UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

APLICACIÓN MÓVIL PARA EL MONITOREO DE LA SEGURIDAD EN REDES INALÁMBRICAS EN INTERNET

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERA CIVIL EN COMPUTACIÓN

KYRA ANTONIA COSSIO GUTIÉRREZ

PROFESOR GUÍA:
ALEJANDRO HEVIA ANGULO

MIEMBROS DE LA COMISIÓN:
ÉRIC TANTER
CESAR GUERRERO SALDIVIA

SANTIAGO DE CHILE
2021

Resumen

El *Laboratorio de Criptografía Aplicada y Ciberseguridad (CLCERT)* es una iniciativa de la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile que monitorea y analiza problemas de seguridad en nuestro país. Dentro de los proyectos del *CLCERT* se encuentra el *Observatorio de Seguridad de la Red Chilena (OSR)*, el que almacena información multifuente referente a escaneos de la Internet chilena.

Una limitante que presenta este observatorio es la carencia de una manera efectiva y representativa para recolectar datos relativos a la redes locales. Para que lo anterior fuese posible, se requería una cantidad no menor de voluntarios dispuestos a medir las redes en las que se conectan. Esto motivó la creación de una aplicación móvil que, por medio de *crowdsourcing*, obtuviera datos sobre varios puntos de accesos WiFi en Chile y la configuración de sus redes locales. Para motivar a que las personas participen como voluntarios, se planteó que la aplicación no sólo debiera recolectar datos, si no que también debiera entregar información útil a sus usuarios, de manera que esta les sirva cómo una herramienta de seguridad cuando se conecten a alguna red inalámbrica WiFi.

Los datos que almacene el CLCERT a través de esta aplicación, permitirán en un futuro obtener información valiosa respecto al estado de las redes internas en Chile. Por ejemplo, con estos datos se podría fiscalizar qué tan preocupados están los ISP de las redes de sus clientes o se podrían crear campañas focalizadas respecto a qué vulnerabilidades se enfrentan las personas y cómo protegerse frente a ellas.

La solución presentada en este trabajo es una prueba de concepto de la aplicación ya mencionada. Consistente en el diseño e implementación de un modelo de datos, una API Rest y una aplicación de Android. Desde el punto de vista funcional, la aplicación permite ejecutar 4 tipos de mediciones: “Medición Básica”, “Medición de Velocidad”, “Búsqueda de Otros Dispositivos” y “Detección del Bloqueo de Sitios Web”. La primera entrega datos respecto a los protocolos de seguridad inalámbrica y el DNS; la segunda sobre la velocidad de subida, bajada y el tiempo de viaje ida y vuelta (ping); la tercera da información sobre los dispositivos que están conectados en la red y finalmente la cuarta permite detectar si una página posee indicios de estar siendo bloqueada.

Luego de una acotada validación se concluye que este trabajo cumple con los requisitos propuestos. Asimismo, la información obtenida en la validación con usuarios, pese a no ser representativa, posibilitó recabar comentarios de caso de uso real a fin de mejorar la aplicación. Para finalizar, cabe señalar que el desarrollo actual permite extender la aplicación con nuevas mediciones de forma fácil, e incluso su posible integración con otras aplicaciones.

A mis padres Francia y Cristian, y a mi hermana Alanis
gracias por el amor incondicional que siempre me entregan.

Agradecimientos

A mis padres, mi hermana, mis abuelas y mi abuelo, mis tías y mis tíos, mis primas y mis primos, gracias por ser parte de mi familia, apoyarme y creer en mi. Espero que cuando se acabe la pandemia podamos volver a encontrarnos.

A Eduardo mi pareja, gracias por acompañarme, darme tu apoyo, ayudarme siempre y diseñar el logo de esta aplicación.

A la Comunidad Felicidad y la ComDibujo. La pasé demasiado bien con ustedes, aprendí mucho y conocí a mucha gente genial. Especialmente a la Nany, Tomimi y la Marcia quienes me acompañaron en la directiva de ComDibujo.

Al DCC por ser el mejor Departamento de la Facultad. Gracias a los profesores por todo lo que pude aprender con ustedes, cuando entré a la facultad no me imaginé que terminaría aquí. A mis compañeros gracias por todas las risas y buenos momentos que pasamos juntos en la salita, clases y en el chat del DCC.

A mi profesor guía Alejandro Hevia por confiar en mi este proyecto y ayudarme durante todo el desarrollo de éste.

A todos quienes se inscribieron para probar esta aplicación.

Al sitio web *Flaticon* desde donde obtuve los íconos presentes en las figuras de esta memoria.

Tabla de Contenido

Índice de Ilustraciones

Introducción

Antecendentes

El *Laboratorio de Criptografía Aplicada y Ciberseguridad (CLCERT)* es una iniciativa de la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile cuya misión es *"monitorear y analizar los problemas de seguridad de los sistemas computacionales en Chile, y generar tanto el conocimiento como el recurso humano especializado para asegurar dichos sistemas"* [5].

Dentro de los proyectos del *CLCERT* se encuentra el *Observatorio de Seguridad de la Red Chilena (OSR)*, un sistema centralizado que procesa y almacena datos referentes a escaneos periódicos realizados sobre la Red Chilena provenientes de distintas fuentes, con la finalidad de proveer al laboratorio información actualizada sobre el estado de la Red Chilena [31]. Actualmente, la información recolectada por estos escaneos está limitada a los datos que se pueden obtener desde la *Internet pública*, por lo que el *OSR* no cuenta con información relacionada a las redes internas chilenas.

Para resolver lo anterior, se propone crear una aplicación que permita a los usuarios ejecutar mediciones desde la red en la que están conectados y enviar los resultados obtenidos al OSR, los que podrán ser utilizados en futuros estudios.

Motivación

Los hogares que poseen Internet fija en Chile han ido al alza, alcanzado un 53,67 % en 2019. Sumado a lo anterior, la penetración total de la Internet fija y móvil (3G y 4G) ha aumentado de 112,9 accesos cada 100 habitantes en diciembre 2018 a 116,1 accesos en diciembre 2019 [6]. Otro antecedente importante a considerar es que a causa a la pandemia de *COVID-19* ha incrementado el uso de la Internet, situación que es reflejada en un estudio realizado por la *OECD* en mayo de 2020, el cual muestra que producto de la pandemia la utilización del ancho de banda en Chile aumentó de un 10,4 % a un 38,3 % [11]. Estos datos evidencian la importancia que ha adquirido la Internet en nuestro país, lo que justifica la relevancia que tiene estudiar los distintos aspectos de esta red.

Por otro lado, para recolectar información útil y representativa sobre las redes locales es necesario conectarse a una cantidad no menor de redes internas, por lo que se requiere conseguir varios voluntarios que estén dispuestos a ejecutar las mediciones que requiere el

laboratorio. Para que lo anterior sea realizable, es necesario que estas mediciones sean fáciles de ejecutar y que den información útil no sólo al CLCERT sino que también a sus usuarios.

Considerando estos antecedentes, este trabajo busca crear una herramienta que apoye la investigación que realiza el CLCERT de la Universidad de Chile sobre la seguridad de las redes inámbricas locales en nuestro país. Para ésto se diseñará y desarrollará una aplicación móvil que permita a quienes la utilizan obtener un reporte sobre la seguridad de las redes locales a las que están conectados y enviar la información recolectada al centro de investigación. Asimismo, esta aplicación tiene como fin ayudar a las personas a prevenir ciberataques provocados por conexiones a redes inalámbricas inseguras.

Objetivos

Objetivo General

El objetivo general de este trabajo es diseñar y desarrollar una aplicación móvil que recolecte información sobre el estado de la red local a la que está conectado un dispositivo, y que a su vez, entregue un reporte a sus usuarios sobre la calidad de su conexión a partir de las pruebas realizadas. La información recolectada será relativa a la seguridad, calidad y privacidad de la red del dispositivo y será utilizada para las recomendaciones del mismo (o futuros) dispositivos.

Objetivos Específicos

1. Identificar información que sea relevante para el estudio del estado de seguridad y privacidad de las redes inalámbricas locales en Chile.
2. Realizar un catastro de mecanismos que permitan obtener información útil e interesante para el análisis del estado de una red inalámbrica en dispositivos *Android*, enumerando las limitaciones que presenta este sistema operativo y los permisos requeridos para el funcionamiento de la aplicación.
3. Desarrollar una herramienta que permita al usuario ejecutar un set de mediciones en la red en la que esté conectado y le entregue información útil sobre el estado de esta red.
4. Validar la utilidad de la herramienta implementada con la finalidad de proponer mejoras y nuevas mediciones.

Descripción General de la Solución

Como se mencionó anteriormente, se implementará una aplicación *crowdsourcing* que muestre al usuario un set de mediciones que tras ser ejecutadas le entregarán un reporte del estado de su red y enviarán los datos recolectados al CLCERT.

Las mediciones que se podrán ejecutar desde la aplicación son las mencionadas a continuación.

1. Medición Básica: Revisa que el protocolo de seguridad inalámbrica de la red soporte WPA2 y que el servidor DNS configurado cumpla con una serie de requerimientos

mínimos de seguridad.

2. Medición de la Velocidad: Mide la velocidad de subida, bajada y el ping de la red, a partir de una prueba externa hecha por M-Lab [16], para luego entregar el percentil al que pertenece la red tras comparar los resultados obtenidos con las mediciones hechas por otros usuarios durante el año anterior.
3. Búsqueda de Otros Dispositivos: Busca los dispositivos que están conectados a la red y obtiene el fabricante de estos a partir de los primeros 3 bytes de los 6 bytes que contiene la dirección MAC.
4. Detección del Bloqueo de Sitios Web: Verifica si existen indicios de bloqueo de un sitio web ingresado por el usuario a partir de una prueba externa realizada por OONI [27].

Además, la aplicación mostrará al usuario los resultados de otras mediciones hechas en la red en la que está conectado.

Estructura de la memoria

Los capítulos de este trabajo están estructurados de la siguiente manera.

En el cápitulo 1 *"Marco Teórico"*, se presentan los conceptos relacionados a las mediciones implementadas, entre ellos los protocolos de seguridad inalámbrica, ping y DNS, también se mencionan ejemplos de iniciativas que buscan medir diferentes aspectos de la Internet por medio de crowdsourcing.

En el cápitulo 2 *"Definición del Problema"*, se plantea el problema que se busca solucionar junto con los requisitos funcionales y las consideraciones de privacidad y calidad solicitadas por el CLCERT.

En el cápitulo 3 *"Solución Propuesta"*, se explica la estructura del software desarrollado, explicando los componentes tanto de *backend* como de *frontend* implementados.

En el cápitulo 4 *"Validación de la Solución Implementada"*, se dan a conocer los resultados de la validación realizada con usuarios finales de la aplicación junto con la hecha con el CLCERT.

Finalmente, en el último cápitulo *"Conclusión"* se exponen las conclusiones y el trabajo futuro.

Capítulo 1

Marco Teórico

1.1. Protocolos de Seguridad Inalámbrica

El *IEEE 802.11 Wireless LAN*, mejor conocido como WiFi, es el estándar de las redes de área local inalámbrica. A lo largo de los años, en éste han definido varios protocolos de seguridad que solucionan las vulnerabilidades detectadas en sus predecesores. A continuación, se mencionarán los protocolos más relevantes creados por este estándar [30].

1. WEP (*Wired Equivalent Privacy*): Protocolo para cifrar los mensajes utilizando el algoritmo RC4. Este protocolo es vulnerable a ataques debido a que el vector aleatorio de inicialización usado para la generación de las llaves es de sólo 24 bits, lo que causa que las llaves utilizadas sean fáciles de obtener por medio de fuerza bruta. Otro aspecto vulnerable es que reutiliza el vector de inicialización, por lo que por medio de técnicas de análisis criptográfico es posible descifrar la información enviada sin siquiera tener la llave de cifrado. Sumado a lo anterior, WEP es vulnerable a ataques de *replay*, estos permiten a un atacante reenviar paquetes que fueron retenidos o previamente enviados por una fuente legitima.
2. WPA (*Wi-Fi Protected Access*): Protocolo que surge en respuesta a los problemas de seguridad presentados en WEP y es retrocompatible con los equipos que soportaban WEP. Asimismo, se caracteriza por cifrar los mensajes por medio del protocolo TKIP (*Temporal Key Integrity Protocol*) y por incluir como método de autenticación EAP (*Extensible Authentication Protocol*). Además, implementa MIC (*Message Integrity Check*) para prevenir ataques de *replay*. Sin embargo, este protocolo ya no es recomendado dado que se descubrió que el método de cifrado que utiliza es vulnerable a ataques criptográficos [1].
3. WPA2 (*Wi-Fi Protected Access, Version 2*): Protocolo que cifra los mensajes por medio de CCMP (*Cipher Block Chaining Message Authentication Code Protocol*), utilizando el cifrado de bloques AES (*Advanced Encryption Standard*) e incluye como métodos de autenticación *WPA2-Personal* y *WPA2-enterprise*.
4. WPA3 (*Wi-Fi Protected Access 3*): Protocolo anunciado en junio de 2018, ofrece protección frente a ataques de fuerza bruta y utiliza como handshake SAE (Simultaneous Authentication of Equals), el que permite *Forward Secrecy*, lo que impide que un ata-

cante descifre mensajes antiguos si encuentra una llave generada posteriormente [2]. Sin embargo, son muy pocos los dispositivos que actualmente lo soportan. Según el proyecto WiGLE, el porcentaje de dispositivos que ellos han detectado que son compatibles con este protocolo es de 0.00014 % [33].

