**ISSN: 2962-3545**
**Prosiding SAINTEK**: Sains dan Teknologi Vol.5 No.1 Tahun 2026
*Call for papers* dan Seminar Nasional Sains dan Teknologi Ke-5 2026
Fakultas Teknik, Universitas Pelita Bangsa, Februari 2026

# Deteksi PE Ransomware Menggunakan *Shallow Learning*

## *Detection of PE Ransomware Using Shallow Learning*

**Iik Muhamad Malik Matin1, Zahra Azizah[2], Ihsan Alamal Ahmad[3]**
[1,2,3]Teknik Informatika dan Komputer, Fakultas Teknik, Politeknik Negeri Jakarta
[1]iik.muhamad.malik.matin@tik.pnj.ac.id, [2]zahra.azizah@tik.pnj.ac.id*,
ihsan.alamalahmad@mhsw.pnj.ac.id*

### *Abstract*

*Ransomware is one of the fastest-growing cybersecurity threats in the last decade. This type of attack not only causes financial losses but also disrupts public services and digital infrastructure. Early detection of ransomware activity is a major challenge due to its rapid and adaptive attack patterns. This study aims to implement a Shallow Learning method in detecting ransomware using the RanSAP dataset. This dataset contains storage access patterns from ransomware activity and normal (benign) applications. Four algorithms were used: Support Vector Machine (SVM), Random Forest (RF), Decision Tree, and Logistic Regression (LR). Evaluation was conducted using a confusion matrix to measure accuracy, precision, recall, and F1-score. Experimental results showed that the SVM model performed best with 95% accuracy, followed by RF with 93%, Decision Tree with 91%, and LR with 89%. This study demonstrates that Shallow Learning is quite effective in detecting ransomware behavior patterns.*

***Keywords**: Ransomware, Shallow Learning, RanSAP, Machine Learning, malware detection*

### Abstrak

Ransomware merupakan salah satu ancaman keamanan siber yang berkembang pesat dalam satu dekade terakhir. Serangan jenis ini tidak hanya mengakibatkan kerugian finansial, tetapi juga gangguan pada layanan publik dan infrastruktur digital. Deteksi dini terhadap aktivitas ransomware menjadi tantangan utama karena pola serangan yang cepat dan adaptif. Penelitian ini bertujuan untuk mengimplementasikan metode *Shallow Learning* dalam mendeteksi ransomware menggunakan dataset RanSAP. Dataset ini memuat pola akses penyimpanan dari aktivitas ransomware dan aplikasi normal (benign). Empat algoritma yang digunakan yaitu *Support Vector Machine* (SVM), *Random Forest* (RF), *Decision Tree* dan *Logistic Regression* (LR). Evaluasi dilakukan dengan confusion matrix untuk mengukur akurasi, presisi, *recall*, dan F1-*score*. Hasil eksperimen menunjukkan bahwa model SVM memiliki kinerja terbaik dengan akurasi 95%, diikuti RF dengan 93%, Desicion Tree 91% dan LR dengan 89%. Penelitian ini menunjukan bahwa *Shallow Learning* cukup efektif dalam mendeteksi pola perilaku ransomware.

**Kata kunci**: Ransomware, *Shallow Learning*, RanSAP, *Machine Learning*, deteksi malware.

## Pendahuluan

Ransomware merupakan jenis perangkat lunak berbahaya yang mengenkripsi data korban dan menuntut pembayaran tebusan agar akses dapat dipulihkan. Dalam lima tahun terakhir, frekuensi serangan ransomware meningkat pesat seiring dengan meningkatnya ketergantungan masyarakat terhadap sistem digital. Laporan Trend Micro [1] menunjukkan bahwa lebih dari 80% Perusahaan global mengalami percobaan serangan ransomware. Ancaman ini juga meluas ke sektor publik, termasuk lembaga pemerintahan dan pendidikan.

Di Indonesia, kasus ransomware terhadap Pusat Data Nasional (PDN) pada pertengahan tahun 2024 menjadi bukti konkret dampak destruktif serangan siber terhadap layanan publik. Serangan yang diduga dilakukan oleh kelompok LockBit 3.0 ini menyebabkan gangguan pada ratusan sistem pemerintah dan menyoroti lemahnya mekanisme cadangan data [2][3].

