

Universidad Cenfotec

Maestría en Ciberseguridad

Documento final del Proyecto de Investigación Aplicada 2

Análisis de seguridad a la aplicación móvil Sistema de Captura de Datos de la Encuesta Nacional Agropecuaria, del Instituto Nacional de Estadística y Censos, tomando como referencia el estándar de Verificación de Seguridad de Aplicaciones Móviles Versión 1.2

Méndez Castillo Emmanuel

Julio 2021

Declaratoria de Derecho de Autor

Este documento es propiedad del autor, en él se presenta una investigación en el área de la seguridad de aplicaciones móviles para la Maestría en Ciberseguridad de la Universidad Cenfotec, se prohíbe la consulta y reproducción de este por el tiempo de 5 años iniciando después de la presentación formal del trabajo final; luego de cumplido ese periodo, se autoriza la reproducción total o parcial de este documento para fines académicos exclusivamente.

Dicha autorización es válida solamente si cuenta con el permiso escrito del autor: Emmanuel Méndez Castillo (emendezc@ucenfotec.ac.cr).

Dedicatoria

Le dedico este proyecto a Diana, una gran mujer y amada esposa, quien me ha apoyado y motivado a seguir adelante.

Agradecimientos

Agradezco a Dios, quien me ha permitido lograr esta meta con constancia y amor.

Agradezco a la Unidad Técnica de Sistemas e Informática del Instituto Nacional de Estadística y Censos, por el apoyo y confianza recibida.

De igual manera, agradezco a todos los profesores(as) de la Maestría en Ciberseguridad de la Universidad Cenfotec, por su profesionalismo, pasión y dedicación; en especial a M.Sc. Dennis Alonso Durán Céspedes, tutor y profesor en la Maestría, quien me ha acompañado y brindado su apoyo en el desarrollo de este proyecto.

Por último, quiero agradecer a los lectores del proyecto, a M.Sc. Miguel Pérez Montero y M.Sc. Ignacio Trejos Zelaya, por su dedicación y apoyo en este proyecto.

Muchas gracias a todos.

universidad cenfotec_
tecnologías digitales

Universidad Cenfotec
Carrera de Postgrado
Maestría en Ciberseguridad

TRIBUNAL EXAMINADOR

Este proyecto fue aprobado por el Tribunal Examinador de la carrera: **Maestría en Ciberseguridad,** requisito para optar por el título de grado de **Maestría,** para el estudiante: **Méndez Castillo Emmanuel.**

DENNIS ALONSO DURAN CESPEDES (FIRMA)
Firmado digitalmente por DENNIS ALONSO DURAN CESPEDES (FIRMA)
Fecha: 2021.07.19 14:09:02 -06'00'

M. Sc. Dennis Durán Céspedes
Tutor

MIGUEL PEREZ MONTERO (FIRMA)

M. Sc. Miguel Pérez Montero
Lector 1

IGNACIO TREJOS ZELAYA (FIRMA)
Firmado digitalmente por IGNACIO TREJOS ZELAYA (FIRMA)
Fecha: 2021.07.21 10:45:44 -06'00'

M. Sc. Ignacio Trejos Zelaya
Lector 2

San José, Costa Rica, 16 de julio de 2021

Firmada digitalmente, de conformidad con la Ley de Certificados, Firmas Digitales y Documentos Electrónicos N° 8454, destacando el artículo 9°-

Tabla de contenido

DECLARATORIA DE DERECHO DE AUTOR **3**
DEDICATORIA **4**
AGRADECIMIENTOS **5**
TABLA DE CONTENIDO **7**
RESUMEN **9**
CAPÍTULO 1. INTRODUCCIÓN **10**
1.1 GENERALIDADES 10
1.2 ANTECEDENTES DEL PROBLEMA 10
1.3 DEFINICIÓN Y DESCRIPCIÓN DEL PROBLEMA 10
1.4 JUSTIFICACIÓN 11
1.5 VIABILIDAD 12
1.5.1 Punto de vista técnico. *12*
1.5.2 Punto de vista operativo. *12*
1.5.3 Punto de vista económico. *13*
1.6 OBJETIVOS 13
1.6.1 Objetivo general. *14*
1.6.2 Objetivos específicos *14*
1.7 ALCANCES Y LIMITACIONES 14
1.7.1 Alcances *14*
1.7.2 Limitaciones *14*
1.8 MARCO DE REFERENCIA ORGANIZACIONAL Y SOCIOECONÓMICO 15
1.8.1 Historia *15*
1.8.2 Tipo de negocio y mercado meta. *16*
1.8.3 Misión, visión y valores. *17*
1.8.4 Políticas institucionales *18*
1.9 ESTADO DE LA CUESTIÓN 18
CAPÍTULO 2. MARCO TEÓRICO **30**
CAPÍTULO 3. MARCO METODOLÓGICO **39**
3.1 TIPO DE INVESTIGACIÓN 39
3.2 ALCANCE INVESTIGATIVO 39
3.3 ENFOQUE 39
3.4 DISEÑO 39
3.5 POBLACIÓN Y MUESTREO 39
3.6 INSTRUMENTOS DE RECOLECCIÓN DE DATOS 40
3.7 TÉCNICAS DE ANÁLISIS DE INFORMACIÓN 40
3.8 ESTRATEGIA DE DESARROLLO DE LA PROPUESTA 40
CAPÍTULO 4. PROPUESTA DE SOLUCIÓN **41**
CAPÍTULO 5. DESARROLLO DE LA SOLUCIÓN **45**
5.1 PREPARACIÓN 45
5.2 RECOLECCIÓN DE INFORMACIÓN 52
5.2.1 Documentación recolectada *52*
5.2.2 Información de la aplicación *54*

5.3 MAPEO DE APLICACIÓN 58
5.4 EVALUACIÓN 63
5.4.1 Ambiente de pruebas 64
5.4.2 Herramientas para las pruebas 67
5.4.3 Hallazgos 69
5.5 DOCUMENTACIÓN 75
5.5.1. Resumen ejecutivo 75
5.5.2. Resultados por cada categoría evaluada. 75
5.5.3. Resultados por niveles de seguridad de MASVS. 76

CAPÍTULO 6. CONCLUSIONES Y RECOMENDACIONES 80

6.1 CONCLUSIONES 80
6.2 RECOMENDACIONES 82

GLOSARIO 83

REFERENCIAS 87

ANEXOS 92

ANEXO 1. CARTA DE AUTORIZACIÓN. 93
ANEXO 2. LISTA DE CHEQUEO DE SEGURIDAD MÓVIL PARA ANDROID DE OWASP 94
ANEXO 3. PLANTILLA UTILIZADA PARA REPORTAR LOS HALLAZGOS. 101

Resumen

Este proyecto desarrolla el análisis de seguridad que define el nivel de confianza de la aplicación móvil Sistema de Captura de Datos de la Encuesta Nacional Agropecuaria del Instituto Nacional de Estadística y Censos. Para dicho análisis, se toma como referencia el Estándar de Verificación de Seguridad de Aplicaciones Móviles, el cual reúne un conjunto de requerimientos necesarios para diseñar, desarrollar y probar la seguridad de aplicaciones móviles. Además, dispone de una guía de pruebas que cubre procesos, técnicas, casos de pruebas y herramientas para verificar la seguridad de aplicaciones móviles.

Se utiliza la metodología de pruebas de seguridad propuesta por el Estándar de Verificación de Seguridad de Aplicaciones Móviles Versión 1.2 publicado por Proyecto Abierto de Seguridad para Aplicaciones Web. Dicha metodología está conformada por cinco fases: preparación, reconocimiento de información, mapeo de la aplicación, evaluación (explotación) y documentación. Con cada una de estas fases, se identifican vulnerabilidades y fallos de seguridad de la aplicación móvil evaluada. Al finalizar la evaluación de los requerimientos de seguridad definidos en el estándar, se lleva a cabo un análisis de los resultados obtenidos para determinar el nivel de confianza de seguridad del aplicativo y proporcionar recomendaciones de seguridad para el desarrollo y verificación de aplicaciones móviles seguras.

Palabras clave: MASVS, MSTG, OWASP, análisis, estándar, seguridad móvil, Android.

Capítulo 1. Introducción

1.1 Generalidades

Es importante considerar que este proyecto de investigación tiene fines académicos. Se publicará la información que esté autorizada, además, algunos datos fueron censurados empleando la ética profesional para no infringir cláusulas del contrato de confidencialidad estipulado en el Instituto Nacional de Estadística y Censos (INEC).

1.2 Antecedentes del problema

El Departamento de Desarrollo de Sistemas de Información del INEC desarrolla aplicaciones móviles para el sistema operativo Android, para diferentes clientes internos del Instituto, tal es el caso de la Encuesta Nacional Agropecuaria (ENA). El equipo de desarrollo de sistemas de información del Instituto cumple con el objetivo de entregar un producto de calidad en un tiempo estimado, sin embargo, desea mejorar en el tema de seguridad en las aplicaciones móviles que desarrollan internamente. No obstante, no existen antecedentes o documentación previa sobre análisis de seguridad de aplicaciones móviles desarrolladas en el Instituto, partiendo de este hecho, el actual proyecto pretende sentar las bases sobre el tema.

1.3 Definición y descripción del problema

El Departamento de Desarrollo de Sistemas de Información del INEC desarrolló la aplicación móvil Sistema de Captura de Datos (SICAD) para la encuesta ENA. Dicha aplicación es clave en la recolección de datos sobre la producción agrícola, forestal, pecuaria y de flores de los productores de Costa Rica. Por lo tanto, el equipo de desarrollo de sistemas de información del INEC desea determinar el nivel de seguridad de esta aplicación móvil desarrollada dentro de la institución; partiendo de este punto, este proyecto pretende analizar el nivel de confianza de seguridad de dicho aplicativo desarrollado para el sistema operativo Android.

1.4 Justificación

En la actualidad, las aplicaciones móviles representan la puerta para gestionar todo tipo de información, inclusive datos sensibles del usuario o empresa, por lo cual, los aplicativos se convierten en objetivos atrayentes para los atacantes cibernéticos. Esta situación promueve la búsqueda de maneras de brindar seguridad de la información durante el desarrollo de estas, por parte de los desarrolladores de aplicativos.

Con respecto al INEC, como institución encargada de la recolección de datos para generar estadísticas nacionales, tiene la Unidad Técnica de Sistemas e Información (UTSI), encargada del desarrollo de aplicaciones móviles para las diferentes operaciones estadísticas. Entre las operaciones en producción, se encuentra la encuesta ENA que utiliza un aplicativo móvil llamado SICAD, el cual es clave en la recolección de datos sobre la producción agrícola, forestal, pecuaria y de flores de los productores de Costa Rica.

Cabe señalar que el aplicativo SICAD se encuentra en producción desde el 2015, sin embargo, se planifica un rediseño en parte de la lógica de negocio de esta encuesta para el 2022; este rediseño implicará cambios en los sistemas informáticos actuales. La UTSI, preocupada por el tema de ciberseguridad en los desarrollos internos, proveyó lo concerniente al aplicativo para el análisis de seguridad de este. De esta forma, los resultados del proyecto serán aprovechados como valor agregado para el futuro rediseño.

Relacionado con la protección de la información, se deben definir requisitos de seguridad en las aplicaciones; estos pueden derivarse de estándares o mejores prácticas, como los definidos por la Asociación Nacional de Garantía de la Información (traducido de National Information Assurance Partnership, [NIAP]), el Proyecto Abierto

de Seguridad para Aplicaciones Web (traducido de Open Web Application Security Project, [OWASP]), la Corporación MITRE (traducido de MITRE Corporation, [MITRE]), e Instituto Nacional de Estándares y Tecnología (traducido de National Institute of Standards and Technology, [NIST]) (NIST, 2019).

Los estándares o mejores prácticas definidas por NIAP, OWASP, MITRE y NIST definen requisitos de seguridad para las aplicaciones móviles. Sin embargo, en la actualidad, el estándar de Verificación de Seguridad de Aplicaciones Móviles (traducido de Mobile Application Security Verification Estándar, [MASVS]) publicado por OWASP es quien define una línea base a seguir al describir el qué y cómo probar los aplicativos. Esto lo hace único (Antonishyn y Misnik, 2019).

En conclusión, la importancia del presente proyecto radica en que los resultados establecerán las bases del tema de seguridad de aplicativos móviles en el INEC como parte de un proceso de mejora continua, utilizando una herramienta como el estándar MASVS en el aplicativo móvil SICAD de la encuesta ENA.

1.5 Viabilidad

1.5.1 Punto de vista técnico.

El proyecto es técnicamente viable, dado que el investigador tiene los conocimientos y la experiencia en la temática de ciberseguridad y desarrollo de aplicaciones móviles para el sistema operativo Android.

1.5.2 Punto de vista operativo.

El desarrollo del proyecto es viable operativamente, puesto que no altera el funcionamiento del Departamento de Desarrollo de Sistemas de Información, solo se requiere un espacio físico para realizar la investigación, el cual será brindado por el investigador. Además, se cuenta con el visto bueno del Departamento de Desarrollo de Sistemas de Información del INEC para el acceso a documentación técnica, código

fuente e instalable del aplicativo evaluado. Para realizar el presente proyecto, se emitió una autorización por parte de la coordinación de la Unidad Técnica de Sistemas e Informática (UTSI) del INEC que se encuentra en el Anexo 1.

1.5.3 Punto de vista económico.

La realización del proyecto propuesto es viable económicamente, pues el costo de la investigación está representado en “horas consultor”, las cuales fueron asumidas por el investigador. También fue necesario un equipo de cómputo con *software* de ofimática y de análisis de seguridad para aplicaciones móviles, estos recursos fueron proporcionados por el investigador. En la Tabla 1 se representa el costo económico del proyecto.

Tabla 1

Costo económico del proyecto

Descripción	Costo	Totales
Horas consultor.	₡ 6000 * 180 horas	₡ 1.080.000
Equipo de cómputo.	₡ 0	₡ 0
Software de ofimática.	₡ 0	₡ 0
Software de Análisis de Seguridad Móvil.	₡ 0	₡ 0
	Total:	**₡ 1.080.000**

Nota. Elaboración propia.

1.6 Objetivos

Se ha seleccionado la taxonomía original de Benjamín Bloom para plantear los objetivos de esta investigación, debido a su robustez y su estructuración jerárquica. Se tomarán en cuenta varios niveles del modelo para llegar de forma más precisa al objetivo general (Bloom, 1956).

1.6.1 Objetivo general

Analizar la seguridad de la aplicación móvil Sistema de Captura de Datos (SICAD) de la Encuesta Nacional Agropecuaria (ENA), del Instituto Nacional de Estadística y Censos (INEC), tomando como referencia el Estándar de Verificación de Seguridad de Aplicaciones Móviles (MASVS) Versión 1.2.

1.6.2 Objetivos específicos

- Conocer el estándar MASVS Versión 1.2.
- Comprender el estándar de MASVS Versión 1.2.
- Efectuar un análisis de seguridad práctico sobre la aplicación móvil SICAD de la encuesta ENA del INEC, tomando como referencia el estándar MASVS Versión 1.2.
- Analizar el nivel de confianza de seguridad que tiene la aplicación móvil SICAD de la encuesta ENA del INEC, tomando como referencia el estándar MASVS Versión 1.2.

1.7 Alcances y limitaciones

1.7.1 Alcances

Se entregará un documento digital al Departamento de Desarrollo de Sistemas de Información del INEC, con el análisis que determina el nivel de confianza de seguridad que tiene la aplicación móvil SICAD de la encuesta ENA, tomando como referencia el estándar de MASVS Versión 1.2.

1.7.2 Limitaciones

Existe información de carácter confidencial para el INEC y el Departamento de Desarrollo de Sistemas de Información de la UTSI que será tomada como referencia, mas no podrá ser expuesta en el presente documento; tales como documentos de uso interno y código fuente.

1.8 Marco de referencia organizacional y socioeconómico

1.8.1 Historia

El INEC cuenta con amplia experiencia en su historia:

La producción de estadísticas oficiales a cargo del INEC tiene sus antecedentes en el siglo XIX, cuyos principales hitos por destacar son:

- En 1824, el primer Congreso Constituyente ordenó levantar un Censo de Población. Sin embargo, por restricciones metodológicas, en realidad no se considera censo, sino un recuento poblacional.
- 1861: se materializa la creación de la Oficina Central de Estadística y la promulgación de la Ley de Censos. Este año se publica el decreto que declara la realización de los censos de población cada 10 años, así como las estadísticas de la riqueza agrícola, fabril y comercial.
- 1862: se crea con una duración de un año, la Dirección General de Estadística, dentro del Ministerio del Interior.
- 1864: se ejecuta del Primer Censo General de Población, el cual se considera el primer censo oficial.
- En 1883: Ley N.º 38 del 12 de julio se crea la Dirección General de Estadística con carácter permanente y como dependencia del Ministerio de Fomento. Se realiza el II Censo Nacional de Población.
- 1948: con los Decretos N.º 61 y N.° 72 se centraliza en la Dirección General de Estadística, la dirección técnica de las estadísticas nacionales y se crea el Consejo Nacional de Estadística, para orientar técnicamente la estructuración y desarrollo de las estadísticas nacionales.
- Para la ejecución de los Censos era habitual crear una “Oficina de Censo”, la última constituida en enero de 1949. A partir del 1 de enero de 1951, se integra

en la Dirección General de Estadística y Censos (DGEC), la cual se mantuvo hasta 1998.

- En mayo de 1953 se promulga la Ley 1565 “Ley General de Estadística”.
- Ley 7839 del 15 de octubre de 1998 (publicada el 4 de noviembre de 1998), se crea el Sistema de Estadística Nacional y el Instituto Nacional de Estadística. Esta ley deroga la anterior. (INEC, 2020).

1.8.2 Tipo de negocio y mercado meta.

El INEC se dedica a:

- Propiciar la integración de la producción y utilización de la información estadística de las instituciones públicas como insumo básico para la formulación y evaluación de políticas, programas y proyectos prioritarios.
- Establecer los mecanismos y procedimientos requeridos para la coordinación de acciones de las diferentes instituciones públicas y privadas involucradas en la producción, difusión y utilización de la información generada en el marco del Sistema de Estadística Nacional.
- Producir la información estadística encomendada por Ley.
- Alcanzar la sostenibilidad del Instituto mediante la definición de mecanismos para contribuir a su financiamiento y el control interno de su accionar.
- Suministrar al público de forma clara y oportuna los resultados de la actividad estadística nacional.
- Definir las políticas y planes estratégicos institucionales para procurar el efectivo cumplimiento de sus funciones.
- Propiciar la adopción y uso intensivo de las modernas tecnologías de procesamiento, transmisión y análisis de la información estadística generada por el Sistema.

- Procurar el adecuado uso de los recursos asignados en consonancia con los procedimientos de formulación y ejecución presupuestaria y de seguimiento y evaluación de las acciones institucionales. (INEC, 2020)

1.8.3 Misión, visión y valores.

Misión: Somos responsables de la gestión de las estadísticas nacionales para orientar las decisiones que promuevan el desarrollo del país.

Visión: Seremos líderes en proveer a la sociedad información geoestadística sobre la realidad costarricense.

Valores institucionales:

Calidad: La producción estadística del INEC y sus servicios se sustenta en metodología estadística sólida, procesos estadísticos adecuados, es oportuna, confiable y accesible.

Credibilidad: Las estadísticas del INEC se elaboran y divulgan respetando la independencia técnica, es objetiva, profesional y transparente, y se reconoce por su amplio respeto y resguardo de la confidencialidad de los datos brindados por sus informantes.

Compromiso: Es la identificación con la misión, metas y valores institucionales, la voluntad de realizar esfuerzos en beneficio de la institución.

Excelencia: El INEC como entidad coordinadora del quehacer estadístico nacional, desarrolla tareas, trabajos y actividades tendientes a producir y entregar productos de calidad con valor añadido y en plazos establecidos, al menor costo posible, y que satisfagan los requerimientos de información estadística de los usuarios. (INEC, 2020)

1.8.4 Políticas institucionales

El contrato de confidencialidad estipulado en el INEC no afecta directamente el objetivo del proyecto, pero sí restringe la publicación de mayor detalle en el documento del proyecto.

1.9 Estado de la cuestión

Foco de la pregunta

En esta revisión sistemática, se pretende localizar trabajos que realicen aportaciones que permitan la comprensión del estándar MASVS de OWASP y su aplicabilidad práctica en el análisis de seguridad sobre aplicaciones móviles Android.

Amplitud y calidad de la pregunta

Problema. Tal como se expresó en el apartado de introducción, se requiere definir el nivel de confianza de seguridad que alcanza una aplicación móvil Android aplicando el estándar MASVS. De modo que el problema se centra en el estudio de trabajos que aporten en la comprensión del estándar MASVS de OWASP y su aplicabilidad práctica en el análisis de seguridad sobre aplicaciones móviles Android.

Pregunta. ¿Qué trabajos aportan en la comprensión del estándar MASVS de OWASP y su aplicabilidad práctica en el análisis de seguridad sobre aplicaciones móviles Android?

Palabras claves. Para la selección de las palabras claves y traducción correcta al inglés de cada término, se revisaron varios artículos sobre análisis de seguridad en aplicaciones móviles para el sistema operativo Android tomando como referencia el estándar de MASVS de OWASP en una búsqueda inicial. Seguidamente, la Tabla 2 muestra una lista de las palabras claves utilizadas para realizar las búsquedas.

Tabla 2

Palabras claves

Área	Palabras claves
Seguridad de aplicaciones móviles Android	MASVS
	OWASP
	Android
	Security
	Mobile

Nota. Elaboración propia.

