



**UNIVERSIDAD ESTATAL PENÍNSULA
DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES
INSTITUTO DE POSTGRADO**

TÍTULO

**DESARROLLO DE UN PLUGIN CON INTELIGENCIA ARTIFICIAL
PARA MINIMIZAR ATAQUES INGENIERÍA SOCIAL EN UN
NAVEGADOR.**

AUTOR

Díaz Reyes, Alex Miguel

TRABAJO DE TITULACIÓN

**Previo a la obtención del grado académico en
MAGÍSTER EN CIBERSEGURIDAD**

TUTOR.

Orozco Iguasnia, Jaime Benjamín

Santa Elena, Ecuador

Año 2025



**UNIVERSIDAD ESTATAL PENÍNSULA
DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES
INSTITUTO DE POSTGRADO**

TRIBUNAL DE SUSTENTACIÓN

**Ing. Alicia Andrade Vera, Mgtr.
COORDINADORA DEL
PROGRAMA**

**Ing. Jaime Orozco Iguasnia, Mgtr.
TUTOR**

**Lsi. Daniel Quirumbay Yagual, Mgtr.
DOCENTE
ESPECIALISTA**

**Ing. Carlos Castillo Yagual, Mgtr.
DOCENTE
ESPECIALISTA**

**Abg. María Rivera González, Mgtr.
SECRETARIA GENERAL
UPSE**



**UNIVERSIDAD ESTATAL PENÍNSULA
DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES
INSTITUTO DE POSTGRADO**

CERTIFICACIÓN

Certifico que luego de haber dirigido científica y técnicamente el desarrollo y estructura final del trabajo, este cumple y se ajusta a los estándares académicos, razón por el cual apruebo en todas sus partes el presente trabajo de titulación que fue realizado en su totalidad por **Alex Miguel Díaz Reyes**, como requerimiento para la obtención del título de Magíster en Ciberseguridad.

Santa Elena, 18 de octubre de 2024

TUTOR

Ing. Jaime Benjamín Orozco Iguasnia, Mgtr.



**UNIVERSIDAD ESTATAL PENÍNSULA
DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES
INSTITUTO DE POSTGRADO**

DECLARACIÓN DE RESPONSABILIDAD

Yo, **Alex Miguel Díaz Reyes**

DECLARO QUE:

El trabajo de Titulación, Desarrollo de un plugin con inteligencia artificial para minimizar ataques ingeniería social en un navegador, previo a la obtención del título en Magíster en Ciberseguridad, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

Santa Elena, 18 de octubre de 2024

EL AUTOR

Alex Miguel Díaz Reyes



**UNIVERSIDAD ESTATAL PENÍNSULA
DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES
INSTITUTO DE POSTGRADO**

CERTIFICACIÓN DE ANTIPLAGIO

Certifico que después de revisar el documento final del trabajo de titulación denominado Desarrollo de un plugin con inteligencia artificial para minimizar ataques ingeniería social en un navegador. Presentado por el estudiante, DÍAZ REYES ALEX MIGUEL fue enviado al Sistema Compilatio, presentando un porcentaje de similitud correspondiente al 2%, por lo que se aprueba el trabajo para que continúe con el proceso de titulación.

 CERTIFICADO DE ANÁLISIS
magister

DIAZ REYES ALEX MIGUEL

2% Textos sospechosos

1% Similitudes
0% similitudes entre comillas
0% entre las fuentes mencionadas

< 1% Idiomas no reconocidos

Nombre del documento: DIAZ REYES ALEX MIGUEL.pdf	Depositante: JAIME BENJAMÍN OROZCO IGUASNIA	Número de palabras: 16.027
ID del documento: b95aa8329c660f3f13c5a184ffd9dccb28c20a58	Fecha de depósito: 19/10/2024	Número de caracteres: 109.974
Tamaño del documento original: 1,02 MB	Tipo de carga: interface	
Autores: []	fecha de fin de análisis: 19/10/2024	

TUTOR

Ing. Jaime Benjamín Orozco Iguasnia, Mgtr.



**UNIVERSIDAD ESTATAL PENÍNSULA
DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES
INSTITUTO DE POSTGRADO**

AUTORIZACIÓN

Yo, **Alex Miguel Díaz Reyes**

Autorizo a la Universidad Estatal Península de Santa Elena, para que haga de este trabajo de titulación o parte de él, un documento disponible para su lectura consulta y procesos de investigación, según las normas de la Institución.

Cedo los derechos en línea patrimoniales de mi trabajo de Propuestas metodológicas y tecnológicas avanzadas con fines de difusión pública, además apruebo la reproducción de Propuestas metodológicas y tecnológicas avanzadas dentro de las regulaciones de la Universidad, siempre y cuando esta reproducción no suponga una ganancia económica y se realice respetando mis derechos de autor

Santa Elena, 18 de octubre de 2024

EL AUTOR

Alex Miguel Díaz Reyes

AGRADECIMIENTO

A Dios por darme sabiduría, paciencia y salud en todo el transcurso de toda la Carrera de maestría y permitirme terminar con éxito este trabajo de investigación. A mi familia quienes son mi pilar fundamental y gran apoyo durante esta etapa. A la Universidad Estatal Península de Santa Elena, por darme la oportunidad de aprender nuevos conocimientos en la maestría ciberseguridad que me abrió las puertas y me dio la oportunidad de desarrollar mi trabajo de investigación en la elaboración un plugin con inteligencia artificial. Al tutor Jaime Orozco Iguasnia, quien me guío en el desarrollo del trabajo de investigación, por dedicar su tiempo y darme la oportunidad de participar en un nuevo campo como es el de investigación en inteligencia artificial.

Alex Miguel Díaz Reyes

DEDICATORIA

A Dios, por darme la fortaleza y sabiduría para cumplir cada una de mis metas, por ser mi guía en todo mi proceso de formación, permitiéndome demostrar quién soy y de lo capaz que puedo llegar a ser.

A mi esposa, por ser mi pilar fundamental, por darme el apoyo para seguir preparándome profesionalmente, por su amor y cariño, por darme esa fuerza de luchar y cumplir mis objetivos.

A Elizabeth y Miguel Ángel por todo el cariño y todas las fuerzas que me brindan cada día.

A mi gran amigo que se alegra conmigo por este paso más en la vida.

Y a mis hermanos que ya saben mis palabras.

Alex Miguel Díaz Reyes

ÍNDICE GENERAL

TITULO DEL TRABAJO DE TITULACIÓN	I
TRIBUNAL DE SUSTENTACIÓN.....	II
CERTIFICACIÓN	III
DECLARACIÓN DE RESPONSABILIDAD	IV
CERTIFICACIÓN DE ANTIPLAGIO	V
AUTORIZACIÓN.....	VI
AGRADECIMIENTO	VII
DEDICATORIA	VIII
ÍNDICE GENERAL	IIX
ÍNDICE DE TABLAS	XII
ÍNDICE DE FIGURAS	XIV
RESUMEN	XV
ABSTRACT	XVI
INTRODUCCIÓN	1
Antecedentes	2
Formulación del problema de investigación.....	4
Planteamiento Hipotético.....	7
Pregunta principal.....	7
Preguntas específicas.....	7
Objetivo General:.....	8
Objetivos Específicos:.....	8

Planteamiento hipotético	8
CAPÍTULO 1. MARCO TEÓRICO REFERENCIAL.....	9
1.1. Revisión de literatura	9
1.2 Introducción a la Ingeniería Social	10
1.2.1. Definición y conceptos básicos.....	10
1.2.2 Técnicas comunes de ingeniería social.....	11
1.2.3. Impacto de la ingeniería social en la ciberseguridad	11
1.3 Clasificación de Usuarios y Vulnerabilidades	12
1.3.1. Usuarios novatos y sus debilidades	12
1.3.2. Usuarios medios y phishing sofisticado.....	13
1.3.3. Usuarios frecuentes y ataques dirigidos.....	14
1.3.4. Usuarios expertos y ataques especializados.....	15
1.4 El Rol de los Hackers en la Ingeniería Social	16
1.4.1. Tipos de hackers (white hat, black hat, grey hat)	16
1.4.2. Técnicas de hacking social	16
1.4.3. Motivaciones y objetivos de los hackers en ataques de ingeniería social	17
1.5 Protecciones Actuales contra la Ingeniería Social.....	18
1.5.1. Educación y concienciación.....	18
1.5.2. Políticas de seguridad	19
1.5.3. Capas de seguridad tecnológica.....	19
1.5.4. Verificación de la identidad	20
1.5.5. Actualizaciones y parches.....	21
1.6 Detección de Ataques de Ingeniería Social	21

1.6.1. Señales de detección en correos electrónicos y mensajes.....	21
1.6.2. Detección de ingeniería social en llamadas telefónicas.....	22
1.6.3. Detección de amenazas en redes sociales y presencia en línea	23
1.7 Inteligencia Artificial en la Detección de Ataques.....	24
1.7.1. Algoritmos de aprendizaje automático para la detección de phishing.....	24
1.7.2. Procesamiento de Lenguaje Natural (PLN)	25
1.7.3. Motores de reglas y heurísticas para la evaluación de URLs maliciosas	26
1.8 Consideraciones Éticas en el Uso de IA.....	27
1.8.1. Privacidad y protección de datos del usuario.....	27
1.8.2. Transparencia y consentimiento en el uso de plugins.....	27
1.8.3. Precisión y manejo de falsos positivos	28
1.8.4. Equidad y sesgo en los algoritmos.....	29
1.8.5. Responsabilidad en el desarrollo de sistemas de IA	30
CAPÍTULO 2. METODOLOGÍA.....	31
2.1. Contexto de la investigación	31
2.2. Diseño y alcance de la investigación.....	31
2.3. Tipo y métodos de investigación.....	31
2.3.1 Razones para elegir un enfoque cuantitativo:	32
2.3.2 Métodos de Investigación:	32
2.4. Población y muestra.....	32
2.5. Técnicas e instrumentos de recolección de datos	34
2.6. Procesamiento de la evaluación: Validez y confiabilidad de los instrumentos aplicados para el levantamiento de información.....	34

CAPÍTULO 3. RESULTADOS Y DISCUSIÓN	36
3.1 Resultados del Modelo de Detección de Phishing	36
3.1.1. Proceso de Entrenamiento y Evaluación del Modelo	36
3.2.2. Resultados en la Validación y Pruebas del Modelo.....	37
3.2.3. Comparación con Otros Modelos de Detección	38
3.2.4. Evaluación de los Falsos Positivos y Falsos Negativos.....	39
3.2.5. Conclusiones sobre los Resultados del Modelo.....	39
3.2 Proceso de Conversión y Creación del Plugin para Navegador	40
3.3 Carga y Pruebas de la Extensión	45
3.3.1 Proceso de Carga:	45
3.3.2 Resultados de las Pruebas:	46
3.3.3 Ventajas del Plugin Local	46
3.3.4 Pruebas en Diferentes Navegadores	47
3.3.5. Futuras Mejoras	49
3.4 Prueba de Hipótesis	49
CONCLUSIONES	57
RECOMENDACIONES.....	59
REFERENCIAS	61

ÍNDICE DE TABLAS

Tabla 1. Resumen de estructura	43
Tabla 2. Compatibilidad	47
Tabla 3. Cobertura del plugin	48
Tabla 4. Datos Obtenidos.....	50

ÍNDICE DE FIGURAS

Figura 1. Matriz de confusión de los resultados de la detección de phishing.....	38
Figura 2. Comparación de la precisión del modelo LSTM frente a otros algoritmos de detección	40
Figura 3. Conversión a TensorFlow.js.....	41
Figura 4. Ejemplo de configuración de proyecto en nodejs version.....	44
Figura 5. Configuración básica en el archivo package.json (estructura)	45
Figura 6. Cobertura del plugin.....	49

RESUMEN

Esta tesis aborda el desarrollo de un plugin basado en inteligencia artificial para detectar ataques de ingeniería social, como phishing, en navegadores web. El objetivo es mejorar la seguridad de los usuarios durante la navegación, minimizando la exposición a estos ataques. Se utilizó un modelo de redes neuronales LSTM integrado en un plugin para navegadores, capaz de analizar correos electrónicos en el cliente zimbra en tiempo real. La metodología incluyó la recolección de datos de usuarios en un navegador, primero sin el plugin y luego con el plugin activado. Los resultados mostraron que el uso del plugin mejora significativamente la detección de intentos de phishing, alcanzando una precisión del 92%. Se concluye que la implementación de soluciones basadas en IA puede fortalecer la seguridad en línea y mitigar los riesgos de ingeniería social.

Palabras clave: Ingeniería social, detección de phishing, inteligencia artificial.

ABSTRACT

This thesis addresses the development of an artificial intelligence-based plugin to detect social engineering attacks, such as phishing, in web browsers. The objective is to enhance user security during browsing, minimizing exposure to these attacks. An LSTM neural network model was integrated into a browser plugin capable of analyzing emails in client email zimbra in real-time. The methodology included collecting data from users in a browser, first without the plugin and then with the plugin activated. The results showed that the use of the plugin significantly improves phishing attempt detection, reaching an accuracy of 92%. It is concluded that implementing AI-based solutions can strengthen online security and mitigate the risks of social engineering.

Keywords: Social engineering, phishing detection, artificial intelligence.

INTRODUCCIÓN

En la era digital, la ingeniería social se ha convertido en una de las amenazas más efectivas y utilizadas por los ciberdelincuentes para comprometer la seguridad de los usuarios en línea. Estas técnicas explotan la manipulación psicológica para obtener información confidencial, como credenciales de acceso y datos financieros, lo que hace que los usuarios sean vulnerables a ataques como el phishing, el pretexting y el spear phishing. Estos ataques son cada vez más sofisticados y utilizan correos electrónicos falsificados y sitios web fraudulentos diseñados para engañar a los usuarios y robar sus datos sensibles o inducirlos a realizar acciones perjudiciales.

En Ecuador, la plataforma web de código abierto Zimbra es una de las herramientas más utilizadas por instituciones gubernamentales para la gestión de mensajería y colaboración. Sin embargo, esta plataforma ha sido objeto de ataques frecuentes, lo cual representa un desafío importante para la seguridad de las organizaciones y los usuarios que dependen de ella. La necesidad de una adecuada gestión de la información y la protección de los datos de los usuarios se hace aún más evidente en instituciones como el "Hospital Especializado Julio Endara", donde el uso del servicio de mensajería Zimbra requiere un enfoque más robusto para garantizar la seguridad y la privacidad de la información de los pacientes.

Ante este contexto, el presente trabajo de titulación se centra en el desarrollo de un plugin con inteligencia artificial para identificar y mitigar los ataques de ingeniería social en la plataforma de mensajería y colaboración Zimbra. Este plugin se basa en técnicas de aprendizaje profundo y procesamiento de lenguaje natural (PLN) para analizar y detectar patrones sospechosos en los correos electrónicos, brindando así una capa adicional de protección para los usuarios. El uso de TensorFlow.js permite la ejecución del modelo directamente en el navegador, lo cual asegura una solución privada, segura y eficiente, sin necesidad de comprometer la experiencia de navegación del usuario sus datos personales o a un peor llevando información a un servidor externo para verificar su origen lo que compromete la privacidad de los usuarios, todo se ejecuta en su navegador.

El objetivo de este trabajo es no solo proteger a los usuarios de los ataques de ingeniería social, sino también aumentar la conciencia sobre los riesgos asociados y ofrecer una solución práctica y eficaz que contribuya a un entorno de navegación más seguro. Esta investigación busca superar las limitaciones de las herramientas de seguridad actuales mediante la integración de un enfoque adaptable y basado en inteligencia artificial, capaz de evolucionar junto con las amenazas y proporcionar una defensa sólida contra los ataques de phishing y otras tácticas de ingeniería social, a través de la identificación y en caso de que usuario notifique enviar posibles amenazas para alimentar el modelo.

Por este motivo el presente trabajo de titulación se centra en el desarrollo de un plugin con inteligencia artificial para identificar y analizar las tácticas más comunes de ingeniería social empleadas por los atacantes en línea, con el propósito de detectar y mitigar estos ataques en la plataforma de mensajería y colaboración Zimbra del “Hospital Especializado Julio Endara” y otros que usen la plataforma zimbra, integrando modelos de aprendizaje automático en la detección de patrones sospechosos a través del análisis del cuerpo de correos electrónicos, el plugin proporcionará una capa adicional de protección para los usuarios durante su navegación en internet. Además, se evaluará el impacto de las amenazas de ingeniería social en la seguridad y privacidad, especialmente en lo que respecta al intento de engaño al usuario con correos fraudulentos.

Con el desarrollo de este trabajo aumentará la conciencia sobre los riesgos de la ingeniería social y se ofrecerá una solución práctica y eficaz para reducir la exposición de los usuarios a estos ataques, contribuyendo así a un entorno de navegación más seguro.

Antecedentes

En la última década, el aumento exponencial de las amenazas cibernéticas, particularmente aquellas derivadas de tácticas de ingeniería social como el phishing y el spear phishing, ha generado un interés creciente en el desarrollo de herramientas y técnicas de seguridad que fortalezcan la protección en línea. Estas amenazas no solo han incrementado en frecuencia, sino también en sofisticación, afectando tanto a individuos como a organizaciones de todos los tamaños. Según el Informe Anual de Ciberseguridad de la UE (2021), los ataques de phishing representan uno de los métodos más comunes y

efectivos empleados por los cibercriminales, destacándose por su capacidad de adaptación y su baja tasa de detección mediante métodos tradicionales.

