Implementación de un sistema de detección de intrusiones a través de la utilización de honeypots especializados, enfocados en la detección de fraude en tarjetas de crédito y en la monitorización de transacciones, diseñados específicamente para su aplicación en entornos de transacciones en línea

Claudio Calvopiña, Mary Elena y Guaján Perugachi, Jimmy Israel

Departamento de Ciencias de la Computación

Carrera de Ingeniería en Software

Trabajo de unidad de integración curricular, previo a la obtención del título de Ingeniero de Software

Dr. Carrillo Medina, José Luis, Ph. D

01 de marzo del 2024

Latacunga

Reporte de Verificación de contenido

Copyleaks

Plagiarism and AI Content Detection Report

V3_Tesina Final_Honeypot_Tarjetas d...

Scan details

Scan time	Total Pages	Total Words
March 1th, 2024 at 16:10 UTC	62	15281

Plagiarism Detection

3.5%

Types of plagiarism		Words
Identical	0.2%	37
Minor Changes	0%	3
Paraphrased	2.7%	416
Omitted Words	15.3%	2331

AI Content Detection

9.3%

Text coverage		Words
AI text	9.3%	1340
Human text	90.7%	11610

Plagiarism Results: (8)

¿Qué es un honeypot? Cómo colaboran los honeypots con la seguridad — 1.1%

https://latam.kaspersky.com/resource-center/threats/what-is-a-honeypot

Skip to main Soluciones para: Para el hogar Empresas pequeñas, 1-50 usuarios Empresas medianas, 51-999 usuarios Corporativo, +1000 usuari...

Recarga tu crédito fácilmente con tu tarjeta de débito en pocos pasos | A... — 0.6%

https://algo.com.ar/credito/se-puede-cargar-credito-con-tarjeta-de-debito/

Saltar al contenido ...

aartolamTFM0723.pdf — 0.4%

https://openaccess.uoc.edu/bitstream/10609/148477/1/aartolamtfm0723.pdf

Mis 18 el 28

Crimen financiero Detección de fraude en tarjetas de crédito aplicando aprendizaje automático Álvaro Artola Moreno Máster Universitario ...

A supervised machine learning algorithm for detecting and predicting fr... — 0.4%

https://www.sciencedirect.com/science/article/pii/s2772662223000036

JavaScript is disabled on your browser. Please enable JavaScript to use all the features on this page. Skip to ...

Dr. Carrillo Medina, José Luis, Ph.D
C. C. 0501553788

Certified by Copyleaks

About this report
help.copyleaks.com

copyleaks.com

Departamento de Ciencias de la Computación

Carrera de Ingeniería en Software

Certificación

Certifico que el trabajo de unidad de integración curricular: "**Implementación de un sistema de detección de intrusiones a través de la utilización de honeypots especializados, enfocados en la detección de fraude en tarjetas de crédito y en la monitorización de transacciones, diseñados específicamente para su aplicación en entornos de transacciones en línea**" fue realizado por los señores **Claudio Calvopiña, Mary Elena y Guaján Perugachi, Jimmy Israel,** el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, además fue revisado y analizada en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que se lo sustente públicamente.

Latacunga, 01 de marzo de 2024

.....

Dr. Carrillo Medina, José Luis, Ph.D
C. C. 0501553788

Departamento de Ciencias de la Computación

Carrera de Ingeniería en Software

Responsabilidad de Autoría

Nosotros, **Claudio Calvopiña, Mary Elena y Guaján Perugachi, Jimmy Israel**, con cédulas de ciudadanía N° 0503960163 y N° 1722761119, declaramos que el contenido, ideas y criterios del trabajo de unidad de integración curricular: **Implementación de un sistema de detección de intrusiones a través de la utilización de honeypots especializados, enfocados en la detección de fraude en tarjetas de crédito y en la monitorización de transacciones, diseñados específicamente para su aplicación en entornos de transacciones en línea,** es de nuestra autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Latacunga, 01 de marzo de 2024

Claudio Calvopiña, Mary Elena
C.C.: 0503960163

Guaján Perugachi, Jimmy Israel
C.C.: 1722761119

Departamento de Ciencias de la Computación

Carrera de Ingeniería en Software

Autorización de Publicación

Nosotros **Claudio Calvopiña, Mary Elena y Guaján Perugachi, Jimmy Israel**, con cédulas de ciudadanía N° 0503960163 y N° 1722761119, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de unidad de integración curricular: "**Implementación de un sistema de detección de intrusiones a través de la utilización de honeypots especializados, enfocados en la detección de fraude en tarjetas de crédito y en la monitorización de transacciones, diseñados específicamente para su aplicación en entornos de transacciones en línea'**, en el Repositorio Institucional, cuyo contenido, ideas y criterios son de nuestra responsabilidad.

Latacunga, 01 de marzo de 2024

.....
Claudio Calvopiña, Mary Elena
C.C.: 0503960163

.....
Guaján Perugachi, Jimmy Israel
C.C.: 1722761119

Dedicatoria

Este proyecto está dedicado con profundo cariño a mis padres, Gonzalo y Clara, así como a mis hermanos Diego y Danny, quienes han sido la fuente inagotable de inspiración en la consecución de mis objetivos. Dedico este trabajo a mi tutor, el Ing. José Luis Carrillo, cuya guía ha sido un pilar esencial en el desarrollo de este proyecto. También, dedico este proyecto a mis amigos que han compartido su amistad, tiempo y risas, en especial a Jimmy, quien ha motivado la conclusión exitosa de esta iniciativa. Y a todos aquellos que han sido parte de este camino, mi más sincero agradecimiento. Finalmente, pero no menos importante, me dedico este logro, reconociendo el esfuerzo diario y los sacrificios realizados, ya que la clave del éxito reside en la constancia.

Claudio Calvopiña, Mary Elena

Ecuador, marzo de 2024

Dedicatoria

Dedico este logro a los pilares fundamentales de mi vida: a mi madre María Perugachi, cuyo amor incondicional y sacrificios han sido mi mayor inspiración y fortaleza. Tu presencia ha sido mi guía constante y tu apoyo inquebrantable ha sido mi sostén en los momentos más difíciles. A mi hermano Richard Guaján, quien ha sido no solo un hermano, sino también una figura paterna, agradezco profundamente por su dedicación, orientación y amor incondicional a lo largo de este viaje académico. A mi hermano Andy Guaján, cuya presencia y apoyo han sido una constante en mi vida, te agradezco por ser un pilar de fuerza y aliento en este camino. A mi hermano Dillan Guaján, quien está destinado a seguir nuestros pasos, le dedico este logro como una inspiración para su propio camino. Que este logro sea un recordatorio de que, con dedicación y perseverancia, cualquier meta es alcanzable. A Diana C., la chispa que ha avivado mi motivación y determinación, te agradezco por tu apoyo constante y tu presencia alentadora en cada paso de este camino académico. Este logro no solo es mío, sino también tuyo. A Mary C., quien, sin su guía y apoyo incondicional, esta tesis no habría culminado. Tu sabiduría y orientación fueron la luz que iluminó el camino hacia este logro. A todos ustedes, les dedico este logro con profundo agradecimiento. Su amor, apoyo y presencia han sido fundamentales en este camino, y por eso estaré eternamente agradecido.

Guaján Perugachi, Jimmy Israel

Ecuador, marzo de 2024

Agradecimiento

Quiero expresar mi profundo agradecimiento a Dios por la fortaleza y bendiciones que ha brindado en cada paso de mi vida. Mi gratitud eterna se dirige a mis padres, quienes han sido la constante motivación y apoyo en la consecución de mis metas personales y profesionales. Agradezco sinceramente a mis docentes por compartir sus conocimientos con paciencia y dedicación en las aulas de clase. A la Universidad de las Fuerzas Armadas ESPE, Sede Latacunga, le agradezco por abrirme las puertas del conocimiento y la formación académica. A mis compañeros y amigos, les estoy agradecida por la amistad y colaboración brindada en cada semestre.

Claudio Calvopiña, Mary Elena

Ecuador, marzo de 2024

Agradecimiento

En primer lugar, doy gracias a Dios por iluminar mi camino y a mi madre, María Perugachi, cuyo amor incondicional y apoyo constante han sido mi roca durante toda esta travesía académica. A mis hermanos Richard, Andy y Dillan Guaján, les agradezco por su constante aliento y apoyo. Son el verdadero sostén de mi vida. A Mary C., mi mejor amiga, gracias por estar siempre a mi lado, por ser mi confidente y por su inquebrantable apoyo en cada paso de este camino. A Anderson L., quien más que un amigo, se convirtió en mi hermano, gracias por tu compañía, tu apoyo incondicional y por compartir cada momento de esta etapa de mi vida. A María Belén T., quiero expresar mi profundo agradecimiento por tu apoyo y aliento durante los momentos más difíciles de esta etapa. Tu presencia ha sido un regalo invaluable en mi vida. A Ismael S. y Nelson M., mis compañeros de música y aventuras, les agradezco por compartir risas, alegrías y por ser parte de los recuerdos más preciados de esta etapa de mi vida. A Helen M., a pesar de nuestras diferencias, agradezco tu contribución en este logro. A Sofia A., una de mis grandes amigas, agradezco su amistad sincera y por estar presente en los momentos más importantes de mi vida. Al Ing. Carrillo, mi tutor, agradezco sinceramente su dedicación y orientación experta que provocó la culminación de este proyecto con éxito. A mis compañeros de carrera y a cada uno de los ingenieros que tuve el privilegio de recibir su enseñanza, les agradezco por compartir su conocimiento y experiencias, que han enriquecido enormemente mi formación académica y profesional. A todos y cada uno de ustedes, gracias por formar parte de mi viaje. Su apoyo, amistad y presencia han sido fundamentales en la consecución de este logro. Estoy eternamente agradecido.

Guaján Perugachi, Jimmy Israel

Ecuador, marzo de 2024

ÍNDICE DE CONTENIDO

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

Resumen

En la era digital, la creciente demanda de seguridad cibernética se hace cada vez más evidente. Los ataques cibernéticos, especialmente en sistemas de transacciones online, presentan desafíos significativos, como fraudes con tarjetas de crédito robadas o perdidas. Para hacer frente a esta problemática, se han desarrollado los honeypots, un sistema de seguridad que simula vulnerabilidades para atraer a los atacantes. El propósito es estudiar sus métodos y tácticas, complementando así las estrategias tradicionales y fortaleciendo la defensa contra amenazas como fraudes con tarjetas de crédito. Se ha implementado un sistema de transacciones online con el patrón arquitectónico Modelo-Vista-Controlador junto a la creación de un honeypot integrando dos modelos de Machine Learning y un dataset simulado. Para las pruebas, la validación cruzada reveló que la matriz de confusión en Random Forest destacó una mejora significativa en la identificación de transacciones anómalas, con altos valores de verdaderos negativos y positivos. No obstante, en regresión logística, se observó un rendimiento considerablemente menor en comparación al modelo anterior. Aunque la Regresión Logística identificó la mayoría de las transacciones anómalas, su rendimiento global fue inferior en términos de precisión y exactitud. La metodología adoptada desempeñó un papel crucial en el logro exitoso de los objetivos. Su enfoque iterativo y la atención constante a la retroalimentación fueron fundamentales para la conclusión exitosa del proyecto.

Palabras clave: honeypot, sistema de detección, bosques aleatorios, regresión logística, tarjetas de crédito.

Abstract

In the digital age, the growing demand for cyber security is becoming increasingly evident. Cyber-attacks, especially in online transaction systems, present significant challenges, such as fraud with stolen or lost credit cards. To address this issue, honeypots, a security system that simulates vulnerabilities to lure attackers, have been developed. The purpose is to study their methods and tactics, thus complementing traditional strategies and strengthening defense against threats such as credit card fraud. An online transaction system has been implemented with the Model-View-Controller architectural pattern along with the creation of a honeypot integrating two Machine Learning models and a simulated dataset. For testing, cross-validation revealed that the confusion matrix in Random Forest highlighted a significant improvement in the identification of anomalous transactions, with high values of true negatives and positives. However, in Logistic Regression, significantly lower performance was observed compared to the previous model. Although Logistic Regression identified most of the anomalous transactions, its overall performance was lower in terms of precision and accuracy. The methodology adopted played a crucial role in the successful achievement of the objectives. Its iterative approach and constant attention to feedback were fundamental to the successful completion of the project.

Keywords: honeypot, detection system, random forest, logistic regression, credit cards.

Capítulo I

Introducción

Propósito y contextualización del tema

La demanda de la seguridad cibernética crece en la era digital, un ataque cibernético puede llegar a ser muy difícil de tratar, tanto para individuos como para organizaciones, por lo cual es crucial comprender los diferentes tipos de ataques y aumentar las estrategias de seguridad para protegerse contra ellos (Kaur et al., 2014).

Uno de los ataques más difundidos son los fraudes en tarjetas de crédito y las transacciones fraudulentas línea, Cuál representa una preocupación constante en el ámbito de la ciberseguridad (Rayo Mondragón, 2020).

Los delincuentes cibernéticos tienen diversas técnicas de ataque como el robo de los datos, el skimming con el cual pueden tener información confidencial para realizar transacciones fraudulentas que significan pérdidas financieras (Bringer et al., 2012).

Uno de los casos destacados concine a la empresa Target, que después de Walmart, es la segunda cadena de tienda más completa en EEUU, entre el 25 de noviembre y el 15 de diciembre de 2013 afectó los datos de más de 40 millones de tarjetas de crédito y débito. Los hackers accedieron a los datos de las bandas magnéticas encontrados en las tarjetas, lo que eventualmente les permitió falsificar las tarjetas y robar su información (Daza, 2013).

Como resultado, Target sufrió pérdidas significativas en términos de reputación y confianza de los clientes. Target implementó medidas de seguridad adicionales, ofreció servicios de monitoreo de crédito gratuitos a los clientes afectados y realizó inversiones en tecnología para fortalecer su infraestructura de seguridad (Daza, 2013).

Para contrarrestar los fraudes de tarjetas de crédito, a lo largo del tiempo, se ha involucrado una combinación de medidas de ciberseguridad y buenas prácticas en la industria financiera. Algunas de estas estrategias y enfoques para abordar este problema incluyen los

sistemas de detección y prevención de intrusiones (IDS/IPS) que ayudan a identificar patrones de tráfico inusuales y actividades maliciosas en la red (Yamamoto & Yamaguchi, 2022).

Otra estrategia es el análisis de comportamiento donde se realiza un monitoreo del comportamiento de los usuarios y las transacciones para detectar anomalías. Incluso, el uso de algoritmos de aprendizaje automático puede detectar patrones y comportamientos anómalos en los datos (Rayo Mondragón, 2020).

Además, se emplean prácticas como la verificación multifactor (MFA), encriptación y protección de datos, la tokenización, el monitoreo continuo de las transacciones. También se incluye la creación de listas negras y listas blancas, junto con la colaboración sectorial entre instituciones financieras y proveedores de servicios, y la educación del usuario (Nicolás Vidal, 2022).

Sin embargo, actualmente, los IPS/IDS no han resultado ser lo suficientemente efectivos ante amenazas, mismas que evolucionan constantemente para realizar fraudes en tarjetas de crédito, aunque estos métodos tradicionales son parte integral de la seguridad de red, la combinación de estos sistemas con soluciones más avanzadas de inteligencia de amenazas proporcionan tácticas que utilizan y cómo operan los ciber-delincuentes en sus intentos de ataques, información valiosa para mejorar las estrategias de seguridad (Rong & Yang, 2003).