Intervención. En el contexto de la revisión sistemática planificada, se estudiaron las propuestas existentes sobre análisis de seguridad a una aplicación móvil Android tomando como referencia el estándar MASVS de OWASP, se extraen las más importantes y se procede a un posterior análisis de estas.

Control. En la presente revisión sistemática, aunque se han observado algunos trabajos para obtener las palabras claves, no se considera ningún dato o trabajo inicial que deba estar incluido como trabajo primario en el conjunto de resultados. De esta manera, todos los trabajos primarios son derivados de la aplicación de los criterios definidos y que cumplan con el objetivo buscado.

Resultado. El resultado esperado de esta revisión es conocer las propuestas existentes en cuanto al análisis de seguridad sobre aplicaciones móviles Android basándose en el estándar de MASVS de OWASP, para, posteriormente, analizarlas.

Medida de salida. Para medir los resultados obtenidos se utilizará, en primer lugar, una agrupación de las propuestas encontradas tomando como base el área en la que se centran.

Población. La población por analizar se compone de publicaciones presentes en el repositorio de fuentes de datos seleccionado que estén relacionadas con el objetivo de esta revisión.

Aplicación. Los beneficiarios de la revisión sistemática serán ingenieros de sistemas involucrados en el desarrollo de aplicaciones móviles para el sistema operativo Android del Departamento de Desarrollo de Sistemas de Información del INEC.

Diseño experimental. El metaanálisis de la revisión está enfocado en analizar las aportaciones en el análisis de seguridad de aplicaciones móviles Android tomando como referencia el estándar de MASVS de OWASP, presentes en los estudios primarios para conocer las tendencias actuales.

Selección de fuentes

Definición del criterio de selección de fuentes. El criterio para la selección de las fuentes de búsqueda está basado en la opinión del autor de este trabajo, así como la disponibilidad y accesibilidad vía web e inclusión de motores de búsqueda con presencia de mecanismo de búsqueda mediante palabras claves y consultas avanzadas.

Lenguaje de estudio. El lenguaje de los estudios primarios será el inglés o español, este último en caso de que cumpla con los criterios de extracción. Los estudios primarios serán extraídos mediante consultas por medio de palabras clave en inglés. El informe de la revisión sistemática se realiza en español.

Identificación de fuentes

Método de selección de fuentes. El autor de este trabajo recomienda, según su experiencia y conocimiento profesional, sobre cuáles fuentes cumplen con los criterios de selección definidos y ofrecen artículos de calidad sobre el área de estudio, además de presentar disponibilidad por web y motores de búsqueda que permitan consultas avanzadas.

Lista de fuentes. La revisión sistemática se ejecutó en la fuente Scholar Google.

Cadenas de búsqueda. Utilizando combinaciones con el operador lógico AND sobre las palabras clave que se identificaron anteriormente, se establece la cadena de búsqueda utilizada: MASVS AND Mobile AND Android AND Security AND OWASP.

Selección de fuentes después de la evaluación. Posterior a la revisión sobre las fuentes seleccionadas, en caso necesario se realiza una fase de refinado en la que se identifican e incluyen los estudios primarios relevantes que, a juicio del autor, sean necesarios para completar el estudio y no hayan sido recuperados en las búsquedas realizadas.

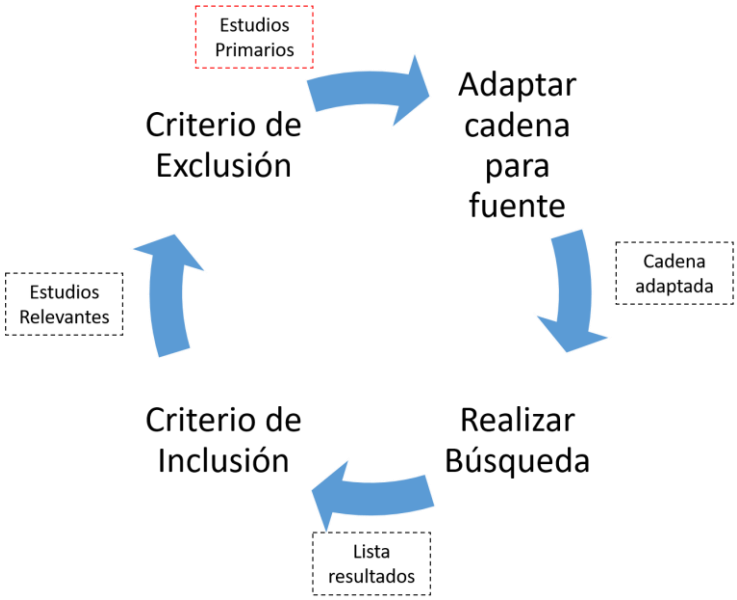
Comprobación de las fuentes. La fuente identificada es Scholar Google, la cual se sometió a una fase de comprobación por parte del investigador.

Selección de los estudios

Procedimiento para la selección de los estudios. En esta revisión se sigue un proceso iterativo e incremental. Por cada fuente seleccionada, se realiza un proceso iterativo formado por búsquedas con cadenas de texto (previamente establecidas), extracción, visualización y análisis de información. En la Figura 1 se muestran las etapas que componen el proceso de revisión.

Figura 1

Proceso de revisión de estudios



Nota. Elaboración propia.

A continuación, se explica el proceso de revisión. Primeramente, se adapta la cadena de texto al motor de búsqueda de la fuente seleccionada y se ejecuta la consulta. Sobre el conjunto de resultados obtenidos, se aplica el criterio de inclusión a modo de filtro, de esta forma, se obtiene un conjunto de estudios relevantes. Posteriormente, al conjunto de estudios relevantes se aplica el criterio de exclusión, para así obtener el conjunto de estudios primarios.

Definición del criterio de inclusión y exclusión de estudios. Los estudios seleccionados deben presentar relación en el análisis de seguridad de aplicaciones móviles para el sistema operativo Android tomando como referencia el estándar de MASVS de OWASP y que brinden aportaciones en la comprensión y aplicabilidad práctica del estándar.

Como criterio de inclusión, se realiza un análisis sobre el título, las palabras claves y resumen de cada documento, de esta forma, se detecta, en primera instancia, cómo están relacionadas estas palabras a la búsqueda de la investigación.

Como criterio de exclusión, se toma, primeramente, el factor de tiempo. Se valoran publicaciones a partir del año 2016, año en que se dan las primeras publicaciones sobre el estándar de MASVS de OWASP. Además, la investigación solo incorpora estudios en idioma español e inglés y se centra, principalmente, en la lectura y análisis del resumen del documento y sus conclusiones; en algunos casos se realiza con una lectura más profunda de este.

Definición de tipos de estudio. Los tipos de estudios primarios seleccionados son artículos presentes en la fuente, que cumplan con los criterios establecidos y hayan sido obtenidos como producto al aplicar el procedimiento de selección.

Ejecución de la revisión

Ejecución de la selección en la fuente Google Scholar.

Selección de estudios iniciales. Al realizar la consulta inicial adaptada al motor de búsqueda, aplicado con todas las palabras, se obtuvo 35 resultados. Tras aplicar el criterio de inclusión, se obtienen como resultado 10 documentos relevantes, de los cuales, posterior a la aplicación de los criterios de exclusión, se obtienen tres estudios primarios que se enlistan en la Tabla 3.

Tabla 3

Estudios primarios de la fuente Google Scholar

Estudios Primarios
Implementation and Security Evaluation of User-Customized Content in a Mobile Application. Linköping University, Department of Computer and Information Science. Wällstedt, Viktor.
Vetting the Security of Mobile Applications. NIST. Michael Ogata, Joshua Franklin, Jeffrey Voas, Vincent Sritapan, Stephen Quirolgico.
Auditoría de seguridad de aplicaciones móviles. Universitat Oberta de Catalunya (UOC). Rodríguez Sánchez, Eva.

Nota. Elaboración propia.

Evaluación de la calidad de los estudios. Todos los documentos presentes en la fuente Google Scholar tienen presunción de calidad, dado que, para estar publicados en esta fuente, han debido pasar una serie de filtros y evaluaciones.

Revisión de la selección. La selección de los estudios primarios realizada ha sido validada, de tal forma que se tenga la seguridad de no haber dejado atrás ningún estudio relevante en esta fuente.

Definición del criterio de inclusión y exclusión de información. Para extraer la información de relevancia de los estudios primarios, se centró, principalmente, en estudiar las aportaciones de interés que realizan sobre la comprensión y aplicabilidad práctica del estándar MASVS de OWASP sobre aplicaciones móviles para el sistema operativo Android.

Formulario para la extracción de información. El formulario utilizado para documentar la extracción de información realizada sobre cada estudio primario consta de varias partes, el cual está representado en la Tabla 4.

Tabla 4

Formato utilizado para documentar la extracción de información

Identificación	Título
	Publicación
	Autores
	Referencia
Descripción	Área
	Resumen
Aspectos por destacar	

Nota. Elaboración propia.

La Tabla 4 se explica de la siguiente manera:

- Una primera parte de identificación del estudio en la que se muestra el título, publicación, autores y referencias.
- La segunda parte de descripción general en la que se analiza el área o áreas de interés sobre las que realiza aportaciones relevantes, así como una breve descripción o resumen del contenido del estudio.
- Por último, una parte en la que se documentan varios aspectos por destacar, los cuales se han considerado importantes a medida que se realizaba el análisis del estudio primario.

Extracción de la información

Extracción de resultados objetivos y subjetivos. A continuación, se enlista, por medio de las Tablas 5, 6 y 7, la extracción de la información de los estudios primarios de la fuente Google Scholar.

Tabla 5

Extracción de información de los estudios primarios de la fuente Google Scholar

Identificación	Título: **Implementation and Security Evaluation of User-Customized Content in a Mobile Application** Publicación: Linköping University, Department of Computer and Information Science. Autores: Wällstedt, Viktor Referencia: https://www.diva-portal.org/smash/get/diva2:1334227/FULLTEXT01.pdf
Descripción	Área: Verificación de seguridad en aplicaciones móviles de Android. Resumen: Proporcionan un estudio de cómo diseñar e implementar un sistema para examinar la posibilidad de permitir a los clientes definir módulos que podrían renderizarse en una aplicación móvil. Además, se identifica y utiliza un método para evaluar la seguridad del sistema y presenta el resultado del nivel de seguridad general del sistema utilizando el estándar MASVS (Wällstedt, 2019).
Aspectos por destacar	Hacen referencia a los documentos proporcionados por OWASP: • El estándar MASVS. • La Guía de Verificación de Seguridad Móvil (MSTG). La guía MSTG apunta a aplicaciones nativas desarrolladas para Android o iOS.

Nota. Elaboración propia.

Tabla 6

Extracción de información de los estudios primarios de la fuente Google Scholar

Identificación	Título: **Vetting the Security of Mobile Applications**. Publicación: National Institute of Standards and Technology (NIST). Autores: Michael Ogata, Joshua Franklin, Jeffrey Voas, Vincent Sritapan, Stephen Quirolgico. Referencia: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-163r1.pdf
Descripción	Área: Verificación de seguridad en aplicaciones móviles de Android. Resumen: Proporciona orientación sobre la verificación de aplicaciones, requisitos de seguridad para aplicaciones móviles, herramientas para probarlas y determinar si una aplicación móvil es aceptable para la implementación en los dispositivos móviles de una organización (NIST, 2019).
Aspectos por destacar	Se destaca la referencia que la NIST hace sobre el estándar MASVS; “El Proyecto de seguridad de aplicaciones web abiertas (OWASP) mantiene múltiples recursos útiles sobre seguridad y pruebas de aplicaciones móviles. Su verificación de seguridad de aplicaciones móviles Estándar (MASVS) es un modelo detallado de seguridad de aplicaciones móviles que se puede utilizar para proporcionar requisitos de seguridad básicos para una organización” (NIST, 2019, p. 5-6).

Nota. Elaboración propia.

Tabla 7

Extracción de información de los estudios primarios de la fuente Google Scholar

Identificación	Título: **Auditoría de seguridad de aplicaciones móviles.** Publicación: Universitat Oberta de Catalunya (UOC). Autores: Rodríguez Sánchez, Eva Referencia: http://openaccess.uoc.edu/webapps/o2/handle/10609/95927
Descripción	Área: Verificación de seguridad en aplicaciones móviles de Android. Resumen: Proporcionan una guía para el proceso de la auditoría de seguridad de las aplicaciones móviles tomando como referencia el estándar MASVS, sobre aplicaciones móviles en la plataforma Android (Rodrígez Sanchez, 2019).
Aspectos por destacar	El estándar MASVS define dos niveles estrictos de verificación de seguridad (L1 y L2), así como un conjunto de requisitos de resistencia a la ingeniería inversa (MASVS-R). Se encuentran a disposición, los siguientes documentos de OWASP para aplicar el estándar: • El estándar MASVS, con los requisitos genéricos de seguridad y los distintos niveles de verificación que se deberá elegir convenientemente en el momento de llevar a cabo la auditoría de seguridad de aplicación móvil. • La guía MSTG, que proporcionará durante todo el proceso de auditoría de seguridad, las instrucciones necesarias para la verificación de cada uno de los requerimientos del estándar MASVS, así como las mejores prácticas en seguridad para cada sistema operativo. • La Lista de Chequeo de Seguridad para Aplicaciones Móviles que permitirá aplicar los requerimientos del estándar MASVS a la evaluación práctica de la aplicación.

Nota. Elaboración propia.

Análisis de resultados

Estudios analizados. Se adjunta un resumen de los estudios analizados por cada fuente en la Tabla 8, cuántos han sido considerados como relevantes y cuántos han sido añadidos como estudios primarios.

Tabla 8

Resumen de los estudios analizados

Fuente	Estudios	Relevantes	Primarios	Primarios (refinamiento)
Scholar Google	35	10	3	0
Total	**35**	**10**	**3**	**0**

Nota. Elaboración propia.

Presentación de resultados. Para presentar los resultados obtenidos, se agrupan los estudios primarios según las áreas sobre las que realiza una aportación significativa, con el fin de conocer las tendencias que actualmente están siguiendo los investigadores del área.

El área por considerar es: verificación de seguridad en aplicaciones móviles para Android. En la Tabla 9, se muestra dicha clasificación, en la que se aprecia cómo la mayoría de los trabajos se centran en verificación de seguridad en aplicaciones móviles de Android.

Tabla 9

Clasificación de estudios

Verificación de seguridad en aplicaciones móviles para Android.	
Wällstedt, Viktor	X
Michael Ogata, Joshua Franklin, Jeffrey Voas, Vincent Sritapan, Stephen Quirolgico	X
Rodríguez Sánchez, Eva	X
Total	**3**

Nota. Elaboración propia.

Hallazgos

El estado de la cuestión permite conocer y comprender las tendencias, conceptualización y aplicabilidad del estándar MASVS de OWASP, sobre aplicaciones móviles para el sistema operativo Android. Por lo cual, para el análisis del estado de la cuestión, se ha empleado la técnica de revisión sistemática, definiendo y aplicando un protocolo formal para la revisión que permita centrar la pregunta de investigación y el tipo de resultados (estudios primarios) que se requieren obtener y, posteriormente, analizar.

Por otra parte, con el análisis de los estudios primarios, se determina que se puede aplicar el estándar MASVS de OWASP, para así determinar el nivel de confianza de seguridad que alcanza una aplicación móvil para el sistema operativo Android desarrollada por el Departamento de Desarrollo de Sistemas del INEC.

Capítulo 2. Marco teórico

En esta sección del documento se desarrollan los fundamentos conceptuales considerados para alcanzar los objetivos de la investigación.

Seguridad de aplicaciones móviles

Las aplicaciones móviles actualmente son parte esencial de la vida, se han convertido en accesorios que están presentes tanto en los negocios como en la vida personal de las personas. No obstante, sin excepción, cada nueva tecnología trae consigo nuevos riesgos de seguridad, que se convierten en un reto por enfrentar para la industria de la ciberseguridad.

A través de las aplicaciones móviles, se brinda acceso a diferentes servicios como la comunicación, búsqueda de información, navegación, gestión de datos personales, empresariales o financieros; debido a los diversos usos de los dispositivos móviles, suelen contener mucha información sobre la vida del propietario, esta última es una de las razones por la cual en la actualidad son atrayentes para los atacantes, por lo cual: “la seguridad móvil tiene que ver con la protección de los datos” (OWASP, 2020, p. 4).

Para un atacante, las aplicaciones móviles son una puerta a obtener acceso sin filtro a la vida de su propietario, por lo tanto, es importante desarrollar y verificar las aplicaciones móviles tomando como referencia un estándar de seguridad. “Este estándar debe centrarse en la forma en que las aplicaciones móviles gestionan, almacenan y protegen la información sensible” (OWASP, 2020, p. 4).

Estándar de Verificación de Seguridad de Aplicaciones Móviles

Conocido en inglés como OWASP Mobile Application Security Verification Standard (MASVS). Es un modelo de seguridad para aplicaciones móviles que brinda requerimientos de seguridad básicos aplicables a cualquier organización (NIST, 2019).

"El MASVS es un esfuerzo comunitario para establecer un marco de requisitos de seguridad necesarios para diseñar, desarrollar y probar la seguridad de aplicaciones móviles iOS y Android" (OWASP, 2020, p. 7).

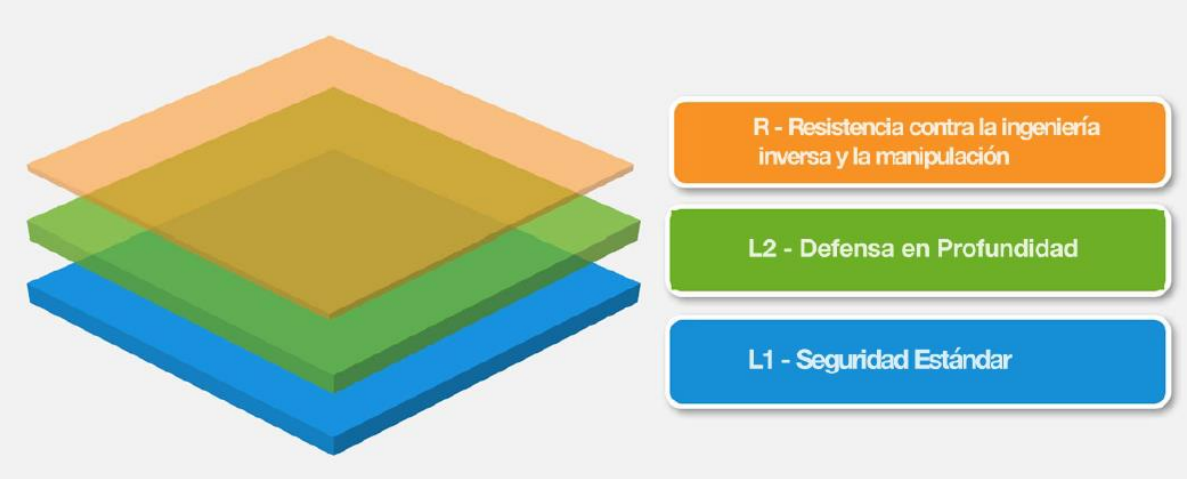
Este estándar puede ser utilizado por arquitectos de *software*, desarrolladores que buscan crear aplicaciones móviles seguras, así como por encargados de calidad de *software*, encargados de seguridad de *software* o auditores de seguridad para verificar la seguridad en aplicaciones móviles (OWASP, s.f.-q).

Niveles de MASVS

El estándar MASVS define tres niveles de verificación de seguridad: MASVS-L1, MASVS-L2 y MASVS-R (OWASP, 2020b). En la Figura 2 se muestran los niveles de seguridad que define el estándar MASVS.

Figura 2

Niveles de seguridad del estándar MASVS



Nota. Tomado del estándar MASVS (OWASP, 2020b).

MASVS-L1: Seguridad Estándar. Este nivel contiene un conjunto de requerimientos genéricos de seguridad recomendados para todas las aplicaciones móviles. Una aplicación móvil que cumpla con los requerimientos de este nivel dará como resultado una aplicación segura que sigue las mejores prácticas de seguridad y no sufre de las vulnerabilidades más comunes (OWASP, 2020b).

MASVS-L2: Defensa en Profundidad. Este nivel contiene requerimientos para aplicaciones que manejan datos sensibles, como las aplicaciones de banca móvil o gestión de información sensible (OWASP, 2020b). Es importante tener en cuenta que todos los requerimientos del MASVS-L1 están integrados dentro del nivel MASVS-L2.

MASVS-R: Resistencia contra la Ingeniería Inversa y la Manipulación. Este nivel contiene requerimientos que ayudan a prevenir amenazas específicas del lado del cliente, como alteración, modificación o ingeniería inversa para extraer código o datos sensibles, es adecuado en aplicaciones móviles que manejan datos altamente confidenciales y puede servir como medio para proteger la propiedad intelectual o dificultar la manipulación de una aplicación (OWASP, 2020b).

Requerimientos del estándar MASVS.

Los requerimientos o controles del estándar MASVS versión 1.2 se agrupan en ocho categorías basadas en el objetivo o alcance técnico, en el Anexo 2 se muestra la lista completa de estos requerimientos. Las categorías de MASVS-V1 a la V7 corresponden a los niveles MASVS-L1 y L2, mientras que la categoría MASVS-V8 solo le concierne al nivel MASVS-R. A continuación, se detalla cada categoría:

MASVS-V1: Requisitos de Arquitectura, Diseño y Modelado de Amenazas. Este conjunto de requerimientos verifica que el tema de la seguridad se tome en cuenta desde la planificación de la aplicación, así como en la documentación del proyecto (OWASP, 2020b).

MASVS-V2: Requerimientos de Almacenamiento de Datos y Privacidad. Los dispositivos móviles pueden perderse o robarse más fácilmente que cualquier otro dispositivo, por lo que es más probable que el atacante pueda tener acceso físico a él, por lo cual deben aplicarse medidas para dificultarle la recuperación de datos sensibles, como las credenciales del usuario y la información privada (OWASP, 2020b).

