Framework basado en inteligencia artificial para mejorar la detección de patrones de ataques de fuerza bruta en los logs de autenticación de seguridad en servidores, 2025

Piero Andres Aliaga Fernandez

Estudiante de Ingeniería de Ciberseguridad de la Facultad de Ingeniería Eléctrica y Electrónica de la Universidad Nacional de Ingeniería

Correo: piero.aliaga.f@uni.pe

Resumen

La detección ineficaz de ataques de fuerza bruta en los logs de servidores representa una vulnerabilidad crítica para la integridad de los sistemas informáticos. Esta investigación propone y valida un framework basado en un modelo de Autoencoder con Redes Neuronales Recurrentes (LSTM) para mejorar la detección de dichos ataques. La metodología abarca desde el procesamiento de logs crudos hasta la implementación y calibración fina de un modelo de Deep Learning. El sistema fue entrenado exclusivamente con secuencias de comportamiento normal para aprender a identificar patrones anómalos a través del error de reconstrucción. Tras un proceso de optimización ponderada del umbral de decisión, el framework final demostró un rendimiento robusto y equilibrado, alcanzando un Recall del 80% y una Precisión del 93% en la detección de anomalías. Estos resultados validan el enfoque como una solución eficaz y pragmática, que prioriza la detección de amenazas reales manteniendo un nivel manejable de falsas alarmas, demostrando su viabilidad para entornos operativos.

Palabras clave: *Inteligencia Artificial, Ciberseguridad, Detección de Ataques, Fuerza Bruta, Logs de Autenticación, LSTM Autoencoder.*

1. Introducción

En la era digital, los logs de servidores se han convertido en registros cruciales para el monitoreo, la auditoría y el análisis forense de sistemas informáticos. Sin embargo, el volumen masivo y la naturaleza semi-estructurada de estos datos hacen que su análisis manual sea inviable y propenso a errores, especialmente durante incidentes de seguridad (Flynn & Olukoya, 2025). La seguridad de estos registros y su autenticidad para detectar manipulaciones es una preocupación central en la literatura (Bajramovic et al., 2023), una de las amenazas más persistentes y comunes contra los servidores es el ataque de fuerza bruta, donde un atacante intenta obtener acceso no autorizado probando sistemáticamente múltiples credenciales.

El problema central radica en la ineficacia de los métodos tradicionales basados en reglas para detectar patrones de ataque complejos o de baja intensidad. Estos sistemas a menudo fallan al no poder contextualizar eventos a lo largo del tiempo. La inteligencia artificial, y en particular el Deep Learning, ofrece un paradigma prometedor para abordar este desafío mediante el aprendizaje de patrones de comportamiento a partir de los propios datos.

Este estudio tiene como objetivo principal diseñar, implementar y evaluar un framework basado en un modelo de LSTM Autoencoder para mejorar la detección de patrones de ataques de fuerza bruta en logs de autenticación. La contribución de este trabajo es

presentar una metodología completa y reproducible, desde el procesamiento de datos crudos hasta la calibración de un modelo optimizado para un equilibrio entre la máxima detección de amenazas y la viabilidad operativa.

2. Metodología

El desarrollo del framework siguió un proceso estructurado en cuatro fases técnicas, utilizando un enfoque cuantitativo de tipo aplicado y diseño experimental.

2.1. Dataset

El conjunto de datos utilizado fue el archivo combined_auth.log, un registro de autenticación real de un servidor que contiene una mezcla de eventos legítimos generados por servicios como sshd y CRON, así como actividades anómalas correspondientes a intentos de acceso no autorizados.

2.2. Procesamiento de Datos

Los logs crudos se procesaron utilizando un script de Python. Se aplicaron expresiones regulares (Regex) para analizar cada línea y extraer características fundamentales como el timestamp, el proceso generador del log, el tipo de evento (ej. "Failed password", "Accepted password"), el usuario y la dirección IP de origen. Esta información se estructuró en un DataFrame de Pandas para su posterior manipulación.

2.3. Ingeniería de Características Para que los datos pudieran ser procesados por una red neuronal, se realizaron dos transformaciones clave:

1. **Codificación One-Hot:** Las variables categóricas como event_type se convirtieron a un formato numérico binario para evitar que el modelo infiriera relaciones ordinales inexistentes.
2. **Creación de Secuencias:** Utilizando una técnica de ventana deslizante, los eventos se agruparon en secuencias de 15 pasos de tiempo consecutivos. Este paso es fundamental para que el modelo LSTM pueda aprender el contexto temporal y los patrones de transición entre eventos.

2.4. Arquitectura y Entrenamiento del Modelo El núcleo del framework es un **LSTM Autoencoder**. Esta arquitectura consta de:

- Un **Encoder** con una capa LSTM que comprime la secuencia de entrada en un vector de características latentes.
- Un **Decoder** con otra capa LSTM que intenta reconstruir la secuencia original a partir de dicho vector.

