Aplicación de herramientas de automatización robótica de procesos (RPA) en procesos de pentesting para MiPyMEs

Application of robotic process automation (RPA) tools in pentesting processes for MSMEs

M. J. Yepes Díaz; G. E. Taborda Blandón

DOI: https://doi.org/10.22517/23447214.25743
Scientific and technological research paper

***Abstract*— Ethical hacking, also known as penetration testing (pentesting), is an essential practice for identifying vulnerabilities in Information Technology (IT) systems through controlled cyberattack simulations, thereby enhancing IT security. However, the traditional manual approach has limitations due to the exponential growth of technological assets and the increasing complexity of infrastructures. These challenges lead to significant time and resource consumption, as well as the need for specialized technical expertise.This paper examines the integration of Robotic Process Automation (RPA) into pentesting as a solution to streamline and optimize these processes. Through a comparative analysis of documented methodologies and existing RPA tools, we propose a specific tool designed to automate penetration testing within a controlled and secure environment. Experimental results demonstrate that the proposed tool is a viable solution for improving the efficiency, accessibility, and scalability of security audits, offering a practical and robust approach to cybersecurity for a broad range of stakeholders, including both organizations and individuals.**

***Index Terms*— Automated pentesting, Cybersecurity, Penetration testing, Pentesting methodologies, Robotic Process Automation (RPA).**

***Resumen*— El hacking ético, también conocido como pentesting, es una práctica clave para identificar vulnerabilidades en sistemas de Tecnologías de la Información (TI) mediante simulaciones controladas de ataques cibernéticos, lo que permite mejorar la seguridad informática. Sin embargo, el enfoque tradicional, que depende de intervenciones manuales, se enfrenta a limitaciones debido al aumento exponencial de activos tecnológicos y la complejidad de las infraestructuras, lo que implica un alto consumo de tiempo, recursos y la necesidad de experiencia técnica especializada. Este artículo explora la integración de la Automatización Robótica de Procesos (RPA) en el pentesting como una solución para optimizar estos procesos. A través de un análisis comparativo de metodologías documentadas y herramientas RPA disponibles, se propone una herramienta específica para automatizar el pentesting en un entorno controlado y seguro. Los resultados experimentales obtenidos indican que esta herramienta es una alternativa viable para mejorar la eficiencia, accesibilidad y escalabilidad de las auditorías de seguridad, lo que la convierte en una solución efectiva en el ámbito de la seguridad informática.**

***Palabras claves*— Automatización Robótica de Procesos (RPA), Ciberseguridad, Metodologías de pentesting, Pentesting automatizado, Pruebas de penetración.**

I. Introduccion

En el ámbito de la evaluación y pruebas de seguridad informática, el hacking ético, también conocido como pentesting, se refiere a la ejecución de pruebas controladas en sistemas de TI para identificar brechas de seguridad y generar informes que orienten a los administradores en la implementación de medidas preventivas. [1] Esta práctica emula ataques cibernéticos reales, permitiendo evaluar la resiliencia de una infraestructura tecnológica frente a potenciales amenazas [2].

Tradicionalmente, el pentesting ha sido un proceso manual que, debido a su complejidad, se limita a un número reducido de activos para garantizar su efectividad. Sin embargo, el crecimiento exponencial de las redes informáticas ha llevado a la necesidad de automatizar herramientas de pentesting para cubrir un mayor espectro de activos en menor tiempo. Hasta hace poco, las pruebas de penetración dependían exclusivamente de especialistas altamente capacitados con años de experiencia, pero esta limitación, combinada con la escasez de expertos y los altos costos de los procesos manuales, ha evidenciado la necesidad de soluciones más eficientes y accesibles [3] [4].

This manuscript was submitted on December 12, 2024. Accepted on March 11, 2025. A1nd published on March 31, 2025.

This article presents progress on the research project "Functional Prototype of a Computer Platform for Information Security Risk Management and Penetration Testing, Using Automation Technologies and Artificial Intelligence Techniques," developed by the Metropolitan Technological Institute (ITM) and the company Grupo Nex, and funded by the Colombian Ministry of Science, Technology and Innovation.

M. J. Yepes Díaz, Software Development Technology student at the Metropolitan Technological Institute. Cybersecurity Research Group (e-mail: mariayepes309613@correo.itm.edu.co).

G.E. Taborda Blandón, PhD in Complex Systems Thinking, MS in Information Security, MS in Computer Science. Researcher in the Automation, Electronics and Computational Sciences group. OTC Teacher at the Metropolitan Technological Institute (e-mail: gabrieltaborda@itm.edu.co).

Este trabajo presenta avances del proyecto de investigación "*Prototipo funcional de una plataforma informática para la gestión del riesgo de seguridad de la información y pentesting, utilizando tecnologías de automatización y técnicas de inteligencia artificial*", desarrollado por el Instituto Tecnológico Metropolitano de Medellín y la empresa Grupo Nex, que es financiado por el Ministerio de Ciencias, Tecnología e Innovación (MinCiencias. La parte que se centra este artículo está relacionada con la Automatización Robótica de Procesos (RPA) en pentesting y fue apoyada por Maria José Yepes como Joven Investigadora vinculada al proyecto.

En este contexto, se propone un estudio que analiza la integración de la automatización en las auditorías de seguridad informática, especialmente en un entorno donde es crucial fortalecer la ciberseguridad frente a amenazas cada vez más sofisticadas. Este enfoque busca emplear herramientas que optimicen procesos manuales, garanticen respuestas oportunas y hagan más accesible este conocimiento a una audiencia más amplia. El objetivo principal del artículo es explorar la incorporación de la Automatización Robótica de Procesos (RPA) en el pentesting, proponiendo una metodología para la ejecución autónoma de pruebas de seguridad. La propuesta incluye una herramienta de RPA capaz de integrar diversas herramientas utilizadas en diferentes fases del pentesting, recolectar resultados de manera eficiente y presentarlos de forma clara, con un enfoque práctico para su aplicación en pequeñas y medianas empresas (MiPyMEs).

El estudio planteó tres objetivos específicos: analizar metodologías de pentesting para identificar las más adecuadas y automatizables, evaluar herramientas de RPA para seleccionar la más idónea, y validar la eficiencia del modelo de automatización mediante pruebas experimentales en un entorno controlado. Este enfoque se encuentra plasmado en el siguiente trabajo.

II. Estado del arte

La metodología aplicada para la construcción del estado del arte se encuentra incluida dentro del anexo 1 "*Metodología aplicada para el estado del arte*".

A. Pentesting

El pentesting, o prueba de penetración, es un proceso mediante el cual se simulan ataques cibernéticos reales sobre una infraestructura tecnológica para evaluar la solidez de su seguridad y obtener información que podría comprometer el funcionamiento correcto del sistema. Este proceso se basa en un conjunto de métodos, técnicas y estrategias diseñadas para evaluar la robustez de un sistema o red, con el fin de identificar y corregir posibles fallos o vulnerabilidades, poniéndose en el lugar de un atacante que podría intervenir en estos sistemas [5].