MASVS-V3: Requerimientos de Criptografía. Este conjunto de requerimientos tiene como propósito asegurar el uso correcto de las mejores prácticas de criptografía en las aplicaciones móviles (OWASP, 2020b).

MASVS-V4: Requerimientos de Autenticación y Manejo de Sesiones. Los requerimientos de esta categoría se centran en la gestión de cuentas y sesiones del usuario (OWASP, 2020b).

MASVS-V5: Requerimientos de Comunicación a través de la Red. Estos requerimientos tienen el objetivo de asegurar la confidencialidad e integridad de la información intercambiada entre la aplicación móvil y los servicios del servidor (OWASP, 2020b).

MASVS-V6: Requerimientos de Interacción con la Plataforma. Los requerimientos de esta categoría buscan verificar el uso seguro de las interfaces de programación de aplicaciones, conocidas como API, de la plataforma y componentes estándar (OWASP, 2020b).

MASVS-V7: Requerimientos de Calidad de Código y Configuración del Compilador. Estos requerimientos buscan asegurar que se siguieron las prácticas de seguridad básicas en el desarrollo de la aplicación (OWASP, 2020b).

MASVS-V8: Requerimientos de Resistencia ante la Ingeniería Inversa. Estos requerimientos buscan añadir controles de protección adicionales y específicos contra las amenazas a las aplicaciones; un ejemplo de estas amenazas es la ingeniería inversa, se concentra en requerimientos que dificulten el acceso del atacante a datos sensibles, configuración o código fuente (OWASP, 2020b).

En la Tabla 10, se observa la cantidad total de requerimientos de cada nivel (MASVS-L1, L2 y R), de igual forma, se muestra la cantidad de requerimientos en cada una de las categorías que conforman cada nivel del estándar.

Tabla 10

Cantidad de requerimientos por categoría de cada nivel MASVS-L1, L2 y R

Categoría	MASVS-L1	MASVS-L2	MASVS-R
MASVS-V1	5	12	0
MASVS-V2	7	15	0
MASVS-V3	6	6	0
MASVS-V4	8	12	0
MASVS-V5	3	6	0
MASVS-V6	8	11	0
MASVS-V7	9	9	0
MASVS-V8	0	0	13
Total	**46**	**71**	**13**

Nota. Elaboración propia basado en el estándar MASVS versión 1.2.

Guía de Pruebas de Seguridad para Móviles

La Guía de Pruebas de Seguridad para Móviles (traducido de Mobile Security Testing Guide, [MSTG]) es una guía para la verificación de la seguridad de las aplicaciones móviles. En la cual se describen procedimientos técnicos y casos de prueba que podrán utilizarse para validar el cumplimiento de los requerimientos del estándar MASVS. La guía MSTG proporciona recomendaciones detalladas, procedimientos técnicos y casos de pruebas de verificación para los sistemas operativos móviles Android o iOS (OWASP, 2020b).

Lista de Chequeo de Seguridad Móvil

Conocida en inglés como Mobile Security Checklist, es una herramienta en formato de Excel proporcionada por OWASP, que contiene el listado de requerimientos de seguridad de MASVS, con el fin de evaluar si la aplicación móvil cumple con el aspecto evaluado. Cada requerimiento contiene enlaces a la guía MSTG para que el evaluador pueda consultar el procedimiento por seguir para verificar cada requisito (OWASP, 2020a).

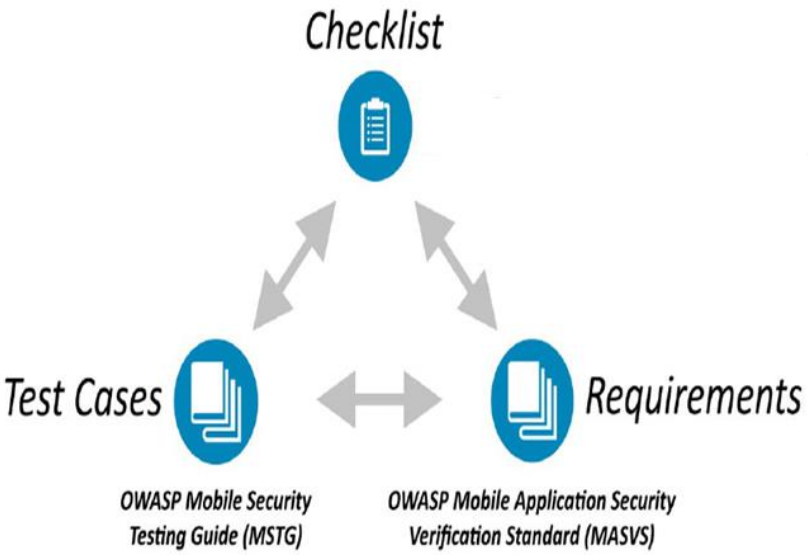
Interrelación del Estándar, Guía de Pruebas y Lista de Chequeo de Seguridad Móvil

El estándar MASVS, la guía MSTG y la lista de chequeo utilizan los mismos requerimientos de seguridad. Dependiendo del uso que se les vaya a dar, se pueden utilizar individualmente o en conjunto (OWASP, s.f.-q).

El estándar MASVS de OWASP es el documento que define los requerimientos que deberían cumplir las aplicaciones móviles para alcanzar el nivel de seguridad adecuado; mientras que la guía MSTG define procedimientos técnicos para verificar los requerimientos de MASVS. Por otro lado, la lista de chequeo es una herramienta para llevar un control de la evaluación y del cumplimiento de la aplicación móvil con los requerimientos (OWASP, 2020b). En la Figura 3 se muestra la interrelación del estándar MASVS, la guía MSTG y la lista de chequeo.

Figura 3

Interrelación de MASVS, MSTG y Checklist de OWASP



Nota. Tomado de la guía MSTG (OWASP, s.f.-l)

Pruebas de seguridad de aplicaciones móviles

En la guía MSTG, el concepto de pruebas de seguridad de aplicaciones móviles se refiere a la evaluación de la seguridad de aplicaciones mediante análisis estático y dinámico (OWASP, s.f.-j).

Análisis de vulnerabilidad

El análisis de vulnerabilidades es el proceso en el cual se buscan fallos de seguridad en una aplicación. El análisis estático y dinámico son tipos de análisis de vulnerabilidades (OWASP, s.f.-r).

Análisis estático. Las pruebas de seguridad estáticas en aplicaciones (traducido de Static Application Security Testing, [SAST]) se centran en evaluar aplicaciones sin ejecutarlas, se revisa el código fuente de manera manual o automática.

En la revisión de código manual, el evaluador revisa manualmente el código fuente en búsqueda de vulnerabilidades. Mientras en el análisis de código fuente automático, se utilizan herramientas que aceleran el proceso y verifican que el código fuente cumpla con un grupo predefinido de reglas o mejores prácticas de la industria.

El enfoque de estas pruebas estáticas es garantizar la implementación de controles de seguridad y buenas prácticas en el código fuente (OWASP, s.f.-o).

Análisis dinámico. Las pruebas de seguridad dinámicas en aplicaciones (traducido de Dynamic Application Security Testing, [DAST]) se centran en examinar la aplicación en tiempo de ejecución, este análisis puede ser manual o automático. El enfoque de estas pruebas es detectar vulnerabilidades o fallos de seguridad mientras que se ejecuta la aplicación (OWASP, s.f.-e).

Prueba de penetración

La guía de MSTG para guiar el proceso de evaluación se apoya de las fases de una prueba de seguridad que se estructura de la siguiente manera:

Preparación. En esta fase se definen los requerimientos de seguridad que se evalúan en la aplicación móvil. De igual forma, se define el nivel de seguridad en que se

probará la aplicación, según recomienda la guía MSTG, este nivel podrá definirse recorriendo la lista de verificación de los controles MASVS-L1 y MASVS-L2, para así seleccionar los requerimientos de seguridad que mejor se adapten a las necesidades de seguridad, legales, regulatorias o financieras de la organización (OWASP, s.f.-m).

A continuación, deben identificarse los datos sensibles, si la organización cuenta con una política de clasificación de datos, se tomará esta como referencia; en caso de que no exista una política, la guía MSTG recomienda utilizar la siguiente lista de información que se considera confidencial (OWASP, s.f.-g). La guía MSTG indica:

> Cuando no haya una política de clasificación de datos disponible, utilice la siguiente lista de información que generalmente se considera sensible:
>
> - Información de autenticación de usuario (credenciales, PIN, etc.).
> - Información de identificación personal (PII) de la que se puede abusar para el robo de identidad: números de seguro social, tarjeta de crédito, números de cuentas bancarias, información médica.
> - Identificadores de dispositivos que pueden identificar a una persona.
> - datos altamente sensibles cuyo compromiso podría causar daños a la reputación y / o costos financieros.
> - Cualquier dato cuya protección sea una obligación legal.
> - Cualquier dato técnico generado por la aplicación (o sus sistemas relacionados) y utilizado para proteger otros datos o el sistema en sí (por ejemplo, claves de cifrado) (OWASP, s.f.-g).

Finalmente, se debe obtener la protección legal necesaria para el evaluador tomando en cuenta que, en diferentes partes del mundo, es ilegal atacar un sistema sin autorización (OWASP, s.f.-m).

Recopilación de información. En esta fase se pretende recolectar información sobre la arquitectura de la aplicación, sus casos de uso empresariales, el contexto medioambiental y arquitectónico de la aplicación móvil.

El contexto medioambiental se enfoca en la interacción de la organización, sus procesos e interesados con la aplicación móvil por evaluar; se incluyen los objetivos de la organización, interacción de los usuarios, la industria y grupos interesados sobre la aplicación. Mientras el contexto arquitectónico se enfoca en aspectos técnicos de la aplicación, se incluyen aspectos de la aplicación móvil, el sistema operativo que la sustenta, la red y sus servicios remotos (OWASP, s.f.-h).

Mapeo de la aplicación. Basado en la información de fases anteriores y complementándola con un escaneo automático o una exploración manual de la aplicación móvil, el evaluador debe identificar puntos de entrada, características, datos y las principales vulnerabilidades potenciales (OWASP, s.f.-i).

Explotación. En esta fase el evaluador intenta atacar la aplicación explotando las vulnerabilidades identificadas en la fase anterior, se valida si las vulnerabilidades son verdaderas y positivas (OWASP, s.f.-f).

Reportando. Esta fase es clave para el cliente porque el evaluador documenta los hallazgos encontrados en la aplicación evaluada. Esta documentación podrá incluir: el resumen ejecutivo, el alcance y el contexto, métodos utilizados, fuentes de información, hallazgos priorizados, hallazgos detallados y recomendaciones para arreglar cada defecto (OWASP, s.f.-n).

Capítulo 3. Marco metodológico

3.1 Tipo de investigación

Este trabajo se considera una investigación aplicada, porque busca utilizar el conocimiento existente y no producir nuevo conocimiento. Se utiliza el estándar MASVS versión 1.2 de OWASP para determinar el nivel de confianza de seguridad del aplicativo móvil SICAD de la encuesta ENA del INEC.

3.2 Alcance investigativo

El estudio tiene un alcance descriptivo, motivado por los objetivos del proyecto; que consisten en brindar un análisis de seguridad de la aplicación móvil SICAD de la encuesta ENA del INEC, tomando como referencia el estándar MASVS. Esto se logra verificando la aplicación con los casos de pruebas de la guía MSTG y la lista de chequeo de seguridad móvil que proporciona el estándar MASVS versión 1.2.

3.3 Enfoque

En esta investigación se utiliza un enfoque cuantitativo, pues, según se aplique la evaluación de los requisitos, requerimientos o controles del estándar MASVS sobre la aplicación móvil, se puede conocer el nivel de confianza de seguridad que tiene la aplicación evaluada con respecto al estándar.

3.4 Diseño

Este estudio busca analizar el nivel de confianza de seguridad de la aplicación móvil SICAD de la encuesta ENA del INEC, es evaluada de manera práctica tomando como referencia el estándar MASVS versión 1.2.

3.5 Población y muestreo

La población está conformada por la aplicación móvil SICAD versión 10.0.30 de la encuesta ENA del INEC. Se utiliza la técnica de muestreo no probabilístico basado en método intencional o por conveniencia, donde el individuo que conviene investigar

es el aplicativo móvil desarrollado en el sistema operativo Android y su situación en materia de su seguridad.

3.6 Instrumentos de recolección de datos

En el proceso de análisis de seguridad de la aplicación móvil, se utilizan los siguientes instrumentos para recolectar información:

- Código fuente.
- Diagramas de flujo.
- Manual de usuario.
- Casos de uso.
- Documentación técnica.

3.7 Técnicas de análisis de información

Se analizan los resultados obtenidos después de la evaluación de los requerimientos o controles del estándar. Después de procesar estos datos, se obtienen los fundamentos necesarios para analizar el objeto de estudio y otros aspectos importantes para la realización del proyecto.

3.8 Estrategia de desarrollo de la propuesta

Se emplea la metodología definida por el estándar MASVS Versión 1.2 para evaluar la aplicación móvil SICAD de la encuesta ENA del INEC.

Para el desarrollo de la propuesta, se debe recopilar información de la aplicación móvil, se obtiene documentación técnica de la aplicación (casos de uso, manual de usuario y diagrama de flujo) y código fuente. Posteriormente, en la fase de evaluación, se utiliza la lista de chequeo de seguridad móvil que proporciona el estándar; esta verifica el cumplimiento de cada requerimiento de seguridad a través de la realización de casos de pruebas definidos en la guía MSTG.

Capítulo 4. Propuesta de solución

En esta propuesta, se pretende analizar de forma práctica el nivel de confianza de seguridad del aplicativo móvil SICAD de la encuesta ENA del INEC, basándose en el estándar MASVS versión 1.2. El desarrollo de esta propuesta se basa en la versión actual del estándar MASVS que incluye los siguientes documentos:

- Estándar MASVS versión 1.2, última fecha de actualización marzo 17, 2020.
- Guía MSTG versión 1.2, última fecha de actualización mayo 12, 2020.
- Lista de chequeo versión 1.2, última fecha de actualización agosto 11, 2020.

A continuación, en la Figura 4 se muestran las cinco fases que conforman la metodología que se utiliza para llevar a cabo la solución de este proyecto:

Figura 4

Fase de la propuesta



Nota. Elaboración propia.

Fase de la propuesta

Preparación

En esta fase se define el alcance, al realizar una revisión a través de toda la lista de chequeo de seguridad móvil de OWASP, para extraer los requerimientos de seguridad aplicables a la aplicación móvil SICAD. En la lista de chequeo que se encuentra en el Anexo 2 en la columna “Estado”, el evaluador debe indicar, por medio de un No Aplica (N/A), cuáles requerimientos no serán evaluados en el análisis.

La selección de los requerimientos de seguridad se lleva a cabo por medio de un análisis de los casos de negocio, obligaciones regulatorias y legales que rigen la información que se gestiona en el aplicativo móvil SICAD. Mientras que los

requerimientos no evaluados son excluidos, debido a que el objetivo del requerimiento del estándar MASVS evalúa un elemento tecnológico que no es necesario en los casos de negocio del aplicativo móvil SICAD.

Recolección de información

En esta fase, se recopila toda información relacionada a la aplicación móvil SICAD de la encuesta ENA del INEC, se incluye documentación de casos de negocio, técnica y código fuente de la aplicación.

Mapeo de la aplicación.

En esta fase, se realiza un análisis de la información recolectada en la fase previa. Con la información recolectada, se pretende comprender el funcionamiento del aplicativo y su código fuente para la posterior evaluación. En caso de requerir alguna aclaración sobre el tema, se contará con el apoyo del ingeniero de sistemas responsable de la aplicación móvil SICAD de la encuesta ENA. Para la próxima fase se utilizan los casos de pruebas definidos en la guía MSTG.

Evaluación de seguridad de la aplicación móvil Android

En esta fase, se utiliza la lista de chequeo de seguridad móvil de OWASP que se encuentra en el Anexo 2, la cual es proporcionada por el estándar MASVS. Esta lista contiene cada requerimiento de seguridad que se evalúa.

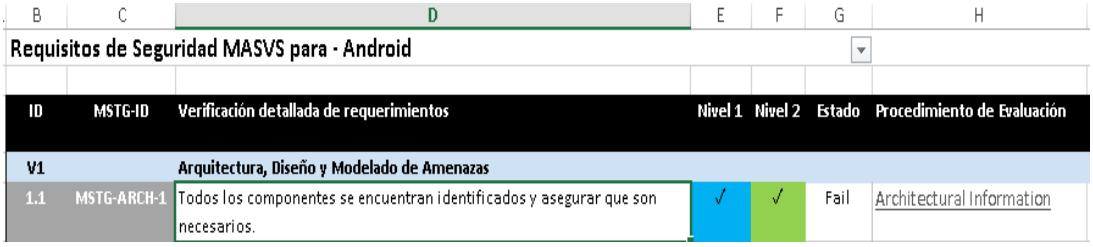
Para verificar cada requerimiento de seguridad, se procede a realizar los casos de pruebas necesarios que establece la guía MSTG; de esta forma, se determina el cumplimiento y nivel de seguridad del aplicativo móvil. A continuación, se detalla la forma como se evalúa cada requerimiento de seguridad definido en el alcance de esta prueba.

1. Primeramente, se toma como referencia la lista de chequeo de seguridad móvil de OWASP, por cada fila del documento de Excel, se realiza la lectura y análisis

de los casos de pruebas señalados en la columna “Procedimiento de Evaluación”, según se muestra en la Figura 5.

Figura 5

Ejemplo de requerimiento de seguridad evaluado



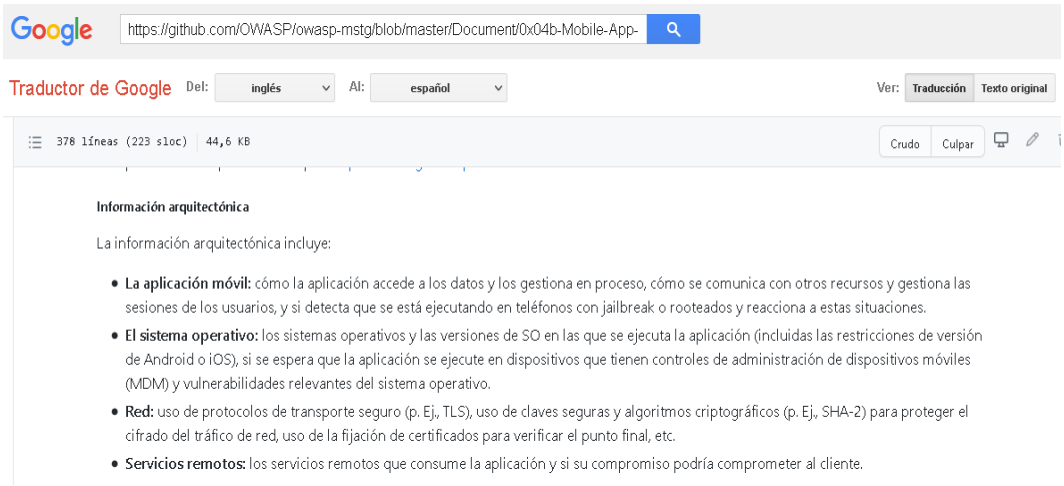
B	C	D	E	F	G	H
Requisitos de Seguridad MASVS para - Android						
ID	MSTG-ID	Verificación detallada de requerimientos	Nivel 1	Nivel 2	Estado	Procedimiento de Evaluación
V1		Arquitectura, Diseño y Modelado de Amenazas				
1.1	MSTG-ARCH-1	Todos los componentes se encuentran identificados y asegurar que son necesarios.	√	√	Fail	Architectural Information

Nota. Tomado de lista de chequeo del estándar MASVS (OWASP, 2020a).

2. Seguidamente, la columna “Procedimiento de Evaluación” contiene enlaces a la guía MSTG que dando clic podrá abrir a través de un navegador web; al ingresar al enlace, se mostrará una página web con los casos de pruebas que definen el cómo y qué debe evaluarse en cada requerimiento de seguridad; según se muestra en la Figura 6.

Figura 6

Casos de pruebas de MSTG



Nota. Ejemplo de caso de pruebas de guía de MSTG (OWASP, s.f.-b).

3. Luego, para validar cada caso de pruebas, se realizan revisiones de documentación, pruebas estáticas y pruebas dinámicas. Para llevar a cabo la verificación de cada caso de prueba, se utiliza un ambiente de pruebas en el cual se instala y configura diferentes herramientas de *software* de análisis de seguridad para aplicaciones móviles (Ver apartado 5.4.2 Herramientas para las Pruebas).

4. Después de realizar los procedimientos técnicos para cada caso de pruebas, el evaluador establece un resultado en la columna “Estado” de la lista de chequeo de seguridad móvil de OWASP, para ello toma como referencia la escala de la Figura 7.

Figura 7

Escala de evaluación

Leyenda		
Símbolo		**Definición**
Pass		Requerimiento es aplicable a la Aplicación y se ha implementado de acuerdo con a las buenas prácticas
Fail		El requisito se aplicó a la aplicación pero no ha sido satisfactorio.
N/A		El requisito no es aplicable a la aplicación móvil

Nota. Tomado de lista de chequeo del estándar MASVS (OWASP, 2020a).

5. Finalmente, para sustentar el resultado asignado en el paso anterior, por cada requerimiento evaluado, se documentan los hallazgos encontrados utilizando la plantilla de elaboración propia que se encuentra en el Anexo 3. Esta plantilla está basada en la forma como se reporta un hallazgo de auditoría, aprovechando los atributos de condición, criterio, efecto y causa.

Documentación.