1.2. El Sistema de Nombres de Dominio (*Domain Name System*, DNS)

DNS es un protocolo de la capa de aplicación que posibilita obtener las direcciones IP que están asociadas a un *hostname*. El componente de este protocolo encargado de traducir un *hostname* en una IP se conoce como *resolver*. En el paper *Evaluating "Health Status" for DNS Resolvers* [19] se plantea que un *resolver* se puede considerar seguro si es capaz de resistir ataques y dar respuestas seguras a sus clientes. Además, se propone allí que este aspecto se puede medir verificando si el resolver soporta DNSSEC (*DNS Security Extensions*) y si la aleatoriedad del puerto de origen es adecuada [19]. Otra métrica no mencionada en este paper pero que se utilizará en la aplicación, es si la aleatoriedad del ID de transacción es apropiada.

1.2.1. DNSSEC (*DNS Security Extensions*)

El protocolo DNSSEC surge de la necesidad de una autenticación más fuerte en el protocolo DNS a causa de problemas de seguridad detectados en el mismo. Para solucionar esto, DNSSEC autentica los datos con una firma digital dada por el propietario de estos. El uso de esta tecnología mejora DNS ya que permite al *resolver* verificar criptográficamente que los datos fueron enviados desde la zona que debieron ser emitidos y que los datos no han sido modificados en el trayecto [12].

Para comprobar que DNSSEC está presente en un resolver DNS, se puede realizar una consulta DNS a un sitio que soporte DNSSEC y luego verificar que los valores de las *flags* AD (*Authenticated Data*) y DO (*DNSSEC OK*) sean 1. También es necesario comprobar que exista el *record* RRSIG (*RRset Signature*) [19].

1.2.2. Envenenamiento de Caché DNS

El envenenamiento de caché de DNS consiste en *"introducir información falsa en una caché DNS, para que las consultas DNS devuelvan una respuesta incorrecta y se dirija a los usuarios a sitios web equivocados"* [4]. Las siguientes deficiencias en algunas implementaciones de DNS facilitan que los resolvers sean vulnerables a este ataque.

1. Insuficiente Aleatoriedad del ID de Transacción: el protocolo DNS especifica que el ID de transacción tiene un tamaño de 16 bits y es generado aleatoriamente, con lo que un atacante requeriría en promedio 32.768 intentos para adivinar este valor. Si una implementación posee un tamaño menor o el generador de aleatoriedad no es fuerte, un atacante necesitaría menos intentos para encontrar el ID, pudiendo ser vulnerable a ataques de envenenamiento.

2. Puerto de Origen Fijo para Generar Consultas: esto ocurre cuando se asigna un puerto de origen de manera arbitraria o aleatoria y se reutiliza en todas las consultas salientes. Debido a que la entropía dada por el ID de transacción no es suficiente, se recomienda que para cada consulta se genere un puerto de origen aleatorio, lo que agrega aproximadamente 16 bits de aleatoriedad a los datos que un atacante debe intentar adivinar para ejecutar un envenenamiento de caché DNS [28].

El DNS Operations, Analysis, and Research Center (DNS-OARC) es una organización *"sin fines de lucro que busca mejorar la seguridad, la estabilidad y la comprensión de la infraestructura del DNS de Internet"* [9]. Esta organización desarrolló una herramienta que permite verificar por medio de una consulta DNS de tipo TXT si el *resolver* que se está utilizando presenta alguno de los problemas mencionados anteriormente en el puerto de origen o en el ID de transacción [8, 7].

1.3. Medición de la Calidad, Seguridad, Privacidad y Censura en la Internet

En esta sección se mencionan algunos ejemplos de iniciativas que buscan medir los niveles de calidad, seguridad, privacidad y censura en Internet, por medio de *crowdsourcing*.

1.3.1. Open Observatory of Network Interference (OONI)

El Open Observatory of Network Interference (OONI) es *"un proyecto de software libre que pretende potenciar iniciativas de descentralización para aumentar la transparencia de la censura en Internet en todo el mundo"* [23].

Desde el 2012, OONI desarrolla *OONI Probe, "un proyecto de software libre (bajo el proyecto Tor) que apunta a descubrir la censura de Internet en el mundo"*. Este programa permite medir el rendimiento de la red y detectar si desde el punto de la red donde se ejecuta existe: sitios bloqueados por el proveedor de acceso a Internet, bloqueo de aplicaciones de mensajería instantánea, bloqueo de herramientas como *Tor* y *Psiphon* y/o la presencia de *middleboxes* [25].

A continuación se detallará el funcionamiento de las pruebas utilizadas en este trabajo.

Detección de Bloqueo de Sitios Web

Esta prueba sigue estos pasos:

- Identificación del *resolver*: Se determina cuál es el resolver que está siendo usado por omisión en la red.
- DNS *Lookup*: Si se entrega una IP, se omite este paso. Si se entrega un *hostname*, se realiza una consulta DNS de tipo A usando el *resolver* identificado en el paso anterior.
- Conexión *TCP*: Se intenta realizar una conexión TCP en el puerto 80 de la IP obtenida anteriormente. En caso de que la URL entregada comience con HTTPS, también se intenta realizar una conexión TCP en el puerto 443.

- Solicitud HTTP GET: Se realiza un solicitud HTTP GET a la URL dada y se guarda la respuesta.
- Comparación con las Medidas de Control: desde un *resolver* de control que no tiene problemas de censura, se realizan paralelamente las 3 últimas mediciones con el fin de hacer las siguientes comparaciones con los resultados obtenidos por la prueba.
  - Respuestas DNS: Si se realizó el paso 2 y la respuesta obtenida es la misma que la dada por el control, se señala que este aspecto es consistente. Si no se cumple lo último, pero al menos alguna de las respuestas es concordante, se califica como *reverse match*. Por otra parte, si ninguna coincide, se considera inconsistente. En el caso en que no se haya realizado el paso 2, se señala que las respuestas son consistentes.
  - Conexión TCP: Se revisa si se logró establecer la conexión TCP, tanto por el control como por la prueba. Si el resultado es diferente, se considera que hubo un bloqueo.
  - Tamaño del cuerpo de la respuesta HTTP: Se calcula la proporción entre el tamaño del cuerpo del control y el del experimento. Si este valor es mayor a 0.7, se considera que las respuestas son concordantes. En el caso contrario, se considera que no lo son.
  - Cabecera HTTP: Se compara la cabecera HTTP del control y la prueba. En el caso de haber diferencias se considera inconsistente.
  - Códigos de estado HTTP: Se compara el código de estado HTTP del control y la prueba. Si son iguales, este valor es consistente.
  - *HTML title tag*: Se compara el *HTML title tag* del control y la prueba. En el caso de ser diferentes, este valor es inconsistente.
- Razón de bloqueo: Si la prueba no se pudo realizar, este valor será *null* y si no se sospecha que exista un bloqueo, será *false*. Para cualquier otra combinación de resultados, este valor entregará en que prueba se detectó bloqueo (TCP, DNS o HTTP). Si las respuestas DNS y HTTP son inconsistentes o si se obtiene *dns_lookup_error*, se señala que el bloqueo es por DNS. Por otra parte, si no hay problemas con DNS, pero el requerimiento HTTP falla, o si la respuesta HTTP es diferente de la dada por el control, se dirá que existe un bloqueo por HTTP. Para el caso del bloqueo por conexión TCP, se dice que este ocurre cuando la petición HTTP falla y se detecta un problema en el paso "Conexión TCP" [26].

NDT (Network Diagnostic Tool)

Esta prueba permite medir la velocidad y el rendimiento de una red. Para ello, se suben y descargan datos aleatorios en el servidor que esté más cercano al usuario y se mide la velocidad con que este proceso se realiza. Además, NDT recopila información de la capa TCP/IP, que le permitirá caracterizar y examinar la ruta que se toma entre el usuario y el servidor [24].

Los servidores utilizados pertenecen a otro proyecto de código abierto llamado *Measurement Lab (M-Lab)*, estos se caracterizan por estar topológicamente distantes de la red de los ISP. Lo anterior posibilita medir el tiempo que demora la conexión entre redes administrativamente independientes (*peering*), lo cual ocurre por ejemplo cuando un usuario intenta

acceder a contenidos o servicios que están alojados fuera de la red su ISP [20]. En otras palabras, *"NDT mide esencialmente la rapidez con la que se puede descargar un archivo a lo largo de una ruta completa de Internet mediante un único flujo de descarga y, como tal, sus mediciones reflejan con precisión y exactitud la experiencia del usuario al acceder a los archivos en Internet"* [29].

1.3.2. Netalyzr

El centro de investigación *ICSI (International Computer Science Institute)* desarrolló *Netalyzr*, una aplicación para plataformas móvil y web que permitía a sus usuarios obtener información sobre sus conexiones a Internet fijo y móvil, *"analizando cuán abierta y transparente es la conexión" [13], "por medio de una serie de pruebas que verifican si el tráfico está intervenido, y si se están tomando medidas para que la conexión sea rápida y libre de spam"* [22]. Sin embargo, este proyecto fue dado de baja a inicios del 2019, sin que su código fuera liberado [15].

1.3.3. WiGLE (Wireless Geographic Logging Engine)

Es un repositorio abierto de redes WiFi a nivel mundial al que se puede acceder por medio de una aplicación de escritorio y web. Estos datos son utilizados tanto en proyectos de investigación, como para supervisar en terreno el estado de una red o incluso para encontrar redes WiFi abiertas para conectarse.

La información es recolectada por medio de una aplicación Android llamada *WiGLE WiFi Wardriving tool*, esta le permite a los usuarios buscar redes WiFi abiertas y reportarlas a WiGLE. La plataforma también posee un ranking de la cantidad de reportes dados por cada usuario [32].

1.3.4. RouterCheck

RouterCheck es una aplicación *Android* de código cerrado desarrollada por *Sericon Technology Inc.*, una empresa de software canadiense. Esta aplicación permite a sus usuarios detectar problemas de seguridad en las redes de sus hogares, entregando un listado con las vulnerabilidades detectadas e instrucciones para solucionarlas.

A partir de los datos que se recolectaron con esta aplicación, la empresa desarrolló en 2015 una investigación sobre el estado de seguridad de las redes de los hogares, sin embargo, no se entrega la información desde cuales países ni en qué proporción se realizaron las mediciones, por lo que los datos no necesariamente reflejan el caso de Chile [14].

1.3.5. Adkintun Mobile y PePa Ping

En Chile, el laboratorio *NICLabs (Laboratorio de Investigación sobre Internet Protocols)* implementó dos aplicaciones para la investigación de la calidad de la internet móvil: *Adkintun Mobile* [3] y su sucesora *PePa Ping* [21]. La primera *"mide la cantidad de datos enviada y recibida durante el día, la calidad de la señal de la antena y el tipo de conexión"*. Con esta información se realiza estudios regionales periódicos accesibles por los usuarios [17]. Mientras

que la segunda aplicación simula una conexión con un servidor VPN cada 15 minutos, de manera de obtener información sobre el tráfico de internet para luego recopilar datos estadísticos sobre estas conexiones, su calidad (latencia y cantidad de paquetes perdidos) y "datos ambientales". Estos últimos corresponden a: la memoria RAM utilizada/total, la ubicación geográfica, la velocidad del usuario, la compañía de teléfono, la intensidad de la señal, la banda de frecuencia para el caso de las conexiones WiFi y para el caso de las conexiones móviles la tecnología (LTE, UMTS, etc) y los IDs antena [18].

Capítulo 2

Definición del problema

2.1. Planteamiento del problema

El CLCERT es un centro de investigación que monitorea y analiza problemas de seguridad en sistemas computaciones chilenos. Dentro de sus proyectos se encuentra el OSR, un sistema centralizado multifuente con información referente a escaneos realizados a la Red Chilena, los que permiten al laboratorio tener una visión general del estado actual de esta red. Sin embargo, los datos están limitados a lo que se puede obtener desde fuera de la red, por lo que no se cuenta con datos sobre cómo están configuradas las redes internamente.

Las redes internas a las que se conecta una persona no están exentas de problemas de seguridad. Sin embargo, no se cuenta actualmente con datos que permitan saber qué tan recurrentes son estos problemas. Esta información es muy valiosa, ya que podría ser utilizada para hacer campañas focalizadas frente a las vulnerabilidades que existan, de manera de informar a la población sobre las amenazas a las que están expuestos y cómo mitigarlas, o incluso se podría obtener estadísticas sobre la seguridad de las redes que son administradas por cada ISP, permitiendo fiscalizarlos.