En la actualidad, el pentesting ha ganado terreno debido a la creciente relevancia de la seguridad de la información, tanto para empresas como para individuos. Como resultado, muchas organizaciones contratan expertos en ciberseguridad con el propósito de realizar pruebas que evalúan la defensa y mecanismos de seguridad de su infraestructura tecnológica y prevengan posibles ataques que son simulados en estas pruebas, mediante la corrección de los errores detectados y el fortalecimiento de sus sistemas [6].

En este sentido, los resultados retornados por las pruebas de pentesting proporcionan, además información relacionada al cumplimiento de las políticas de seguridad y recuperación ante desastres que son definidas por la organización ante estas situaciones, los mecanismos de mitigación de las brechas de seguridad e incluso la conciencia de seguridad de los empleados de esta [5].

Fig. 1. Fases de un ataque informático (Elaboración Propia)

En Fases de un ataque a un Sistema Informático [7] fig. 1 se revela la complejidad de los ataques cibernéticos, que, como señalan [2], buscan aprovechar las vulnerabilidades de los sistemas informáticos para causar un impacto negativo, incluso tomando el control completo de un sistema. Estos ataques representan una seria amenaza para la integridad y seguridad de los sistemas, y se manifiestan a través de distintas fases, desde el reconocimiento inicial hasta la eliminación de huellas para evitar ser detectados como se describe a continuación.

B. Metodologías de Pentesting

De acuerdo con la estructura de un ataque informático descrita anteriormente, se parte de metodologías y frameworks

de aplicación de Pentesting con métodos específicos y enfoques que pueden variar de acuerdo con las necesidades que se presentan para organizaciones y personas. A continuación, en la tabla I, se detallan algunas de las metodologías más conocidas:

TABLA I
METODOLOGÍAS DE PENTESTING

Metodología	Descripción	Fuente
PTES	Proporciona pasos para auditorías: reconocimiento, análisis de vulnerabilidades, explotación, explotación posterior y elaboración de informes detallados.	[1] [6]
OSSTMM	Ofrece una metodología revisada por pares para pruebas de penetración y auditorías. Cubre la seguridad informática, de procesos, tecnologías, comunicaciones e inalámbrica.	[2] [8] [9] [10]
OWASP	Metodología de pruebas de seguridad en aplicaciones web. Define fases para pruebas a lo largo del ciclo de vida del software: diseño, desarrollo, despliegue y producción.	[11]
ISAAF	Framework de pentesting que divide el proceso en planificación, evaluación y reporte. Se enfoca en requisitos de evaluación de seguridad y gestión de riesgos.	[2] [12]
NIST SP 800-115	Guía técnica para pruebas de seguridad, define fases como planificación, descubrimiento, ataque, notificación de resultados y sugerencias de mejora.	[13]

C. Comparación de Metodologías de Pentesting

En el marco de referencia ISO/IEC 250:10:2013, se definen algunos parámetros claves para definir la calidad de un modelo como son la extensibilidad, mantenibilidad, cobertura de dominio, usabilidad, disponibilidad y confiabilidad. Estos indicadores se utilizarán para medir la metodología más sólida en cuanto a la aplicación de automatización de procesos y a partir de la valoración dada, se seleccionará la mejor valorada según la literatura, que tenga una estructura sólida para el proceso de pentesting y existencia de medidas de contención claras y definidas que permitan la automatización del proceso de pentesting, definiendo los criterios a utilizar para ser medidos de la siguiente manera:

1) *Extensibilidad*
Evalúa si la metodología es clara en sus directrices para extender o personalizar los procedimientos de esta en diferentes escenarios y casos de estudio usando herramientas o técnicas variables.

2) *Mantenibilidad*
Verifica si la documentación de la metodología está bien organizada y es fácil de entender, la frecuencia con la que se actualiza la metodología para reflejar nuevas amenazas y tecnologías y si está dividida en módulos o fases que pueden ser modificados de manera independiente.

3) *Cobertura*
La cobertura revisa la amplitud de áreas de seguridad que la metodología cubre y su adaptación a diferentes industrias y entornos.

4) *Usabilidad*
Evalúa si la metodología contiene materiales de formación y soporte adecuados y qué tipo de errores o problemas encuentran los usuarios al aplicar la metodología

5) *Disponibilidad*
Este criterio verifica si la metodología está fácilmente disponible para los profesionales, evalúa la disponibilidad de herramientas, guías y plantillas que la complementen.

En la Tabla II se sintetiza un comparativo de las metodologías de pentesting con basa a diversos autores.

D. Herramientas de Pentesting

De acuerdo con lo anteriormente descrito sobre pentesting, los tipos existentes y las fases que lo componen, es importante destacar algunas herramientas que en cada fase del pentesting, desde el reconocimiento hasta la explotación, ofrecen utilidades específicas que maximizan la efectividad del proceso.

En la tabla III se hará la descripción de algunas de las herramientas más utilizadas para el proceso de pentesting debido a su reconocimiento, soporte de la comunidad, solidez en el mercado y funcionalidades.

E. RPA

La automatización de procesos ha sido definida como la ejecución por parte de una máquina o un agente virtual de una función que anteriormente fue realizada por un humano, de esta manera se permite la gestión automatizada parcial o total de actividades que son manuales y basadas en reglas [14] [15].

RPA permite pensar sobre robots que ejecutan tareas humanas donde estos, en realidad, corresponden a una solución de software que puede significar una extrapolación de las tareas de un humano ejecutando tareas estructuradas y repetitivas [16].

TABLA II
COMPARACIÓN DE METODOLOGÍAS DE PENTESTING

Factor	PTES	OSSTM	OWASP	ISSAF	NIST SP 800-115
(1)	Reconocida y utilizada ampliamente Potencial para desarrollarse en un framework.	Alta, con resultados medibles y verificables en varios ámbitos de seguridad.	Muy extensible con herramientas y recursos integrables en diversos entornos de desarrollo web	Marco detallado y útil para evaluaciones de seguridad en varias áreas tecnológicas.	Flexible y adaptable a diversos entornos y necesidades organizacionales
(2)	Documentación accesible y estructurada. Última actualización en 2016.	Actualizada por una comunidad activa desde 2010, con un enfoque científico y métrico.	Revisiones y actualizaciones frecuentes por una comunidad global. Última versión en 2020.	Sin actualizaciones activas desde 2006.	Publicada en 2008, última revisión en 2021. Mantenida por NIST, asegurando autoridad y reconocimiento
(3)	Define el alcance en la interacción previa con el cliente. No tan amplia como OSSTMM, ISSAF u OWASP.	Amplia, cubriendo seguridad física, telecomunicaciones, redes, interacciones humanas y aplicaciones.	Amplia variedad de pruebas de seguridad en aplicaciones web. Cubre todo el ciclo de desarrollo de software.	Cobertura integral desde seguridad de infraestructura hasta aplicaciones y evaluación de riesgos.	Amplia cobertura, incluyendo pruebas de penetración, revisiones de configuración y controles de seguridad operacionales.
(4)	Estructura clara con guías técnicas detalladas para cada fase.	Recursos detallados y ejemplos prácticos. Complejidad media y fácil implementación en procesos de reestructuración de seguridad informática	Documentación detallada, guías y herramientas accesibles y fáciles de usar, incluso para principiantes.	Directrices claras y detalladas. Ejemplos prácticos y estudios de caso para mejorar la implementación.	Procedimientos claramente definidos, accesibles y fáciles de seguir. Documentación disponible públicamente.
(5)	Recursos en línea variados, pero sin historial de cambios detallado.	Documentación y recursos disponibles a través de ISECOM y canales autorizados	Herramientas, documentos y foros gratuitos y abiertos a cualquier persona interesada en seguridad de aplicaciones.	Documentación detallada públicamente accesible, pero sin un repositorio oficial fácilmente reconocible	Disponible gratuitamente en el sitio web de NIST, junto con documentación complementaria.

