Terakreditasi Peringkat 4
Keputusan Direktur Jenderal Pendidikan Tinggi, Riset, dan Teknologi, No. 225/E/KPT/2022
Masa berlaku dari Volume 8 Nomor 1 Tahun 2022 sampai dengan Volume 12 Nomor 2 Tahun 2026

Published online at: **https://ejournal.raharja.ac.id/**

JOURNAL CERITA:
Creative Education of Research in Information Technology and Artificial informatics

Vol. 11 No. 1 (2025) 64-72 e-ISSN: 2655 - 2574

Penerapan PSO-SVM Untuk Deteksi Serangan *Web* Dengan Pendekatan *Hybrid Anomaly-Signature Based*

Novandi Kevin Pratama[1], Achmad Junaidi[*2], Afina Lina Nurlaili[3]
[1,2,3]Informatika, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional "Veteran" Jawa Timur, Surabaya, Indonesia
Email: [1]novandikevinp@gmail.com; [*2]**achmadjunaidi.if@upnjatim.ac.id**; [3]afina.lina.if@upnjatim.ac.id

Pratama, N, K., Junaidi, A., & Nurlaili, A, L., (2025). Penerapan PSO-SVM Untuk Deteksi Serangan Web Dengan Pendekatan Hybrid Anomaly-Signature Based. *Journal Cerita: Creative Education of Research in Information Technology and Artificial Informatics*, 11(1), 64-72

DOI: https://doi.org/10.33050/cerita.v11i1.3497

ABSTRAK

Keamanan aplikasi web menjadi semakin penting seiring dengan meningkatnya penggunaan web dalam pendidikan dan bisnis, terutama karena adanya pengelolaan data sensitif. Serangan seperti *SQL Injection* sering kali menjadi ancaman serius terhadap integritas data, dengan mengeksploitasi kelemahan dalam validasi input. Pendekatan berbasis *signature* digunakan untuk mendeteksi serangan yang sudah dikenal, namun sering kali tidak efektif terhadap serangan baru. Di sisi lain, pendekatan *anomaly* berbasis pembelajaran mesin mampu mengidentifikasi pola anomali, tetapi biasanya lambat untuk deteksi secara *real-time*. Penelitian ini menerapkan PSO-SVM *(Particle Swarm Optimization-Support Vector Machine*) untuk meningkatkan deteksi serangan pada aplikasi web dengan menggabungkan pendekatan *anomaly* dan *signature*. PSO digunakan untuk mengoptimalkan parameter SVM, dengan tujuan meningkatkan akurasi dalam mendeteksi serangan baru dan mengurangi jumlah serangan yang tidak terdeteksi. Evaluasi melalui skenario pengujian menunjukkan peningkatan akurasi hingga 99,3%, yang menegaskan bahwa pendekatan *hybrid* ini efektif dalam meningkatkan keamanan aplikasi web.

Kata kunci: Keamanan Aplikasi Web, PSO-SVM, Deteksi Serangan, *Anomaly Based*, *Signature Based.*

Received: 23-08-2024 | Accepted: 10-09-2024 | Published Online: 08-02-2025

ABSTRACT

The security of web applications is becoming increasingly crucial with the growing use of web platforms in education and business, especially due to the management of sensitive data. Attacks such as SQL Injection often pose serious threats to data integrity by exploiting weaknesses in input validation. Signature-based approaches are employed to detect known attacks, but they are often ineffective against new threats. On the other hand, anomaly-based approaches using Machine Learning can identify anomalous patterns but are typically slow for real-time detection. This study implements PSO-SVM (Particle Swarm Optimization-Support Vector Machine) to enhance the detection of attacks on web applications by combining anomaly and signature-based approaches. PSO is utilized to optimize SVM parameters, aiming to improve the accuracy of detecting new attacks and reduce the number of undetected threats. Evaluation through testing scenarios demonstrated an accuracy improvement of up to 99.3%, confirming that this hybrid approach is effective in enhancing the security of web applications.

Keywords: *Web Application Security, PSO-SVM, Attack Detection, Anomaly-Based, Signature-Based.*

I. PENDAHULUAN

Penggunaan aplikasi web telah menjadi bagian integral dari kehidupan sehari-hari, terutama di bidang pendidikan dan bisnis. Dalam pendidikan, aplikasi web memungkinkan siswa untuk belajar secara daring, mengikuti diskusi, dan menyelesaikan tugas dari mana saja. Di sektor bisnis, aplikasi web memudahkan komunikasi antar tim dan manajemen proyek. Fenomena ini mencerminkan perubahan dalam cara kita belajar dan bekerja sama, dengan akses yang lebih mudah dan fleksibel terhadap informasi serta peningkatan kolaborasi global. Seiring dengan kemajuan teknologi dan meningkatnya kebutuhan akan keterlibatan digital, penggunaan aplikasi web diprediksi akan terus tumbuh sebagai elemen vital dalam kehidupan sehari-hari.