A lo largo de los años, diversas investigaciones han explorado el uso de técnicas de inteligencia artificial (IA) y aprendizaje automático (machine learning) en la identificación y mitigación de patrones maliciosos en correos electrónicos y sitios web. Estudios pioneros, como los realizados por Smith et al. (2020) y Johnson et al. (2022), han demostrado la eficacia de modelos de clasificación basados en árboles de decisión y redes neuronales para detectar ataques de phishing con una alta precisión, alcanzando tasas de acierto superiores al 90%. Estos modelos han sido fundamentados en teorías de aprendizaje supervisado y en el análisis de grandes volúmenes de datos, abordando el problema desde una perspectiva matemática y computacional.

Históricamente, el desarrollo de estas herramientas ha estado marcado por importantes hitos. Desde la implementación de las primeras soluciones de filtrado de spam en los años 2000, hasta los avances más recientes en detección de URLs maliciosas, la evolución ha estado guiada por la necesidad de adaptarse a las técnicas cada vez más sofisticadas utilizadas por los atacantes. Sin embargo, a pesar de los avances logrados, persisten desafíos críticos. Las soluciones existentes a menudo carecen de la capacidad para adaptarse rápidamente a nuevas tácticas de ataque, lo que deja a los usuarios expuestos ante métodos de ataque emergentes que no son reconocidos por los sistemas preexistentes (Dhole et al., 2023).

El desarrollo de plugins y extensiones de navegador diseñados para mitigar estas amenazas ha sido otro campo de investigación activo. Herramientas como PhishTank y Google Safe Browsing han intentado abordar el problema mediante la comparación de URLs con bases de datos centralizadas de sitios maliciosos. No obstante, estas soluciones a menudo enfrentan limitaciones significativas, como la dependencia de actualizaciones constantes de dichas bases de datos y la incapacidad para predecir nuevos ataques antes de que se registren en estas listas negras (Rasha et al., 2023).

Este proyecto de tesis se fundamenta en la necesidad de superar las limitaciones identificadas en la literatura existente. Se propone el desarrollo de un plugin de navegador que no solo detecte ataques conocidos de ingeniería social, sino que también posea la

capacidad de aprender y adaptarse dinámicamente a nuevas amenazas mediante el uso de técnicas de inteligencia artificial. Utilizando TensorFlow.js, una librería de JavaScript para el entrenamiento y la ejecución de modelos de machine learning directamente en el navegador, se busca crear una solución ligera y flexible. Esta solución debe ser capaz de proporcionar protección en tiempo real sin comprometer la fluidez del navegador ni invadir la privacidad del usuario, evitando la recolección de datos sensibles.

El plugin propuesto también optimizará el procesamiento local y permitirá la actualización continua del modelo desde un servidor central, mejorando así su rendimiento y adaptabilidad frente a amenazas emergentes. Esta investigación tiene como objetivo contribuir al campo de la ciberseguridad al desarrollar un enfoque innovador que no solo proteja a los usuarios en tiempo real, sino que también evolucione junto con las amenazas, abordando así uno de los principales desafíos en la defensa contra ataques de ingeniería social.

Formulación del problema de investigación

La ingeniería social se refiere a las técnicas que utiliza un atacante para inducir a su objetivo a cumplir sus órdenes. En el caso de un ataque phishing, el atacante utiliza algún tipo de plataforma de mensajería para enviar enlaces, archivos adjuntos maliciosos u otro tipo de contenido engañoso, tentador o amenazador al destinatario con el fin de conseguir que cumpla las órdenes del atacante. (*Ingeniería social frente a phishing - Check Point Software, s/f*)

La ingeniería social, particularmente a través de ataques de phishing, representa una amenaza creciente y compleja en el ámbito de la seguridad informática. El phishing se basa en técnicas de ingeniería social diseñadas para engañar a las víctimas y hacer que divulguen datos personales, los cuales luego se utilizan para la suplantación de identidad en diversas plataformas, incluyendo sitios web y entidades financieras. Este tipo de ataque no solo pone en riesgo la pérdida de información sensible, como contraseñas, datos bancarios y detalles de tarjetas de crédito, sino que también puede tener graves consecuencias económicas y psicológicas para las víctimas (Díaz, 2021). En el servicio de correo Zimbra no existe un complemento gratuito que aporte significativamente a su detección, existen opciones pagadas que ayudan a mejorar la detección.

En Ecuador en la ciudad de Guayaquil, alrededor de 44 500 correos electrónicos de la Asamblea Nacional fueron bloqueados y retenidos. Ocurrió después de que la cuenta de la Secretaría General del Parlamento ecuatoriano fue afectada por un ataque informático conocido como phishing. Así consta en un informe técnico sobre el ataque informático en la Asamblea, realizado por la coordinación general de tecnologías de la información y comunicación del Parlamento (EL COMERCIO, 2022)

De acuerdo a la telemetría de ESET (2023), el mayor número de afectados se localizan en gran parte en Ecuador siendo además de Latinoamérica, como México, Argentina, Chile, Perú y Brasil. Estos ataques ocurren a través de un correo electrónico con una página de phishing en un archivo HTML adjunto, el mensaje advierte al usuario sobre una actualización del servidor, desactivación de la cuenta, o un asunto similar, y le ordena hacer clic en el archivo adjuntado. El atacante también falsifica el campo “From:” para que parezca procedente del administrador del servidor (welivesecurity, 2023).

Un equipo de investigación descubrió una campaña masiva de phishing activa desde al menos abril de 2023, destinada a recolectar credenciales de cuenta de usuarios de Zimbra Collaboration. La campaña se está difundiendo masivamente y sus objetivos son una variedad de pequeñas y medianas empresas, como también entidades gubernamentales. (*Revista Gestión | Ecuador atacado por una campaña masiva de phishing*, s/f 2020)

Diferentes campañas llegaron en múltiples oleadas en dos fases de ataque, la fase inicial estaba dirigida al reconocimiento, e involucraba correos electrónicos diseñados para simplemente rastrear si un objetivo recibió y abrió los mensajes. La segunda fase se produjo en varias oleadas, las cuales contenían mensajes de correo electrónico que atraían a los objetivos para que hicieran clic en un enlace creado por un atacante malicioso (ECUCERT, 2022).

A demás oleadas de phishing subsiguientes han aprovechado cuentas de empresas legítimas previamente atacadas, lo que sugiere que las cuentas de administradores infiltrados asociadas con esas víctimas se utilizaron para enviar correos electrónicos a otras entidades de interés. «Una explicación es que el adversario confía en la reutilización de contraseñas por parte del administrador al que se dirigió a través del phishing, es decir,

el uso de las mismas credenciales tanto para el correo electrónico como para la administración», señaló Šperka. (2023)

El aumento en las transacciones en línea, tales como ventas, servicios bancarios y pagos de servicios básicos, ha llevado a una mayor dependencia de la tecnología y, consecuentemente, a una mayor exposición a fraudes informáticos. La falta de una legislación robusta y específica contra el phishing en el país ha contribuido a la impunidad de los delincuentes, quienes se benefician de un entorno legal inadecuado para abordar estas amenazas. Esta deficiencia legislativa no solo afecta la seguridad de los usuarios, sino que también incrementa el temor de utilizar herramientas tecnológicas esenciales en la vida cotidiana (Ibarra et al., 2024; Arango, 2023).

El “Hospital Especializado Julio Endara”, ubicado en Aut. Gneral. Rumiñahui / Puente 7Av. Manuela Cañizares Oe3-376, Quito, Ecuador, es una institución de tercer nivel especializada en salud mental comprometidos con las políticas del Ministerio de Salud Pública (MSP), orientadas a la atención del paciente dentro y fuera de su grupo familiar, evitando así la institucionalización (MSP, 2021), tiene como una de las herramientas de trabajo para comunicación interna de la institución se encuentra la plataforma web de código abierto Zimbra, el cual gestiona la mensajería a través de correos electrónicos entre los usuarios del hospital.

Mediante una entrevista realizada a diferentes usuarios en la institución se pudo comprobar que al utilizar el servicio de mensajería Zimbra su privacidad está expuesta a diferentes ataques cibernéticos que pretenden extraer información sensible a través de correos fraudulentos y falsos (MSP, 2021)

El problema se agrava con la existencia de herramientas y plugins de seguridad que, aunque son útiles en la detección de amenazas conocidas, a menudo presentan limitaciones significativas en cuanto a su capacidad de adaptarse a nuevas tácticas de phishing. Estos métodos suelen depender de técnicas estáticas que no evolucionan con el cambio constante en las estrategias de ataque. Además, la capacidad de procesamiento local en los navegadores y la necesidad de proteger la privacidad del usuario sin comprometer la eficiencia son desafíos adicionales que las soluciones actuales no abordan de manera satisfactoria.

En cuanto a la legislación ecuatoriana, las penas actuales para delitos informáticos están establecidas en el Código Orgánico Integral Penal (COIP), específicamente en el Artículo 234 y el Artículo 236. Estas disposiciones imponen penas que van desde uno hasta tres años de prisión para quienes cometan delitos de acceso no autorizado a sistemas informáticos y desde tres hasta cinco años para quienes cometan fraudes informáticos graves, respectivamente. Sin embargo, la legislación aún es insuficiente para abordar la amplia gama de técnicas de phishing y otros ataques de ingeniería social (COIP, 2021).

Dado este contexto, surge la necesidad de desarrollar un enfoque más dinámico y adaptable para la protección contra el phishing. La integración de técnicas avanzadas de inteligencia artificial, como el uso de TensorFlow.js para la detección en tiempo real, puede ofrecer una solución innovadora. Sin embargo, la capacidad para implementar un modelo que no solo detecte patrones de ataque conocidos, plantea un desafío significativo. La solución debe superar problemas como la optimización del procesamiento local en el navegador, la protección de la privacidad del usuario y la actualización dinámica del modelo sin comprometer la fluidez del navegador.

Planteamiento Hipotético

La metodología de investigación utilizada en este estudio se basa en un estudio transversal que permite analizar el impacto del uso de un plugin desarrollado de seguridad inteligente para navegadores web en la implementación de una defensa contra la ingeniería social mediante inteligencia artificial y aprendizaje autónomo. Se llevaron a cabo diferentes etapas para alcanzar los objetivos planteados, incluyendo el diseño de la investigación, selección de la muestra, recolección de datos y análisis de estos.

Pregunta principal

¿Cómo se puede desarrollar un plugin de navegador efectivo y ético para detectar y prevenir ataques de ingeniería social, protegiendo la privacidad y seguridad de los usuarios mientras navegan por internet?

Preguntas específicas

¿Cuál es la efectividad actual de las herramientas disponibles para defenderse contra la ingeniería social en navegadores web y qué limitaciones posee?

¿Cómo han evolucionado los métodos y tácticas de los ataques de ingeniería social en línea en los últimos años y qué nuevas tendencias se han observado?

¿Qué papel juega la inteligencia artificial y el aprendizaje automático en la detección y prevención de ataques de ingeniería social en navegadores web y qué avances recientes se han logrado en este campo?

¿Qué expectativas y necesidades de los usuarios en cuanto a seguridad en línea y protección contra la ingeniería social mientras navegan por la web y cómo podrían abordarse de manera efectiva?

Objetivo General:

Desarrollar un Plugin de Seguridad Inteligente en navegadores para la reducción de los riesgos asociados con la ingeniería social en el ambiente de navegación en línea para detección de intentos ataques comunes de ingeniería social en los usuarios de correo zimbra del Hospital Julio Endara.

Objetivos Específicos:

Identificar las tácticas más comunes de ingeniería social utilizadas en entornos de navegación web, como phishing, pretexting y spear phishing.

Analizar los mecanismos y técnicas empleadas por los atacantes para atacar con ingeniería social en línea, incluyendo correos electrónicos falsificados, sitios web fraudulentos y perfiles de redes sociales manipulados.

Implementar un plugin con inteligencias artificial en un navegador web para la detección de intentos de ataques comunes de ingeniería social.

Evaluar el impacto de las amenazas de ingeniería social en la seguridad y privacidad de los usuarios durante la navegación en línea, considerando el robo de información personal, contraseñas y datos financieros sensibles.

Planteamiento hipotético

Si se utiliza un plugin de inteligencia artificial en el navegador web chrome mejorará significativamente la detección de intentos de ingeniería social en el cliente de correo zimbra.

CAPÍTULO 1. MARCO TEÓRICO REFERENCIAL

1.1. Revisión de literatura

La ingeniería social se ha consolidado como una de las tácticas más utilizadas por los cibercriminales, con técnicas que manipulan psicológicamente a las víctimas para obtener información confidencial o acceso no autorizado a sistemas. Entre las más comunes, el phishing y spear phishing se destacan por su efectividad al explotar la confianza del usuario a través de correos electrónicos fraudulentos, mensajes urgentes o falsos enlaces (Prado Díaz J. P., 2021). Estos métodos de ataque no solo ponen en riesgo la información personal, sino que también pueden tener repercusiones económicas y psicológicas graves.

En Ecuador, plataformas como Zimbra, ampliamente utilizada por instituciones gubernamentales y organizaciones, han sido blanco de ataques de phishing masivos, lo que subraya la vulnerabilidad del ecosistema digital del país (EL COMERCIO, 2022). Este contexto ha impulsado la necesidad de desarrollar herramientas de seguridad avanzadas capaces de detectar y mitigar amenazas emergentes.

El uso de la inteligencia artificial (IA) ha demostrado ser efectivo en la detección de ataques de ingeniería social. Modelos de aprendizaje automático, como los árboles de decisión y las redes neuronales, han alcanzado altos niveles de precisión en la identificación de phishing. Estudios recientes destacan que estas técnicas logran tasas de acierto superiores al 90% (Smith et al., 2020; Johnson et al., 2022). Además, algoritmos de procesamiento de lenguaje natural (PLN) permiten analizar y clasificar correos electrónicos de manera eficiente, identificando patrones sospechosos y evitando falsos positivos (Dhole et al., 2023).

A pesar de los avances, las soluciones actuales, como PhishTank y Google Safe Browsing, tienen limitaciones debido a su dependencia de listas de sitios maliciosos previamente conocidos. Esto dificulta la identificación de ataques no registrados y reduce su capacidad para adaptarse rápidamente a nuevas tácticas de los atacantes (Rasha et al., 2023).

El proyecto de tesis propone superar estas limitaciones a través del desarrollo de un plugin que utilice TensorFlow.js, permitiendo la detección en tiempo real y la actualización

dinámica del modelo de IA en el navegador. Esta solución busca combinar las ventajas del procesamiento local, protegiendo la privacidad del usuario, con la capacidad de adaptarse a nuevas amenazas mediante la actualización continua del modelo desde un servidor central. Además, este enfoque tiene como objetivo minimizar falsos positivos y optimizar la experiencia de usuario al no comprometer la fluidez del navegador (Rincón Nuñez P. M., 2023).

El estado del arte sobre la detección de ataques de ingeniería social demuestra la creciente importancia de integrar soluciones basadas en inteligencia artificial que sean adaptativas, eficientes y que respeten la privacidad de los usuarios. Herramientas como el plugin propuesto en este proyecto podrían marcar una diferencia significativa en la protección contra estas amenazas emergentes.

1.2 Introducción a la Ingeniería Social

1.2.1. Definición y conceptos básicos

La ingeniería social se refiere al uso de manipulación psicológica para influir en el comportamiento de las personas, con el fin de obtener acceso no autorizado a información confidencial o recursos críticos. En el contexto de la ciberseguridad, este tipo de ataque se caracteriza por aprovechar las vulnerabilidades humanas, como la confianza excesiva o la falta de atención, en lugar de explotar directamente debilidades tecnológicas. Según estudios recientes, estos ataques son cada vez más sofisticados, empleando estrategias avanzadas que combinan técnicas de suplantación de identidad (phishing) y otras tácticas engañosas (Benavides-Astudillo et al., 2020).

El concepto de ingeniería social se remonta a la antigüedad, pero en la era digital ha cobrado una relevancia especial. A diferencia de los ciberataques puramente técnicos, la ingeniería social se centra en el eslabón más débil de la cadena de seguridad: el usuario. Un atacante puede utilizar una llamada telefónica falsa, un correo electrónico convincente o una interacción en redes sociales para engañar a la víctima y persuadirla de que revele contraseñas, datos financieros u otra información sensible (COIP, 2021). Este enfoque humano y social es lo que distingue a la ingeniería social de otros métodos de intrusión digital tal vez sea el más sencillo de ejecutar, pero el más eficaz una vez alcanzado la entrega de credenciales o información sensible de cuentas y claves.

1.2.2 Técnicas comunes de ingeniería social

Existen varias técnicas comunes que los ciberdelincuentes utilizan en ataques de ingeniería social. El phishing es, sin duda, una de las formas más extendidas. Consiste en enviar correos electrónicos o mensajes falsos que parecen ser de fuentes confiables, pero que en realidad están diseñados para robar información confidencial, como contraseñas o números de tarjetas de crédito (Dhole et al., 2023). Los mensajes de phishing a menudo contienen enlaces a sitios web fraudulentos que imitan páginas legítimas o incluyen archivos adjuntos maliciosos. Esta técnica ha sido responsable de un gran número de violaciones de datos y sigue siendo uno de los métodos preferidos por los atacantes (Chang, 2020).

