AZCATL
Revista de divulgación en ciencias, ingeniería e innovación

Ciberseguridad mediante inteligencia artificial

Luis Alberto Flores Montaño
Universidad Autónoma Metropolitana, Unidad Lerma
luisfloresmontano@hotmail.com

Jacobo Sandoval Gutiérrez
Universidad Autónoma Metropolitana, Unidad Lerma
j.sandoval@correo.ler.uam.mx

Resumen

Esta investigación implementa un modelo de inteligencia artificial (IA) para proteger un sistema embebido, representado por un robot móvil con una Raspberry Pi, frente a ciberataques. La metodología incluye el diseño del sistema, el desarrollo de un entorno de prueba de ciberataques y la integración de un modelo ligero de IA entrenado para detectar y mitigar amenazas. Los experimentos evalúan la eficacia del sistema en la detección de ataques, el tiempo de respuesta y el impacto en el rendimiento del *hardware* de recursos limitados. El enfoque se valida para su uso en otros dispositivos embebidos (internet de las cosas [IoT]), asegurando su viabilidad en aplicaciones críticas.

Palabras clave

Ciberseguridad, IA, sistemas embebidos.

Abstract

This research implements an artificial intelligence (AI) model to protect an embedded system, represented by a mobile robot with a Raspberry Pi, against cyberattacks. The methodology includes system design, the development of a cyberattack test environment, and the integration of a lightweight AI model trained to detect and mitigate threats. The experiments evaluate the system's effectiveness in detecting attacks, response time, and impact on the performance of resource-constrained hardware. The approach is validated for use in other embedded (IoT) devices, ensuring its viability in critical applications.

Keywords

Cybersecurity, AI, embedded systems.

APA: Flores, L. y Sandoval, J. (2025). Ciberseguridad mediante inteligencia artificial. *Azcatl, 4,* 32-37. DOI: 10.24275/AZC2025A006

Introducción

La ciberseguridad es fundamental para un mundo interconectado en donde las amenazas son más complejas (Guembe *et al.,* 2022). Es por ello que la inteligencia artificial (IA) emerge como una herramienta prometedora para realizar la detección, respuesta y prevención ante ataques en tiempo real. El presente estudio se enfoca en aplicar la IA a sistemas embebidos, presentes hoy día en diversos dispositivos, como automóviles, aparatos del hogar e incluso en equipos médicos, los cuales enfrentan limitaciones de procesamiento y de energía, por lo que son vulnerables a los ciberataques. La IA permite proteger estos dispositivos de manera eficiente, incluso con *hardware* de bajo costo, garantizando su funcionamiento seguro en redes complejas (Álvarez, 2024; Ayerbe, 2020).

En el ámbito de los sistemas embebidos, varios estudios han demostrado el uso de la IA para mejorar la ciberseguridad (Hladun, 2024; Zhang y Li, 2023). Sin embargo, muchos de estos enfoques se centran en *hardware* de alto rendimiento y rara vez se aplican en dispositivos con restricciones de procesamiento y memoria. En el caso de los sistemas autónomos y los robots móviles con sistemas embebidos, la necesidad de soluciones ligeras y adaptativas es crítica. Mientras que las investigaciones previas han explorado técnicas de IA para la navegación autónoma y el control (Kaur *et al.,* 2023), la integración de la IA para la protección contra ciberataques en estos sistemas aún es un área en desarrollo (Li, 2018).

La investigación expuesta en este artículo utiliza el Freenove-Kit de coche inteligente 4WD con Raspberry Pi —un sistema embebido— para implementar un modelo ligero de inteligencia artificial (IA), el cual es una versión optimizada y reducida de un modelo diseñado para funcionar en dispositivos de recursos limitados basado en técnicas de TinyML y modelos de aprendizaje automático que protejen el vehículo contra ciberataques, como inyección de código o ataques de denegación de servicio (DoS). Este vehículo robótico, equipado con sensores y módulos de control, permite experimentar en entornos de simulación representativo de sistemas autónomos. Además, demuestra la viabilidad de aplicar IA para identificar y mitigar ciberataques en tiempo real, incluso en *hardware* con recursos limitados. Por lo que el modelo propuesto podría adaptarse a otros dispositivos IoT y sistemas embebidos, fortaleciendo la ciberseguridad en aplicaciones críticas.

