**Modelo de inteligencia artificial para mejorar la detección de ataques de fuerza bruta en logs en Servidor Linux. 2025**

Logotipo

El contenido generado por IA puede ser incorrecto.

Jose Dario Menendez Acosta

Facultad de Ingeniería Eléctrica y Electrónica, Universidad Nacional de Ingeniería

CBS06 M Inteligencia Artificial II

Ing. Yury Oscar Tello

16 de Octubre de 2025

**Resumen**

El avance tecnológico en áreas como la computación en la nube, redes vehiculares e Internet de las Cosas (IoT) ha incrementado el volumen de datos transmitidos, facilitando ciberataques que comprometen la seguridad de sistemas distribuidos. En este contexto, los ataques de fuerza bruta representan una amenaza persistente, especialmente en servidores Linux, donde intentos repetidos de acceso no autorizado generan logs que pueden analizarse para detectar intrusiones. Este estudio propone un modelo de inteligencia artificial (IA) basado en machine learning (ML) para mejorar la detección de estos ataques en logs de autenticación, superando limitaciones de métodos tradicionales como los Sistemas de Detección de Intrusiones basados en firmas (SIDS) y anomalías (A-IDS), que fallan ante ataques nuevos o generan falsos positivos.

La revisión de literatura destaca el uso de técnicas avanzadas de ML y deep learning (DL), como XGBoost, RNNs y modelos híbridos con lógica difusa, que logran precisiones superiores al 99% en entornos IoT. La metodología involucra el preprocesamiento de logs de btmp y auth.log, ingeniería de características (e.g., frecuencia de intentos por IP, tasa de éxito histórica) y balanceo de clases con SMOTE. Se emplea XGBoost para clasificación supervisada e IA explicable (XAI) con SHAP para interpretar el modelo.

Los resultados muestran una exactitud global de 1.00 en un conjunto de prueba con 3,756 eventos, con precisión y recall perfectos para ataques y 0.88 para eventos normales. La feature engineering, particularmente la reputación IP, fue clave para el rendimiento, demostrando eficiencia computacional (entrenamiento en 0.05 segundos). Este modelo ofrece aplicaciones en SIEM e IPS para detección en tiempo real y análisis forense, enfatizando que la calidad de datos supera ajustes algorítmicos en ciberseguridad.

**Palabras clave**

Detección de intrusiones, Machine learning, XGBoost, IA explicable

1. **Introducción**

El avance acelerado en tecnologías como la computación en la nube, los sistemas de redes vehiculares y el Internet de las Cosas (IoT), ha resultado en un incremento masivo en la cantidad de información transmitida a través de las infraestructuras de comunicación. Este crecimiento ha sido aprovechado por los atacantes, quienes han intensificado sus esfuerzos para comprometer la seguridad de los sistemas de red (Kasongo, 2023).

La ciberseguridad es un componente crítico de las redes distribuidas, enfrentando intentos constantes de acceso no autorizado para robar, alterar, exponer o destruir datos (Sarantos et al., 2025). Las intrusiones en la red se consideran una de las amenazas de seguridad más peligrosas para las organizaciones en la actualidad (Kasongo, 2023). Dado que las soluciones de seguridad actuales a menudo proporcionan solo protección a corto plazo, la expansión continua de las tecnologías hace que mantener la ciberseguridad sea un desafío perpetuo (Sarantos et al., 2025). En este panorama, los Sistemas de Detección de Intrusiones (IDS) son implementados para asegurar los sistemas y redes informáticas, monitoreando el tráfico en busca de amenazas (Kasongo, 2023).

Una de las amenazas más persistentes y comunes es el ataque de fuerza bruta, que puede causar daños graves, especialmente en redes IoT (Otoom et al., 2023). Un ataque de fuerza bruta consiste en el envío de todos los valores posibles para contraseñas o cuentas en un intento de obtener acceso no autorizado a la información del sistema (Park et al., 2021). En el contexto de la infraestructura de TI, la mayoría de los logs generados por accesos no autorizados son causados por ataques de fuerza bruta de SSH (Secure Shell), los cuales pueden ser detectados a través del análisis de dichos registros (Park et al., 2021).