Un sistema de seguridad que simula tener vulnerabilidades para atraer a los atacantes con la finalidad de poder estudiar sus métodos y tácticas, son las denominadas honeypot que, junto a los métodos tradicionales, se complementan y fortalecen para contrarrestar amenazas como los fraudes de tarjetas de crédito (Muhamad Malik Matin & Rahardjo, 2020).

Para detectar y prevenir el fraude en tarjetas de crédito y las transacciones anómalas en línea, existen Honeypots especializados que surgen como una herramienta efectiva. De esta manera, son configurados para simular ambientes de transacciones en línea que detecten y analicen los intentos del fraude (Rong & Yang, 2003). Por ejemplo, los honeypot que capturan datos o información, de manera especial los de tarjetas de crédito y los formularios de entrada

en páginas web o aplicaciones (*ICCSM2015-3rd International Conference on Cloud Security and Management*, 2015).

Además, existen honeypots que trabajan sobre formularios financieros, los cuales imitan páginas de registro de datos financieros más amplios, como formularios de inicio de sesión de cuentas bancarias o aplicaciones de pago en línea (*ICCSM2015-3rd International Conference on Cloud Security and Management*, 2015).

Justificación e Importancia

La implementación de un sistema de detección de intrusiones a través de la utilización de honeypots especializados para la detección de fraude en tarjetas de crédito y la monitorización de transacciones en entornos de transacciones en línea se ha vuelto cada vez más relevante en el contexto actual de ciberseguridad (Lutkevich, 2023). Existe la necesidad apremiante de abordar las amenazas cibernéticas que se dirigen a las transacciones financieras en línea.

Los honeypot son una estrategia proactiva con la capacidad de hacerle frente a estos desafíos, puesto que, proporcionan una trampa digital e incitan a los atacantes sin comprometer sistemas verdaderos (Ilg et al., 2023). Simulan de manera realista las interacciones de pago y datos financieros, aumentando la probabilidad de detección y disuasión de ataques financieros con el beneficio de fortalecer la ciberseguridad (Safaei Pour et al., 2023). Reduce las pérdidas económicas y persevera la confianza de los usuarios al tiempo que permite una mejor comprensión y mitigación de las amenazas (David Tidmarsh, 2023).

Los honeypot financieros han evolucionado en respuesta a los cambios en las tácticas de ataque y las necesidades de seguridad en el ámbito financiero, esto ha llevado a mejoras significativas en la protección de áreas clave (Muhamad Malik Matin & Rahardjo, 2020).

Actualmente, los señuelos informáticos han elevado su grado de realismo y autenticidad al simular entornos de transacciones. Esto se logra mediante el uso de técnicas avanzadas de detección, como análisis de comportamiento y aprendizaje automático (Kumar et al., 2019). Los

cuales capturan acciones de los atacantes en tiempo real, generan señuelos personalizados, comparten inteligencia y permiten respuestas automatizadas, además, estos se integran con sistemas de seguridad y se centran en el ciclo de vida completo de un ataque (Yan et al., 2022).

Objetivos

Objetivo General

Desarrollar un sistema de detección de intrusiones a través de la utilización de la utilización de honeypots especializados, enfocados en la detección de fraude en tarjetas de crédito y en la monitorización de transacciones, diseñados específicamente para su aplicación en entornos de transacciones en línea.

Objetivos Específicos

- Investigar el estado del arte sobre el funcionamiento de los honeypot, así como de los distintos tipos de honeypots diseñados para rastrear actividades de la línea, las técnicas computacionales avanzadas empleadas en su implementación y adquirir un conocimiento detallado acerca de su proceso de desarrollo en un sistema de transacciones online.
- Implementar un honeypot de tarjetas de crédito que permita detectar el fraude en un entorno de transacciones en línea y realizar la monitorización de las transacciones completadas.
- Validar los resultados, analizar los errores y ajustar el honeypot en un sistema de transacciones online.

Metodología

Para cumplir el objetivo, se empleará una metodología compuesta de tres etapas, en la primera etapa se realizará un estudio de la literatura centrado en la revisión y análisis de las fuertes existentes. Además de buscar información sobre la terminología, la funcionalidad y el

conocimiento sobre el proceso de desarrollo, así como de los distintos honeypot empleados en el rastreo de actividades en línea.

En la segunda fase se implementa un sistema simulado de transacciones online conectado con la entidad de las tarjetas de crédito, de dónde se obtiene el historial del usuario, esta información recopilada se dirige al honeypot donde, con el modelo, se detectará el fraude o la autenticidad de la tarjeta de crédito, acorde a la decisión, se guardará en la base de datos.

Dentro de los pasos se empieza por establecer los objetivos, elegir las herramientas y crear un entorno seguro. Diseñar un honeypot que simule las transacciones y el fraude elaborando un registro. Generar transacciones falsas y actividades maliciosas para implementar el registro y el análisis para capturar las interacciones, además de definir reglas y umbrales para las alertas. Transferir el honeypot a la etapa de prueba siguiendo mejores prácticas. Monitorear para detectar ataques maliciosos con la clonación de tarjetas de crédito. Analizar datos del historial de las tarjetas de crédito en la búsqueda de patrones relacionados con el análisis del Internet Protocol (IP) del dispositivo que hizo la transacción, el lugar (ciudad), el monto o cantidad y el horario representado en fecha y hora. Se analiza con base a un comportamiento similar en cada transacción, y al detectar una acción diferente, se determina el fraude a través de algoritmos de Machine Learning (bosques aleatorios o regresión logística), y generan informes de respuesta.

El dataset necesario requiere de datos de tarjetas de crédito, los cuales serán creados aleatoriamente con un algoritmo de Python, el cual genera los números de las tarjetas y los datos para el historial, explicados en el Capítulo III. Se ha optado por esta medida, debido a la confidencialidad de las empresas al generar las tarjetas de crédito. Los algoritmos de aprendizaje automático son puestos a prueba para seleccionar el de mejor rendimiento de acuerdo con los objetivos esperados.

En la tercera etapa se realiza la validación, analizando los errores y ajustando los modelos. Además, se analizará el rendimiento de los modelos de Machine Learning utilizando

técnicas como la validación cruzada y métricas de evaluación como precisión, exactitud y exhaustividad, para asegurar que los modelos sean capaces de detectar con satisfacción.

Capítulo II

Marco Teórico

Este capítulo tiene como objetivo hacer una investigación teórica para conocer los elementos de un sistema de detección de intrusiones, abarcando la definición y terminología sobre honeypots, clases, tipos y usos. Así como una revisión de la literatura sobre técnicas, modelos y/o algoritmos usados para el desarrollo e implementación de un sistema, que, a través de la utilización de honeypots, pueda detectar el fraude en las tarjetas de crédito, además de realizar una monitorización de las transacciones. Estos objetivos se esperan cumplir con la metodología seleccionada que se explica al final de capitulo.

Para efectuar la investigación, se llevó a cabo una revisión de la literatura en Scopus, una base de datos bibliográfica de resuenes y citas de artículos de revistas. Como paso inicial, se planteó determinar la ceda de búsqueda la cual fue construida con palabras caves del tema y aplicada en Scopus, después de realizar varias iteraciones de búsqueda, fue ajustada a las necesidades de estudio. Con la cadena revinada, se encontró y selecciono los artículos que tienen una estrecha relación con el tema y son relevantes, los cuales sirvieron para la elaboración de este capítulo.

Sistema de detección de intrusiones

Un sistema de detección de intrusiones es una herramienta de seguridad para redes, onde, ya sea con software o hardware, supervisa actividades maliciosas o detecta violaciones de políticas en una red o sistema (Rakesh et al., 2019).

Dentro de las funciones y características principales está el monitoreo continuo para observar el tráfico de red y las actividades en los hosts, con el objetivo de detectar (supervisar o identificar) patrones anómalos mediante una comparación entre la actividad actual con un perfil normal establecido, y detecta firmas al utilizar bases de datos de firmas conocidas de ataques para identificar patrones específicos asociados con malware o intrusiones, y entrega

alertas y notificaciones inmediatas en caso de detectar actividades sospechosas, permitiendo una respuesta rápida ante posibles amenazas, finalmente se registra y se reporta (IBM, s. f.).

Un honeypot es una herramienta importante para identificar intrusiones en la red. Estos no solo enfrentan al atacante, sino que también recolectan datos que resultan útiles para entender los ataques y a los propios atacantes en la red (Edwin et al., 2022).

Honeypots

Los honeypots son creados para ser usados con diversos fines en el campo de la ciberseguridad y la protección contra amenazas informáticas, primero se presenta como una herramienta para investigar las tácticas que usan los atacantes, logrando recopilar información sobre las ciber amenazas conocidas como el escaneo de puertos o ataques de fuerza bruta, y aquellas nuevas que aparecen en esta evolución, además, ayuda en la confusión y desvía la atención de los sistemas reales, y al acercarse al ámbito académico, este tipo de tecnología es capaz de apoyar los procesos en el momento del aprendizaje de los estudiantes que se desarrollan en el área de seguridad informática (Díaz & Cuervo, 2018).

A continuación, se presenta la definición de honeypot según los autores:

En 1986 se da la primera mención del término “honeypot” definido como un tarro de miel, gracias a Clifford Stoll en su libro The Cuckoo's Egg, donde no lo menciona literalmente, pero es usado para dar un nombre a la experiencia que relata cuando estuvo persiguiendo a un hacker quien había atacado a la red Lawrence Berkeley National Laboratory, con el término describe cómo creó una "trampa" para atraer y rastrear al intruso (Stoll, 1989).

Poco tiempo después, en el año 2000, se desarrolló la idea, de honeypots, convirtiéndose en algo más popular y otorgaron el concepto de que un honeypot en el libro: “Honeypots: Tracking Hackers” según (Spitzner, 2003) es "una herramienta de seguridad destinada a ser atacada y comprometida, con el objetivo de recopilar datos sobre las tácticas, técnicas y procedimientos de los atacantes".

Por otro lado, en el año 2007, se define a un honeypot en el libro Virtual Honeypots de (Vidalis & Kazmi, 2007) como “una trampa de seguridad que tiene como objetivo detectar intentos de intrusión o actividad maliciosa en una red informática”.

Los honeypots pueden utilizarse para una variedad de propósitos, como la detección de intrusos, la investigación de amenazas y la capacitación de personal de seguridad, recopilar información sobre los atacantes, como sus métodos, herramientas y motivaciones (Vidalis & Kazmi, 2007).

En definición, un honeypot es una herramienta de seguridad informática que tiene como objetivo atraer a los atacantes y desviar su atención de los sistemas reales de la red, permitiendo así estudiar sus métodos y tácticas. Funciona simulando ser un sistema vulnerable o recurso atractivo para los atacantes, de modo que cuando un ataque es detectado en el honeypot, se activan alertas y se recopila información sobre el atacante y sus técnicas, lo que puede ayudar a fortalecer la seguridad de la red principal.

A continuación, se plantean los hitos resumidos a lo largo de los inicios hasta la presente fecha de investigación que se han identificado en relación con los honeypots:

En 1986, Clifford Stoll utiliza el término "honeypot" para describir un sistema de trampa diseñado para detectar intrusos en su red (Stoll, 1989).

En 1991, Clifford Stoll acuña oficialmente el término "honeypot" en su libro (Stoll, 1989). En 1997, se lanzó la versión 0.1 del Deception Toolkit de Fred Cohen, el cual sería una de las primeras soluciones de tipo honeypot disponibles para los miembros de la comunidad de seguridad (Ahmed, 2023).

En1998, comenzó el desarrollo de CyberCop Sting, uno de los primeros Honeypots comerciales disponibles al público. Presenta la idea de múltiples sistemas virtuales que se encuentran vinculados a un único honeypot (Ahmed, 2023).

Además, es publicado el NetFacade (con el concepto Snort) por Marty Roesch y GTE Internetworking y el BackOfficer Friendly, un Honeypot público y gratuito basado en Windows

(González, 2021). En 1999, se funda el "Honeynet Project" marcando el inicio de la investigación y el desarrollo de honeypots a gran escala (*The Honeynet Project*, 2021).
En 2000, Lance Spitzner, consultor y analista informático quien es experto en seguridad, construyó una red de seis ordenadores ubicados en su propia casa con el objetivo de realizar una investigación sobre los tipos de ataques y, sobre todo, la manera en la que estos eran efectuados, para lo que usó Honeypots. La experiencia de laboratorio fue recogida en su libro “Honeypots: Tracking Hackers” (Spitzner, 2003).

Además, en el año 2000, se introduce el concepto de honeypot de baja y alta interacción.

Entre 2000 y 2001, se incrementa el uso de estos sistemas, usados no solo para capturar información, sino para estudiar la actividad de los gusanos. Muchas organizaciones implementan Honeypots para investigar amenazas nuevas. En 2002, se utiliza un Honeypot para detectar y capturar en acción un ataque desconocido en ese entonces, específicamente el exploit que perjudicó al servicio de panel de control de escritorio (dtspcd) del sistema Solaris del sistema operativo Unix (González, 2021).

En 2003, The Honeynet Project publica el libro "Know Your Enemy: Learning about Security Threats" cuyo fin es proporcionar información detallada sobre amenazas y ataques (*The Honeynet Project*, 2021).

En 2006, las empresas empezaron a integrar la tecnología de señuelos como método de seguridad (Baykara & Das, 2015).

Más adelante, en 2012, salen al público los honeypots de código abierto, por mencionar algunos, están Dionaea y Glastopf (Zambrano et al., 2021).

En el año 2016, el usó de Honeypot incrementó para los ataques de tipo malware y ransomware, dos años más tarde, en 2018, el estudio y la curiosidad se toman las publicaciones sobre el tema, por ejemplo, lo aplican con el Internet de las Cosas (IoT)

(González, 2021). Para 2019, son una herramienta fundamental para la ciberseguridad, puesto que no solo detectan amenazas, sino que, permiten investigar los ataques y recopilar la información para aplicarlos con modelos y/o algoritmos de aprendizaje automático en el enfoque de inteligencia de amenazas que hasta la fecha sigue en evolución (Leyden, 2020).

Un honeypot puede actuar de dos maneras, pasivo y activo, dependerá del tipo y la configuración elegida (Yang, 2008). Los señuelos pasivos registran el movimiento del atacante, sin embargo, no están diseñados para detectar, disuadir o mitigar ataques como un honeypot activo (Yang, 2008).

Clasificación de Honeypots

Un honeypot no se configura para un problema en específico, como un firewall o un antivirus, es una herramienta que informa sobre el ataque y su movimiento, permitiendo al administrador tomar decisiones respecto a la seguridad (González, 2021).

Con base al criterio de uso, los honeypots se pueden clasificar en tres, por la implementación, por el medio o entorno y por la complejidad o nivel de interacción (Osorio, 2021).

Respecto al propósito, se subdividen en dos, de investigación y de producción, con el mismo principio de ciberseguridad, pero con dos objetivos distintos.

Honeypots de Investigación. Son sistemas que sirven como recursos educativos, el objetivo es recopilar información del ataque para estudiar los patrones de la amenaza. Se despliegan en entornos académicos y de investigación, sin la necesidad fundamental de proteger las redes (Osorio, 2021).

Honeypot de Producción. Son sistemas que desvelan los ataques, protegiendo los datos reales de las medianas y grandes organizaciones, se despliegan para proteger las redes

e infraestructuras tecnológicas de información. Además, comprenden las motivaciones que llevó al atacante a invadir el sistema (Osorio, 2021).

Por el medio usado, se clasifican en virtuales o simulados, en emulados y los dispositivos reales.

Simulados. Imita a un dispositivo real mediante el uso de recursos ficticios, salidas preprogramadas o patrones definidos. La principal ventaja es la escalabilidad, y el mantenimiento es relativamente sencillo (Gómez, 2018).