TABLA III
HERRAMIENTAS DE PENTESTING

Herramienta	Fase de Pentesting	Descripción	Fuente
The Harvester	Reconocimiento pasivo	Recolecta información de internet sobre el objetivo, como correos, subdominios y metadatos.	[17]
Nmap	Reconocimiento activo	Escanea puertos y revela información de sistemas operativos, servicios y versiones utilizadas.	[18]
Shodan	Reconocimiento activo	Permite identificar configuraciones de red, puertos expuestos y geolocalización de dispositivos.	[11]
OWASP ZAP	Análisis de vulnerabilidades	Detecta brechas de seguridad mediante escaneo pasivo y ataques de fuerza bruta.	[17]
Metasploit	Explotación	Framework avanzado para pruebas de explotación, análisis de código y ataques de fuerza bruta.	[19]

De esta manera RPA define una manera simple para crear, desplegar y manejar sistemas de software robóticos que imitan los movimientos humanos mientras manipulan la información generada por otros sistemas digitales. Estos robots como las personas pueden comprender lo que se muestra en pantalla, que tipo de teclas deben ser presionadas, a que sistema deben moverse y cómo localizar y extraer información para ejecutar una variedad de otras tareas [20].

La Automatización Robótica de Procesos (RPA) ofrece múltiples ventajas para las empresas, destacando su facilidad de configuración, permitiendo su implementación incluso por desarrolladores sin experiencia en programación. Además, es una tecnología no invasiva que se adapta a los sistemas existentes sin necesidad de costosas plataformas nuevas [16].

RPA incrementa la productividad y precisión, reduciendo errores y perfeccionando procedimientos, lo que mejora la calidad y la satisfacción en las industrias que la adoptan. También alivia la carga laboral del personal, mejora la consistencia de los datos y fortalece la competitividad.

Con un costo inicial bajo y un retorno de inversión alto, RPA fomenta la eficiencia y permite generar ahorros significativos, facilitando iniciativas tácticas [21]. Según [20], combinar RPA con la fuerza laboral humana puede reducir los costos operativos entre un 30% y un 50% en actividades transaccionales.

Según [22], los criterios típicos para los procesos adecuados para RPA son:

1) Requisitos cognitivos bajos

Es difícil que los procesos complejos que contienen una alta cantidad de tareas puedan ser manejados por RPA.

2) Poco acceso a múltiples sistemas

RPA se aplica sobre las aplicaciones existentes o cuenta con integración a unas herramientas específicas y puntuales.

3) Procesos realizados con frecuencia

Los procesos que tienen una alta frecuencia de ejecución o repetición son buenos candidatos para la implementación de RPA.

4) Procesos con alta probabilidad de error humano

Los procesos que requieren un alto nivel de detalle y presentan un elevado riesgo de errores debido a su complejidad, amplitud o manejo de grandes volúmenes de información deberían priorizarse para la implementación de RPA.

F. Herramientas de RPA

RPA es considerada como la tecnología que "automatiza la automatización", debido a su enorme potencial para automatizar procesos. Dicha automatización se realiza mediante agentes de software denominados robots, los cuales se encargan de realizar la ejecución de las tareas; por ejemplo, establecer comunicación entre las interfaces gráficas de usuario de dos aplicaciones de manera autónoma sin la intervención humana. Entre estas herramientas podríamos mencionar algunas como lo son: *Uipath, Automation Anywhere, Blue Prism, WorkFusion.* [17].

Las herramientas RPA son básicamente programas de software que operan sobre la interfaz gráfica de otros sistemas informáticos como si fueran humanos. Hay muchas herramientas disponibles en el mercado que se utilizan para desarrollar RPA como UiPath, Automation Anywhere, Blue Prism, etc [20].

La elección de la herramienta de Robotic Process Automation (RPA) adecuada es crucial para el éxito de la automatización de procesos en una organización. Una selección incorrecta puede resultar en un retorno de inversión (ROI) deficiente, ya que la herramienta elegida podría no ser capaz de manejar las especificidades de los procesos que se desean automatizar. [23] Algunas de las herramientas más destacadas se presentan en la tabla IV.

TABLA IV
HERRAMIENTAS DE RPA

Herramienta	Descripción	Fuente
UiPath	Herramienta basada en C# y vb.net para construir y desplegar robots de software. Ofrece funcionalidades de RPA con IA como reconocimiento de imágenes y minería de texto. Interfaz intuitiva y fácil de usar.	[20] [24] [25] [26] [27]
Automation Anywhere	Plataforma enfocada en escalabilidad y seguridad. Proporciona controles centralizados, integración con ERP, capacidades de IA, y opciones para automatización compleja.	[25] [26] [27] [28]
Blue Prism	Basada en .NET, con arquitectura cliente-servidor. Enfocada en gobernanza y seguridad, ideal para empresas con datos sensibles. Permite modelar y diagramar procesos de automatización con énfasis en eficiencia y seguridad.	[23] [26] [27] [29]
Robocorp	Herramienta open-source basada en Python (Robot Framework). Versátil y asequible, permite crear robots personalizados y utilizar editores como Visual Studio. Ideal para organizaciones que buscan adaptabilidad y reducción de costos.	[24] [30] [31]

G. Comparación de Herramientas de RPA

Para la elaboración del análisis comparativo se definen los siguientes criterios de comparación

1) Arquitectura

Este criterio se refiere a la estructura técnica de la herramienta, ya sea basada en cliente-servidor o por medio de un orquestador en la web, esto permite definir de manera clara el proceso de implementación y despliegue de RPA para pentesting.

2) Integración

La integración evalúa la capacidad de la herramienta para interactuar con otras aplicaciones y sistemas existentes. Una buena integración es crucial para la automatización fluida de procesos que involucran múltiples plataformas como es el caso de las pruebas de pentesting.

3) Procesos que pueden ser automatizados

Indica si la herramienta es adecuada para la automatización de procesos de oficina, atención al cliente o procesos más complejos en diferentes sectores.

4) Precio

Se considera el costo de la licencia del software o si es de código abierto, lo que es un factor determinante para la ejecución de pruebas en entornos que tengan herramientas y funcionalidades avanzadas.

5) Soporte y Comunidad

Se evalúa la calidad del soporte que se ofrece con la herramienta y la existencia de una comunidad activa que proporcione documentación y de razón de la mantenibilidad de la herramienta.

6) Popularidad

Este criterio considera la popularidad de la herramienta a través del análisis de búsquedas y de herramientas que proporcionan distintas métricas con el fin de determinar su grado de aceptación y reconocimiento por parte del público objetivo.