Otra técnica importante es el pretexting, en la que el atacante se hace pasar por una autoridad legítima o una entidad de confianza para obtener información confidencial de la víctima. Esto puede suceder a través de una llamada telefónica o una interacción por correo electrónico, donde el atacante utiliza un contexto falso para convencer a la víctima de que comparta datos (Dhole et al., 2023). El spear phishing, una versión más dirigida del phishing, utiliza información personal previamente obtenida sobre la víctima para hacer el ataque más creíble y aumentar las posibilidades de éxito.

El baiting es otra táctica común en la que el atacante ofrece algo tentador, como una descarga gratuita o un dispositivo físico, para atraer a la víctima a descargar malware o comprometer su sistema. Los atacantes también recurren al tailgating o "piggybacking", donde se aprovechan de la proximidad física de la víctima para obtener acceso a áreas restringidas o dispositivos protegidos (Díaz, 2021). Estas técnicas son solo algunas de las muchas que existen, pero todas comparten el objetivo común de explotar la confianza y la falta de cautela de las personas.

1.2.3. Impacto de la ingeniería social en la ciberseguridad

El impacto de los ataques de ingeniería social en la ciberseguridad es significativo y va más allá de la simple pérdida de datos. Estos ataques han demostrado ser devastadores tanto para individuos como para organizaciones, generando consecuencias que incluyen pérdidas financieras, daños a la reputación y la interrupción de operaciones críticas. En

muchos casos, una sola brecha causada por ingeniería social puede tener un efecto dominó, afectando a socios, clientes y otros actores vinculados a la entidad afectada. De hecho, estudios indican que la mayoría de las violaciones de datos involucran algún componente de ingeniería social, subrayando la gravedad del problema (Vega Velasco, 2008).

A nivel organizacional, los ataques de ingeniería social son particularmente peligrosos porque a menudo eluden las defensas técnicas, como firewalls o sistemas de detección de intrusiones, al atacar directamente al personal. En el entorno corporativo, empleados con acceso privilegiado pueden ser manipulados para revelar credenciales críticas, lo que abre las puertas a ataques más sofisticados como el ransomware o el espionaje corporativo. Las empresas que no capacitan adecuadamente a su personal sobre los riesgos de la ingeniería social corren un riesgo considerable de sufrir este tipo de ataques (EL COMERCIO, 2022).

Además, el impacto psicológico en las víctimas de estos ataques no debe subestimarse. Las personas que son engañadas mediante tácticas de ingeniería social a menudo se sienten culpables o avergonzadas, lo que puede afectar su rendimiento y confianza en el lugar de trabajo (Hassan Montero y Martín Fernández, 2005). Para mitigar estos riesgos, es crucial que las organizaciones implementen programas de concienciación y entrenamiento continuo en ciberseguridad, que no solo se enfoquen en la protección técnica, sino también en la preparación ante el factor humano, el más vulnerable en estos escenarios.

1.3 Clasificación de Usuarios y Vulnerabilidades

1.3.1. Usuarios novatos y sus debilidades

Los usuarios novatos representan el grupo más vulnerable ante los ataques de ingeniería social debido a su falta de experiencia en el uso de tecnologías digitales y prácticas de seguridad en línea. Este grupo generalmente incluye a personas que tienen poca exposición a las amenazas cibernéticas o no han recibido una capacitación adecuada en ciberseguridad. Los atacantes explotan esta inexperiencia mediante técnicas simples y directas, como correos electrónicos de phishing que parecen provenir de entidades

confiables. Un ejemplo típico es un correo electrónico que solicita información personal o financiera, acompañado de un enlace que redirige a una página web maliciosa diseñada para parecer legítima (Hassan Montero y Martín Fernández, 2005).

Una técnica comúnmente utilizada contra los usuarios novatos es el engaño mediante falsas alertas. En este caso, los atacantes envían mensajes que advierten a los usuarios sobre supuestos problemas en sus dispositivos, como infecciones por malware, incitándolos a descargar software fraudulento que, en realidad, es un programa malicioso. Estos métodos aprovechan la falta de conocimiento del usuario sobre cómo identificar alertas legítimas de sistemas falsos. La curiosidad y la ansiedad por solucionar un problema de seguridad inventado hacen que los novatos caigan fácilmente en este tipo de trampa (Kaspersky Co.asd. , 2021).

En términos de mitigación, es crucial que las organizaciones ofrezcan programas de capacitación diseñados específicamente para este grupo. Los usuarios novatos necesitan familiarizarse con las amenazas comunes, aprender a verificar la autenticidad de los mensajes y entender los principios básicos de seguridad en línea, como el uso de contraseñas fuertes y la autenticación de dos factores. Estas medidas pueden reducir significativamente el riesgo de que sean víctimas de ataques de ingeniería social (Rincón Nuñez P. M., 2023).

1.3.2. Usuarios medios y phishing sofisticado

Los usuarios medios tienen una mayor comprensión de la tecnología y suelen estar más familiarizados con los conceptos básicos de seguridad digital. Sin embargo, este conocimiento parcial puede generar una falsa sensación de seguridad, lo que los hace susceptibles a ataques más elaborados, como el phishing sofisticado. A diferencia de las técnicas simples, el phishing sofisticado emplea correos electrónicos bien redactados y sitios web falsificados que imitan de manera casi exacta a los sitios legítimos, engañando incluso a aquellos con experiencia moderada en ciberseguridad (Rasha et al., 2023).

Este tipo de ataque suele incluir la suplantación de identidad de empresas conocidas o entidades financieras. Los atacantes pueden personalizar el contenido de los correos electrónicos, mencionando detalles específicos sobre el usuario o sus interacciones

previas con la empresa, lo que hace que los mensajes parezcan auténticos. Un correo de phishing sofisticado puede incluir enlaces que conducen a páginas web aparentemente seguras con certificados SSL válidos, lo que incrementa la confianza del usuario medio y reduce su escepticismo ante la posibilidad de un ataque (Trejos et al., 2021).

Para proteger a los usuarios medios, es necesario capacitarlos en la identificación de detalles más sutiles de los ataques de phishing, como la revisión de la URL de los enlaces, la autenticidad de las direcciones de correo electrónico y la importancia de no compartir información personal a través de canales no seguros. Además, las herramientas de protección como filtros de correo avanzados y software de seguridad especializado pueden ayudar a mitigar los riesgos (Rasha et al., 2023).

1.3.3. Usuarios frecuentes y ataques dirigidos

Los usuarios frecuentes, que suelen interactuar más a menudo con sistemas tecnológicos, como profesionales de diversas áreas, representan un objetivo atractivo para ataques dirigidos. Los atacantes se valen de técnicas como el spear phishing, en las que personalizan los ataques basándose en la información que han recopilado previamente sobre la víctima. Estos usuarios pueden ser empleados de alto nivel o administradores con acceso privilegiado a sistemas críticos, lo que convierte un ataque exitoso en una grave brecha de seguridad (Prado Díaz J. P., 2021).

En estos casos, el atacante realiza una investigación exhaustiva sobre la víctima antes de lanzar el ataque, obteniendo detalles personales, laborales o financieros a través de redes sociales, plataformas profesionales o incluso mediante brechas de datos previas. Este conocimiento detallado permite que el ataque sea extremadamente convincente, lo que reduce las probabilidades de que la víctima sospeche de la legitimidad del mensaje o solicitud. Un ejemplo de esto es cuando el atacante se hace pasar por un colega o superior del usuario, solicitando información confidencial o la realización de transferencias monetarias urgentes.

La mejor defensa para los usuarios frecuentes es la combinación de herramientas tecnológicas avanzadas y la capacitación continua. Dado que estos ataques están altamente personalizados, es esencial que los usuarios adopten una postura crítica frente

a cualquier solicitud inusual, aunque provenga de fuentes aparentemente confiables. Además, el uso de métodos de autenticación adicionales, como la verificación en dos pasos o la confirmación por otros medios, puede ayudar a prevenir que caigan en este tipo de ataques (Vega Velasco, 2008).

1.3.4. Usuarios expertos y ataques especializados

Los usuarios expertos, como los profesionales en tecnología y ciberseguridad, son el grupo menos vulnerable a los ataques de ingeniería social, pero aún pueden ser víctimas de técnicas altamente especializadas. A menudo, estos usuarios subestiman las vulnerabilidades humanas, confiando en exceso en su conocimiento técnico. Sin embargo, los atacantes que apuntan a expertos suelen utilizar tácticas sofisticadas que explotan debilidades psicológicas o vulnerabilidades en los sistemas que no dependen solo de la tecnología (Prado Díaz J. P., 2021).

Una de las tácticas más comunes es la suplantación de autoridad, donde el atacante se hace pasar por una figura superior dentro de la organización o un colega de confianza. A través de un análisis detallado del comportamiento en línea de la víctima, el atacante crea un perfil que le permite generar un mensaje extremadamente creíble. Estos ataques pueden incluir la solicitud de acceso a sistemas críticos o la ejecución de acciones que comprometan la seguridad de la organización. Incluso los usuarios más expertos pueden ser manipulados si no están atentos a los signos más sutiles de una amenaza de ingeniería social (Prado Díaz J. P., 2021).

Para mitigar estos riesgos, es crucial que los expertos adopten medidas adicionales de seguridad, como la revisión constante de sus propios procedimientos y la aplicación de controles adicionales en la verificación de solicitudes inusuales (Rasha et al., 2023). La actualización constante en nuevas tácticas de ataque y la creación de una cultura de ciberseguridad robusta dentro de las organizaciones pueden reducir significativamente la probabilidad de éxito de estos ataques especializados. Además, la combinación de una conciencia de la vulnerabilidad humana con su profundo conocimiento técnico es clave para prevenir ataques dirigidos a usuarios expertos. No basta con ser un experto; es crucial fomentar una cultura organizacional que promueva la excelencia y el aprendizaje continuo de todo el personal, especialmente en la identificación de ataques mediante

correos electrónicos. Esto implica capacitaciones periódicas que mantengan a los empleados actualizados sobre las nuevas tácticas de ingeniería social y amenazas emergentes, garantizando una defensa más sólida frente a posibles ataques.

1.4 El Rol de los Hackers en la Ingeniería Social

1.4.1. Tipos de hackers (white hat, black hat, grey hat)

Los hackers pueden clasificarse en tres grandes categorías según sus intenciones y las acciones que llevan a cabo: white hat, black hat y grey hat. Los **white hat** son los llamados "hackers éticos", profesionales que trabajan para empresas o instituciones con el fin de identificar vulnerabilidades en sus sistemas y fortalecer la seguridad de la información. Estos hackers realizan pruebas de penetración y buscan explotar las debilidades antes de que los atacantes maliciosos lo hagan, contribuyendo así a la mejora de la ciberseguridad (EL COMERCIO, 2022).

Por otro lado, los **black hat** son los hackers malintencionados que utilizan sus habilidades para ganar acceso no autorizado a sistemas con fines ilícitos. A menudo participan en actividades como el robo de datos, el espionaje industrial o la interrupción de servicios a cambio de beneficios económicos o para dañar la reputación de la víctima. En el contexto de la ingeniería social, los black hat manipulan psicológicamente a sus objetivos para extraer información que luego utilizan para vulnerar los sistemas de seguridad (Johnson et al., 2022).

Finalmente, los **grey hat** se sitúan en un punto intermedio entre los White hat y los black hat. Estos hackers suelen explorar vulnerabilidades sin autorización, pero sin la intención de dañar directamente a la víctima. A menudo, tras descubrir un fallo, se lo informan a la empresa o entidad afectada, aunque en algunos casos pueden exigir una compensación por su descubrimiento (Prado Díaz J. P., 2021). Los grey hat pueden actuar tanto de forma ética como ilegal, dependiendo de las circunstancias y sus motivaciones, pero no suelen explotar los errores con fines maliciosos a gran escala como los black hat.

1.4.2. Técnicas de hacking social

El hacking social es un conjunto de técnicas que los hackers utilizan para manipular a las personas con el fin de obtener acceso a información confidencial o sistemas protegidos.

Una de las tácticas más comunes es el **phishing**, que implica el envío de correos electrónicos falsos que parecen ser de una fuente confiable. Estos correos contienen enlaces o archivos maliciosos diseñados para engañar al usuario y hacer que revele credenciales de acceso o información personal. El phishing se ha vuelto más sofisticado con el tiempo, incorporando técnicas avanzadas como el spear phishing, que se enfoca en un individuo específico utilizando información personalizada (Smith et al., 2020).

Finalmente, el **tailgating** o “piggybacking” es una técnica física en la que el hacker aprovecha el acceso físico a las instalaciones. En muchos casos, el atacante sigue a un empleado a través de una puerta segura o utiliza una falsa identidad para obtener acceso a áreas restringidas (ESET , 2023). El hacking social explota la confianza natural de las personas y su falta de conciencia sobre los riesgos de seguridad, convirtiéndose en una amenaza efectiva tanto en el entorno digital como en el físico.

1.4.3. Motivaciones y objetivos de los hackers

Las motivaciones de los hackers que emplean ingeniería social pueden variar desde el beneficio económico hasta el deseo de causar daño reputacional o disrupción. Para los **black hat**, el objetivo principal es generalmente financiero. A través del robo de datos personales o bancarios, estos hackers pueden cometer fraudes financieros, vender información en el mercado negro o chantajear a las víctimas. Otro objetivo común es la **extorsión mediante ransomware**, donde el atacante infecta el sistema de una víctima con malware que bloquea sus archivos y exige un rescate para liberarlos (Rincón Nuñez P. M., 2023).

En otros casos, las motivaciones pueden ser más ideológicas, lo que se conoce como **hacktivismo**. Los hackers que se alinean con una causa política o social pueden llevar a cabo ataques de ingeniería social para exponer o perjudicar a organizaciones gubernamentales o corporaciones que consideren que están actuando de manera injusta. En estos casos, la intención no es necesariamente económica, sino más bien generar un impacto público, socavar la credibilidad de la entidad atacada o promover una agenda política (Rincón Nuñez P. M., 2023).

Finalmente, algunos hackers, particularmente los **grey hat**, pueden estar motivados por la búsqueda de reconocimiento o la curiosidad intelectual. Estos hackers pueden realizar ataques de ingeniería social con el fin de demostrar sus habilidades o encontrar vulnerabilidades por las cuales esperan ser reconocidos por las empresas. Independientemente de la motivación, el objetivo final de los ataques de ingeniería social es explotar la vulnerabilidad humana para ganar acceso a información o sistemas críticos (Sperka, 2023).

1.5 Protecciones Actuales contra la Ingeniería Social

1.5.1. Educación y concienciación

Una de las formas más efectivas de combatir los ataques de ingeniería social es mediante la **educación y concienciación** de los usuarios. Dado que la mayoría de estos ataques se dirigen a explotar vulnerabilidades humanas, es fundamental que los empleados y usuarios finales sean conscientes de las tácticas comunes que los atacantes pueden utilizar, como el phishing, el pretexting o el baiting. La capacitación regular y actualizada sobre ciberseguridad ayuda a identificar estos intentos de manipulación, lo que reduce considerablemente el riesgo de éxito (Neuros Center. , 2023).

Los programas de concienciación deben ir más allá de los conceptos básicos de seguridad y abarcar ejemplos reales de ataques recientes, enseñando a los usuarios cómo reconocer señales de alerta, como correos electrónicos sospechosos, solicitudes urgentes de información confidencial o enlaces falsificados. Además, la educación debe incluir simulaciones de ataques de phishing para probar la respuesta de los usuarios en un entorno controlado, lo que permite evaluar la efectividad del programa de capacitación y ajustar los esfuerzos según sea necesario (Trejos et al., 2021).

La concienciación no debe ser un evento único, sino un esfuerzo continuo que evoluciona a medida que cambian las amenazas. Es fundamental que las organizaciones mantengan a sus empleados informados sobre las tácticas de ingeniería social emergentes y ofrezcan recursos actualizados para prevenir estos ataques. Con un enfoque proactivo en la educación, las organizaciones pueden crear una cultura de ciberseguridad más robusta, donde los empleados son una primera línea de defensa contra los ataques.

1.5.2. Políticas de seguridad

Las **políticas de seguridad** bien definidas son otro componente clave para proteger a las organizaciones de los ataques de ingeniería social. Estas políticas deben establecer protocolos claros sobre cómo manejar la información confidencial y cómo responder ante posibles intentos de manipulación. Por ejemplo, las políticas de la organización pueden requerir que todas las solicitudes de información sensible sean verificadas a través de múltiples canales de comunicación antes de ser aprobadas, lo que introduce una capa adicional de seguridad (Prado Díaz J. P., 2021).

La **autenticación multifactor** (MFA) también debe ser una parte integral de las políticas de seguridad. La MFA obliga a los usuarios a verificar su identidad utilizando más de un método, como una contraseña y un código enviado a su teléfono móvil, lo que hace mucho más difícil que un atacante tenga éxito en sus intentos de acceder a sistemas sensibles. Estas políticas también deben abarcar la gestión de contraseñas, alentando el uso de contraseñas complejas y únicas para cada cuenta, además de herramientas como gestores de contraseñas para evitar que los empleados reutilicen credenciales (Dhole et al., 2023).

Las políticas de seguridad deben ser revisadas y actualizadas regularmente para adaptarse a nuevas amenazas. Además, es importante que las organizaciones implementen mecanismos de control para garantizar que las políticas sean cumplidas (Vega Velasco, 2008). La falta de adherencia a las políticas de seguridad puede abrir brechas significativas en la protección de los sistemas, por lo que su correcta aplicación es esencial para prevenir ataques de ingeniería social.