Emulados. Son plataformas que proporcionan una infraestructura virtual para que el honeypot se desempeñe, en ocasiones, son usados como cajas de arena, donde es fácil regresar a la versión anterior (Gómez, 2018).

Dispositivos Reales. Es un dispositivo hardware, resulta más costoso y el mantenimiento requiere de mayor esfuerzo. La ventaja es la elevada interacción que tiene con el atacante (Gómez, 2018).

Para la interacción con el atacante o el nivel de complejidad de la implementación del honeypot, se clasifica en alta, baja, y media, destacando la premisa que, a mayor interacción, más funciones y dificultades tendrán el atacante, y quien configure, requerirá de un trabajo con más dedicación.

Honeypot de Baja Interacción. Mantienen un diseño básico, son relativamente fáciles de implementar. Pueden llevarse a cabo en entornos o máquinas virtuales, o en máquinas físicas. Principalmente recolectan información como, fechas, horas, dirección IP, y puertos (González, 2021).

Honeypot de Media Interacción. Presentan una interacción más amplia respecto a los servicios, en ciertos casos, requiere de conocimientos adicionales como protocolos de implementación para un correcto funcionamiento (González, 2021).

Honeypot de Alta Interacción. Se instalan en sistemas reales, con zonas accesibles para cualquier usuario en una organización, sean entornos físicos o virtuales. La implementación es la más complicada y requiere de más tiempo. El beneficio es la obtención sobre los atacantes debido a que mantiene al atacante en el sistema un tiempo prolongado, y conoce los movimientos de ataque (González, 2021).

Honeypot Puro. Es un sistema a gran escala que imita completamente al de producción y se ejecuta en varios servidores, contiene datos confidenciales, sensores e información de usuarios. Son complejos y difíciles de mantener (González, 2021).

La clasificación se basa en criterios generales como la complejidad en la implementación y el nivel de interacción, mientras que, al dividir en tipos de honeypots, se refiere a categorías específicas en el diseño y la función que cumplen en el entorno.

Tipos de Honeypot

Tabla 1

Tipos de Honeypot

Honeypot	Descripción	Referencia
De Malware	Enfocado en la detección de malware, lo que significa que reconoce los puntos específicos donde el malware lleva a cabo sus ataques y procesos de reproducción.	(Lutkevich, 2023)
De Spam	Detecta las técnicas que emplea el atacante para los correos no deseados, supervisar las actividades,	(Lutkevich, 2023)

Honeypot	Descripción	Referencia
	además de bloquear el envío de correos no deseados.	
De Base de Datos	Usan métodos que los cortafuegos pasan por alto para engañar a los atacantes y proteger las bases de datos.	(Lutkevich, 2023)
De clientes	Rastrean servidores sospechosos, en lugar de esperar de forma pasiva el ataque, detectan modificaciones anómalas en el honeypot.	(Lutkevich, 2023)

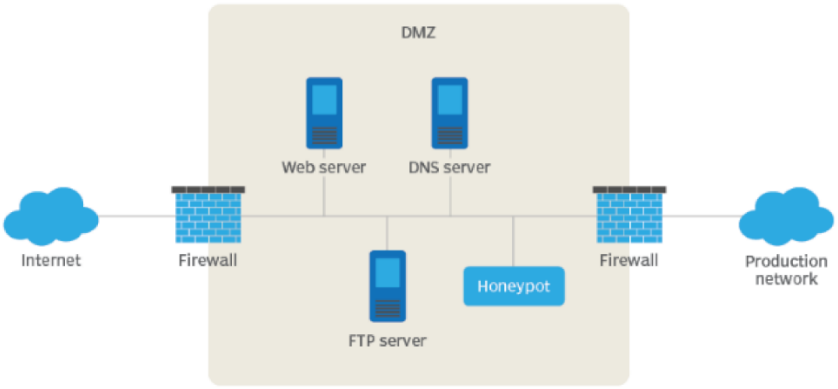
Nota. En la tabla se muestra alguno de los tipos de honeypot, con la descripción y el autor.

Ubicación del Honeypot

En términos generales, un honeypot opera mediante una configuración que incluye una computadora, aplicaciones y datos diseñados para imitar el comportamiento de un sistema real, con el propósito de atraer a posibles atacantes. Por ejemplo, podría simular un sistema financiero, dispositivos de Internet de las cosas (IoT) o una red de transporte o servicio público. Aunque el honeypot aparenta ser parte de una red, en realidad está aislado y es objeto de un monitoreo cercano. Dado que no hay razón legítima para que usuarios reales accedan al honeypot, cualquier intento de comunicación con él se considera hostil. Se presenta a continuación, en la Figura 1, un ejemplo sobre la ubicación de un honeypot, sin poner en riesgo la red principal.

Figura 1

Ubicación de un honeypot



Nota. En la figura se muestra la ubicación del honeypot en una red dentro de la zona desmilitarizada. Tomado de (Zola, 2023).

Conforme se evidencia en la Figura 1, la colocación típica de los Honeypots ocurre en una zona desmilitarizada (DMZ) de la red. Esta ubicación asegura su aislamiento de la red principal de producción, aunque sigue siendo una entidad conectada a ella. Además, en la DMZ, un honeypot puede ser supervisado de manera remota mientras los posibles atacantes acceden a él, lo que reduce al mínimo el riesgo de que la red principal se vea comprometida (Zola, 2023).

Honeypots en sistemas transaccionales online

Debido a la naturaleza sensible de las operaciones de los honeypots y los riesgos de seguridad que implica la divulgación de detalles, la información específica sobre qué honeypots han sido ocupados en los sitios web de ventas generalmente no se hace pública. Esto es para evitar alertar a los atacantes y permitirles adaptar sus tácticas para evitar futuros honeypots (Chiluiza, 2008).

Tipos de Honeypots

Estos honeypots atraen a posibles atacantes que buscan llevar a cabo actividades como inyecciones en sitios web, extracción de información de precios y contenido, o intentos no

autorizados, incluyendo el acceso a cuentas de usuario (Lutkevich, 2023). A continuación, se presentan algunos tipos de honeypots enfocados en transacciones online.

Honeypots de red. Estos honeypots se configuran en la red y se utilizan para atraer a los atacantes que escanean la red en busca de sistemas vulnerables (Lutkevich, 2023). Honeypots de sistema. Estos honeypots se configuran en un sistema individual y se utilizan para atraer a los atacantes que intentan comprometer un sistema específico (Lutkevich, 2023).

Honeypots de datos. Estos honeypots contienen datos confidenciales, como números de tarjetas de crédito o contraseñas. Se utilizan para atraer a los atacantes que buscan robar estos datos (Lutkevich, 2023).

Honeynets. Estas son redes de honeypots que se utilizan para estudiar el comportamiento de los atacantes en un entorno realista (Walter, 2023).

Honeypots en Tarjetas de Crédito

El honeypot simula ser un sistema informático real, que está completo con aplicaciones e incluye datos, tiene el propósito de engañar a los ciberdelincuentes y hacerles creer que están en un programa u objetivo real (Mokube & Adams, 2007).

Una vez que los hackers logran acceso, es posible rastrearlos y evaluar su comportamiento con el objetivo de obtener pistas sobre cómo actúan. Estos honeypots resultan atractivos para los atacantes al incorporar deliberadamente vulnerabilidades de seguridad.

Por ejemplo, pueden existir puertos que respondan a un escaneo o incluso, el caso de las contraseñas débiles. Además, se pueden dejar intencionadamente puertos vulnerables abiertos para atraer a los atacantes al entorno del honeypot, en lugar de la red real (Mokube & Adams, 2007).

Métodos Fraude con Tarjetas de Crédito y Monitorización de Transacciones

El fraude con tarjetas de crédito puede incluir el uso no autorizado de tarjetas de crédito o débito para realizar compras, ya sea a través del robo físico de la tarjeta o, también es común el robo de información de la tarjeta a través de internet (Roldan et al., 2017).

Algunas tácticas incluyen el uso de tarjetas de crédito robadas para realizar compras en línea, el cual es el enfoque del proyecto, la creación de sitios web de comercio electrónico falsos para robar información de tarjetas, o la manipulación de sistemas de revisión para engañar a los compradores (Roldan et al., 2017).

Evitar estos peligros es una preocupación fundamental para las empresas, y hoy en día, la tecnología está a su favor, mediante la detección de conductas de comportamiento inusuales (Rayo Mondragón, 2020), siendo esta última, la usada en el proyecto.

Pérdida de Tarjetas de Crédito

La pérdida o robo de tarjetas de crédito conlleva riesgos significativos de fraude, particularmente en el ámbito de las transacciones en línea. Cuando una tarjeta de crédito cae en manos equivocadas, el individuo que la posee puede acceder fácilmente a la información crucial de la tarjeta, incluyendo el número, la fecha de vencimiento y el código de seguridad (CVV), permitiéndole realizar compras no autorizadas en internet o por teléfono, donde no es necesario presentar físicamente la tarjeta.

Esta situación se hace más complicada por la existencia de sitios web fraudulentos diseñados específicamente para capturar y abusar de los detalles de tarjetas de crédito (Alkhalil et al., 2021).

La pérdida o robo de una tarjeta de crédito puede conllevar a una serie de eventos ilegales, lo que pone en riesgo la seguridad financiera del titular de la tarjeta (Langwagen, 2019).

Clonación de Tarjetas de Crédito

La clonación de tarjetas de crédito es un acto ilegal que implica hacer copias no autorizadas de las mismas, este proceso puede llevarse a cabo de diversas maneras, como el uso de dispositivos electrónicos llamados skimmers para copiar la información de la tarjeta y crear una tarjeta nueva (Roldan et al., 2017).

Los skimmers son dispositivos que se colocan en cajeros automáticos, restaurantes, gasolineras, tiendas u otros lugares donde se pueda perder se use la tarjeta, y la clonación es rápida y discreta. Una vez que la información ha sido clonada en una nueva tarjeta, los delincuentes pueden utilizarla (Roldan et al., 2017).

Para el sistema de detección de intrusiones se ha tomado en cuenta las dos formas de fraude, la pérdida y clonación de tarjetas de crédito aplicados en un entorno de transacciones en línea.

Análisis de datos y técnicas computacionales para de honeypots

Los modelos de machine learning se utilizan cada vez más en los honeypots para la detección de transacciones online fraudulentas. Estos modelos pueden ayudar a identificar patrones en los datos que pueden ser indicativos de actividad maliciosa (Vishwakarma & Jain, 2019).

En el tema de los sistemas de detección de intrusiones, existen diversos modelos de machine learning que son ampliamente empleados en honeypots, en la Tabla 2 se detallan algunos:

Tabla 2

Modelos de machine learning para sistemas de detección de intrusiones

Modelo y/o algoritmo	Descripción	Precisión	Referencia
Clasificadores de árboles de decisión (Decision Trees)	Son modelo de aprendizaje automático supervisado que se utiliza para clasificar los datos en dos o más categorías. Se pueden utilizar para	91.14%	(Huang et al., 2019)

Modelo y/o algoritmo	Descripción	Precisión	Referencia
	clasificar las transacciones como legítimas o fraudulentas		
Bosques aleatorios (Random Forest)	Son un conjunto de árboles de decisión entrenados en distintos subconjuntos de datos y features. Para la clasificación, cada árbol vota por una clase y se elige la que obtiene más votos. Para regresión, se promedia el valor predicho por cada árbol.	97.7%	(Alvarado et al., 2022)
Regresión Logística	Es una técnica de aprendizaje automático supervisado para modelar la probabilidad de que una observación pertenezca a una	97.44%	(Calvo Pérez, 2021)

Modelo y/o algoritmo	Descripción	Precisión	Referencia
	categoría específica. Se adapta de manera óptima para anticipar si una transacción es fraudulenta o legítima.		
Redes neuronales	Es un modelo de aprendizaje automático no supervisado que se utiliza para aprender patrones en los datos. Se pueden utilizar para identificar patrones que pueden ser indicativos de actividad maliciosa.	81.15%	(Rb & Kr, 2021)
Aprendizaje automático basado en reglas	Es un modelo de aprendizaje automático supervisado que se utiliza para generar reglas que se pueden utilizar para clasificar los datos. Se puede utilizar para generar	94.5%	(Islam et al., 2024)

Modelo y/o algoritmo	Descripción	Precisión	Referencia
	reglas que se pueden utilizar para identificar transacciones fraudulentas.		

Nota. En la tabla se presenta el nombre del modelo, la descripción, el porcentaje de precisión y la fuente de donde fue extraído.

De la Tabla 2, se ha identificado que los modelos con mayor precisión son Random Forest y Regresión Logística, los cuales serán tomados en cuenta para la realización del proyecto.

Para implementar un honeypot es bueno considerar diferentes factores como el objetivo, el uso y la complejidad. Una guía empieza por seleccionar o desarrollar los diferentes tipos de honeypots considerando cuál tipo de honeypot se ajuste mejor a las necesidades y objetivos de seguridad (Verdejo, 2018),

Como siguiente paso, está la configuración del honeypot puede ser manual o automatizada, dependiendo del tipo de honeypot y de las herramientas disponibles. Se determina la ubicación dentro de la red, aislándolo de la principal, para minimizar el riesgo de ser vulnerada. De esta manera comienza la monitorización y análisis del tráfico.

En el proceso de selección de modelos para entrenar el honeypot, se ha dado prioridad a dos enfoques destacados de inteligencia artificial: los árboles de decisión (Benedict, 2023): y los bosques aleatorios (Wang et al., 2022).

Ambos modelos han sido reconocidos por su eficacia en la detección y prevención de amenazas cibernéticas, lo que los convierte en elecciones fundamentales para fortalecer la seguridad.

Dentro del análisis de datos que se puede hacer a un dataset con el objetivo de identificar patrones se encuentra el análisis de anomalías, o también conocido como detección

de outliers, se centra en identificar puntos de datos que son inusuales o atípicos en comparación con el resto del conjunto de datos (Boukela et al., 2020).

Para recapitular, en el presente proyecto se desarrolla un sistema de detección de intrusiones en un entorno de transacciones en línea, implementando un honeypot de baja interacción usando los modelos y/o algoritmos de Regresión Logística y Random Forest para detectar las dos formas de fraude con tarjetas de crédito, en caso de pérdida y de clonación, mediante el análisis de anomalías enfocados en los hábitos de las transacciones de los usuarios.

Métricas de evaluación

La evaluación del rendimiento de un modelo es crucial para determinar su eficacia y confiabilidad, se emplea en el proceso de pruebas para identificar el estado del sistema, reajustarlo y poder probarlo nuevamente. Existen diversas métricas que se pueden emplear, el proyecto se centrará en tres métricas clave: Precisión (Precision), Exactitud (Accuracy) y Exhaustividad (Recall), a partir de una matriz de confusión (Yao & Shepperd, 2021), como se muestra en la Tabla 03.

Tabla 3

Matriz de confusión

		Predicción	
		Positivo	**Negativo**
Actual	**Positivo**	Verdaderos Positivos (TP)	Falsos Negativos (FN)
	Negativo	Falsos Positivos (FP)	Verdaderos Negativos (TN)

Nota. La tabla muestra los elementos de la matriz de confusión. Basado en (Yao & Shepperd, 2021).

Donde**:**

Verdaderos Positivos (TP): Las instancias correctamente clasificadas como positivas.

Verdaderos Negativos (TN): Las instancias correctamente clasificadas como negativas.

Falsos Positivos (FP): Las instancias incorrectamente clasificadas como positivas.

Falsos Negativos (FN): Las instancias incorrectamente clasificadas como negativas.

