IMPLEMENTASI ALGORITMA NAIVE BAYES UNTUK KLASIFIKASI AKTIVITAS USER PADA SISTEM KEAMANAN SSH SERVER

Paskalia Br Barus[1], Wirma Naibaho[2], Boy mountavani sembiring[3], Lotar Mateus Sinaga[4]

[1,2,3,4]*Universitas Katolik Santo Thomas*
E-mail : paskaliabarus@gmail.com[1], wirmanaibaho@gmail.com[2], boysembiring583@gmail.com[3], lotarmateus88@gmail.com[4]

ABSTRAK

Layanan *Secure Shell* (SSH) merupakan salah satu protokol utama yang digunakan untuk manajemen jarak jauh pada server berbasis Linux Debian. Namun, karena aksesibilitasnya yang terbuka, protokol ini sering menjadi target utama serangan siber otomatis, khususnya serangan *brute force* yang mencoba menembak kombinasi kata sandi secara masif. Penelitian ini bertujuan untuk mengimplementasikan algoritma Naive Bayes dalam melakukan klasifikasi dan deteksi dini terhadap aktivitas intrusi *brute force* berdasarkan rekaman data log SSH. Untuk meningkatkan akurasi klasifikasi pada kondisi data yang tidak seimbang (*imbalanced dataset*), dilakukan ekstraksi fitur perilaku yang meliputi kategori waktu akses, frekuensi koneksi alamat IP, dan alokasi port, serta menerapkan teknik *Laplace Smoothing*. Pengumpulan data dilakukan melalui simulasi serangan menggunakan *tools* Hydra serta pengujian aktivitas login normal hingga menghimpun total dataset sebanyak 2.513 baris log (2.000 data ATTACK dan 513 data NORMAL). Pengujian model dilakukan menggunakan skenario *split data* dengan proporsi 80% data *training* dan 20% data *testing*, serta divalidasi menggunakan *K-Fold Cross Validation* (K=5). Hasil pengujian menunjukkan bahwa model Naive Bayes yang diusulkan mampu mengklasifikasikan aktivitas log dengan performa yang sangat tinggi, menghasilkan nilai akurasi sebesar 99,60%, *precision* 99,75%, *recall* 99,75%, dan *F1-score* 99,75%. Validasi silang menggunakan *K-Fold* juga menunjukkan konsistensi dan stabilitas model dengan rata-rata akurasi sebesar 99,64%. Penelitian ini membuktikan bahwa algoritma Naive Bayes sangat efektif dan konsisten untuk diterapkan sebagai sistem deteksi intrusi pada layanan SSH server.

Kata Kunci : *Naive Bayes, SSH, Keamanan Jaringan, Klasifikasi, Kali Linux*

ABSTRACT

Secure Shell (SSH) service is one of the primary protocols used for remote management of Linux Debian-based servers. However, due to its open accessibility, this protocol is frequently targeted by automated cyber attacks, particularly brute force attacks that attempt to guess password combinations massively. This study aims to implement the Naive Bayes algorithm to classify and provide early detection of brute force intrusion activities based on SSH log data records. To improve classification accuracy under imbalanced dataset conditions, behavioral feature extraction was performed, including access time categories, IP address connection frequency, and port allocation, as well as applying the Laplace Smoothing technique. Data collection was conducted through attack simulations using the Hydra tool and normal login activity testing, accumulating a total dataset of 2,513 log entries consisting of 2,000 ATTACK data and 513 NORMAL data, divided into 2,010 training data (80%) and 503 testing data (20%). Model testing was performed using an 80/20 data split scenario and validated using K-Fold Cross Validation (K=5). The test results show that the proposed Naive Bayes model is capable of classifying log activities with very high performance, achieving an accuracy of 99.60%, precision of 99.75%, recall of 99.75%, and F1-score of 99.75%. Cross-validation using K-Fold also demonstrated model consistency and stability with an average accuracy of 99.64%, highest accuracy of 99.80%, and lowest accuracy of 99.20%. This study proves that the Naive Bayes algorithm is highly effective and consistent for application as an intrusion detection system on SSH server services.