Metodología

La metodología se desarrolla en cinco etapas clave: diseño del sistema, desarrollo del entorno de prueba, implementación de la IA, experimentación con ciberataques y evaluación del rendimiento del sistema. Con lo anterior, se busca cubrir un *framework* de ciberseguridad enfocado en el marco NIST SP 800-53, que abarca las funciones de identificar, proteger, detectar, responder y recuperar (Dempsey, 2014). En la Figura 1 se muestra gráficamente el proceso de la metodología aplicada.

1. Diseño del sistema embebido y arquitectura de seguridad

En esta etapa se consideran componentes físicos como el Freenove-Kit de coche inteligente 4WD (Figura 2), el cual proporciona la plataforma física móvil con sensores integrados (ultrasónicos, infrarrojos, giroscopios, etcétera) que simulan las funcionalidades de los vehículos autónomos; una Raspberry Pi que actúa como el núcleo de procesamiento, controlando los sensores y actuadores del vehículo, teniendo en cuenta que es el dispositivo donde se ejecutará el modelo de IA; asimismo, se tiene una conexión de red bajo una comunicación inalámbrica (wifi), esto con el propósito de simular los ciberataques y para monitorear el sistema.

Para los componentes del *software* se tiene un sistema operativo Raspbian OS (Linux) que se utiliza en la Raspberry Pi, un modelo de IA ligero como es el aprendizaje automático, entrenado para la detección de patrones anómalos, es decir, creación de procesos desconocidos, aumento repentino de procesos, saturación de CPU o RAM e incremento inusual del tráfico de red, todos éstos consecuencia de los ciberataques.

2. Desarrollo del entorno de prueba

Esta fase consiste en la creación de un entorno simulado para realizar los experimentos, así como los cibera-

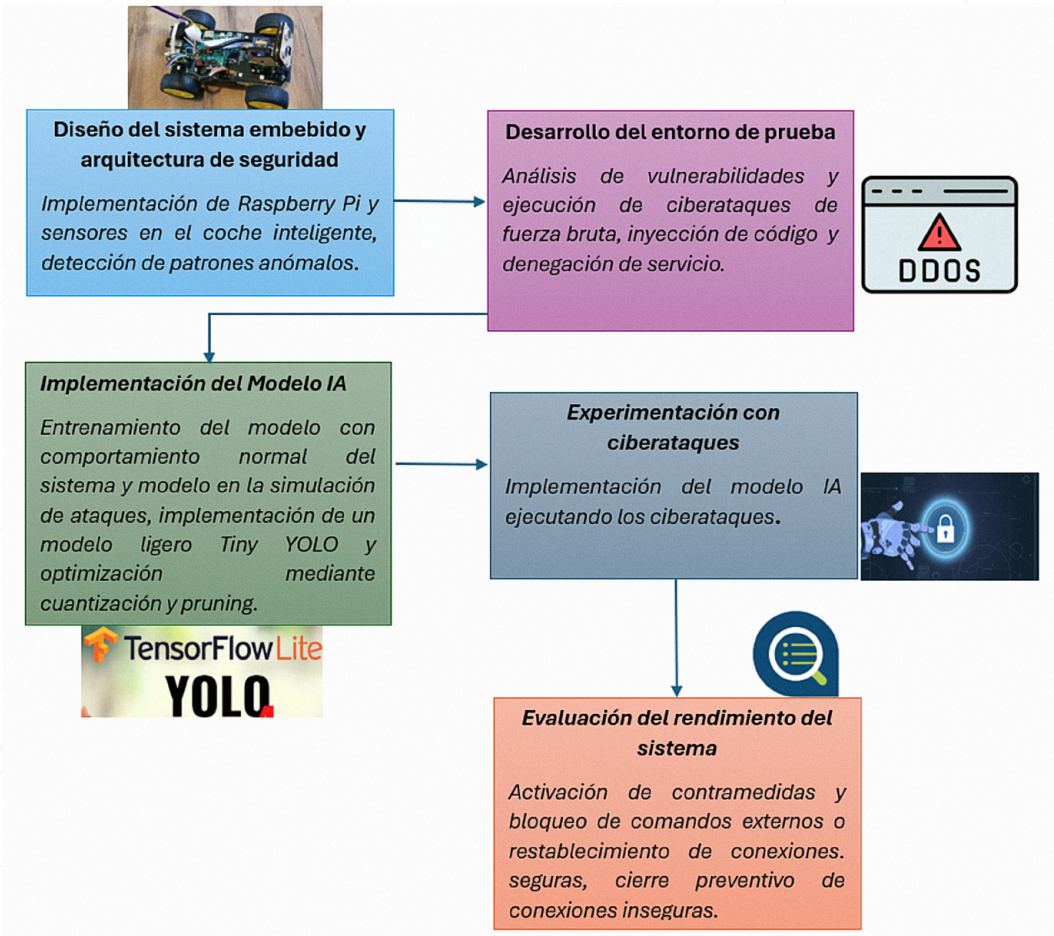

Figura 1. Diagrama de la metodología.

taques al coche inteligente; por lo que se programan rutas predefinidas, se configuran los sensores y se establece una comunicación con un servidor remoto para simular actualizaciones y comandos.