A continuación, en la Tabla 4 se presentan las fórmulas para el cálculo de las métricas clave para medir un sistema, el rango de resultados es de 0 a 1.

Tabla 4

Fórmulas para el cálculo de la precisión, exactitud y exhaustividad

Métrica	Definición	Rango
Precision	$\frac{TP}{TP + FP}$	[0,1]
Accuracy	$\frac{TP + TN}{N}$	[0,1]
Recall	$\frac{TP}{TP + FN}$	[0,1]

Nota. La tabla muestra las fórmulas para el cálculo de las métricas de evaluación, los términos son tomados de la matriz de confusión expuesta en la Tabla 3. Donde N es el número total de datos. Basado en (Yao & Shepperd, 2021).

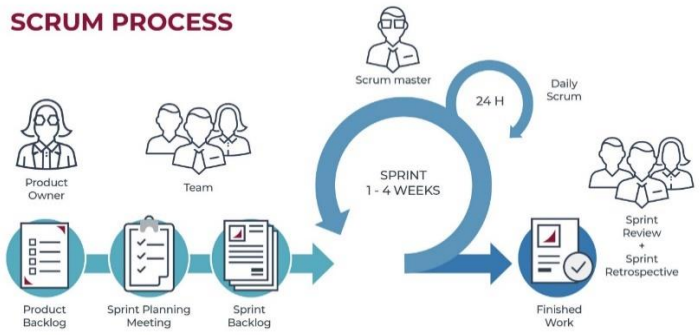
Metodología Ágil Scrum

Para el desarrollo de este sistema se empleará la metodología ágil SCRUM, debido a los posibles cambios al momento de analizar los ataques o fraudes de las tarjetas de crédito, y permitirá acelerar las actividades dentro del proyecto, mejor colaboración en equipo y comunicación diaria para la retroalimentación y solución de problemas (Hock, 2009).

En la Figura 2 se muestra un esquema sobre este marco de trabajo.

Figura 2

Marco de trabajo Scrum



Nota. Los elementos de scrum como eventos, artefactos y stakeholders. Tomado de (Hock, 2009).

Al analizar la Figura 2, se presenta a Scrum como un enfoque ágil para el desarrollo de proyectos que se centra en la flexibilidad y la adaptabilidad. En lugar de seguir un plan rígido desde el principio, Scrum permite un proceso iterativo e incremental, dividiendo el trabajo en períodos llamados Sprint. El proceso comienza con la creación de un Product Backlog, una lista priorizada de todas las características y tareas del proyecto.

En la reunión de Sprint Planning, el equipo selecciona las tareas más importantes para abordar durante el próximo Sprint. Durante el Sprint, que es una fase de desarrollo de corta duración, generalmente de dos a cuatro semanas, el equipo trabaja en las tareas seleccionadas del Sprint Backlog.

Cada día, el equipo tiene una breve reunión llamada Daily Scrum para mantenerse sincronizado. Al final del Sprint, el equipo presenta un Incremento Potencialmente Entregable, una versión mejorada del producto que puede ser liberada si es necesario. A continuación, tiene lugar la Revisión del Sprint, donde se muestra el trabajo completado y se recibe retroalimentación.

La Retrospectiva del Sprint sigue, proporcionando al equipo la oportunidad de reflexionar sobre el proceso e identificar mejoras para el próximo Sprint. Con esta retroalimentación, el Product Owner actualiza el Product Backlog y el ciclo comienza de nuevo.

Capítulo III

Desarrollo del Sistema

Implementación del Sistema

En las siguientes páginas, se detalla el procedimiento utilizado para la implementación del sistema de detección de intrusiones usando honeypots. El proceso inicia dividiendo al equipo en roles para conocer las responsabilidades de cada miembro, se construyen las Historias de Usuario (H.U.) y las tareas en el Product Backlog, del cual se analizan los Sprint. Se revela el contexto general del sistema representado en la arquitectura física y, además, se explica el patrón de diseño usado, ayudándose de una breve explicación de las herramientas para el desarrollo de cada parte del proyecto software. Al final, se detalla el proceso de los Sprint hasta su conclusión.

El sistema está diseñado para funcionar desde que el usuario selecciona un producto en la interfaz de la página de ventas y desea realizar el pago usando su tarjeta de crédito, al digitar los datos, estos se enrutan pasando por el honeypot al sistema de la entidad financiera, donde verifica los datos de la tarjeta, la disponibilidad de fondos y autoriza la transacción, al regresar al honeypot, mediante el modelo de Regresión Logística y Random Forest, detecta si es una transacción anómala por el análisis de patrones basados en la IP, fecha, hora y monto de la operación, se guarda en la base de datos con la clasificación correspondiente normal o fraudulenta, para terminar la transacción y efectuarse el pago, y el usuario recibe una alerta sobre el tipo de transacción realizada. De esta manera, se detecta el fraude al usar tarjetas de crédito robadas o clonadas y se monitorea las transacciones mediante los patrones de comportamiento y actividades anómalas sin comprometer la base de datos real de la entidad financiera.

Roles y Técnicas en Scrum

Roles

Al utilizar Scrum, se favorece la división de roles y la comprensión en las necesidades del usuario, en este contexto, asignar tareas específicas a cada miembro del proyecto es fundamental para el éxito del marco de trabajo ágil.

Scrum define tres roles principales, cada uno con responsabilidades y funciones específicas como el Product Owner, el Scrum Master y el Team Development (Schwaber & Sutherland, 2020).

En el presente proyecto se ha dividido de la siguiente manera, como es explicado en la Tabla 5.

Tabla 5

Designación de roles de Scrum

N°	Rol	Responsables	Descripción
01	Product Owner	Dr. José Luis Carrillo Medina	Representa los intereses del cliente y del negocio.
02	Scrum Master	Mary Elena Claudio Calvopiña	Es el líder y facilitando la colaboración entre todo el equipo.
03	Team Development	Jimmy Israel Guaján Perugachi Mary Elena Claudio Calvopiña	Desarrollo y diseño de la aplicación.

Nota. La tabla muestra la división de roles, la persona responsable y la descripción de la actividad principal que cumple.

Como se muestra en la Tabla 5, el proyecto se compone de dos integrantes principales, uno de ellos cumplirá con dos roles de desarrollador y scrum master, y para la comunicación con las partes interesadas se encuentra el tutor del trabajo.

Historias de Usuario

En scrum se pueden realizar las historias de usuario como una técnica importante para la captura y descripción de requisitos (Schwaber & Sutherland, 2020), recordadas por ser breves y concretas definiciones de las funcionalidades que desea el usuario para el sistema.

En la Tabla 6 se pueden encontrar las historias de usuarios que describen las funcionalidades que el sistema debe tener, además, se detalla la razón por la cual es necesaria o el resultado esperado al entregar el producto software.

Tabla 6

Historias de usuario

ID	Nombre	Rol	Característica/Funcionalidad	Razón/Resultado
1	H.U. 01	Como usuario	Quiero un sistema de transacciones en línea enfocados en el pago con tarjetas de crédito.	Un sistema enfocado en ventas de productos donde se visualice las páginas para el pago usando tarjetas de crédito.
2	H.U. 02	Como usuario	Quiero que el sistema detecte el fraude en tarjetas de crédito mediante la implementación de honeypots y modelos de machine learning	Usar los modelos con mayor precisión y un honeypot que detecte el fraude en los casos de pérdida o clonación de tarjetas de crédito e identifique los ataques.

ID	Nombre	Rol	Característica/Funcionalidad	Razón/Resultado
3	H.U. 03	Como usuario	Quiero que el sistema permita la monitorización de transacciones emitiendo una alerta al finalizar la transacción y genere un informe de los ataques monitoreados.	Que el sistema me avise el tipo de transacción como legítima o fraudulenta registrando la transacción.

Nota. La Tabla presenta las historias de usuario que relata las características/funcionalidades y el resultado por el cual se desarrolla.

Product Backlog y lista de tareas

En la Tabla 7, se encuentra el artefacto inicial del marco de trabajo Scrum, denominado Product Backlog (Schwaber & Sutherland, 2020).

Donde se realiza principalmente la estimación de tiempo, fecha inicial y fecha final para la solución de las tareas contenidas en los Sprint.

Tabla 7

Product Backlog del proyecto

Historia de usuario	Nombre	Estimación (días)	Fecha de inicio	Fecha de fin	N° de Sprint
1	H.U. 01	20	13/11/2023	08/12/2024	01
2	H.U. 02	18	11/12/2023	05/01/2024	02
3	H.U. 03	20	08/01/2024	02/02/2024	03

Nota. La tabla presenta la estimación del desarrollo del sistema, se complementa con la Tabla 9 y 10.

En el Producto Backlog, se definen las tareas a desarrollarse en cada uno de los Sprint según las necesidades y los cambios en las prioridades. Es una técnica clave para conseguir una gestión efectiva del trabajo, en la Tabla 8 se puede observar la lista de tareas.

Tabla 8

Lista de tareas del Product Backlog del proyecto

ID	Nombre	Característica/Funcionalidad	Razón/Resultado
1	L.T. 1	Implementar un sistema de transacciones en línea que acepte solamente pagos con tarjetas de crédito.	Un sistema con página de acceso, página de productos, sección de pago y verificación de tarjeta.
2	L.T. 2	Implementar un honeypot y modelos de machine learning para la detección del fraude en tarjetas de crédito y la monitorización de transacciones.	Honeypot y modelos de mayor precisión que detecte el fraude en pagos con tarjetas de crédito robadas y clonadas.
3	L.T. 3	Emisión de alerta al finalizar la transacción y generación de un informe de los ataques monitoreados.	Alerta e informe de ataques.

Nota. La Tabla tiene la lista de las tres tareas del Product Backlog, las cuales se complementan con la Tabla 7 y 9.

En la Tabla 9 se presenta la estimación del tiempo en días, la fecha de inicio, la fecha de finalización junto al número de Sprint al cual pertenece cada tarea mencionada en la Tabla 8, que parten de las historias de usuario especificadas.

Tabla 9

Estimación de las tareas del Product Backlog

Lista de Tareas	Nombre	Estimación (días)	Fecha de inicio	Fecha de fin	N° de Sprint
1	L.T. 01	20	13/11/2023	08/12/2023	01
2	L.T. 02	18	11/12/2023	05/01/2024	02
3	L.T. 03	20	08/01/2024	02/02/2024	03

Nota. Se detalla los plazos para la entrega de las tareas a implementarse en los Sprint.

Con las tablas, se tiene en claro las tareas y el tiempo de estimación para poder desarrollar con mejor organización el proyecto. Se han obtenido tres historias de usuario, divididas en tres Sprint explicados en el Product Backlog, con un período de cuatro semanas, para cada característica del sistema.

Arquitectura del sistema

En el apartado anterior, se han establecido la lista de tareas del proyecto, sin embargo, es importante conocer de manera general cómo se compondrá el sistema, por esta razón, a continuación, se expone sobre la arquitectura que ayudará al correcto desarrollo y cumplimiento del propósito.

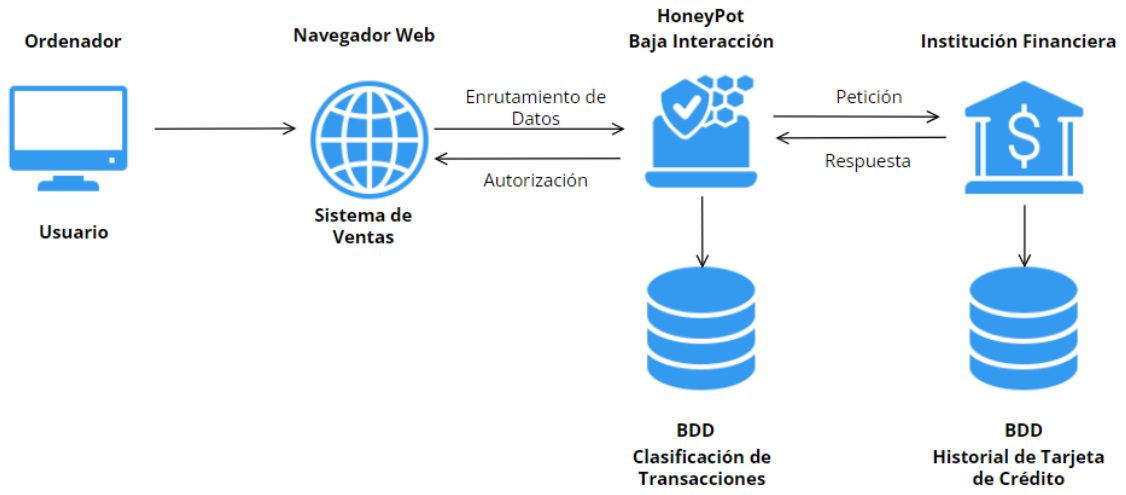
La arquitectura es la estructura fundamental de cómo se organiza el sistema software, al analizarla, proporciona una visión global que determina la interacción de los diferentes componentes y cómo estos trabajan juntos para conseguir los objetivos iniciales (Majeed & Rauf, 2018).

La ventaja principal es establecer las bases para el desarrollo, evolución y mantenimiento de la aplicación.

En la Figura 3 se presenta el diagrama de la estructura física del sistema. Mas adelante se habla sobre la arquitectura en capas que ha sido implementada.

Figura 3

Arquitectura física del sistema



Nota. Se describe el flujo de interacción entre los elementos de la estructura física.

De la Figura 3, se describe que la arquitectura permite al usuario acceder al sistema desde un ordenador, ingresa a la página de ventas en un navegador, al seleccionar el pago para el producto elegido y digitar los datos, estos se enrutan hacia la entidad financiera, donde verifica con el historial de los datos de la tarjeta ingresada y autoriza el pago si al analizar los datos corresponden al usuario y existe fondos o crédito en la cuenta, la respuesta regresa al honeypot, donde recoge el monto, IP, puerto usado y ataque que, junto con el modelo, detecta si es una transacción anómala basada en las reglas y umbrales como si excede el monto, es fuera de las horas habituales, es una ubicación desconocida o una IP ajena, y clasifica la transacción guardando en la base de datos como legítima o anómala, y se efectúa el pago, como es un sistema de detección, únicamente emite una alerta del tipo de transacción, mas no ayuda a prevenir este tipo de fraudes, sin embargo, al registrar el ataque se tiene suficiente información para un sistema de prevención.

El sistema está construido en capas, donde se organiza el sistema en niveles, y cada uno tiene una responsabilidad específica, ofrece servicios a las capas superiores y se

comunican con las capas adyacentes, de esta manera promueve la modularidad, la separación de responsabilidades y la claridad del software siguiendo un flujo bien estructurado (Quiñones, 2021).

Hardware

Para el desarrollo del proyecto se ha utilizado un ordenador con las siguientes características. Un procesador AMD Ryzen 3 3200U de 2.60 GHz, con una memoria RAM de 16.0 GB y el sistema operativo Windows 11 Home.

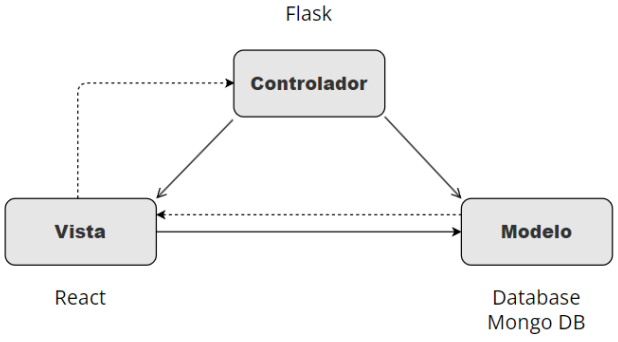
Diseño y herramientas del sistema

La arquitectura en capas del sistema está representada por el patrón arquitectónico Modelo, Vista y Controlador, su primera capa está relacionada con la infraestructura de datos y las operaciones de un sistema de transacciones online, la segunda capa se utiliza para implementar la interfaz de usuario y la tercera está enfocada en la lógica empresarial (Majeed & Rauf, 2018).