Keywords : *Naïve Bayes, Secure Shell (SSH), Network Security, Classification, Kali Linux.*

PENDAHULUAN

Perkembangan infrastruktur jaringan dan pemanfaatan server berbasis Linux Debian terus mengalami peningkatan seiring dengan kebutuhan akan sistem yang andal dalam pengelolaan data dan penyediaan layanan. Dalam administrasi server jarak jauh, Secure Shell (SSH) merupakan salah satu protokol yang paling banyak digunakan karena mampu menyediakan komunikasi yang aman melalui mekanisme enkripsi, sehingga administrator jaringan dapat mengelola dan memantau server secara efisien tanpa mengabaikan aspek keamanan (Balqis & Rahman, 2025). Namun, karakteristik protokol SSH yang umumnya terbuka secara publik pada port standar menjadikannya salah satu target utama berbagai aktivitas ilegal (Jeremya et al., 2025). Di antara berbagai ancaman yang mengincar layanan tersebut, serangan brute force merupakan jenis intrusi yang paling sering terjadi dan menimbulkan beban kerja signifikan pada sistem. Serangan ini bekerja secara masif dengan memanfaatkan perangkat lunak otomatis untuk menguji ribuan kombinasi nama pengguna dan kata sandi hanya dalam hitungan detik hingga menit . Aktivitas tersebut tidak hanya mengancam kerahasiaan data apabila kata sandi berhasil ditembus, tetapi juga membebani sumber daya komputasi server akibat membanjirnya log percobaan login yang gagal (Failed password).

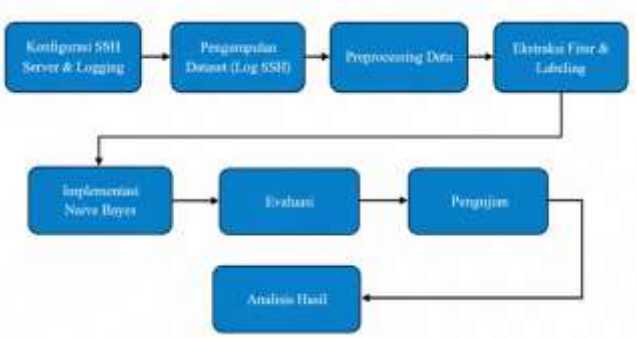
Masalah utama yang sering dihadapi yaitu bagaimana mendeteksi dan mengklasifikasikan tumpukan data log SSH yang besar secara instan. Metode pemantauan log secara manual oleh administrator jaringan sudah tidak lagi relevan karena membutuhkan waktu yang lama dan rentan terhadap kelalaian manusia (Balqis & Rahman, 2025). Beberapa penelitian terkait (*overview*) telah dilakukan untuk mengatasi permasalahan ini. Sebagai contoh, penelitian terdahulu menerapkan sistem deteksi intrusi berbasis pencocokan pola teks statis (*rule-based*) untuk menyaring log serangan. (Jeremya et al., 2025) menjelaskan bahwa Intrusion Detection System berbasis signature atau rule-based memiliki keterbatasan dalam menghadapi pola serangan yang semakin kompleks sehingga diperlukan pendekatan probabilistik seperti Naïve Bayes.

Sebagai usulan metode untuk menyelesaikan permasalahan tersebut (*state of the art*), penelitian ini menerapkan teknik komputasi cerdas menggunakan algoritma *machine learning* Naive Bayes yang diperkuat dengan teknik *Laplace Smoothing* (Rizky et al., 2024). Algoritma Naive Bayes dipilih karena memiliki efisiensi komputasi yang sangat cepat serta performa yang andal dalam memproses data log berukuran besar berbasis nilai probabilitas bersyarat. Penambahan *Laplace Smoothing* berfungsi sebagai solusi untuk mengeliminasi nilai probabilitas nol pada variasi fitur baru yang belum pernah muncul pada data latih. Selanjutnya, guna memastikan stabilitas performa model dari indikasi bias data, pengujian validasi dilakukan menggunakan metode *K-Fold Cross Validation* (K=5) (Ridwansyah, 2022). Melalui skenario tersebut, penelitian ini bertujuan untuk mengimplementasikan arsitektur deteksi cerdas pada server Linux Debian, meminimalkan tingkat salah deteksi, serta memberikan pembuktian ilmiah berupa metrik performa yang valid dalam menangani ancaman *brute force* pada protokol SSH.

METODE

Tahapan Penelitian

Metode penelitian ini dirancang secara sistematis untuk mengimplementasikan algoritma Naive Bayes dalam mengklasifikasikan aktivitas user pada SSH Server Debian 12. Tahapan penelitian secara keseluruhan dapat dilihat pada Gambar 1.