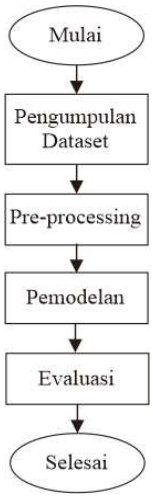
Berbagai pendekatan telah dilakukan untuk mendeteksi ransomware. Deteksi berbasis tanda tangan (*signature-based detection*) efektif untuk varian lama, tetapi gagal menghadapi varian baru yang melakukan obfuscation. Pendekatan berbasis perilaku (*behavior-based detection*) lebih menarik karena mengamati aktivitas sistem, seperti operasi *file*, proses, jaringan, dan penyimpanan. Namun metode *signature* maupun metode berbasis perilaku tidak lagi efisien [4]. *Machine Learning* merupakan bagian dari kecerdasan buatan yang memungkinkan komputer untuk belajar dari data dan membangun model prediktif. Algoritma pada *Machine Learning* dapat mempelajari pola-pola dan karakteristik sampel yang sudah dikenal untuk mengklasifikasikan instance ransomware yang baru sehingga dapat diidentifikasi berbahaya atau aman [5].

Beberapa penelitian telah dilakukan untuk mendeteksi ransomware. Pada ransomware android dilakukan oleh Benny dkk [6] menggunakan DNN untuk mendeteksi serangan ransomware dengan akurasi mencapai 96% dengan dataset CIC-InvesAndMal2019 [7]. Pada API dilakukan oleh shina sheen [8] Penelitian ini memanfaatkan panggilan API ransomware yang diperoleh dari proses analisis statis malware. Hasil ekstraksi panggilan API tersebut kemudian digunakan sebagai input pada beberapa algoritma pembelajaran mesin untuk proses training dan pengujian. Berdasarkan hasil penelitian, algoritma *Random Forest* menunjukkan performa terbaik dengan tingkat akurasi mencapai 99%. Penelitian berakitan dengan API dilakukan oleh [9], [10], [11] dengan menerapkan algoritma LSTM, ANN, TextCNN, dan DNN untuk mempelajari pola panggilan API dari berbagai sampel ransomware. Hasil dari penerapan metode-metode ini menunjukkan tingkat akurasi rata-rata sebesar 98%.

Berbeda dengan literatur sebelumnya, penelitian ini mengusulkan pendekatan deteksi ransomware menggunakan metode *Shallow Learning* dengan dataset RanSAP. Algoritma yang digunakan pada penelitian ini terdiri dari SVM, *Random Forest, Decision Tree* dan *Logistic Regression*.

## Metode Penelitian

Tahapan penelitian ini digambarkan pada gambar 1.




Gambar 1 Tahapan Penelitian

### Pengumpulan Dataset

Dataset RanSAP (*Ransomware Storage Access Patterns*) dikembangkan oleh Hirano, Hodota, dan Kobayashi [12]. Dataset ini berisi kumpulan data aktivitas baca-tulis (I/O) pada tingkat blok penyimpanan dari tujuh varian ransomware dan lima program benign. Setiap entri data mencatat waktu dalam detik dan nanodetik, alamat blok logis (Logical Block Address/LBA), ukuran operasi, serta nilai entropi tulis. Atribut entropi ini menunjukan tingkat keacakan data yang ransomware biasanya menulis data terenkripsi dengan entropi tinggi. Tabel 1 menunjukan varian data pada dataset RanSAP