Como se mencionó anteriormente, para obtener estos datos es necesario estar conectado dentro de la red interna. Para que esto sea representativo, se requiere que una cantidad no menor de voluntarios dispuestos a ejecutar una serie de pruebas de manera periódica. Es por ello que es necesario que la herramienta a desarrollar sea fácil de utilizar y que el usuario obtenga algo que lo motive a usarla.

Para responder a estas necesidades, se propone crear una aplicación para celular que permita ejecutar distintas pruebas de seguridad. Además, la aplicación debe no sólo entregar información útil al CLCERT sobre el estado de la red, sino que también al usuario, de tal manera que éste sea un incentivo para utilizar la aplicación.

2.2. Requisitos

2.2.1. Requisitos funcionales

Información básica sobre la conexión

El servidor tendrá la siguiente información básica sobre las mediciones recibidas: la IP pública, el ISP, el *timestamp* del *test*, el tipo de lugar desde donde se realizó la conexión (*hogar*, *transporte público*, *lugar de trabajo o estudio*, *lugar público*, *otro*) y el nombre del usuario.

Revisión de los Protocolos de Seguridad Inalámbrica

La aplicación detectará si el protocolo de seguridad inalámbrica de la red desde la cual se ejecuta la prueba es WPA2 o WPA3. Si la red no soporta la utilización de al menos uno de estos, se considerará que es insegura, lo que se debe informar al usuario.

Desde el punto de vista del servidor, la aplicación debe enviarle todos los protocolos que soporta la red, de manera de tener un registro de la compatibilidad de protocolos a nivel nacional.

Revisión DNS

La aplicación entregará información al usuario sobre la seguridad de su DNS. Para ello, se tomarán las siguientes medidas:

- Comprobar que el DNS primario y secundario configurados en el dispositivo sean diferentes.
- Revisar si DNSSEC es soportado por el *resolver* DNS.
- Verificar que la asignación de los puertos de origen por parte del *resolver* DNS tenga una aleatoriedad adecuada.
- Comprobar que la asignación de los IDs de transacción dada por el *resolver* DNS tenga una aleatoriedad adecuada.

Por parte del servidor, este recibirá la siguiente información:

- IPs del DNS primario y secundario configurados.
- IP del *resolver* DNS utilizado.
- Informar si existe el *record* RRSIG y los valores de las *flags* AD y DO, tras hacer la revisión de DNSSEC.
- Los valores de *rating* y desviación estándar dados tras medir la aleatoriedad del puerto de origen.
- Los valores de *rating* y desviación estándar obtenidos tras medir la aleatoriedad del ID de transacción.

Búsqueda de otros dispositivos

La aplicación permitirá al usuario buscar todos los dispositivos que están conectados en la red interna a la cual pertenece. Después, para cada dispositivo encontrado se identificarán los siguientes datos: la IP privada del dispositivo en la red interna, los primeros 3 bytes de la dirección MAC, el fabricante y si corresponde o no al router de la red interna.

Esta información se guardarán en el servidor y se mostrarán solo a quienes se encuentran conectados a la misma IP.

Medición de la velocidad

La aplicación medirá la velocidad y el rendimiento de la red por medio de la prueba externa NTD. Luego, mostrará al usuario el percentil al que pertenecen su velocidad de subida, bajada y el ping, tras comparar esos valores con mediciones previas hechas por M-Lab en Chile.

El servidor recibirá los siguientes valores obtenidos por el test de M-Lab: la velocidad de subida y bajada, el RTT (*Round Trip Time*) promedio, máximo y mínimo, el ping, el MSS (*Maximum Segment Size*), la tasa de retransmisión y el ID del reporte en OONI.

Bloqueo de sitios web

La aplicación comprobará si un sitio ingresado por el usuario presenta indicios de estar bloqueado por medio de la prueba externa de "Detección de Bloqueos de Sitios Web de OONI". Al usuario se le mostrará el sitio que ingresó y si este tiene o no indicios de estar bloqueado. En caso de que sea posible la existencia de bloqueo se le indicará qué aspecto de su conexión genera esta sospecha.

El servidor deberá recibir la siguiente información: la IP del resolver, el estado de detección de bloqueos, el tipo de Bloqueo, las IPs dadas por el test TCP con su estado respectivo, el estado de la consistencia del header HTTP, el tamaño de respuesta HTTP, el *Tag Title* de la respuesta HTTP y el ID del reporte de la medición en OONI.

Información Histórica

La aplicación mostrará al usuario los últimos 5 resultados de cada medición para la IP pública de la red en la qué esté conectado.

2.2.2. Consideraciones de Privacidad

A partir de los datos almacenados en el servidor, no se debe poder identificar a qué persona pertenecen. Para lograr esto, sólo se almacenará la IP pública del usuario. Este dato se almacenará ya que permite entregar a los usuarios un registro de mediciones previas y permitirá más adelante al CLCERT restringir la cantidad de reportes que son enviados desde la misma IP, mitigando de esta manera los ataques de denegación de servicio.

La ubicación del lugar desde el cual se realiza la medición no se almacenará en la base de datos, a pesar de es necesario pedir el permiso en la aplicación para realizar la prueba de prótocolo. Se decidió que este valor no se enviará ni guardará en el servidor debido a

que crearía una asociación entre la ubicación, fecha y la IP. Además, si bien en un inicio se pensó que este podría ser útil para entregar a los usuarios un mapa con todos los puntos de accesos públicos encontrados por otras personas en su zona, esto podría ser mal utilizado para reportar redes de hogar cómo públicas y en caso de tener vulnerabilidades dejarlas aún más expuesta a otros usuarios. Dado que tener información sobre el lugar desde el cuál se está realizando la medición puede ser útil para las investigaciones del CLCERT, se decidió que bastaría con recolectar información sobre la categoría del lugar, la que puede ser *hogar*, *transporte público*, *lugar de trabajo o estudio*, *lugar público*, *otro*.

Otro aspecto que se discutió fue la creación de cuentas de usuario que permitirían al CLCERT mantener un muro de agradecimientos con los colaboradores destacados. Debido a que este dato potencialmente podría llegar a generar una asociación entre una persona la fecha y su IP, se determinó que únicamente se le pedirá al usuario que ingrese un nombre voluntariamente, el cual no tendrá una autentificación. Lo anterior permitirá que varias personas tengan el mismo nombre en la base de datos, lo que fomentará que las personas formen equipos para reportar sus mediciones.

Respecto a la identificación del fabricante a partir de su MAC en la medición de la búsqueda de dispositivos, de los 6 bytes que contiene una MAC sólo se enviarán y guardarán en el servidor los 3 primeros bytes, dado que estos permiten identificar a la mayoría de los fabricantes. Sumado a que la MAC completa de un dispositivo es un dato extremadamente sensible.

2.2.3. Consideraciones de Calidad

El trabajo presentado corresponde a una prueba de concepto, por lo que no será lanzado en la *App Store* inmediatamente tras su realización. Sin embargo, el software debe ser un producto funcional, que pueda ser probado con un grupo cerrado de usuarios.

Capítulo 3

Solución Propuesta

3.1. Arquitectura de Software

La aplicación posee una arquitectura clásica de cliente servidor que permite la comunicación por medio de requests HTTP entre la aplicación móvil y el servidor del CLCERT. Además, algunas mediciones requieren que la aplicación se comunique con servicios externos, lo que se ve reflejado en la figura 3.1 donde las flechas 1, 2 y 3 muestran que la aplicación interactúa con los servidores de Akamai, DNS-OARC, OONI y M-Lab.

3.1.1. Tecnologías Utilizadas

Para el desarrollo de la aplicación móvil se determinó utilizar Android Nativo con un soporte mínimo de SDK de Android 5 *"Lollipop"* (API de nivel 21) dado que de esta manera la aplicación podría ser utilizada por el 94,1 % de los celulares que poseen Android. También, se decidió que la aplicación será sólo desarrollada para Android debido a que Apple es más restrictivo para el desarrollo de sus aplicaciones. Un ejemplo de esto es que se requiere un computador con este sistema operativo para poder compilar la aplicación a desarrollar.

Para el caso del *backend* de la aplicación para la creación de los *endpoints* se decidió utilizar el *Framework FastApi*[1] debido a que es *"uno de los frameworks de Python más rápidos que existen"* [10]. Además, posee soporte para la validación de datos y serialización. Para la base de datos se usó *PostgreSQL* con un ORM llamado *SQLAlchemy*. Se determinó usar *PostgreSQL* debido a que es utilizado en el OSR y posee varias funciones útiles para el manejo de direcciones IP y MAC. Se decidió también utilizar un ORM, pues ello facilita la mantención de la base de datos.

3.1.2. Diseño del Workflow y Procesos

Existen 2 flujos principales en la aplicación: ejecución de las mediciones y mostrar resultados. Para el primero, el usuario ejecuta alguna prueba, una vez obtenidos los resultados estos son enviados al servidor, el cual responde si se pudieron o no guardar con éxito, tras recibir

[1] `https://fastapi.tiangolo.com/`

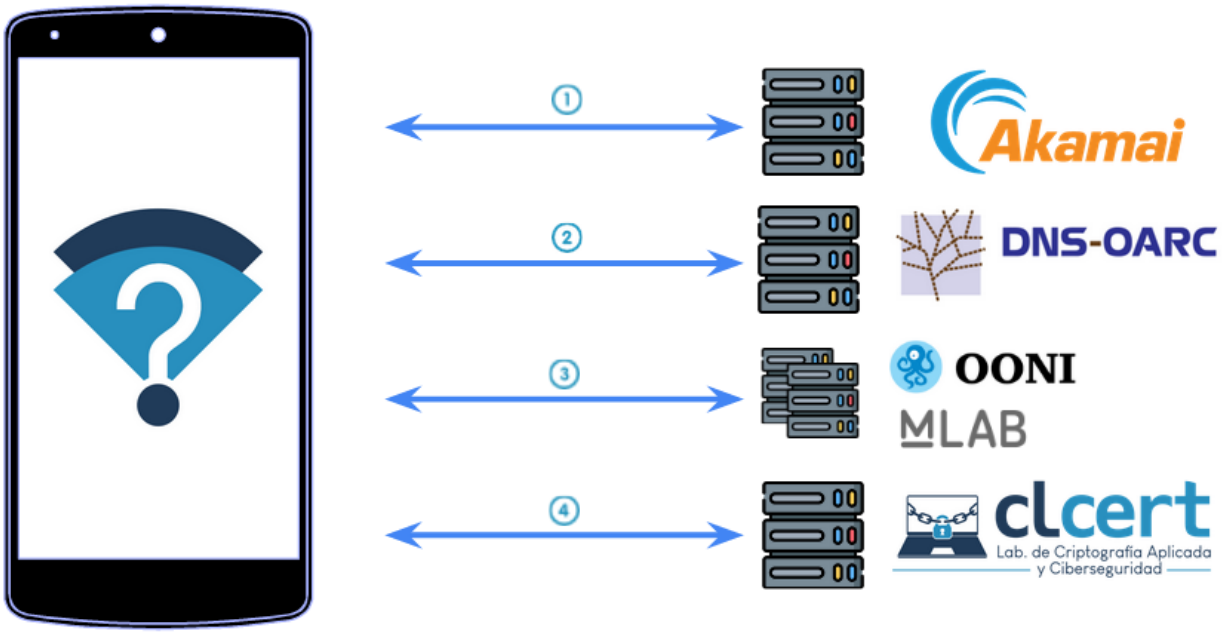

Figura 3.1: Flecha 1: Comunicación entre la aplicación y el servicio de Akamai que permite obtener la IP del *resolver* DNS. Flecha 2: Comunicación entre la aplicación con el servicio de DNS-OARC para obtener la aleatoriedad del puerto de origen y del ID de transacción del *resolver*. Flecha 3: Comunicación de la aplicación con OONI y M-Lab para acceder a las pruebas de bloqueo de sitios web y medición de la velocidad de la red. Flecha 4: Comunicación entre la aplicación y el servidor del CLCERT.

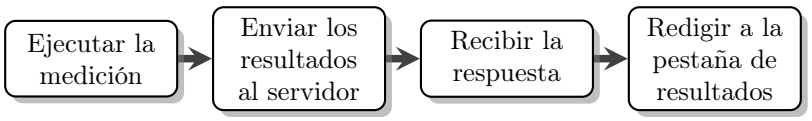

Figura 3.2: Flujo de ejecución de una medición.

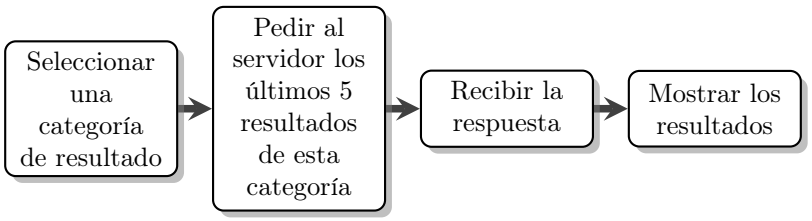

Figura 3.3: Flujo para obtener últimos 5 resultados de una categoría de medición.

esta respuesta la aplicación redirige al usuario a la pestaña "Resultados" (ver figura 3.2). Para el caso del segundo, el usuario está en la pestaña "Resultados" donde puede seleccionar una pestaña correspondiente a la categoría del resultado de su interés, tras lo cual la aplicación solicita al servidor los últimos 5 resultados de la categoría seleccionada. Después de recibir la respuesta del servidor se muestran los resultados en la aplicación (ver figura 3.3).