En esta fase se documentan y analizan los resultados obtenidos después de evaluar el aplicativo móvil SICAD, para, finalmente, obtener el nivel de confianza de seguridad de la aplicación evaluada.

Capítulo 5. Desarrollo de la solución

5.1 Preparación

En esta fase, después de realizar la selección de los requerimientos por evaluar, se define como alcance 38 requerimientos del nivel MASVS-L1 y 55 requerimientos del nivel MASVS-L2. Es necesario aclarar que los mismos 38 requerimientos del nivel L1 se repiten dentro del nivel L2.

Se evalúan requerimientos de seguridad de nivel MASVS-L1, teniendo en cuenta que es la recomendación del estándar MASVS, debido a que todas las aplicaciones móviles deben cumplir con las mejores prácticas de seguridad y no sufrir de las vulnerabilidades más comunes. Además, se consideran requerimientos de seguridad de nivel MASVS-L2, ya que la aplicación móvil SICAD gestiona información sensible que puede ser utilizada para el robo de identidad o una variedad de esquemas de fraude.

Fuera del alcance queda el nivel MASVS-R o nivel de resistencia contra la ingeniería inversa y la manipulación, debido que estos requerimientos son recomendados para aquellas aplicaciones que manejan datos altamente confidenciales, requieren proteger la propiedad intelectual o dificultar la manipulación de una aplicación.

Los requerimientos de seguridad MASVS-L1 y MASVS-L2 evaluados son agrupados en siete categorías basadas en el objetivo o alcance técnico, detalladas a continuación:

MASVS-V1. Arquitectura, Diseño y Modelado de Amenazas

Se evalúan 12 requerimientos de esta categoría, como se observa en la Figura 8.

Figura 8

Requerimientos de seguridad de MASVS-V1

ID	MSTG-ID	Verificación detallada de requerimientos	Nivel 1	Nivel 2
V1		**Arquitectura, Diseño y Modelado de Amenazas**		
1.1	MSTG-ARCH-1	Todos los componentes se encuentran identificados y asegurar que son necesarios.	✓	✓
1.2	MSTG-ARCH-2	Los controles de seguridad nunca se aplican sólo en el cliente, sino que también en los respectivos servidores.	✓	✓
1.3	MSTG-ARCH-3	Se definió una arquitectura de alto nivel para la aplicación y los servicios y se incluyeron controles de seguridad en la misma.	✓	✓
1.4	MSTG-ARCH-4	Se identificó claramente la información considerada sensible en el contexto de la aplicación móvil.	✓	✓
1.5	MSTG-ARCH-5	Todos los componentes de la aplicación están definidos en términos de la lógica de negocio o las funciones de seguridad que proveen.		✓
1.6	MSTG-ARCH-6	Se realizó un modelado de amenazas para la aplicación móvil y los servicios en el que se definieron las mismas y sus contramedidas.		✓
1.7	MSTG-ARCH-7	Todos los controles de seguridad poseen una implementación centralizada.		✓
1.8	MSTG-ARCH-8	Existe una política explícita sobre el uso de claves criptográficas (si se usan) a través de todo su ciclo de vida. Idealmente siguiendo un estándar de gestión de claves como el NIST SP 800-57.		✓
1.9	MSTG-ARCH-9	Existe un mecanismo para forzar las actualizaciones de la aplicación móvil.		✓
1.10	MSTG-ARCH-10	La implementación de medidas de seguridad es una parte esencial durante todo el ciclo de vida del desarrollo de software de la aplicación.		✓
1.11	MSTG-ARCH-11	Existe una política de divulgación responsable y es llevada a cabo adecuadamente.		✓
1.12	MSTG-ARCH-12	La aplicación debería de cumplir con las leyes y regulaciones de privacidad.	✓	✓

Nota. Tomado de lista de chequeo del estándar MASVS (OWASP, 2020a).

MASVS-V2. Almacenamiento de Datos y la Privacidad

En esta categoría se evalúan 14 requerimientos (Figura 9).

Figura 9

Requerimientos de seguridad de MASVS-V2

ID	MSTG-ID	Verificación detallada de requerimientos	Nivel 1	Nivel 2
V2		**Almacenamiento de datos y la Privacidad**		
2.1	MSTG-STORAGE-1	Las funcionalidades de almacenamiento de credenciales del sistema deben de ser utilizadas para almacenar información sensible, tal como información personal, credenciales de usuario o claves criptográficas.	✓	✓
2.2	MSTG-STORAGE-2	No se debe almacenar información sensible fuera del contenedor de la aplicación o del almacenamiento de credenciales del sistema.	✓	✓
2.3	MSTG-STORAGE-3	No se escribe información sensible en los registros (logs) de la aplicación.	✓	✓
2.5	MSTG-STORAGE-5	Se desactiva la caché del teclado en los campos de texto que contienen información sensible.	✓	✓
2.6	MSTG-STORAGE-6	No se expone información sensible mediante mecanismos de comunicación entre procesos (IPC).	✓	✓
2.7	MSTG-STORAGE-7	No se expone información sensible como contraseñas y números de tarjetas de crédito a través de la interfaz o capturas de pantalla.	✓	✓
2.8	MSTG-STORAGE-8	No se incluye información sensible en las copias de seguridad generadas por el sistema operativo.		✓
2.9	MSTG-STORAGE-9	La aplicación elimina toda información sensible de la vista cuando la aplicación pasa a un segundo plano.		✓
2.10	MSTG-STORAGE-10	La aplicación no conserva ninguna información sensible en memoria más de lo necesario y la memoria se limpia trás su uso.		✓
2.11	MSTG-STORAGE-11	La aplicación obliga a que exista una política mínima de seguridad en el dispositivo, como que el usuario deba configurar un código de acceso.		✓
2.12	MSTG-STORAGE-12	La aplicación educa al usuario acerca de los tipos de información personal que procesa y de las mejores prácticas en seguridad que el usuario debería seguir al utilizar la aplicación.		✓
2.13	MSTG-STORAGE-13	No se guarda ningún tipo de información sensible de forma local en el dispositivo móvil. En su lugar, esa información debería ser obtenida desde un sistema remoto sólo cuando es necesario y únicamente residir en memoria.		✓
2.14	MSTG-STORAGE-14	En caso de ser necesario guardar información sensible de forma local, ésta debe de ser cifrada usando una clave derivada del hardware de almacenamiento seguro, el cual debe requerir autenticación previa.		✓
2.15	MSTG-STORAGE-15	El almacenamiento local de la aplicación debe de ser borrado trás un número excesivo de intentos fallidos de autenticación.		✓

Nota. Tomado de lista de chequeo del estándar MASVS *(OWASP, 2020a)*.

MASVS-V3. Criptografía

Como se observa en la Figura 10, se evalúan seis requerimientos en esta categoría.

Figura 10

Requerimientos de seguridad de MASVS-V3

ID	MSTG-ID	Verificación detallada de requerimientos	Nivel 1	Nivel 2
V3		Criptografía		
3.1	MSTG-CRYPTO-1	La aplicación no depende únicamente de criptografía simétrica cuyas claves se encuentran directamente en el código fuente de la misma.	✓	✓
3.2	MSTG-CRYPTO-2	La aplicación utiliza implementaciones de criptografía probadas.	✓	✓
3.3	MSTG-CRYPTO-3	La aplicación utiliza primitivas de seguridad que son apropiadas para el caso particular y su configuración y parámetros siguen las mejores prácticas de la industria.	✓	✓
3.4	MSTG-CRYPTO-4	La aplicación no utiliza protocolos o algoritmos criptográficos ampliamente considerados obsoletos para su uso en seguridad.	✓	✓
3.5	MSTG-CRYPTO-5	La aplicación no reutiliza una misma clave criptográfica para varios propósitos.	✓	✓
3.6	MSTG-CRYPTO-6	Los valores aleatorios son generados utilizando un generador de números aleatorios suficientemente seguro.	✓	✓

Nota. Tomado de lista de chequeo del estándar MASVS *(OWASP, 2020a)*.

MASVS-V4. Autenticación y Manejo de Sesiones

Se evalúan nueve requerimientos en esta categoría (Figura 11).

Figura 11

Requerimientos de seguridad de MASVS-V4

ID	MSTG-ID	Verificación detallada de requerimientos	Nivel 1	Nivel 2
V4		Autenticación y Manejo de Sesiones		
4.1	MSTG-AUTH-1	Si la aplicación provee acceso a un servicio remoto, un mecanismo aceptable de autenticación como usuario y contraseña es realizado en el servidor remoto.	✓	✓
4.2	MSTG-AUTH-2	Si se utiliza la gestión de sesión por estado, el servidor remoto usa tokens de acceso aleatorios para autenticar los pedidos del cliente sin requerir el envío de las credenciales del usuario en cada uno.	✓	✓
4.3	MSTG-AUTH-3	Si se utiliza la autenticación basada en tokens sin estado, el servidor proporciona un token que se ha firmado utilizando un algoritmo seguro.	✓	✓
4.4	MSTG-AUTH-4	Cuando el usuario cierra sesión se termina la sesión también en el servidor.	✓	✓
4.5	MSTG-AUTH-5	Existe una política de contraseñas y es aplicada en el servidor.	✓	✓
4.6	MSTG-AUTH-6	El servidor implementa mecanismos, cuando credenciales de autenticación son ingresadas una cantidad excesiva de veces.	✓	✓
4.7	MSTG-AUTH-7	Las sesiones y los tokens de acceso expiran luego de un tiempo predefinido de inactividad.	✓	✓
4.11	MSTG-AUTH-11	La aplicación informa al usuario acerca de todas las actividades sensibles en su cuenta. El usuario es capaz de ver una lista de los dispositivos conectados, información contextual (dirección IP, localización, etc.), y es capaz de bloquear ciertos dispositivos.		✓
4.12	MSTG-AUTH-12	Los modelos de autorización deberían de ser definidos e impuestos por el sistema remoto.	✓	✓

Nota. Tomado de lista de chequeo del estándar MASVS *(OWASP, 2020a)*.

MASVS-V5. Comunicación a través de la Red

En la Figura 12, se muestran los cuatro requerimientos evaluados en esta categoría.

Figura 12

Requerimientos de seguridad de MASVS-V5

ID	MSTG-ID	Verificación detallada de requerimientos	Nivel 1	Nivel 2
V5		**Comunicación a través de la red**		
5.1	MSTG-NETWORK-1	La información es enviada cifrada utilizando TLS. El canal seguro es usado consistentemente en la aplicación.	✓	✓
5.2	MSTG-NETWORK-2	Las configuraciones del protocolo TLS siguen las mejores prácticas de la industria, o lo hacen lo mejor posible en caso de que el sistema operativo del dispositivo no soporte los estándares recomendados.	✓	✓
5.3	MSTG-NETWORK-3	La aplicación verifica el certificado X.509 del sistema remoto al establecer el canal seguro y sólo se aceptan certificados firmados por una CA de confianza.	✓	✓
5.6	MSTG-NETWORK-6	La aplicación sólo depende de bibliotecas de conectividad y seguridad actualizadas.		✓

Nota. Tomado de lista de chequeo del estándar MASVS *(OWASP, 2020a)*.

MASVS-V6. Interacción con la Plataforma

Se evalúan cuatro requerimientos en esta categoría (Figura 13).

Figura 13

Requerimientos de seguridad de MASVS-V6.

ID	MSTG-ID	Verificación detallada de requerimientos	Nivel 1	Nivel 2
V6		**Interacción con la Plataforma**		
6.1	MSTG-PLATFORM-1	La aplicación requiere la cantidad de permisos mínimamente necesaria.	✓	✓
6.2	MSTG-PLATFORM-2	Todo dato ingresado por el usuario o cualquier fuente externa debe ser validado y, si es necesario, saneado. Esto incluye información recibida por la UI o mecanismos IPC como los Intents, URLs y datos provenientes de la red.	✓	✓
6.4	MSTG-PLATFORM-4	La aplicación no expone ninguna funcionalidad sensible a través de mecanismos IPC salvo que dichos mecanismos estén debidamente protegidos.	✓	✓
6.8	MSTG-PLATFORM-8	La serialización de objetos, si se realiza, debe implementarse utilizando API seguras.	✓	✓

Nota. Tomado de lista de chequeo del estándar MASVS *(OWASP, 2020a)*.

MASVS-V7. Calidad de Código y Configuración del Compilador

La Figura 14 muestra los seis requerimientos evaluados en esta categoría.

Figura 14

Requerimientos de seguridad de MASVS-V7

ID	MSTG-ID	Verificación detallada de requerimientos	Nivel 1	Nivel 2
V7		Calidad de Código y Configuración del Compilador		
7.1	MSTG-CODE-1	La aplicación es firmada y provista con un certificado válido, cuya clave privada está debidamente protegida.	✓	✓
7.2	MSTG-CODE-2	La aplicación fue publicada en modo release y con las configuraciones apropiadas para el mismo (por ejemplo, non-debuggable).	✓	✓
7.4	MSTG-CODE-4	Cualquier código de depuración y/o de asistencia al desarrollador (p. ej. código de test, backdoors, configuraciones ocultas) debe ser eliminado. La aplicación no hace logs detallados de errores ni de mensajes de depuración.	✓	✓
7.5	MSTG-CODE-5	Todos los componentes de terceros se encuentran identificados y revisados en cuanto a vulnerabilidades conocidas.	✓	✓
7.6	MSTG-CODE-6	La aplicación captura y gestiona debidamente las posibles excepciones.	✓	✓
7.9	MSTG-CODE-9	Las funcionalidades de seguridad gratuitas de las herramientas, tales como minificación del byte-code, protección de la pila, soporte PIE y conteo automático de referencias, se encuentran activadas.	✓	✓

Nota. Tomado de lista de chequeo del estándar MASVS *(OWASP, 2020a)*.

En esta fase también debe conocerse la clasificación de datos donde, claramente, se define la información sensible que la aplicación móvil gestiona, para lo cual se verifica la documentación relacionada a la aplicación móvil y se consulta a los responsables del proyecto sobre la existencia de una política de clasificación de datos o similar. Posteriormente, se determina que no existe dicha clasificación, por lo cual para esta evaluación, se partirá de la definición de datos sensibles que brinda la guía MSTG, la cual indica:

> Cuando no haya una política de clasificación de datos disponible, utilice la siguiente lista de información que generalmente se considera sensible:

- Información de autenticación de usuario (credenciales, PIN, etc.).
- Información de identificación personal (PII) de la que se puede abusar para el robo de identidad: números de seguro social, tarjeta de crédito, números de cuentas bancarias, información médica.

- Identificadores de dispositivos que pueden identificar a una persona.
- Datos altamente sensibles cuyo compromiso podría causar daños a la reputación y / o costos financieros.
- Cualquier dato cuya protección sea una obligación legal.
- Cualquier dato técnico generado por la aplicación (o sus sistemas relacionados) y utilizado para proteger otros datos o el sistema en sí (por ejemplo, claves de cifrado) (OWASP, s.f.-g).

Para realizar la evaluación a la aplicación móvil SICAD, se emitió un permiso por parte de la coordinación de la Unidad Técnica de Sistemas de Informática (UTSI) del INEC (Anexo 1) para liberar de cualquier responsabilidad legal al evaluador durante la realización del análisis de seguridad.

5.2 Recolección de información

5.2.1 Documentación recolectada

En esta fase de recolección de datos, se recopila documentación o insumos relacionados a la aplicación móvil SICAD, que ayudan a comprender el contexto ambiental, el cual incluye los objetivos de la organización para la aplicación, la funcionalidad, los interesados y responsables de la aplicación. La documentación proporcionada por el INEC que está relacionada a la aplicación móvil SICAD se enlista en la Tabla 11.

Tabla 11

Documentación relacionada a la aplicación móvil SICAD

Documento	**Propósito**
Reglas Generales ETAPA actualizado.docx	Este documento es una recopilación de las necesidades funcionales de los interesados del proyecto estadístico de la encuesta ENA del SICAD. Dicho sistema brinda soporte operativo al proyecto estadístico en campo, este aplicativo es utilizado por entrevistadores para recopilar información sobre la producción agrícola, forestal, pecuaria y de flores de Costa Rica.
Presentación Capacitación enero 2019 ENA.pptx	Este documento educa al personal de campo (entrevistadores) sobre las funcionalidades y uso del SICAD.
Diagrama de procesos ENA.pdf	Este documento ilustra el diagrama de los procesos operativos de la encuesta ENA.
BoletaAnualENA 5-2018.pdf	Este documento se refiere al cuestionario de preguntas de la encuesta ENA.
SICAD_ENA.zip	Este archivo contiene el código fuente de la aplicación móvil SICAD.
SICAD_ENA_10.0.30_Beta (PRUEBAS).apk	Este archivo es el instalador del aplicativo móvil en su versión de pruebas.
BaseDatos.db	Estos archivos de las bases de datos de SQLite son utilizados por el aplicativo móvil.

Nota. Esta lista de documentos fue proporcionada por el INEC. Elaboración propia.

También se recopila la siguiente documentación que se relaciona con el desarrollo de *software* y seguridad de la información utilizada en el INEC; se enlista en la Tabla 12.

Tabla 12

Documentación relacionada al desarrollo de software y seguridad en el INEC

Documento	**Propósito**
Metodología de Desarrollo de Sistemas	Esta documentación recopila la metodología de desarrollo de sistemas utilizada en la UTSI.
Estándar de desarrollo para Android	Este documento recopila los estándares de desarrollo de sistemas en Android utilizada en la UTSI.
Marco de seguridad de la información para la Gestión de las Tecnologías de Información y Comunicaciones (TICS)	Este documento recopila las disposiciones de seguridad de la información utilizada en el INEC

Nota. Esta lista de documentos fue proporcionada por el INEC. Elaboración propia.

5.2.2 Información de la aplicación

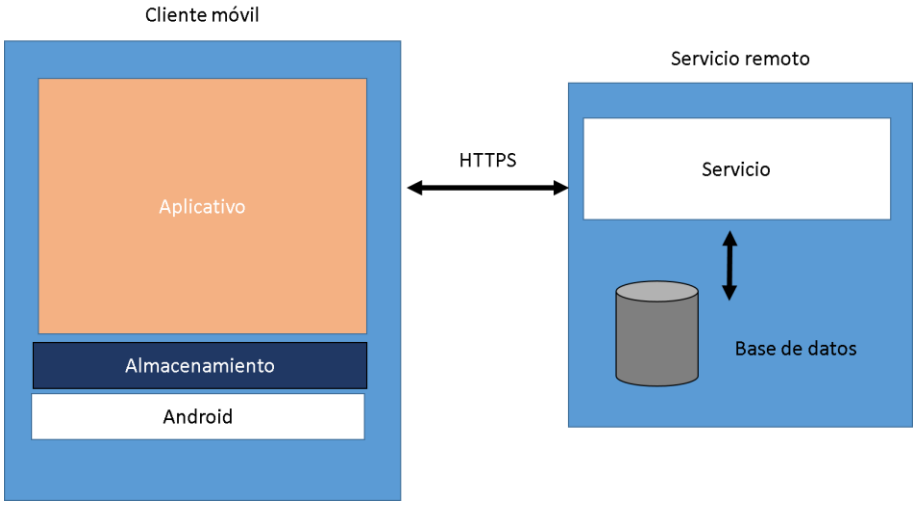
La aplicación seleccionada es SICAD. Es una aplicación móvil de recolección de datos por medio de un dispositivo móvil para el sistema operativo Android, la cual permite a un entrevistador realizar una entrevista a los productores de Costa Rica sobre la producción agrícola, forestal, pecuaria y de flores.

La aplicación SICAD cuenta con las siguientes funcionalidades: sistema de autenticación; recopilación y respaldo de respuestas; gestión de carga de trabajo del entrevistador y sincronización de datos. En esta fase se recopila la siguiente información relacionada a la aplicación móvil del SICAD, que ayuda a comprender el contexto arquitectónico de la aplicación.

- El lenguaje nativo del aplicativo SICAD es en Java.
- La Figura 15 es una representación gráfica general de la arquitectura de la aplicación móvil SICAD.

Figura 15

Arquitectura general del aplicativo SICAD



Nota. Elaboración propia.

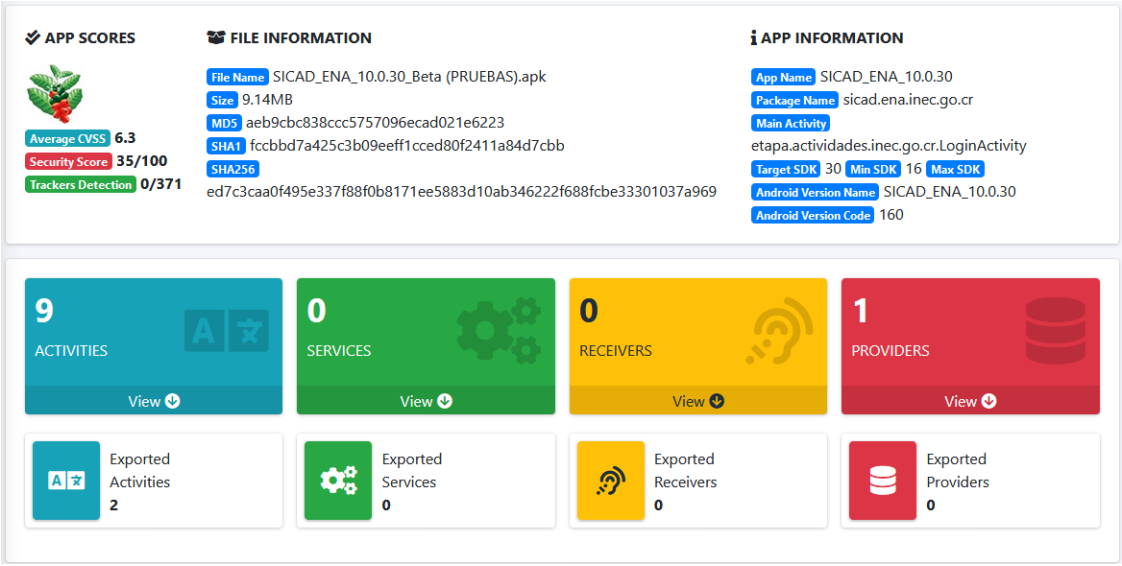
- La aplicación interactúa con servicios de PlayStore para obtener actualizaciones.
- La aplicación sincroniza los datos por medio de un API de RESTful.
- La aplicación utiliza el almacenamiento local de bases de datos SQLite.
- La aplicación puede instalarse desde el API 16 (conocido como Jelly Bean 4.1.x) hasta el API 31 (conocido como Android11) (Google, 2021).