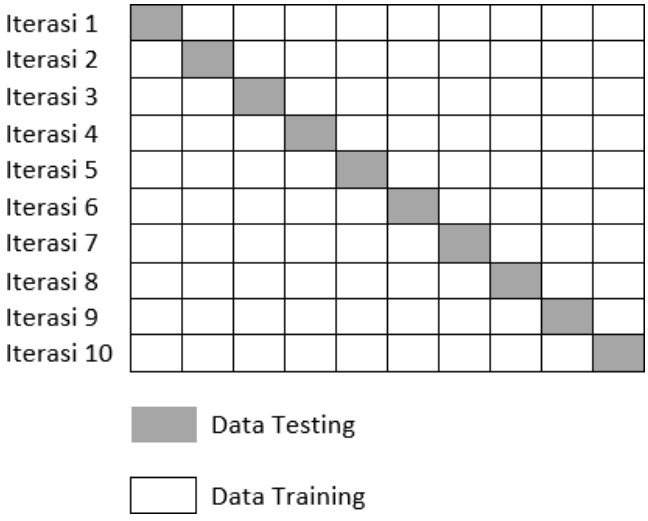
Tabel 1 Variasi Data RanSAP

| Ransomware | Jinak |
|---|---|
| TeslaCrypt | AESCrypt |
| Cerber | Sdelete |
| WannaCry | Zip |
| GandCrab v4 | Excel |
| Ryuk | Firefox |
| Sodinokibi | |
| Darkside | |

Dalam penelitian ini, data digunakan untuk melatih dan menguji model *Machine Learning*. Setiap sampel aktivitas dikonversi menjadi vektor fitur yang menggambarkan karakteristik pola baca-tulis. Fitur yang digunakan antara lain jumlah operasi baca/tulis, rata-rata ukuran blok, rasio baca terhadap tulis, serta nilai entropi rata-rata per sesi. Data kemudian dinormalisasi menggunakan *Min-Max Scaler* agar semua fitur berada dalam rentang [0,1].

**Pre-processing**

Langkah pra-pemrosesan juga meliputi penghapusan *outlier* dan data duplikat agar data tetap konsisten. Dataset dibagi menjadi data pelatihan 80% dan data pengujian 20% dengan metode *stratified sampling*, agar proporsi data ransomware dan benign tetap seimbang. *Cross validation* dilakukan dengan nilai CV=10. Pembagian dataset pada cross validation dapat digambarkan pada gambar 2.




Gambar 2 *cross validation*

**Model**

Penelitian ini menggunakan tiga algoritma *Shallow Learning*: *Support Vector Machine* (SVM), *Random Forest* (RF), dan *Logistic Regression* (LR). Ketiganya dipilih karena representatif dalam menggambarkan pendekatan berbasis margin, ansambel, dan linear probabilistik.

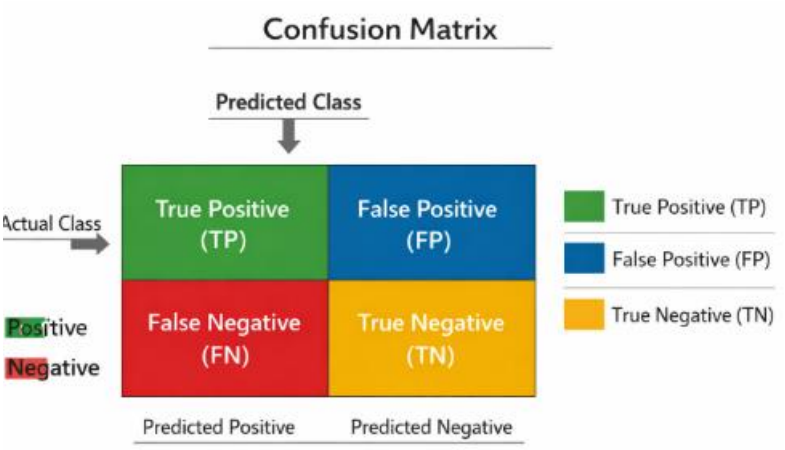
*Support Vector Machine* (SVM) bekerja dengan mencari *hyperplane* optimal yang memisahkan dua kelas data. Prinsip dasarnya adalah memaksimalkan jarak antar kelas (margin) agar model memiliki kemampuan generalisasi yang baik. Kernel digunakan untuk memetakan data nonlinear ke ruang berdimensi lebih tinggi. Pada penelitian ini digunakan kernel Radial Basis Function (RBF) karena kemampuannya mengatasi distribusi data non-linear yang sering ditemukan pada pola ransomware.

*Random Forest* (RF) merupakan algoritma ansambel yang terdiri dari sejumlah pohon keputusan independen [14],[15]. Setiap pohon dibangun dari subset acak data dan fitur, kemudian hasil prediksi dikombinasikan dengan majority voting. Keunggulan RF terletak pada ketahanannya terhadap *overfitting* dan kemampuannya menangani data dengan banyak variabel. Dalam konteks deteksi ransomware, RF efektif karena dapat mengenali pola interaksi kompleks antar fitur seperti ukuran blok dan entropi tulis.