Keamanan aplikasi web menjadi perhatian utama, hal ini disebabkan meningkatnya jumlah pengguna aplikasi web yang menyebabkan banyaknya serangan yang dapat mengambil data sensitif pengguna yang tidak dilindungi secara maksimal. Sebagai contoh serangan seperti *SQL Injection*, memanfaatkan celah keamanan dalam aplikasi web yang tidak memvalidasi atau menyaring input pengguna dengan benar, memungkinkan penyerang untuk memasukkan perintah SQL untuk mengakses dan memodifikasi data pengguna (Soesanto et al., 2023). Untuk memperkuat perlindungan terhadap serangan-serangan tersebut, Web Application Firewall (WAF) menjadi solusi yang efektif. WAF berfungsi sebagai penyaring antara pengguna dan aplikasi web, menganalisis setiap permintaan dan respons HTTP untuk mendeteksi serta mencegah potensi serangan yang dapat merusak. WAF (Web Application Firewall) mendeteksi serangan melalui dua pendekatan utama: berbasis *signature* dan *anomaly*. Pendekatan berbasis *signature* membandingkan permintaan HTTP dengan pola serangan yang telah dikenal, namun kurang efektif terhadap serangan baru, memerlukan pembaruan rutin (Calvo & Beltrán, 2022). Sebaliknya, pendekatan berbasis *anomaly* menggunakan pembelajaran mesin untuk mengidentifikasi pola yang mencurigakan, tetapi cenderung lambat untuk deteksi *real-time* (Tekerek & Bay, 2019). Menggabungkan kedua pendekatan ini menjadi solusi yang lebih efektif, di mana deteksi awal dilakukan dengan pendekatan berbasis *signature*, dan jika pola tidak dikenali, dilanjutkan dengan pendekatan berbasis *anomaly*. Hasil *anomaly* ini juga dapat memperbarui basis data *signature*, sehingga WAF lebih adaptif dalam menghadapi serangan baru.

Pendekatan *hybrid* yang menggabungkan metode berbasis *signature* dan *anomaly* dengan PSO-SVM dipilih untuk deteksi serangan melalui HTTP. Kombinasi ini memanfaatkan kekuatan *signature* untuk pola yang dikenal dan *anomaly* untuk pola yang tidak biasa, memungkinkan identifikasi serangan baru. PSO digunakan untuk mengoptimalkan parameter SVM, meningkatkan akurasi klasifikasi antara data normal dan serangan. Dengan evaluasi melalui berbagai skenario, penelitian ini diharapkan meningkatkan deteksi serangan serta kinerja keseluruhan sistem keamanan.

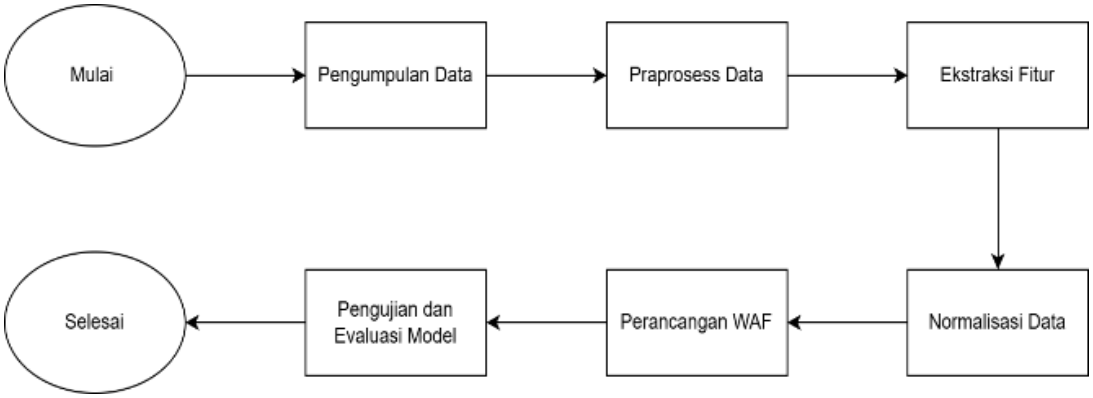
Pada penelitian ini digunakan dua jenis dataset yang berbeda dari segi karakteristik dan jenis format data untuk mendukung didapatnya

hasil terbaik dalam skenario pengujian yaitu CSIC 2010 dan ECML/PKDD 2007. Dataset CSIC 2010 berbentuk data TXT serta sebaliknya, ECML/PKDD 2007 berbentuk data XML.

II. METODE PENELITIAN

A. Tahapan Penelitian

Tahapan penelitian untuk Penerapan PSO-SVM Untuk Deteksi Serangan Web Dengan Pendekatan *Hybrid Anomaly-Signature Based* dilakukan dengan enam langkah sistematis, yang disajikan pada gambar 1.