A continuación, se presenta una comparación de las herramientas UiPath, Automation Anywhere, Blue Prism y Robocorp, en relación con estos criterios definidos:

UiPath

- *Arquitectura:* Basada en un orquestador web con componentes como UiPath Studio, Orchestrator y Robots, para administrar robots en la nube o localmente [23] [29].
- *Integración:* Compatible con herramientas de ofimática (Word, Excel, correo), plataformas como AWS, Oracle, Microsoft y SAP, facilitando la interacción con objetos en pantalla [26] [32].
- *Procesos que automatiza:* Inicialmente, desarrolló bibliotecas para IBM, Google y Microsoft, enfocándose en sectores como BFSI, salud, telecomunicaciones y retail [23] [29].
- *Precio*: Ofrece una edición gratuita (Community) con limitaciones para distribución de bots y una empresarial con costos iniciales de $3,000–$5,000 USD, con una prueba gratuita de 60 días [26] [28].
- *Soporte y Comunidad:* Más de 1.5 millones de descargas, 750,000 desarrolladores y 250 socios tecnológicos respaldan la herramienta [28].
- *Popularidad*: Tendencia de crecimiento estable, con alta popularidad en Norteamérica, Europa y Asia [34].

Automation Anywhere

- *Arquitectura*: Cliente-servidor con tres componentes principales: Bot Creator, Control Room y Bot Runner, que permiten medir el rendimiento del robot [23] [29].
- *Integración*: Compatible con Google, Salesforce, SAP, Azure y otras APIs. [26]
- *Procesos que automatiza:* Utilizada en procesos de sectores como BFSI, salud, manufactura y telecomunicaciones (p.e.j., General Motors, JP Morgan Chase) [23] [29].
- *Precio*: Cuenta con una versión Community gratuita y opciones empresariales desde $9,000 USD anuales. No incluye pruebas dedicadas [24] [26].
- *Soporte y Comunidad:* Academia en línea, eventos, webinars y soporte especializado [33].
- Popularidad: Posición destacada en Norteamérica, Asia y Latinoamérica [34].

Blue Prism

- *Arquitectura*: Cliente-servidor, con componentes para diagramado, modelado y despliegue en la nube, local o híbrido [23] [29] [33].
- *Integración*: Compatible con Microsoft Power Platform, Azure, Salesforce, SAP y Google, aunque con mayor complejidad en la identificación de objetos [26] [35].
- *Procesos a Automatizar:* Sectores regulados como BFSI, telecomunicaciones y manufactura, con enfoque en gobernanza y cumplimiento normativo [27].
- *Precio:* Learning Edition gratuita por 180 días, luego una prueba de 30 días. Costos anuales de $15,000 USD [26] [35].
- *Soporte y Comunidad:* Foro Digital Exchange (DX) para compartir herramientas y certificaciones en línea [33] [36].
- *Popularidad*: Base de usuarios fieles en Europa y Norteamérica, especialmente en sectores regulados [34].

Robocorp

- *Arquitectura*: Nativa en la nube, soporta despliegue en instalaciones propias, nube o entornos híbridos [25] [36].
- *Integración*: Compatible con SAP, Salesforce, HubSpot, navegadores, APIs, herramientas ofimáticas y funciones de IA [36].
- *Procesos a Automatizar:* Flexible para sectores financieros, consultoría y salud, con casos como Grant Thornton LLP [36].
- *Precio:* Open-source y gratuito para proyectos individuales, con opciones empresariales para orquestación en la nube [24] [36].
- *Soporte y Comunidad:* Comunidad activa, documentación extensa y soporte para pequeñas y medianas empresas [33].
- *Popularidad:* Crecimiento sostenido en Norteamérica y Europa [24] [34].

En función de los criterios anteriormente mencionados y la descripción de las herramientas, se muestra a continuación en la fig. 2 la valoración de las características:

Herramienta	Arquitectura	Integración	Procesos	Precio	Comunidad / Soporte	Popularidad
UiPath	5	5	5	3	5	5
Automation Anywhere	4	4	4	3	4	4
Blue Prism	4	3	4	2	4	3
Robocorp	5	5	5	4	4	3

Fig. 2. Valoración por medio de criterios de Comparación de herramientas de RPA (Elaboración Propia)

En la fig. 3 se presenta la gráfica de búsqueda en Google de la diferente herramienta de RPA, que apoya el criterio de popularidad.

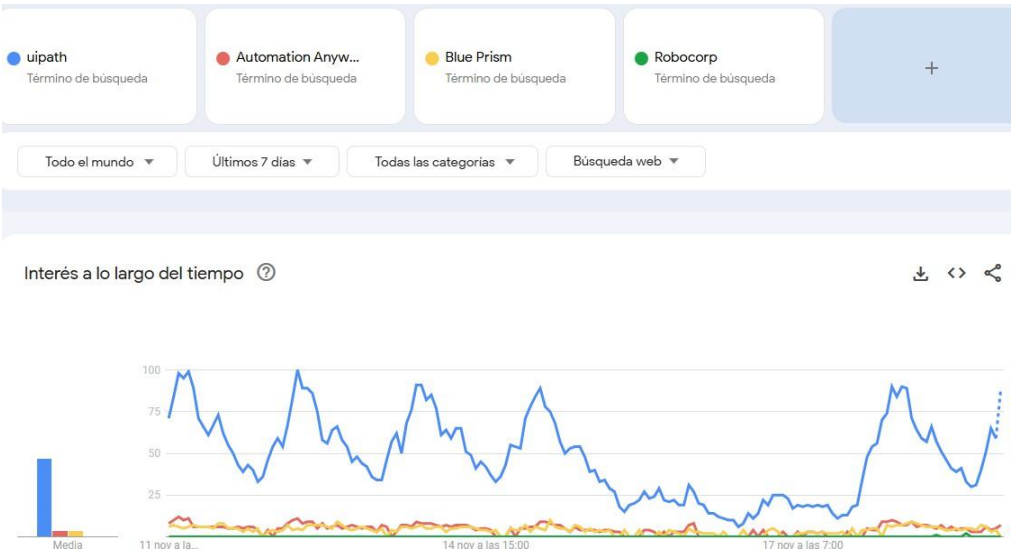

Fig. 3. Tendencias de búsqueda de herramientas de RPA por Google Trends [33]

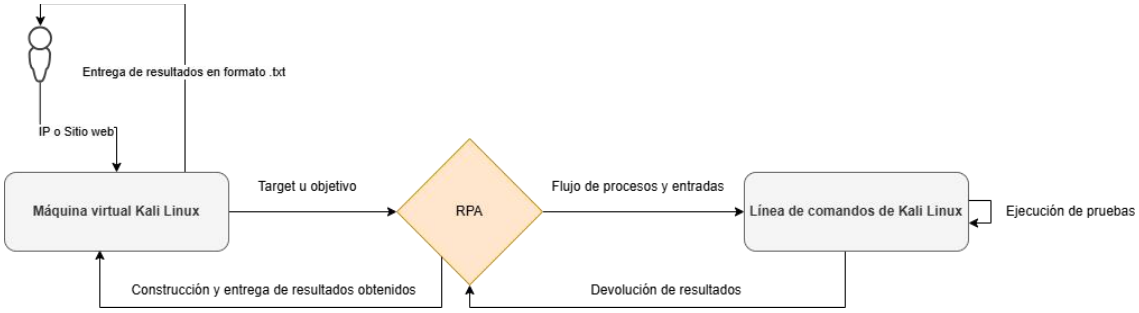

Fig. 4. Modelo conceptual del funcionamiento de RPA en ambiente controlado de pruebas (Elaboración Propia)

La necesidad de construir un prototipo de RPA que permita la automatización de tareas de pentesting de manera eficiente, con capacidad de escalabilidad y de tipo open-source, se decide que la herramienta más adecuada es Robocorp.