Se crearon las siguientes 4 categorías de mediciones: Medición Básica, Medición de la Velocidad, Búsqueda de Otros Dispositivos y Detección de Bloqueo de Sitios Web. Cada una posee un flujo de ejecución particular, los cuales serán explicados a continuación.

Medición Básica

Esta medición agrupa las pruebas sobre DNS y protocolos de seguridad inalámbrica. El flujo de la aplicación es el siguiente y se puede ver reflejado en la figura 3.4. Primero se obtienen las IPs del DNS primario y secundario, luego se obtiene el resolver que es utilizado por el celular, para ello se utiliza el servicio de Akamai que permite obtener esta información por medio de una consulta DNS de tipo TXT a `whoami.ipv4.akahelp.net`, con lo que obtiene una respuesta con la IP del *resolver* de la red. Después se obtiene la aleatoriedad de asignación de puertos de origen e IDs de transacción por parte del *resolver* utilizando el servicio de DNS-OARC, para ello se hace una consulta DNS de tipo TXT a `porttest.dns-oarc.net` y `txidtest.dns-oarc.net` respectivamente, las que son respondidas con el rating y la desviación estándar obtenida en cada caso.

A continuación, se ejecuta la comprobación del soporte de DNSSEC por parte del *resolver*. Para conseguir lo anterior, se hace una consulta DNS solicitando que los *records* DNSSEC para la página `paypal.com`, sólo si en la respuesta obtenida existe el *record* RRSIG y las *flags* AD y DO son iguales a 1, se concluirá que DNSSEC es utilizado, en cualquier otro caso se concluirá lo contrario.

Finalmente, se ejecutará la prueba de Protocolos de Seguridad Inalámbrica, en la que se obtendrán todos los protocolos que son soportados por la red y se verificará si entre ellos está WPA2 y/o WPA3.

Medición de la Velocidad

Para realizar esta medición se utiliza la librería de OONI, la que permite comunicarse con los servidores de M-Lab. El flujo de la aplicación es el mostrado en la figura 3.5, primero se asigna un servidor de M-Lab con el que se realizará la medición, aquí se suben y descargan una serie de archivos aleatorios que permiten medir la velocidad de subida y bajada de la red. Una vez terminada la medición, se envían los resultados a la aplicación, la cual los envía a OONI y posteriormente al CLCERT.

Para mostrar los resultados de esta medición el CLCERT enviará los datos con el percentil obtenido para la velocidad de subida, bajada y el ping, los que fueron obtenidos utilizando los datos de M-Lab almacenados BigQuery [2].

Búsqueda de Otros Dispositivos

Para buscar los dispositivos que están en la red primero se calculan todas las IPs internas posibles, luego se intenta hacer ping a cada una. Si se recibe una respuesta, la IP se guardará como alcanzable. Por el contrario, si no se recibe una respuesta antes de 1 segundo, se asume que el dispositivo no existe o no es alcanzable, con lo que se deja de intentar hacer ping (ver figura 3.6). Una vez terminado lo anterior, se revisa la tabla ARP para obtener las direcciones MAC de todos los dispositivos que fueron alcanzables, las que se guardarán censurando los últimos 3 bytes. Después, se obtiene el *gateway* para señalar en los datos la IP interna que corresponde al router.

Finalmente, los datos son enviados al servidor. Cabe señalar que cuando se muestran los resultados en la aplicación el servidor agrega la información relacionada al fabricante asociado a cada MAC.

Detección de Bloqueo de Sitios Web

Para detectar el bloqueo de un sitio web primero un usuario ingresa un sitio del que tiene sospechas que puede estar bloqueado. Después, la aplicación se comunica con el servidor de OONI para que inicie las pruebas de detección de bloqueo desde un servidor de control. Luego, tanto el control como la aplicación ejecutan las pruebas que permiten determinar si existen indicios de bloqueo del sitio ingresado (esto está descrito en mayor detalle en el capítulo *Marco Teórico)*. Una vez terminado lo anterior, el control envía sus resultados a la aplicación, donde se comparan lo obtenido por ambas partes para obtener un veredicto a partir de ambos. Finalmente se envían los resultados al servidor de OONI y después al del CLCERT (ver figura 3.7).

3.2. Implementación Backend

3.2.1. Diseño de la base de datos

La base de datos posee el modelo entidad relación presentado en la figura 3.8. El modelo posee una tabla central llamada `tests`, esta contiene la información básica de una medición,

[2] `https://www.measurementlab.net/data/docs/bq/quickstart/`

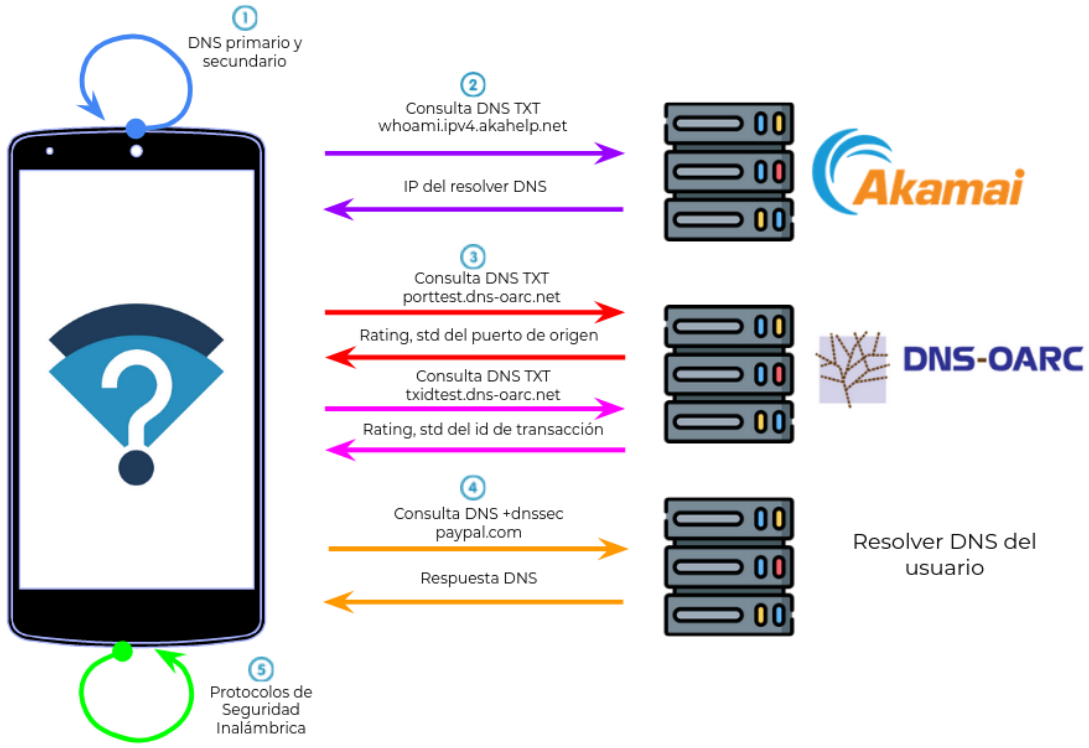

Figura 3.4: Flujo de prueba de Medición Básica. Los números indican en el orden en que se ejecutan los grupos de pruebas asociadas a esta medición. La fecha azul muestra la obtención del DNS primario y secundario configurado en el celular. Las fechas moradas corresponden a la consulta para conseguir la IP del *resolver* utilizado. Las flechas naranjas representan la adquisición de la aleatoriedad de la asignación de puertos de origen. Las flechas rosadas muestra la obtención de la aleatoriedad de asignación de los IDs de transacción. Las flechas amarillas representan la verificación del soporte para DNSSEC. La flecha verde corresponde a la obtención de los Protocolos de Seguridad Inalámbrica soportados.

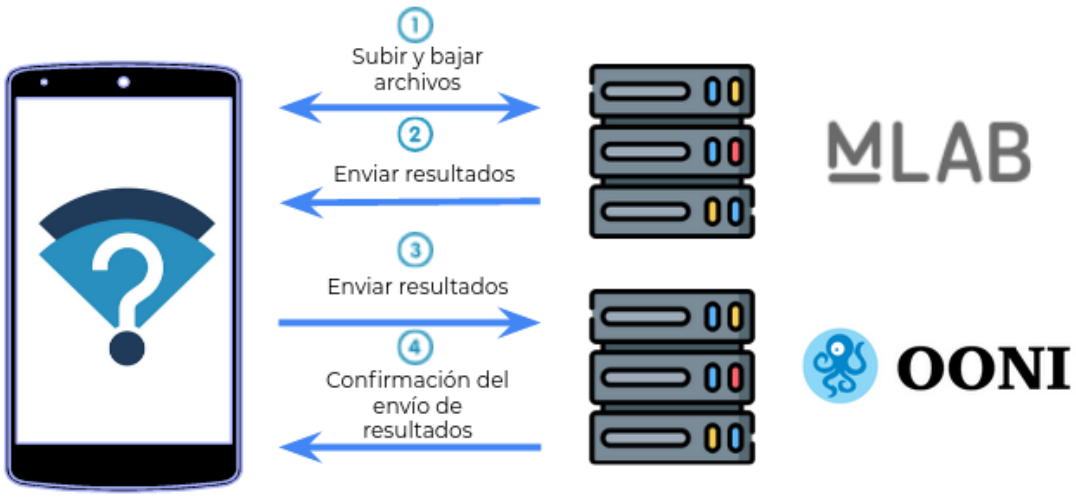

Figura 3.5: Para realizar esta medición se utiliza un servidor de M-Lab, desde el cual se descargan y suben archivos para obtener la velocidad de subida y bajada de la red. Una vez terminada esta medición, M-Lab envía los resultados a la aplicación, la que luego los envía OONI.

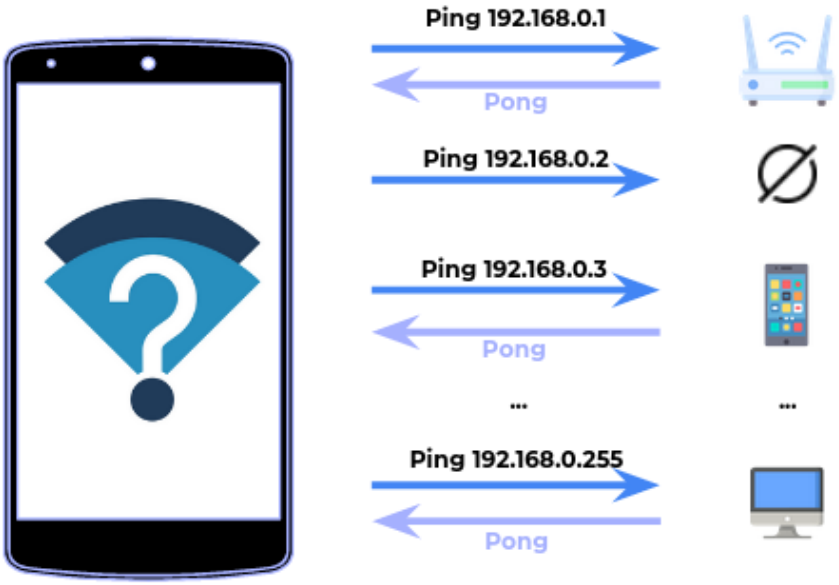

Figura 3.6: Para buscar los dispositivos que están en la red se calculan todas las posibles IPs privadas que puedan existir, luego se intenta hacer ping a cada una por un máximo de 1 segundo. Si se responde al ping, se guarda el dispositivo que tiene esa IP asignada como alcanzable. En el caso contrario, se asume que ese dispositivo no existe o no es alcanzable, por lo cual no se guarda.