*Logistic Regression* (LR) digunakan sebagai model dasar yang mengukur hubungan linear antara fitur dan probabilitas kelas . Fungsi sigmoid digunakan untuk memetakan hasil prediksi ke rentang 0–1. Meskipun sederhana, LR banyak digunakan karena interpretabilitasnya yang tinggi dan waktu pelatihan yang cepat. Dalam penelitian ini, LR digunakan sebagai baseline untuk membandingkan kinerja model yang lebih kompleks.

**Metode pengukuran**

valuasi model dilakukan dengan confusion matrix untuk menghitung empat metrik utama akurasi, presisi, *recall*, dan F1-*score*. Empat metrik utama didasarkan pada confusion matrix yang digambarkan pada Gambar 3.




Gambar 3. *Confusion Matrix*

model klasifikasi. Metrik ini mengukur proporsi *instance* yang diklasifikasikan secara benar terhadap keseluruhan jumlah instance. Tingkat akurasi yang tinggi mencerminkan kemampuan model dalam menghasilkan prediksi yang tepat. Perhitungan akurasi dilakukan dengan menggunakan rumus (1).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Presisi merupakan metrik evaluasi kinerja lain yang digunakan dalam konteks klasifikasi, khususnya ketika menghadapi dataset yang tidak seimbang. Nilai presisi yang tinggi menunjukkan bahwa model memiliki tingkat kesalahan yang rendah dalam memprediksi instance positif. Perhitungan presisi dapat dilakukan menggunakan rumus berikut, di mana TP melambangkan jumlah *true positive*, TN jumlah *true negative*, dan FP jumlah *false positif*. Perhitungan Presisi dilakukan dengan menggunakan rumus (2).

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

*Recall* (sensitivitas) mengukur kemampuan model dalam mendeteksi instance positif, yaitu serangan ransomware, dengan benar. Metrik ini dihitung sebagai rasio prediksi true positive terhadap total jumlah serangan ransomware yang sebenarnya dalam dataset. *Recall* menjadi sangat penting dalam konteks keamanan siber, karena mendeteksi semua serangan ransomware secara akurat sangat krusial untuk mencegah kerugian sistem. Nilai *recall* yang tinggi menunjukkan bahwa model mampu menangkap sebagian besar serangan ransomware secara efektif. Perhitungan *Recall* dilakukan dengan menggunakan rumus (3).

$$\text{Recall} = \frac{TP}{TP+FN} \quad (3)$$

F1-*score* merupakan rata-rata dari presisi dan *recall*, yang berfungsi untuk menyeimbangkan kinerja model. Metrik ini sangat berguna ketika terdapat ketidakseimbangan signifikan antara kelas negatif dan positif dalam dataset. F1-*score* menjadi metrik penting dalam penelitian ini, mengingat fokusnya pada deteksi ransomware, di mana identifikasi yang akurat terhadap seluruh instance false positive maupun false negative sangat krusial. Perhitungan F1-*Score* dilakukan dengan menggunakan rumus (4).

$$\text{F1} - \text{Score} = \frac{2 \times Precission \times Recall}{Precission \times Recall} \quad (4)$$

## Hasil dan Pembahasan

Hasil klasifikasi model *Machine Learning* dapat ditunjukan pada Tabel 2.

Tabel 2. hasil Pemodelan

| Model | Matriks Pengukuran (%) | | | |
|---|---|---|---|---|
| | **Akurasi** | **Presisi** | ***Recall*** | **F1-*Score*** |
| **SVM** | 95 | 96 | 94 | 95 |
| **Random Forest** | 93 | 92 | 94 | 93 |
| **Decision Tree** | 91 | 90 | 91 | 90 |
| **Logistic Regression** | 89 | 90 | 88 | 89 |

Model *Support Vector Machine* (SVM) memberikan performa paling baik dalam mendeteksi ransomware. Dengan akurasi 95%, presisi 96%, *recall* 94%, dan F1-*score* 95%, SVM menunjukan paling baik dalam mengidentifikasi ransomware tanpa banyak kesalahan klasifikasi. Nilai presisi yang menandakan model jarang salah mendeteksi file aman sebagai ancaman atau false positif, sementara *recall* yang juga tinggi memastikan sebagian besar file ransomware berhasil terdeteksi.