Robocorp ofrece una arquitectura nativa de la nube que permite un despliegue versátil y eficiente, ya sea en la nube, en las instalaciones o con configuraciones híbridas; además de que el modelo de código abierto de la herramienta reduce significativamente los costos asociados al desarrollo y despliegue de robots.

El modelo de código abierto de Robocorp permite escalar las operaciones de manera más eficiente, con una implementación personalizada que integra herramientas de pentesting, en contraste con otras ofertas donde las operaciones bajo esquemas de licencia pueden resultar prohibitivas.

H. Pentesting Automatizado con RPA

En el trabajo de Matzenberger, [31] Robocorp se usa junto a Process Mining para identificar procesos repetitivos en organizaciones y desarrollar robots que los ejecuten eficientemente. Este enfoque asegura que los robots optimicen el flujo de trabajo mediante un ciclo de análisis, pruebas y monitoreo continuo, maximizando la eficiencia en la operación de tareas estructuradas.

El artículo de Delilovic [30] expone cómo Robocorp puede integrarse de manera segura con Amazon AWS, utilizando el AWS Security Token Service (STS) para la autenticación con credenciales temporales. Este tipo de integración permite automatizar tareas en la nube sin comprometer la seguridad, al cumplir con requisitos de autenticación multifactor y acceso controlado. Esta implementación ofrece un modelo extensible para cualquier entorno de nube que demande altos estándares de seguridad.

En [37] Robocorp ha demostrado ser una herramienta efectiva para automatizar pruebas de seguridad. Correia presenta una implementación donde Robocorp se integra con el escáner de vulnerabilidades ZAP (Zed Attack Proxy), utilizando el lenguaje ASL4RPA para definir y ejecutar pruebas de seguridad en aplicaciones web. Esta configuración permite simular ataques de inyección SQL y cross-site scripting (XSS), documentando los resultados en tiempo real.

Esta integración optimiza el proceso de auditoría al reducir en un 75-85% procesos manuales, permitiendo que los

especialistas en seguridad se enfoquen en el análisis de vulnerabilidades en lugar de la configuración de herramientas. La combinación de Robocorp con ZAP no solo incrementa la eficiencia en pruebas de seguridad, sino que facilita la adopción de RPA como parte de las estrategias de ciberseguridad, permitiendo auditorías no asistidas o parcialmente asistidas.

III. METODOLOGIA

Robocorp, como herramienta de automatización robótica de procesos (RPA), ha sido ampliamente adoptada en distintas industrias para optimizar tareas repetitivas y complejas. Su flexibilidad le permite adaptarse tanto a la automatización de procesos convencionales como a aplicaciones avanzadas en entornos seguros.

Con el propósito de la construcción de una herramienta de automatización de pentesting por medio de automatización robótica de procesos, se propone el marco de la herramienta Robocorp y los procesos descritos por la metodología PTES dada la secuencialidad y síntesis de esta.

Para respaldar y dar un acercamiento a la herramienta con necesidades reales, se construye una encuesta la cual tiene por objetivo, evaluar dentro de un rango de expertos en ciberseguridad las herramientas con más potencial de automatización en el marco de RPA y que procesos de pentesting serían eficientemente automatizados por herramientas de RPA. Los resultados detallados de la encuesta se encuentran en el anexo 2 "*Tablero de Power BI con resultados de la encuesta*" De acuerdo con esta encuesta, fue posible recolectar la siguiente información:

- El nivel de experiencia de área de los encuestados se sitúa en un nivel Avanzado e Intermedio comprendiendo el uso de técnicas de pentesting o la familiaridad con procesos de seguridad informática en un período desde los 2 años a más de 5 años, esto con un porcentaje del 38% respectivamente para cada uno, y de un 24% de principiantes para conocimiento inferior a 2 años.

- De estos expertos, la segmentación de áreas donde trabajan se ubica dentro del área de la gestión de vulnerabilidades en un 20%, la gestión de riesgos de ciberseguridad, pruebas de pentesting y auditorías de seguridad en una mayor medida con un porcentaje de 17,78% cada una. También se detallan algunas otras áreas de actividad de las personas encuestados a continuación.

- Se evidencia que el 72,73% de los encuestados no ha utilizado herramientas de pentesting automatizado para ejecutar pruebas de seguridad en contraste con un 27,2% que afirma haber utilizado estas herramientas, de esta manera es posible inferir que no es muy conocido el uso de automatización dentro de los procesos de pentesting.
- Para los encuestados que han utilizado estas herramientas, se destaca la funcionalidad de automatización de escaneos, en la que sobresalen Nessus, OpenVAS y Tennable, cada una con un 20% de preferencia. Estas herramientas permiten realizar análisis de vulnerabilidades sobre direcciones IP o dominios, eliminando la necesidad de intervenciones manuales para obtener información de seguridad.

- Otras herramientas mencionadas en las respuestas incluyen: Burp Suite que permite automatizar pruebas de penetración y detectar vulnerabilidades, Checkmarx que realiza análisis estático de código (SAST) de forma automatizada, Immuni Web para facilitar escaneos automáticos en aplicaciones web, móviles y API y Nikto para escanear servidores web en busca de configuraciones inseguras.

- Se logra evidenciar además una tendencia dentro del uso de herramientas de pentesting, destacándose como herramientas principales Nmap (12,57%), Nessus (11,98%); Wireshark (10,78%), OWASP ZAP (9,58%) y Metasploit (8,98%).

- Los encuestados proporcionan su visión de algunos de los procesos con potencial de automatización, donde se destacan principalmente el escaneo de puertos y servicios (12,27%), el escaneo de vulnerabilidades (11,66%), la generación de informes (10,43%) y las pruebas de fuerza bruta (9,82%) pero también se presentan otras opciones con potencial de automatización como las pruebas de inyección de código, inyección SQL y de DoS y DDoS.

- De esta manera y en concordancia con lo anterior, se solicita especificar qué herramientas cuentan con un mayor potencial de automatización donde se destaca Nmap (20,22%) como una de las herramientas más aptas para automatización, seguida de Nessus (16,85%), Metasploit (14,61%), OWASP ZAP (8,99%) y BurpSuite (7,87%).

Gracias a la encuesta realizada fue posible determinar algunas perspectivas claves de cómo algunas herramientas existentes proporcionan una funcionalidad de automatización como es el caso de Nessus, OpenVAS, Burp Suite, donde se es posible por medio de una entrada de información de un target específico ejecutar una serie de pruebas.

