ABSTRAK

Keamanan jaringan komputer saat ini menghadapi tantangan besar akibat meningkatnya kompleksitas serangan siber, khususnya penyebaran malware yang memanipulasi struktur file melalui teknik obfuscation atau polymorphic. Sistem deteksi kesulitan mengenali ancaman yang telah dimodifikasi pada level *byte*, meskipun konten berbahaya di dalamnya tetap aktif. Penelitian ini menganalisis performa algoritma *fuzzy hashing* TLSH dan SSDeep dalam mendeteksi kemiripan payload TLS *Client Hello* yang mengalami perubahan parsial.

Metode eksperimen murni dengan pendekatan kuantitatif komparatif digunakan pada 1.193 entri data dari dataset CICIDS2017. Lima strategi mutasi (*NOP Injection*, *Cipher Shuffle*, *Extension Append*, *Null Pad*, dan *Byte Flip*) diterapkan pada 9 tingkat *mutation rate* (1% hingga 50%) dengan 5 kali trial per skenario, menghasilkan total 268.425 percobaan. Evaluasi performa dilakukan berdasarkan dua parameter utama: *robustness* yang diukur melalui nilai distance TLSH dan similarity score SSDeep, serta efisiensi komputasi yang diukur melalui waktu eksekusi dan penggunaan memori puncak.

Hasil menunjukkan SSDeep unggul dalam *robustness* dengan titik evasi >50%, namun memiliki kelemahan skor similarity 0 untuk empat strategi mutasi akibat pendekatan string-based comparison yang digunakan. Sementara itu, TLSH menunjukkan pola lebih informatif dengan menghasilkan nilai distance terukur pada semua strategi mutasi, namun mencapai titik evasi lebih cepat pada rate 10-20% dan fluktuatif pada perubahan ekstrem. Dalam efisiensi komputasi, TLSH unggul dengan waktu eksekusi rata-rata 171,8 ms per entri data (10,9% lebih cepat dari SSDeep) dan penggunaan memori 0,17 KB (2,35 kali lebih hemat). Kontribusi utama penelitian ini adalah rekomendasi teknis objektif bagi pengembang sistem deteksi intrusi dan administrator jaringan dalam memilih algoritma *fuzzy hashing* paling optimal sesuai skenario penggunaan, serta referensi akademis bagi pengembangan metode deteksi ancaman berbasis kesamaan file di masa depan.

Kata Kunci: TLSH, SSDeep, *Fuzzy Hashing*, *Robustness*, Keamanan Jaringan

ABSTRACT

Computer network security currently faces significant challenges due to the increasing complexity of cyber attacks, particularly the spread of malware that manipulates file structures through obfuscation or polymorphic techniques. Detection systems struggle to recognize threats that have been modified at the byte level, even though the malicious content within them remains active. This study analyzes the performance of TLSH and SSDeep fuzzy hashing algorithms in detecting similarity in TLS Client Hello payloads that have undergone partial changes.

A pure experimental method with a comparative quantitative approach was employed using 1,193 data entries from the CICIDS2017 dataset. Five mutation strategies (NOP Injection, Cipher Shuffle, Extension Append, Null Pad, and Byte Flip) were applied at 9 mutation rates (1% to 50%) with 5 trials per scenario, resulting in a total of 268,425 experiments. Performance evaluation was conducted based on two main parameters: robustness, measured through TLSH distance values and SSDeep similarity scores, and computational efficiency, measured through execution time and peak memory usage.

The results show that SSDeep excels in robustness with an evasion point >50%, but has a weakness with similarity scores of 0 for four mutation strategies due to its string-based comparison approach. Meanwhile, TLSH demonstrates a more informative pattern by producing measurable distance values across all mutation strategies, although it reaches the evasion point faster at 10-20% rate and fluctuates under extreme changes. In terms of computational efficiency, TLSH is superior with an average execution time of 171.8 ms per data entry (10.9% faster than SSDeep) and memory usage of 0.17 KB (2.35 times more efficient). The main contribution of this research is providing objective technical recommendations for intrusion detection system developers and network administrators in selecting the most optimal fuzzy hashing algorithm according to their usage scenarios, as well as serving as an academic reference for the future development of file similarity-based threat detection methods to enhance information infrastructure resilience against continuously evolving cyber attacks.

Keywords: *TLSH, SSDeep, Fuzzy Hashing, Robustness, Network Security*