Con el objetivo de recopilar la mayor cantidad de información general de la aplicación SICAD, se realiza un análisis automático utilizando la herramienta local Mobile Security Framework (MobSF), cuyos resultados se presentan a continuación.

La Figura 16 brinda información general del aplicativo móvil SICAD.

Figura 16.

Información general del SICAD del análisis con MobSF



Nota. Captura de pantalla de los resultados de la herramienta MobSF.

La Figura 16 muestra información sobre el archivo (.apk) del SICAD como su nombre, tamaño y el valor hash; también muestra información general del aplicativo (*app*) del SICAD como su nombre, *activity* principal, versión, entre otras características. En la figura se puede apreciar que la aplicación contiene nueve *activities* y un *providers*.

La Figura 17 enlista las nueve pantallas conocidas como *activities* pertenecientes a la aplicación móvil SICAD.

Figura 17

Lista de pantallas de SICAD del análisis con MobSF



Nota. Captura de pantalla de los resultados de la herramienta MobSF.

A continuación, se enlista los permisos requeridos por la aplicación móvil SICAD:

- Lectura y escritura de almacenamiento externo (READ_EXTERNAL_STORAGE y WRITE_EXTERNAL_STORAGE).
- Acceso al estado de la red (ACCESS_NETWORK_STATE).
- Permiso de Internet (INTERNET).
- Lectura del estado del teléfono (READ_PHONE_STATE).
- Permiso de control de bloqueo o suspensión (WAKE_LOCK).

5.3 Mapeo de aplicación

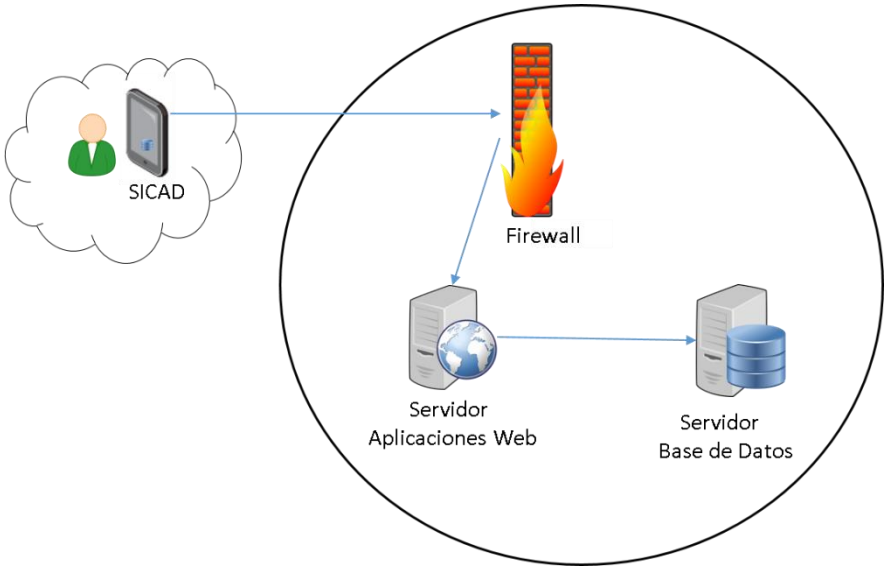
En esta etapa, se analiza la información recolectada de las fases anteriores en conjunto con hallazgos del escaneo automatizado, para así comprender mejor la aplicación móvil SICAD. Primeramente, la aplicación móvil utiliza como entradas de datos los campos de texto y sus salidas son archivos de logs, bases de datos SQLite; también existe una interacción con un API RESTful para obtener y transferir información.

Por otro lado, la aplicación móvil requiere una tarjeta de memoria microSD, para almacenar los respaldos de base de datos SQLite y archivos de logs de errores; también requiere conexión a internet para sincronización de datos.

La Figura 18 es una representación gráfica general del flujo de datos donde interviene la aplicación móvil SICAD.

Figura 18

Flujos de datos del aplicativo móvil SICAD



Nota. Elaboración propia.

La aplicación móvil SICAD utiliza librerías y componentes de terceros. Sobre estas se realiza una búsqueda de vulnerabilidades utilizando la página web especializada en el tema CVE Details (CVE Details, s.f-a); se determina que las librerías de terceros, enlistadas en la Tabla 13, son vulnerables.

Tabla 13

Librerías de terceros vulnerables

Librería	Vulnerabilidad
guava-15.0.jar	CVE-2018-10237 [a]
com.squareup.retrofit2:retrofit:2.0.2	CVE-2018-1000850 [b]

Nota. [a] (CVE Details, 2018b), [b] (CVE Details, 2018a). Elaboración propia.

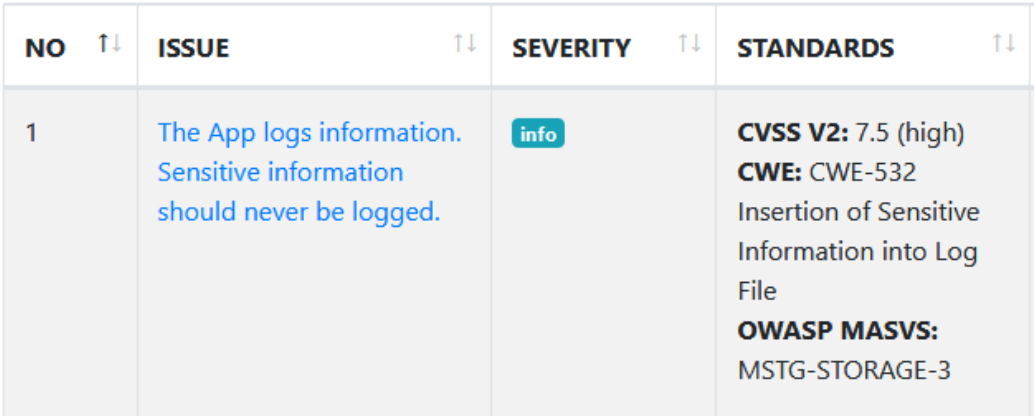
Dentro de esta fase, se realiza un diagnóstico utilizando la herramienta local MobSF. De este diagnóstico, se obtienen los siguientes hallazgos, los cuales se consideran posibles vulnerabilidades o fallos de seguridad; se analizarán con mayor detalle en la siguiente fase.

Hallazgo 1

Destaca el posible fallo de seguridad de fuga de información sensible a través de los logs de errores, como se evidencia en la Figura 19.

Figura 19

Hallazgo de seguridad #1 con la herramienta MobSF



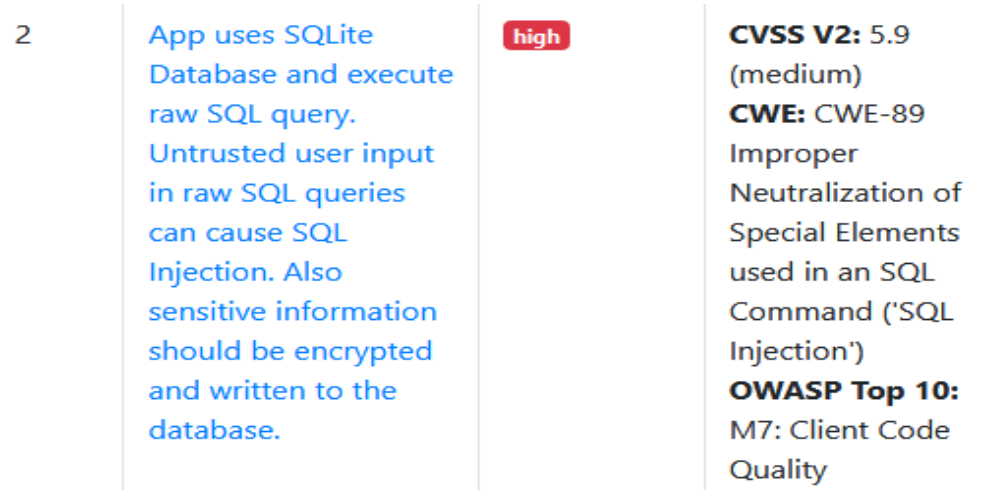
Nota. Captura de pantalla de los resultados de la herramienta MobSF.

Hallazgo 2

Destaca la posible vulnerabilidad de ataque de inyección de comandos SQL a través de los componentes gráficos de la aplicación móvil (Figura 20).

Figura 20

Hallazgo de seguridad #2 con la herramienta MobSF



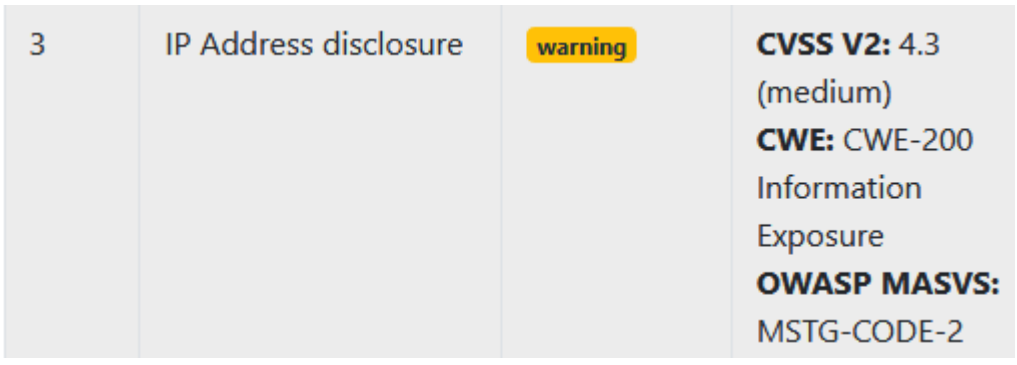
Nota. Captura de pantalla de los resultados de la herramienta MobSF.

Hallazgo 3

Destaca la posible exposición de datos de configuración de servidor en el código fuente de la aplicación móvil, como se muestra en la Figura 21.

Figura 21

Hallazgo de seguridad #3 con la herramienta MobSF



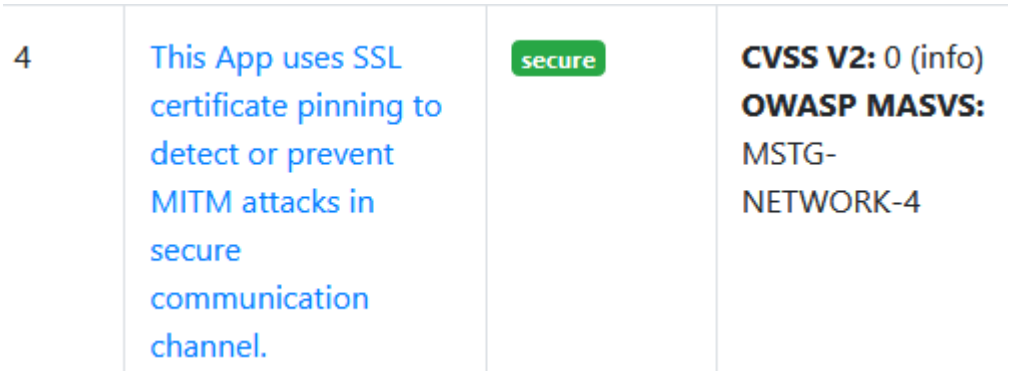
Nota. Captura de pantalla de los resultados de la herramienta MobSF.

Hallazgo 4

La Figura 22 destaca que la aplicación móvil SICAD previene y detecta el ataque de SSL certificate pinning.

Figura 22

Hallazgo de seguridad #4 con la herramienta MobSF



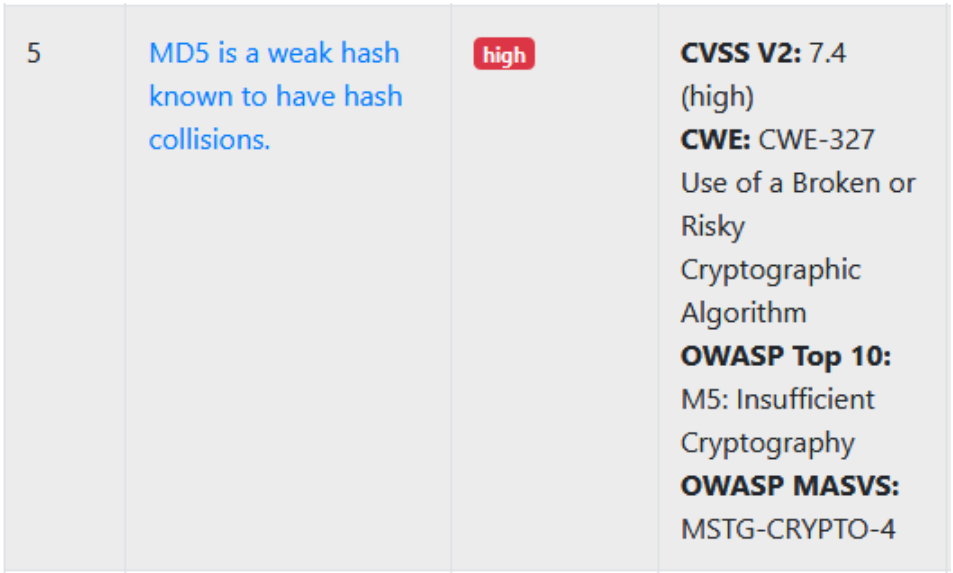
Nota. Captura de pantalla de los resultados de la herramienta MobSF.

Hallazgo 5

Destaca que la aplicación utiliza el algoritmo hash MD5 que se considera vulnerable (Figura 23).

Figura 23

Hallazgo de seguridad #5 con la herramienta MobSF



Nota. Captura de pantalla de los resultados de la herramienta MobSF.

Hallazgo 6

Destaca que la aplicación tiene habilitada la funcionalidad de escribir y leer en el almacenamiento externo, lo que se considera como un posible fallo de seguridad; como se evidencia en la Figura 24.

Figura 24

Hallazgo de seguridad #6 con la herramienta MobSF

6	App can read/write to External Storage. Any App can read data written to External Storage.	high	**CVSS V2:** 5.5 (medium) **CWE:** CWE-276 Incorrect Default Permissions **OWASP Top 10:** M2: Insecure Data Storage **OWASP MASVS:** MSTG-STORAGE-2

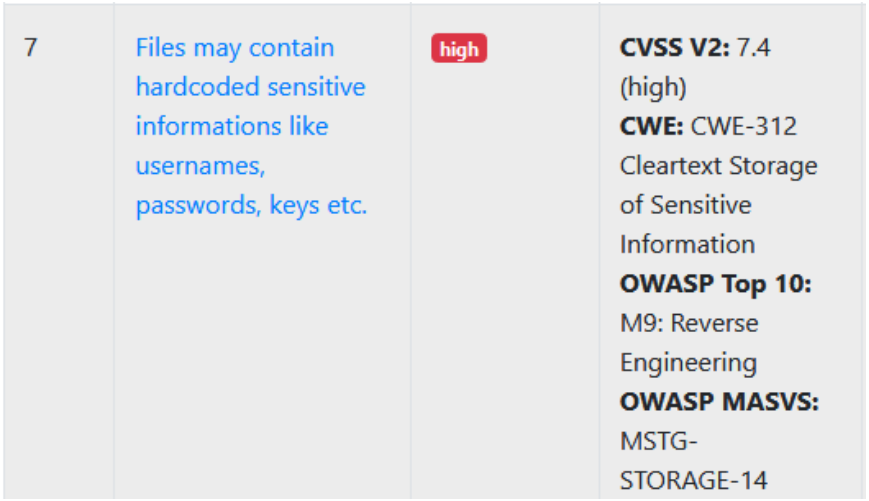
Nota. Captura de pantalla de los resultados de la herramienta MobSF.

Hallazgo 7

La Figura 25 evidencia la posibilidad de que la aplicación contiene información sensible o de configuración “hardcodeada” en el código fuente.

Figura 25

Hallazgo de seguridad #7 con la herramienta MobSF



Nota. Captura de pantalla de los resultados la herramienta MobSF.

5.4 Evaluación

En esta fase se realiza la revisión de 38 requerimientos del nivel MASVS-L1 y 55 requerimientos del nivel MASVS-L2. Es necesario aclarar que los mismos 38 requerimientos del nivel L1 se repiten dentro del nivel L2.

Para evaluar cada requerimiento, se toma como referencia los casos de pruebas de la guía MSTG. Estos casos de pruebas comprenden diferentes técnicas de evaluación que incluyen la revisión de documentación, de código fuente y ejecución de la aplicación.

Para la evaluación se cuenta con el aplicativo móvil (.apk), las bases de datos de SQLite necesarias para el funcionamiento del aplicativo móvil, el acceso al API Restful y el código fuente del aplicativo.

Esta etapa se desarrolla en un ambiente controlado, el aplicativo se ejecuta en un dispositivo virtual y el API Restful está disponible en un servidor de pruebas; se utilizan datos de pruebas para verificar la aplicación.

5.4.1 Ambiente de pruebas

Para realizar la revisión de código fuente y ejecución del aplicativo móvil SICAD, se establece el siguiente ambiente de pruebas: un computador con Windows 10 y un dispositivo virtual con el sistema operativo Android.

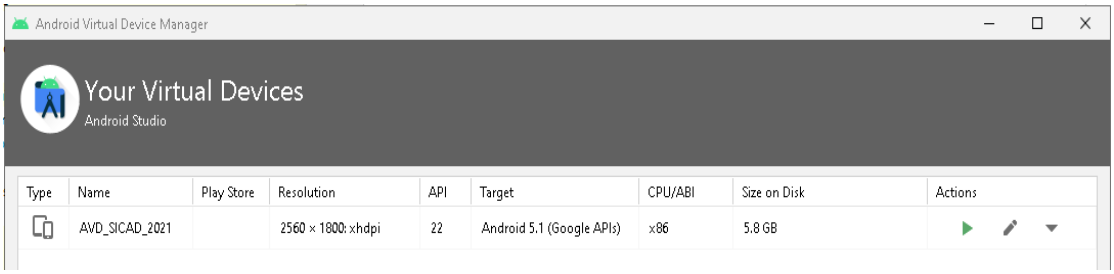
A continuación, se describe la configuración de la computadora para las pruebas:

- Nombre de máquina: Análisis
- Sistema Operativo: Windows 10 (64-bit)
- Memoria: 8GB
- Tamaño del disco: 1 TB.
- Habilita conexión de internet.

A continuación, se describe la configuración del dispositivo virtual con el fin de realizar las pruebas dinámicas, se utiliza la herramienta Android Virtual Device (AVD) para crear el dispositivo, como se demuestra en la Figura 26:

Figura 26

Configuración del dispositivo virtual



Nota. Tomada de la creación de dispositivo virtual. Elaboración propia.

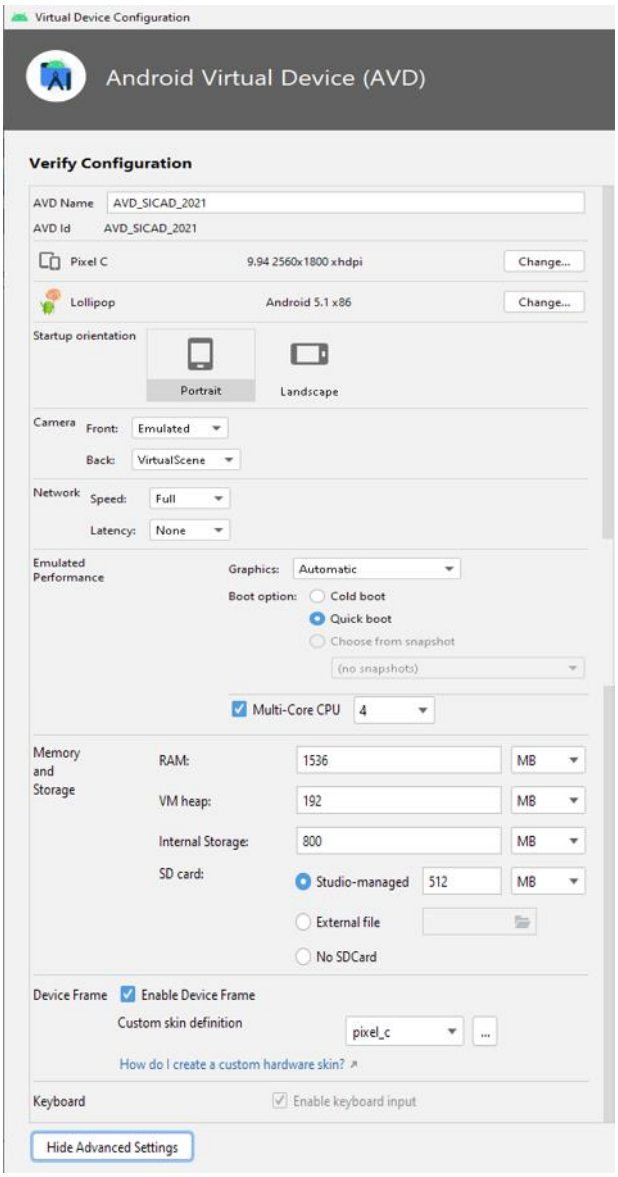
Además, el emulador se configura de la siguiente forma:

- Nombre de máquina: AVD_SICAD_2021
- Sistema Operativo: Android 5.1 (Google APIs) “Lollipop”
- Versión de API:22
- Tamaño del disco: 5.8 GB
- Memoria RAM: 1536 MB
- SD Card: 512 MB
- Habilitada la conexión de internet.