También se identifica una tendencia de automatización de procesos de pentesting en torno a procesos que involucran escaneos de puertos, identificación de servicios y configuraciones de red, además de destacarse la realización de pruebas y generación de reportes.

Finalmente, frente a la apreciación descrita para algunas herramientas que tienen potencial de automatización, se logra ver que algunas de las respuestas reflejan utilidades que ya

cuentan con una automatización propia dentro de su estructura como es el caso de Nessus y OWASP ZAP, por lo que da a entender un concepto de automatización por parte de los encuestados que involucre automatización en integración con otras herramientas o que permita aplicar automatización con respecto a otros procesos como los seleccionados por los encuestados. En la Fig. 4 se ilustra el modelo conceptual implementado en esta investigación.

IV. Desarrollo de la solucion

De esta manera, para la ejecución de las pruebas se propone la implementación de un RPA que permita por medio de una entrada de tipo IP o dominio web, realizar diferentes procesos con las herramientas mejor valoradas con potencial de automatización por parte de los encuestados y que logren cubrir en su mayoría los procesos descritos.

Se seleccionan en orden de relevancia, la herramienta Nmap utilizada para el proceso de escaneo de puertos y reconocimiento de servicios, Nessus para el escaneo integral de vulnerabilidades y configuraciones, Metasploit en la realización de pruebas de explotación y SQLMap con participación activa en pruebas de inyección de código SQL. Estas fases están descritas dentro de la metodología PTES como las etapas principales para la elaboración de pentesting, todo esto finalizando con la recolección de los resultados de estas pruebas por parte del RPA en un archivo que pueda sintetizarlos.

El RPA se encontrará dentro de una máquina virtual de Kali Linux, lo cual le proporcionará la capacidad de elaborar la integración con herramientas de pentesting de manera mucho más eficiente y donde recorrerá de manera ordenada cada uno de los procedimientos con las herramientas utilizando la herramienta Robocorp dentro del entorno de desarrollo de Visual Studio Code en el lenguaje de programación de Python. El diagrama de flujo que describe el proceso del RPA se describe a continuación.

Dentro del entorno de desarrollo controlado que se dispuso dentro de la máquina virtual, se implementó en Python un script que realiza mediante diferentes pasos, procedimientos con las diferentes herramientas seleccionadas. En primer lugar, se realiza para la fase de reconocimiento y escaneo de puertos un escaneo con Nmap fig. 5 que incluye los servicios que se están ejecutando sobre los puertos que el mismo software logra encontrar y se almacenan dentro de un archivo con los resultados:

```python
def scan_with_nmap(target):

    print(f"Ejecutando escaneo Nmap en el target: {target}")
    result = subprocess.run(["nmap", "-sV", "-A", target], capture_output=True, text=True)
    save_results("Nmap", result.stdout)
```

Fig. 5. Código Python utilizado en Robocorp para ejecución de escaneo en Nmap.

Seguidamente, se ejecuta el proceso con el escáner de vulnerabilidades de Nessus fig.6, haciendo una petición a la API para poder ejecutar un análisis de red básico, y recolectar también los resultados de la operación

```python
def scan_with_nessus(target):
    """Crear, lanzar y obtener resultados de un escaneo en Nessus"""
    scan_data = {
        "uuid": "731a8e52-3ea6-a291-ec0a-d2ff0619c19d7bd788d6be818b65",
        "settings": {
            "name": "ScanExample",
            "description": "This is a test scan",
            "text_targets": target,
            "enabled": True
        }
    }

    try:

        response = make_request_with_retries(f"{NESSUS_URL}/scans", data=scan_data, method="POST")
        response_data = response.json()

        if "scan" in response_data:
            scan_id = response_data["scan"]["id"]
            logging.info(f"Escaneo creado con ID: {scan_id}")
        else:
            raise ValueError(f"No se pudo crear el escaneo: {response.text}")

        start_scan = make_request_with_retries(
            f"{NESSUS_URL}/scans/{scan_id}/launch",
            method="POST"
        )

        if start_scan.status_code == 200:

            logging.info("Escaneo lanzado correctamente.")

        else:
            error_msg = start_scan.json().get("error", "Error desconocido")
            logging.error(f"No se pudo lanzar el escaneo: {error_msg}")
            return
```

Fig. 6. Fragmento de código en Python utilizado para realizar escaneo de vulnerabilidades con la Herramienta Nessus.

Para finalizar, se realiza una prueba de *directory listing* por medio de Metasploit sobre el target y una prueba de inyección SQL por medio de la herramienta SQLMap utilizando las siguientes instrucciones como se muestra en la fig. 7.

```python
def exploit_with_metasploit(target):

    print(f"Ejecutando Metasploit en el target: {target}")

    with open("metasploit_commands.rc", "w") as f:
        f.write(f"use auxiliary/scanner/http/dir_scanner\n")
        f.write(f"set RHOSTS {target}\n")
        f.write("run\n")

    result = subprocess.run(["msfconsole", "-q", "-r", "metasploit_commands.rc"], capture_output=True,
    save_results("Metasploit", result.stdout)

def scan_with_sqlmap(target):

    print(f"Ejecutando SQLMap en el target: {target}")
    url = f"http://{target}/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit"
    result = subprocess.run(["sqlmap", "-u", url, "--batch"], capture_output=True, text=True)
    save_results("SQLMap", result.stdout)
```

Fig. 7. Fragmento de código en Python para realizar escaneo de vulnerabilidades con la Herramienta SQLMap

V. Resultados

La ejecución del robot se da por medio del comando *ROBOT_TARGET= {target} rcc run* el cual permite recibir como parámetro una dirección IP o dominio para realizar las pruebas. Para la ejecución de pruebas se utiliza la IP local, sobre una aplicación desplegada dentro de la máquina virtual, corriendo dentro del mismo servidor y que es conocida por su amplia gama de vulnerabilidades para ejecutar pruebas de seguridad.

Los resultados devueltos por cada proceso de las herramientas se encuentran en un documento que recoge los hallazgos principales del escaneo realizado a la aplicación desplegada de manera local. Principalmente se encuentran los resultados de la fase de Reconocimiento con la herramienta de Nmap fig. 8.

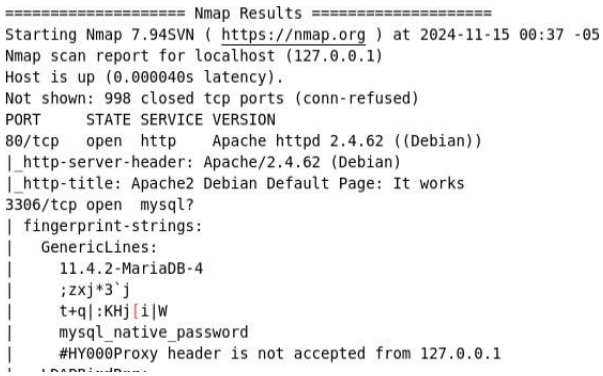

Fig. 8. Resultados del escaneo Nmap por medio del RPA incluidos en un documento de texto.

Los resultados evidenciados anteriormente permiten obtener más información sobre los puertos abiertos dentro del objetivo y algunos de los posibles servicios ejecutándose en los mismos. De manera similar se ejecuta el escaneo de vulnerabilidades utilizando la API de Nessus como insumo principal para dirigir el escaneo.