Model *Random Forest* juga menunjukkan performa kuat dengan akurasi 0.93, presisi 0.92, *recall* 0.94, dan F1-*score* 0.93. Nilai *recall* yang tinggi menunjukkan kemampuannya dalam mendeteksi hampir semua ransomware, meskipun presisinya sedikit lebih rendah dibandingkan SVM. Artinya, *Random Forest* lebih berhati-hati dalam klasifikasi, cenderung memberikan beberapa false positive untuk memastikan tidak ada ancaman yang terlewat. Model ini cocok digunakan dalam sistem keamanan yang menekankan early detection, di mana sedikit kesalahan positif masih dapat diterima dibanding risiko lolosnya ransomware sebenarnya.

Model *Decision Tree* memiliki performa cukup baik dengan akurasi 91%, presisi 90%, *recall* 91%, dan F1-*score* 90%. Model ini mampu memberikan hasil yang kompetitif dan mudah diinterpretasikan, karena setiap keputusan klasifikasi dapat dijelaskan melalui struktur pohon keputusan. Namun, dibandingkan dengan *Random Forest* atau SVM, *Decision Tree* cenderung lebih rentan terhadap overfitting, terutama pada data ransomware yang kompleks dan bervariasi. Meski demikian, model ini tetap relevan untuk digunakan pada sistem deteksi dengan kebutuhan interpretabilitas tinggi atau sumber daya komputasi terbatas.

Model *Logistic Regression* menghasilkan performa paling rendah dengan akurasi 89%, presisi 90%, *recall* 88%, dan F1-*score* 89%. Meskipun hasilnya masih dalam kisaran yang baik, *Logistic Regression* memiliki keterbatasan dalam menangani pola data non-linear yang umum ditemukan pada perilaku ransomware. Hal ini membuat model kurang efektif dalam mengenali varian baru yang memiliki karakteristik kompleks. Namun karena kesederhanaan dan efisiensinya, *Logistic Regression* tetap bermanfaat sebagai model dasar atau pembanding dalam eksperimen deteksi malware.

Secara keseluruhan, SVM merupakan model terbaik untuk deteksi ransomware. Berdasarkan nilai performa yang ditunjukan oleh Tabel 2, model ini mampu mendeteksi ransomware secara tanpa banyak kesalahan klasifikasi.

## Kesimpulan

Penelitian ini menggunakan metode *Shallow Learning* untuk mendeteksi ransomware berbasis PE. Empat algoritma *Machine Learning* digunakan, yaitu *Support Vector Machine*, Random Forest, Decision Tree, dan Logistic Regression, dengan evaluasi berdasarkan metrik akurasi, presisi, *recall*, dan F1-*score*.

Hasil eksperimen menunjukkan bahwa *Support Vector Machine* memberikan kinerja terbaik dengan akurasi sebesar 95%, diikuti oleh *Random Forest* sebesar 93%, *Decision Tree* sebesar 91%, dan *Logistic Regression* sebesar 89%. Kinerja SVM menunjukkan bahwa pendekatan berbasis margin efektif dalam membedakan pola perilaku ransomware dan benign berdasarkan karakteristik akses penyimpanan. Sementara itu, *Random Forest* juga menunjukkan performa yang baik, terutama dalam hal kemampuan mendeteksi sebagian besar serangan ransomware.

Berdasarkan hasil tersebut, dapat disimpulkan bahwa metode *Shallow Learning* cukup efektif untuk deteksi ransomware dengan komputasi yang relatif rendah, sehingga dapat diterapkan pada sistem dengan keterbatasan sumber daya. Penelitian selanjutnya dapat dikembangkan dengan memperluas variasi dataset, menambahkan fitur perilaku lain, serta membandingkan kinerja *Shallow Learning* dengan pendekatan deep learning untuk memperoleh model deteksi ransomware yang lebih *robust*.

## Ucapan Terima Kasih

Penelitian ini dapat terlaksana berkat dukungan pendanaan dari Politeknik Negeri Jakarta melalui Hibah Penelitian Asisten Ahli Tahun 2025 dengan Nomor SPK 258/PL3.A.10/PT.00.06/2025.

## Daftar Rujukan

[1] Trend Micro, "Trend 2025 Cyber Risk Report," 2025.