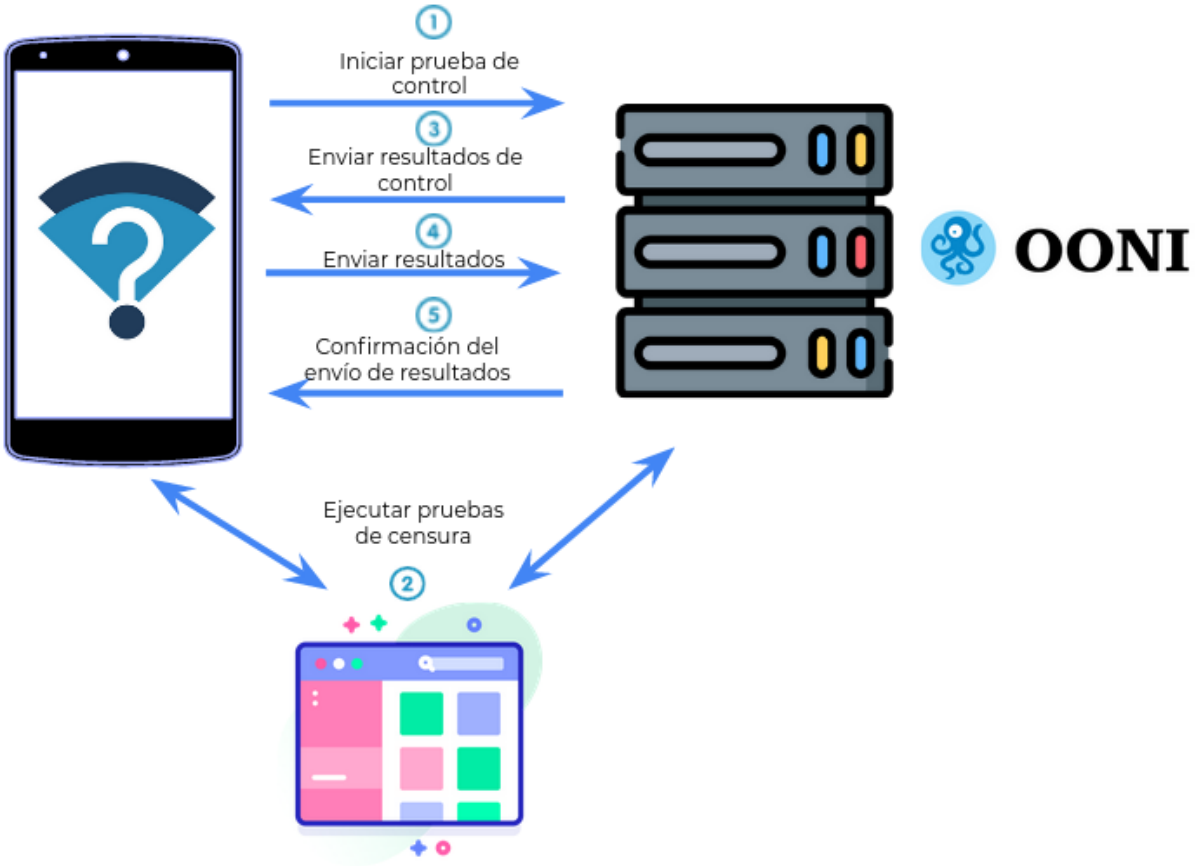

Figura 3.7: Para detectar el bloqueo de un sitio web, se ejecutan tanto en la aplicación como en un servidor de control una serie de pruebas que permiten determinar si hay problemas para conectarse al sitio, luego se comparan los resultados y se envían al servidor de OONI.

tales como la IP pública de la medición, el ASN, la identificación de dispositivo desde el que se realizó la medición(MAC, marca), lugar desde el que se hizo la medición y el nombre del usuario que realizó el test.

Las tablas `dns_tests`, `protocol_test`, `devices_test`, `ndt_tests_ooni` y `web_test_ooni` guardan la información referente a las mediciones de DNS, los protocolo de seguridad inalámbrica, búsqueda de dispositivos, medición de la velocidad y la detección de bloqueo de sitios web respectivamente. Estas están asociadas por medio de una *foreign key* `test_id` a la tabla central tests.

3.2.2. Diseño de los Endpoints HTTP

La aplicación posee los siguientes *endpoints* (ver figura 3.9).

- GET `/asn`: En caso de éxito responde 200 y entrega la IP pública desde la cual se está realizando la *request*, junto con el ASN correspondiente. Por otro lado, si falla se responde con 404.
- GET `/tests/<type_text>`: Recibe un parámetro correspondiente al tipo de test del cual se requiere información. En caso de éxito devuelve los 5 últimos resultados asociados al tipo de medición enviado y responde con código 200. En caso de error responde con 404 y si el formato ingresado es inválido se retorna 422, junto con la explicación de qué parámetros fueron mal ingresados.
- POST `/test/protocol`: Responde 201 si se crea se crea una nueva entrada en la tabla `protocol_test`. En el caso de ocurrir un error responde 404 y si se ingresa información en un formato invalido devuelve 422, junto con la explicación de qué parámetros fueron mal ingresados.
- POST `/tests/devices`: Responde 201 si se crea se crea una nueva entrada en la tabla `devices_test`. En el caso de ocurrir un error responde 404 y si se ingresa información en un formato invalido devuelve 422, junto con la explicación de qué parámetros fueron mal ingresados.
- POST `/tests/dns`: Responde 201 si se crea se crea una nueva entrada en la tabla `dns_tests`. En el caso de ocurrir un error responde 404 y si se ingresa información en un formato invalido devuelve 422, junto con la explicación de qué parámetros fueron mal ingresados.
- POST `/tests/ooni/ndt`: Responde 201 si se crea se crea una nueva entrada en la tabla `ndt_tests_ooni`. En el caso de ocurrir un error responde 404 y si se ingresa información en un formato invalido devuelve 422, junto con la explicación de qué parámetros fueron mal ingresados.
- POST `/tests/ooni/web`: Responde 201 si se crea se crea una nueva entrada en la tabla `web_test_ooni`. En el caso de ocurrir un error responde 404 y si se ingresa información en un formato invalido devuelve 422, junto con la explicación de qué parámetros fueron mal ingresados.

Además, cabe señalar que para la verificación del formato de entrada se utilizó la librería *Pydantic*[3] por medio de la que se definieron las reglas que deben seguir los parámetros de

[3]https://pydantic-docs.helpmanual.io/

entrada que ingresará el usuario de manera tal que si lo enviado es inválido se entregue un código de error 422 junto con un mensaje explicativo.

3.3. Implementación Frontend

3.3.1. Diseño del Modelo de Clases

La aplicación contiene los paquetes descritos a continuación (ver figura 3.10).

- User Test: Agrupa al código que se ejecuta para realizar las distintas mediciones.
- Activities: Corresponden a las clases que extienden *Activity*. Estas representan a cada pantalla mostrada al usuario [4].
- Data: Posee a la clase que se encarga de crear el cliente que realizará las conexiones con el servidor del CLCERT.
- Main Tabs: Agrupa a las clases que representan lo que está contenido dentro de las pestañas: *Inicio*, *Resultados* y *Configuración*. Estas extienden la clase *Fragment* [5] la cual permite definir que se mostrará en una sección particular de la pantalla.
- Adapter Recycler Views: Incluye los adaptadores que se implementaron con el fin de utilizar *Recycler Views* al momento de mostrar los resultados de la aplicación. Esto permite utilizar la memoria del teléfono de una manera eficiente cuando se muestra una lista en la aplicación [6].
- Moshi: Posee las clases que permiten utilizar la librería *Moshi* para *parsear* los JSON que se reciben y envían.

3.3.2. Integración de Librerías Externas

Se utilizaron las siguientes librerías externas en la aplicación.

- OkHttp [7]: Permite realizar solicitudes HTTP de manera sencilla. Fue utilizada para comunicarse con el servidor del CLCERT.
- MiniDNS [8]: Librería cuyo fin es facilitar a realización de consultas DNS desde Android y Java. Se usó para hacer las mediciones de seguridad en el resolver DNS.
- Moshi [9]: Permite parsear JSONS en objectos de Java y viceversa. Se utilizó para recibir y enviar JSONS.
- OONI [10]: Permite realizar las pruebas de OONI. Se utilizó en las pruebas de detección de bloqueo de sitios web y medición de la de velocidad.

3.3.3. Diseño de la Interfaz de Usuario

A continuación se mostrarán algunas de las interfaces implementadas.

[4] `https://developer.android.com/guide/components/activities/intro-activities`
[5] `https://developer.android.com/guide/fragments`
[6] `https://developer.android.com/jetpack/androidx/releases/recyclerview`
[7] `https://square.github.io/okhttp/`
[8] `https://github.com/MiniDNS/minidns`
[9] `https://github.com/square/moshi`
[10] `https://github.com/ooni/probe-engine`

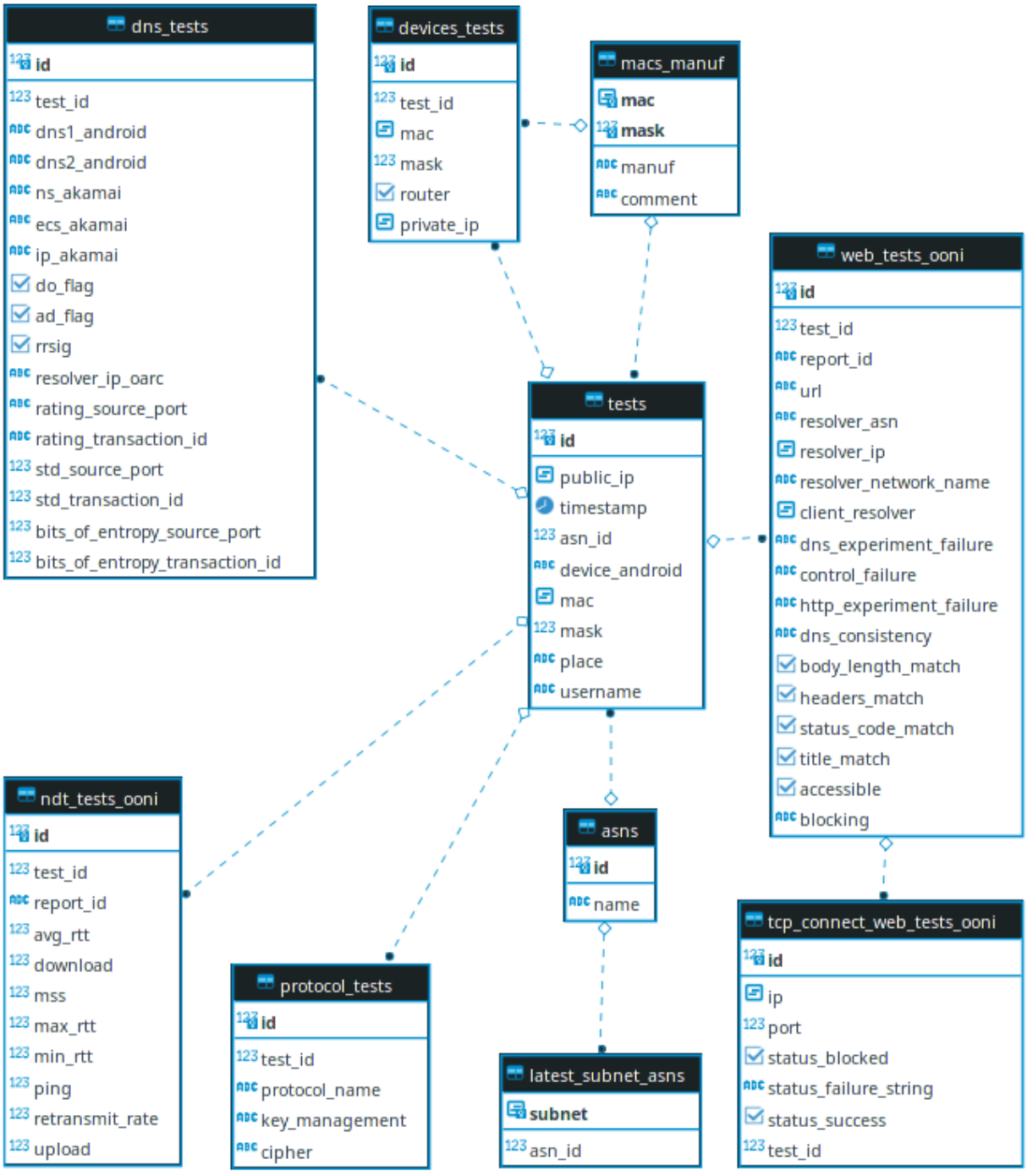

Figura 3.8: Modelo Entidad Relación utilizado en el *backend* de la aplicación desarrollada.

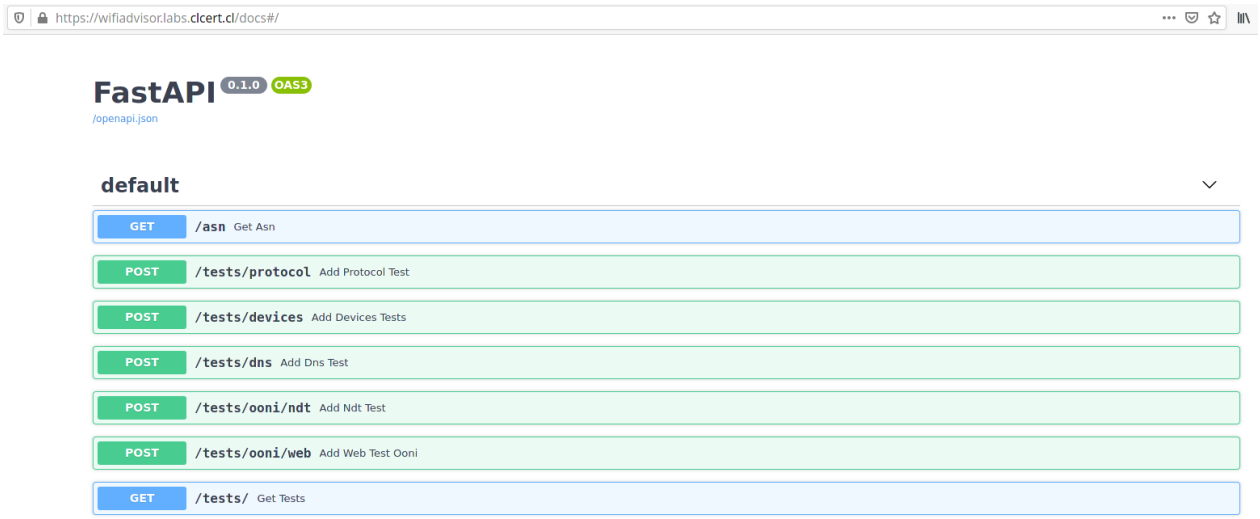

Figura 3.9: Documentación de los *endpoints* de la aplicación generada a partir de FastApi.