La Figura 27 demuestra el detalle de las configuraciones realizadas en el dispositivo virtual.

Figura 27

Configuración específica del dispositivo virtual.

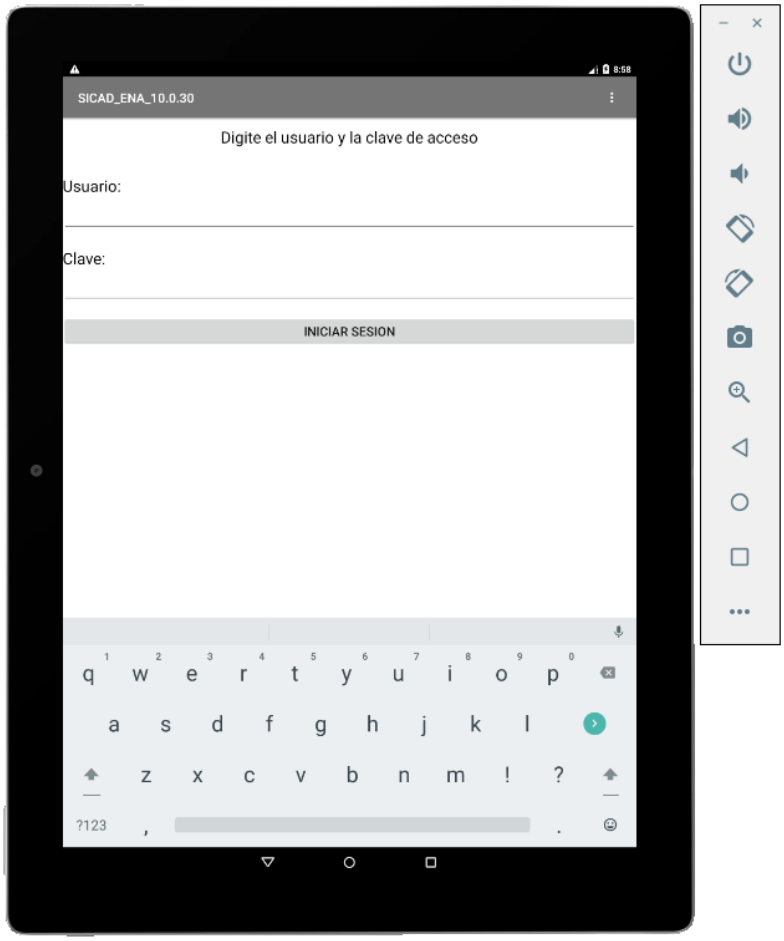


Nota. Elaboración propia.

En la Figura 28, se puede apreciar funcionando el dispositivo virtual con el aplicativo móvil SICAD.

Figura 28

SICAD ejecutando en el dispositivo virtual



Nota. Elaboración propia.

5.4.2 Herramientas para las pruebas

A continuación, se enumeran las herramientas instaladas y configuradas para la revisión de código fuente y ejecución de la aplicación.

1. **Android Studio.** Es utilizada para realizar las pruebas estáticas con el objetivo de revisar el código fuente de la aplicación móvil (GoogleDevelopers, s.f).
2. **Mobile Security Framework (MobSF).** Se utiliza para realizar las pruebas estáticas automáticas a la aplicación móvil (Abraham, s.f).
3. **DB Browser (SQLite).** Empleada para realizar las pruebas dinámicas con el objetivo de revisar las bases de datos de SQLite (DB4S, s.f).

4. **Android Debug Bridge (ADB).** Utilizada para realizar las pruebas dinámicas con el objetivo de interactuar con el emulador de Android y la aplicación móvil (GoogleDevelopers, 2021a).
5. **Android Emulator.** Se emplea para realizar pruebas dinámicas con el objetivo de ejecutar la aplicación móvil en un ambiente controlado (GoogleDevelopers, 2020c).
6. **Android Virtual Device (AVD).** Se usa para realizar pruebas dinámicas con el objetivo de gestionar los dispositivos virtuales donde se ejecutará el aplicativo móvil (GoogleDevelopers, 2020b).
7. **Drozer.** Utilizada para realizar pruebas dinámicas con el objetivo de comprobar configuraciones de la aplicación móvil (F-Secure, s.f).
8. **Android backup extractor (ABE).** Es utilizada para realizar pruebas dinámicas con el objetivo de extraer un *backup* de Android en el emulador (Elenkov , s.f).
9. **Android Profiler.** Empleada para realizar pruebas dinámicas con el objetivo de visualizar el contenido de la memoria del emulador durante la ejecución del aplicativo móvil (GoogleDevelopers, 2021c).
10. **Burp Suite.** Utilizada para realizar pruebas dinámicas con el objetivo de intersecar las solicitudes y respuestas (http o https) entre el aplicativo móvil y el API Restful alojado en el servidor de aplicaciones (PortSwigger, s.f).
11. **Wireshark.** Se usa para realizar pruebas dinámicas con el objetivo de intersecar el tráfico de red entre el aplicativo móvil y el API Restful alojado en el servidor de aplicaciones (Wireshark Foundation, s.f).
12. **SSL Server Test (Qualys).** Utilizada para realizar pruebas dinámicas con el objetivo de revisar las configuraciones del certificado TLS del servidor de aplicaciones (Qualys, s.f).

13. **mvnrepository.com.** Se utiliza para realizar pruebas dinámicas con el objetivo de revisar las dependencias de librerías de código fuente utilizadas en el aplicativo móvil (Mvnrepository, s.f).
14. **Apksigner.** Utilizadas para realizar pruebas dinámicas con el objetivo de revisar los certificados que firman el aplicativo móvil (GoogleDevelopers, 2021b).
15. **Jarsigner.** Se emplean para realizar pruebas dinámicas con el objetivo de revisar los certificados que firman el aplicativo móvil (Oracle, s.f-a).
16. **dex2jar.** Utilizadas para realizar pruebas dinámicas con el objetivo de convertir el archivo .dex a .java del aplicativo móvil (Pan, Barrios , Schuberth, & Papp , s.f).
17. **Java Descompiler-GUI (jd-gui-1.6.6.jar).** Se usan para realizar pruebas dinámicas con el objetivo de descompilar y analizar el “código de bytes” de Java del aplicativo móvil (Java Decompiler Project, s.f).

5.4.3 Hallazgos

Para el lector de este documento, se explican únicamente dos ejemplos de hallazgos encontrados del análisis del aplicativo móvil SICAD, debido a que se debe respetar las cláusulas del contrato de confidencialidad estipulado en el INEC. Sin embargo, al Departamento de Desarrollo de Sistemas del INEC, se le entregará un documento digital con la totalidad de los hallazgos.

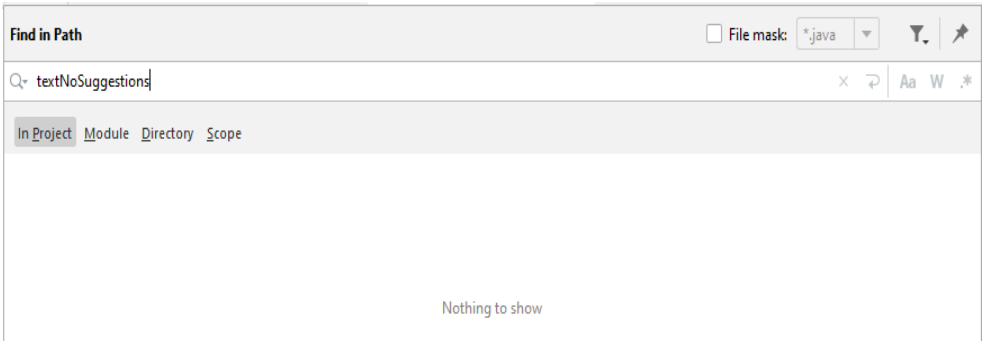
Los hallazgos se presentan por medio de una plantilla de elaboración propia que se detalla en el Anexo 3. Primeramente, se expone el hallazgo del requerimiento MASVS-V2.5, el cual fue desaprobado; posteriormente, se presenta el hallazgo del requerimiento MASVS-V2.7, el cual fue aprobado. Para ambos hallazgos sE realizaron pruebas estáticas y pruebas dinámicas con el fin de respaldar el resultado obtenido.

MASVS-V2.5	Se desactiva la caché del teclado en los campos de texto que contienen información sensible.	DESAPROBADO
Análisis estático.		

Condición: como parte de la verificación realizada en la aplicación móvil SICAD, se procedió a aplicar una revisión en el código fuente de la aplicación. Producto de la revisión, no se encontró el atributo “textNoSuggestions” en el código fuente, lo que significa que todos los campos de texto permiten que la caché del teclado se muestre en pantalla, como evidencia se muestra la Figura 29.

Figura 29

Resultado de búsqueda de la configuración “textSuggestions” en código fuente



Nota. Tomada al ejecutar la búsqueda con Android Studio. Elaboración propia.

Criterio: la guía MSTG indica que: “Si el atributo XML android: inputType recibe el valor textNoSuggestions, la caché del teclado no se mostrará cuando se seleccione el campo de entrada” (OWASP, s.f.-p).

Causa: de acuerdo con el análisis realizado, se determinó que, en la documentación detallada en la Tabla 12, no se encuentran disposiciones técnicas que establezcan cómo debe trabajarse la caché del teclado en los desarrollos de aplicaciones móviles.

Efecto: esta situación posibilita obtener información sensible a través de la caché de los campos de texto.

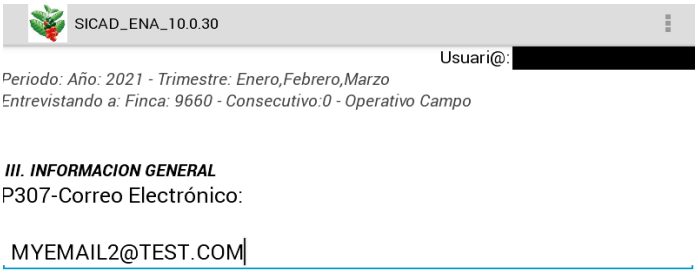
Herramienta(s) utilizada(s) en esta prueba: Android Studio.

Análisis dinámico.

Condición: como parte de la verificación realizada en la aplicación móvil SICAD, para comprobar que se desactive la caché del teclado en los campos de texto, se procedió a instalar y ejecutar la aplicación móvil SICAD. Producto de la revisión, se encontró que el aplicativo permite en los espacios de texto la sugerencia de cadenas, los valores ingresados son aprendidos por el sistema operativo y luego sugeridos en cualquier campo de texto en la misma u otra aplicación, como se muestra en las Figuras 30 y 31.

Figura 30

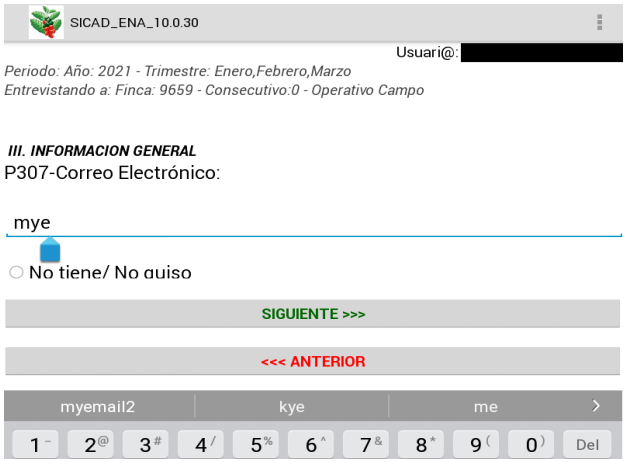
Registro de un correo electrónico en un campo de texto



Nota. Tomada de la ejecución de SICAD. Elaboración propia.

Figura 31

El sistema operativo Android sugiere el correo electrónico previamente escrito en la Figura 30



Nota. Tomada de la ejecución de SICAD. Elaboración propia.

Criterio: la guía MSTG indica: “Inicie la aplicación y haga clic en los campos de entrada que contienen datos confidenciales. Si se sugieren cadenas, la caché del teclado no se ha deshabilitado para estos campos” (OWASP, s.f.-p). **Causa:** de acuerdo con el análisis realizado, se determinó que, en la documentación detallada en la Tabla 12, no se encuentran disposiciones técnicas que establezcan cómo debe trabajarse la caché del teclado en el desarrollo de aplicaciones móviles. **Efecto:** posibilita que un *malware*, otra aplicación o usuario malintencionado obtenga información sensible a través de la caché de los campos de texto de la aplicación móvil. **Herramienta(s) utilizada(s) en esta prueba:** Android emulador.
Recomendaciones: se recomienda desactivar la caché del teclado en todos los campos de texto que capturan información sensible como correos electrónicos.

MASVS-V2.7	No se expone información sensible como contraseñas y números de tarjetas de crédito a través de la interfaz o capturas de pantalla.	APROBADO
Análisis estático.		

Condición: para comprobar que no se expone información sensible a través de la interfaz o captura de pantalla, se procedió a aplicar una revisión en el código fuente de la aplicación. Producto de la revisión, se encontró en el archivo “login.xml”, como se muestra en la Figura 32, la configuración habilitada android: inputType = "textPassword", la cual enmascara el campo de texto de la contraseña cuando el usuario ingresa algún valor.

Figura 32

Configuración referente campos de texto tipo contraseña

```
<EditText
    android:id="@+id/login_password"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:inputType="textPassword"
    android:textSize="25sp" />
```

Nota. Tomada al ejecutar la búsqueda con Android Studio. Elaboración propia.

Criterio: la guía MSTG indica que: “para asegurarse de que una aplicación enmascara la entrada sensible del usuario, verifique el siguiente atributo en la definición de EditText: android: inputType = "textPassword”” (OWASP, s.f.-d)

Causa: se determinó que el desarrollador de esta aplicación aplicó las buenas prácticas de seguridad en la programación de aplicaciones móviles en Android.

Efecto: protección de la información sensible (contraseña del usuario) por medio del enmascarado en el campo de texto.

Herramienta(s) utilizada(s) en esta prueba: Android Studio.

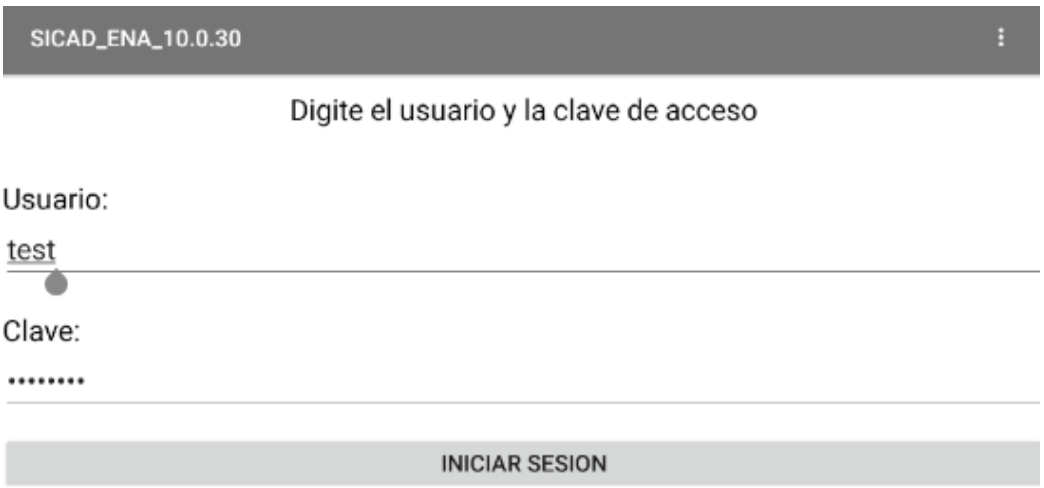
Análisis dinámico.

Condición:

Como parte de la verificación realizada en la aplicación móvil SICAD para comprobar que no se expone información sensible a través de la interfaz o captura de pantalla, se procedió a instalar y ejecutar la aplicación móvil en un dispositivo virtual. Producto de la revisión, se encontró que el aplicativo no expone información confidencial (contraseña), ya que dicha información es enmascarada, es decir, remplazada la entrada por puntos, como se muestra en la Figuras 33.

Figura 33

El campo de texto tipo contraseña enmascara los valores ingresados.



Nota. Tomada de la ejecución de SICAD. Elaboración propia.

Criterio: la guía MSTG indica que: “para determinar si la aplicación filtra información confidencial a la interfaz de usuario, ejecute la aplicación e identifique los componentes que muestran dicha información o la toman como entrada. Si la información está enmascarada, por ejemplo, reemplazando la entrada con asteriscos o puntos, la aplicación no está filtrando datos a la interfaz de usuario” (OWASP, s.f.-d).

Causa: se determinó que el desarrollador de esta aplicación aplicó las buenas prácticas de seguridad en la programación de aplicaciones móviles en Android.

Efecto: protección de la información sensible (contraseña del usuario) por medio del enmascarado en el campo de texto. **Herramienta(s) utilizada(s) en esta prueba:** Android Emulator.
Recomendaciones: se recomienda documentar las buenas prácticas de seguridad para que todos los miembros del Departamento de Desarrollo de Sistemas de Información en la UTSI las conozcan y las pongan en práctica en el desarrollo de aplicaciones móviles.

5.5 Documentación

5.5.1. Resumen ejecutivo

Se realiza una revisión de seguridad para el aplicativo móvil SICAD versión 10.0.30 de la encuesta ENA del INEC, se evalúan 38 requerimientos del nivel MASVS-L1 y 55 requerimientos pertenecientes al nivel MASVS-L2. Es necesario aclarar que los mismos 38 requerimientos del nivel L1 se repiten dentro del nivel L2.

El aplicativo móvil SICAD es utilizado por los entrevistadores en la encuesta ENA para recopilar información sobre la producción agrícola, forestal, pecuaria y de flores de Costa Rica. Este aplicativo móvil utiliza una API RESTful para la sincronización de datos.

Para evaluar el aplicativo móvil SICAD, se toma como referencia el estándar de MASVS versión 1.2 que incluye el estándar MASVS, la guía MSTG y la lista de chequeo.

Se evalúa cada requerimiento definido por MASVS utilizando la lista de chequeo; para verificar el requerimiento, se ejecuta revisión de documentación, pruebas estáticas y dinámicas para cada caso de pruebas de la guía MSTG; luego se documenta y evalúa cada hallazgo.

5.5.2. Resultados por cada categoría evaluada.

Después de ejecutar la verificación de controles de seguridad con la lista de chequeo, esta genera los siguientes resultados por categoría (Tabla 14).

Tabla 14

Resultados de los requerimientos de seguridad por categoría

Categoría	Aprobados	Desaprobados	No Aplica	Porcentaje
MASVS-V1	3	9	0	25,00 %
MASVS-V2	2	12	1	14,29 %
MASVS-V3	0	6	0	0,00 %
MASVS-V4	0	9	3	0,00 %
MASVS-V5	2	2	2	50,00 %
MASVS-V6	3	1	7	75,00 %
MASVS-V7	3	3	3	50,00 %

Nota. Esta tabla muestra la cantidad de requerimientos por categoría que fueron aprobados, desaprobados y no aplicaron, así como el porcentaje de aprobación. Basado en los resultados de la lista de chequeo del estándar MASVS versión 1.2.

Las categorías MASVS-V5, V6 y V7 alcanzaron el mayor porcentaje de cumplimiento en requerimientos de red, interacción con la plataforma y calidad de código fuente. Mientras que las categorías MASVS-V1, V2, V3 y V4 tienen los porcentajes menores de cumplimiento en requerimientos de planificación de la seguridad del desarrollo de aplicaciones, almacenamiento de datos, privacidad, criptografía, autenticación y manejo de sesiones. De esta forma, se revela que estas últimas categorías deben reforzarse en el desarrollo de las aplicaciones móviles.

5.5.3. Resultados por niveles de seguridad de MASVS.

Para la aplicación móvil SICAD, se evaluaron 38 requerimientos del nivel MASVS-L1 y 55 requerimientos del nivel MASVS-L2. Es necesario aclarar que los mismos 38 requerimientos del nivel L1 se repiten dentro del nivel L2. A continuación, se segregan los resultados por el nivel de seguridad:

Tabla 15

Resultados de los requerimientos evaluados del nivel MASVS-L1

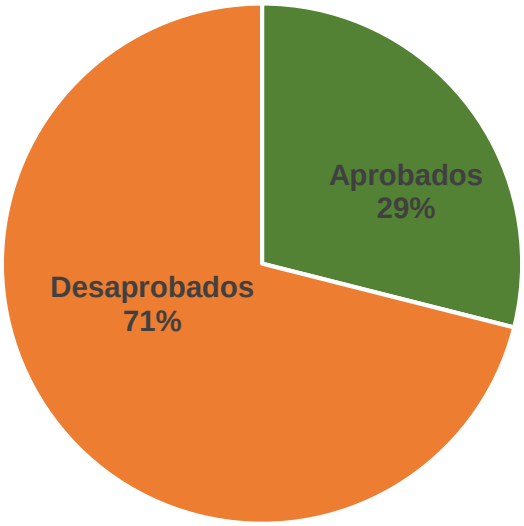
Categoría	Evaluados	Aprobados	Desaprobados	No Aplicados
MASVS-V1	5	1	4	0
MASVS-V2	6	2	4	1
MASVS-V3	6	0	6	0
MASVS-V4	8	0	8	0
MASVS-V5	3	2	1	0
MASVS-V6	4	3	1	4
MASVS-V7	6	3	3	3
Totales	**38**	**11**	**27**	**8**

Nota. Esta tabla muestra la cantidad de requerimientos que fueron evaluados, cuántos aprobaron y desaprobaron, así mismo, cuántos no aplicaron distribuidos por categoría dentro del nivel MASVS-L1. Elaboración propia.