Gambar 1. Tahapan Penelitian
Sumber: diolah dari data primer

B. Pengumpulan Data

Pengumpulan data dalam penelitian ini melibatkan dua dataset utama, yaitu ECML PKDD 2007 dan CSIC 2010, yang diunduh dari repositori *GitHub* milik Matthew Sudol pada 28 April 2020. Dataset ECML PKDD 2007 terdiri dari sampel permintaan HTTP dalam format XML, mencakup permintaan normal serta berbagai serangan seperti *Cross-Site Scripting* (XSS), *SQL Injection*, *Path Traversal*, dan lainnya. Setiap sampel menyertakan informasi detail seperti metode, protokol, URI, *query*, *header*, dan *body*. Sementara itu, dataset CSIC 2010 berisi kumpulan lalu lintas yang dihasilkan dari aplikasi web *e-commerce* yang dikembangkan di departemen penelitian, dengan data disimpan dalam format TXT. Dataset ini mencakup lalu lintas normal serta anomali, termasuk serangan seperti *SQL Injection*, *Buffer Overflow*, dan lainnya. Dalam penelitian ini, data dari kedua dataset tersebut akan diklasifikasikan menjadi dua kelas utama, yaitu anomali dan normal.

a) SQL Injection

SQL Injection adalah kerentanan keamanan yang muncul ketika penyerang dapat mengubah perintah SQL standar dengan menyisipkan atau menambahkan kode SQL yang berbahaya (Tasevski & Jakimoski, 2020). Kerentanan ini biasanya terjadi pada aplikasi web yang mengolah input pengguna untuk membuat perintah SQL tanpa melakukan validasi atau sanitasi yang memadai.

b) Buffer Overflow

Buffer Overflow adalah masalah keamanan penting yang telah menjadi perhatian selama lebih dari satu dekade. Masalah ini sering menyebabkan kerusakan pada program, korupsi data, dan pelanggaran keamanan(Wolfson et al., 2023). Kerentanan ini terjadi ketika sebuah program menulis lebih banyak data ke dalam *buffer* daripada kapasitas yang tersedia.

c) Cross-Site Scripting

Cross-Site Scripting (XSS) adalah jenis kerentanan keamanan yang terjadi pada aplikasi web yang tidak dapat memfilter input pengguna yang berisi kode berbahaya. Kode tersebut bisa ditulis dalam berbagai bahasa pemrograman, seperti *JavaScript* atau HTML(Nagarjun & Ahamad, 2020).

d) Path Traversal

Path Traversal adalah kerentanan keamanan yang terjadi ketika aplikasi web menerima input pengguna untuk mengakses *file* dan *direktori* yang berada di luar folder *root* web(Liu et al., 2023). Penyerang memanfaatkan kerentanan *path traversal* untuk mengakses *file* dan *direktori* di sistem *file* server web, termasuk kode sumber aplikasi, data, dan *file* konfigurasi. Hal ini dapat mengakibatkan akses tidak sah dan pengungkapan informasi sensitif.

C. Pra-proses Data

Pada tahap pra-proses, fokus utama adalah menyamakan format antar setiap dataset menjadi format yang sama. Pada penelitian ini, kedua dataset dikonversi menjadi format JSON. Proses transformasi ini dilakukan untuk mempermudah pengolahan dan analisis data pada tahap-tahap selanjutnya, memastikan data siap digunakan dalam model pembelajaran mesin. Setelah proses konversi JSON selesai, data yang telah diubah ke dalam format yang lebih mudah dikelola ini siap untuk tahap pemodelan lebih lanjut, di mana data akan digunakan untuk melatih model PSO-SVM dalam mendeteksi serangan pada aplikasi web. Proses pelatihan melibatkan pengoptimalan parameter model untuk meningkatkan akurasi deteksi dan memastikan bahwa model dapat mengenali pola-pola serangan dengan lebih efektif, termasuk serangan yang belum pernah dikenali sebelumnya.

a) JSON Parsing

JSON parsing adalah proses mengonversi data dalam format JSON (*JavaScript Object Notation*) menjadi objek yang dapat digunakan oleh program atau aplikasi(Dann et al., 2022). Proses ini melibatkan pembacaan *string* JSON dan transformasinya menjadi struktur data yang dapat diakses dan dimanipulasi dalam bahasa pemrograman yang digunakan.

D. Ekstraksi Fitur

Pada tahap ekstraksi fitur, penelitian ini mengacu pada studi yang dilakukan oleh Nguyen dkk. sebagai dasar untuk menentukan fitur-fitur yang akan digunakan. Setelah proses *parsing* selesai, penelitian ini memanfaatkan 28 fitur yang dipilih untuk analisis lebih lanjut. Fitur tersebut dapat dilihat pada Tabel 1.