Pantalla de Inicio

Desde esta pantalla se pueden ejecutar las mediciones por parte del usuario (ver figura 3.11), para ello debe hacer click en la prueba que desea iniciar lo que llevará a una nueva pantalla donde se mostrará el progreso de la medición. Por ejemplo si el usuario presiona el botón que tiene por título "Otros dispositivos", se mostrará una vista como la de la figura 3.12. Para el caso de las otras mediciones la vista que se muestra es similar, sólo se cambian los mensajes. Una vez terminado lo anterior, se redigirá al usuario a la Pantalla de Resultados.

Pantalla de Resultados

La pantalla de resultados posee 4 pestañas en su barra superior, cada una tras ser presionada muestra los resultados de las mediciones siguiendo este orden: "Medición Básica" (figura 3.13), "Medición de la Velocidad" (figura 3.14), "Búsqueda de Otros Dispositivos" (figura 3.15) y 'Búsqueda de Otros Dispositivos" (figura 3.16).

Las dos primeras pantallas, muestran un icono que señala si el resultado fue positivo (ticket verde) o negativo (equis roja) junto a un texto que indica el resultado obtenido.

En la tercera pantalla se muestra el icono de un router para la IP privada que corresponde al *gateway*. Para el resto se muestra el icono de una casa. Además, se indica la IP privada de los dispositivos, seguida por el fabricante en caso de exista y los 3 primeros bytes de la dirección MAC del dispositivo censurando con ceros el resto.

Finalmente, en la cuarta pantalla se incluye un icono de advertencia amarillo si la página ingresada no existe. Para los casos de un resultado positivo y negativo se usan los mismos iconos que en la primera y segunda pantalla. Además, en caso de que se detecte un indicio de bloqueo se muestra el motivo de esta sospecha.

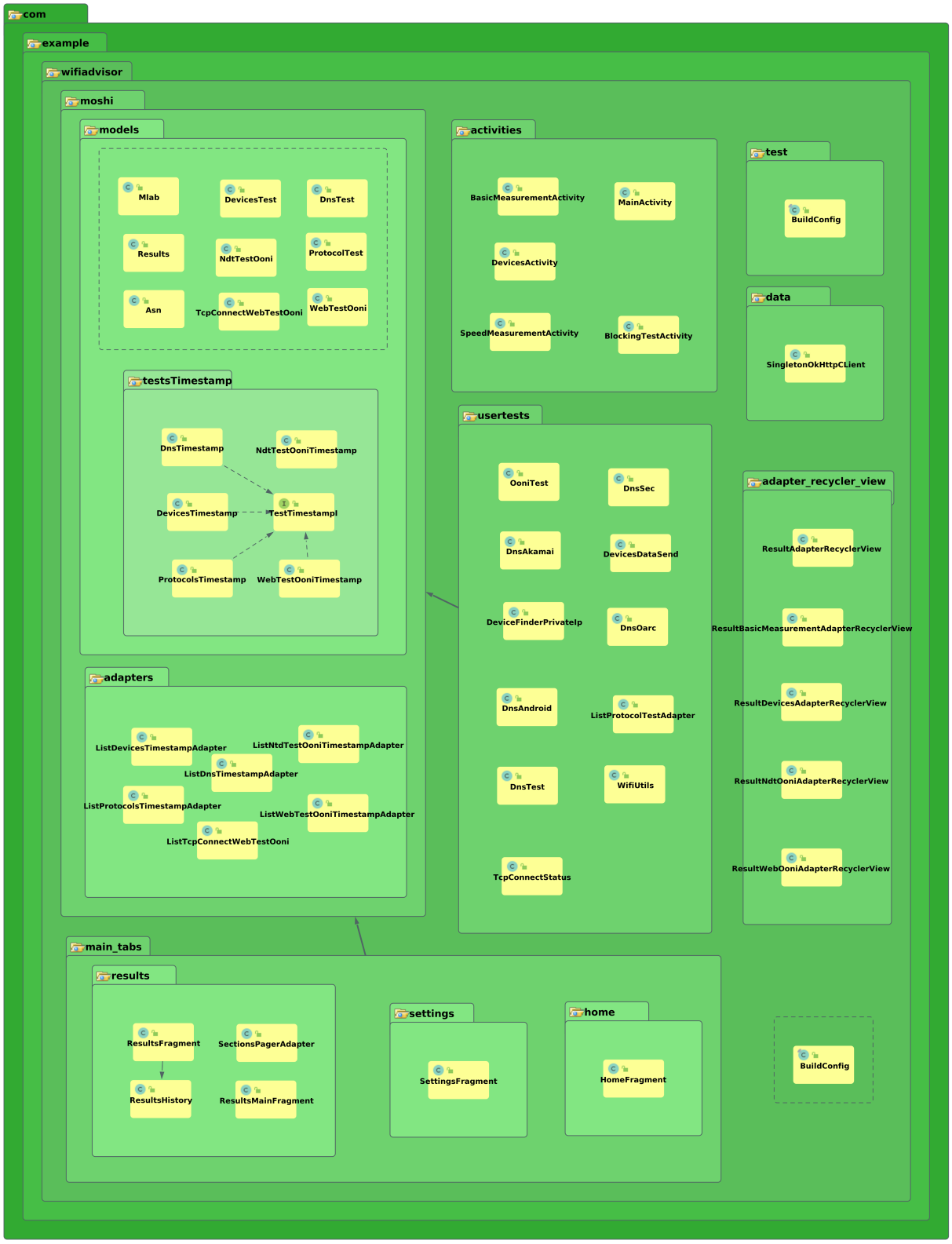

Figura 3.10: Modelo de Clases de la aplicación.

Figura 3.11: Pantalla de Inicio de la Aplicación.

Figura 3.12: Pantalla mostrada mientras se ejecuta la Medición de Búsqueda de Dispositivos.

Figura 3.13: Pantalla de Resultados de la Medición Básica de la Aplicación.

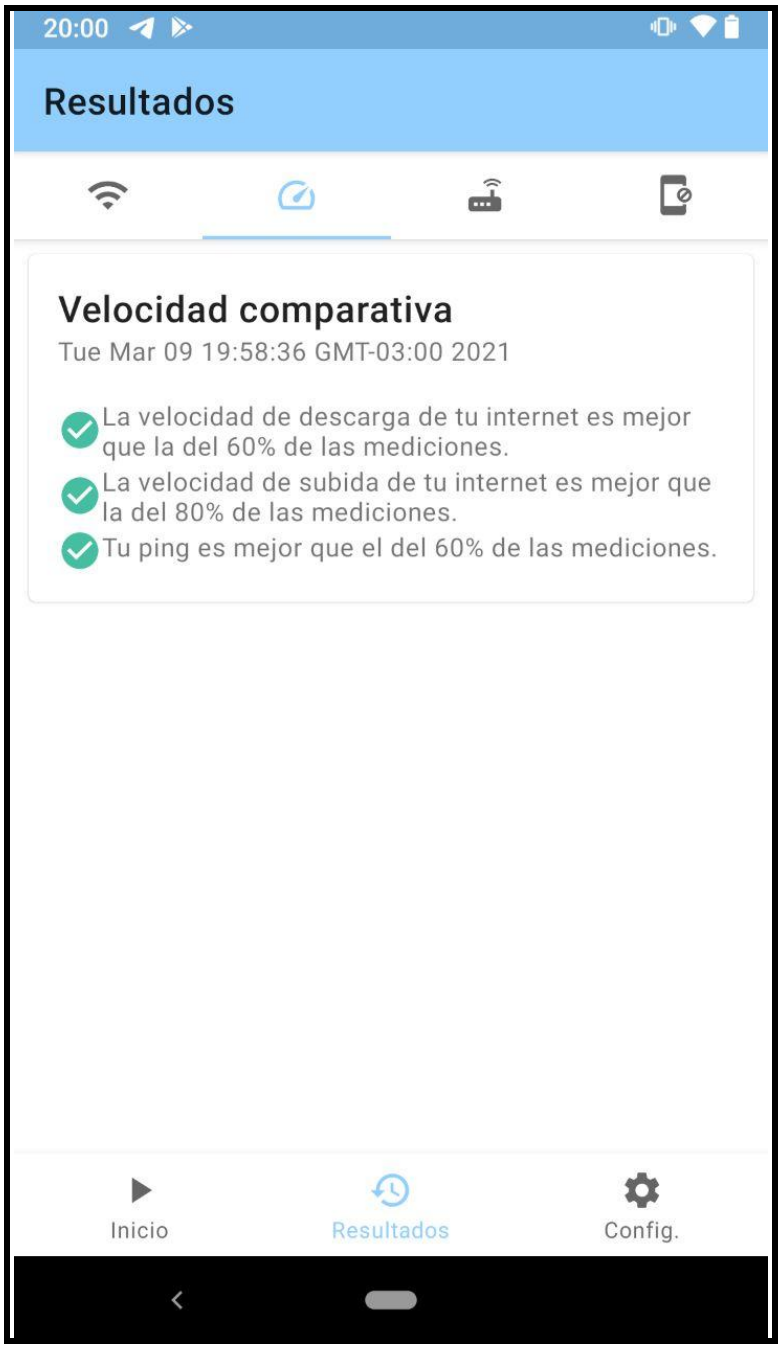

Figura 3.14: Pantalla de Resultados de la Medición de Velocidad de la Red de la Aplicación.

Figura 3.15: Pantalla de Resultados de la Medición de Búsqueda de los Dispositivos conectados en la Red de la Aplicación.

Figura 3.16: Pantalla de Resultados de la Medición de Detección de Bloqueo de Sitios Web de la Aplicación. El supuesto bloqueo del sitio `https://emol.com/` ocurre debido a que OONI no puede conectarse a las IPs dadas por la consulta DNS. Cabe señalar que no existen problemas de conexión para los casos de `http://emol.com/`, `https://www.emol.com/` y `http://www.emol.com/`.

Capítulo 4

Validación de la Solución Implementada

4.1. Para el usuario

Para validar la aplicación se realizó una prueba cerrada en la que se inscribieron 26 personas. De ellas, sólo 11 contestaron la encuesta que se les envió. Sin embargo, en el servidor hay datos correspondientes a 32 IPs públicas diferentes, por lo que se cree que algunas de las personas que no contestaron la encuesta de todos modos probaron la aplicación.

A continuación se mostrará los resultados de la encuesta realizada junto con algunos comentarios hechos por los encuestados. Cabe señalar que estos datos no son representativos debido al bajo números de respuestas, pero de todos modos constituyen un primer paso en la evaluación de esta aplicación.

De estos resultados se puede observar que quienes respondieron la encuesta son personas jóvenes, en su mayoría con un nivel avanzado de conocimientos de computación y con un nivel alto de estudios académicos, siendo educación superior incompleta el más bajo reportado (ver figuras 4.1, 4.2, 4.3).

4.1.1. Uso de la Aplicación

A continuación se muestran los resultados obtenidos para las preguntas referentes a las mediciones realizadas en la aplicación. Se les pidió a las personas calificar de uno a cinco puntos (donde uno es el peor puntaje y cinco el más alto) tres aspectos de cada prueba: utilidad de los resultados, tiempo de ejecución y fácilidad para entender el resultado (ver figuras 4.4, 4.5, 4.6).

De los resultados se puede observar que los tiempos de espera mientras se ejecutaba la aplicación parecieron aceptables por parte de los usuarios. Respecto a la utilidad y la facilidad para entender los resultados de las mediciones, la que obtuvo mayor puntaje fue la medición de velocidad.

Respecto a los comentarios dados en las respuestas se indicó como retroalimentación que faltaba incluir el valor numérico obtenido para la velocidad de subida, bajada y ping.