En la Figura 4, se representa el patrón arquitectónico con las herramientas utilizadas para el cumplimiento de los requisitos y el desarrollo del proyecto.

Figura 4

Arquitectura Modelo, Vista y Controlador con las herramientas



Nota. El proyecto está diseñado con el patrón arquitectónico MVC, y en la Figura se detallan las herramientas que se usan.

De la Figura 4, se tiene a continuación, la especificación del contenido de cada una de las capas con la definición de las tecnologías usadas.

Modelo

La base de datos está creada con Mongo DB, la cual almacena los datos transaccionales de la entidad financiera. Además, en este nivel se encuentran contenidas las funciones del negocio realizadas con Python con su Framework Flask. La responsabilidad de este nivel es gestionar las transacciones, encargándose de realizar las operaciones, como, por ejemplo, restar el monto correspondiente de la cuenta de la entidad financiera conforme a la transacción realizada, este proceso se relaciona con otras actividades como los cálculos resultantes de una adquisición múltiple de productos, además de las validaciones del pago y las actualizaciones de datos en el historial de las transacciones.

Mongo DB (versión 7.0.3). Es un sistema de gestión de bases de datos NoSQL (Not Only SQL) que se utiliza para almacenar y gestionar datos de manera flexible y escalable. Está diseñado y recomendado para manejar grandes volúmenes de datos y proporcionar un rendimiento eficiente en entornos distribuidos (Pérez Román, 2020).

Vista

La vista representa la interfaz de usuario desarrollada con React, donde se encuentra el login, la lista de compras y la ventana de pagos con la tarjeta de crédito. Además del historial de transacciones con la tarjeta de crédito para el usuario con permisos.

React (versión 18.2.0). Es un Framework de JavaScript, se utiliza para construir interfaces de usuario interactivas, centrándose en la creación eficiente de componentes reutilizables y proporciona un enfoque declarativo para definir cómo debe verse la interfaz de usuario en función del estado de la aplicación (Kovtun, 2018).

Controlador

Manejan las solicitudes de los usuarios para la coordinación entre la Vista y el Modelo, permitiendo guardar en la base de datos correspondiente una vez que se ocupen los servicios del honeypot que, en caso de ser una transacción anómala, se usan controladores falsos. Además, al tener el resultado de la transacción sea anómala o legítima, emite la alerta de detección. La tecnología principal usada es Flask.

Flask (versión 3.0.1). Es un Framework de Python, que proporciona las herramientas esenciales para construir aplicaciones web de manera rápida y eficiente. Su simplicidad y flexibilidad permiten a los desarrolladores organizar su código de manera libre y elegir las bibliotecas adicionales según sus necesidades (Kovtun, 2018).

Sprint 1: Implementación de un sistema de transacciones en línea

En las siguientes líneas, se detalla el proceso de los tres Sprint para el desarrollo del producto software, en otras palabras, es el período de tiempo fijo durante el cual se desarrollan una lista de tareas para obtener un incremento entregable del producto (Schwaber & Sutherland, 2020).

El primer Sprint trata sobre el desarrollo del sistema de transacciones online donde se representará el proceso de una compra y el ingreso de datos de las tarjetas de crédito.

Historia de usuario detallada

La historia de usuario H.U. 01 de la Tabla 6, junto a la tarea L.T. 01 de la Tabla 8, se refieren a la implementación del sistema de transacciones online, el cual se explica con mayor detalle en la Tabla 10, donde se encuentra la historia de usuario del Sprint 1 con los criterios de aceptación.

Tabla 10

Historia de usuario para la implementación del sistema online.

Historia de Usuario			
Nombre: HU001		**Usuario:** Usuario de internet	
Nombre de historia:	Sistema de transacciones en línea.		
Prioridad del Negocio:	Alta	**Riesgo en desarrollo**:	Alta
Puntos estimados (días):	20	**Interacción asignada:**	1
Programadores Responsables:	Jimmy Guaján, Mary Claudio.		
Descripción: Como usuario quiero un sistema de transacciones en línea que acepte solamente pagos con tarjetas de crédito.			
Validación (Criterios de aceptación): • El sistema debe permitir a los usuarios realizar transacciones en línea con éxito. • El sistema debe aceptar tarjetas de crédito. • El sistema debe validar la información de la tarjeta de crédito, incluyendo el número de tarjeta, fecha de vencimiento y código de seguridad (CVV). • En caso de errores en la información de la tarjeta, el sistema debe mostrar mensajes de error claros.			

Nota. La tabla muestra los detalles de la Historia de Usuario HU001 con información relevante como la prioridad, el riesgo de desarrollo, los días estimados para desarrollarse, la interacción asignada, el responsable y así mismo una descripción y criterios de aceptación para la historia de Usuario.

Sprint backlog 01

El Sprint Backlog 01 detalla las actividades realizadas durante el primer Sprint, la persona responsable de su ejecución, los datos de planificación para la ejecución del Sprint, las estimaciones de tiempo en horas para cada tarea y el estado actual de cada tarea, expresado en la Tabla 11.

Tabla 11

Sprint Backlog 01

Sprint 01						
Fecha Inicio		**Fecha Fin**			**Jornada**	
13/11/2023		08/12/2023			8 horas diarias	
H.U.	**Tareas**	**Horas**	**Fecha Inicio**	**Fecha Fin**	**Responsables**	**Estado**
01	Investigación inicial.	8	13/11/2023	13/11/2023	Jimmy Guaján y Mary Claudio	Finalizada
02	Configuración inicial de las bibliotecas de React y Flask.	8	14/11/2023	14/11/2023	Jimmy Guaján y Mary Claudio	Finalizada
03	Desarrollo de mockups del sistema de transacciones online y la entidad financiera.	4	15/11/2023	15/11/2023	Jimmy Guaján y Mary Claudio	Finalizada
04	Elección y construcción de la	4	15/11/2023	15/11/2023	Jimmy Guaján y Mary Claudio	Finalizada

Sprint 01						
	base de datos (MongoDB).					
05	Desarrollo del login del sistema con seguridades.	24	16/11/2023	20/11/2023	Jimmy Guaján y Mary Claudio	Finalizada
06	Desarrollo del módulo de compras.	24	21/11/2023	23/11/2023	Jimmy Guaján y Mary Claudio	Finalizada
	Desarrollo del módulo de pagos.	24	24/11/2023	28/11/2023		
07	Verificación de la tarjeta durante el proceso de pago.	16	29/11/2023	30/11/2023	Jimmy Guaján y Mary Claudio	Finalizada
08	Definición de rutas de gestión de usuarios para las solicitudes y construcción del backend del módulo.	8	01/12/2023	01/12/2023	Jimmy Guaján y Mary Claudio	Finalizada
09	Definición de rutas de gestión de transacciones construcción del	8	04/12/2023	04/12/2023	Jimmy Guaján y Mary Claudio	Finalizada

Sprint 01						
	backend del módulo.					
10	Definición de rutas de compra y construcción del backend del módulo.	8	05/12/2023	05/12/2023	Jimmy Guaján y Mary Claudio	Finalizada
11	Configuración de ambientes de desarrollo, prueba y producción.	8	06/12/2023	06/12/2023	Jimmy Guaján y Mary Claudio	Finalizada
12	Manejo de errores en el backend.	16	07/12/2023	08/12/2023	Jimmy Guaján y Mary Claudio	Finalizada

Nota. En la tabla se encuentran descritas las tareas que corresponden al Sprint Backlog 01, definiendo el número de horas estimadas, la fecha de inicio y fin, los responsables y el estado al momento de la entrega del proyecto.

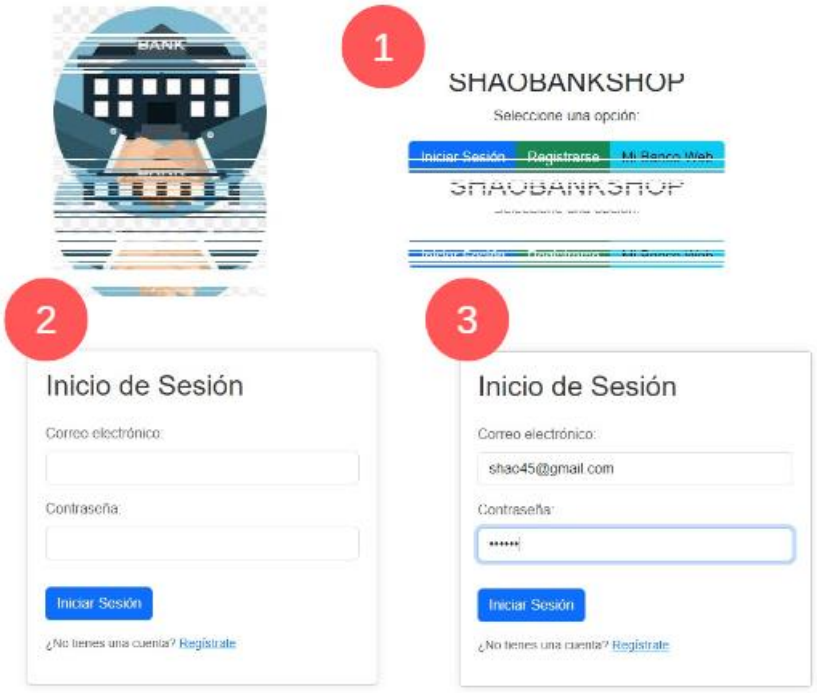
Resultados del Sprint 1

Del Sprint 1 se obtuvo el sistema de transacciones online, primero, para el usuario, se presentan las páginas de Login, ventas, formulario de pago y de la transacción completada. Por el lado del backend, se encuentra la definición de rutas y operaciones. De esta manera, el sistema ha conseguido procesar una transacción, sin embargo, como aún no está implementado el honeypot ni el modelo, no detecta si es anómala o legítima.

El proceso de la transacción inicia en el navegador, ingresando al sistema desde el Login, con el correo y la contraseña, representados en la Figura 5.

Figura 5

Login del sistema de transacciones online.

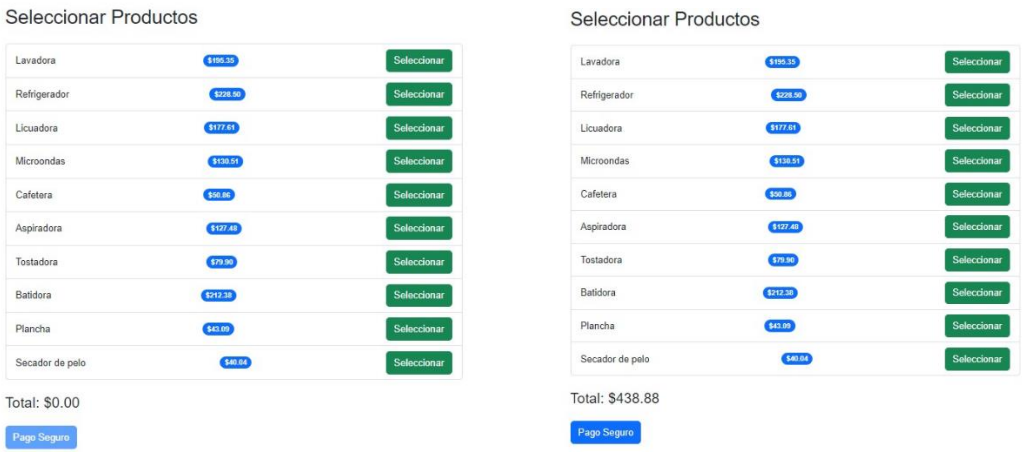


Nota. La figura muestra en el paso uno el sistema, en el paso dos el inicio de sesión, y en el paso tres el ingreso de datos.

Al ingresar, se puede observar la página de ventas, donde el usuario selecciona el o los productos, y al final de la página, puede encontrar el valor a cancelar, para proceder con la operación puede dar clic en el botón *pago seguro,* como se observa en la Figura 6.

Figura 6

Página de ventas del sistema de transacciones online

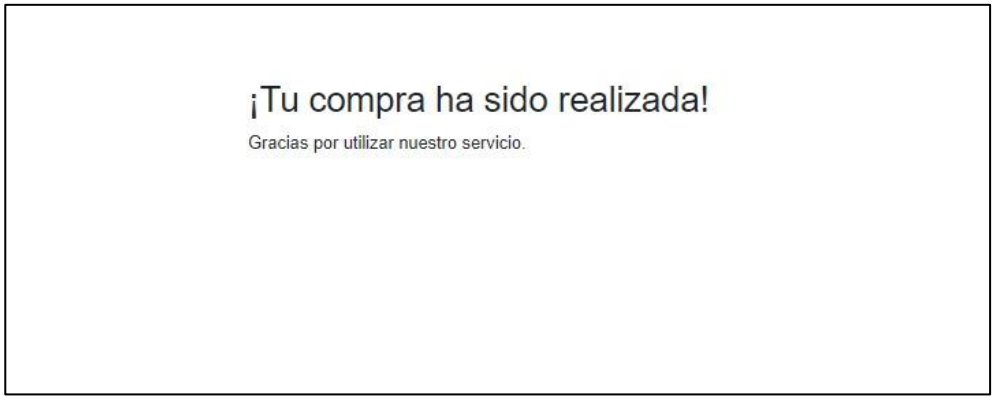


Nota. En la figura a la izquierda se muestra la sección de productos, y en la derecha el monto a pagar.

El usuario ingresa la información de su tarjeta de crédito para poder adquirir el o los productos, y selecciona el botón *pagar*, se enviarán los datos a la entidad financiera, se espera la validación y autorización, una vez que se ha recibido la respuesta, el pago se completará con éxito, como se observa en la Figura 7.

Figura 7

Página de pagos del sistema de transacciones online.



Nota. En la figura se muestra el paso exitoso de la transacción.

De esta manera, el sistema ha cumplido satisfactoriamente con los criterios de aceptación, puesto que permite realizar transacciones en línea con éxito, acepta las tarjetas de crédito, valida su información, incluyendo el número de la tarjeta, la fecha de vencimiento y el código de seguridad (CVV).

Sprint 2: Honeypot y modelos de machine learning

A continuación, se redactan los pasos de la creación del dataset, la implementación del honeypot, y la elección y aplicación de los modelos de machine learning con los cuales se puede detectar el fraude en las tarjetas de crédito robadas y clonadas.

Historia de usuario detalladas

La historia de usuario H.U. 02 de la Tabla 6 y tarea L.T. 02 de la Tabla 8, hacen referencia a la implementación de un honeypot con modelos de machine learning en el sistema de transacciones online, con el objetivo de cumplir con la detección de fraude en las tarjetas de crédito. Y para mayor explicación, en la Tabla 12, se encuentra detallada la historia de usuario del Sprint 2 con los criterios de aceptación correspondientes.

Tabla 12

Historia de usuario para el honeypot y modelos de ML

Historia de Usuario			
Nombre: HU002		**Usuario:** Usuario de internet	
Nombre de historia:	Honeypots y modelos de machine learning		
Prioridad del Negocio:	Alta	**Riesgo en desarrollo:**	Alta
Puntos estimados (días):	18	**Interacción asignada:**	2
Programadores Responsables:	Jimmy Guaján, Mary Claudio.		