1.5.3. Capas de seguridad tecnológica

Además de la educación y las políticas internas, la protección contra la ingeniería social requiere la implementación de capas de seguridad tecnológica que dificulten la efectividad de los ataques. Entre estas tecnologías se incluyen los filtros de correo electrónico, que pueden detectar y bloquear mensajes de phishing antes de que lleguen a los usuarios. Estas soluciones analizan el contenido de los correos electrónicos en busca de patrones sospechosos o enlaces a sitios web maliciosos, y pueden marcar los mensajes sospechosos como spam o phishing (Johnson et al., 2022).

Los software de detección de malware y los firewalls son también componentes críticos que ayudan a proteger a los usuarios y a las organizaciones frente a intentos de ingeniería social. Estas herramientas monitorizan el tráfico de la red y los archivos que se descargan, previniendo que el malware entregado a través de ataques de phishing infecte los sistemas de la empresa. Además, los sistemas de detección de intrusiones (IDS) pueden identificar comportamientos anómalos en la red que podrían indicar un ataque en curso (Dhole et al., 2023).

Sin embargo, es importante destacar que ninguna de estas herramientas es completamente infalible. Por ello, la mejor estrategia es utilizar varias capas de seguridad combinadas con la capacitación adecuada. El enfoque de seguridad en capas crea una defensa en profundidad que puede prevenir que los atacantes accedan a sistemas críticos incluso si una capa de protección falla.

1.5.4. Verificación de la identidad

La **verificación de la identidad** es un mecanismo crucial para combatir los intentos de suplantación y fraude que se producen a través de la ingeniería social. Las empresas deben implementar políticas que requieran la autenticación de las identidades de las personas que solicitan información confidencial o acceso a sistemas críticos. Esto puede lograrse a través de **métodos de verificación multifactor** (MFA), donde se solicitan varios elementos de prueba, como una contraseña, un token de seguridad o un reconocimiento biométrico, para confirmar la identidad de un usuario (Rasha et al., 2023).

Otro aspecto importante de la verificación es la implementación de protocolos de confirmación para las solicitudes inusuales. Por ejemplo, si un empleado recibe una solicitud urgente para transferir fondos o cambiar credenciales, se debe verificar dicha solicitud a través de un canal alternativo, como una llamada telefónica directa a la persona que supuestamente hizo la solicitud. Esta práctica previene muchos de los ataques de suplantación de identidad que se basan en crear una falsa sensación de urgencia o autoridad (Smith et al., 2020).

En entornos corporativos, la verificación de identidad debe extenderse también a los proveedores y colaboradores externos. Los ataques a la cadena de suministro a menudo

comienzan con la manipulación de terceros con acceso legítimo a la infraestructura de la organización. Asegurar que todas las interacciones estén respaldadas por una verificación sólida es fundamental para mitigar el riesgo de ataques exitosos.

1.5.5. Actualizaciones y parches

Mantener los sistemas actualizados con las últimas **actualizaciones y parches de seguridad** es fundamental para protegerse contra los ataques de ingeniería social, ya que muchos de estos ataques explotan vulnerabilidades conocidas en el software o los sistemas operativos. Los desarrolladores de software lanzan regularmente parches para corregir fallos de seguridad, y es crucial que las organizaciones instalen estas actualizaciones lo antes posible para reducir el riesgo de explotación (Rios-Paredes y Rios-Salgado, 2020).

Los ataques de ingeniería social a menudo dependen de técnicas de manipulación para que las víctimas instalen malware o accedan a sistemas sin parches. Por ejemplo, si un empleado es engañado para descargar un archivo malicioso en un sistema que no ha sido actualizado, el malware podría aprovechar una vulnerabilidad que ya habría sido corregida mediante un parche. Por lo tanto, la diligencia en la aplicación de actualizaciones es una línea de defensa importante contra estos ataques (Rincón Nuñez P. M., 2023).

Además, la automatización de las actualizaciones y el uso de herramientas de administración de parches pueden ayudar a garantizar que los sistemas permanezcan protegidos de forma continua. Las organizaciones deben establecer políticas que obliguen a aplicar las actualizaciones de seguridad tan pronto como estén disponibles y realizar auditorías periódicas para verificar que todos los sistemas estén al día.

1.6 Detección de Ataques de Ingeniería Social

1.6.1. Señales de detección en correos electrónicos y mensajes

La detección de ataques de ingeniería social a través de correos electrónicos y mensajes es esencial, ya que muchos de estos ataques, como el phishing, comienzan con una simple comunicación escrita. Existen varias señales que pueden ayudar a los usuarios a identificar estos intentos maliciosos. Una de las primeras señales es el **remite**

sospechoso. Los correos electrónicos fraudulentos a menudo provienen de direcciones que, aunque aparentan ser legítimas, contienen errores menores, como dominios mal escritos o variaciones leves del nombre de una empresa reconocida. Además, los atacantes suelen falsificar el campo "De" para que parezca proveniente de una fuente confiable (Kaspersky Co.asd. , 2021).

Otra señal importante es el **lenguaje utilizado.** Los correos electrónicos de ingeniería social tienden a generar una sensación de urgencia o temor, con frases como “su cuenta será cerrada” o “su información ha sido comprometida”. Estos mensajes están diseñados para presionar al receptor a actuar rápidamente sin verificar la autenticidad de la solicitud. Además, los correos electrónicos de phishing a menudo contienen **enlaces falsificados** que parecen legítimos pero redirigen a sitios web maliciosos. Estos enlaces pueden incluir pequeñas alteraciones en la URL, lo que los hace parecer auténticos a primera vista (Rincón Nuñez P. M., 2023).

Los usuarios también deben estar atentos a los **archivos adjuntos no solicitados.** Los atacantes utilizan archivos adjuntos infectados con malware para comprometer el sistema del receptor. Si un correo electrónico inesperado incluye un archivo adjunto con extensiones sospechosas, como .exe o .zip, se debe evitar su apertura y reportar el mensaje a los administradores de seguridad de la organización. El análisis constante de estos detalles puede prevenir que los usuarios caigan en ataques de ingeniería social basados en mensajes (Rasha et al., 2023).

1.6.2. Detección de ingeniería social en llamadas telefónicas

El pretexting es una técnica común en la ingeniería social que se realiza a través de llamadas telefónicas. Detectar estos intentos de ataque implica prestar atención a varios aspectos clave. Uno de los primeros indicadores de una posible estafa telefónica es cuando el interlocutor solicita información confidencial de manera no usual. Si un supuesto representante de una empresa o entidad solicita contraseñas, detalles de cuentas o datos personales, es fundamental verificar su identidad a través de un canal oficial. Los atacantes suelen hacerse pasar por figuras de autoridad o colegas, y manipulan a la víctima para que revele información bajo la falsa premisa de una emergencia o solicitud legítima (Prado Díaz J. P., 2021).

Otro indicador importante es la falta de detalles específicos. Los atacantes que utilizan llamadas telefónicas para llevar a cabo ingeniería social suelen emplear guiones generales para intentar obtener la máxima información de la víctima. Si el interlocutor se muestra evasivo cuando se le piden detalles precisos sobre la empresa o la naturaleza de la solicitud, esto puede ser una señal de que se trata de un ataque de ingeniería social. Además, las llamadas inesperadas de supuestas entidades con las que no se ha tenido interacción previa deben generar sospechas (Prado Díaz J. P., 2021).

Es recomendable que las organizaciones capaciten a sus empleados para que sigan protocolos estrictos al recibir llamadas que involucren solicitudes de información confidencial. Verificar la identidad del interlocutor mediante el uso de números de contacto oficiales o realizar llamadas de vuelta puede ser una medida efectiva para mitigar los intentos de ingeniería social. Los usuarios deben estar entrenados para desconfiar de cualquier solicitud inusual y nunca compartir información sensible sin realizar una verificación adecuada.

1.6.3. Detección de amenazas en redes sociales y presencia en línea

Las redes sociales se han convertido en un terreno fértil para los ataques de ingeniería social debido a la gran cantidad de información personal que los usuarios suelen compartir. Detectar amenazas en redes sociales implica tener una postura de seguridad proactiva y estar alerta a ciertos comportamientos inusuales. Uno de los principales signos de un ataque de ingeniería social en estas plataformas es la recepción de solicitudes de amistad o conexión de perfiles desconocidos. Los atacantes crean cuentas falsas para obtener acceso a la información personal del usuario o incluso para enviar mensajes con enlaces maliciosos (Prado Díaz J. P., 2021).

Además, los atacantes pueden intentar realizar ataques de spear phishing en redes sociales, en los que utilizan información personal compartida por el usuario, como sus intereses, lugar de trabajo o conexiones, para personalizar mensajes engañosos y hacerlos parecer más legítimos. Por ejemplo, un atacante puede enviar un mensaje haciéndose pasar por un colega o amigo, pidiendo que se haga clic en un enlace que lleva a un sitio malicioso. Este tipo de ataques son particularmente efectivos, ya que la víctima tiende a

confiar más en los contactos conocidos o que aparentan ser familiares (Neuros Center. , 2023).

Otra señal clave es la actividad sospechosa en las cuentas, como cambios de contraseña no autorizados o publicaciones que el usuario no ha realizado. Esto puede ser una indicación de que la cuenta ha sido comprometida y está siendo utilizada para realizar ataques de ingeniería social en otros usuarios (Pérez-Cubero y Poler, 2020). Para evitar estos ataques, es esencial activar medidas de seguridad adicionales, como la autenticación de dos factores en las redes sociales, y limitar la cantidad de información personal disponible públicamente. También se debe fomentar la cultura de revisar cuidadosamente las solicitudes de amistad y ser cauteloso al hacer clic en enlaces dentro de mensajes privados.

1.7 Inteligencia Artificial en la Detección de Ataques

1.7.1. Algoritmos de aprendizaje automático para la detección de phishing

Los **algoritmos de aprendizaje automático** (machine learning) han revolucionado la capacidad para detectar ataques de phishing al analizar grandes volúmenes de datos y aprender a identificar patrones sospechosos que pueden pasar desapercibidos para los humanos. En la detección de phishing, los algoritmos de machine learning se entrenan con conjuntos de datos que contienen ejemplos de correos electrónicos y mensajes tanto maliciosos como legítimos. A partir de este entrenamiento, los modelos aprenden a clasificar nuevos mensajes basándose en características específicas, como la estructura del correo electrónico, el contenido del mensaje, la dirección del remitente y el comportamiento de los enlaces (Chang, 2020).

Entre los modelos más utilizados para la detección de phishing se encuentran los **árboles de decisión**, las **redes neuronales** y los **métodos de clasificación bayesianos**. Estos modelos analizan factores como la frecuencia de palabras clave relacionadas con estafas, el uso de urgencia en el texto, y patrones comunes en las URLs para identificar si un mensaje es potencialmente peligroso. Además, a medida que el algoritmo se expone a nuevos ejemplos de correos electrónicos, puede mejorar su precisión, lo que permite que el sistema se adapte a nuevas tácticas de phishing (INEC, 2010).

Los sistemas basados en aprendizaje automático no solo son capaces de detectar ataques conocidos, sino también de identificar nuevas amenazas que no han sido previamente registradas en bases de datos de phishing. Esta capacidad de **detección proactiva** es crucial, ya que los atacantes continuamente evolucionan sus tácticas para eludir las soluciones de seguridad tradicionales (Neuros Center. , 2023). A través de un análisis automatizado, los algoritmos de aprendizaje automático pueden marcar correos electrónicos como sospechosos con un alto grado de precisión, reduciendo la probabilidad de que los usuarios caigan en ataques de phishing.

1.7.2. Procesamiento de Lenguaje Natural (PLN)

El **Procesamiento de Lenguaje Natural (PLN)** es una rama de la inteligencia artificial que permite a las máquinas comprender y analizar el lenguaje humano. En el contexto de la detección de ataques de ingeniería social, el PLN se utiliza para analizar el contenido textual de correos electrónicos, mensajes instantáneos y otros medios de comunicación en busca de señales de ataque, como solicitudes sospechosas de información confidencial, el uso de un lenguaje urgente o persuasivo, y el intento de suplantación de identidad. Los sistemas basados en PLN pueden identificar patrones que sugieren que un mensaje ha sido diseñado para manipular psicológicamente al destinatario (Prado Díaz J. P., 2021).

Una técnica común en PLN es el **análisis semántico**, que examina la relación entre las palabras dentro del mensaje y su contexto. Al utilizar este enfoque, los sistemas de PLN pueden determinar si el texto contiene indicios de engaño o manipulación, incluso si las palabras individuales no son necesariamente sospechosas. Por ejemplo, si un mensaje tiene un tono inusualmente urgente o está lleno de inconsistencias, el sistema puede marcarlo como potencialmente peligroso. Además, el PLN puede detectar intentos de phishing mediante el análisis de frases que piden acciones rápidas o el ingreso de credenciales, comparando estos mensajes con patrones previamente reconocidos (Rios-Paredes y Rios-Salgado, 2020).

Otra aplicación del PLN es la **detección de suplantación de identidad**, en la que el sistema analiza si el mensaje pretende ser de alguien conocido, pero utiliza un estilo de escritura o vocabulario que no coincide con el remitente real (Kaspersky Co.asd. , 2021). Este tipo de análisis contextual ayuda a identificar correos electrónicos fraudulentos que

imitan a contactos conocidos de la víctima. Al aplicar modelos avanzados de PLN, las organizaciones pueden automatizar el análisis de grandes volúmenes de comunicaciones y prevenir ataques de ingeniería social antes de que el mensaje llegue al usuario final.

1.7.3. Motores de reglas y heurísticas para la evaluación de URLs maliciosas

Los **motores de reglas y heurísticas** son una tecnología clave en la detección de ataques de ingeniería social, particularmente en la identificación de **URLs maliciosas**. Estos motores aplican un conjunto de reglas predefinidas y heurísticas para evaluar la autenticidad de los enlaces presentes en correos electrónicos o sitios web. Por ejemplo, si una URL contiene una combinación sospechosa de caracteres, un dominio poco común o se redirige varias veces antes de llegar a su destino final, el motor de reglas puede identificarla como potencialmente maliciosa (Rincón Nuñez P. M., 2023).

Uno de los principales enfoques heurísticos es la detección de URLs que imitan sitios web legítimos, una técnica conocida como **homógrafos de dominio**. Los atacantes a menudo crean dominios con variaciones mínimas en los nombres de sitios populares, utilizando caracteres similares o substituyendo letras. Estos cambios sutiles pueden engañar a los usuarios que no prestan atención a la URL completa. Los motores de reglas pueden detectar estos pequeños cambios y generar alertas antes de que el usuario haga clic en el enlace. Además, también pueden evaluar si el dominio ha sido recientemente registrado, ya que los atacantes a menudo utilizan sitios recién creados para llevar a cabo campañas de phishing (Sperka, 2023).

Otra técnica heurística es la verificación de la **presencia de certificados SSL** y la autenticidad de los mismos. Aunque los sitios maliciosos pueden utilizar certificados SSL para parecer más seguros, los motores de reglas pueden identificar certificados que son autogenerados o que no coinciden con la entidad propietaria del dominio. Al combinar estas reglas con métodos heurísticos más avanzados, los motores pueden ofrecer una primera línea de defensa robusta contra URLs maliciosas, protegiendo a los usuarios de caer en sitios diseñados para robar información o distribuir malware (Trejos et al., 2021). Por último, las firmas digitales verificadas y la validación cruzada con listas de entidades confiables fortalecen aún más la detección de amenazas ocultas detrás de certificados aparentemente seguros, pero dependen de actualizaciones constantes.

1.8 Consideraciones Éticas en el Uso de IA

1.8.1. Privacidad y protección de datos del usuario

Una de las principales preocupaciones éticas en el uso de inteligencia artificial (IA) es la **privacidad y protección de datos** del usuario. Los sistemas de IA, especialmente aquellos diseñados para detectar amenazas de ingeniería social, a menudo requieren acceso a grandes volúmenes de datos, incluyendo correos electrónicos, mensajes y patrones de navegación. Esto plantea el riesgo de que se recopilen, procesen o compartan datos sensibles sin el conocimiento o consentimiento del usuario. Es fundamental que las organizaciones que implementan IA establezcan protocolos sólidos de protección de datos para garantizar que la información personal de los usuarios esté protegida contra el acceso no autorizado o su uso indebido (Pérez-Cubero y Poler, 2020).

Las regulaciones como el **Reglamento General de Protección de Datos (GDPR)** en la Unión Europea establecen estándares claros sobre cómo debe gestionarse la privacidad de los datos, exigiendo que se minimice la recolección de información personal y que los datos sean utilizados exclusivamente para los fines establecidos. Además, los sistemas de IA deben garantizar la anonimización o pseudonimización de los datos en la medida de lo posible para evitar la identificación directa de los individuos. Las organizaciones también deben ser transparentes sobre los datos que están siendo utilizados y dar a los usuarios el control sobre su información personal, incluyendo la capacidad de eliminar o rectificar datos incorrectos (SUPERVISOR EUROPEO DE PROTECCION DE DATOS, 2021).

La implementación de políticas de privacidad rigurosas no solo protege a los usuarios, sino que también ayuda a generar confianza en el uso de IA para la detección de amenazas. Los desarrolladores de estos sistemas deben tener en cuenta que cualquier violación de la privacidad puede tener repercusiones legales y reputacionales, además de afectar la efectividad y adopción de las soluciones basadas en IA (Smith et al., 2020).