Gambar 1. Diagram Alur Metodologi Penelitian

Konfigurasi SSH Server dan Logging

Pengumpulan data log SSH dilakukan melalui dua skenario secara bersamaan. Skenario pertama adalah simulasi serangan brute force menggunakan tool Hydra yang dijalankan dari mesin Kali Linux dengan alamat IP 192.168.56.106 yang menyerang server SSH Debian 12 pada alamat IP 192.168.56.107. Hydra dijalankan menggunakan wordlist rockyou.txt yang berisi daftar kombinasi password yang umum digunakan. Skenario kedua adalah aktivitas login normal yang dilakukan oleh pengguna sah menggunakan kredensial yang benar. Kedua mesin virtual tersebut dihubungkan melalui jaringan Host-Only Adapter pada Oracle VirtualBox sebagaimana ditunjukkan pada Gambar 2.

Preprocessing Data

Tahapan *preprocessing* dilakukan untuk membersihkan data log SSH dari entri yang tidak relevan, seperti log sistem operasi, log aplikasi GNOME, dan log layanan lainnya yang tidak berkaitan dengan aktivitas autentikasi SSH(Nasution et al., 2015). Proses *filtering* dilakukan menggunakan perintah grep untuk memisahkan hanya entri log yang mengandung kata kunci *Failed password* dan *Accepted password*. Tahapan ini bertujuan untuk menghilangkan data yang tidak diperlukan sehingga hanya informasi yang berkaitan dengan aktivitas login SSH yang diproses lebih lanjut. Hasil *preprocessing* menghasilkan dataset yang bersih dan siap digunakan pada tahap ekstraksi fitur serta proses klasifikasi.

Ekstraksi Fitur dan Labeling

Setiap baris log SSH yang telah melalui tahapan preprocessing kemudian diproses untuk mengekstrak fitur-fitur perilaku yang relevan. Fitur yang digunakan tidak menyertakan label status login secara langsung untuk menghindari kebocoran data (data leakage), melainkan menggunakan karakteristik perilaku akses yang dapat membedakan aktivitas normal dan serangan secara implisit. Detail fitur yang digunakan dapat dilihat pada Tabel 2.

Tabel 1. Fitur yang Digunakan dalam Klasifikasi

No	Fitur	Keterangan
1.	Kategori Jam	Waktu akses login
2.	Username	Nama pengguna yang digunakan
3.	Alamat IP	IP address sumber koneksi
4.	Frekuensi IP	Jumlah percobaan per IP

5.	Kategori Port	Port sumber koneksi

Setelah ekstraksi fitur, setiap baris data diberikan label klasifikasi secara otomatis berdasarkan status login yang terdapat pada log. Baris yang mengandung Failed password diberi label ATTACK, sedangkan baris yang mengandung Accepted password diberi label NORMAL.

Algoritma Naive Bayes

Algoritma Naive Bayes merupakan algoritma klasifikasi probabilistik yang bekerja berdasarkan teorema Bayes dengan asumsi bahwa setiap fitur bersifat independen satu sama lain (Harahap et al., 2021) . Persamaan dasar teorema Bayes yang digunakan dalam penelitian ini adalah sebagai berikut:

$$P(C|X) = P(X|C) \times P(C) / P(X)$$

Keterangan:

- $P(C \mid X)$= Probabilitas kelas C given fitur X
- $P(X \mid C)$= Probabilitas fitur X given kelas C
- $P(C)$= Probabilitas prior kelas C
- $P(X)$= Probabilitas fitur X

$$P(xi|C) = (1/\sqrt{2\pi\sigma^2}) \times exp(-(xi-\mu)^2/2\sigma^2)$$

Keterangan:

- n_c= Jumlah kemunculan fitur pada kelas C
- N_c= Total fitur pada kelas C
- k= Jumlah fitur unik (vocabulary size)