Tabel 1. Fitur yang digunakan

Nama Fitur	**Nama Fitur**
Panjang permintaan HTTP	Panjang huruf pada *query*
Panjang *query*	Jumlah *body*
Panjang *Header* '*Accept-Encoding*'	Panjang angka pada *body*
Panjang *Header* '*Accept-Language*'	Panjang karakter spesial pada *body*
Panjang *Header* '*Content-Length*'	Panjang huruf pada *body*
Panjang *Header* '*User-Agent*'	Panjang *path*
Panjang *Header* '*Cookie*'	Panjang angka pada *path*
Panjang *Header* '*Accept*'	Panjang karakter spesial pada *path*
Panjang *Header* '*Accept-Charset*'	Panjang huruf pada *path*
Panjang *Header* '*Content-Type*'	*Method*
Panjang *Header* '*Referer*'	Jumlah *keyword* pada *path*
Jumlah *query*	Jumlah *keyword* pada *query*
Panjang angka pada *query*	Jumlah *keyword* pada *body*
Panjang karakter spesial pada *query*	*Entropy*
Panjang *Header* '*Accept*'	Panjang karakter spesial pada *path*

Sumber: Data Nguyen dkk. (2013)

E. Normalisasi Data

Setelah proses ekstraksi fitur selesai, tahap berikutnya adalah normalisasi data. Normalisasi ini merupakan langkah krusial dalam analisis data, bertujuan untuk menyamakan skala nilai-nilai fitur agar lebih konsisten(Permana & Salisah, 2022). Dalam penelitian ini, teknik yang digunakan untuk normalisasi adalah *Min-Max Scaling*, yang mengonversi nilai fitur menjadi rentang antara 0 dan 1. Langkah ini penting untuk memastikan bahwa fitur-fitur dengan skala berbeda tidak mendominasi analisis, sehingga hasil yang diperoleh menjadi lebih akurat dan konsisten. Adapun untuk melakukan proses normalisasi data dengan *Min-Max Scaling* digunakan persamaan aritmatika berikut ini:

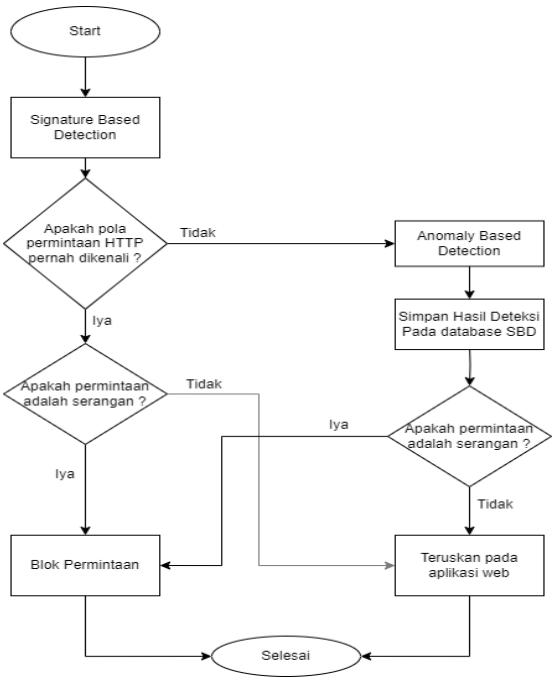
$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

Persamaan 1 merupakan persamaan yang digunakan untuk melakukan normalisasi data menggunakan metode *Min-Max Scaling*. Proses normalisasi ini mengubah nilai asli ke dalam rentang 0 hingga 1 berdasarkan nilai minimum dan nilai maksimum dari data.

F. Perancangan WAF

Pada proses ini akan dilakukan perancangan WAF yang menggunakan Pendekatan *Hybrid Anomaly-Signature Based*. Jika pola tidak dapat dikenali oleh pendekatan berbasis *signature*, maka proses akan berlanjut ke tahap berbasis *anomaly*. Jika permintaan HTTP teridentifikasi sebagai serangan, maka akan diblokir.

Sebaliknya, jika permintaan HTTP dianggap normal, maka akan diteruskan ke aplikasi web.



Gambar 2. Rancangan WAF
Sumber: diolah dari data primer

Setiap hasil deteksi dari pendekatan berbasis *anomaly* akan selalu disimpan pada *database* sehingga jika ada permintaan HTTP serupa pendekatan berbasis *signature* dapat mengenali. Rancangan dari WAF dapat dilihat pada Gambar 2.

a) Anomaly Based

Anomaly Based adalah pendekatan dalam keamanan komputer yang bertujuan `untuk mengidentifikasi perilaku yang tidak biasa atau anomali dalam sistem atau jaringan (Yang et al., 2022). Pada penelitian ini pendekatan *anomaly based* menggunakan Algoritma PSO-SVM. PSO digunakan untuk mencari nilai *hyperparameter* optimal, yaitu *C* dan *gamma*, yang sangat penting dalam meningkatkan kinerja SVM. Algoritma SVM sendiri akan menggunakan kernel *Radial Basis Function* (RBF).

b) Signature Based

Signature based adalah pendekatan dalam keamanan komputer yang memeriksa data atau aktivitas terhadap pola serangan yang telah dikenal sebelumnya, dengan membandingkan fitur atau tanda-tanda dari data tersebut dengan entri yang ada dalam *database signature* (Kumari et al., 2022).

c) *Particle Swarn Optimization* (PSO)