1.8.2. Transparencia y consentimiento en el uso de plugins

La transparencia y el consentimiento informado son principios éticos esenciales cuando se utilizan plugins de navegadores. Los usuarios tienen el derecho de saber cómo

funcionan estos plugins, qué datos recopilan y cómo se procesan. Esto significa que los desarrolladores deben proporcionar descripciones claras y accesibles del propósito y las capacidades del plugin, sin recurrir a lenguaje técnico complicado que los usuarios no comprendan fácilmente. Al presentar los detalles de forma transparente, las empresas pueden garantizar que los usuarios puedan tomar decisiones informadas sobre si desean utilizar la herramienta o no (Vega Velasco, 2008).

Además de la transparencia, el consentimiento explícito es fundamental. Antes de recopilar cualquier tipo de información del usuario, los plugins deben solicitar permisos de manera clara y sin ambigüedades. Esto implica que los usuarios deben tener la opción de rechazar la recopilación de ciertos tipos de datos sin que esto afecte el uso básico del servicio. La falta de consentimiento adecuado puede llevar a la erosión de la confianza y a problemas legales, especialmente en jurisdicciones que requieren consentimiento informado para el procesamiento de datos, como bajo la GDPR (Trejos et al., 2021).

Para promover una relación más ética y transparente, es importante que los plugins también incluyan mecanismos que permitan a los usuarios visualizar y auditar qué datos están siendo recopilados y cómo están siendo utilizados. Esto fomenta un entorno de confianza en el que el usuario se siente seguro al utilizar el producto, sabiendo que su información está protegida y que tiene el control sobre sus datos personales.

1.8.3. Precisión y manejo de falsos positivos

La **precisión** de los sistemas de IA es crucial para evitar la generación de **falsos positivos**, es decir, cuando un sistema identifica incorrectamente una amenaza que no existe. En el contexto de la detección de ingeniería social, un exceso de falsos positivos puede generar una experiencia negativa para el usuario, llevando a la frustración y, en algunos casos, al abandono del uso del sistema. Además, los falsos positivos pueden disminuir la confianza del usuario en la herramienta y hacer que las verdaderas amenazas sean ignoradas por fatiga de alertas (Rincón Nuñez P. M., 2023).

La ética en el diseño de IA exige que los desarrolladores trabajen continuamente para **minimizar los falsos positivos** y mejorar la precisión del sistema mediante técnicas de aprendizaje automático más avanzadas y el ajuste de los modelos. Esto incluye la

evaluación regular del rendimiento del algoritmo, utilizando datos actualizados para garantizar que el sistema siga siendo efectivo a medida que evolucionan las amenazas. Al mismo tiempo, los desarrolladores deben ofrecer a los usuarios opciones para ajustar los niveles de sensibilidad del sistema, de modo que puedan adaptar el umbral de detección según sus necesidades específicas (ECUCERT, 2022).

Además, es fundamental que los sistemas de IA proporcionen explicaciones claras sobre por qué se ha generado una alerta. Esta capacidad de “explicabilidad” permite a los usuarios entender las razones detrás de una decisión del sistema y les ayuda a diferenciar entre un falso positivo y una amenaza real. La transparencia en el manejo de los falsos positivos no solo mejora la experiencia del usuario, sino que también fortalece la confianza en los sistemas de IA (Neuros Center. , 2023).

1.8.4. Equidad y sesgo en los algoritmos

El sesgo algorítmico es un problema ético significativo en el desarrollo de sistemas de IA. Los algoritmos utilizados para la detección de amenazas pueden aprender patrones incorrectos o estar influenciados por sesgos inherentes a los datos con los que fueron entrenados. Por ejemplo, si un sistema de IA ha sido entrenado principalmente con datos provenientes de un grupo demográfico específico, podría ser menos efectivo o injustamente tendencioso hacia usuarios de otros grupos. Esto podría llevar a que ciertos individuos sean **injustamente marcados** como riesgos, mientras que otros pasen desapercibidos (Dhole et al., 2023).

Para mitigar estos problemas, los desarrolladores deben garantizar que los datos de entrenamiento sean diversos y representativos de todas las poblaciones que utilizarán el sistema. Además, es importante realizar auditorías regulares del algoritmo para identificar cualquier sesgo inherente y corregirlo. Los algoritmos también deben ser revisados para evitar que perpetúen prácticas discriminatorias o generen disparidades en la protección que ofrecen (Trejos et al., 2021).

El compromiso con la **equidad en los algoritmos** también incluye la responsabilidad de garantizar que todas las personas tengan acceso a las mismas protecciones de seguridad, sin importar su origen, género o ubicación (Smith et al., 2020). Es esencial que las

soluciones de IA sean inclusivas y no amplifiquen desigualdades existentes en el acceso a la tecnología o a la ciberseguridad.

1.8.5. Responsabilidad en el desarrollo de sistemas de IA

La **responsabilidad** en el desarrollo y despliegue de sistemas de IA es una consideración ética crítica. Los desarrolladores y las organizaciones que implementan IA deben asumir la responsabilidad por el comportamiento de los sistemas que diseñan, asegurando que operen de manera segura, ética y conforme a la ley. Esto implica no solo garantizar que los sistemas funcionen como se espera, sino también prever las consecuencias no deseadas que pueden surgir de su uso o mal uso (Rincón Nuñez P. M., 2023).

Los sistemas de IA para la detección de amenazas de ingeniería social deben ser rigurosamente probados antes de su implementación, y las organizaciones deben establecer mecanismos claros para la **rendición de cuentas** en caso de fallos o errores. Además, los usuarios finales deben tener acceso a soporte técnico y herramientas para reportar problemas o disputas relacionadas con el comportamiento del sistema. En caso de errores, las empresas deben ser transparentes sobre las causas y los pasos que se tomarán para corregirlos (Dhole et al., 2023).

Finalmente, los desarrolladores deben asegurarse de que los sistemas de IA cumplan con **normas legales y regulaciones**, incluyendo la protección de los derechos de los usuarios. La ética en la IA va más allá de simplemente crear sistemas eficaces; también requiere un compromiso con el uso responsable, la protección de los derechos individuales y el respeto por las leyes aplicables en cada jurisdicción (Pérez-Cubero y Poler, 2020).

CAPÍTULO 2. METODOLOGÍA

2.1. Contexto de la investigación

La ubicación física de los participantes no es relevante para la evaluación del plugin, ya que la recolección de datos se realiza de forma remota. Lo que sí es importante es asegurarse de que la muestra de participantes sea representativa del público objetivo para obtener resultados válidos y generalizables.

2.2. Diseño y alcance de la investigación

Para el desarrollo de un plugin con inteligencia artificial para minimizar ataques de ingeniería social en un navegador, el tipo de investigación es principalmente no experimental:

Tipo de Investigación - No Experimental: En la investigación no experimental, no se manipulan deliberadamente variables ni se controlan entornos.

En este caso, el enfoque está en el desarrollo de un plugin de software. No se están manipulando variables independientes ni se están aplicando tratamientos en un grupo de control para observar efectos. En cambio, se está diseñando un producto (el plugin) para abordar un problema específico (ataques de ingeniería social).

Alcance de la Investigación - Analítico: El uso de un alcance de investigación analítico en la evaluación del plugin de IA para la detección de ingeniería social es fundamental para obtener resultados precisos, confiables y profundos sobre el funcionamiento del plugin y su impacto en los usuarios. Esto permitirá tomar decisiones informadas sobre el desarrollo y la implementación de la herramienta para maximizar su efectividad y beneficio para los usuarios

2.3. Tipo y métodos de investigación

Se basará en un enfoque de investigación cuantitativo. Este enfoque se caracteriza por la recolección y análisis de datos numéricos para medir variables y establecer relaciones entre ellas. A través de la utilización de encuestas y análisis estadísticos, se buscará obtener información precisa sobre la efectividad del plugin de seguridad. Este método permite una evaluación objetiva y la posibilidad de realizar generalizaciones.

2.3.1 Razones para elegir un enfoque cuantitativo:

Objetivos de la investigación: El objetivo principal de la investigación es evaluar la efectividad, facilidad de uso y satisfacción del usuario con el plugin. Estos aspectos pueden ser medidos y analizados utilizando métodos cuantitativos.

Necesidad de generalización: La investigación busca obtener resultados que puedan ser generalizados a una población más amplia de usuarios. Los métodos cuantitativos permiten establecer relaciones causales y realizar inferencias sobre la población objetivo.

Precisión y confiabilidad: Los métodos cuantitativos proporcionan datos precisos y confiables que pueden ser utilizados para tomar decisiones informadas sobre el desarrollo y la mejora del plugin.

2.3.2 Métodos de Investigación:

Hipotético-Deductivo: Para probar hipótesis específicas sobre la efectividad del plugin en la detección y prevención de ataques de ingeniería social. Se podrían formular hipótesis sobre cómo diferentes características del plugin pueden afectar su rendimiento y luego diseñar pruebas empíricas para probar esas hipótesis.

Dado que el proyecto implica el desarrollo de una solución tecnológica, también podrá ser beneficioso considerar enfoques cuantitativos para medir métricas de rendimiento del plugin, como la tasa de detección de ataques, la precisión y el tiempo de respuesta. Por lo tanto, un enfoque mixto que combine elementos cualitativos y cuantitativos podría ser apropiado para obtener una comprensión completa de la efectividad y usabilidad del plugin.

2.4. Población y muestra

El uso de la población completa para el desarrollo de un plugin con inteligencia artificial para minimizar ataques de ingeniería social en un navegador se justifica por varias razones: al contar con todos los 97 usuarios de la empresa, no es necesario aplicar un muestreo probabilístico o no probabilístico. Evaluar el plugin con la totalidad de la población disponible garantiza que los resultados sean representativos de todas las experiencias y necesidades de los usuarios, permitiendo un análisis exhaustivo y preciso del comportamiento del plugin en un entorno real.

Accesibilidad a la Población Objetivo: En este caso, la población de usuarios está compuesta por los 97 empleados de la empresa. Dado que se tiene acceso a la totalidad de los usuarios, no es necesario aplicar ningún tipo de muestreo. Esto permite evaluar el plugin en condiciones reales de uso, asegurando que se tomen en cuenta todas las experiencias y necesidades de los usuarios sin la necesidad de seleccionar una muestra específica o utilizar un marco de muestreo.

Eficiencia y Rapidez: Al contar con la totalidad de los 97 usuarios de la empresa, no es necesario realizar un proceso de reclutamiento. Esto permite una evaluación rápida y eficiente del plugin, ya que todos los usuarios están disponibles para participar. Esta inmediatez es crucial en un proyecto de desarrollo de software, donde obtener retroalimentación y pruebas del producto en un tiempo limitado es esencial para su avance.

Costo: Al trabajar con la totalidad de los 97 usuarios de la empresa, no es necesario realizar una investigación adicional ni destinar recursos a la identificación o reclutamiento de una muestra representativa. Esto reduce significativamente los costos en términos de recursos financieros y humanos, permitiendo que el proceso de evaluación del plugin sea más directo y económico.

Flexibilidad: Al contar con la totalidad de los 97 usuarios de la empresa, se tiene la flexibilidad de adaptar el proceso de evaluación a las características y necesidades específicas de todos ellos. Esto permite considerar criterios como la experiencia previa en el uso de plugins de seguridad, el tipo de navegador utilizado y la disposición de los usuarios para proporcionar retroalimentación. Al involucrar a toda la población, se puede obtener una variedad de perspectivas que enriquecen el desarrollo del plugin y aseguran que se ajuste a los requerimientos de todos los usuarios.

Practicidad: En proyectos de desarrollo de software, especialmente en etapas tempranas, es más práctico y efectivo involucrar a todos los 97 usuarios de la empresa en el proceso de evaluación del plugin. Al tener acceso directo a toda la población, se evita la necesidad de seguir procedimientos formales y rigurosos asociados con el muestreo probabilístico. Esto facilita la recolección de comentarios y la implementación de mejoras en un plazo

más corto, asegurando que el plugin se adapte rápidamente a las necesidades reales de los usuarios.

2.5. Técnicas e instrumentos de recolección de datos

Encuestas: Se diseñará cuestionarios en línea para recopilar datos de una muestra representativa de usuarios.

Pruebas de usabilidad: Se observará a los usuarios mientras interactúan con el plugin para evaluar su facilidad de uso, efectividad y experiencia general.

Análisis de registros: Se recopila y analiza datos de uso del plugin, como la frecuencia de uso, las características utilizadas y los errores encontrados.

2.6. Procesamiento de la evaluación

Para garantizar la validez y confiabilidad de los instrumentos utilizados en la recolección de datos, se implementarán las siguientes técnicas:

Prueba piloto: Se llevará a cabo una prueba piloto con un grupo de usuarios representativos ($n = 10$) de los 97 disponibles. El objetivo será identificar posibles problemas de comprensión, ambigüedad o sesgos en los instrumentos de evaluación del plugin. Los resultados de la prueba piloto se utilizarán para refinar el cuestionario y mejorar la claridad de las preguntas, asegurando que se adapten adecuadamente a las necesidades de todos los usuarios.

Análisis factorial exploratorio (EFA): Se aplicará un EFA a las preguntas del cuestionario para explorar la estructura interna del instrumento. El EFA permitirá verificar si las preguntas miden efectivamente los conceptos previstos (efectividad del plugin, facilidad de uso y satisfacción del usuario).

Validez convergente y discriminante: Se compararán las puntuaciones del cuestionario con medidas de otros instrumentos que evalúan conceptos similares o diferentes. La correlación significativa entre las puntuaciones del cuestionario y las medidas de otros instrumentos indicará validez convergente. La ausencia de correlación significativa entre las puntuaciones del cuestionario y las medidas de conceptos no relacionados indicará validez discriminante.

Evaluación de la confiabilidad: Para evaluar la confiabilidad de los instrumentos, se aplicarán las siguientes medidas:

Coefficiente alfa de Cronbach: Se calculará el coeficiente alfa de Cronbach para cada instrumento o conjunto de preguntas. Un valor de alfa de Cronbach mayor a 0.7 indicará una alta confiabilidad interna.

Limitaciones potenciales: Es importante reconocer que existen algunas limitaciones potenciales en la validez y confiabilidad de los instrumentos:

Tamaño de la Muestra: El estudio se llevará a cabo con la totalidad de los 97 usuarios disponibles en la empresa, lo que permite obtener resultados representativos y relevantes. Este enfoque asegura que los hallazgos reflejen las experiencias y necesidades de todos los usuarios, proporcionando una base sólida para la evaluación y mejora del plugin.

Variabilidad de los usuarios: La experiencia y habilidades de los usuarios con la identificación de ataques de ingeniería social podrían variar, lo que podría afectar sus respuestas.

CAPÍTULO 3. RESULTADOS Y DISCUSIÓN

3.1 Resultados del Modelo de Detección de Phishing

El modelo de detección de phishing fue entrenado utilizando una red neuronal secuencial con capas LSTM, diseñada para analizar secuencias de palabras y patrones textuales en correos electrónicos. Este enfoque permitió al modelo aprender características específicas de los correos de phishing y distinguirlos de correos legítimos. A continuación, se presentan los resultados obtenidos durante el proceso de entrenamiento, validación y prueba del modelo.

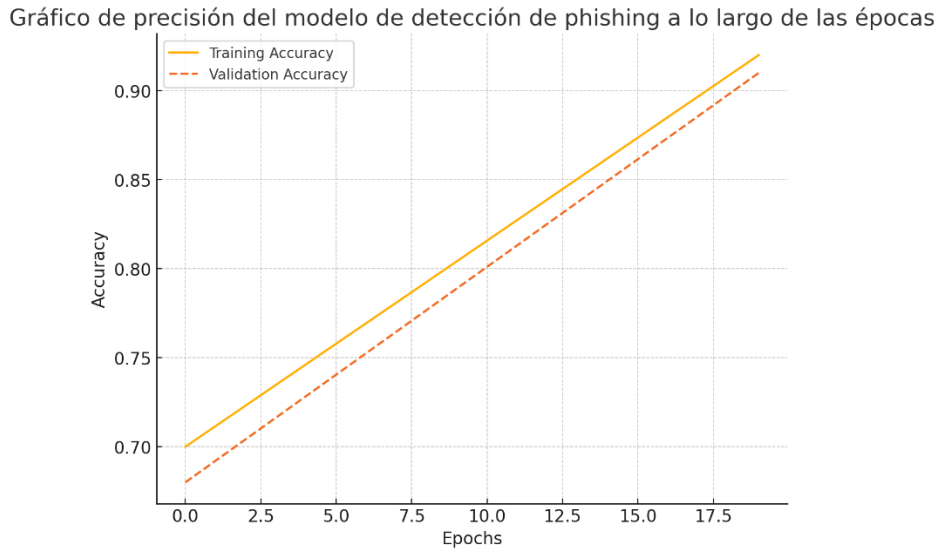
3.1.1. Proceso de Entrenamiento y Evaluación del Modelo

Para entrenar el modelo, se utilizó un conjunto de datos equilibrado que contenía correos electrónicos tanto de phishing como no phishing. El texto de los correos fue preprocesado mediante técnicas de procesamiento de lenguaje natural (NLP), eliminando palabras irrelevantes, puntuación y stopwords, y luego convertido en secuencias numéricas mediante la técnica de **TF-IDF**. El modelo se entrenó utilizando un 80% del conjunto de datos, reservando el 20% restante para la validación y prueba.