La Tabla 15 indica que se evaluaron un total de 38 requerimientos de seguridad del nivel MASVS-L1, de los cuales solamente 11 requerimientos fueron aprobados y 27 desaprobados, no se aplicaron 8 requerimientos. Estos datos revelan que el porcentaje de cumplimiento de los requisitos del aplicativo móvil SICAD es apenas del 29%, versus un 71% de requisitos desaprobados dentro de los controles que fueron evaluados. En la Figura 34 se grafica el porcentaje de requisitos aprobados y desaprobados del nivel MASVS-L1.

Figura 34

Porcentaje de requerimientos aprobados y desaprobados del Nivel MASVS-L1



Nota. Elaborado de los datos del nivel MARVS-L1

Los resultados de los requerimientos evaluados del nivel MASVS-L2 se demuestran en la Tabla 16.

Tabla 16

Resultados de los requerimientos evaluados del nivel MASVS-L2

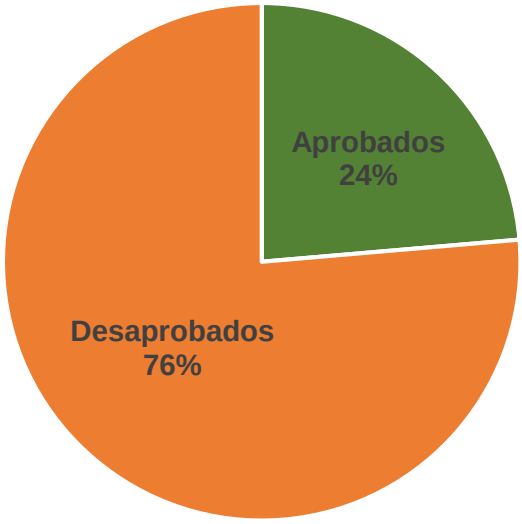
Categoría	Evaluados	Aprobados	Desaprobados	No Aplicados
MASVS-V1	12	3	9	0
MASVS-V2	14	2	12	1
MASVS-V3	6	0	6	0
MASVS-V4	9	0	9	3
MASVS-V5	4	2	2	2
MASVS-V6	4	3	1	7
MASVS-V7	6	3	3	3
Totales	**55**	**13**	**42**	**16**

Nota. Esta tabla muestra la cantidad de requerimientos que fueron evaluados, cuántos aprobaron y desaprobaron, así mismo cuántos no aplicaron, distribuidos por categoría dentro del nivel MASVS-L2. Elaboración propia.

La Tabla 16 indica que se evaluaron 55 requerimientos de seguridad del nivel MASVS-L2, de los cuales solamente 13 requerimientos fueron aprobados y 42 desaprobados. Esto da como resultado que el porcentaje de aprobación sea de 24%, y el de desaprobación de 76%, se muestran en la Figura 35.

Figura 35

Porcentaje de requerimientos aprobados y desaprobados del Nivel MASVS-L2



Nota. Elaborado de los datos del nivel MARVS-L2.

Después de este análisis de seguridad de la aplicación móvil SICAD de la encuesta ENA del INEC, empleando como referencia el Estándar MASVS Versión 1.2, se concluye que la aplicación demuestra un nivel de confianza de seguridad bajo. Se evidencia que se requiere correcciones de los hallazgos encontrados en futuras versiones.

Capítulo 6. Conclusiones y recomendaciones

6.1 Conclusiones

- Se determina que el nivel de confianza de seguridad del aplicativo móvil es bajo, cuenta con un porcentaje de cumplimiento de 29% del nivel MASVS-L1, es decir, la aplicación no alcanza el nivel más básico recomendado por OWASP, por lo cual es prioritario que el Departamento de Desarrollo de Sistemas del INEC valore la corrección de cada hallazgo encontrado.
- Durante el análisis del aplicativo móvil SICAD, se identifica que posee aspectos por mejorar. Se podrá tomar como línea base el estándar MASVS con el fin de definir los criterios de seguridad para el desarrollo y pruebas de las aplicaciones móviles en la plataforma Android.
- Durante la evaluación de seguridad sobre la aplicación móvil SICAD, se evidenció que un enfoque híbrido entre pruebas automáticas y manuales en conjunto con la intuición humana permite realizar un análisis detallado del aplicativo, puesto que, sin duda, el uso de herramientas automáticas como MobSF reducen el tiempo del análisis, con el inconveniente de que estas herramientas solo encontrarán lo que han definido previamente por sus creadores.
- El estándar MASVS propone una metodología para la verificación de seguridad de aplicaciones móviles que es adaptable a las necesidades de seguridad y la tolerancia al riesgo de una organización.
- Se resalta la importancia de contar con la documentación de negocio, técnica y código fuente del aplicativo analizado, esta información fue brindada por la institución y permitió una mayor comprensión y cobertura en el análisis de seguridad de una aplicación móvil.
- Al utilizar una plantilla personalizada para documentar los hallazgos encontrados durante el análisis de seguridad del aplicativo, facilitó la explicación de estos; ya que

en la lista de chequeo del estándar MASVS, el espacio destinado para observaciones es reducido y poco detallado.

6.2 Recomendaciones

- Durante la verificación de los requerimientos de seguridad, el evaluador no debe limitarse a las pruebas definidas en la guía MSTG, sino que también debe apoyarse en su experiencia y conocimiento para validar que se cumpla el requerimiento.
- Es importante tener presente el tema de la seguridad desde las primeras etapas del ciclo de vida del desarrollo de sistemas, ya que esto disminuye en gran medida las vulnerabilidades y fallos de seguridad de los aplicativos en producción.
- Plasmar en un documento y socializar las buenas prácticas de seguridad para los miembros del Departamento de Desarrollo de Sistemas; para así garantizar el conocimiento y ejercicio de estas en el desarrollo de aplicaciones en Android.
- Es necesario verificar los controles de seguridad en el desarrollo de aplicaciones móviles y elementos asociados, por lo cual, la ejecución de pruebas de seguridad se convierte en una actividad imprescindible para disminuir el riesgo de fuga de información sensible o acciones maliciosas.
- Adoptar en la documentación, proceso de desarrollo y verificación de aplicaciones móviles en Android, las recomendaciones y buenas prácticas que brinda el estándar MASVS publicado por OWASP.
- Brindar capacitación al personal de desarrollo sobre las mejores prácticas de seguridad en desarrollo de aplicaciones móviles.

Glosario

API: interfaz de programación de aplicaciones (traducido de Application Programming Interface). “Es un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones” (Redhat, s.f).

APK: paquete de aplicación Android (traducido de Android Application Package). “Es un archivo incluye todos los contenidos de una aplicación de Android y es el archivo que usan los dispositivos con tecnología Android para instalar la aplicación” (GoogleDevelopers, 2020a).

Aplicaciones nativas: “Los sistemas operativos móviles, incluidos Android e iOS, vienen con un kit de desarrollo de software (SDK) para desarrollar aplicaciones específicas para el sistema operativo. Estas aplicaciones se denominan nativas del sistema para el que se han desarrollado” (OWASP, s.f.-k).

CVE: vulnerabilidades y exposiciones comunes (traducido de Common Vulnerabilities and Exposures). “Es una lista de vulnerabilidades y exposiciones de seguridad de la información que tiene como objetivo proporcionar nombres comunes para problemas conocidos públicamente” (CVE Details, s.f-b).

DAST: pruebas dinámicas de seguridad de aplicaciones (traducido de Dynamic Application Security Testing). “Están diseñadas para detectar indicios de vulnerabilidades de seguridad en una aplicación mientras se está ejecutando” (OWASP, 2020, p. 42).

ENA: acrónimo de Encuesta Nacional Agropecuaria (INEC, s.f-a).

Hardcodeada: término informático que hace referencia a incrustar datos directamente en el código fuente del programa.

HTTP: protocolo de transferencia de hipertexto (traducido de Hyper Text Transfer Protocol). “Un protocolo de aplicación para sistemas de información distribuidos y

colaborativos. Es la base de la comunicación de datos para la World Wide Web" (OWASP, 2020, p. 43).

INEC: acrónimo de Instituto Nacional de Estadística y Censos. "Es responsable de la gestión de las estadísticas nacionales para orientar las decisiones que promuevan el desarrollo del país" (INEC, s.f-b).

Ingeniería Inversa: "Es el proceso de desmontar una aplicación para descubrir cómo funciona. Puede hacer esto examinando la aplicación compilada (análisis estático), observación de la aplicación durante el tiempo de ejecución (análisis dinámico) o una combinación de ambos" (OWASP, s.f.-ñ).

Inyección SQL (SQLi): "Una técnica de inyección de código utilizada para atacar aplicaciones basadas en datos, en la que se insertan instrucciones SQL maliciosas en un punto de entrada" (OWASP, 2020, p. 44).

Malware: "Código ejecutable que se introduce en una aplicación durante el tiempo de ejecución sin el conocimiento del usuario o administrador de la misma" (OWASP, 2020, p. 43).

MASVS: estándar de verificación de seguridad de aplicaciones móviles (traducido de Mobile Application Security Verification Standard). "Es un esfuerzo comunitario para establecer un marco de requisitos de seguridad necesarios para diseñar, desarrollar y probar la seguridad de aplicaciones móviles iOS y Android" (OWASP, 2020, p. 7).

MD5: algoritmo de resumen del mensaje 5 (traducido de Message-Digest Algorithm 5). "Es un algoritmo de resumen de mensajes que toma como entrada un mensaje de longitud arbitraria y produce como salida una "huella digital" de 128 bits o "resumen de mensaje" de la entrada" (Internet Engineering Task Force [IETF], 2011).

MITRE: la Corporación MITRE (traducido de MITRE Corporation). "Como organización sin fines de lucro, trabaja en el interés público en los gobiernos federales, estatales y locales, así como en la industria y el mundo académico. Generamos ideas innovadoras

en áreas tan variadas como la inteligencia artificial, la ciencia de datos intuitivos, la ciencia de la información cuántica, la informática de la salud, la seguridad espacial, la experiencia política y económica, la autonomía de confianza, el intercambio de amenazas cibernéticas y la resiliencia cibernética" (MITRE, s.f).

MSTG: guía de pruebas de seguridad para móviles (traducido de Mobile Security Testing Guide). "Es un manual completo para probar la seguridad de las aplicaciones móviles. Describe los procesos y técnicas para verificar los requisitos enumerados en el Estándar de verificación de seguridad de aplicaciones móviles (MASVS) y proporciona una línea de base para pruebas de seguridad completas y consistentes" (OWASP, s.f.-a).

NIAP: Asociación Nacional de Garantía de la Información (traducido de National Information Assurance Partnership). "Es responsable de la implementación en los Estados Unidos de los Criterios Comunes" (NIAP, s.f).

NIST: Instituto Nacional de Estándares y Tecnología (traducido de National Institute of Standards and Technology). "El instituto que tiene como misión de promover la innovación y la competitividad industrial de los EE. UU" (NIST, 2015).

Ofimática: "Automatización, mediante sistemas electrónicos, de las comunicaciones y procesos administrativos en las oficinas" (Real Academia Española, 2021).

OWASP: proyecto abierto de seguridad de aplicaciones web (traducido de Open Web Application Security Project). "Es una comunidad abierta a nivel mundial y sin ánimo de lucro enfocada en mejorar la seguridad del software de aplicaciones" (OWASP, s.f.-s).

PII: información de identificación personal (traducido de Personally Identifiable Information). "Es la información que se puede utilizar por sí sola o junto con otra información para identificar, contactar o localizar a una sola persona, o para identificar a un individuo en su contexto" (OWASP, 2020, p. 43).

PIN: número de identificación personal (traducido de Personal Identification Number). “Es una cadena numérica relativamente corta (normalmente de 4 a 8 dígitos) que se utiliza como contraseña para autenticar a un usuario en un dispositivo” (Springer, s.f).

REST: Transferencia de Estado Representacional (traducido de Representational State Transfer). “Es un estilo arquitectónico REST, los datos y la funcionalidad se consideran recursos y se accede a ellos mediante identificadores uniformes de recursos (URI), generalmente enlaces en la Web” (Oracle, s.f-b).

RESTful: hace referencia a un servicio web que implementa la arquitectura REST.

SSL certificate pinning: “La fijación de certificados es el proceso de asociar el servidor backend con un certificado X.509 particular o una clave pública en lugar de aceptar cualquier certificado firmado por una autoridad certificadora de confianza” (OWASP, s.f.-c).

SAST: pruebas estáticas de seguridad de aplicaciones (traducido de Static Application Security Testing). “Son un conjunto de tecnologías diseñadas para analizar el código fuente de la aplicación” (OWASP, 2020, p. 43).

TICS: acrónimo de Tecnologías de Información y Comunicaciones.

URI: Identificador Uniforme de Recursos (traducido de Uniform Resource Identifier). “Es una cadena de caracteres que se utiliza para identificar un nombre o un recurso web” (OWASP, 2020, p. 44).

URL: Localizador Uniforme de Recursos (traducido de Uniform Resource Locator). “Se utiliza a menudo como referencia a un recurso” (OWASP, 2020, p. 44).

UTSI: acrónimo de Unidad Técnica de Sistemas e Informática.

XML: lenguaje de marcas generalizado (traducido de eXtensible Markup Language). “Es un lenguaje usado para estructurar información en cualquier documento” (ri5, s.f).

Referencias

Abraham, A. (s.f). *Mobile Security Framework (3.4 beta) [Software]*. Obtenido de https://github.com/MobSF/Mobile-Security-Framework-MobSF

Antonishyn, M., & Misnik, O. (2019). Analysis of testing approaches to Android mobile application vulnerabilities. Obtenido de http://ceur-ws.org/Vol-2577/paper22.pdf

Bloom, B. S. (1956). *Taxonomy of educational objectives: The classification of educational goals.* New York: Longmans, Green.

CVE Details. (20 de Diciembre de 2018a). *CVE-2018-1000850*. Obtenido de https://www.cvedetails.com/cve/CVE-2018-1000850/

CVE Details. (26 de Abril de 2018b). *CVE-2018-10237*. Obtenido de https://www.cvedetails.com/cve/CVE-2018-10237/

CVE Details. (s.f-a). *Current CVSS Score Distribution For All Vulnerabilities*. Obtenido de www.cvedetails.com

CVE Details. (s.f-b). *CVE Terminology and FAQ.* Obtenido de https://www.cvedetails.com/cve-help.php

DB4S. (s.f). *DB Browser for SQLite (3.12.1) [Software]*. Obtenido de https://sqlitebrowser.org/dl/

Elenkov , N. (s.f). *Android backup extractor (20210309062218-e30cc24) [Software]*. Obtenido de https://github.com/nelenkov/android-backup-extractor

F-Secure. (s.f). *Drozer (2.4.4) [Software]*. Obtenido de https://labs.f-secure.com/tools/drozer/

Google. (22 de Mayo de 2021). *Nombres internos, etiquetas y números de compilación*. Obtenido de https://source.android.com/setup/start/build-numbers?hl=es-419

GoogleDevelopers. (7 de octubre de 2020a). *Aspectos fundamentales de la aplicación*. Obtenido de https://developer.android.com/guide/components/fundamentals

GoogleDevelopers. (21 de Diciembre de 2020b). *Cómo crear y administrar dispositivos virtuales*. Obtenido de https://developer.android.com/studio/run/managing-avds

GoogleDevelopers. (16 de Dociembre de 2020c). *Cómo ejecutar apps en Android Emulator*. Obtenido de https://developer.android.com/studio/run/emulator

GoogleDevelopers. (17 de mayo de 2021a). *Android Debug Bridge (ADB): (31.0.0) [Software]*. Obtenido de https://developer.android.com/studio/releases/platform-tools?hl=es-419

GoogleDevelopers. (18 de Marzo de 2021b). *apksigner*. Obtenido de https://developer.android.com/studio/command-line/apksigner

GoogleDevelopers. (3 de Mayo de 2021c). *Cómo medir el rendimiento de las apps con Android Profiler*. Obtenido de https://developer.android.com/studio/profile/android-profiler?hl=es-419

GoogleDevelopers. (s.f). *Android Studio (4.1.3) [Software]*. Obtenido de https://developer.android.com/studio?hl=es-419

INEC. (27 de octubre de 2020). *Quienes Somos.* Obtenido de http://www.inec.go.cr/quienes-somos

INEC. (s.f-a). *Encuesta Nacional Agropecuaria*. Obtenido de https://www.inec.cr/encuesta-nacional-agropecuaria

INEC. (s.f-b). *Instituto Nacional de Estadística y Censos*. Obtenido de https://www.inec.cr/acerca-inec

Internet Engineering Task Force [IETF]. (2011). *Updated Security Considerations for the MD5 Message-Digest and the HMAC-MD5 Algorithms*. Obtenido de https://tools.ietf.org/html/rfc6151

Java Decompiler Project. (s.f). *JD-GUI (1.6.6) [Software]*. Obtenido de http://java-decompiler.github.io/

MITRE. (s.f). *MITRE*. Obtenido de https://www.mitre.org/about/corporate-overview

Mvnrepository. (s.f). *MvnRepository*. Obtenido de https://mvnrepository.com/

NIAP. (s.f). *National Infomation Assurance Partnership*. Obtenido de https://www.niap-ccevs.org/

NIST. (05 de Mayo de 2015). *Senate Confirms May as 15th NIST Director*. Obtenido de https://www.nist.gov/news-events/news/2015/05/senate-confirms-may-15th-nist-director

NIST. (2019). Vetting the Security of Mobile Applications. doi:https://doi.org/10.6028/NIST.SP.800-163r1

Oracle. (s.f-a). *jarsigner*. Obtenido de https://docs.oracle.com/javase/7/docs/technotes/tools/windows/jarsigner.html

Oracle. (s.f-b). *What Are RESTful Web Services?* Obtenido de https://docs.oracle.com/javaee/6/tutorial/doc/gijqy.html

OWASP. (2020a). *Checklists.* Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Checklists/Mobile_App_Security_Checklist-Spanish_1.2.xlsx

OWASP. (2020b). *MASVS.* Obtenido de https://github.com/OWASP/owasp-masvs/releases/tag/v1.2

OWASP. (s.f.-a). *About the OWASP Mobile Security Testing Guide*. Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/0x02-Frontispiece.md

OWASP. (s.f.-b). *Architectural Information*. Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/0x04b-Mobile-App-Security-Testing.md#architectural-information

OWASP. (s.f.-c). *Certificate and Public Key Pinning*. Obtenido de https://owasp.org/www-community/controls/Certificate_and_Public_Key_Pinning#When_Do_You_Pin.3F

OWASP. (s.f.-d). *Checking for Sensitive Data Disclosure Through the User Interface (MSTG-STORAGE-7)*. Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/0x05d-Testing-Data-Storage.md#checking-for-sensitive-data-disclosure-through-the-user-interface-mstg-storage-7

OWASP. (s.f.-e). *Dynamic Analysis*. Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/0x04b-Mobile-App-Security-Testing.md#dynamic-analysis

OWASP. (s.f.-f). *Exploitation*. Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/0x04b-Mobile-App-Security-Testing.md#exploitation

OWASP. (s.f.-g). *Identifying Sensitive Data.* Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/0x04b-Mobile-App-Security-Testing.md#identifying-sensitive-data

OWASP. (s.f.-h). *Intelligence Gathering*. Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/0x04b-Mobile-App-Security-Testing.md#intelligence-gathering

OWASP. (s.f.-i). *Mapping the Application*. Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/0x04b-Mobile-App-Security-Testing.md#mapping-the-application

OWASP. (s.f.-j). *Mobile App Security Testing*. Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/0x04b-Mobile-App-Security-Testing.md#mobile-app-security-testing

OWASP. (s.f.-k). *Native App.* Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/0x04a-Mobile-App-Taxonomy.md#native-app

OWASP. (s.f.-l). *OWASP Mobile Overview*. Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/Images/Chapters/0x03/owasp-mobile-overview.jpg

OWASP. (s.f.-m). *Preparation*. Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/0x04b-Mobile-App-Security-Testing.md#preparation

OWASP. (s.f.-n). *Reporting*. Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/0x04b-Mobile-App-Security-Testing.md#reporting

OWASP. (s.f.-ñ). *Reverse Engineering*. Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/0x05c-Reverse-Engineering-and-Tampering.md#reverse-engineering

OWASP. (s.f.-o). *Static Analysis*. Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/0x04b-Mobile-App-Security-Testing.md#static-analysis

OWASP. (s.f.-p). *Testing Logs for Sensitive Data (MSTG-STORAGE-3)*. Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/0x05d-Testing-Data-Storage.md#determining-whether-the-keyboard-cache-is-disabled-for-text-input-fields-mstg-storage-5

OWASP. (s.f.-q). *The OWASP Mobile AppSec Verification Standard*. Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/0x03-Overview.md#the-owasp-mobile-appsec-verification-standard

OWASP. (s.f.-r). *Vulnerability Analysis*. Obtenido de https://github.com/OWASP/owasp-mstg/blob/master/Document/0x04b-Mobile-App-Security-Testing.md#vulnerability-analysis

OWASP. (s.f.-s). *Who is the OWASP® Foundation?* Obtenido de https://www.owasp.org/

Pan, B., Barrios , N., Schuberth, S., & Papp , R. (s.f). *Dex2jar (2.1) [Software].* Obtenido de https://github.com/pxb1988/dex2jar/

PortSwigger. (s.f). *BurpSuite (2021.3.1) [Software]*. Obtenido de https://portswigger.net/burp/communitydownload

Qualys. (s.f). *SSL Server Test (2.1.8) [Software]*. Obtenido de https://www.ssllabs.com/ssltest/

Real Academia Española. (2021). *Diccionario de la lengua española, 23.ª ed., [versión 23.4 en línea]*. Recuperado el 7 de 5 de 2021, de https://dle.rae.es

Redhat. (s.f). *Qué son las API y para qué sirven*. Obtenido de https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces

ri5. (s.f). *Que es XML y para que sirve*. Obtenido de https://www.ri5.com.ar/wiki/Que_es_XML.php

Rodrígez Sanchez, E. (2019). Auditoría de seguridad de aplicaciones móviles. *[Tesis de Maestría, Universitat Oberta de Catalunya (UOC)]*. Obtenido de http://openaccess.uoc.edu/webapps/o2/handle/10609/95927

Springer. (s.f). *Personal Identification Number (PIN)*. Obtenido de https://link.springer.com/referenceworkentry/10.1007%2F978-1-4419-5906-5_91

Wällstedt, V. (2019). Implementation and Security Evaluation of User-Customized Content in a Mobile Application. *[Tesis de Maestría,Linköping University]*. Obtenido de https://www.diva-portal.org/smash/get/diva2:1334227/FULLTEXT01.pdf

Wireshark Foundation. (s.f). *Wireshark (3.4.4) [Software]*. Obtenido de https://www.wireshark.org/#download

Anexos

Anexo 1. Carta de autorización.