Particle Swarm Optimization (PSO) adalah algoritma metaheuristik berbasis populasi yang terinspirasi oleh perilaku sosial hewan, seperti gerakan berkumpul pada burung atau kelompok ikan(Saputra et al., 2020). PSO bekerja dengan menggerakkan sekelompok partikel dalam ruang pencarian, berdasarkan pengalaman individu masing-masing partikel serta pengalaman kolektif dari seluruh populasi. Adapun persamaan aritmatika dari PSO sebagai berikut:

$$v_i^t = wv_i^{t-1} + c_1 r_1 (p_i - x_i^t) + c_2 r_2 (g - x_i^t) \quad (2)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (3)$$

Persamaan 2 merupakan persamaan untuk memperbarui kecepatan (*velocity*) pada PSO, sedangkan persamaan 3 merupakan persamaan untuk mencari kecepatan partikel pada PSO. Untuk *fitness* function pada PSO dalam penelitian ini adalah nilai komplemen dari akurasi. Adapun persamaan aritmatika dari *fitness function* dari PSO yang digunakan sebagai berikut:

$$f(p_{gamma}, p_C) = (1 - Accuracy_{train}, 1 - Accuracy_{val}) \quad (4)$$

Persamaan 4 merupakan persamaan untuk *fitness function* yang digunakan pada penelitian ini. Tujuannya untuk menentukan nilai terbaik dari setiap partikel dan nilai terbaik global.

Semakin kecil maka dikatakan partikel mendekati dengan tujuan.

d) *Sequential Minimal Optimization* (SMO)

Sequential Minimal Optimization (SMO) adalah algoritma yang dirancang untuk menyelesaikan masalah optimasi ganda dalam pelatihan SVM dengan meningkatkan efisiensi dalam memecahkan masalah pemrograman kuadratik yang kompleks (Zhao et al., 2023). Pada penelitian ini algoritma *Support Vector Machine* (SVM) menggunakan pendekatan fungsi SMO tersebut. Adapun tahapan perhitungan dari SMO terdapat pada persamaan aritmatika sebagai berikut:

$$E_i = \sum_{k=1}^{N} a_k y_k K(x_{k,} x_i) + b - y_i \quad (5)$$

$$y_i E_i < -tol \;\&\&\; a_i < C \quad (6)$$

$$y_i E_i > tol \;\&\&\; a_i > 0 \quad (7)$$

$$b_1 = b - E_i - y_i(\alpha_i^{new} - \alpha_i)K(x_i, x_i) - y_j(\alpha_j^{new} - \alpha_j)K(x_i, x_j) \quad (8)$$

$$b_2 = b - E_j - y_i(\alpha_i^{new} - \alpha_i)K(x_i, x_j) - y_j(\alpha_j^{new} - \alpha_j)K(x_j, x_j) \quad (9)$$

$$b_{new} = \begin{cases} b_1 \;\; jika \; 0 < \alpha_i^{new} < C \\ b_2 \;\; jika \; 0 < \alpha_j^{new} < C \\ \frac{b_1 + b_2}{2} \;\; sebaliknya \end{cases} \quad (10)$$

Persamaan 5 merupakan persamaan aritmatika pada SMO yang digunakan untuk menemukan nilai optimal dari *Langrange* dan nilai bias, persamaan 6 dan 7 digunakan untuk mencari nilai optimasi dari nilai toleransi, kemudian persamaan 8 serta 9 digunakan untuk pertimbangan perubahan nilai alfa pada bias, selanjutnya persamaan 10 digunakan untuk mengetahui kondisi dari nilai bias yang terbaru.

G. Evaluasi Model

Pada tahap evaluasi model, metrik yang digunakan sebagai poin utama dari model terbaik adalah Akurasi, *True Positive Rate* (TPR), *False Positive Rate* (FPR), dan *False Discovery Rate* (FDR). Akurasi adalah metrik keseluruhan yang mengukur seberapa efektif sistem dalam mengklasifikasikan semua permintaan dengan benar, baik yang merupakan serangan maupun yang normal(Felisya et al., 2024). *True Positive Rate* (TPR) menggambarkan kemampuan WAF untuk secara tepat mengidentifikasi ancaman atau serangan yang sebenarnya terjadi. *False Positive Rate* (FPR) mengukur seberapa sering sistem salah mengklasifikasikan permintaan yang normal sebagai serangan. *False Discovery Rate* (FDR) memberikan indikasi tentang seberapa banyak peringatan yang dihasilkan oleh sistem yang tidak memiliki relevansi atau tidak memberikan kontribusi pada deteksi ancaman yang sebenarnya. Evaluasi ini memberikan gambaran komprehensif tentang kinerja sistem dalam hal ketepatan klasifikasi. Adapun persamaan aritmatika dari berbagai metrik yang digunakan dalam evaluasi sebagai berikut:

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \quad (11)$$

$$TPR = \frac{TP}{TP+FN} \quad (12)$$

$$FPR = \frac{FP}{TN+FP} \quad (13)$$

$$FDR = \frac{FP}{TP+FP} \quad (14)$$

Persamaan 11 menghitung akurasi, yaitu rasio prediksi yang benar (*True Positive* dan *True Negative*) terhadap total data prediksi (*True Positive*, *True Negative*, *False Positive*, dan *False Negative*). Persamaan 12 menghitung *True Positive Rate* (TPR) atau sensitivitas, yang menunjukkan kemampuan model mendeteksi serangan dengan benar. Persamaan 13 menghitung *False Positive Rate* (FPR), yang menunjukkan seberapa sering model salah mendeteksi serangan pada permintaan HTTP Normal. Persamaan 14 menghitung *False Discovery Rate* (FDR), yaitu proporsi prediksi serangan yang sebenarnya salah.