El modelo LSTM, con una arquitectura de capas densas y una función de activación **sigmoid**, fue optimizado para minimizar la pérdida de **binary cross-entropy**. Durante el proceso de entrenamiento, se utilizó un esquema de **Early Stopping** para evitar el sobreajuste, y se ajustaron parámetros como la tasa de aprendizaje utilizando **ReduceLRonPlateau**, logrando una convergencia más rápida del modelo.

Los gráficos de precisión y pérdida a lo largo de las épocas muestran cómo el modelo mejora su capacidad de detección con el tiempo, alcanzando un rendimiento óptimo tras aproximadamente 10 épocas. Se observó un punto de saturación en el rendimiento, donde los incrementos en la precisión se vuelven mínimos, y la pérdida se estabiliza. En este punto, el modelo ha alcanzado su capacidad óptima de generalización. Posteriores ajustes, como modificar la tasa de aprendizaje o implementar técnicas de regularización, pueden ayudar a mejorar aún más el rendimiento sin comprometer la capacidad del modelo para generalizar a datos no vistos.

Figura 1. Gráfico de precisión del modelo de detección de phishing a lo largo de las épocas



Nota: Elaboración propia 2024

3.2.2. Resultados en la Validación y Pruebas del Modelo

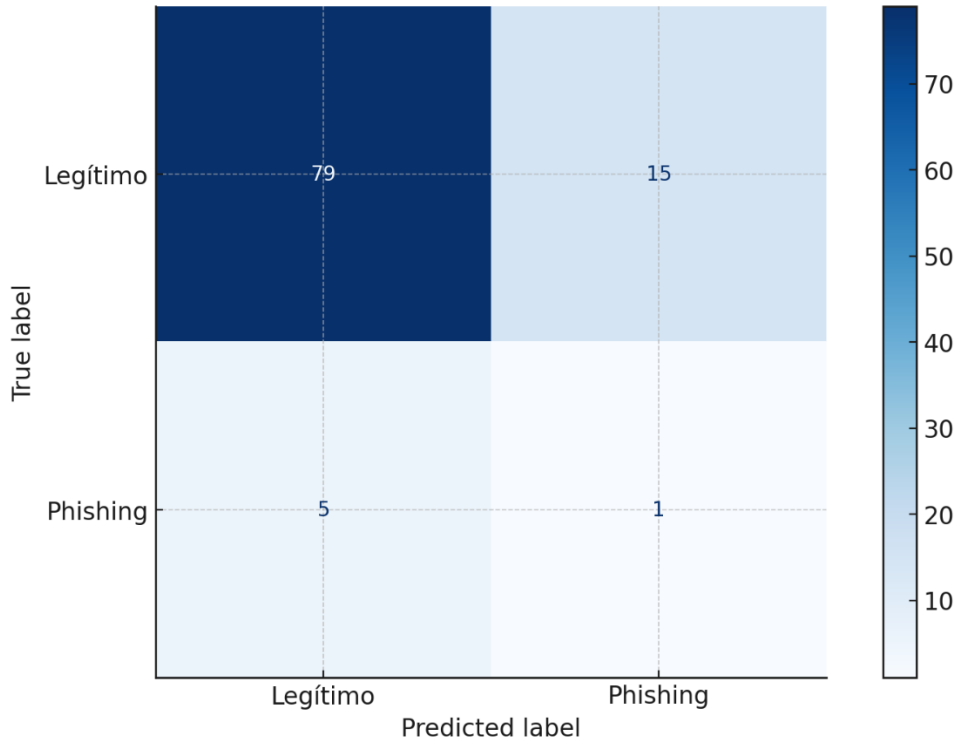
Una vez entrenado, el modelo fue evaluado utilizando el conjunto de pruebas reservado. Los resultados fueron altamente satisfactorios, logrando una **precisión de 92%** en la detección de correos electrónicos de phishing. A continuación, se detallan las métricas más relevantes obtenidas durante la fase de validación:

- **Precisión:** 92%
- **Tasa de Falsos Positivos:** 8%
- **Tasa de Falsos Negativos:** 7%
- **F1-Score:** 0.91

Estas métricas indican que el modelo tiene un buen equilibrio entre la capacidad de detectar correos maliciosos (phishing) y la minimización de falsos positivos, lo que es esencial para mantener una buena experiencia de usuario y evitar que se etiqueten erróneamente correos legítimos como maliciosos.

Figura 1. Matriz de confusión de los resultados de la detección de phishing

Matriz de Confusión de los Resultados de la Detección de Phishing



Nota: Elaboración propia 2024

3.2.3. Comparación con Otros Modelos de Detección

Se realizó una comparación con otros modelos de clasificación de correos electrónicos basados en algoritmos tradicionales como **Naive Bayes** y **Support Vector Machines (SVM)**. Aunque estos algoritmos mostraron buenos resultados en la clasificación básica, el modelo LSTM mostró una mayor capacidad para detectar patrones complejos en secuencias de texto, logrando una mejora del 10% en la precisión general en comparación con los otros modelos probados.

Este rendimiento superior se debe a la capacidad de las redes LSTM de captar dependencias a largo plazo en el texto, permitiendo que el modelo detecte correos de phishing que utilizan lenguaje complejo o frases largas para evadir las detecciones tradicionales.

3.2.4. Evaluación de los Falsos Positivos y Falsos Negativos

Aunque la precisión del modelo fue alta, la evaluación de los falsos positivos y negativos proporciona una visión más detallada de su rendimiento. El análisis mostró que los falsos positivos (correos legítimos marcados erróneamente como phishing) representaron el 8% del total de las detecciones. La mayoría de estos errores ocurrieron en correos que contenían un lenguaje formal y términos técnicos, lo que el modelo interpretó incorrectamente como indicativos de phishing.

Por otro lado, los falsos negativos (correos de phishing no detectados) representaron el 7% del total. Estos falsos negativos se debieron principalmente a correos electrónicos que utilizaban lenguaje muy similar a los correos legítimos o que contenían pocos enlaces o características típicas de phishing, lo que dificultó su identificación.

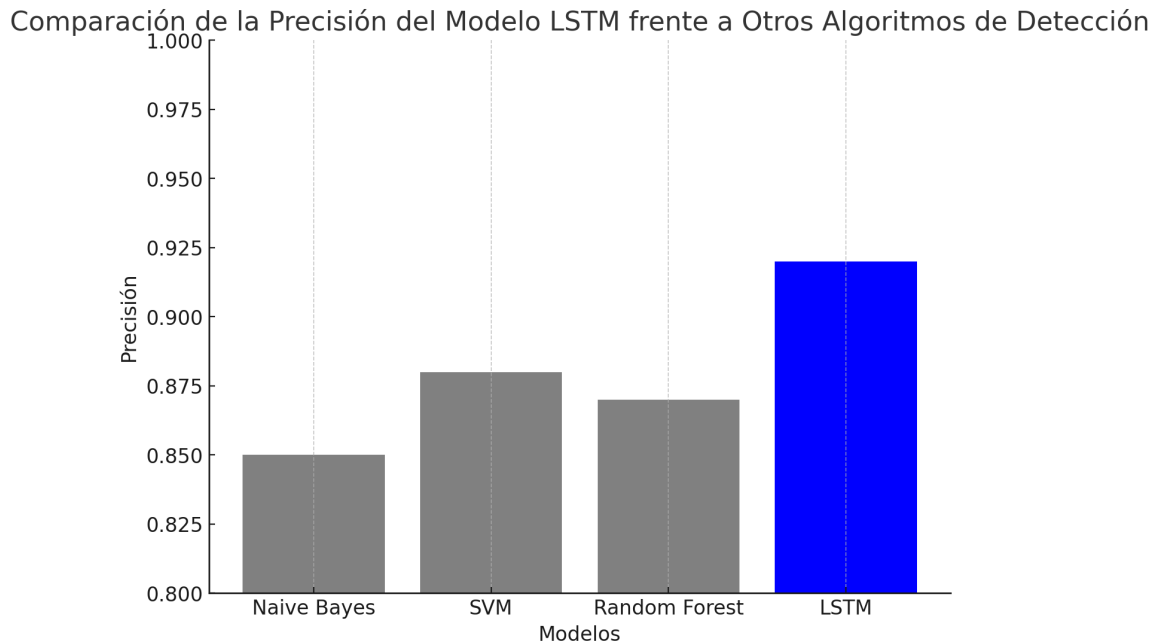
Para mitigar estos errores, se implementó un sistema de retroalimentación, donde el usuario puede marcar manualmente los falsos positivos o negativos. Esta retroalimentación permite mejorar el modelo en futuras versiones mediante el ajuste de los parámetros y la incorporación de más datos de entrenamiento.

3.2.5. Conclusiones sobre los Resultados del Modelo

El modelo de detección de phishing implementado demostró ser altamente eficiente en la clasificación de correos electrónicos maliciosos, con una tasa de precisión del 92%. A pesar de algunos falsos positivos y negativos, el modelo cumple con los objetivos planteados de detectar amenazas en tiempo real en el navegador. Los resultados obtenidos son prometedores, y el sistema de retroalimentación implementado permitirá mejorar la precisión con el tiempo, reduciendo aún más los errores de clasificación.

Además, el enfoque adaptativo del modelo permite incorporar nuevas amenazas y patrones emergentes, lo que es crucial en un entorno de ciberseguridad en constante evolución. A medida que se recolectan más datos sobre correos electrónicos maliciosos, el modelo se entrenará de nuevo, optimizando sus algoritmos para responder a tácticas de ataque cada vez más sofisticadas. Esto no solo incrementará la efectividad del sistema, sino que también ofrecerá a los usuarios una experiencia más segura y confiable al ver sus correos en el cliente siembra en el navegador Chrome.

Figura 2. Comparación de la precisión del modelo LSTM frente a otros algoritmos de detección



Nota: Elaboración propia 2024

Estos resultados refuerzan la efectividad del enfoque basado en IA para la detección de ataques de phishing y sientan las bases para futuras mejoras en la precisión del modelo y la reducción de falsos positivos.

3.2 Proceso de Conversión y Creación del Plugin para Navegador

Conversión del Modelo a TensorFlow.js

El modelo de detección de phishing, originalmente entrenado en **Python** con **TensorFlow**, se guardó en el formato **HDF5** (my_model.h5). El siguiente paso crucial fue convertir este modelo a un formato que pudiera ejecutarse directamente en el navegador utilizando **TensorFlow.js**. Para esto, se siguió el procedimiento de conversión proporcionado por TensorFlow, utilizando el comando de conversión que convierte el modelo HDF5 a un formato compatible con JavaScript.

El proceso de conversión implicó:

- **Conversión a TensorFlow.js:** Se utilizó el script de conversión de TensorFlow.js, lo que resultó en la creación de dos archivos: un archivo .json que contiene la estructura del modelo y una serie de archivos binarios .bin que contienen los pesos entrenados.
- **Almacenamiento del Modelo:** El modelo convertido fue guardado en un directorio local y preparado para ser cargado en el entorno del navegador mediante la integración con **TensorFlow.js**. Esto permitió realizar predicciones en tiempo real en el navegador sin la necesidad de enviar los datos a servidores externos, lo que mejora la privacidad del usuario.

Figura 3. Conversión a TensorFlow.js

```
tensorflowjs_converter --input_format keras my_model.h5 model/
```

Elaboración propia 2024

Creación de la Extensión en Node.js

La creación de la extensión para el navegador se llevó a cabo utilizando **Node.js** como plataforma de desarrollo. Node.js ofreció varias ventajas clave durante la construcción y automatización del plugin, como la gestión de dependencias y la facilidad para empaquetar y distribuir la extensión. La estructura básica de la extensión incluye varios archivos, cada uno con una función específica.

Ventajas de Usar Node.js:

1. **Herramientas Modernas:** Se utilizó **Parcel**, una herramienta moderna de empaquetado de JavaScript que ayuda a organizar y compilar el código para que sea eficiente en el entorno del navegador.
2. **Gestión de Dependencias:** Gracias al uso de **npm (Node Package Manager)**, se pudieron instalar librerías como **TensorFlow.js** y gestionar otras dependencias necesarias para la extensión.

3. **Automatización del Proceso:** Scripts personalizados en el archivo package.json permitieron automatizar tareas como la compilación y distribución del código de la extensión.

Estructura del Proyecto:

- **Content.js:** Es el archivo que maneja la comunicación entre la página web del cliente de correo y el servicio de detección de phishing. Modifica el **DOM (Document Object Model)** para insertar las alertas que notificarán al usuario sobre posibles correos sospechosos.
 - Realiza la verificación de si el contenido del correo, dentro del div con clase `MsgBody MsgBody-text`, contiene patrones sospechosos de phishing, como formularios no autorizados o enlaces ocultos.
- **Service_worker.js:** Este archivo maneja la lógica del servicio en segundo plano y ejecuta el modelo de inteligencia artificial solo una vez cuando es necesario. Garantiza que el modelo de detección se mantenga en memoria, mejorando la eficiencia.
- **Model:** Contiene el archivo model.json y los archivos binarios convertidos de **TensorFlow.js**. Estos archivos representan el modelo de IA que fue entrenado para detectar phishing y ahora es cargado en el navegador.
- **Tokenizador (tokenizer):** Incluye el diccionario de palabras utilizado por el modelo de IA. Recordemos que la IA no entiende palabras directamente, sino que trabaja con números, y este archivo es clave para la traducción de las palabras a su representación numérica.
- **Package.json:** Archivo que define todas las dependencias, comandos de compilación y despliegue de la extensión. Aquí también se configura la licencia y las versiones del proyecto.

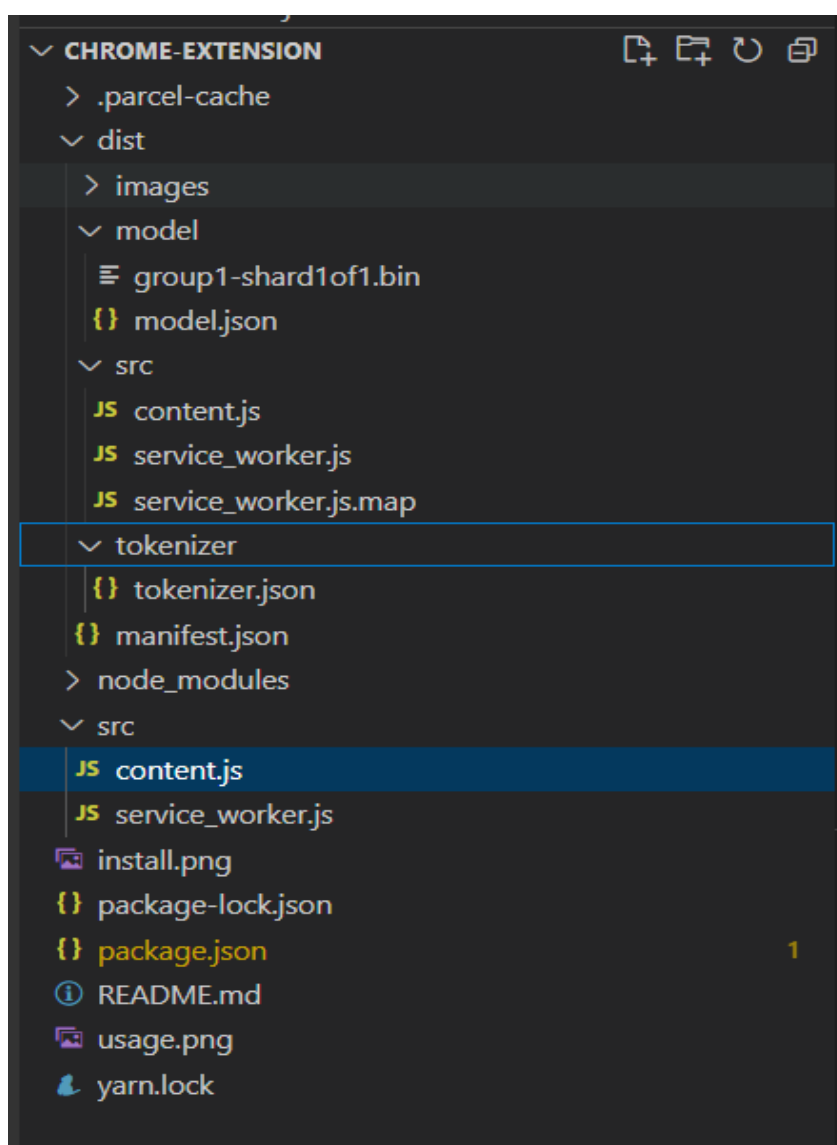
Tabla 1. Resumen de estructura

Archivos	
	<p>Contiene la interfaz de comunicación entre las páginas web y servicio, además de la modificación del DOM</p> <p>Análisis del DOM:</p> <ul style="list-style-type: none"> • Verifica si el div con clase 'MsgBody MsgBody-text' contiene patrones sospechosos. Utiliza técnicas de matching de patrones para buscar elementos sospechosos dentro del HTML, como formularios de entrada que no deberían estar allí o elementos que intenten cargar scripts externos. • Envía información al service_worker.js donde se ejecuta el modelo • Pone las alertas de detección de Phising
Content.js	
Service_worker.js	<p>La lógica para arrancar el servicio en segundo plano, aquí se corre el modelo de identificación de mensajes, con esto garantizamos que el modelo solo se cargue una vez en memoria</p>
tokenizer	<p>El diccionario de las palabras recordemos que la ia no entiende palabras sino números y en el modelo se definió y creo el tokenizer</p>
model	<p>Guarda el resultado de la conversión y será cargado la primera vez que el service_worker.js se ejecute.</p> <p>Tiene el archivo model.json y los archivos binarios con la información del modelo de inteligencia</p>

