers de la red son del tipo 192.168.1.0/24, por lo que se debe cambiar el rango de una de las redes (en este caso la interna) a uno diferente, por ejemplo, 192.168.11.0/24, con tal de evitar colisiones. Destacar que el DHCP del router se mantiene activo, pero no supone un problema ya que la dirección IP asignada expira en un lapso de tiempo suficiente como para hacer las pruebas necesarias.

En este punto, el tráfico del Echo llega al servidor Zentyal, donde podemos controlarlo con acciones como excluir un tipo de tráfico y/o proviniente de un dispositivo concreto. El último paso es redirigir el tráfico proviniente del Echo hacia la red doméstica, usando la herramienta *ip* de Linux. Para ello se añade una regla por defecto para enrutar el tráfico desde el proxy hasta el router de la red doméstica.

Una vez realizada la instalación y comprobar que se tienen acceso desde la red secundaria a Internet, se conecta el Amazon Echo mientras se ejecuta Wireshark, de modo que se pueda analizar la conexión entre el Echo y el servidor de Amazon.

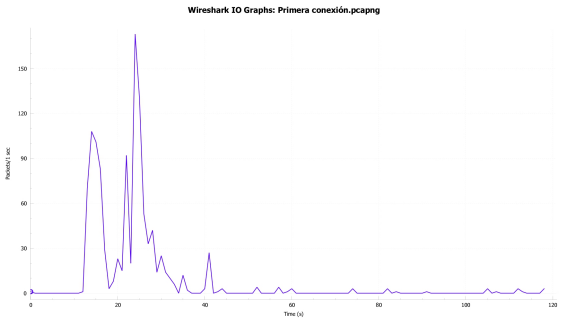

Fig. 8: Tráfico durante la conexión a la red Wi-Fi

Como se puede observar en la Figura 8, del segundo 10 al 40, aproximadamente, es cuando se realiza el registro del dispositivo en Amazon, se intercambian certificados y se finalizan los preparativos. Analizando más pausadamente este lapso de tiempo encontramos, podemos observar las siguientes comunicaciones, en orden:

1. Autenticación en los servicios activados (Amazon, Spotify...)
2. Comprobación de conectividad con los servidores de Amazon
3. *TLS Handshake*

En la Figura 9 podemos observar que la comprobación de conectividad con los distintos servidores se realiza en HTTP plano, ya que todavía no se ha producido el intercambio de claves TLS.

Source	Destination	Info
192.168.11.103	224.0.0.22	Membership Report / Join group 224.0.0.251 for any sources
192.168.11.103	104.199.65.35	58887 → 4070 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1 TSval=18685905 T
8.8.8.8	192.168.11.103	Standard query response 0xbd83 A d3p8zr0ffa9t17.cloudfront.net A 143.204.222.7
52.216.98.19	192.168.11.103	80 → 35936 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1432 WS=256 SACK_PERM=1
192.168.11.103	143.204.222.72	51967 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1 TSval=18685907 TSe
192.168.11.103	52.216.98.19	35936 → 80 [ACK] Seq=1 Ack=1 Win=87616 Len=0
192.168.11.103	52.216.98.19	GET /kindle-wifi/wifistub-echo.html HTTP/1.1
104.199.65.35	192.168.11.103	4070 → 58887 [SYN, ACK] Seq=0 Ack=1 Win=42980 Len=0 MSS=1240 SACK_PERM=1 TSval
192.168.11.103	104.199.65.35	58887 → 4070 [ACK] Seq=1 Ack=1 Win=87616 Len=0 TSval=18685909 TSecr=137553432
143.204.222.72	192.168.11.103	80 → 51967 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=1
192.168.11.103	143.204.222.72	51967 → 80 [ACK] Seq=1 Ack=1 Win=87616 Len=0 TSval=18685910 TSecr=1858034052
192.168.11.103	143.204.222.72	GET /HTTPConnTest.txt HTTP/1.1
143.204.222.72	192.168.11.103	80 → 51967 [ACK] Seq=1 Ack=191 Win=30208 Len=0 TSval=1858034056 TSecr=18685910
192.168.11.103	104.199.65.35	58887 → 4070 [PSH, ACK] Seq=1 Ack=1 Win=87616 Len=160 TSval=18685914 TSecr=137
143.204.222.72	192.168.11.103	HTTP/1.1 200 OK (text/plain)
192.168.11.103	143.204.222.72	51967 → 80 [ACK] Seq=191 Ack=475 Win=88704 Len=0 TSval=18685916 TSecr=18580340
52.216.98.19	192.168.11.103	80 → 35936 [ACK] Seq=1 Ack=147 Win=30464 Len=0
52.216.98.19	192.168.11.103	80 → 35936 [PSH, ACK] Seq=1 Ack=147 Win=30464 Len=356 [TCP segment of a reasse
52.216.98.19	192.168.11.103	HTTP/1.1 200 OK (text/html)
192.168.11.103	224.0.0.251	Standard query 0x0000 ANY Android.local, "QM" question ANY Android.local, "QM"