H. Skenario Pengujian

Pada penelitian ini, akan dilakukan enam skenario pengujian yang berbeda, masing-masing berfokus pada dua dataset dan pembagian data yang berbeda dari setiap dataset. Seluruh skenario akan diterapkan menggunakan algoritma PSO-SVM serta penggunaan *fitness function* akurasi. Setelah itu, skenario dengan hasil terbaik akan diuji kembali menggunakan pendekatan *hybrid anomaly-signature based*. Rincian skenario pengujian dapat dilihat pada Tabel 2.

Tabel 2. Skenario Pengujian

No	Jenis Dataset	Pembagian Data
1	CSIC 2010	85 : 15
2	CSIC 2010	80 : 20
3	CSIC 2010	75 : 25

4	ECML PKD 2007	85 : 15
5	ECML PKD 2007	80 : 20
6	ECML PKD 2007	75 : 25

Sumber: diolah dari data primer

III. HASIL DAN PEMBAHASAN

A. Standarisasi Data

Pada tahapan standarisasi data ini dilakukan tahapan pra-proses data dari dua dataset berbeda yaitu dataset CSIC 2010 dan ECML PKDD 2007.

Kedua dataset ini akan diubah ke dalam bentuk JSON dengan cara diubah dari format TXT ke JSON dan XML ke JSON. Hal ini dilakukan untuk menyamaratakan format dari kedua dataset dan juga memudahkan dalam ekstraksi fitur pada tahapan selanjutnya.

B. Ekstraksi Fitur pada Data

Tahapan selanjutnya adalah melakukan ekstraksi fitur dari data yang telah dilakukan pra-proses. Beberapa fitur yang digunakan adalah panjang karakter dari beberapa bagian dari permintaan HTTP yang penting dalam mendeteksi serangan *buffer overflow*.

Selain itu, dilakukan pengumpulan beberapa kata kunci dari beberapa bahasa pemrograman yang sering digunakan dalam serangan dan menghitung jumlah kemunculan kata kunci tersebut pada *path*, *query*, dan *body*

Panjang permintaan HTTP	Panjang Query	...	Entropy
587	0	...	5.307
692	35	...	5.257
762	0	...	5.226

Gambar 3. Hasil Ekstraksi Fitur
Sumber: diolah dari data primer

Gambar 3 merupakan hasil dari proses ekstraksi fitur. Data yang sebelumnya berupa teks diubah menjadi bentuk angka sesuai metode ekstraksi fitur yang telah ditentukan sebelumnya.

C. Normalisasi Data

Tahapan berikutnya setelah ekstraksi fitur adalah normalisasi data dengan *min-max scalling*. Nilai fitur yang didapat dari proses ekstraksi fitur sebelumnya diubah ke dalam bentuk angka dalam rentang 0 sampai 1 yang dapat dilihat pada Gambar 4.

Panjang permintaan HTTP	Panjang Query	...	Entropy
0.02	0	...	0.49
0.153	0.068	...	0.314
0.242	0	...	0.204

Gambar 4. Hasil Normalisasi Data
Sumber: diolah dari data primer

Gambar 4 merupakan hasil dari normalisasi data. Data yang sebelumnya memiliki rentang yang cukup jauh dilakukan normalisasi menjadi dalam rentang 0 sampai 1.

D. Evaluasi skenario Pengujian

Setelah ekstraksi fitur dan normalisasi data, dilakukan evaluasi sesuai skenario pengujian yang telah ditentukan sebelumnya dengan menguji semua model yang telah dilatih, seperti yang disajikan dalam Tabel 3.

Tabel 3. Hasil Skenario Pengujian

Skenario	Jenis Dataset	Pembagian Data	Akurasi	TPR	FPR	FDR
1	CSIC 2010	85 : 15	0.992	0.983	0.001	0.001
2	CSIC 2010	80 : 20	0.993	0.991	0.005	0.007
3	CSIC 2010	75 : 25	0.993	0.988	0.004	0.006
4	ECML PKD 2007	85 : 15	0.953	0.882	0.016	0.04
5	ECML PKD 2007	80 : 20	0.953	0.884	0.017	0.042
6	ECML PKD 2007	75 : 25	0.954	0.881	0.015	0.037