Historia de Usuario
Descripción: Implementar un honeypot y modelos de machine learning para la detección del fraude en tarjetas de crédito y la monitorización de transacciones.
Validación (Criterios de aceptación): • Se han identificado y seleccionado modelos de machine learning adecuados para la detección de fraudes en tarjetas de crédito. • Los modelos de machine learning y el honeypot han sido integrados con éxito en el sistema de transacciones existente. • El sistema detecta y registra de manera efectiva las transacciones anómalas y fraudulentas. • El sistema monitoriza las transacciones de manera efectiva las transacciones anómalas y fraudulentas.

Nota. La tabla muestra los detalles de la Historia de Usuario HU002 con información relevante como la prioridad, el riesgo de desarrollo, los días estimados para desarrollarse, la interacción asignada, el responsable y una descripción, además de los criterios de aceptación para la historia de usuario.

Sprint backlog 02

El Sprint Backlog 02 muestra las tareas realizadas durante el segundo Sprint, las cuales se encuentran detalladas en la Tabla 13, con el número de horas, fecha de inicio y fin, los responsables y el estado.

Tabla 13

Sprint Backlog 02

Sprint 02						
Fecha Inicio		**Fecha Fin**			**Jornada**	
11/12/2023		05/01/2024			8 horas diarias	
H.U.	**Tareas**	**Horas**	**Fecha Inicio**	**Fecha Fin**	**Responsables**	**Estado**
01	Extracción de características para la simulación de las tarjetas de crédito y el historial.	20	11/12/2023	13/12/2023	Jimmy Guaján y Mary Claudio	Finalizada
02	Instalación del entorno de trabajo (librerías pandas, numpy, random).	8	13/12/2023	14/12/2023	Jimmy Guaján y Mary Claudio	Finalizada
03	Definición de la cantidad de tarjetas de crédito a simular.	2	14/12/2023	14/12/2023	Jimmy Guaján y Mary Claudio	Finalizada
04	Definición de la cantidad transacciones a simular para las tarjetas de crédito seleccionadas.	2	14/12/2023	14/12/2023	Jimmy Guaján y Mary Claudio	Finalizada

			Sprint 02			
05	Simulación de datos para tarjetas de crédito.	16	15/12/2023	18/12/2023	Jimmy Guaján y Mary Claudio	Finalizada
06	Simulación de datos para el historial de tarjetas de crédito.	16	19/12/2023	20/12/2023	Jimmy Guaján y Mary Claudio	Finalizada
07	Crear un dataframe para transacciones legítimas y anómalas de las tarjetas de crédito.	4	21/12/2023	21/12/2023	Jimmy Guaján y Mary Claudio	Finalizada
08	Almacenar los datos en un archivo .csv	4	21/12/2023	21/12/2023	Jimmy Guaján y Mary Claudio	Finalizada
09	Configuración del honeypot.	16	22/12/2023	26/12/2023	Jimmy Guaján y Mary Claudio	Finalizada
10	Revisión y elección del modelo.	8	27/12/2023	27/12/2023	Jimmy Guaján y Mary Claudio	Finalizada
11	Inicialización del modelo.	1	28/12/2023	28/12/2023	Jimmy Guaján y Mary Claudio	Finalizada
12	Entrenamiento del modelo	15	28/12/2023	29/12/2023	Jimmy Guaján y Mary Claudio	Finalizada
13	Análisis de características y	8	02/01/2024	02/01/2024	Jimmy Guaján y Mary Claudio	Finalizada

Sprint 02						
	detección de anomalías.					
14	Configuración de ambientes de desarrollo, prueba y producción.	8	03/01/2024	03/01/2024	Jimmy Guaján y Mary Claudio	Finalizada
15	Manejo de errores.	16	04/01/2024	05/01/2024	Jimmy Guaján y Mary Claudio	Finalizada

Nota. En la Tabla se encuentran descritas las tareas que corresponden al Sprint Backlog 02, definiendo el número de horas estimadas, la fecha de inicio y fin, los responsables y el estado al momento de la entrega del proyecto.

Resultados del Sprint 2

Del Sprint 02 se obtuvo el archivo .CSV, con las características de las tarjetas de crédito y el historial, los cuales serán explicados, a detalle, más adelante. Además, se consiguió la implementación del honeypot y del modelo en el sistema, cumpliendo así con la funcionalidad de la detección de transacciones fraudulentas.

Como primer paso, se estableció una búsqueda de la información sobre las características necesarias para la construcción del dataset. Con el logro alcanzado, se habilitó el entorno con la instalación y configuración de las herramientas y librerías. En este último, se consideraron: Pandas, NumPy y Random, las cuales son necesarias para los procesos matemáticos que se incluyen en el algoritmo con el cual se generan los datos aleatorios del dataset de tarjetas de crédito e historial.

Para el dataset de tarjetas de crédito, se han implementado los campos representados por el nombre, el número de la tarjeta de crédito, la fecha de expiración y el CVV, los cuales se

encuentran contemplados en la información hallada en el documento denominado Algoritmos de aprendizaje automático para detección de fraudes con tarjetas de crédito: Análisis y comparativa de (Calvo Pérez, 2021).

El archivo .CSV, que representa el historial, contiene 13 columnas, los campos contemplados son monto, ubicación, frecuencia de uso, horario, tipo de tarjeta, día de la semana, saldo disponible, IP, número de tarjeta, estado de tarjeta, titular de tarjeta, fecha de vencimiento y el Card Verification Value (CVV). Estas características han sido seleccionadas y analizadas con base a las revisiones que se han desarrollado (Calvo Pérez, 2021), y con los datos se procede a entrenar a los modelos.

En la simulación del número de tarjetas de crédito se han completado 10,000 (diez mil) elementos, para el estudio se han seleccionado 50, para cada una de las tarjetas de crédito se realizan 200 transacciones legítimas, y 300 anómalas. Dando un total de 25,000 (veinticinco mil) transacciones, con las cuales se ha construido el archivo .CSV.

Una vez que se han desarrollado los datos, estos se convierten en un dataframe, donde las claves son los nombres de las columnas y las listas son los datos seleccionados. Finalmente, de estos datos se crea el archivo .CSV para el entrenamiento del modelo.

El sistema se levantó en el puerto 5,000, y es usado por el honeypot para la detección del ataque, y se ha establecido el buffer de 1024 kb/s, en otras palabras, es el tamaño asignado para el honeypot para que pueda hacer su trabajo.

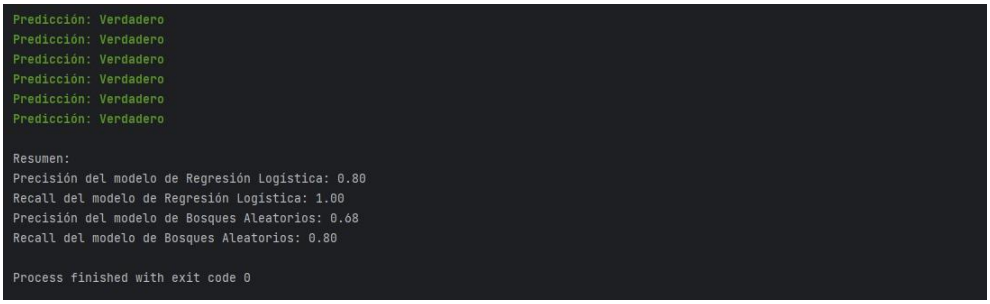
Se inicializó el modelo llamando a las librerías para random forest y regresión logística, usadas para la detección de las anomalías, son los algoritmos elegidos por el análisis de la Tabla 02 del Capítulo II.

Se seleccionan los datos aleatoriamente, 80% para la validación, el restante se convierte en mi 100% del cual se ha seleccionado el 75% para el entrenamiento y el 25% para las pruebas. Se entregan directo a los modelos para ser entrenados y probados.

Como resultados en cuanto a *precision* y *recall*, en regresión logística se obtuvo un 0.80 y 1.00, respectivamente, mientras que, en bosques aleatorios 0.68 y 0.80, teniendo un resultado más favorable para regresión logística, expuestos en la Figura 8.

Figura 8

Resultados del Modelo



Nota. En la figura se muestran algunos datos, se complementan en el siguiente capítulo.

Sin embargo, después de una revisión, se identificó que los modelos no están reconociendo la característica del lugar. Se corrigió y analizó la característica faltante. Se empleó un reajuste para mejorar los resultados, el cual consistió en aumentar los datos de 25,000 (veinticinco mil) a 40,000 (cuarenta mil).

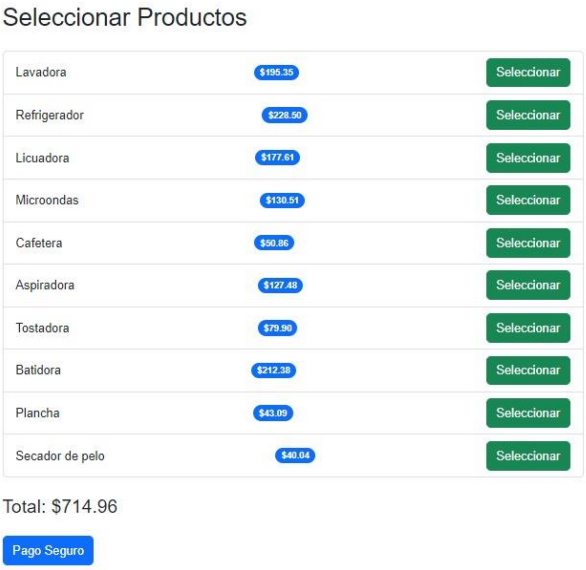
Se definieron las funciones y umbrales para el monto, la IP, ubicación y horario: si excede el monto común, es fuera de las horas habituales, es una ubicación desconocida o una IP ajena. Al cumplir tres de los cuatro umbrales, se clasifica la transacción, se guarda en la base de datos como anómala, o si es buena y no incumple, se detecta como legítima.

Mediante los datos se ha obtenido un sistema de detección de intrusiones que identifica el fraude de tarjetas de crédito robadas o clonadas en una transacción en línea.

Se sigue el mismo proceso de los resultados del Sprint 02 para el proceso de la transacción, en este caso, con una de tipo anómalo por el monto y el horario que cumplen con los umbrales, superando el valor de $500, como se visualiza en la Figura 9.

Figura 9

Transacción anómala



Nota. En la figura se presenta el monto mayor a 500 dólares, relacionado al umbral.

En la Figura 10, se presenta el resultado de la transacción, sin implementar aun la alerta de una operación anómala.

Figura 10

Resultado de una transacción

Nota. La figura representa el mensaje de una transacción finalizada.

De esta manera, el sistema cumple con los criterios de aceptación porque se han identificado y seleccionado dos modelos de machine learning denominados regresión logística y bosques aleatorios, los cuales han resultado adecuados para la detección de fraudes en tarjetas de crédito que, junto al honeypot, han sido integrados con éxito en el sistema de transacciones, detecta y registra de manera efectiva las transacciones anómalas e ilegítimas, y las monitoriza guardando en una base de datos.

Sprint 3: Alerta y generación del informe de los ataques

A continuación, se presenta la implementación de una alerta al finalizar la transacción junto con la generación de un informe sobre los ataques recibidos como manera visual para poder monitorizar las transacciones legítimas o anómalas.

Historia de usuario detalladas

La siguiente historia de usuario representa la H.U. 03 de la Tabla 06 y la tarea L.T. 03 de la Tabla 08 del Product Backlog. Hace referencia a la implementación de una alerta que, detecta el ataque, además de incluir un informe para la monitorización de las transacciones. En la Tabla 14 se describe la historia de usuario del Sprint 03, con los criterios de aceptación.

Tabla 14

Historia de usuario para las alertas y el informe de ataques.

Historia de Usuario			
Nombre: HU003		**Usuario**: Usuario de internet	
Nombre de historia:	Alerta e informe de ataques.		
Prioridad del Negocio:	Alta	Riesgo en desarrollo:	Alta
Puntos estimados (días):	20	Interacción asignada:	2
Programadores Responsables:	Jimmy Guaján, Mary Claudio.		
Descripción:			

Historia de Usuario
Como usuario quiero que el sistema emita una alerta al finalizar la transacción con el tipo de operación sea anómala o legítima, además, debe generar un informe de los ataques para su monitorización.
Validación (Criterios de aceptación): • La funcionalidad debe generar automáticamente una alerta al finalizar cada transacción. • La alerta debe incluir información sobre si la operación se considera anómala o legítima. • Los informes generados deben almacenarse en un lugar seguro y accesible para la monitorización y análisis posteriores.

Nota. La tabla muestra los detalles de la Historia de Usuario HU003 del Sprint 03, con la información relevante como la prioridad, el riesgo de desarrollo, los días estimados para desarrollarse, la interacción asignada, el responsable y así mismo una descripción y criterios de aceptación para la historia de usuario.

Sprint backlog 03

El Sprint Backlog 03 detalla las actividades realizadas durante el tercer Sprint, representados en la Tabla 15.

Tabla 15

Sprint Backlog 03

Sprint 03		
Fecha Inicio	**Fecha Fin**	**Jornada**
08/01/2024	02/02/2024	8 horas diarias

Sprint 03						
H.U.	**Tareas**	**Horas**	**Fecha Inicio**	**Fecha Fin**	**Responsables**	**Estado**
01	Análisis de umbral del honeypot para la emisión de la alerta	24	08/01/2024	10/01/2024	Jimmy Guaján y Mary Claudio	Finalizada
02	Codificación de la lógica para generar alertas automáticamente al finalizar cada transacción.	24	11/01/2024	15/01/2024	Jimmy Guaján y Mary Claudio	Finalizada
03	Integrar las alertas con el sistema de notificación correspondiente: monto, horario, IP y ubicación.	32	16/01/2024	19/01/2024	Jimmy Guaján y Mary Claudio	Finalizada
04	Generar la función de informe sobre transacciones anómalas.	8	22/01/2024	22/01/2024	Jimmy Guaján y Mary Claudio	Finalizada
05	Incluir la información clave en los informes, como el	16	23/01/2024	24/01/2024		

Sprint 03				
	ataque, monto, IP, puerto, horario y tipo.			
06	Configuración de ambientes de desarrollo, prueba y producción.	24	25/01/2024	29/01/2024
07	Manejo de errores.	32	30/01/2024	02/02/2024

Nota. En la Tabla se encuentran descritas las tareas que corresponden al Sprint Backlog 03, definiendo el número de horas estimadas, la fecha de inicio y fin, los responsables y el estado al momento de la entrega del proyecto.

Resultados del Sprint 3

Del Sprint 03 se obtuvo el sistema con la implementación de la alerta, además de un informe de ataques que se detallan en los siguientes párrafos.

Con relación al primer punto, se inició el análisis de los umbrales que serán presentados en las alertas en diversos escenarios. Este análisis se centró en la evaluación de los campos de monto, IP, horario y ubicación. Dichas alertas fueron categorizadas en dos: transacciones legítimas y anómalas. Las transacciones anómalas se identificaron al superar tres de los cuatro umbrales específicos explicados en la Tabla 16.

Tabla 16

Alertas.