Los ciberataques controlados incluyen un escaneo de puertos con Nmap para identificar los dispositivos que están conectados, seguido de un análisis de vulnerabilidades realizado con Nessus (Colque, 2020). Una vez que se tienen estas vulnerabilidades, se ejecutan ataques de fuerza bruta utilizando el diccionario Rockyou (Chaudhary y Kumar, 2024), de inyección de código con Metasploit (Raj y Walia, 2020) y de denegación de servicio (DoS) usando Hping3 (Ahda *et al.*, 2023). Estas técnicas buscan infiltrarse en el sistema, saturar la red y afectar la capacidad de respuesta del vehículo.

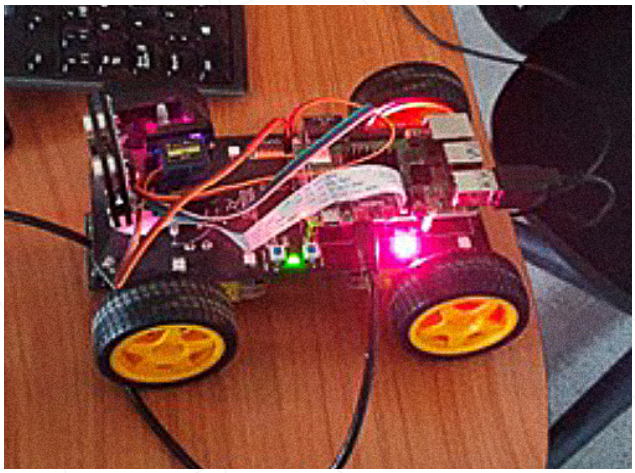

Figura 2. Freenove-Kit de coche inteligente 4WD con sensores.

3. Implementación del modelo IA

En esta etapa se selecciona e implementa un modelo de IA ligero que pueda operar de manera eficiente en la Raspberry Pi con recursos limitados. El modelo se entrena para detectar patrones de comportamiento anómalos en los datos de los sensores y la red; para ello, primero se recopilan datos sobre el comportamiento normal del sistema (navegación sin interferencias) y luego se realizan simulaciones de ciberataques.

El proceso de entrenamiento para implementar un modelo de IA en una Raspberry Pi 4 comienza con la selección de un modelo ligero, como es el caso de Tiny YOLO (Ma *et al.,* 2018), el cual es adecuado para sistemas de bajo procesamiento. Dicho modelo se entrena inicialmente en una máquina con mayor capacidad computacional para utilizar un conjunto de datos relevante para la tarea específica. Una vez completado el entrenamiento, el modelo se optimiza mediante una cuantización y un *pruning* para reducir su tamaño y consumo de recursos, luego se convierte a un formato compatible en TensorFlow Lite (David *et al.,* 2021; Demosthenous y Vassiliades, 2021). Finalmente, el modelo optimizado se implementa en la Raspberry Pi 4 para realizar inferencia en tiempo real, aprovechando —si es posible— el hardware acelerado para mejorar su rendimiento.

4. Experimentación con ciberataques

Una vez implementado el modelo de IA, se llevan a cabo experimentos en condiciones controladas para evaluar su capacidad de detección y respuesta frente a diferentes tipos de ciberataques.

Primero se realiza una prueba de inyección de código, la cual intenta manipular el comportamiento del coche mediante la inyección de comandos maliciosos. El objetivo del modelo de IA es que identifique estos comandos y bloquee su ejecución. Adicionalmente, se realiza una prueba de denegación de servicio (DoS), la cual inunda la red del vehículo con solicitudes falsas, simulando un ataque DoS. El sistema debe detectar la sobrecarga y mantener la estabilidad de las operaciones esenciales; adicionalmente, el sistema responde en tiempo real generando alertas y registros.

5. Evaluación del rendimiento del sistema

En esta última fase, se realiza una validación para el modelo utilizando un conjunto de datos que incluye tanto comportamiento normal como anómalo para ajustar su precisión. Posteriormente, el modelo se integra en el sistema embebido, en el *software* de control del coche inteligente, donde monitorea en tiempo real los datos de los sensores y la red.