Pengujian dan Evaluasi Model

Pengujian model dilakukan menggunakan dua skenario. Skenario pertama adalah pembagian data dengan proporsi 80% data training sebanyak 2010 data dan 20% data testing sebanyak 503 data. Sebelum pembagian data, dataset diacak terlebih dahulu (shuffle) menggunakan random seed 42 untuk memastikan distribusi data yang merata. Skenario kedua adalah K-Fold Cross Validation dengan nilai K=5 yang membagi dataset menjadi 5 bagian secara bergantian sebagai data training dan testing untuk memastikan konsistensi dan stabilitas model.
Metrik evaluasi yang digunakan meliputi akurasi, precision, recall, dan F1-Score dengan persamaan sebagai berikut:

$$Accuracy = (TP + TN) / (TP + TN + FP + FN)$$

$$Precision = TP / (TP + FP)$$

$$Recall = TP / (TP + FN)$$

$$F1\text{-}Score = 2 \times (Precision \times Recall) / (Precision + Recall)$$

Keterangan:

- TP (True Positive) = Data ATTACK diprediksi ATTACK
- TN (True Negative) = Data NORMAL diprediksi NORMAL
- FP (False Positive) = Data NORMAL diprediksi ATTACK
- FN (False Negative) = Data ATTACK diprediksi NORMAL

HASIL PENELITIAN DAN PEMBAHASAN

Penelitian ini berhasil mengimplementasikan algoritma Naive Bayes untuk mengklasifikasikan aktivitas user pada SSH Server Debian 12 menggunakan lingkungan

virtualisasi Oracle VirtualBox. Pada server Debian 12, layanan SSH telah tersedia dan aktif, sehingga proses implementasi dapat langsung difokuskan pada pengumpulan data log, ekstraksi fitur, dan pengujian model klasifikasi. Proses implementasi mengikuti alur tahapan yang terdiri dari empat fase utama: persiapan, pengumpulan dataset, pengujian, dan evaluasi, sebagaimana ditampilkan pada Gambar 1.

Persiapan

Konfigurasi Server SSH

Setelah lingkungan virtualisasi disiapkan, selanjutnya yaitu melakukan tahap implementasi yang akan difokuskan pada konfigurasi layanan SSH pada server Debian 12, agar mendukung skenario pengujian serangan brute force. Konfigurasi dilakukan pada file *"sudo nano /etc/ssh/sshd_config"* yang mengatur perilaku layanan OpenSSH pada server Debian 12.

Sebagaimana ditunjukkan pada Gambar 3, konfigurasi server telah disesuaikan agar

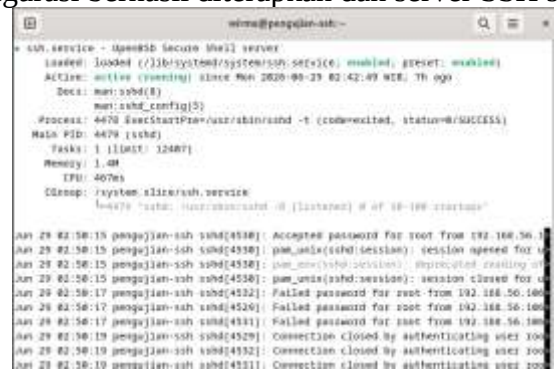
1. Mengizinkan login langsung sebagai root dengan pengaturan PermitRootLogin yes, sehingga simulasi serangan brute force dapat dilakukan langsung pada akun root.
2. Mengaktifkan autentikasi berbasis password melalui PasswordAuthentication yes, sehingga Hydra dapat melakukan percobaan login menggunakan kombinasi password secara otomatis.

Setelah konfigurasi selesai dilakukan, layanan SSH di-restart menggunakan perintah berikut agar perubahan konfigurasi dapat diterapkan:

"systemctl restart ssh"

"systemctl status ssh"

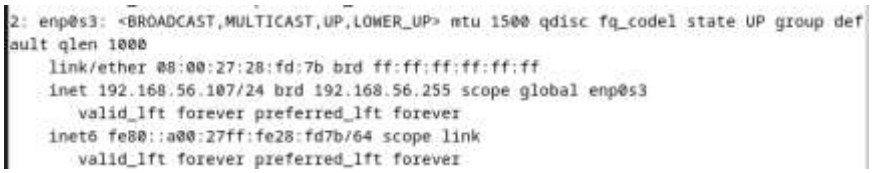
Hasil verifikasi menunjukkan bahwa layanan SSH berstatus active (running) yang membuktikan bahwa konfigurasi berhasil diterapkan dan server SSH siap menerima koneksi.