El modelo fue entrenado exclusivamente con secuencias de comportamiento normal. El objetivo del entrenamiento es minimizar el error de reconstrucción (medido con el Error Absoluto Medio - MAE), haciendo que el modelo se vuelva un experto en la reconstrucción de patrones legítimos. La hipótesis es que las secuencias de ataque, al ser desconocidas para el modelo, generarán un error de reconstrucción significativamente mayor.

2.5. Evaluación y Calibración del Umbral Un autoencoder no clasifica directamente, sino que produce un puntaje de error. La decisión de si una secuencia es anómala depende de un umbral. Para encontrar el punto de operación óptimo, se implementó un algoritmo de

búsqueda que probó sistemáticamente 99 umbrales distintos. El umbral final fue seleccionado por su capacidad para maximizar un **F1-Score ponderado**, asignando un peso del 70% al rendimiento en la detección de anomalías y un 30% al de eventos normales. Este enfoque asegura que la prioridad del modelo sea la detección de amenazas, sin ignorar por completo la necesidad de un rendimiento equilibrado. El valor del umbral óptimo encontrado fue de 0.0561

2.6. Métricas de Evaluación: Formulación Matemática

El rendimiento del clasificador binario se evalúa a partir de los cuatro posibles resultados de la matriz de confusión:

- **Verdadero Positivo (TP):** Anomalía correctamente identificada como Anomalía.
- **Verdadero Negativo (TN):** Normal correctamente identificado como Normal.
- **Falso Positivo (FP):** Normal incorrectamente identificado como Anomalía (Error Tipo I).
- **Falso Negativo (FN):** Anomalía incorrectamente identificada como Normal (Error Tipo II).

A partir de estos valores, se definen las siguientes métricas:

Precisión (Precision): Mide la fiabilidad de las predicciones positivas.

$$Precision = \frac{TP}{TP + FP}$$

Sensibilidad (Recall): Mide la capacidad del modelo para encontrar todas las muestras positivas.

$$Recall = \frac{TP}{TP + FN}$$

Puntuación F1 (F1-Score): Es la media armónica de Precisión y Recall, proporcionando un balance entre ambas.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

3. Resultados

Tras aplicar el umbral ponderado óptimo a las predicciones del modelo sobre el conjunto de prueba, se obtuvieron los resultados consolidados en la Tabla 1.

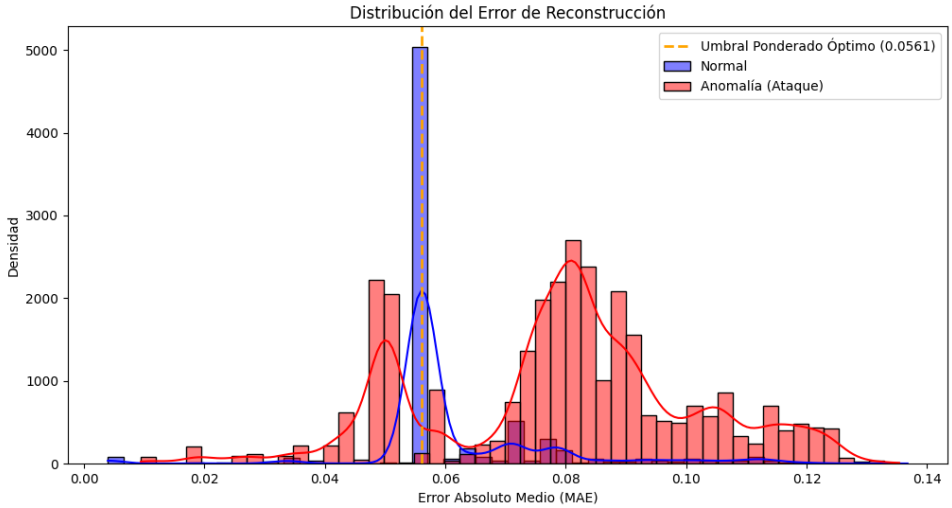
Tabla 1. Reporte de Clasificación Final del Modelo

Clase	Precisión	Recall	F1-Score	Soporte
Normal	0.46	0.73	0.56	7129
Anomalía	0.93	0.8	0.86	30576

Accuracy: 0.79

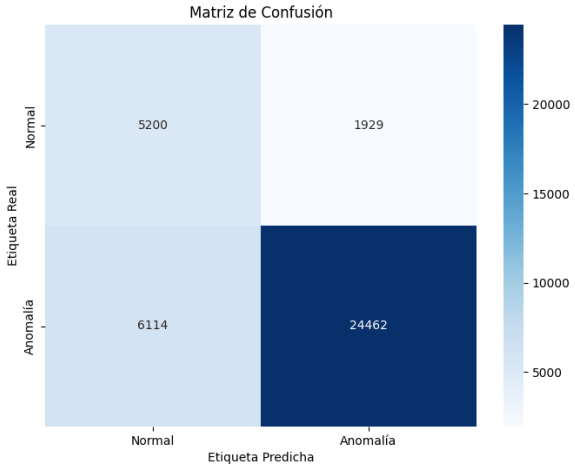
La Figura 1 muestra la distribución de los errores de reconstrucción para las secuencias normales y anómalas, donde se puede observar una clara separación entre ambas clases.