La motivación principal para emplear inteligencia artificial (IA) reside en la necesidad de superar las limitaciones de los métodos tradicionales de detección (Sarantos et al., 2025). Los IDS modernos ya aprovechan los modelos de aprendizaje automático (Machine Learning, ML) para clasificar la actividad de la red como benigna o maliciosa (Sarantos et al., 2025). Los enfoques basados en ML son esenciales para construir IDS inteligentes capaces de detectar intrusiones de fuerza bruta de manera automatizada y tomar medidas directas (Otoom et al., 2023).

Es crucial el uso de modelos avanzados, como los de aprendizaje profundo (Deep Learning, DL), una rama de ML inspirada en el funcionamiento de las neuronas del cerebro humano (Kasongo, 2023), ya que las técnicas convencionales se ven obstaculizadas de varias maneras (Sarantos et al., 2025):

1. Los IDS basados en firmas (Signature-based IDS - SIDS) dependen de patrones predefinidos y no pueden detectar ataques nuevos o de día cero.
2. Los sistemas basados en anomalías (Anomaly-based IDS - A-IDS) tienden a disparar demasiadas alarmas de falsos positivos, lo que los hace inconvenientes para los profesionales de seguridad.

Las técnicas de ML y DL se aplican para defenderse de los ataques de red mediante el aprendizaje de patrones de uso y registros de acceso (Park et al., 2021). Además, modelos híbridos que incorporan la lógica difusa (fuzzy logic) y ML han demostrado la capacidad de mejorar significativamente la precisión de detección (Mehmmod et al., 2025). Estos modelos pueden ofrecer una tasa de detección notable del 99% con considerablemente menos falsos positivos en comparación con los modelos tradicionales.

Por lo tanto, la implementación de un modelo de inteligencia artificial (IA) se propone como una solución fundamental y adaptable, diseñada para mejorar la precisión y la capacidad de generalización en la detección de ataques de fuerza bruta en Servidores Linux, superando las deficiencias observadas en los métodos existentes.

1. **Revisión de la literatura**

Para asegurar los sistemas y redes informáticas, se implementan los Sistemas de Detección de Intrusiones (IDS) (Kasongo, 2023; Sarantos et al., 2025). Un IDS es un artefacto (hardware o software) que monitorea el tráfico de una red en busca de amenazas (Kasongo, 2023; Sarantos et al., 2025). Los IDS se clasifican típicamente en tres categorías principales basadas en la técnica de detección que emplean: IDS basados en firmas (SIDS), IDS basados en anomalías (A-IDS) e IDS híbridos (Kasongo, 2023). Los SIDS, también conocidos como Detección Basada en Conocimiento o Detección de Mal Uso, detectan ataques basándose en patrones preexistentes (firmas) almacenados en una base de datos (Sarantos et al., 2025). La principal limitación de los SIDS es su incapacidad para detectar ataques nuevos o de día cero (Sarantos et al., 2025), ya que dependen de firmas predefinidas (Sarantos et al., 2025). En contraste, los A-IDS escanean la red para identificar patrones o comportamientos que se desvían de lo que se considera normal (Kasongo, 2023; Sarantos et al., 2025), un enfoque que intenta mitigar la desventaja de los SIDS (Sarantos et al., 2025). Sin embargo, los A-IDS tienden a disparar un alto número de alarmas de falsos positivos (Sarantos et al., 2025), lo que los hace ruidosos e inconvenientes para los profesionales de seguridad (Sarantos et al., 2025).

2.1. El Problema de la Fuerza Bruta en Servidores y Entornos IoT

En el contexto de la infraestructura de TI, la mayoría de los logs generados por intentos de acceso no autorizado son causados por ataques de fuerza bruta SSH (Secure Shell) (Park et al., 2021). SSH es un protocolo fundamental para el acceso remoto seguro (Park et al., 2021). Sin embargo, la permisión de solicitudes de acceso remoto desde cualquier origen puede exponer un sistema a amenazas (Park et al., 2021). Los ataques de fuerza bruta SSH generan logs que tienen el mismo patrón de mensaje, incluyendo frases como "SSH user failed to log in from [attacker’s IP address] on VTY0 due to IP restriction" (Park et al., 2021). Al analizar estos registros, es posible recolectar la dirección IP del atacante, que luego se utiliza para controlar el acceso, por ejemplo, mediante listas negras (blacklists) (Park et al., 2021).