El sistema está diseñado para detectar actividades sospechosas y activar contramedidas (por ejemplo, bloquear comandos externos o restablecer conexiones seguras). En el cierre preventivo de conexiones inseguras al detectar una anomalía, la IA activa una respuesta de contención que incluye el bloqueo inmediato de todas las conexiones entrantes o salientes que no sean críticas para el sistema; esto debe evitar la entrada de paquetes maliciosos o la propagación del ataque a otros sistemas embebidos en la red, es decir, si el sistema embebido se ve comprometido, la IA desconecta el dispositivo de la red principal para contener la amenaza, impidiendo que el ataque se propague a otros sistemas críticos.

Después se evalúa la eficacia del modelo de IA en términos de su capacidad de detección de ciberataques, respuesta y eficiencia operativa. Esto se realiza mediante métricas de evaluación, a través de las cuales se obtiene la precisión de detección por tasa de verdaderos positivos y negativos frente a falsos positivos y negativos. Por otro lado, se tiene la medición del tiempo de respuesta: cuánto tarda el sistema en detectar y mitigar un ataque. Asimismo, se obtiene el impacto en el rendimiento del sistema embebido, el cual mide el uso del CPU, memoria y energía para asegurar que el modelo de IA no sobrecargue el *hardware*. Por último, se mide la robustez, esto es la capacidad del sistema para seguir funcionando correctamente bajo ataques sostenidos. Todo esto consigue la obtención de datos para la capacidad de recuperación y para evaluar qué tan bien se recupera el sistema tras un ataque exitoso.

Resultados

La implementación del modelo de IA en la Raspberry Pi 4 para mejorar la ciberseguridad del coche inteligente

Freenove-Kit 4WD fue exitosa. El modelo, optimizado con TensorFlow Lite, mostró una capacidad destacada para detectar patrones anómalos en los datos de los sensores. Se llevaron a cabo pruebas con distintos tipos de ataques: escaneo de puertos, denegación de servicio, acceso no autorizado e inyección de código. El sistema respondió en tiempo real, generando alertas y registros. En la Tabla 1 se presentan los resultados respecto a cada ciberataque.

Durante las pruebas de ciberataques, el modelo identificó y bloqueó eficazmente inyecciones de código maliciosos y también mitigó los efectos de ataques DoS, manteniendo la estabilidad de los sensores y actuadores esenciales con sólo un 3 % de pérdida de datos. Se evaluó el rendimiento del sistema embebido ejecutando la IA en condiciones reales. Se midieron métricas, como consumo energético, latencia, precisión y robustez ante ruido en la red. En la Tabla 2 se muestran los datos de las métricas evaluadas.

En cuanto a la robustez, el sistema demostró una alta capacidad de recuperación, con tiempos promedio de restauración de 1.2 minutos después de ataques DoS exitosos. Adicionalmente, los mecanismos de contención, como el ajuste dinámico del *firewall* y el aislamiento del sistema comprometido, garantizaron la seguridad del sistema sin comprometer su funcionalidad.

Conclusiones

La implementación de un modelo de inteligencia artificial ligero en un sistema embebido basado en Raspberry Pi 4 ha demostrado ser efectiva para mejorar la ciberseguridad del coche inteligente Freeno-ve-Kit 4WD, dentro de un entorno de prueba controlado. Los resultados obtenidos muestran que la solución es capaz de detectar patrones anómalos en los datos de sensores y de red, respondiendo de manera rápida y eficaz ante ciberataques, como inyecciones de código malicioso y ataques de denegación de servicio (DoS).

El uso del modelo optimizado como lo es Tiny YOLO, adaptado para operar en un dispositivo con recursos limitados, permitió una detección precisa con un consumo de recursos adecuado, esto sin comprometer el rendimiento del sistema embebido. El modelo también fue eficaz para la implementación de contramedidas, como el bloqueo de conexiones inseguras y el aislamiento del sistema comprometido, lo que garantizó la seguridad continua del vehículo.

Tabla 1. Resultados ante cada ciberataque.

Tipo de ataque	Descripción	Precisión de detección (%)	Falsos positivos (%)	Tiempo de detección (ms)
Escaneo de puertos	Uso de Nmap para identificar servicios activos.	100.0	1.2	98
Ataque DoS	Saturación de red mediante múltiples peticiones.	92.3	4.1	132
Acceso no Autorizado	Intento de login remoto SSH con fuerza bruta.	96.5	2.6	115
Inyección de Comandos	Uso de comandos maliciosos en formularios.	94.8	3.3	121

Tabla 2. Métricas evaluadas.