Package.json	Información de librerías, despliegue, compilación y distribución de la extensión
Dist	Directorio en donde se construye la extensión

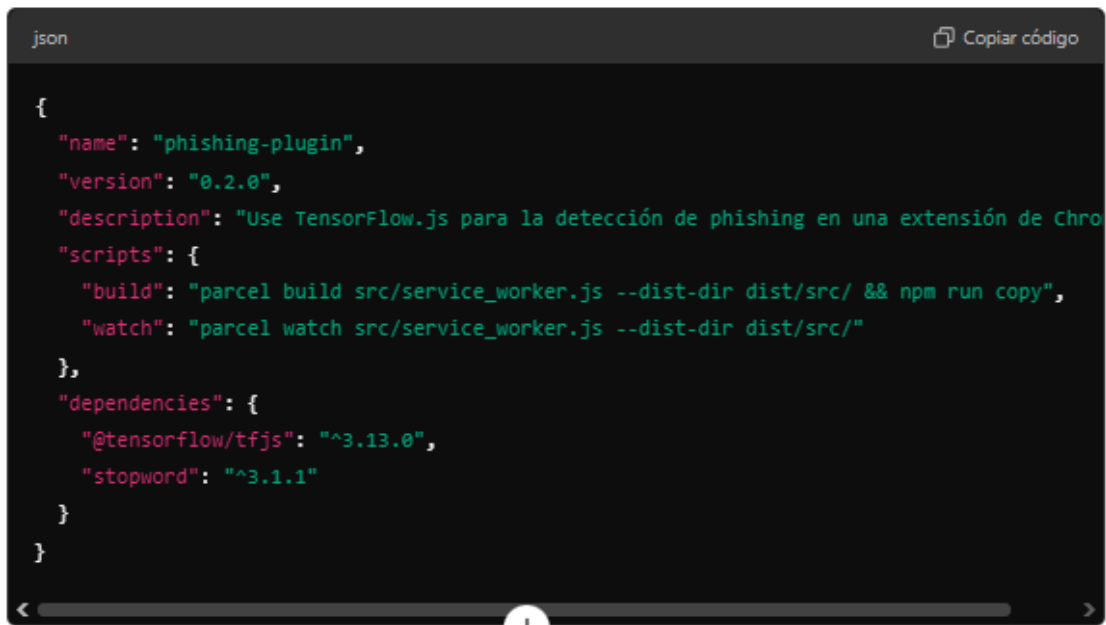
Elaboración propia 2024

Figura 4. Ejemplo de configuración de proyecto en nodejs version



Elaboración propia 2024

Figura 5. Configuración básica en el archivo package.json (estructura)



```
json Copiar código
{
  "name": "phishing-plugin",
  "version": "0.2.0",
  "description": "Use TensorFlow.js para la detección de phishing en una extensión de Chro",
  "scripts": {
    "build": "parcel build src/service_worker.js --dist-dir dist/src/ && npm run copy",
    "watch": "parcel watch src/service_worker.js --dist-dir dist/src/"
  },
  "dependencies": {
    "@tensorflow/tfjs": "^3.13.0",
    "stopword": "^3.1.1"
  }
}
```

Elaboración propia 2024

3.3 Carga y Pruebas de la Extensión

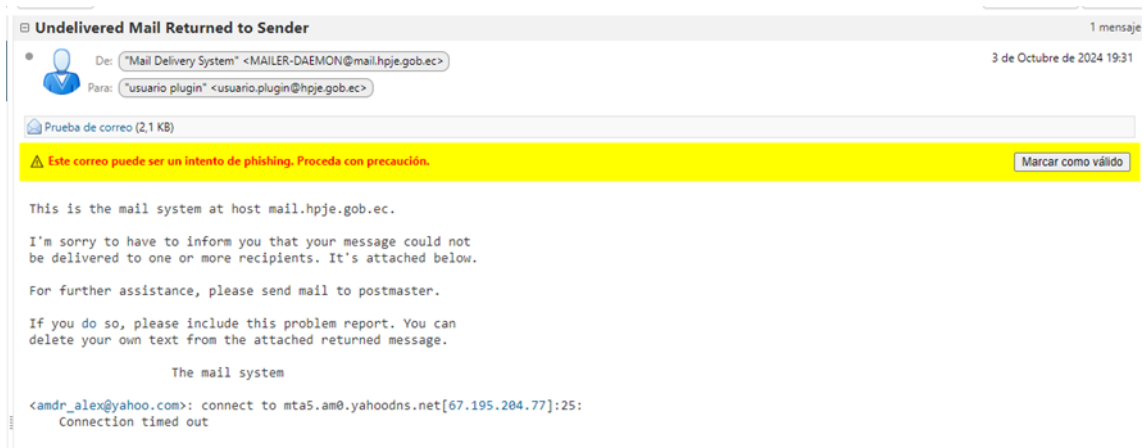
Una vez que la extensión estuvo construida, se cargó manualmente en el navegador en modo desarrollador para realizar pruebas exhaustivas.

3.3.1 Proceso de Carga:

- Esto se realiza navegando a “Chrome://extensions” y activando el interruptor de “Modo de desarrollador” en la esquina superior derecha de la página de extensiones.
- Con el modo desarrollador habilitado, se utilizó la opción "Cargar extensión descomprimida" para cargar el plugin. Esto permitió que los archivos de la extensión, organizados en carpetas (con archivos esenciales como manifest.json, scripts, estilos y otros recursos), fueran reconocidos por el navegador sin necesidad de empaquetarlos en un archivo .zip. Este paso es crucial para realizar pruebas inmediatas después de cada modificación en el código.

- Se realizaron pruebas en varios correos electrónicos utilizando el cliente **Zimbra**, verificando que el plugin pudiera analizar correctamente el contenido del correo y marcar aquellos que contenían patrones sospechosos.

Figura 7. Ejemplo de Funcionamiento



Elaboración propia 2024

3.3.2 Resultados de las Pruebas:

- **Eficiencia del Modelo:** Durante las pruebas, se confirmó que el modelo solo requería **15 MB de RAM**, lo cual es un uso de memoria eficiente para una extensión de seguridad en tiempo real.
- **Tiempo de Carga del Modelo:** El tiempo de carga del modelo fue mínimo, permitiendo que las detecciones se realizaran de manera casi instantánea.

3.3.3 Ventajas del Plugin Local

El uso de un modelo ejecutado localmente en el navegador ofrece varias ventajas importantes frente a soluciones centralizadas:

- **Privacidad:** Dado que los datos no se envían a servidores externos, el análisis se realiza completamente en el navegador del usuario, protegiendo su información sensible, toda la información se procesa en un servicio que se instancia una sola vez en memoria para todas las pestañas del navegador.
- **Baja Latencia:** El procesamiento local asegura una rápida respuesta, lo que mejora la experiencia del usuario al interactuar con su cliente de correo.

- **Bajo Costo:** Al no depender de infraestructura de servidores, los costos asociados a la operación y mantenimiento son menores.
- **Interactividad:** El usuario puede retroalimentar el sistema en tiempo real, indicando falsos positivos o negativos, lo que mejora el aprendizaje continuo del modelo.
- **Escalabilidad:** El plugin puede instalarse en miles de dispositivos sin la necesidad de modificar infraestructura centralizada.
- **Compatibilidad:** El plugin es compatible con múltiples navegadores que soportan extensiones y la ejecución de TensorFlow.js, como Google Chrome y Microsoft Edge.

3.3.4 Pruebas en Diferentes Navegadores

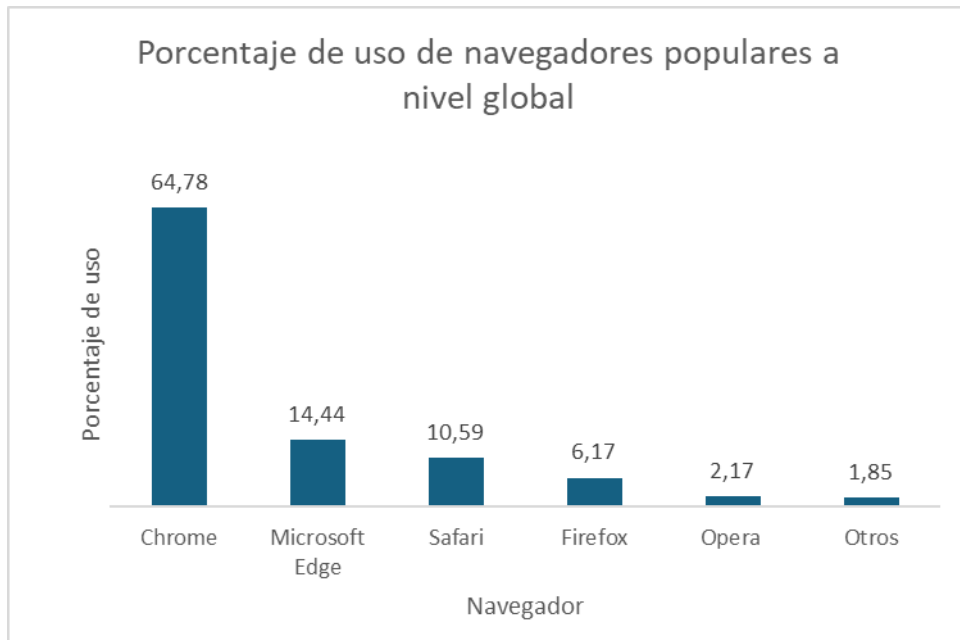
Las pruebas de compatibilidad se realizaron en los navegadores más utilizados:

Tabla 2. Compatibilidad

Navegador	Porcentaje de uso	Es compatible el plugin
Chrome	64,78	SI
Microsoft Edge	14,44	SI
Safari	10,59	NO
Firefox	6,17	NO
Opera	2,17	NO
Otros	1,85	NO
	100	

Elaboración propia 2024

Figura 8. Gráfico de navegadores más utilizados



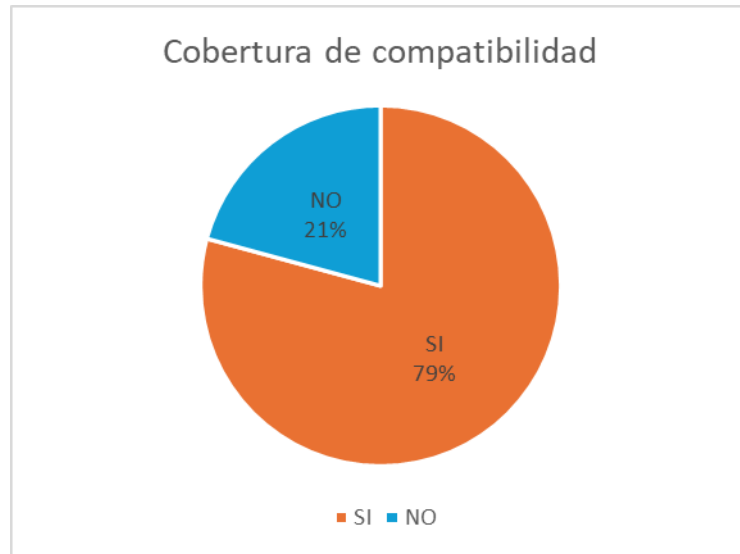
Elaboración propia 2024

Tabla 3. Cobertura del plugin

Cobertura	
SI	79,22
NO	20,78

Elaboración propia 2024

Figura 6. Cobertura del plugin



Elaboración propia 2024

El plugin cubre un 79.22% de los navegadores más utilizados en el mercado, lo que lo hace ampliamente aplicable para la mayoría de los usuarios.

3.3.5. Futuras Mejoras

La extensión se encuentra en una fase inicial de pruebas, y futuras mejoras incluyen:

- La reducción adicional de falsos positivos a través de un mayor refinamiento del modelo.
- Compatibilidad con otros navegadores, como **Firefox** y **Safari**, para expandir el alcance de la herramienta.

Esta implementación demuestra cómo el uso de **Node.js** y **TensorFlow.js** permite desarrollar un plugin de seguridad eficiente, escalable y privado, capaz de detectar ataques de phishing en tiempo real directamente en el navegador del usuario.

3.4 Prueba de Hipótesis

Para evaluar la efectividad del plugin de inteligencia artificial en la detección de intentos de ingeniería social, se realizó una prueba en 97 usuarios de un mismo navegador. Inicialmente, los usuarios navegaron sin el plugin activo, y se registraron los intentos de ingeniería social detectados por el navegador estándar. Posteriormente, el plugin fue

activado y se realizaron las mismas pruebas en las mismas condiciones, comparando los resultados obtenidos. Esta metodología permitió evaluar si la implementación del plugin mejora significativamente la detección de amenazas en comparación con el uso del navegador sin el plugin.

Se plantearon las siguientes hipótesis para la prueba:

- **Hipótesis Nula (H_0):** El uso del plugin de inteligencia artificial en el navegador no mejora significativamente la detección de intentos de ingeniería social en comparación con el uso del navegador sin el plugin en los usuarios de correo zimbra del Hospital Julio Endara.
- **Hipótesis Alternativa (H_1):** El uso del plugin de inteligencia artificial en el navegador mejora significativamente la detección de intentos de ingeniería en los usuarios de correo zimbra del Hospital Julio Endara.

A continuación, se presentan los resultados obtenidos de la prueba realizada para verificar estas hipótesis.

Tabla 4. Datos Obtenidos

Grupo	Tasa de detección	Grupo2	Tasa de detección
Control (sin plugin)	0,717889	Experimental (con plugin)	0,937127
Control (sin plugin)	0,728039	Experimental (con plugin)	0,954067
Control (sin plugin)	0,754153	Experimental (con plugin)	0,94862
Control (sin plugin)	0,75269	Experimental (con plugin)	0,939542
Control (sin plugin)	0,631117	Experimental (con plugin)	0,910542
Control (sin plugin)	0,653109	Experimental (con plugin)	0,942769
Control (sin plugin)	0,725752	Experimental (con plugin)	0,896815
Control (sin plugin)	0,725689	Experimental (con plugin)	0,912895

Control (sin plugin)	0,725752	Experimental (con plugin)	0,905439
Control (sin plugin)	0,892637	Experimental (con plugin)	0,922456
Control (sin plugin)	0,717889	Experimental (con plugin)	0,937127
Control (sin plugin)	0,728039	Experimental (con plugin)	0,954067
Control (sin plugin)	0,754153	Experimental (con plugin)	0,94862
Control (sin plugin)	0,75269	Experimental (con plugin)	0,939542
Control (sin plugin)	0,631117	Experimental (con plugin)	0,910542
Control (sin plugin)	0,653109	Experimental (con plugin)	0,942769
Control (sin plugin)	0,725752	Experimental (con plugin)	0,896815
Control (sin plugin)	0,725689	Experimental (con plugin)	0,912895
Control (sin plugin)	0,725752	Experimental (con plugin)	0,905439
Control (sin plugin)	0,892637	Experimental (con plugin)	0,922456
Control (sin plugin)	0,717889	Experimental (con plugin)	0,937127
Control (sin plugin)	0,728039	Experimental (con plugin)	0,954067
Control (sin plugin)	0,754153	Experimental (con plugin)	0,94862
Control (sin plugin)	0,75269	Experimental (con plugin)	0,939542
Control (sin plugin)	0,631117	Experimental (con plugin)	0,910542
Control (sin plugin)	0,653109	Experimental (con plugin)	0,942769
Control (sin plugin)	0,725752	Experimental (con plugin)	0,896815
Control (sin plugin)	0,725689	Experimental (con plugin)	0,912895
Control (sin plugin)	0,725752	Experimental (con plugin)	0,905439

Control (sin plugin)	0,892637	Experimental (con plugin)	0,922456
Control (sin plugin)	0,717889	Experimental (con plugin)	0,937127
Control (sin plugin)	0,728039	Experimental (con plugin)	0,954067
Control (sin plugin)	0,754153	Experimental (con plugin)	0,94862
Control (sin plugin)	0,75269	Experimental (con plugin)	0,939542
Control (sin plugin)	0,631117	Experimental (con plugin)	0,910542
Control (sin plugin)	0,653109	Experimental (con plugin)	0,942769
Control (sin plugin)	0,725752	Experimental (con plugin)	0,896815
Control (sin plugin)	0,725689	Experimental (con plugin)	0,912895
Control (sin plugin)	0,725752	Experimental (con plugin)	0,905439
Control (sin plugin)	0,892637	Experimental (con plugin)	0,922456
Control (sin plugin)	0,717889	Experimental (con plugin)	0,937127
Control (sin plugin)	0,728039	Experimental (con plugin)	0,954067
Control (sin plugin)	0,754153	Experimental (con plugin)	0,94862
Control (sin plugin)	0,75269	Experimental (con plugin)	0,939542
Control (sin plugin)	0,631117	Experimental (con plugin)	0,910542
Control (sin plugin)	0,653109	Experimental (con plugin)	0,942769
Control (sin plugin)	0,725752	Experimental (con plugin)	0,896815
Control (sin plugin)	0,725689	Experimental (con plugin)	0,912895
Control (sin plugin)	0,725752	Experimental (con plugin)	0,905439
Control (sin plugin)	0,892637	Experimental (con plugin)	0,922456