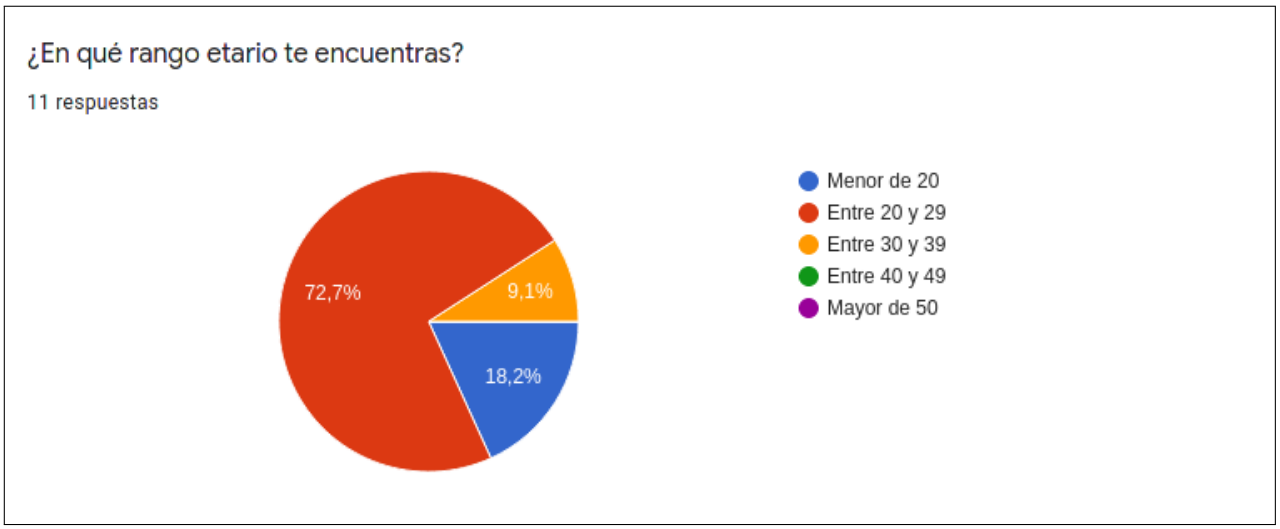

Figura 4.1: Respuestas a la pregunta ¿en qué rango etario te encuentras?

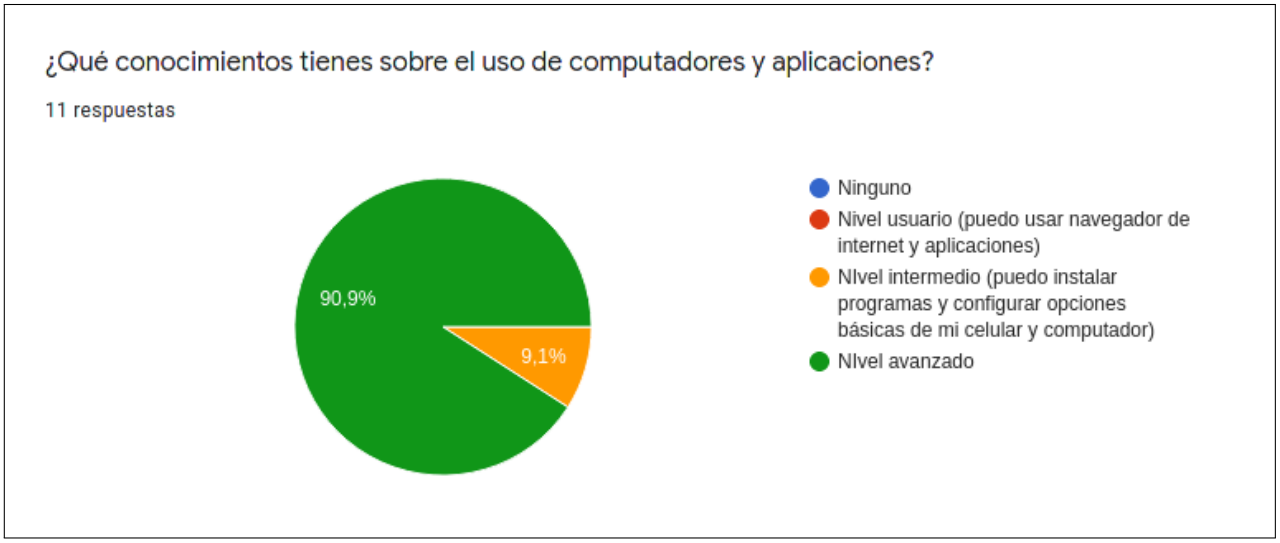

Figura 4.2: Respuestas a la pregunta ¿qué conocimientos tienes sobre el uso de computadores y aplicaciones?

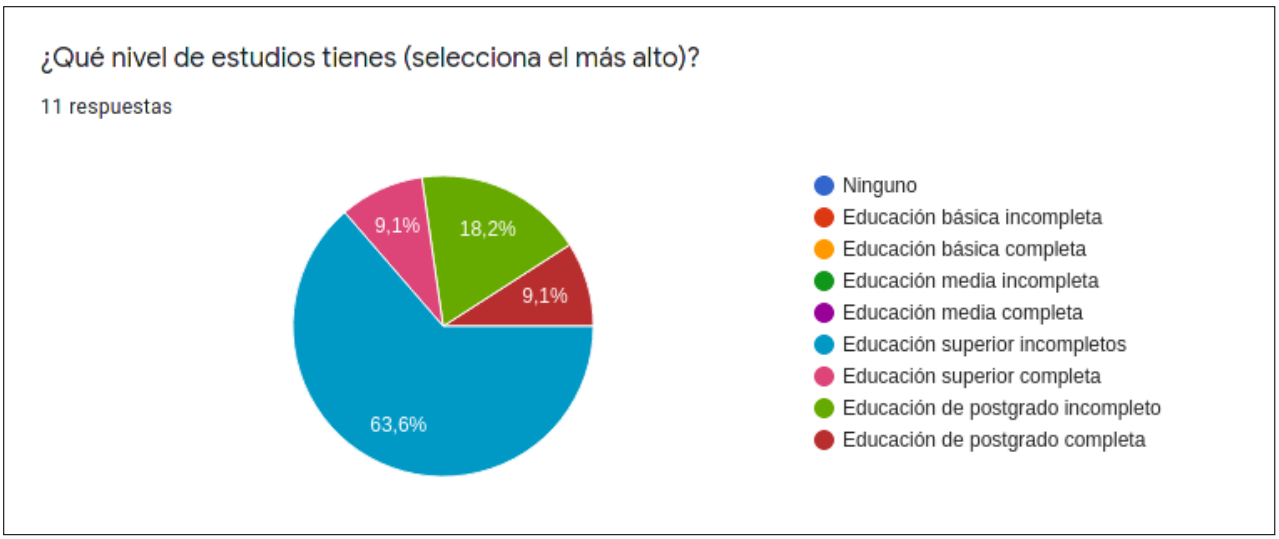

Figura 4.3: Respuestas a la pregunta ¿qué nivel de estudios tienes (selecciona el más alto)?

Dentro de los comentario referentes a la medición básica, alguien señaló que no sabía que significaba DNSSEC. Otra persona dijo que si bien entendía lo que significaba DNS debido a que había tomado un curso donde se explicaba este concepto, sugería incluir una explicación de este concepto para quienes no lo supieran.

Además, hubieron varios comentarios que señalaron que tuvieron problemas ejecutando la medición de búsqueda de dispositivos, por lo que es necesario revisar esta implementación en otras redes y en más celulares.

Sobre la detección de bloqueo de sitios web 2 personas comentaron que no sospechaban de ningún sitio por lo que calificaron la utilidad de esta medición con un 3. Una mejora que surge a partir de esto es tener más adelante una lista sugerida de páginas en caso de que el usuario no sospeche de ninguna.

4.1.2. Información sobre los encuestados

4.2. Para el CLCERT

En el CLCERT, se comentó que la aplicación tenía mucho potencial para crecer e incluir nuevas funcionalidades. Además, se conversó con NICLabs sobre la posibilidad de integrar algunas mediciones en Pepa Ping.

Respecto a las mejoras se señaló que era necesario incluir explicaciones más largas de los resultados y se propuso un nuevo flujo para que un usuario vea sus resultados. Ambas mejoras quedarán propuestas como trabajo futuro.

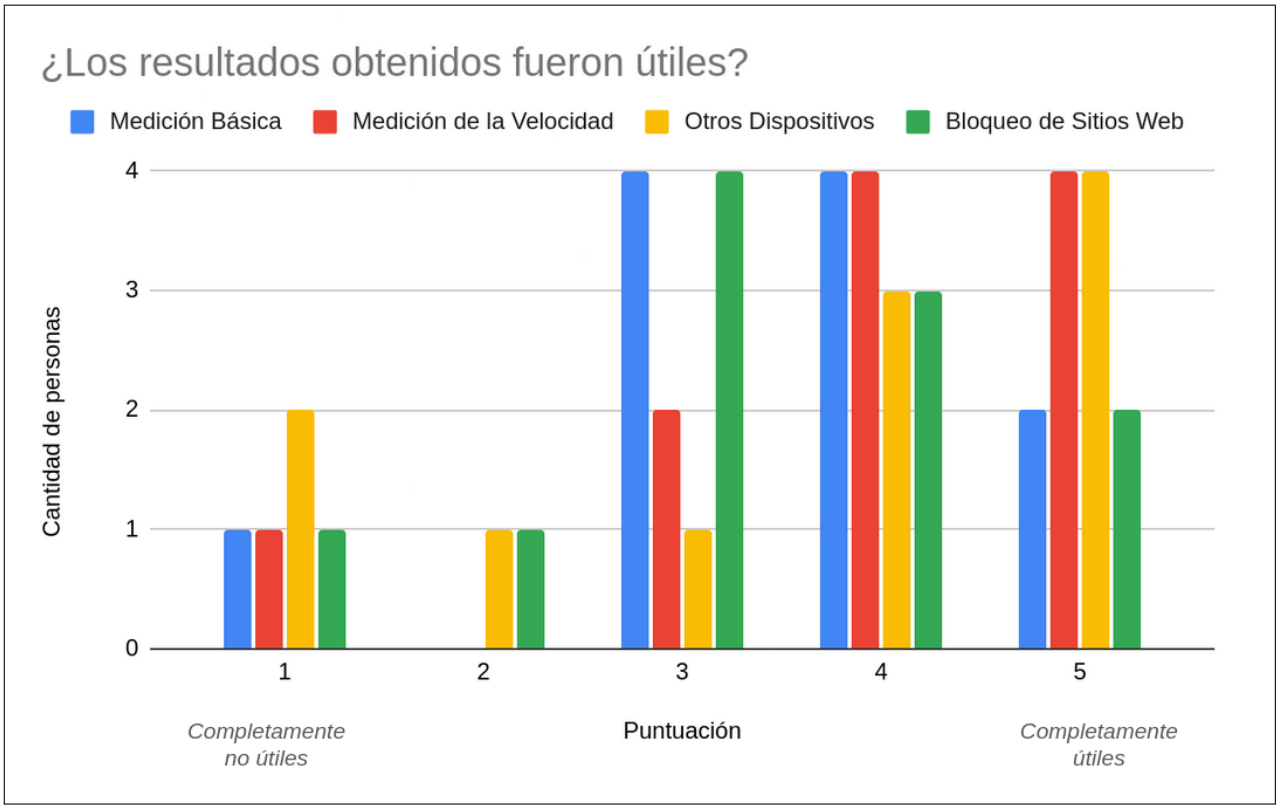

Figura 4.4: Respuestas a la pregunta ¿los resultados obtenidos fueron útiles? Donde 1 es completamente no útiles y 5 es completamente útiles.

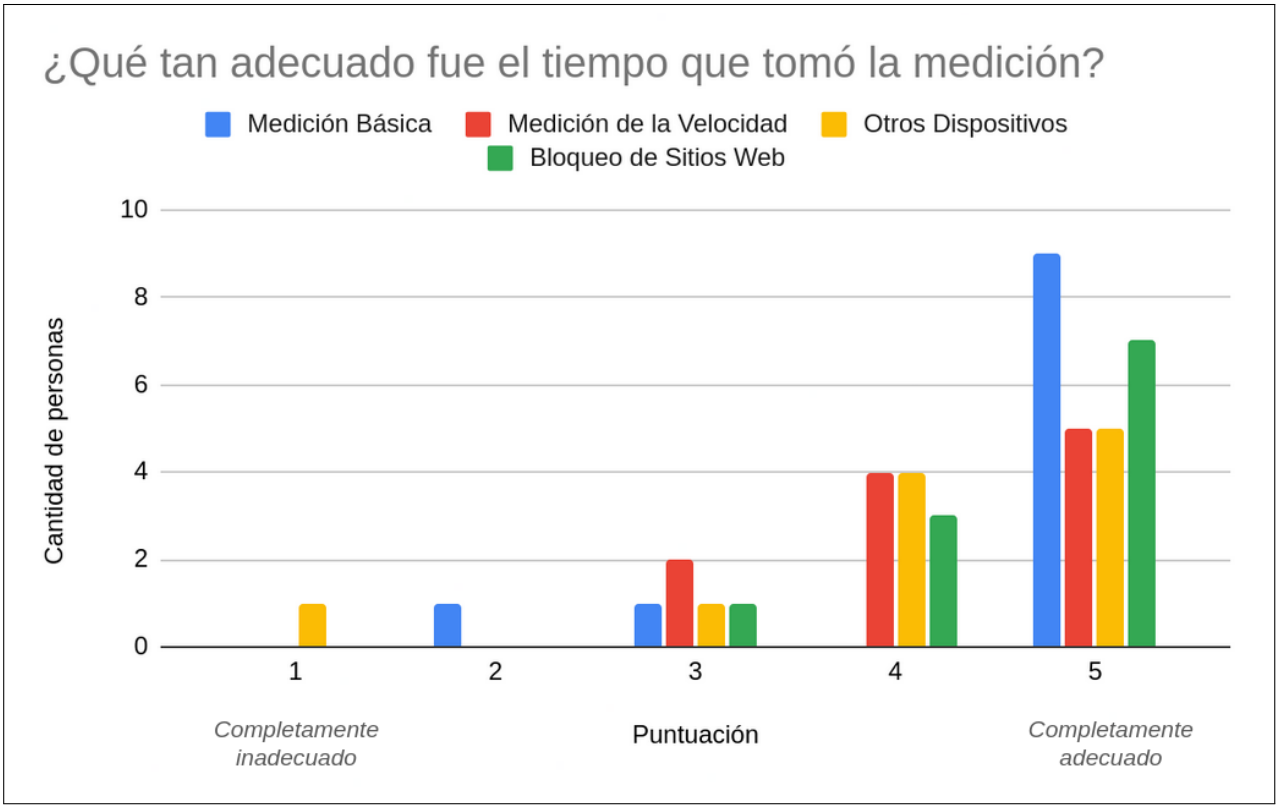

Figura 4.5: Respuestas a la pregunta ¿qué tan adecuado fue el tiempo que tomó la medición? Donde 1 es completamente inadecuado y 5 es completamente adecuado.