Métrica evaluada	Sin IA	Con IA	Impacto (%)	Observación
Precisión de detección	NA	95.86 %	NA	Modelo entrenado con dataset etiquetado localmente.
Tiempo promedio de respuesta	15 ms	135 ms	+800 %	Aceptable en aplicaciones no críticas.
Consumo energético promedio	3.2 W	3.7 W	+15.6 %	Aumento moderado, viable para sistemas IoT.
Uso promedio de CPU	22 %	57 %	+159 %	CPU ARM Quad-core 1.5 GHz (Raspberry Pi 4).
Tasa de falsos positivos	NA	3.25 %	NA	Dentro de rango aceptable para sistemas de alerta.
Robustez ante tráfico ruidoso	Baja	Alta	-	La IA mantiene detección efectiva con ruido de red.

Los tiempos de recuperación fueron breves y la capacidad del sistema para mantenerse operativo durante los ciberataques sostenidos evidencia la robustez de la solución. Esto remarca el potencial para ser aplicado en entornos de sistemas embebidos más complejos, como en el caso de vehículos autónomos. Para futuras investigaciones, se espera mejorar la precisión del modelo en escenarios que sean más dinámicos y ampliar la capacidad de detección frente a amenazas completamente nuevas.

Referencias

Ahda, A., Wulandari, C., Husellvi, H. P., Alhuda, M. Y., Reda, M., Zahwa, P. y Ananda, S. (2023). Information security implementation of DDoS attack using hping3 tools. *Journal of Computer Science*, *1*(4).

Álvarez, A. F. (2024). *Estado del arte de técnicas de inteligencia artificial que aporten en la ciberseguridad* [Tesis de ingeniería]. Universidad Politécnica Salesiana.

Ayerbe, A. (2020). La ciberseguridad y su relación con la inteligencia artificial. *Real Instituto Elcano*, *128*.

Chaudhary, A. y Kumar, K. (24-28 de junio de 2024). *Vulnerability Analysis of WPA Security Protocols*. 15th International IEEE Conference on Computing Communication and Networking Technologies, Mandy, India.

Colque, S. I. J. (2020). Escáner de vulnerabilidades aplicando Nessus. *Revista Ciencia y Tecnología Informática*, *1*(1), 5-10.

David, R., Duke, J., Jain, A., Janapa Reddi, V., Jeffries, N., Li, J., ... Rhodes, R. (2021). Tensorflow lite micro: embedded machine learning for tinyml systems. *Proceedings of Machine Learning and Systems*, *3*, 800-811.

Demosthenous, G. y Vassiliades, V. (2021). *Continual learning on the edge with tensorflow lite*. Arxiv.

Dempsey, K. L., Witte, G. A. y Rike, D. (2014). *Summary of NIST sp 800-53, revision 4, security and privacy controls for federal information systems and organizations*. Computer Security Division-National Institute of Standards and Technology.

Guembe, B., Azeta, A., Misra, S., Osamor, V. C., Fernandez-Sanz, L. y Pospelova, V. (2022). The emerging threat of ai-driven cyber attacks: a review. *Applied Artificial Intelligence*, *36*(1).

Hladun, I. (2024). *Embedded AI systems: a guide to integrating ML in embedded systems*. Waverley. https://waverley*software*.com/blog/embedded-ai-systems-guide/

Kaur, R., Gabrijelčič, D. y Klobučar, T. (2023). Artificial intelligence for cybersecurity: literature review and future research directions. *International Journal on Information Fusion*, *97*. https://doi.org/10.1016/j.inffus.2023.101804

Li, J. H. (2018). Cyber security meets artificial intelligence: a survey. *Frontiers of Information Technology & Electronic Engineering*, *19*(12), 1462-1474.

Ma, J., Chen, L. y Gao, Z. (2018). Hardware implementation and optimization of Tiny-YOLO network. En G. Zhai, J. Zhou, H. Yang, P. An y X. Yang (Eds.), *Digital TV and wireless multimedia communication*. Springer.

Raj, S. y Walia, N. K. (2-4 de julio de 2020). A study on metasploit framework: a pen-testing tool. *2020 International Conference on Computational Performance Evaluation*, Shillong, India.

Zhang, Z. y Li, J. (2023). A review of artificial intelligence in embedded systems. *Micromachines*, *14*(5), p. 897. https://doi.org/10.3390/mi14050897