Se obtiene la lista de vulnerabilidades por parte de Nessus que logra encontrar en el target local, mostrando además dentro del archivo algunos de los controles aplicables como se evidencia en la fig. 9. Comprendiendo la fase de análisis de vulnerabilidades y modelamiento de amenazas propuesta por la metodología PTES.

Fig. 9. Resultados del escaneo Nessus por medio del RPA incluidos en un documento de texto.

De acuerdo con los hallazgos se procede a realizar pruebas de explotación sobre el objetivo utilizando el framework Metasploit y la herramienta SQLMap.

Se completa la prueba retornando algunas de las rutas que fue posible encontrar dentro de la aplicación seguidos de un código que indica la respuesta del target al intentar acceder a estas rutas. fig. 10

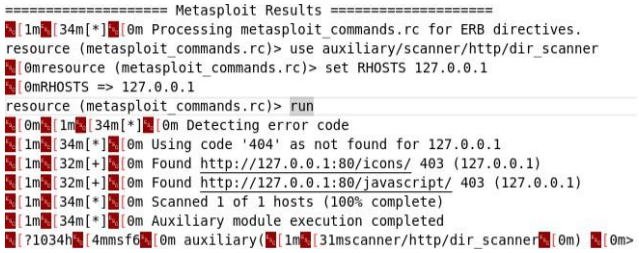

Fig. 10. Resultados de prueba de directory listing utilizando Metasploit Framework por medio del RPA incluidos en un documento de texto.

Finalmente, por medio de la herramienta SQLMap fue posible identificar la tecnología principal que utiliza el target a nivel de base de datos, como se ilustra en la fig. 11 y que es información útil que conlleva a la realización de pruebas combinadas.

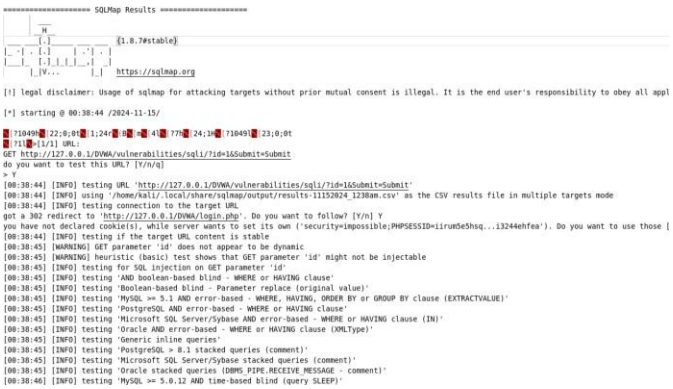

Fig. 11. Resultados de prueba de SQL Injection utilizando SQLMap por medio del RPA incluidos en un documento de texto.

VI. Conclusiones

Dada la creciente importancia de la ciberseguridad en las micro, medianas y pequeñas empresas la implementación de una solución basada en Automatización Robótica de Procesos (RPA) para optimizar tareas de pentesting ha demostrado ser viable y altamente beneficiosa permitiendo que mediante una instrucción puedan ejecutarse pruebas específicas que pueden ser relevantes para una organización y le den resultados que pueda utilizar para comprender qué tipo de información podría tener un agente externo si llegase a darse un ataque a un activo de información.

Mediante el uso del software RPA Robocorp como plataforma base, se logró desarrollar una herramienta de código abierto, escalable y personalizable, adecuada para entornos controlados como una máquina virtual Kali Linux.

Este prototipo ha permitido automatizar de manera eficiente las fases principales de la metodología PTES (desde el reconocimiento hasta la explotación), integrando herramientas clave de ciberseguridad como Nmap, Nessus, Metasploit y SQLMap y algunas pruebas simples como son el escaneo de puertos, escaneo de vulnerabilidades, enumeración de usuarios e inyección de código SQL.

El desarrollo e integración de esta solución destacan por su capacidad para simplificar procesos complejos, reducir significativamente los costos y tiempos de ejecución, además, su flexibilidad permite ampliar el alcance de las auditorías mediante la integración de nuevas herramientas y procesos, ofreciendo un enfoque escalable y adaptable a diferentes escenarios.

Este trabajo no sólo valida el uso de RPA en el contexto del pentesting, sino que también abre nuevas posibilidades para la automatización en ciberseguridad, especialmente en procesos definidos. Sin embargo, queda mucho terreno por explorar en la automatización de tareas más dinámicas y no estructuradas, lo que subraya la necesidad de continuar investigando y desarrollando soluciones innovadoras en esta área que puedan ofrecer además una mejor organización de la información y un catálogo de controles que apoyen a las MiPyMEs.

La herramienta desarrollada representa un avance significativo hacia la optimización del pentesting, demostrando que la automatización no solo es posible, sino también práctica y efectiva en entornos controlados. Este enfoque marca un punto de partida para futuras investigaciones y aplicaciones en la automatización de pruebas de seguridad.

ANEXOS

Anexo 1: Método empleado para la revisión sistemática de la literatura, para no hacer muy extenso este manuscrito es ubico en el siguiente enlace: **Metodología aplicada para el estado del arte**

Anexo 2: La encuesta o consulta a expertos de seguridad informática sobre herramientas de pentesting y los resultados obtenidos se encuentran en el enlace: **Tablero de Power BI con resultados de la encuesta**

Anexo 3: Las preguntas que fueron llevadas a cabo en la encuesta hacia los expertos de seguridad informática es posible encontrarlas en el enlace a continuación: **Encuesta sobre Herramientas de Pentesting y su Automatización con RPA**

REFERENCIAS

[1] I. B. Lahmar, Cybersecurity: Hacking and penetration testing techniques and methodologies, 2021.

[2] I. A. Coronel and D. I. Quirumbay, "Seguridad informática, metodologías, estándares y marco de gestión en un enfoque hacia las aplicaciones web," 2022.

[3] M. C. Ghanem, T. M. Chen, and E. G. Nepomuceno, "Hierarchical reinforcement learning for efficient and effective automated penetration testing of large networks," *Journal of Intelligent Information Systems*, vol. 60, 2022. [Online]. Available: https://doi.org/10.1007/s10844-022-00738-0

[4] E. A. Altulaihan, A. Alismail, and M. Frikha, "A survey on web application penetration testing," *Electronics*, vol. 12, no. 5, 2023. [Online]. Available: https://doi.org/10.3390/electronics12051229

[5] A. M. Ortiz, *Introducción a las pruebas de penetración*, 2020.

[6] A. Arce Rendón, A. Samacá Burbano, and C. Urcuqui López, "Artificial intelligence model for the automation of information collection in the recognition phase of pentesting," 2023.

[7] J. Calle Condori, "Fases de un ataque a un Sistema Informático," *Revista PGI. Investigación, Ciencia y Tecnología en Informática*, no. 7, pp. 52-55, 2020.

[8] J. F. Caranqui Allaica, "Auditoría de la seguridad informática siguiendo la metodología Open Source Security Testing Methodology Manual (OSSTMM) para la empresa MEGAPROFER S.A.," 2020.