Fig. 9: Pruebas de conectividad

A destacar también, se puede observar, que se van produciendo pequeños intercambios de información después del segundo 40. Analizando estos paquetes observamos que las claves de cifrado de las comunicaciones se van actualizando periódicamente. Algunas otras comunicaciones que se producen sin hacer uso de Alexa se pueden tratar, según *Security Analysis of the Amazon Echo* [1], de actualizaciones de estado.

Analizando el tráfico que se genera durante el uso de Alexa, podemos observar que está cifrado mediante *TLS v1.2*, un protocolo criptográfico, con curvas elípticas y RSA.

En este punto, se propone la realización de un ataque de reproducción de paquetes, el cual consisten en capturar el tráfico que recibe el Echo durante un uso normal para replicarlo posteriormente. Cabe recordar que las claves de cifrado se renuevan periódicamente, de manera que el ataque se tiene que reproducir brevemente después de la captura de tráfico.

Para la captura de tráfico se hace uso de TCPDump[17] desde la máquina que está actuando como proxy. Una vez en posesión de un archivo del tráfico recibido mientras se le daban diferentes instrucciones a Alexa, se reenvía el tráfico sin cambiar ningún parámetro. Como resultado, aunque la herramienta indique que la totalidad de los paquetes han sido entregados, el Echo no ha reaccionado a estos paquetes, obteniendo así el mismo resultado que en [1].

El motivo de que un ataque de *replaying packets* falle viene derivado del uso del protocolo *TLS*, que desde la versión 1.1, ofrece dos ventajas que no se suelen destacar: garantía de la integridad y prevención de reproducción. Esto quiere decir que las comunicaciones tienen mecanismos de control para evitar la manipulación de cualquier porción de datos encriptados, además, estos mecanismos incluyen una serie de parámetros que evitan que el tráfico pueda ser replicado más tarde[18].

Así pues, no es posible realizar un ataque de suplantación de identidad, dado que para ello habría que romper la encriptación con curvas elípticas en un lapso de tiempo muy breve para evitar la renovación de las claves, mientras que no es posible hacer un ataque de replicación de paquetes ya que el protocolo *TLS* incluye mecanismos de protección para ello.

6.6. Vulnerabilidades en los dispositivos de terceros

Recordemos que el origen de los asistentes virtuales es hacer más cómoda la vida cotidiana de sus usuarios, como por ejemplo actuando como método de acceso común a los diferentes dispositivos del hogar, como bombillas, tomas de corriente o alarmas, de modo que se considera necesario también el análisis de estos dispositivos.

Se pueden clasificar estos dispositivos según el método de conexión que utilizan. En primer lugar, el más común hasta el momento, y con el cual trabajamos, es el caso en el que el dispositivo se conecta a la red y recibe las instrucciones desde el servidor del fabricante, el cual a su vez recibe las instrucciones desde el sistema de Amazon. Esto implica que a pesar de que Alexa sea uno de los métodos de control del dispositivo, también pueden existir otros, como por ejemplo una aplicación móvil del mismo fabricante que haga posible su interacción. Estos dispositivos siguen el mismo protocolo *TLS* que el Echo a la hora de conectarse a la red, por lo que nos encontramos en la misma casuística que en el apartado anterior y no es posible vulnerar su seguridad. Sin embargo, en caso de que el fabricante no utilice este tipo de protección, es posible suplantar la identidad del servidor para enviar ordenes al dispositivo, usando ingeniería inversa, tal y como se explica en *How easy is it to hack my WiFi light bulb?* [19].