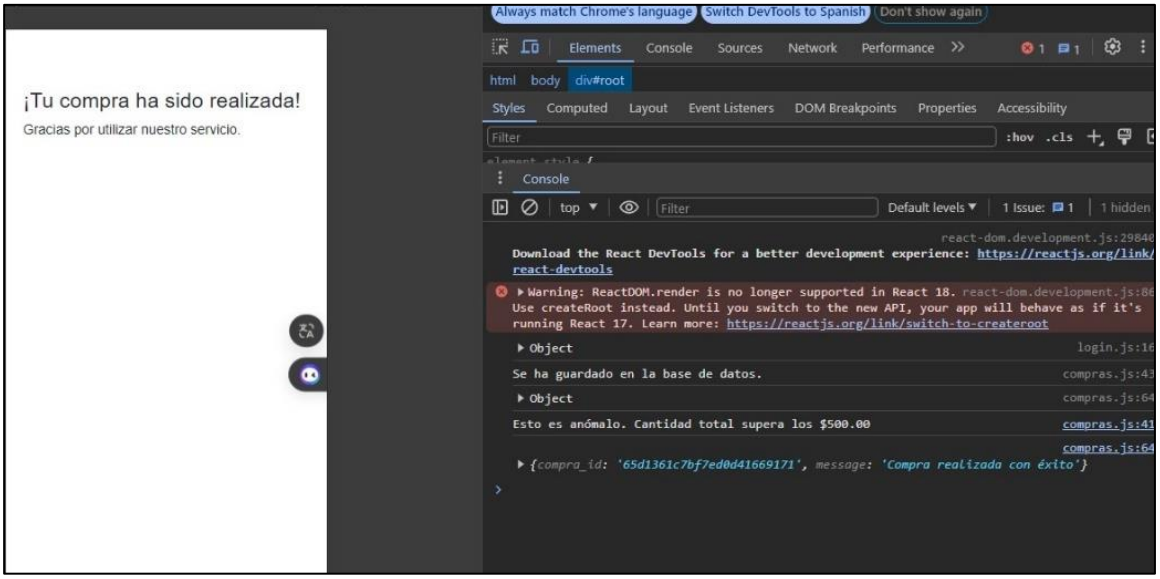
N.	Umbral	Estado	Mensaje
1	Monto superior al establecido (500)	Cumple	Esto es anómalo. Cantidad total supera los $500.
2	Dirección IP diferente a la ecuatoriana	Cumple	Esto es anómalo. IP no recurrente.
3	Horario comprendido entre las 11 pm y 6 am.	Cumple	Esto es anómalo. Horario fuera del recurrente.
4	Ubicación registrada en una ciudad distinta a la(s) habitual(es).	Cumple	Esto es anómalo. Ubicación fuera de la recurrente.

Nota. En la tabla se muestran los umbrales que debe cumplir la transacción.

Como resultados, se obtiene el tipo de transacción, y si es anómala, presenta el mensaje del principal umbral que ha sido superado, evidenciado en la Figura 11.

Figura 11

Evidencia de alertas.



Nota. La figura muestra en el lado izquierdo el mensaje al finalizar la transacción, y a la derecha la alerta dependiendo del umbral cumplido.

Para la generación de un informe, se toman los registros de la base de datos donde se han guardado las transacciones anómalas y legítimas (normales), mantiene el tipo de ataque que puede ser clonado o robado, el monto, la IP, el puerto, el horario y finalmente, el tipo, explicado en la Figura 12.

Figura 12

Generación de informe del dataset con los nuevos ataques.

Ataque	Monto	IP	Puerto	Horario	Tipo	Lugar
tarjeta clonada	603.18	155.59.48.218	5000	12:15	anomalo	Quito
reportada como robada	284.1	204.7.114.107	5000	2:21	anomalo	Latacunga
tarjeta clonada	674.71	192.168.66.230	5000	18:25	anomalo	Guayaquil
tarjeta clonada	537.13	36.226.92.134	5000	1:14	anomalo	Ambato
tarjeta clonada	364.58	192.168.29.142	5000	15:40	normal	Quito
tarjeta clonada	680.24	93.164.120.118	5000	13:4	anomalo	Latacunga

Nota. La figura muestra el registro de los ataques.

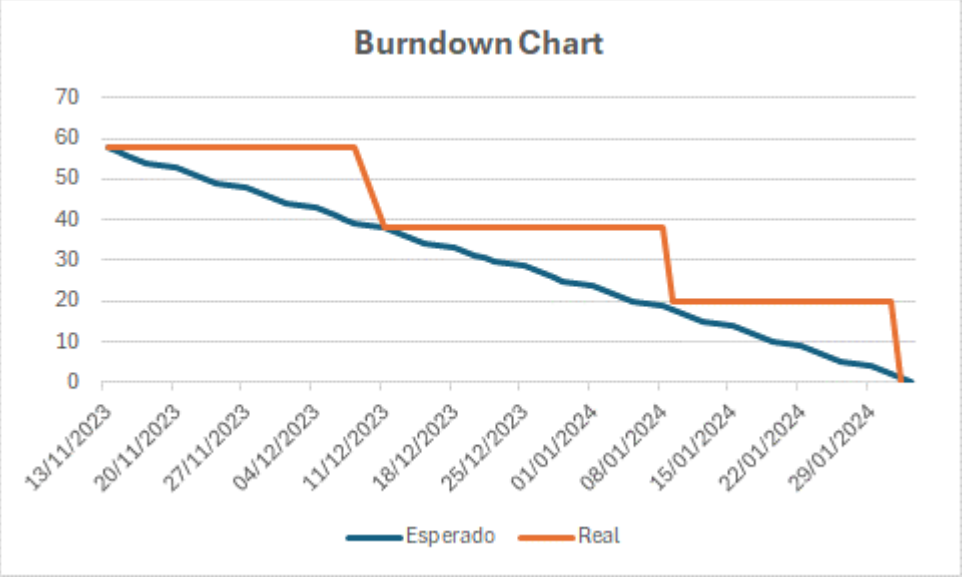
Este informe utiliza los registros de transacciones tanto anómalas como legítimas para analizar y comprender los patrones de actividad. Al mantener la información sobre el tipo de ataque (ya sea clonado o robado), el monto de la transacción, la dirección IP, el puerto utilizado, el horario de la transacción y, por último, el tipo de transacción (anómala o legítima), se obtiene una perspectiva completa para la toma de decisiones informadas. Este análisis puede ayudar a identificar posibles vulnerabilidades, patrones de comportamiento sospechosos y fortalecer las medidas de seguridad para prevenir futuros incidentes.

Burndown Chart

Se presenta el gráfico Burndown Chart de los tres Sprints, ilustrando la estimación de esfuerzo frente al avance real que se ha conseguido a lo largo del proyecto. En el eje X del gráfico, se representan las fechas correspondientes a los días planificados, abarcando desde el 13 de noviembre de 2023 hasta el 02 de febrero de 2024, conforme a lo establecido en la Tabla 07 de planificación. Por otro lado, el eje Y representa el número de puntos estimados para el esfuerzo que toma en completarlos, que, al sumar todos los establecidos en las historias de usuario detalladas al inicio de cada Sprint, se obtuvo un total de 58 puntos, entonces, una vez que el equipo de desarrollo termina una historia de usuario, se coloca en el gráfico, si no ha sido concluida, no se debe tomar en cuenta aún. Cuando se ha finalizado todas las historias de usuario, es como se consigue llegar a cero.

Figura 13

Burndown Chart del proyecto



Nota. El eje X contiene las fechas, en el eje Y los puntos de estimación de esfuerzo, el trazo azul representa la planificación y la de color naranja el progreso real.

La Figura 14, representa el progreso alcanzado en los tres Sprint con tres historias de usuario, la línea azul es la estimación que se calcula al dividir los 58 puntos para los 58 días planificados, mientras que, el trazo de color naranja es el progreso real que ha tenido el equipo de desarrollo en el proyecto, cada historia de usuario se ha cumplido en el plazo establecido. Únicamente, cuando ha sido presentada y aprobada, se registra la fecha en el BurnDown Chart. Culminando con éxito el desarrollo.

Capítulo IV

Validación Del Sistema

Al no encontrar una herramienta para la simulación de ataques para el honeypot de tarjetas de crédito, se aplicó validación cruzada, utilizando las funciones cross_val_score() y train_test_split() para evaluar el rendimiento de los modelos de aprendizaje automático de Random Forest y Regresión Logística. Esta técnica dividió los datos en cinco pliegues diferentes, en cada ocasión fue 80% para entrenamiento, 10% para pruebas y 10% para validación.

Resultados

Los resultados obtenidos aplicando validación cruzada durante las cinco iteraciones para el modelo de Random Forest, mostraron los siguientes números presentados en la Tabla 17, con un promedio de 0.999968.

Tabla 17

Acurracy de las iteraciones de Random Forest

Iteraciones	Acurracy
1	0,999841
2	1
3	1
4	1
5	1
Promedio	0,999968

Nota. La tabla presenta las cinco iteraciones en la columna izquierda, y el valor de acurracy en la derecha, sobre el modelo Random Forest.

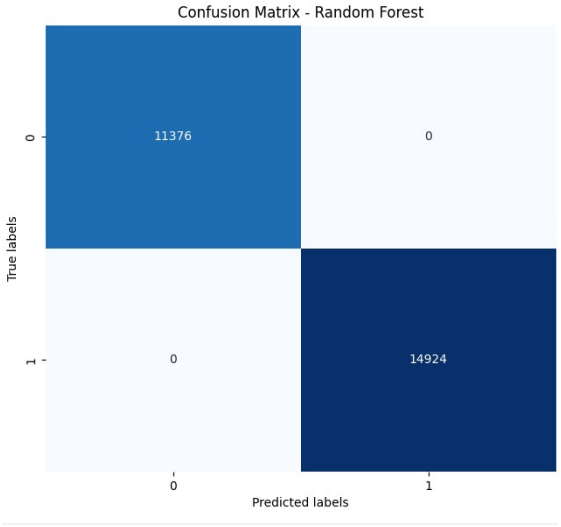
A continuación, se muestra la matriz de confusión donde se puede visualizar el rendimiento del modelo Random Forest, clasificando los datos en diferentes categorías. En esta matriz, las filas representan las clases reales, mientras que las columnas representan las clases predichas por el modelo. La Figura 15 proporciona una visión detallada de los aciertos y errores del modelo al probar con 40,000 (cuarenta mil) datos. Es importante destacar que, al

validar con datos seleccionados de manera aleatoria, se observa una disparidad en el número de instancias propuestas y evaluadas.

Esto se debe a la eventual exclusión de algunas muestras de las clases seleccionadas, o simplemente por tratarse de un conjunto más reducido en comparación con el conjunto de datos completo, tal como se ha observado en investigaciones previas (Calvo Pérez, 2021).

Figura 14

Matriz de confusión del modelo de Random Forest



Nota. Las filas representan las clases reales, y las columnas las clases predichas.

La matriz permite construir la siguiente Tabla 18 como herramienta esencial en la evaluación de modelos de machine learning. Esta tabla proporciona un desglose detallado de cómo el modelo clasifica los datos en distintas categorías. Los verdaderos positivos representan las instancias correctamente clasificadas como positivas (11376), los verdaderos negativos son las instancias correctamente clasificadas como negativas (14924), los falsos positivos son las instancias incorrectamente clasificadas como positivas (0), y los falsos negativos son las instancias incorrectamente clasificadas como negativas (0).

Tabla 18

Datos clasificados de Random Forest

	Positivos Predicción	Negativos Predicción
Positivos Reales	11376	0
Negativos Reales	0	14924

Nota. La tabla contiene los valores de las predicciones del modelo con las clases reales. De izquierda a derecha y de arriba hacia abajo, verdaderos positivos, falsos negativos, falsos positivos y verdaderos negativos.

Con los datos de la tabla, se pueden analizar los indicadores claves finales de desempeño del modelo como la precisión, exactitud y exhaustividad, presentadas en ese orden en la Tabla 19. Estos números proporcionan información esencial sobre la calidad de las predicciones y la capacidad del modelo para identificar correctamente las instancias de interés.

Tabla 19

Precisión, exactitud y exhaustividad de Random Forest.

	Valor	Porcentaje
Precision	0,99984127	99,98%
Acurracy	1	100%
Recall	1	100%

Nota. La tabla representa, de arriba hacia abajo, el resultado de la precisión, exactitud y exhaustividad, y de izquierda a derecha el valor en el rango de 0 a 1, y el porcentaje.

De la Tabla 19, la precisión del 99.98% indica que casi todas las predicciones positivas realizadas por el modelo son correctas. La exactitud del 100% señala que el modelo clasifica correctamente todos los casos, tanto positivos como negativos. La exhaustividad del 100% destaca la capacidad del modelo para identificar la totalidad de las instancias positivas.

Por otro lado, para el modelo de Regresión Logística, los resultados de la validación cruzada mostraron una exactitud promedio de alrededor de 0.891, con ligeros cambios en cada pliegue, como se observa en la Tabla 20.

Tabla 20

Acurracy de las iteraciones de Logistic Regression.

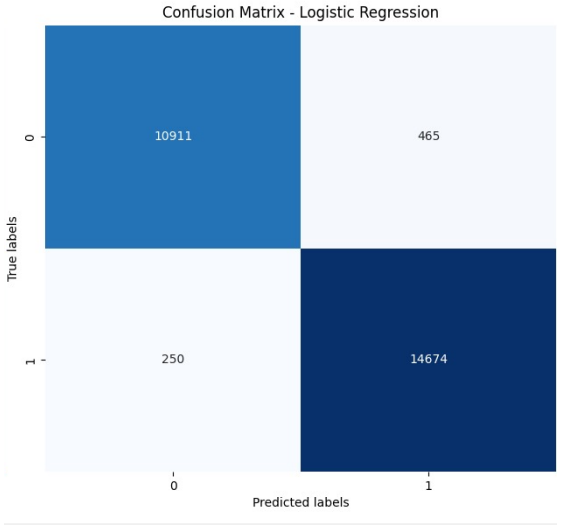
Iteraciones	Acurracy
1	0,896031
2	0,883650
3	0,895714
4	0,889682
5	0,887936
Promedio	0,890603

Nota. La tabla presenta las cinco iteraciones en la columna izquierda, y el valor de acurracy en la derecha, sobre el modelo Logistic Regression.

La Figura 15 proporciona una visión detallada de los aciertos y errores del modelo de regresión logística representados en la matriz de confusión.

Figura 15

Matriz de confusión de Logistic Regression.



Nota. Las filas representan las clases reales, y las columnas las clases predichas

Con la matriz se puede crear la Tabla 21, donde los verdaderos positivos son 10911, los verdaderos negativos 14674, los falsos positivos 250, y los falsos negativos 465.

Tabla 21

Datos clasificados de Logistic Regression.

	Positivos Predicción	**Negativos Predicción**
Positivos Reales	10911	465
Negativos Reales	250	14674

Nota. La tabla contiene los valores de las predicciones del modelo con las clases reales. De izquierda a derecha y de arriba hacia abajo, verdaderos positivos, falsos negativos, falsos positivos y verdaderos negativos.

Al tener los datos, se puede presentar la precisión, exactitud y exhaustividad en la Tabla 22. Estos números proporcionan información esencial sobre la calidad de las predicciones y la capacidad del modelo para identificar correctamente las instancias de interés.

Tabla 22

Precisión, exactitud y exhaustividad de Logistic Regression.

	Valor	**Porcentaje**
Precision	0,7847	78,47%
Acurracy	0,8865	88,65%
Recall	0,6621	66,21%

Nota. La tabla representa, de arriba hacia abajo, el resultado de la precisión, exactitud y exhaustividad, y de izquierda a derecha el valor en el rango de 0 a 1, y el porcentaje.

La precisión indica que aproximadamente el 78.47% de las predicciones positivas realizadas por el modelo son correctas. La exactitud muestra que el modelo clasifica correctamente el 88.65% de todos los casos, tanto positivos como negativos. La exhaustividad destaca la capacidad del modelo para identificar el 66.21% de todas las instancias positivas. Aunque la precisión y la exactitud son relativamente altas, el recall más bajo sugiere que el modelo podría estar perdiendo algunas instancias positivas.