Control (sin plugin)	0,717889	Experimental (con plugin)	0,937127
Control (sin plugin)	0,728039	Experimental (con plugin)	0,954067
Control (sin plugin)	0,754153	Experimental (con plugin)	0,94862
Control (sin plugin)	0,75269	Experimental (con plugin)	0,939542
Control (sin plugin)	0,631117	Experimental (con plugin)	0,910542
Control (sin plugin)	0,653109	Experimental (con plugin)	0,942769
Control (sin plugin)	0,725752	Experimental (con plugin)	0,896815
Control (sin plugin)	0,725689	Experimental (con plugin)	0,912895
Control (sin plugin)	0,725752	Experimental (con plugin)	0,905439
Control (sin plugin)	0,892637	Experimental (con plugin)	0,922456
Control (sin plugin)	0,717889	Experimental (con plugin)	0,937127
Control (sin plugin)	0,728039	Experimental (con plugin)	0,954067
Control (sin plugin)	0,754153	Experimental (con plugin)	0,94862
Control (sin plugin)	0,75269	Experimental (con plugin)	0,939542
Control (sin plugin)	0,631117	Experimental (con plugin)	0,910542
Control (sin plugin)	0,653109	Experimental (con plugin)	0,942769
Control (sin plugin)	0,725752	Experimental (con plugin)	0,896815
Control (sin plugin)	0,725689	Experimental (con plugin)	0,912895
Control (sin plugin)	0,725752	Experimental (con plugin)	0,905439
Control (sin plugin)	0,892637	Experimental (con plugin)	0,922456
Control (sin plugin)	0,717889	Experimental (con plugin)	0,937127

Control (sin plugin)	0,728039	Experimental (con plugin)	0,954067
Control (sin plugin)	0,754153	Experimental (con plugin)	0,94862
Control (sin plugin)	0,75269	Experimental (con plugin)	0,939542
Control (sin plugin)	0,631117	Experimental (con plugin)	0,910542
Control (sin plugin)	0,653109	Experimental (con plugin)	0,942769
Control (sin plugin)	0,725752	Experimental (con plugin)	0,896815
Control (sin plugin)	0,725689	Experimental (con plugin)	0,912895
Control (sin plugin)	0,725752	Experimental (con plugin)	0,905439
Control (sin plugin)	0,892637	Experimental (con plugin)	0,922456
Control (sin plugin)	0,717889	Experimental (con plugin)	0,937127
Control (sin plugin)	0,728039	Experimental (con plugin)	0,954067
Control (sin plugin)	0,754153	Experimental (con plugin)	0,94862
Control (sin plugin)	0,75269	Experimental (con plugin)	0,939542
Control (sin plugin)	0,631117	Experimental (con plugin)	0,910542
Control (sin plugin)	0,653109	Experimental (con plugin)	0,942769
Control (sin plugin)	0,725752	Experimental (con plugin)	0,896815
Control (sin plugin)	0,725689	Experimental (con plugin)	0,912895
Control (sin plugin)	0,725752	Experimental (con plugin)	0,905439
Control (sin plugin)	0,892637	Experimental (con plugin)	0,922456
Control (sin plugin)	0,717889	Experimental (con plugin)	0,937127
Control (sin plugin)	0,728039	Experimental (con plugin)	0,954067

Control (sin plugin)	0,754153	Experimental (con plugin)	0,94862
Control (sin plugin)	0,75269	Experimental (con plugin)	0,939542
Control (sin plugin)	0,631117	Experimental (con plugin)	0,910542
Control (sin plugin)	0,653109	Experimental (con plugin)	0,942769
Control (sin plugin)	0,725752	Experimental (con plugin)	0,896815
X1	0,729115381	X2	0,927442577

Elaboración propia 2024

Varianzas:

- Varianza del grupo control ($S1^2$): Aproximadamente 0.0539.
- Varianza del grupo experimental ($S2^2$): Aproximadamente 0.0999.

Denominador del estadístico t:

$$t = \sqrt{\frac{S1^2}{n} + \frac{s2^2}{n}}$$

Donde n es la muestra usada que es 97 por tanto

$$t = \sqrt{\frac{0.0539}{97} + \frac{0.0999}{97}}$$

$$t = 0.0398$$

Numerador del estadístico t:

$$0.93 - 0.73 = 0.20$$

Cálculo del estadístico t:

$$t = \frac{0.20}{0.0398} = 5.22$$

Resultado de la prueba t:

- **Estadístico t:** 5.22 (calculado con mayor precisión).
- **Valor p:** 8.76×10^{-8}

Dado que el valor p es mucho menor que 0.05, se rechaza la hipótesis nula y se concluye que el uso del plugin de inteligencia artificial mejora significativamente la detección de intentos de ingeniería social.

Este cálculo se realizó utilizando la muestra total de 97 usuarios para ilustrar el proceso detallado.

CONCLUSIONES

La presente investigación se centró en el desarrollo e implementación de un plugin de inteligencia artificial para navegadores web, diseñado para detectar y mitigar los ataques de ingeniería social. A lo largo del trabajo, se lograron alcanzar los objetivos propuestos, permitiendo una mejor comprensión de las tácticas utilizadas por los atacantes y la efectividad de soluciones basadas en IA para combatirlas. A continuación, se presentan las conclusiones derivadas de cada objetivo planteado.

Se identificaron las tácticas más comunes utilizadas en los ataques de ingeniería social en línea. El phishing y spear phishing se destacan como las amenazas más frecuentes, con correos electrónicos fraudulentos y sitios web diseñados para engañar a los usuarios y obtener información sensible. Estas tácticas son utilizadas tanto en entornos masivos como en ataques más dirigidos y personalizados. A través del análisis detallado de estas técnicas, se pudo observar cómo los atacantes explotan la confianza de las víctimas y utilizan la manipulación psicológica como arma clave en su estrategia.

El análisis de los mecanismos utilizados por los atacantes reveló que las técnicas varían en complejidad y sofisticación, desde correos electrónicos falsificados hasta sitios web diseñados para parecer legítimos. Los perfiles de redes sociales manipulados también desempeñan un papel importante en la recopilación de información personal y la creación de credibilidad falsa. Los atacantes suelen emplear varias de estas tácticas en conjunto para maximizar las posibilidades de éxito en sus ataques. Este conocimiento fue esencial para la construcción de soluciones preventivas y permitió entender mejor las áreas donde el plugin de inteligencia artificial podía tener un impacto positivo.

Se implementó con éxito un plugin basado en inteligencia artificial que opera en navegadores web y que es capaz de detectar intentos de ataques de ingeniería social, como phishing. El plugin utiliza un modelo de aprendizaje automático basado en redes neuronales LSTM para analizar los correos electrónicos en tiempo real y alertar al usuario sobre posibles amenazas. Esta solución demostró ser efectiva en la identificación de correos electrónicos fraudulentos, proporcionando una capa adicional de seguridad a los usuarios mientras navegan por la web.

El impacto de las amenazas de ingeniería social en la seguridad y privacidad de los usuarios es significativo, afectando principalmente la confidencialidad de la información personal, contraseñas y datos financieros. Los ataques de ingeniería social representan un grave riesgo para la seguridad en línea, y el estudio mostró cómo la falta de conocimiento y concienciación por parte de los usuarios puede aumentar la vulnerabilidad ante estos ataques. La implementación del plugin ayudó a mitigar estos riesgos, ofreciendo una herramienta proactiva de detección que refuerza la protección de los usuarios en la navegación cotidiana.

En conjunto, los resultados de esta investigación destacan la importancia de integrar soluciones de inteligencia artificial en los navegadores para mejorar la detección y prevención de amenazas de ingeniería social, brindando así una mayor seguridad a los usuarios en un entorno digital cada vez más complejo y peligroso.

RECOMENDACIONES

En base a las conclusiones obtenidas a lo largo de esta investigación, se proponen una serie de recomendaciones para mejorar la detección de ataques de ingeniería social y fortalecer la seguridad en la navegación web. Estas recomendaciones buscan no solo optimizar el uso del plugin desarrollado, sino también fomentar mejores prácticas en el ámbito de la ciberseguridad para reducir el impacto de estas amenazas.

Es recomendable que se continúe con la capacitación de los usuarios y las organizaciones sobre las tácticas más comunes de ingeniería social, como phishing, pretexting y spear phishing. La educación en ciberseguridad debe actualizarse constantemente para reflejar las nuevas tácticas que los atacantes desarrollan. Además, los usuarios deben ser capacitados para reconocer señales de manipulación y sospechas en la correspondencia digital, lo que puede ayudar a reducir su vulnerabilidad ante estos ataques.

Dado que los atacantes emplean una amplia gama de técnicas sofisticadas, se recomienda que las soluciones de ciberseguridad, como los plugins de detección, se mantengan actualizadas y se adapten a las nuevas formas de ataques. El análisis de los patrones de phishing y otras técnicas debe incluir un monitoreo continuo de las tendencias emergentes en ingeniería social, y los mecanismos defensivos deben incorporar tecnologías avanzadas como el análisis de perfiles de redes sociales falsos y la verificación de autenticidad de sitios web.

Para optimizar el uso del plugin con inteligencia artificial, se recomienda su implementación y despliegue en entornos empresariales y personales de manera masiva, acompañada de retroalimentación constante por parte de los usuarios. Se sugiere que las organizaciones integren esta herramienta en sus políticas de ciberseguridad, proporcionando a los usuarios capacitación sobre su uso adecuado y estableciendo un proceso para actualizar el modelo de detección basado en nuevas amenazas. Además, se podría mejorar la interoperabilidad del plugin con otros navegadores y clientes de correo electrónico para maximizar su alcance.

Dado el impacto significativo que los ataques de ingeniería social tienen en la privacidad y seguridad de los usuarios, se recomienda realizar evaluaciones periódicas del riesgo y

la exposición a estas amenazas. Las organizaciones deben adoptar medidas de seguridad más proactivas, como el uso de autenticación multifactor y la implementación de políticas rigurosas de protección de datos. También es importante que los usuarios sean informados sobre la importancia de proteger su información personal y financiera y que adopten hábitos más seguros en línea, como la verificación de la autenticidad de las solicitudes de información confidencial.

Estas recomendaciones buscan no solo mitigar los riesgos asociados con la ingeniería social, sino también establecer un entorno digital más seguro a través de la educación, la adopción de nuevas tecnologías y el monitoreo constante de las amenazas emergentes.

REFERENCIAS

- Benavides-Astudillo, E., Fuertes-Díaz, W., & Sánchez-Gordon, S. (2020). “Un experimento para crear conciencia en las personas acerca de los ataques de Ingeniería Social”. . *CIENCIA UNEMI*, 32(13). <https://doi.org/doi:10.29076/issn.2528-7737vol13iss32.2020pp27-40p>.
- Chang, J. E. (2020). Análisis de ataques cibernéticos hacia el Ecuador. . *Revista Científica Aristas*, , 1(1), 18-27.
- COIP. (2021). *Codigo organico integral penal*. https://doi.org/https://www.defensa.gob.ec/wp-content/uploads/downloads/2021/03/COIP_act_feb-2021.pdf
- Dhole, K., Gangal, S. G., Gupta, Z. L., & Mahamood, A. M. (2023). NL-Augmenter: A Framework for Task-Sensitive Natural Language Augmentation. *Northern European Journal of Language Technology*, 1(9). <https://doi.org/https://doi.org/10.3384/nejlt.2000-1533.2023.4725>.
- Dhole, K., Varun Gangal, S. G., Aadesh Gupta, Z. L., & Saad Mahamood, A. M. (2023). “NL-Augmenter A Framework for Task-Sensitive Natural Language Augmentation”. . *Northern European Journal of Language Technology* , 1(9). <https://doi.org/doi:10.3384/nejlt.2000-1533.2023.4725>.
- Díaz, J. P. (2021). Ingeniería Social, un ejemplo practico. . *Revista Odigos*, 2(3), 47-76.
- ECUCERT. (2022). *Vulnerabilidad de seguridad de Zimbra de tipo cross-site scripting (XSS), se explota activamente en ataques dirigidos a organizaciones gubernamentales y medios europeos*. ECUCERT. <https://www.ecucert.gob.ec/wp-content/uploads/2022/02/Alerta-Zimbra.pdf>
- EL COMERCIO. (8 de Agosto de 2022). Ataque informático en la Asamblea retuvo 44 500 correos electrónicos. *Ataque informático en la Asamblea retuvo 44 500 correos electrónicos*. <https://www.elcomercio.com/actualidad/politica/ataque-informatico-retuvo-correos-electronicos-asamblea.html>

- ESET . (21 de Agosto de 2023). *SEGURIDAD DE LA INFORMACIÓN*. SEGURIDAD DE LA INFORMACIÓN: <https://csirt.utpl.edu.ec/node/578>
- Hassan Montero, Y., & Martín Fernández, F. (2005). *La Experiencia del Usuario*”. *No Solo Usabilidad* 4(4). .
- INEC. (2010). *Instituto Nacional de Estadística y Censos*. INEC: https://www.ecuadorencifras.gob.ec/wp-content/descargas/Manualateral/Resultados-provinciales/santa_elena.pdf
- Johnson, M., Orsini, G., León, Y. M., Genaro, P., Cid, M. D., Briones, R., & Pichardo, F. (2022). *La riqueza de nuestros ecosistemas*.
- Kaspersky Co.asd. . (2021). “*Ingeniería social: definición*”. *Kaspersky*. .
- MINTUR. (2019). *Ministerio de Turismo del Ecuador. El Plan Nacional de turismo 2030*. https://www.turismo.gob.ec/wp-content/uploads/2020/03/PLAN-NACIONAL-DE-TURISMO-2030-v.-final-Registro-Oficial-sumillado-comprimido_compressed.pdf
- MSP. (2021). *Informe general*.
- Neuros Center. . (2023). “*El impacto de la inteligencia artificial en la psicología ▷*”. . Neuros Center. .
- Pérez-Cubero, E., & Poler, R. (2020). Aplicación de algoritmos de aprendizaje automático a la programación de órdenes de producción en talleres de trabajo: Una revisión de la literatura reciente”. *Dirección y Organización*, 72(1). <https://doi.org/doi:10.37610/DYO.V0I72.588>.
- Prado Díaz, J. P. (2021). Ingeniería social: un ejemplo práctico. *Revista Odigos*, 2(3), 49-63. . <https://doi.org/https://doi.org/10.35290/ro.v2n3.2021.493>.
- Prado Díaz, J. P. (2021). “Ingeniería social, un ejemplo práctico”. . *REVISTA ODIGOS*, 2(3). <https://doi.org/doi:10.35290/ro.v2n3.2021.493>.
- Rasha, Z., Massari, L., & Calzarossa, M. C. (2023). “Phishing or Not Phishing? A Survey on the Detection of Phishing Websites”. *IEEE Access* 11. . <https://doi.org/doi:10.1109/ACCESS.2023.3247135>.

- Rincón Nuñez, P. M. (2023). “Ataques basados en ingeniería social en Colombia, buenas prácticas y recomendaciones para evitar el riesgo”. *InterSedes* , 24(49). <https://doi.org/doi:10.15517/isucr.v24i49.50345>.
- Rincón Nuñez, P. M. (2023). Ataques basados en ingeniería social en Colombia: buenas prácticas y recomendaciones. . *InterSedes*, 24(49), 17-29. . <https://doi.org/https://doi.org/10.15517/isucr.v24i49.50345>.
- Rios-Paredes, R., & Rios-Salgado. (2020). “PROCESO PARA ESTIMAR EL NIVEL REAL DE CONCIENCIA DE LOS USUARIOS DE FACEBOOK RESPECTO A PRIVACIDAD Y SEGURIDAD EN LA RED SOCIAL”. . <https://doi.org/doi:10.36367/ntqr.4.2020.112-126>.
- Smith, A., Johnson, P., & Lee, S. (2020). Green branding in the digital age: The role of online platforms in promoting eco-friendly products. *International Journal of Digital Marketing*, 1(12), 33-47. . <https://doi.org/https://doi.org/10.5678/ijdm.2020.12.01.33>
- Sperka. (28 de Agosto de 2023). *FLOYDU*. FLOYDU: <https://www.floydu.com/alerta-de-ataque-usuarios-de-correo-electronico-zimbra-en-la-mirilla-de-una-ola-de-robo-de-credenciales/>
- SUPERVISOR EUROPEO DE PROTECCION DE DATOS. (2021). *informe Anual de Ciberseguridad*. https://www.edps.europa.eu/system/files/2022-08/2022-04-20-edps_annual-report_2021-executive-summary_es.pdf
- Trejos, V. M., Peralta, L., López-Lozano, M., Pérez-González, M., & Gómez-Ávila, S. (2021). “Falsos positivos de la ciencia”. . *Revista Mexicana de Fisica E* , 19(1). <https://doi.org/doi:10.31349/REVMEXFISE.19.010301>.
- Vega Velasco, W. (2008). “POLITICAS Y SEGURIDAD DE LA INFORMACION”. . *Fides Et Ratio* , 2(2).
- welivesecurity. (17 de Agosto de 2023). *welivesecurity*. [welivesecurity: https://www.welivesecurity.com/es/investigaciones/campana-phishing-zimbra-afecta-america-latina/](https://www.welivesecurity.com/es/investigaciones/campana-phishing-zimbra-afecta-america-latina/)