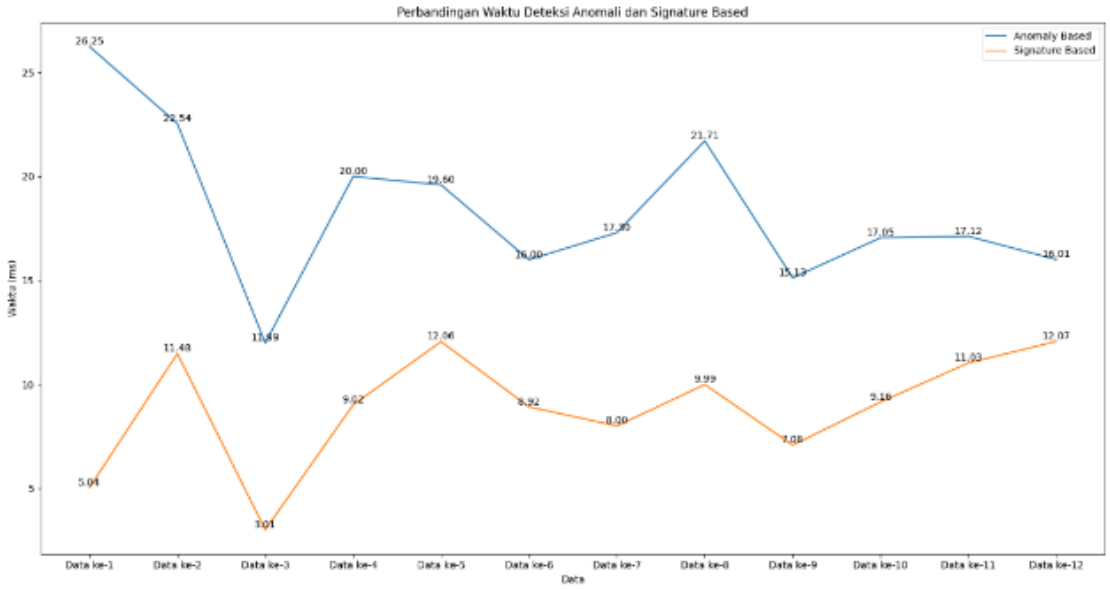
Sumber: diolah dari data primer

Dari Tabel 3 didapatkan hasil akurasi terbaik yakni dari skenario ke 2 dan 3 yang mendapatkan nilai akurasi sebesar 0.993 atau 99.3 % yang mana dari hasil tersebut skenario ke 3 diambil untuk mengukur kecepatan dari model ketika dilakukan pengujian kecepatan deteksi dari setiap pendekatan yaitu *anomaly* dan *signature* based. Skenario ke 3 dipilih dikarenakan dari pembagian data lebih besar skenario ini dibandingkan skenario ke 2.

E. Pengujian kecepatan deteksi

Pada tahapan ini skenario ke 3 akan dilakukan pengujian kecepatan deteksi dari model yang telah dilatih. Pengujian dilakukan dua kali untuk

menguji kecepatan deteksi dari setiap pendekatan yaitu *anomaly* dan *signature*.



Gambar 5. Hasil Deteksi Kecepatan Setiap Pendekatan
Sumber: diolah dari data primer

Pada iterasi pertama, deteksi akan dilakukan menggunakan pendekatan *anomaly* untuk setiap permintaan HTTP, dengan hasilnya disimpan dalam *database*. Selanjutnya, pada iterasi kedua, deteksi akan menggunakan pendekatan berbasis *signature* untuk memeriksa permintaan yang sama, guna membandingkan kecepatan dan efektivitas kedua pendekatan dalam mendeteksi serangan. Hasil dari kecepatan setiap pendekatan deteksi dapat dilihat pada Gambar 5.

Grafik pada gambar ke 5 kecepatan deteksi menunjukkan bahwa pendekatan *anomaly* dengan PSO-SVM cenderung memiliki waktu deteksi lebih lambat dibandingkan dengan pendekatan *signature*. Pendekatan *anomaly* mencatat kecepatan deteksi tercepat 11,99 ms pada data ke-3 dan yang terlama 26,25 ms pada data ke-1, sedangkan pendekatan *signature* mencatat waktu tercepat 3,01 ms pada data ke-3 dan terlama 12,07 ms pada data ke-12. Perbedaan ini terjadi karena pendekatan *signature* hanya memeriksa hasil *hash* fitur yang sudah ada dalam *database*, sehingga deteksi lebih cepat. Sebaliknya, pendekatan *anomaly* memerlukan perhitungan yang lebih kompleks untuk deteksi permintaan HTTP. Meskipun lebih lambat, pendekatan *anomaly* dapat mendeteksi serangan baru yang belum dikenali dalam *database*, menjadikan pendekatan *hybrid* yang menggabungkan kedua metode ini optimal dalam hal akurasi dan kecepatan.