La línea vertical indica el umbral óptimo seleccionado.



(Figura 1. Distribución del Error de Reconstrucción)

La Figura 2 presenta la matriz de confusión, que visualiza el rendimiento detallado del clasificador, mostrando los verdaderos positivos, verdaderos negativos, falsos positivos y falsos negativos.



(Figura 2. Matriz de Confusión con Umbral Óptimo)

4. Discusión

Los resultados obtenidos validan la eficacia del framework propuesto. El modelo demuestra un rendimiento excepcional en su tarea principal: la detección de anomalías. Un Recall de 0.80 indica que el sistema es capaz de identificar 8 de cada 10 ataques reales, minimizando el riesgo de incidentes no detectados. A su vez, una Precisión de 0.93 asegura que las alertas generadas son altamente fiables, lo cual es fundamental para reducir la carga de trabajo de los analistas de seguridad en un entorno real.

El F1-Score de 0.86 para la clase "Anomalía" confirma que se ha alcanzado un excelente equilibrio entre sensibilidad y fiabilidad. Es importante analizar el rendimiento en la clase "Normal". Un Recall de 0.73 y un F1-Score de 0.56, aunque inferiores, son el resultado

directo de una calibración deliberada. En ciberseguridad, es preferible incurrir en un mayor número de falsos positivos (eventos normales clasificados como anómalos) que permitir falsos negativos (ataques no detectados). Este "trade-off" es una decisión estratégica para priorizar la seguridad, y los resultados demuestran que el modelo fue calibrado exitosamente bajo este principio.

A diferencia de otros trabajos que pueden incorporar tecnologías como blockchain para la integridad de los logs (Gudivaka et al., 2025; Islam et al., 2023), o centrarse en esquemas descentralizados de autenticación y autorización basados en identificadores descentralizados y compartición de secretos (Krishna et al., 2023), nuestro enfoque se centra en la eficiencia del modelo de detección en sí mismo, presentando una solución que puede ser implementada con una infraestructura computacional estándar.

Por su parte, Islam et al. (2023) proponen un marco de autenticación y control de acceso para datos de logs forenses basados en blockchain, cuyo objetivo principal es almacenar datos de logs de usuario en un almacenamiento a prueba de manipulaciones (tamper-proof) gracias a la inmutabilidad de la cadena de bloques

5. Conclusiones

Este estudio ha demostrado que un framework basado en un modelo LSTM Autoencoder es una herramienta potente y eficaz para la detección de patrones de ataques de fuerza bruta en logs de autenticación. La metodología presentada, que abarca desde el procesamiento de datos hasta la optimización ponderada del umbral de decisión, ha resultado en un sistema que no solo es preciso, sino que también está calibrado para un balance pragmático entre la máxima detección de amenazas y la viabilidad operativa.

La principal contribución del trabajo es la validación de un flujo de trabajo completo y reproducible que puede servir como base para el desarrollo de sistemas de monitoreo de seguridad inteligentes. Como limitaciones, el modelo fue entrenado y evaluado en un dataset específico, y su generalización a otros tipos de logs requeriría un re-entrenamiento. Futuras líneas de investigación podrían explorar arquitecturas más complejas, como Bi-LSTMs o la incorporación de mecanismos de atención, así como la integración de este framework en una plataforma de gestión de eventos e información de seguridad (SIEM) para su operación en tiempo real.

Referencias Bibliograficas

Bajramovic et al. (2023) Bajramovic, E., Fein, C., Frinken, M., Rösler, P., & Freiling, F. (2023). LAVA: Log Authentication and Verification Algorithm. *Digital Threats: Research and Practice*, *4*(3), Article 35. https://doi.org/10.1145/3609233

Flynn & Olukoya (2025) Flynn, R., & Olukoya, O. (2025). Using approximate matching and machine learning to uncover malicious activity in logs. *Computers & Security*, *151*, 104312. https://doi.org/10.1016/j.cose.2025.104312

Gudivaka et al. (2025) Gudivaka, B. R., Gudivaka, R. L., Gudivaka, R. K., Basani, D. K. R., Grandhi, S. H., Murugesan, S., & Kamruzzaman, M. M. (2025). A predominant intrusion detection system in IIoT using ELCG-DSA AND LWS-BiOLSTM with blockchain. *Sustainable*

Computing: Informatics and Systems, *46*, 101127. https://doi.org/10.1016/j.suscom.2025.101127

Islam, M. E., Islam, M. R., Chetty, M., Lim, S., & Chadhar, M. (2023). User authentication and access control to blockchain-based forensic log data. *EURASIP Journal on Information Security,* *2023*(7). https://www.scopus.com/pages/publications/85165706702?origin=resultslist

Krishna, D. P., Ramaguru, R., Praveen, K., Sethumadhavan, M., Ravichandran, K. S., Krishankumar, R., & Gandomi, A. H. (2023). SSH-DAuth: secret sharing based decentralized OAuth using decentralized identifier. *Scientific Reports,* *13*(18335). https://www.scopus.com/pages/publications/85175003812?origin=resultslist