El método de ataque a este tipo de dispositivos IoT que no utilizan un protocolo de encriptación, se basa en capturar un paquete proviniente del servidor y modificar únicamente el código hexadecimal que identifica a la instrucción, lo que permitiría encender, apagar o incluso conseguir datos de la cuenta asociada al dispositivo. Aunque usando *TLS* se podrían evitar este tipo de ataques, otros métodos que se proponen son el comprobar la marca temporal del paquete o calcular *checksums* de las peticiones.

Por otro lado encontramos *Zigbee*, un estándar global de comunicación para la radiodifusión digital de datos, diseñado para permitir el control y monitorización de dispositivos conectados, y formado por un conjunto de protocolos de alto nivel que opera sobre la especificación de radio IEEE 802.15.4 de la IEEE[20]. El funcionamiento de las redes Zigbee se fundamenta en un nodo que actúa de coordinador de la red, que es quien recibe, en este caso, las instrucciones desde Alexa y hace actuar a los diferentes dispositivos. Al no disponer de este tipo de dispositivos no es posible indagar en este tipo de redes y conocer posibles vulnerabilidades.

Así pues, no se han encontrado posibles vulnerabilidades en estos dispositivos. De todos modos, los datos que pueden tener estos dispositivos no son de gran importancia en lo relativo a la privacidad del usuario.

7 Conclusiones

A pesar de todas las pruebas realizadas, no se ha sido capaz de reproducir ningún ataque a una brecha de seguridad. Aun así, se considera necesario que algunas de las pruebas que no han podido ser realizadas se estudien en un futuro, por ejemplo, la comprobación de que un dispositivo Amazon Echo no puede ser atacado por el software que aprovecha las vulnerabilidades CVE-2017-1000250 y CVE-2017-1000250. Otra de las pruebas que sería necesario reproducir con el equipamiento y conocimientos adecuados es la de la creación de un audio que resulte inaudible y/o ininteligible por el oído humano.

Otra vía de ampliación que se considera importante es la comprobación de las mismas pruebas en otros asistentes virtuales, como puede ser, por ejemplo, Google Home. Además, estos otros asistentes virtuales, también pueden presentar otras brechas de seguridad, debido a su funcionamiento, arquitectura o componentes.

Tras el estudio del funcionamiento y de las posibles brechas de seguridad de Amazon Alexa, podemos concluir en que la política de privacidad y seguridad que Amazon redacta en su documentación es totalmente fiable, y efectivamente, se toman muy en serio su compromiso con la seguridad de los datos de sus usuarios, debiéndose tratar

este caso como ejemplo para futuros proyectos. Siguiendo en la misma línea, parece ser que la política de publicación de *skills* también está funcionando como se espera, de modo que si mantienen los mecanismos o incluso se revisan periódicamente estos requisitos, los usuarios no deberían preocuparse por utilizar *skills* maliciosas, con o sin intenciones de ello.

Finalmente, y tras la realización de este estudio, hay que plantearse dónde tienen origen estos riesgos que son objeto de estudio. Como se comenta anteriormente, los asistentes virtuales nacen para hacer más cómoda la vida de los usuarios. De todos modos, es necesario reconsiderar si esta necesidad de comodidad nace en los propios usuarios, o si por un contrario se crea esta necesidad en ellos, fomentando así el consumismo en la sociedad.

REFERENCIAS

[1] W. Haack, M. Severance, M. Wallace and J. Wohlwend, ”Security Analysis of the Amazon Echo”, MIT, 2017. [En línea]. Disponible en: https://courses.csail.mit.edu/6.857/2017/project/8.pdf. [Accedido: 10- Jun- 2019].

[2] N. Zhang, X. Mi, X. Feng, X. Wang, Y. Tian and F. Qian, Ünderstanding and Mitigating the Security Risks of Voice-Controlled Third-Party Skills on Amazon Alexa and Google Home”, Arxiv.org, 2018. [En línea]. Disponible en: https://arxiv.org/pdf/1805.01525.pdf. [Accedido: 10 - Jun- 2019].

[3] Amazon.com Help: .ªlexa Feature Help”, Amazon.com, 2019. [En línea]. Disponible en: https://www.amazon.com/gp/help/customer/display.html/ref=hp_bc_nav?ie=UTF8nodeId=201952240. [Accedido: 11- Jun- 2019].