IV. KESIMPULAN

Kesimpulan penelitian ini menunjukkan bahwa *Particle Swarm Optimization* (PSO) secara efektif mengoptimalkan parameter *Support Vector Machine* (SVM), mencapai akurasi lebih dari 99% pada pengujian dengan dataset CSIC 2010. Pembagian dataset 80:20 dan 75:25 menunjukkan kinerja sangat baik dalam hal akurasi tinggi, *True Positive Rate* (TPR) yang tinggi, dan *False Positive Rate* (FPR) yang rendah. Meskipun dataset ECML PKD 2007 sedikit tertinggal dalam akurasi dan TPR, hasilnya tetap memuaskan. Dalam hal kecepatan deteksi, pendekatan *signature* terbukti lebih cepat daripada *anomaly*. Namun, pendekatan *anomaly* memiliki keunggulan dalam mendeteksi serangan baru yang belum dikenal. Dengan menggunakan pendekatan *hybrid* yang menggabungkan kedua pendekatan, kecepatan deteksi dapat ditingkatkan sambil tetap mempertahankan kemampuan pendekatan *anomaly* untuk mendeteksi serangan baru yang belum dikenal. Pendekatan *hybrid* ini memungkinkan sistem untuk mendeteksi serangan dengan lebih efisien dan komprehensif.

DAFTAR PUSTAKA

[1] Calvo, M., & Beltrán, M. (2022). An Adaptive Web Application Firewall.

Proceedings of the International Conference on Security and Cryptography, *1*(Secrypt), 96–107. DOI: 10.5220/0011146900003283

[2] Dann, J., Wagner, R., Ritter, D., Faerber, C., & Froening, H. (2022). PipeJSON: Parsing JSON at Line Speed on FPGAs. *18th International Workshop on Data Management on New Hardware, DaMoN 2022*. DOI: 10.1145/3533737.3535094

[3] Felisya, E., Purba, B., Salsabilla, R. P., & Rahadiansyah, N. A. (2024). *2024 Madani : Jurnal Ilmiah Multidisipline Mengoptimalkan Deteksi Intrusi Jaringan : Perbandingan SVM dan KNN Menggunakan Dataset KddCup99 2024 Madani : Jurnal Ilmiah Multidisipline*. *2*(7), 276–283.

[4] Kumari, S., Singh, M., Singh, R., & Tewari, H. (2022). Signature based Merkle Hash Multiplication algorithm to secure the communication in IoT devices. *Knowledge-Based Systems*, *253*, 1–17. DOI: 10.1016/j.knosys.2022.109543

[5] Liu, Z., Wen, C., Xiao, Y., Fu, W., Wang, H., & Meng, Z. (2023). Path Tracking Control Algorithm of Tractor-Implement. *Smart Agriculture*, *5*(4), 45–57. DOI: 10.12133/j.smartag.SA202308004

[6] Nagarjun, P. M. D., & Ahamad, S. S. (2020). Cross-site scripting research: A review. *International Journal of Advanced Computer Science and Applications*, *11*(4), 626–631. DOI: 10.14569/IJACSA.2020.0110481

[7] Permana, I., & Salisah, F. N. S. (2022). Pengaruh Normalisasi Data Terhadap Performa Hasil Klasifikasi Algoritma Backpropagation. *Indonesian Journal of Informatic Research and Software Engineering (IJIRSE)*, *2*(1), 67–72. DOI: 10.57152/ijirse.v2i1.311

[8] Saputra, R. A., Agustina, C., Puspitasari, D., Ramanda, R., Warjiyono, Pribadi, D., Lisnawanty, & Indriani, K. (2020). Detecting Alzheimer's Disease by the Decision Tree Methods Based on Particle Swarm Optimization. *Journal of Physics: Conference Series*, *1641*(1), 61–67. DOI: 10.1088/1742-6596/1641/1/012025

[9] Soesanto, E., Romadhon, A., Dwi Mardika, B., & Fahmi Setiawan, M. (2023). Analisis dan Peningkatan Keamanan Cyber: Studi Kasus Ancaman dan Solusi dalam Lingkungan Digital Untuk Mengamankan Objek Vital dan File. *SAMMAJIVA : Jurnal Penelitian Bisnisdan Manajemen*, *1*(2), 186.

[10] Tasevski, I., & Jakimoski, K. (2020). Overview of SQL injection defense mechanisms. *2020 28th Telecommunications Forum, TELFOR 2020 - Proceedings*. DOI: 10.1109/TELFOR51502.2020.9306676

[11] Tekerek, A., & Bay, O. F. (2019). DESIGN AND IMPLEMENTATION OF AN ARTIFICIAL INTELLIGENCE-BASED WEB APPLICATION FIREWALL MODEL. *Neural Network World*, *4*, 189–206. DOI: 10.14311/NNW.2019.29.013

[12] Wolfson, S., Mohammed, A., Xu, A., Magness, A., Edmonds, C., Dai, D., Peng, J., Jiang, M., Kirdani-ryan, M., Auradkar, N., & Song, Y. (2023). *Buffer Overflows*.

[13] Yang, Z., Liu, X., Li, T., Wu, D., Wang, J., Zhao, Y., & Han, H. (2022). A systematic literature review of methods and datasets for anomaly-based network intrusion detection. *Computers and Security*, *116*. DOI: 10.1016/j.cose.2022.102675

[14] Zhao, C., Dai, L., & Huang, Y. (2023). Fractional Order Sequential Minimal Optimization Classification Method. *Fractal and Fractional*, *7*(8). DOI: 10.3390/fractalfract7080637