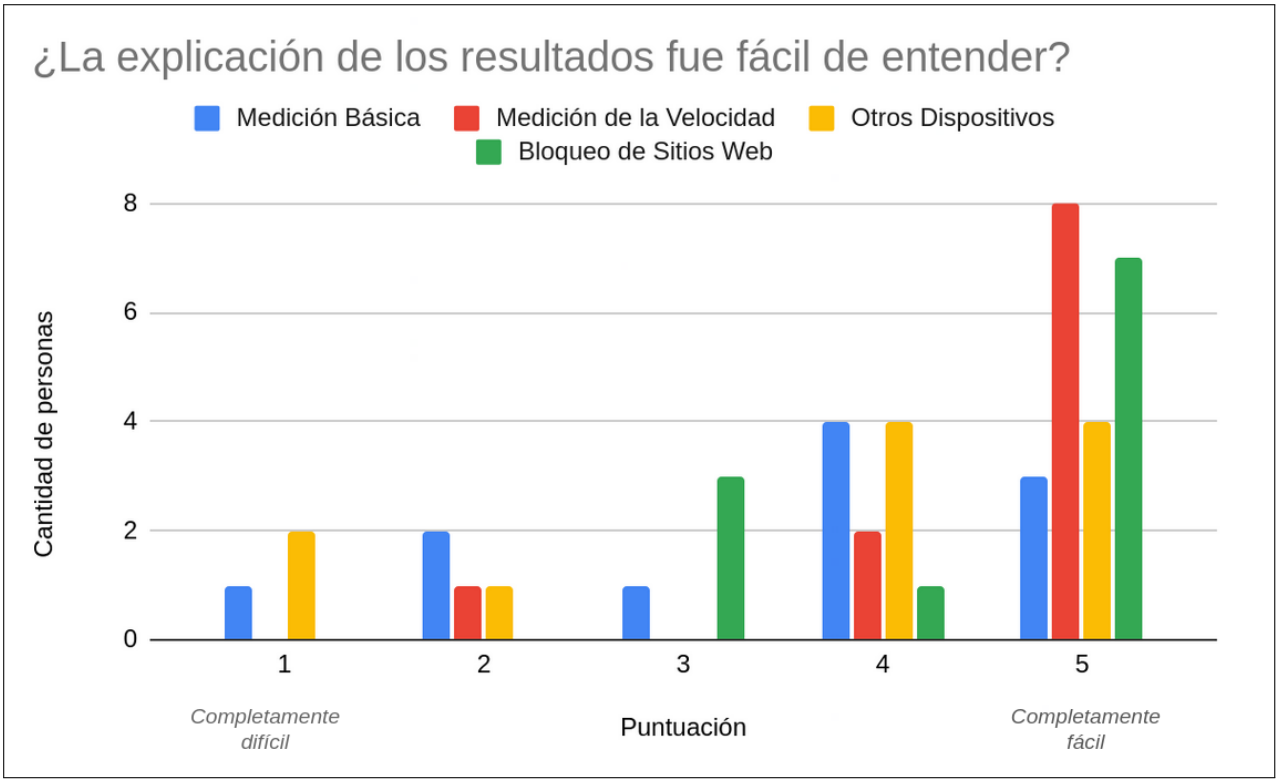

Figura 4.6: Respuestas a la pregunta ¿la explicación de los resultados fue fácil de entender? Donde 1 es completamente difícil y 5 es completamente fácil.

Conclusión

En este trabajo se muestra el desarrollo de una aplicación móvil que permite ejecutar mediciones de seguridad y privacidad de la red en la que se utiliza. Además, los resultados de estas pruebas son enviados al CLCERT para servir cómo insumo de futuras investigaciones respecto a las redes inalámbricas locales en Chile.

En relación a los objetivos planteados al inicio de este trabajo se evalúa que estos fueron alcanzados. Sin embargo, la validación no contó con tantos voluntarios dispuestos a responder la encuesta, de manera que los resultados obtenidos no fueron representativos. Además, debido a la pandemia del COVID-19 no fue posible probar la aplicación en puntos de acceso WiFi públicos.

Respecto a la relevancia de esta aplicación, esta es un primer paso en el desarrollo que conlleva un proyecto de esta envergadura, permitiendo aterrizar algunas ideas que se tenía respecto a que es lo que se búscaba medir y cómo realizarlo. Además, los datos obtenidos si bien no son representativos permitirán evaluar qué tan útiles son estás medidas y cómo se pueden mejorar.

Sobre los aprendizajes, se encuentran los conocimientos adquiridos respecto a seguridad en redes inalámbricas. Sumado a que se obtuvo experiencia en integración con otros servicios y desarrollo en Android, FastApi y SqlAlchemy. También, se mejoró en la planificación de tareas y se ganó experiencia en la toma de decisiones de diseño y levantamiento de requisitos.

Trabajo Futuro

Se propone incorporar las siguientes mediciones a la aplicación.

- Incluir una medición para comprobar de si un router sigue teniendo configurada su clave por defecto.
- Añadir una revisión de si alguno de los puertos están abiertos en los dispositivos que se encuentren en la red.
- Hacer *fingerprinting* del router.
- Incluir un listado de URLs para la prueba de detección de sitios bloqueados.
- Integrar más pruebas de *OONI Probe*.

Además, se propone mejorar el flujo que actualmente entrega los resultados al usuario, de manera que estos sean compresibles por personas sin conocimientos en redes y seguridad

de redes inálambricas WiFi. Sumado a esto, se sugiere la creación de un sitio donde se pueda acceder a mayor información sobre las mediciones. Aquí se podría también presentar estadísticas y un ranking con los usuarios que más colaboran. Finalmente, se plantea la necesidad de más adelante implementar un control del números de pruebas diarias que puede realizar cada IP para prevenir ataques de denegación de servicio.

Bibliografía

[1] Wi-Fi Alliance. Technical note removal of tkip from wi-fi devices. https://www.wi-fi.org/download.php?file=/sites/default/files/private/Wi-Fi_Alliance_Technical_Note_TKIP_v1.0.pdf, 2015.

[2] Wi-Fi Alliance. Wi-fi alliance introduces wi-fi certified wpa3 security. https://www.wi-fi.org/news-events/newsroom/wi-fi-alliance-introduces-wi-fi-certified-wpa3-security, 2018.

[3] Javier Bustos-Jiménez, Gabriel Del Canto, Sebastián Pereira, Felipe Lalanne, José M. Piquer, Gabriel Hourton, Alfredo Cádiz, and Victor Ramiro. How adkintunmobile measured the world. In Friedemann Mattern, Silvia Santini, John F. Canny, Marc Langheinrich, and Jun Rekimoto, editors, *The 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp '13, Zurich, Switzerland, September 8-12, 2013 - Adjunct Publication*, pages 1457–1462. ACM, 2013.

[4] Cloudflare. cloudflare. https://www.cloudflare.com/es-es/learning/dns/dns-cache-poisoning/.

[5] Laboratorio de Criptografía Aplicada y Ciberseguridad (CLCERT). El laboratorio. https://www.clcert.cl/laboratorio/, 2020.

[6] Subsecretaría de Telecomunicaciones. Informe anual del sector telecomunicaciones 2019. https://www.subtel.gob.cl/estudios-y-estadisticas/informes-sectoriales-anuales/, Marzo 2020.

[7] Analysis DNS Operations and Research Center (DNS-OARC). Txidtest. https://www.dns-oarc.net/oarc/services/txidtest.

[8] Analysis DNS Operations and Research Center (DNS-OARC). Porttest. https://www.dns-oarc.net/oarc/services/porttest, 2008.

[9] Analysis DNS Operations and Research Center (DNS-OARC). Dns-oarc briefing. https://www.dns-oarc.net/files/web-brochure.pdf, 2020.

[10] Fastapi. Benchmarks. https://fastapi.tiangolo.com/benchmarks/.

[11] The Organisation for Economic Co-operation and Development(OECD). Keeping the internet up and running in times of crisis. https:

//read.oecd-ilibrary.org/view/?ref=130_130768-5vgoglwswy&title=Keeping-the-Internet-up-and-running-in_times-of-crisis, 2020.

[12] ICANN. Dnssec, what is it and why is it important? https://www.icann.org/resources/pages/dnssec-what-is-it-why-important-2019-03-05-en, 2019.

[13] International Computer Science Institute (ICSI). Netalyzr releases app for android smart phones. https://www.icsi.berkeley.edu/icsi/blog/netalyzr-releases-android-app, 10 2013.

[14] Sericon Technology Inc. The real state of wifisecurityin the connected home. https://routercheck.com/WhitePapers/The_Real_State_of_WiFi_Security_in_the_Connected_Home.pdf, 2015.

[15] Christian Kreibich. Shutting down netalyzr. http://mailman.icsi.berkeley.edu/pipermail/netalyzr/2019-February/000166.html, 02 2019.

[16] Measurement Lab(M-Lab). Measurement lab. https://www.measurementlab.net/.

[17] NIC Chile Research Labs. Adkintun mobile. https://play.google.com/store/apps/details?id=cl.niclabs.adkintunmobile.

[18] NIC Chile Research Labs. Pepa ping: Medición de la calidad de internet. https://play.google.com/store/apps/details?id=cl.niclabs.vpnpassiveping&hl=en_US.

[19] Keyu Lu and Zhaoxin Zhang. Evaluating "Health Status" for DNS Resolvers. *IEICE Transactions on Communications*, 101(12):2409–2424, December 2018.

[20] M-Lab. Why are my m-lab results different from other speed tests? https://support.measurementlab.net/help/en-us/6-measurement-services/11-why-are-my-m-lab-results-different-from-other-speed-tests.

[21] Diego Madariaga and Gabriela Mendoza. Pepa ping: Android tool to take and predict periodic passive ping measurements. In Sandra Céspedes and Javier Bustos-Jiménez, editors, *Proceedings of the IV School on Systems and Networks, SSN 2018, Valdivia, Chile, October 29-31, 2018*, volume 2178 of *CEUR Workshop Proceedings*, pages 9–12. CEUR-WS.org, 2018.

[22] NewScientist. Understanding your netalyzr results. https://www.newscientist.com/article/dn18953-understanding-your-netalyzr-results/?ignored=irrelevant, 2010.

[23] Open Observatory of Network Interference (OONI). About. https://ooni.org/about/.

[24] Open Observatory of Network Interference (OONI). Ndt speed test. https://ooni.org/nettest/ndt/.

[25] Open Observatory of Network Interference (OONI). Open observatory. https://play.google.com/store/apps/details?id=org.openobservatory.ooniprobe&hl=es.

[26] Open Observatory of Network Interference (OONI). Web connectivity test specification. https://github.com/ooni/spec/blob/master/nettests/ts-017-web-connectivity.md, 2020.

[27] Open Observatory of Network Interference(OONI). Open observatory of network interference. https://ooni.org/.

[28] CERT Division of the Software Engineering Institute at Carnegie Mellon University. Multiple dns implementations vulnerable to cache poisoning. https://www.kb.cert.org/vuls/id/800113, 2008.

[29] Georgia Bullen Peter Boothe. How fast is my internet? speed tests, accuracy, ndt and m-lab. https://www.measurementlab.net/blog/speed-tests-accuracy/#how-fast-is-my-internet?-speed-tests,-accuracy,-ndt-&-m-lab, 2019.

[30] B. Reddy and V. Srikanth. Review on wireless security protocols (wep, wpa, wpa2 & wpa3). *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, pages 28–35, 07 2019.

[31] E Riveros Roca. Nuevas estrategias de análisis de datos de escaneos de la red chilena para el monitoreo periódico de su seguridad. http://repositorio.uchile.cl/handle/2250/175511, 2020.

[32] WiGLE. Frequently asked questions. https://wigle.net/faq.

[33] WiGLE. Statistics. https://wigle.net/stats, 3 2021.