Gambar 3. Status Layanan SSH

Verifikasi Konektivitas Jaringan

Setelah konfigurasi SSH selesai dilakukan, tahap selanjutnya adalah memastikan konektivitas jaringan antara server Debian 12 dan mesin penyerang Kali Linux. Pengecekan alamat IP dilakukan menggunakan perintah ip a pada masing-masing mesin virtual. Hasil pengecekan menunjukkan bahwa server Debian 12 mendapatkan alamat IP 192.168.56.107 pada interface enp0s3, sedangkan mesin penyerang Kali Linux mendapatkan alamat IP 192.168.56.106 pada interface eth0 sebagaimana ditunjukkan pada Gambar 5.

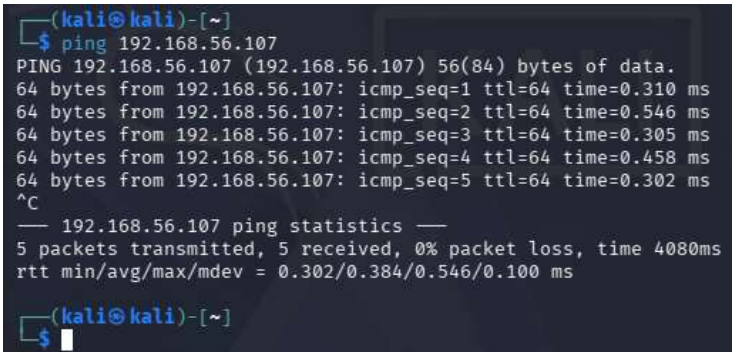


Gambar 5. Hasil ip a Server Debian 12

Untuk memastikan kedua mesin dapat saling berkomunikasi, dilakukan pengujian konektivitas menggunakan perintah ping dari mesin Kali Linux ke server Debian 12 sebagai berikut:

"ping 192.168.56.107"

Hasil pengujian ping menunjukkan bahwa paket data berhasil dikirim dan diterima dengan baik antara kedua mesin virtual, yang membuktikan bahwa konektivitas jaringan Host-Only antara Kali Linux dan server SSH Debian 12 telah terhubung dengan sempurna sebagaimana ditunjukkan pada Gambar 6.

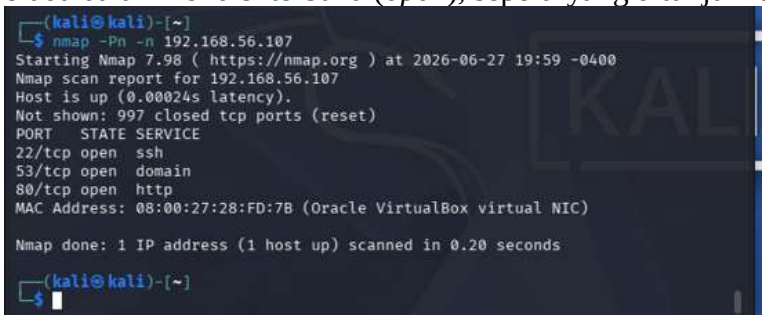


Gambar 6. Hasil Ping dari Kali Linux ke Server Debian 12

Pengumpulan Dataset

Simulasi Serangan Brute Force

Setelah konektivitas jaringan dipastikan berjalan dengan baik, tahap selanjutnya adalah melakukan pengujian penetrasi berupa simulasi serangan *brute force* dari mesin Kali Linux menggunakan *tool* Hydra. Langkah awal yang dilakukan adalah melakukan pemindaian *port* menggunakan Nmap untuk memastikan status *port* layanan SSH pada server target (192.168.56.107) berada dalam kondisi terbuka (*open*), seperti yang ditunjukkan pada Gambar 6.

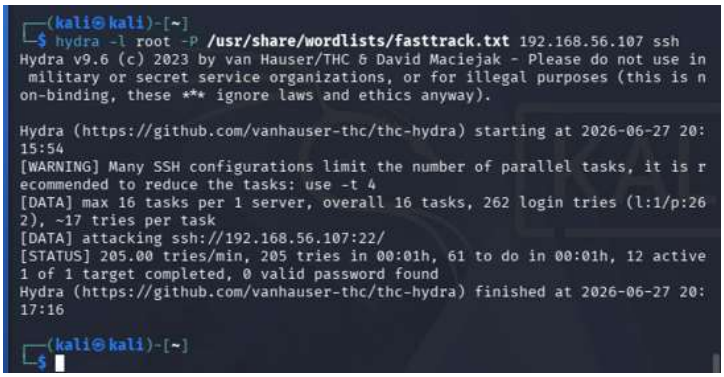


Gambar 6. Hasil Pemindaian Port Menggunakan Nmap pada Kali Linux

Setelah port 22/tcp terkonfirmasi aktif dan terbuka secara publik, simulasi penembakan kata sandi secara massal (*brute force*) dijalankan menggunakan perintah berikut:

"hydra -l root -P /usr/share/wordlists/fasttrack.txt 192.168.56.107 ssh"

Proses jalannya simulasi serangan otomatisasi ini memanfaatkan daftar kata sandi bawaan dari berkas kamus *fasttrack.txt* untuk menebak akun pengguna tertinggi (*root*) pada server target. Hasil akhir dari keseluruhan proses eksekusi *tool* Hydra ini ditunjukkan secara menyeluruh pada Gambar 7.



Gambar 7. Hasil Akhir Eksekusi Serangan Brute Force Menggunakan Hydra

Ekstraksi dan Distribusi Dataset

Setelah seluruh rangkaian simulasi serangan dan aktivitas akses normal selesai terekam, tahap selanjutnya adalah melakukan ekstraksi dan penyaringan komponen log secara spesifik. Proses pemisahan data dilakukan langsung melalui perintah terminal Linux Debian dengan membagi log ke dalam dua berkas sementara, yaitu failed_2000.txt untuk label serangan (ATTACK) dan normal_513.txt untuk label aktivitas sah (NORMAL) menggunakan perintah berikut

"grep 'Failed password' dataset_gabung.txt | head -2000 > failed_2000.txt"

"grep 'Accepted' dataset_gabung.txt > normal_513.txt"

Output dari penggabungan kedua berkas tersebut disimpan dengan nama dataset_final2.txt melalui perintah berikut:

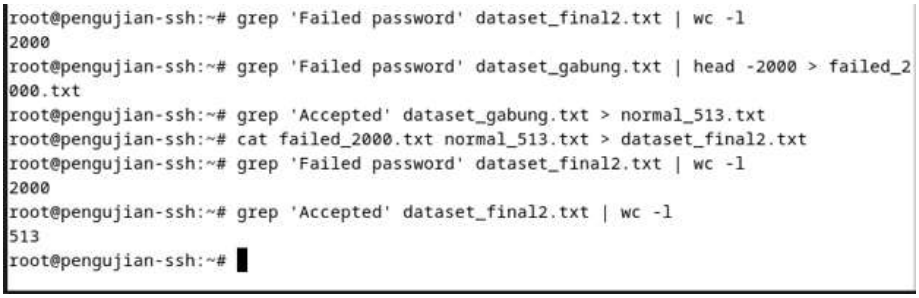
"cat failed_2000.txt normal_513.txt > dataset_final2.txt"

Untuk memverifikasi jumlah data pada dataset final, dilakukan pengecekan menggunakan perintah berikut:

"grep 'Failed password' dataset_final2.txt | wc -l"

"grep 'Accepted' dataset_final2.txt | wc -l"

Gambar hasil verifikasi dan penggabungan baris log pada dataset final ditunjukkan pada Gambar 8.



Gambar 8. Proses Verifikasi dan Penggabungan Baris Log pada Dataset Final

Hasil pengumpulan dan filtering data menghasilkan total 2513 baris log SSH yang terdiri dari 2000 data serangan dan 513 data normal sebagaimana ditunjukkan pada Tabel 3.

Tabel 3. Distribusi Datset Log SSH

Label	Keterangan	Jumlah
ATTACK	Failed password	2000
NORMAL	Accepted password	513
Total		2513

Implementasi Naive Bayes

Setelah dataset final terbentuk dengan total 2513 baris log SSH, tahap selanjutnya adalah mengimplementasikan algoritma Naive Bayes menggunakan bahasa pemrograman Python yang dijalankan langsung di terminal server Debian 12. Script Python yang dibangun mencakup proses parsing log SSH, ekstraksi fitur perilaku, training model, dan evaluasi.