2.2. Aprendizaje Automático (ML) y Aprendizaje Profundo (DL) para IDS

Los IDS modernos aprovechan los modelos de Aprendizaje Automático (ML) (Sarantos et al., 2025), que otorgan a los programas informáticos la capacidad de aprender de la experiencia (Kasongo, 2023), para clasificar la actividad de la red como benigna o maliciosa (Sarantos et al., 2025). Los enfoques basados en ML son necesarios para construir IDS inteligentes capaces de detectar intrusiones de fuerza bruta de manera automatizada (Otoom et al., 2023). Las técnicas de ML se aplican para defenderse de ataques de red mediante el aprendizaje de patrones de uso y registros de acceso (Park et al., 2021).

Las RNNs, y sus variantes modificadas como la Memoria a Largo Corto Plazo (LSTM) y la Unidad Recurrente Cerrada (GRU) (Kasongo, 2023), son cruciales porque tienen la capacidad de memorizar temporalmente estados previos y aplicarlos en el cálculo actual (Kasongo, 2023). El uso de algoritmos de selección de características, como el basado en XGBoost (Extreme Gradient Boosting) (Kasongo, 2023), en conjunto con RNNs, mejora el rendimiento de los IDS (Kasongo, 2023) al reducir la dimensión del espacio de características y la complejidad del modelo (Kasongo, 2023).

2.3. Desafíos de Generalización y Soluciones Avanzadas

Los IDS basados en Aprendizaje Supervisado (SL) son entrenados con conjuntos de datos etiquetados (Kasongo, 2023). Una limitación crítica de estos modelos es que su rendimiento se degrada significativamente cuando se implementan en un entorno de red diferente al que generó sus datos de entrenamiento (Sarantos et al., 2025). Este es el problema de la generalización de dominio (Sarantos et al., 2025). La construcción de conjuntos de datos etiquetados para cada dominio es intensiva en recursos y tiempo (Sarantos et al., 2025). El etiquetado de registros de datos es una tarea costosa que requiere el trabajo meticuloso de anotadores expertos (Sarantos et al., 2025).

1. **Metodología**
   1. Descripción y Preprocesamiento del Dataset

El conjunto de datos se construyó a partir de logs de autenticación de un servidor Linux, específicamente de los archivos btmp (intentos de inicio de sesión fallidos) y auth.log (eventos de autenticación de sshd y login).

El preprocesamiento incluyó:

* + 1. Parsing y Extracción Estructurada: Se usaron expresiones regulares (regex) para transformar el texto no estructurado en datos tabulares.
    2. Estandarización y Etiquetado: Las marcas de tiempo se normalizaron a UTC. Se generó una variable binaria (is\_attack), asignando 1 a eventos fallidos (ataque) y 0 a eventos exitosos (normal). El dataset resultante mostró un severo desbalance de clases. Técnicas de Inteligencia Artificial Empleadas
  1. Ingeniería de Características (Feature Engineering)

Este fue un paso crucial para proveer al modelo de información contextual y de comportamiento. Se diseñaron y calcularon ocho características clave:

|  |  |  |
| --- | --- | --- |
| Categoría | Característica Clave | Descripción |
| Frecuencia/Variedad | ip\_attempts\_5min | Frecuencia de intentos desde la misma IP en 5 minutos. |
| Frecuencia/Variedad | ip\_distinct\_users\_15min | Conteo de usuarios únicos intentados por la misma IP. |
| Comportamiento Temporal | time\_since\_last\_attempt\_ip | Diferencia temporal (en segundos) desde el evento anterior de la misma IP. |
| Reputación/Contexto | ip\_success\_rate | Tasa de éxito histórica y acumulativa de la dirección IP (Métrica de reputación dinámica). |
| Reputación/Contexto | is\_common\_user | Indicador si el usuario es un objetivo común (ej. root, admin). |

* 1. Técnicas de Inteligencia Artificial
* Algoritmo de ML: Se seleccionó XGBoost (Extreme Gradient Boosting) por su eficacia en datos tabulares y su alta eficiencia computacional.
* Tratamiento del Desbalance de Clases: Se aplicó la técnica SMOTE (Synthetic Minority Over-sampling Technique) exclusivamente sobre el conjunto de entrenamiento para generar instancias sintéticas de la clase minoritaria ("eventos normales") y mitigar el sesgo.
* Evaluación: El conjunto de datos se dividió 80% para entrenamiento y 20% para prueba, evaluándose con métricas estándar (Precisión, Recall, Exactitud) y midiendo la eficiencia computacional.
* IA Explicable (XAI): Se utilizó el framework SHAP (SHapley Additive exPlanations) para interpretar la "caja negra" del modelo, calculando la contribución marginal de cada característica a la predicción.