[9] ISECOM, *OSSTMM 3*, 2010.

[10] C. Núñez Alcalá, *Penetration testing: Auditoría profesional*, 2021.

[11] OWASP, *Web security testing guide. WSTG - Stable OWASP Foundation*. [Online]. Available: https://owasp.org/www-project-web-security-testing-guide/stable/. [Accessed: 2024].

[12] A. Shanley and M. N. Johnstone, "Selection of penetration testing methodologies: A comparison and evaluation," in *Australian Information Security Management Conference*, 2015.

[13] I. M. Raazia, M. Malahayati, B. Basrulb, R. Maliac, and M. Fadhli, "Analysis server security assessment of staffing management information system using the NIST SP 800-115 method at UIN Ar-Raniry Banda Aceh," *Circuit: Jurnal Ilmiah Pendidikan Teknik Elektro*, vol. 8, 2024. [Online]. Available: https://doi.org/10.22373/crc.v8i1.20808

[14] C. A. Bermúdez Irreño, "RPA - Automatización robótica de procesos: Una revisión de la literatura," *Rev. Ingeniería, Matemáticas y Ciencias de la Información*, vol. 8, 2021. [Online]. Available: https://dx.doi.org/10.21017/rimci.2021.v8.n15.a97

[15] E. K. Chiou and J. D. Lee, "Trusting automation: Designing for responsivity and resilience," *Human Factors*, vol. 65, no. 1, 2023. [Online]. Available: https://doi.org/10.1177/00187208211009995

[16] J. G. Enríquez, A. Jímenez-Ramírez, F. J. Domínguez-Mayo, and J. A. García-García, "Robotic process automation: A scientific and industrial systematic mapping study," *IEEE Access*, vol. 8, 2020. [Online]. Available: https://doi.org/10.1109/ACCESS.2020.2974934

[17] S. Maji, H. Jain, V. Pandey, and V. A. Siddiqui, "White hat security: An overview of penetration testing tools," in *2nd International Conference on Advancement in Electronics & Communication Engineering (AECE 2022)*, 2022.

[18] Z. Asrak, *Penetration testing tools: The use of penetration testing tools in Kali Linux*, 2020.

[19] I. U. Haq and T. A. Khan, "Penetration frameworks and development issues in secure mobile application development," 2021. [Online]. Available: https://doi.org/10.1109/ACCESS.2021.3088229

[20] R. Mehta and R. Chaher, "Implementation of robotic process automation (RPA) in digital marketing," in *3rd International Conference for Emerging Technology (INCET)*, 2022.

[21] J. I. Amador Escalera, "Propuesta metodológica para implementar RPA's," 2020.

[22] J. Siderska, "Robotic process automation — A driver of digital transformation?," *Engineering Management in Production and Services*, vol. 12, no. 2, 2020. [Online]. Available: https://doi.org/10.2478/emj-2020-0009

[23] S. Khan, "Comparative analysis of RPA tools - UiPath, Automation Anywhere and BluePrism," 2020. [Online]. Available: https://doi.org/10.47760/ijcsma.2020.v08i11.001

[24] R. Sindhuja, P. T. Modugu, S. A. Goud, E. R. Kumar, G. S. Babu, and R. Reddy, "A comparative analysis of RPA tools: UiPath, Automation Anywhere and Robocorp," in 2024 OPJU International Technology Conference (OTCON) on Smart Computing for Innovation and Advancement in Industry 4.0, Raigarh, India, 2024, pp. 1-6, doi: 10.1109/OTCON60325.2024.10688237.

[25] J. Ribeiro, R. Lima, T. Eckhardt, and S. Paiva, "Robotic process automation and artificial intelligence in Industry 4.0 – A literature review," in *CENTERIS - International Conference on ENTERprise Information System*, 2021. [Online]. Available: https://doi.org/10.1016/j.procs.2021.01.104

[26] B. Axmann and H. Harmoko, "Process & software selection for robotic process automation (RPA)," 2022. [Online]. Available: https://doi.org/10.31803/tg-20220417182552

[27] P. Desai, S. Joshi, Y. Desai, N. Kothari, and D. Sawant, "Leading platforms in robotic process automation: Review," in *Proceedings of the International Conference on Cognitive and Intelligent Computing*, 2022. [Online]. Available: https://doi.org/10.1007/978-981-19-2350-0_62

[28] D. Andrade, "Challenges of automated software testing with robotic process automation RPA - A comparative analysis of UiPath and Automation Anywhere," International Journal of Intelligent Computing Research, vol. 11, pp. 1066-1072, 2020. doi:10.20533/ijicr.2042.4655.2020.0129.

[29] S. Baweja, "Exploring advanced process automation with Blue Prism," 2023.
[30] N. Delilovic, Implementing Advanced Amazon AWS Authentication Capabilities for the Robot Test-Automation Framework, 2022.
[31] R. Matzenberger, *Exploring open-source robotic process automation: The Robocorp approach*, 2022.
[32] UiPath, "Integrations with enterprise applications - Automation partners." [Online]. Available: https://www.uipath.com/partners/technology-alliances. [Accessed: Nov. 21, 2024].
[33] S. Mandvikar, "Indexing robotic process automation products," International Journal of Computer Trends and Technology, vol. 71, pp. 52-56, 2023. doi: 10.14445/22312803/IJCTT-V71I8P108.
[34] Google Trends, "Explore Google Trends." [Online]. Available: https://trends.google.com. [Accessed: Nov. 5, 2024].
[35] Blue Prism, "Blue Prism RPA software." [Online]. Available: https://www.blueprism.com. [Accessed: Nov. 5, 2024].
[36] Robocorp, "Robocorp - Open source RPA for developers." [Online]. Available: https://robocorp.com. [Accessed: Nov. 21, 2024]
[37] C. Correia, A. Silva, and V. Lobo, "Cybersecurity test automation: Experiences with RPA tools and ZAP technologies using ASL4RPA," in *2024 International Conference on Emerging Computing and Engineering Technologies (ICECET)*, 2024, pp. 1-6, doi: 10.1109/ICECET61485.2024.10698536.

Maria José Yepes Díaz: Egresada del programa Tecnología en Desarrollo de Software del Instituto Tecnológico Metropolitano y estudiante de Ingeniería de Sistemas. Actualmente es miembro activa del semillero de investigación en Ciberseguridad, participando en proyectos enfocados en la automatización de procesos para mejorar la eficiencia en pruebas de penetración (pentesting). https://orcid.org/0009-0000-7369-7687.

Gabriel Enríque Taborda Blandón: PhD en Pensamiento Complejo de Multiversidad Mundo Real Edgar Morin (México), Master en Seguridad Informática por la Universidad Internacional de La Rioja (España) y Master en Computer Science de la Atlantic International University (EEUU). Profesor investigador en el área de Ingeniería de Software y Seguridad Informática. Profesor en el área de Sistemas en educación superior. Actualmente vinculado al Instituto Tecnológico Metropolitano (docente OTC). Liderando línea de investigación en Ciencias Computacionales del Grupo de Investigación en Automatización, Electrónica y Ciencias computacionales del Instituto Tecnológico Metropolitano. https://orcid.org/0000-0002-8067-1490.