Script dijalankan menggunakan perintah berikut:

"python3 naive_bayes_final.py"

Sebelum proses training dilakukan, dataset diacak terlebih dahulu menggunakan random seed 42 untuk memastikan distribusi data yang merata, kemudian dibagi menjadi 2010 data training (80%) dan 503 data testing (20%) sebagaimana ditunjukkan pada Tabel 4.

Tabel 4. Pembagian Data Training dan Testing

Kategori	Jumlah	Persentase
Data Training	2010	80%
Data Testing	503 503	20%
Total	2513	100%

Pengujian Model

Pengujian model dilakukan menggunakan dua skenario untuk memastikan performa dan konsistensi model Naive Bayes yang telah dibangun. Hasil pengujian model ditunjukkan pada Gambar 9.

```
HASIL EVALUASI NAIVE BAYES

Akurasi         : 99.60                 %
Precision       : 99.75                 %
Recall          : 99.75                 %
F1-Score        : 99.75                 %
```

Gambar 9. Output Hasil Pengujian Naive Bayes di Terminal Debian

Selanjutnya dilakukan analisis confusion matrix untuk melihat detail hasil prediksi model pada setiap kelas sebagaimana ditunjukkan pada Gambar 10.

```
CONFUSION MATRIX

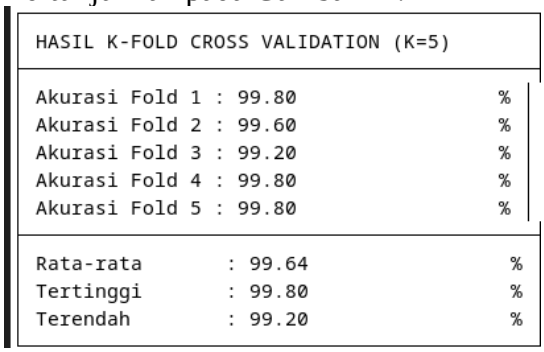
                | Prediksi ATTACK | Prediksi NORMAL
Aktual ATTACK   | 403             | 1
Aktual NORMAL   | 1               | 98
```

Gambar 10. Output Confusion Matrix di Terminal Debian

Berdasarkan hasil confusion matrix pada Gambar 10 dapat dianalisis bahwa:

1. True Positive (TP) = 403 model berhasil mendeteksi 403 data serangan brute force dengan benar sebagai ATTACK.
2. True Negative (TN) = 98 model berhasil mengklasifikasikan 98 data login normal dengan benar sebagai NORMAL.
3. False Positive (FP) = 1 hanya terdapat 1 data login normal yang salah diklasifikasikan sebagai ATTACK, sehingga model hampir tidak menghasilkan peringatan palsu.
4. False Negative (FN) = 1 hanya terdapat 1 data serangan yang tidak berhasil terdeteksi oleh model sebagai ATTACK.

Skenario kedua adalah pengujian menggunakan K-Fold Cross Validation dengan nilai K=5 yang membagi dataset menjadi 5 bagian secara bergantian sebagai data training dan data testing. Hasil K-Fold Cross Validation ditunjukkan pada Gambar 11.



Gambar 11. Output K-Fold Cross Validation di Terminal Debian

Analisis Hasil Uji

Berdasarkan seluruh hasil pengujian yang telah dilakukan, dapat dianalisis bahwa algoritma Naive Bayes mampu mengklasifikasikan aktivitas user pada SSH Server Debian 12 dengan performa yang sangat tinggi. Nilai akurasi sebesar 99,60% menunjukkan bahwa dari 503 data testing hanya 2 data yang salah diklasifikasikan oleh model. Tingginya nilai akurasi ini tidak terlepas dari pendekatan rekayasa fitur perilaku yang digunakan, dimana model menganalisis pola perilaku akses seperti kategori waktu login, frekuensi koneksi per alamat IP, dan alokasi port.

Nilai precision sebesar 99,75% membuktikan bahwa model hampir tidak menghasilkan peringatan palsu (False Positive = 1), sehingga administrator jaringan tidak akan terganggu oleh notifikasi serangan yang tidak relevan. Nilai recall sebesar 99,75% menunjukkan bahwa model mampu mendeteksi hampir seluruh serangan brute force yang terjadi dengan hanya 1 serangan yang lolos tidak terdeteksi (False Negative = 1). Keseimbangan antara nilai precision dan recall yang sama-sama 99,75% menghasilkan nilai F1-Score sebesar 99,75%, yang menunjukkan bahwa model memiliki keseimbangan yang sangat baik dalam mendeteksi kedua kelas secara bersamaan.