1. **Resultados y discusión**

El modelo final, entrenado con las 8 características y datos balanceados con SMOTE, demostró un rendimiento excepcional en un conjunto de prueba de 3,756 eventos.

* 1. Métricas de Evaluación
* Exactitud (Accuracy): Se alcanzó una exactitud global de 1.00.
* Métricas para Ataques (Clase 1): Precisión y Recall de 1.00. Esto indica que el modelo identificó todos los ataques reales y no generó falsas alarmas sobre eventos clasificados como ataques.
* Métricas para Eventos Normales (Clase 0): Precisión y Recall de 0.88. Esto representa un resultado sobresaliente para la clase minoritaria, logrando identificar correctamente 7 de los 8 eventos legítimos.
* Errores: La Matriz de Confusión mostró solo 2 errores en 3,756 predicciones: un Falso Negativo y un Falso Positivo.
* Eficiencia Computacional: El tiempo de entrenamiento fue de 0.05 segundos, lo que subraya su viabilidad para entornos de producción en tiempo real.
  1. Impacto de la Ingeniería de Características

La comparación de escenarios demostró que la adición de características de contexto y reputación fue el factor determinante para el rendimiento casi perfecto, más que el simple uso de SMOTE:

* Un modelo base con solo 2 características y sin balanceo tuvo un Accuracy de 0.54.
* La inclusión de características avanzadas de reputación (ip\_success\_rate) y ritmo (time\_since\_last\_attempt\_ip) en el Escenario 4 fue el factor clave para alcanzar un Accuracy de 1.00.
  1. Interpretación (IA Explicable - SHAP)

El análisis con SHAP reveló que la característica más influyente fue ip\_success\_rate. Este es el hallazgo principal del estudio: la reputación histórica de una dirección IP es el predictor más potente de su intención futura, superando las métricas de comportamiento instantáneo.

El análisis detallado con SHAP confirmó la lógica del modelo:

* ip\_success\_rate: Valores bajos (historial de fallos) empujan fuertemente la predicción hacia "Ataque".
* time\_since\_last\_attempt\_ip: Valores bajos (tiempo corto entre intentos) son identificados como comportamiento robótico y malicioso.
* ip\_attempts\_5min: Valores altos (alta frecuencia) son un claro indicador de ataque.

1. **Conclusiones**

El estudio concluye que el éxito del modelo no residió solo en la elección del algoritmo (XGBoost), sino en una metodología de ingeniería de características avanzada. La creación de variables que capturan la reputación histórica y el ritmo del comportamiento fue fundamental para el discernimiento de patrones complejos.

Se reafirmó el principio fundamental de que la calidad y riqueza de los datos son prioritarias sobre los ajustes algorítmicos. Si bien SMOTE corrigió el sesgo, su efectividad se maximizó solo después de enriquecer el dataset con características altamente informativas.

El análisis con XAI validó que el modelo aprendió una estrategia de detección coherente con el conocimiento experto en ciberseguridad, priorizando el contexto a largo plazo sobre el comportamiento instantáneo.

# Referencias

Mehmmod, A., Batool, K., Sajid, A., Alam, M. M., Su’ud, M. M., & Khan, I. U. (2025). ERBM: A machine learning-driven rule-based model for intrusion detection in IoT environments. Computers, Materials & Continua, 81(1), 1017–1036. https://doi.org/10.32604/cmc.2025.062971

Kasongo, S. M. (2023). A deep learning technique for intrusion detection system using a Recurrent Neural Networks based framework. Computer Communications, 195, 145–158. https://doi.org/10.1016/j.comcom.2022.12.010

Otoom, A. F., Eleisah, W., & Abdallah, E. E. (2023). Deep learning for accurate detection of brute force attacks on IoT networks. Procedia Computer Science, 215, 157–164. https://doi.org/10.1016/j.procs.2023.03.038

Park, J., Kim, J., Gupta, B. B., & Park, N. (2021). Network log-based SSH brute-force attack detection model. Complexity, 2021, Article ID 6617592. https://doi.org/10.32604/cmc.2021.015172

Sarantos, P., Violos, J., & Leivadeas, A. (2025). Enabling semi-supervised learning in intrusion detection systems. Journal of Parallel and Distributed Computing, 179, 27–40. https://doi.org/10.1016/j.jpdc.2024.105010