San José, 19 de octubre del 2020

Señores

Universidad CENFOTEC

Estimados señores:

Para lo que corresponda, autorizo al funcionario Emmanuel Méndez Castillo con cédula 3-0429-0963 para que realice el Trabajo Final de Graduación denominado **"Análisis de seguridad a la aplicación móvil "Sistema de Captura de Datos" (SICAD) de la Encuesta Nacional Agropecuaria (ENA), del Instituto Nacional de Estadísticas y Censos (INEC), tomando como referencia el estándar de Verificación de Seguridad de Aplicaciones Móviles (MASVS) Versión 1.2."**, correspondiente al plan de estudio de la Maestría en Ciberseguridad en el proyecto de la Encuesta Nacional Agropecuaria (ENA) que realiza el Instituto Nacional de Estadística y Censos (INEC).

La autorización consiste en que el funcionario realice un análisis de seguridad de las posibles vulnerabilidades que pueda tener el aplicativo móvil utilizado para capturar información desde tabletas con sistema operativo Android. El trabajo será únicamente para fines académicos, el código fuente es propiedad exclusiva del INEC por lo cual no será transferido ni donado de manera parcial o total, no se brindará ningún dato detallado que exponga la funcionalidad o seguridad del aplicativo.

Atentamente

LORAINE VARGAS CORDERO (FIRMA)

Firmado digitalmente por LORAINE VARGAS CORDERO (FIRMA)
Fecha: 2020.10.19 11:23:30 -06'00'

Licda. Loraine Vargas Cordero

Coordinadora de Unidad Técnica de Sistemas e Informática.

Anexo 2. Lista de chequeo de seguridad móvil para Android de OWASP

OWASP Lista de Verificación de Seguridad en Aplicaciones Móviles

Basado en el Estándar de Verificación de Seguridad en Aplicaciones Móviles 1.0 de OWASP.

Información general de evaluación	
Versión del MASVS	1.2
Versión online del MAS	https://github.com/OWASP/owasp-masvs/blob/1.2/Document/
Versión del MSTG	1.2
Versión online del MSTG	https://github.com/OWASP/owasp-mstg/tree/master/Document
Las dos filas anteriores se utilizan para construir la base para todos los hipervínculos en las listas de control de Android e iOS. Ajuste a su caso de uso específico para actualizar todos los hipervínculos a una versión específica del MSTG	
Nombre del Cliente:	
Lugar de la evaluación:	
Fecha de Inicio:	
Fecha de Cierre:	
Nombre del Evaluador:	
Alcance de la Evaluació	
Nivel de verificación	**Nivel 1:** Seguridad estándar. **Nivel 2:** Defensa en profundidad. **Nivel- 3:** Resistencia contra la ingeniería inversa y la manipulación.

Información de Evaluación Plataforma Android	
Nombre de la aplicación:	
Enlace de Google Play	
Nombre del Archivo	
Versión	
Hash MD5 del archivo A	

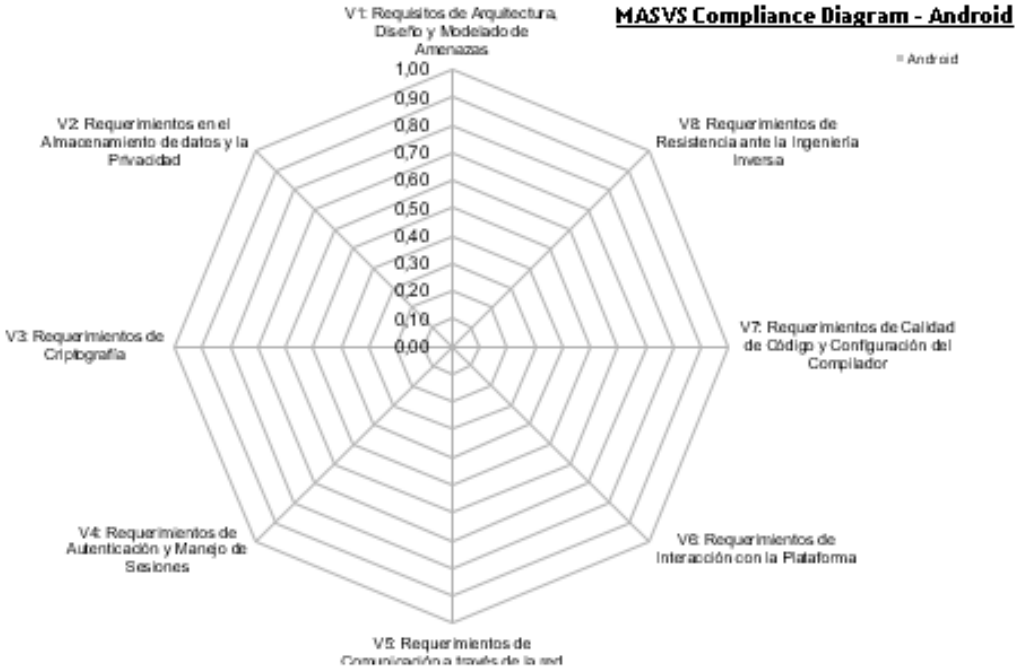
Testing information iOS	
Nombre de la aplicación:	
Enlace de Apple Store	
Nombre del Archivo	
Versión	
Hash MD5 del archivo II	

Representantes del Cliente e Información de Contacto	
Nombres:	
Empresa:	
Posición:	
Teléfono:	
Correo Electrónico:	
Nombres:	
Empresa:	
Posición:	
Teléfono:	
Correo Electrónico:	

Management Summary

Cumplimiento Estándar MASVS (/5)

0



	Android				iOS			
	P	F	NA	%	P	F	NA	%
V1: Requisitos de Arquitectura, Diseño y Modelado de Amenaz	0	0	7	0,00%	0	0	7	0,00%
V2: Requerimientos en el Almacenamiento de datos y la Privaci	0	0	8	0,00%	0	0	8	0,00%
V3: Requerimientos de Criptografía	0	0	0	0,00%	0	0	0	0,00%
V4: Requerimientos de Autenticación y Manejo de Sesiones	0	0	4	0,00%	0	0	4	0,00%
V5: Requerimientos de Comunicación a través de la red	0	0	3	0,00%	0	0	3	0,00%
V6: Requerimientos de Interacción con la Plataforma	0	0	3	0,00%	0	0	3	0,00%
V7: Requerimientos de Calidad de Código y Configuración del C	0	0	0	0,00%	0	0	0	0,00%
V8: Requerimientos de Resistencia ante la Ingeniería Inversa	0	0	13	0,00%	0	0	13	0,00%

ID	MSTG-ID	Verificación detallada de requerimientos	Nivel 1	Nivel 2	Estado
V1		**Arquitectura, diseño y modelado de amenazas**			
1.1	MSTG-ARCH-1	Todos los componentes se encuentran identificados y asegurar que son necesarios.	✓	✓	
1.2	MSTG-ARCH-2	Los controles de seguridad nunca se aplican solo en el cliente, sino también en los respectivos servidores.	✓	✓	
1.3	MSTG-ARCH-3	Se definió una arquitectura de alto nivel para la aplicación y los servicios, y se incluyeron controles de seguridad en esta.	✓	✓	
1.4	MSTG-ARCH-4	Se identificó claramente la información considerada sensible en el contexto de la aplicación móvil.	✓	✓	
1.5	MSTG-ARCH-5	Todos los componentes de la aplicación están definidos en términos de la lógica de negocio o las funciones de seguridad que proveen.		✓	N/A
1.6	MSTG-ARCH-6	Se realizó un modelado de amenazas para la aplicación móvil y los servicios en el que se definieron estas y sus contramedidas.		✓	N/A
1.7	MSTG-ARCH-7	Todos los controles de seguridad poseen una implementación centralizada.		✓	N/A
1.8	MSTG-ARCH-8	Existe una política explícita sobre el uso de claves criptográficas (si se usan) a través de todo su ciclo de vida. Idealmente siguiendo un estándar de gestión de claves como el NIST SP 800-57.		✓	N/A
1.9	MSTG-ARCH-9	Existe un mecanismo para forzar las actualizaciones de la aplicación móvil.		✓	N/A
1.10	MSTG-ARCH-10	La implementación de medidas de seguridad es una parte esencial durante todo el ciclo de vida del desarrollo de *software* de la aplicación.		✓	N/A
1.11	MSTG-ARCH-11	Existe una política de divulgación responsable y es llevada a cabo adecuadamente.		✓	N/A
1.12	MSTG-ARCH-12	La aplicación debería cumplir con las leyes y regulaciones de privacidad.	✓	✓	
V2		**Almacenamiento de datos y la privacidad**			
2.1	MSTG-STORAGE-1	Las funcionalidades de almacenamiento de credenciales del sistema deben ser utilizadas para almacenar información sensible, tal como información personal, credenciales de usuario o claves criptográficas.	✓	✓	
2.2	MSTG-STORAGE-2	No se debe almacenar información sensible fuera del contenedor de la aplicación o del almacenamiento de credenciales del sistema.	✓	✓	
2.3	MSTG-STORAGE-3	No se escribe información sensible en los registros (logs) de la aplicación.	✓	✓	
2.4	MSTG-STORAGE-4	No se comparte información sensible con servicios externos salvo que sea una necesidad de la arquitectura.	✓	✓	
2.5	MSTG-STORAGE-5	Se desactiva la caché del teclado en los campos de texto que contienen información sensible.	✓	✓	

2.6	**MSTG-STORAGE-6**	No se expone información sensible mediante mecanismos de comunicación entre procesos (IPC).	✓	✓	
2.7	**MSTG-STORAGE-7**	No se expone información sensible como contraseñas y números de tarjetas de crédito a través de la interfaz o capturas de pantalla.	✓	✓	
2.8	**MSTG-STORAGE-8**	No se incluye información sensible en las copias de seguridad generadas por el sistema operativo.		✓	N/A
2.9	**MSTG-STORAGE-9**	La aplicación elimina toda información sensible de la vista cuando la aplicación pasa a un segundo plano.		✓	N/A
2.10	**MSTG-STORAGE-10**	La aplicación no conserva ninguna información sensible en memoria más de lo necesario y la memoria se limpia tras su uso.		✓	N/A
2.11	**MSTG-STORAGE-11**	La aplicación obliga a que exista una política mínima de seguridad en el dispositivo, como que el usuario deba configurar un código de acceso.		✓	N/A
2.12	**MSTG-STORAGE-12**	La aplicación educa al usuario acerca de los tipos de información personal que procesa y de las mejores prácticas en seguridad que el usuario debería seguir al utilizar la aplicación.		✓	N/A
2.13	**MSTG-STORAGE-13**	No se guarda ningún tipo de información sensible de forma local en el dispositivo móvil. En su lugar, esa información debería ser obtenida desde un sistema remoto solo cuando es necesario y únicamente residir en memoria.		✓	N/A
2.14	**MSTG-STORAGE-14**	En caso de ser necesario, guardar información sensible de forma local, esta debe ser cifrada usando una clave derivada del *hardware* de almacenamiento seguro, el cual debe requerir autenticación previa.		✓	N/A
2.15	**MSTG-STORAGE-15**	El almacenamiento local de la aplicación debe ser borrado tras un número excesivo de intentos fallidos de autenticación.		✓	N/A
V3		**Criptografía**			
3.1	**MSTG-CRYPTO-1**	La aplicación no depende únicamente de criptografía simétrica, cuyas claves se encuentran directamente en el código fuente de esta.	✓	✓	
3.2	**MSTG-CRYPTO-2**	La aplicación utiliza implementaciones de criptografía probadas.	✓	✓	
3.3	**MSTG-CRYPTO-3**	La aplicación utiliza primitivas de seguridad que son apropiadas para el caso particular y su configuración y parámetros siguen las mejores prácticas de la industria.	✓	✓	
3.4	**MSTG-CRYPTO-4**	La aplicación no utiliza protocolos o algoritmos criptográficos ampliamente considerados obsoletos para su uso en seguridad.	✓	✓	
3.5	**MSTG-CRYPTO-5**	La aplicación no reutiliza una misma clave criptográfica para varios propósitos.	✓	✓	
3.6	**MSTG-CRYPTO-6**	Los valores aleatorios son generados utilizando un generador de números aleatorios suficientemente seguro.	✓	✓	

V4		Autenticación y manejo de sesiones			
4.1	MSTG-AUTH-1	Si la aplicación provee acceso a un servicio remoto, un mecanismo aceptable de autenticación como usuario y contraseña es realizado en el servidor remoto.	✓	✓	
4.2	MSTG-AUTH-2	Si se utiliza la gestión de sesión por estado, el servidor remoto usa tokens de acceso aleatorios para autenticar los pedidos del cliente sin requerir el envío de las credenciales del usuario en cada uno.	✓	✓	
4.3	MSTG-AUTH-3	Si se utiliza la autenticación basada en tokens sin estado, el servidor proporciona un token que se ha firmado utilizando un algoritmo seguro.	✓	✓	
4.4	MSTG-AUTH-4	Cuando el usuario cierra sesión, se termina la sesión también en el servidor.			
4.5	MSTG-AUTH-5	Existe una política de contraseñas y es aplicada en el servidor.	✓	✓	
4.6	MSTG-AUTH-6	El servidor implementa mecanismos, cuando credenciales de autenticación son ingresadas una cantidad excesiva de veces.	✓	✓	
4.7	MSTG-AUTH-7	Las sesiones y los tokens de acceso expiran luego de un tiempo predefinido de inactividad.	✓	✓	
4.8	MSTG-AUTH-8	La autenticación biométrica, si la hay, no está asociada a eventos (p. ej. usando una API que simplemente retorna "true" o "false"), sino basada en el desbloqueo del keychain/keystore (almacenamiento seguro).		✓	N/A
4.9	MSTG-AUTH-9	El sistema remoto implementa un mecanismo de segundo factor de autenticación (2FA) y lo impone consistentemente.		✓	N/A
4.10	MSTG-AUTH-10	Para realizar transacciones críticas, se requiere una autenticación adicional (step-up).		✓	N/A
4.11	MSTG-AUTH-11	La aplicación informa al usuario acerca de todas las actividades sensibles en su cuenta. El usuario es capaz de ver una lista de los dispositivos conectados, información contextual (dirección IP, localización, etc.) y es capaz de bloquear ciertos dispositivos.		✓	N/A
4.12	MSTG-AUTH-12	Los modelos de autorización deberían ser definidos e impuestos por el sistema remoto.	✓	✓	
V5		Comunicación a través de la red			
5.1	MSTG-NETWORK-1	La información es enviada cifrada utilizando TLS. El canal seguro es usado consistentemente en la aplicación.	✓	✓	
5.2	MSTG-NETWORK-2	Las configuraciones del protocolo TLS siguen las mejores prácticas de la industria o lo hacen lo mejor posible en caso de que el sistema operativo del dispositivo no soporte los estándares recomendados.	✓	✓	
5.3	MSTG-NETWORK-3	La aplicación verifica el certificado X.509 del sistema remoto al establecer el canal seguro y solo se aceptan certificados firmados por una CA de confianza.	✓	✓	

5.4	MSTG-NET-WORK-4	La aplicación utiliza su propio almacén de certificados o realiza _pinning_ del certificado o la clave pública del servidor. Bajo ningún concepto establecerá conexiones con servidores que ofrecen otros certificados o claves, incluso si están firmados por una CA de confianza.		✓	N/A
5.5	MSTG-NET-WORK-5	La aplicación no depende de un único canal de comunicaciones inseguro (email o SMS) para operaciones críticas como registro de usuarios o recuperación de cuentas.		✓	N/A
5.6	MSTG-NET-WORK-6	La aplicación solo depende de bibliotecas de conectividad y seguridad actualizadas.		✓	N/A
V6		**Interacción con la plataforma**			
6.1	MSTG-PLAT-FORM-1	La aplicación requiere la cantidad de permisos mínimamente necesaria.	✓	✓	
6.2	MSTG-PLAT-FORM-2	Todo dato ingresado por el usuario o cualquier fuente externa debe ser validado y, si es necesario, saneado. Esto incluye información recibida por la UI o mecanismos IPC como los Intents, URLs y datos provenientes de la red.	✓	✓	
6.3	MSTG-PLAT-FORM-3	La aplicación no expone ninguna funcionalidad sensible a través esquemas de URL, salvo que dichos mecanismos estén debidamente protegidos.	✓	✓	
6.4	MSTG-PLAT-FORM-4	La aplicación no expone ninguna funcionalidad sensible a través de mecanismos IPC, salvo que dichos mecanismos estén debidamente protegidos.	✓	✓	
6.5	MSTG-PLAT-FORM-5	JavaScript se encuentra deshabilitado en los WebViews, salvo que sea necesario.	✓	✓	
6.6	MSTG-PLAT-FORM-6	Las WebViews se configuran para permitir el mínimo de los esquemas (idealmente, solo https). Esquemas peligrosos como file, tel y app-id están deshabilitados.	✓	✓	
6.7	MSTG-PLAT-FORM-7	Si objetos nativos son expuestos en WebViews, debe verificarse que cualquier componente JavaScript se carga exclusivamente desde el contenedor de la aplicación.	✓	✓	
6.8	MSTG-PLAT-FORM-8	La serialización de objetos, si se realiza, debe implementarse utilizando API seguras.	✓	✓	
6.9	MSTG-PLAT-FORM-9	La aplicación se protege contra ataques de tipo screen overlay (solo Android),		✓	N/A
6.10	MSTG-PLAT-FORM-10	La caché, el almacenamiento y los recursos cargados (JavaScript, etc.) de las WebViews deben borrarse antes de destruir la WebView.		✓	N/A
6.11	MSTG-PLAT-FORM-11	Verificar que la aplicación impide el uso de teclados de terceros siempre que se introduzca información sensible.		✓	N/A

V7		Calidad de código y configuración del compilador			
7.1	MSTG-CODE-1	La aplicación es firmada y provista con un certificado válido, cuya clave privada está debidamente protegida.	✓	✓	
7.2	MSTG-CODE-2	La aplicación fue publicada en modo release y con las configuraciones apropiadas para el mismo (por ejemplo, non-debuggable).	✓	✓	
7.3	MSTG-CODE-3	Los símbolos de depuración fueron eliminados de los binarios nativos.	✓	✓	
7.4	MSTG-CODE-4	Cualquier código de depuración o de asistencia al desarrollador (p. ej. código de test, backdoors, configuraciones ocultas) debe ser eliminado. La aplicación no hace logs detallados de errores ni de mensajes de depuración.	✓	✓	
7.5	MSTG-CODE-5	Todos los componentes de terceros se encuentran identificados y revisados en cuanto a vulnerabilidades conocidas.	✓	✓	
7.6	MSTG-CODE-6	La aplicación captura y gestiona debidamente las posibles excepciones.	✓	✓	
7.7	MSTG-CODE-7	Los controles de seguridad deniegan el acceso por defecto.	✓	✓	
7.8	MSTG-CODE-8	En código no administrado, la memoria es solicitada, utilizada y liberada de manera correcta.	✓	✓	
7.9	MSTG-CODE-9	Las funcionalidades de seguridad gratuitas de las herramientas, tales como minificación del bytecode, protección de la pila, soporte PIE y conteo automático de referencias, se encuentran activadas.	✓	✓	

Anexo 3. Plantilla utilizada para reportar los hallazgos.

MASVS-V# (*En esta sección debe agregarse el número de requerimiento)*	**Descripción del requerimiento** (*En esta sección debe agregarse el requerimiento a evaluar)*	**Resultado** *(En esta sección debe agregarse Aprobado o Desaprobado)*
Análisis Estático o Dinámico.		
Condición: (*En esta sección debe agregarse la situación actual encontrada.)* **Criterio:** (*En esta sección debe agregarse la normativa legal o técnica aplicable.)* **Causa:** (*En esta sección debe agregarse las razones de desviación, determina por qué la condición no cumple con el criterio.)* **Efecto:** (*En esta sección debe agregarse la determinación de los resultados de las consecuencias de la condición al no cumplir el criterio.)* **Herramienta(s) utilizada(s) en esta prueba:** (*En esta sección debe agregar el nombre de herramienta de software utilizada en el caso de pruebas.)*		
Recomendaciones: (*En esta sección debe agregar las acciones que debe realizarse para mejorar la condición.)*		