[2] A. Laras, "Begini Serangan Ransomware BSI Tahun Lalu, Mirip dengan Penyebab PDN Down?," Bisnis.com, 2024. [Online]. Available: https://finansial.bisnis.com/read/20240627/90/1777564/begini-serangan-ransomware-bsitahun-lalu-mirip-dengan-penyebab-pdn-down

[3] S. Mashabi and A. P. Kasih, "PDN Diretas, Bagaimana Nasib Data Penerima Beasiswa di Kemendikbud?," Kompas.com. Accessed: Feb. 15, 2025. [Online]. Available: https://www.kompas.com/edu/read/2024/06/30/093207471/pdn-diretas-bagaimana-nasibdata-penerima-beasiswa-di-kemendikbud

[4] D. Prayitno, "Systematic Literature Review: Implementasi Metode Statis Dan Dinamis Pada Analisa Malware," Simetris, vol. 16, no. 2, pp. 53–57, 2022, [Online]. Available: https://www.sttrcepu.ac.id/jurnal/index.php/simetris/article/view/255%0Ahttps://www.sttrcepu.ac.id/jurnal/index.php/simetris/article/download/255/165

[5] K. Liu, S. Xu, G. Xu, M. Zhang, D. Sun, and H. Liu, "A Review of Android Malware Detection Approaches Based on *Machine Learning*," IEEE Access, vol. 8, pp. 124579–124607, 2020, doi: 10.1109/ACCESS.2020.3006143.

[6] B. Purnama, E. A. Winarto, S. Shairuppdin, I. S. Wijaya, and I. S. Wijaya, "Deteksi Malware. Ransomware Menggunakan Deep Neural Network," *J. Edukasi dan Penelit. Inform.*, vol. 10, no. 1, p. 8, 2024, doi: 10.26418/jp.v10i1.68492.

[7] L. Taheri, A. F. A. Kadir, and A. H. Lashkari, "Extensible android malware detection and family classification using network-flows and API-calls," *Proc. - Int. Carnahan Conf. Secur. Technol.*, vol. 2019-October, no. Cic, 2019, doi: 10.1109/CCST.2019.8888430.

[8] S. Sheen and A. Yadav, "Ransomware detection by mining API call usage," *2018 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2018*, pp. 983–987, 2018, doi: 10.1109/ICACCI.2018.8554938.

[9] B. Qin, Y. Wang, and C. Ma, "API Call Based Ransomware Dynamic Detection Approach Using TextCNN," *Proc. - 2020 Int. Conf. Big Data, Artif. Intell. Internet Things Eng. ICBAIE 2020*, pp. 162–166, 2020, doi: 10.1109/ICBAIE49996.2020.00041.

[10] A. Ashraf, A. Aziz, U. Zahoora, and A. Khan, "Ransomware Analysis using Feature Engineering and Deep Neural Networks," pp. 1–15, 2019, [Online]. Available: http://arxiv.org/abs/1910.00286

[11] Y. A. Ahmed, B. Koçer, and B. A. S. Al-Rimy, "Automated Analysis Approach for the Detection of High Survivable Ransomware," *KSII Trans. Internet Inf. Syst.*, vol. 14, no. 5, pp. 2236–2257, 2020, doi: 10.3837/tiis.2020.05.021.

[12] M. Hirano, R. Hodota, and R. Kobayashi, "RanSAP: An open dataset of ransomware storage access patterns for training *Machine Learning* models," *Forensic Sci. Int. Digit. Investig.*, vol. 40, p. 301314, 2022, doi: 10.1016/j.fsidi.2021.301314.

[13] I. M. M. Matin, M. Agustin, B. Sugiarto, and A. N. Asri, "Malware Detection Using *Machine Learning* With Ensemble Method," *Pros. Sains Nas. dan Teknol.*, vol. 13, no. 1, p. 265, 2023.

[14] L. Breiman, "Random Forests," *Mach. Learn.*, vol. 45, pp. 5–32, 2001, doi: 10.1109/ICCECE51280.2021.9342376.

[15] Prasetyo and E. S. Nugraha, "Implementasi Random Forest untuk Klasifikasi Serangan pada Database Server," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 9, no. 2, pp. 201-210, 2025, doi: 10.29207/resti.v9i2.6789.