[4] Day, M., Turner, G. and Drozdiak, N. (2019). Amazon WorkersAre Listening to What You Tell Alexa. [En línea]. Disponible en: https://www.bloomberg.com/news/articles/2019-04-10/is-anyone-listeningto-you-on-alexa-a-global-team-reviews-audio [Accedido: 11- Jun- 2019]

[5] ”CVE - About”,cve.mitre.org,2019. [En líınea]. Disponible en: https://cve.mitre.org/about/index.html. [Accedido: 11- Jun- 2019]

[6] ”Common Vulnerability Scoring System SIG”, FIRST — Forum of In-cident Response and Security Teams, 2019. [En l ınea]. Disponible en: https://www.first.org/cvss/. [Accedido: 11- Jun- 2019]

[7] ÇVSS v3.1 User Guide”, FIRST — Forum of Incident Response and Security Teams, 2019. [En línea]. Disponible en https://www.first.org/cvss/user-guide. [Accedido: 12- Jun- 2019].

[8] ”NVD - CVE-2018-11567”, Nvd.nist.gov, 2019. [En línea]. Disponible en: https://nvd.nist.gov/vuln/detail/CVE-2018-11567. [Accedido: 12- Jun- 2019].

[9] ”NVD - CVE-2017-1000251”, nvd.nist.gov, 2017. [En lınea]. Disponible en: https://nvd.nist.gov/vuln/detail/CVE-2017-1000251. [Accedido: 12 - Jun- 2019].

[10] Armis. BlueBorne Information from the Research Team - Armis Labs. [En lınea] Disponible en: https://armis.com/blueborne [Accedido: 12 - Jun- 2019].

[11] P. Masson, ”Accessing Amazon Echo Data with JavaScript”, analyticphysics.com, 2015.[En lınea]. Disponible en: http://analyticphysics.com/Diversions/Accessing %20Amazon %20Echo %20Data %20with %20JavaScript.htm. [Accedido: 12 - Jun- 2019]

[12] ”Watson Text to Speech”, Ibm.com, 2019. [En lınea]. Disponible en: https://www.ibm.com/watson/services/text-to-speech/. [Accedido: 12 - Jun- 2019]

[13] L. Song and P. Mittal, ”Inaudible Voice Commands”, Arxiv.org, 2019. [En lınea]. Disponible en: https://arxiv.org/pdf/1708.07238.pdf. [Accedido: 12 - Jun- 2019]

[14] D. Kumar et al., ”Skill Squatting Attacks on Amazon Alexa”, Usenix.org, 2019. [En línea]. Disponible en: https://www.usenix.org/system/files/conference/usenixsecurity18/sec18-kumar.pdf. [Accedido: 12 - Jun- 2019]

[15] ”Wireshark · Go Deep.”, Wireshark.org, 2019. [En línea]. Disponible en: https://www.wireshark.org/. [Accedido: 12- Jun- 2019].

[16] Ïnicio - Zentyal”, Zentyal, 2019. [En línea]. Disponible en: https://zentyal.com/es/inicio/. [Accedido: 12- Jun- 2019].

[17] ”TCPDUMP/LIBPCAP public repository”, Tcpdump.org, 2019. [En línea]. Disponible en: https://www.tcpdump.org/. [Accedido: 13- Jun- 2019].

[18] E. Saad, J. Mackowsk, D. Righetto and J. Manico, ”Providing Transport Layer Protection with SSL/TLS-Benefits”, GitHub, 2019. [En línea] Disponible en: https://github.com/OWASP/CheatSheetSeries/blob/master/cheatsheets/TransportLayerProtectionCheatSheet.md. [Accedido: 13- Jun- 2019].

[19] M. Van de Wynckel, ”How easy is it to hack my WiFi light bulb? - Maxim Van de Wynckel - Medium”, Medium, 2016. [En línea]. Disponible en: https://medium.com/@Maximvdw/how-easy-is-it-to-hack-my-wifi-light-bulb-b6ba9192176d. [Accedido: 28- Jun- 2019].

[20] Y. F.M., Çonectividad ZigBee de Amazon Echo Plus: qué es, cómo funciona y otros dispositivos compatibles”, Xataka.com, 2018. [En línea]. Disponible en: https://www.xataka.com/basics/conectividad-zigbee-amazon-echo-plus-que-como-funciona-otros-dispositivos-compatibles. [Accedido: 13- Jun- 2019].