Análisis de resultados

La matriz de confusión en random forest reflejó un alto número de verdaderos negativos y verdaderos positivos, lo que sugiere una mejora significativa en la capacidad del modelo para identificar correctamente las transacciones anómalas. Sin embargo, en regresión logística, hubo un rendimiento considerablemente menor en comparación al anterior modelo. Aunque el modelo de Regresión Logística logró identificar correctamente la mayoría de las transacciones anómalas, su rendimiento general fue inferior en términos de precisión y exactitud.

Al obtener los resultados de los dos modelos y el honeypot implementado para la detección de fraude en las tarjetas de crédito, Random Forest ha superado a Logistic Regression. Contrastando con el proyecto de fin de grado '*Algoritmos de aprendizaje automático para detección de fraudes con tarjetas de crédito: Análisis y comparativa'* (Calvo Pérez, 2021), el resultado con mayor precisión fue regresión logística, que, con la técnica de submuestreo aleatorio obtuvo un 98,7%, y en el caso de este proyecto, con validación cruzada es de 99,98%.

Capítulo V

Conclusiones y Recomendaciones

Conclusiones

El proceso ha finalizado exitosamente, y se ha conseguido la implementación del sistema de detección de intrusiones a través de honeypots en entornos de transacciones en línea, con las siguientes afirmaciones:

- Se realizó una revisión de la literatura donde se investigó el funcionamiento de los honeypots, así como los distintos tipos de honeypots diseñados para rastrear actividades en línea, encontrando las bases para crear uno de baja interacción, además de las técnicas computacionales avanzadas empleadas en su implementación con los modelos y/o algoritmos de machine learning, se adquirió un conocimiento detallado acerca del proceso de implementación en un sistema de transacciones online.
- En un sistema de transacciones online, se implementó un honeypot que detecta el ataque de dos tipos de fraude de tarjetas de crédito: robadas y clonadas. Estableciendo umbrales relacionados a los datos en una transacción. Además, mediante el uso de dos algoritmos de machine learning, Random Forest y Logistic Regression, se consiguió clasificar en transacción anómala o legítima y realizar la monitorización de las transacciones guardando en una base de datos.
- Los resultados de los modelos fueron validados, se detectaron los errores que ocasionaron un bajo acurracy, como la omisión de la lectura del campo de lugar, y el reducido número de datos para el entrenamiento, corrigiendo así en el código el análisis del campo y aumentando el número de datos del dataset, de esta forma se implementó el honeypot en un sistema de transacciones online. Logistic Regression presentó una precisión de 0.7847 (78.47%), acurracy de 0.8865 (88.65%) y

exhaustividad de 0.6621 (66.21%). Mientras que, Random Forest superó al anterior modelo con una precisión de (0.9998) 99.98%, acurracy de 1.0 (100%) y exhaustividad de 1.0 (100%).

- La metodología ha desempeñado un papel crucial en la conclusión exitosa de los objetivos establecidos, gracias a su enfoque iterativo y la importancia que otorga a la retroalimentación constante. Se ha investigado sobre el tema, dividiendo en tres Sprint enfocados en la construcción del sistema de transacciones online. La creación del dataset, la implementación del honeypot y los dos modelos para la detección, para finalmente, emitir alertas y monitorizar las transacciones con un informe. Cumpliendo efectivamente la planificación.

Recomendaciones

- Con el fin de obtener una comprensión abundante del tema y no perderse en los conceptos, las métricas y las tecnologías a emplear en la fase de desarrollo, se aconseja llevar a cabo una revisión literaria antes del estudio.
- Para garantizar la fiabilidad de las fuentes para el desarrollo del trabajo y obtener comparativas relevantes en cuanto a resultados, se sugiere la utilización de bases de datos indexadas como SCOPUS, IEEE y Web of Science, para extraer estudios alineados con el tema de investigación.
- Para el desarrollo de la aplicación, es importante el uso de un conjunto de datos de entrenamiento más extensos que contengan una mayor cantidad de datos significativa, con el objetivo de mejorar la precisión de las predicciones.
- Con el objetivo de mejorar la precisión y la capacidad predictiva, se aconseja ampliar la variedad de características, y encontrar una base de datos de una institución real.

Bibliografía

Ahmed, F. (2023). *Honeypot*. https://forum.huawei.com/enterprise/en/honeypot/thread/668884869385699328-667213854934970368

Alvarado, J., Martillo, I., & Guzmán, J. (2022). *Revisión de literatura sobre las técnicas de Machine Learning en la detección de fraudes bancarios by Julio Alvarado Zabala, Ivette Martillo Alchundia, Jomar Elizabeth Guzmán Seraquive · 10.51798/sijis.v3i1.257 · OA.mg*. https://oa.mg/work/10.51798/sijis.v3i1.257

Baykara, M., & Das, R. (2015). *A Survey on Potential Applications of Honeypot Technology in Intrusion Detection Systems*. *2*.

Benedict, S. (2023). EA-POT: An Explainable AI Assisted Blockchain Framework for Honeypot IP Predictions. *Acta Cybernetica*, *26*(2), 149-173. Scopus. https://doi.org/10.14232/actacyb.293319

Boukela, L., Zhang, G., Bouzefrane, S., & Zhou, J. (2020). An outlier ensemble for unsupervised anomaly detection in honeypots data. *Intelligent Data Analysis*, *24*(4), 743-758. Scopus. https://doi.org/10.3233/IDA-194656

Bringer, M., Chelmecki, C., & Fujinoki, H. (2012). A Survey: Recent Advances and Future Trends in Honeypot Research. *International Journal of Computer Network and Information Security*, *4*. https://doi.org/10.5815/ijcnis.2012.10.07

Calvo Pérez, I. (2021, julio). *Algoritmos de aprendizaje automático para detección de fraudes con tarjetas de crédito: Análisis y comparativa* [Info:eu-repo/semantics/bachelorThesis]. E.T.S.I de Sistemas Informáticos (UPM). https://oa.upm.es/67976/

Chiluiza, D. V. A. (2008). *ESTUDIO Y DISEÑO DE UN SISTEMA DE SEGURIDAD PERIMETRAL PARA LA RED QUITO MOTORS, UTILIZANDO TECNOLOGÍA UTM (UNIFIED THREAT MANAGEMENT)*.

David Tidmarsh. (2023, julio 31). What Is a Honeypot in Cybersecurity? Types, Implementation, and Real-World Applications. *Cybersecurity Exchange*. https://www.eccouncil.org/cybersecurity-exchange/ethical-hacking/what-are-honeypots-benefits-types/

Daza, P. A. (2013, diciembre 19). «Hackers» roban datos de 40 millones de clientes de Target en EE.UU. *CNN*. https://cnnespanol.cnn.com/2013/12/19/hackers-roban-datos-de-40-millones-de-clientes-de-target-en-ee-uu/

Díaz, M. A., & Cuervo, C. (2018, febrero 7). *Análisis de Ataques Informáticos mediante Honeypots para el Apoyo de Actividades Académicas en la Universidad Distrital Francisco José de Caldas*. https://www.semanticscholar.org/paper/An%C3%A1lisis-de-Ataques-Inform%C3%A1ticos-mediante-Honeypots-D%C3%ADaz-Cuervo/f82086fa307fee6a5885f3947a687298a414c27f

Edwin, G. K., Ewards, S. E. V., Willsie Kathrine, G. J., Palmer, G. M., Bertia, A., & Vijay, S. J. (2022). *Honeypot based Intrusion Detection System for Cyber Physical System*. 958-962. Scopus. https://doi.org/10.1109/ICAISS55157.2022.10010931

Gómez, A. A. (2018). *Análisis de vulnerabilidades en IoT para el despliegue de honeypots*.

González, A. (2021). *Honeypot, análisis e implementación. Análisis de resultados y aplicación práctica* [Universidad de Almería]. https://repositorio.ual.es/bitstream/handle/10835/13521/GONZALEZ%20LOZANO,%20ANTONIO%20JESUS.pdf?sequence=1&isAllowed=y

Hardening a honeynet against honeypot-aware botnet attacks: Toward secure cloud. (2015, octubre). International Conference on Cloud Security Management, USA.

Hock, S. (2009). *Review of Agile Methodologies in Software Development*. https://www.researchgate.net/publication/41392207_Review_of_Agile_Methodologies_in_Software_Development

Huang, C., Han, J., Zhang, X., & Liu, J. (2019). *Automatic Identification of Honeypot Server*

Using Machine Learning Techniques. https://www.hindawi.com/journals/scn/2019/2627608/

IBM. (s. f.). *¿Qué es un sistema de detección de intrusiones (IDS)?* | *IBM*. Recuperado 1 de febrero de 2024, de https://www.ibm.com/es-es/topics/intrusion-detection-system

Ilg, N., Duplys, P., Sisejkovic, D., & Menth, M. (2023). A survey of contemporary open-source honeypots, frameworks, and tools. *Journal of Network and Computer Applications*, *220*. Scopus. https://doi.org/10.1016/j.jnca.2023.103737

Islam, S., Haque, M., Naser, A., & Rezaul Karim, A. N. M. (2024). A rule-based machine learning model for financial fraud detection. *International Journal of Electrical and Computer Engineering (IJECE)*, *Vol. 14*, 759-771. https://doi.org/10.11591/ijece.v14i1.pp759-771

Kaur, T., Malhotra, V., & Singh, D. D. (2014). *Comparison of network security tools- Firewall, Intrusion Detection System and Honeypot*. *3*(2).

Kovtun, M. (2018). *Scalable Honeypot Monitoring and Analytics*. https://aaltodoc.aalto.fi/handle/123456789/33660

Kumar, M. S., Ben-Othman, J., Srinivasagan, K. G., & Krishnan, G. U. (2019). Artificial Intelligence Managed Network Defense System against Port Scanning Outbreaks. *2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN)*, 1-5. https://doi.org/10.1109/ViTECoN.2019.8899380

Leyden, J. (2020, marzo 9). AI-powered honeypots: Machine learning may help improve intrusion detection. *The Daily Swig* | *Cybersecurity News and Views*. https://portswigger.net/daily-swig/ai-powered-honeypots-machine-learning-may-help-improve-intrusion-detection

Lutkevich, B. (2023). *What is a honeypot? How it protects against cyber attacks*. Security. https://www.techtarget.com/searchsecurity/definition/honey-pot

Majeed, A., & Rauf, I. (2018). MVC Architecture: A Detailed Insight to the Modern Web

Applications Development. *Peer Review Journal of Solar & Photoenergy Systems*, *1*(1), 1-7.

Mokube, I., & Adams, M. (2007). Honeypots: Concepts, approaches, and challenges. *Proceedings of the 45th Annual Southeast Regional Conference*, 321-326. https://doi.org/10.1145/1233341.1233399

Muhamad Malik Matin, I., & Rahardjo, B. (2020). *A Framework for Collecting and Analysis PE Malware Using Modern Honey Network (MHN)*. 2020 8th International Conference on Cyber and IT Service Management, CITSM 2020. Scopus. https://doi.org/10.1109/CITSM50537.2020.9268810

Nicolás Vidal, P. I. (2022). *Identidad digital en pasarelas de pago*. https://openaccess.uoc.edu/handle/10609/138692

Osorio, I. C. (2021). *Análisis, diseño e implementación de una honeynet para el estudio de malware en entornos industriales IoT*.

Pérez Román, A. M. (2020). *Perfomance comparison among Relational, NoSQL databases and Blockchain*. https://riuma.uma.es/xmlui/bitstream/handle/10630/19413/P%C3%A9rez%20Rom%C3%A1n%20Alberto%20Memoria.pdf?sequence=1&isAllowed=y

Quiñones, D. H. V. (2021). Honeypots dinámicos basados en Blockchain. *INF-FCPN-PGI Revista PGI*, 65-67.

Rakesh, R. K. U., Ye, B., Roden, D., Beazley, C., Gadiya, K., Abraham, B., Brown, D. E., & Veeraraghavan, M. (2019). An Autonomous Labeling Pipeline for Intrusion Detection on Enterprise Networks. *2019 Systems and Information Engineering Design Symposium (SIEDS)*, 1-6. https://doi.org/10.1109/SIEDS.2019.8735629

Rayo Mondragón, C. A. (2020). Prototipo de detección de fraudes con tarjetas de crédito basado en inteligencia artificial aplicado a un banco peruano. *Repositorio Institucional - Ulima*. https://repositorio.ulima.edu.pe/handle/20.500.12724/15294

Rb, A., & Kr, S. K. (2021). Credit card fraud detection using artificial neural network. *Global Transitions Proceedings*, *2*(1), 35-41. https://doi.org/10.1016/j.gltp.2021.01.006

Roldan, L. M., Rincón, L. M., & Taborda, D. A. (2017). *CLONACIÓN DE TARJETAS DE CREDITO*. *2017*.

Rong, C., & Yang, G. (2003). *Honeypots in Blackhat Mode and its Implications*. 185-188. Scopus. https://doi.org/10.1109/PDCAT.2003.1236284

Safaei Pour, M., Nader, C., Friday, K., & Bou-Harb, E. (2023). A Comprehensive Survey of Recent Internet Measurement Techniques for Cyber Security. *Computers and Security*, *128*. Scopus. https://doi.org/10.1016/j.cose.2023.103123

Schwaber, K., & Sutherland, J. (2020). *La Guía Scrum*. https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish-European.pdf

Spitzner, L. (2003). *Honeypots: Tracking Hackers*.

Stoll, C. (1989). *The Cuckoo's Egg: Tracking a Spy Through the Maze of Computer Espionage (1989)*. https://icdt.osu.edu/cuckoos-egg

The Honeynet Project. (2021). https://www.honeynet.org/

Verdejo, G. (2018). *SEGURIDAD EN REDES IP: Honeypots y Honeynets*. https://www.cs.upc.edu/~gabriel/files/DEA-es-4HoneypotsyHoneynets.pdf

Vidalis, S., & Kazmi, Z. (2007). Security Through Deception. *Information Systems Security*, *16*, 34-41. https://doi.org/10.1080/10658980601051458

Vishwakarma, R., & Jain, A. K. (2019). A Honeypot with Machine Learning based Detection Framework for defending IoT based Botnet DDoS Attacks. *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, 1019-1024. https://doi.org/10.1109/ICOEI.2019.8862720

Wang, B.-X., Chen, J.-L., & Yu, C.-L. (2022). An AI-Powered Network Threat Detection System. *IEEE Access*, *10*, 54029-54037. Scopus.

https://doi.org/10.1109/ACCESS.2022.3175886

Yamamoto, Y., & Yamaguchi, S. (2022). A Method to Prevent Known Attacks and Their Variants by Combining Honeypots and IPS. *2022 IEEE 11th Global Conference on Consumer Electronics (GCCE)*, 302-305. https://doi.org/10.1109/GCCE56475.2022.10014099

Yan, J., Wan, M., Jia, X., Ying, L., Su, P., & Wang, Z. (2022). *DitDetector: Bimodal Learning based on Deceptive Image and Text for Macro Malware Detection*. 227-239. Scopus. https://doi.org/10.1145/3564625.3567982

Yang, W. (2008). *A Practical Guide to Honeypots*. https://www.academia.edu/33786315/A_Practical_Guide_to_Honeypots

Yao, J., & Shepperd, M. (2021). The impact of using biased performance metrics on software defect prediction research. *Information and Software Technology*, *139*. Scopus. https://doi.org/10.1016/j.infsof.2021.106664

Zambrano, A., Demera, V., & Vaca-Cardenas, L. (2021). Mecanismos de ciberseguridad basados en honeypots. *Informática y Sistemas: Revista de Tecnologías de la Informática y las Comunicaciones*, *5*, 1. https://doi.org/10.33936/isrtic.v5i2.3708

Anexos