Hasil K-Fold Cross Validation dengan rata-rata akurasi 99,64% dan rentang akurasi antara 99,20% hingga 99,80% membuktikan bahwa model tidak hanya baik pada satu skenario pembagian data tertentu, melainkan konsisten dan stabil pada berbagai skenario pembagian data. Hal ini memperkuat kesimpulan bahwa algoritma Naive Bayes dengan rekayasa fitur perilaku yang diusulkan dalam penelitian ini layak diterapkan sebagai sistem deteksi intrusi pada layanan SSH server.

KESIMPULAN

Berdasarkan hasil implementasi dan pengujian yang telah dilakukan, algoritma Naive Bayes berhasil mengklasifikasikan aktivitas user pada SSH Server Debian 12 dengan nilai akurasi 99,60%, precision 99,75%, recall 99,75%, dan F1-Score 99,75%. Hasil K-Fold Cross Validation dengan K=5 menghasilkan rata-rata akurasi 99,64% yang membuktikan bahwa model memiliki konsistensi dan stabilitas yang baik. Penerapan teknik Laplace Smoothing dan rekayasa fitur perilaku berbasis kategori waktu akses, frekuensi koneksi per alamat IP, dan alokasi port terbukti efektif dalam membedakan aktivitas serangan brute force dari aktivitas login normal pada layanan SSH Server

Penelitian selanjutnya disarankan untuk menambah variasi fitur perilaku dan membandingkan performa algoritma Naive Bayes dengan algoritma machine learning lainnya seperti Decision Tree atau Random Forest. Selain itu, sistem deteksi ini dapat dikembangkan menjadi sistem monitoring real-time yang mampu memberikan peringatan otomatis kepada administrator jaringan ketika terdeteksi aktivitas serangan pada layanan SSH.

DAFTAR PUSTAKA

Balqis, P., & Rahman, R. (2025). Analisis Keamanan Layanan SSH terhadap Brute Force Attack. *Merkurius : Jurnal Riset Sistem Informasi Dan Teknik Informatika*, *3*(4), 240–246. https://journal.arteii.or.id/index.php/Merkurius/article/view/949

Harahap, F., Saragih, N. E., Siregar, E. T., & Sariangsah, H. (2021). Penerapan Data Mining Dengan Algoritma Naive Bayes Classifier Dalam Memprediksi Pembelian Cat KATA KUNCI Data Mining Purchase of paint Naive Bayes. KORESPONDENSI. *Jurnal Ilmiah Informatika*, *09*(01), 19–23.

Jeremya, T., Lombu, A., Daeli, L., & Bulolo, S. (2025). *Jurnal Inovasi Teknologi dan Komputer Implementasi dan Analisis Keamanan Layanan SSH Server Menggunakan Autentikasi RSA dalam Lingkungan Virtualisasi dan Putty*. *03*(01), 33–45.

Nasution, N., Djahara, K., & Zamsuri, A. (2015). Evaluasi Kinerja Akademik Mahasiswa Menggunakan Algoritma Naive Bayes (Studi Kasus: Fasilkom Unilak). *Jurnal Teknologi Informasi Dan Komunikasi Digital Zone*, *6*(2), 1–11.

Ridwansyah, T. (2022). KLIK: Kajian Ilmiah Informatika dan Komputer Implementasi Text Mining Terhadap Analisis Sentimen Masyarakat Dunia Di Twitter Terhadap Kota Medan Menggunakan K-Fold Cross Validation Dan Naïve Bayes Classifier. *Media Online*, *2*(5), 178–185. https://djournals.com/klik

Rizky, Y. A., Aziz, A., & Harianto, W. (2024). Implementasi Naive Bayes Dengan Menggunakan Metode Laplace Smoothing. *RAINSTEK: Jurnal Terapan Sains Dan Teknologi*, *6*(3), 164–172. https://doi.org/10.21067/jtst.v6i3.9132

Bustami, Bustami. "Penerapan algoritma Naive Bayes untuk mengklasifikasi data nasabah asuransi." *TECHSI-Jurnal Teknik Informatika* 5.2 (2013).

Nasution, Nurliana, Khairani Djahara, and Ahmad Zamsuri. "Evaluasi Kinerja Akademik Mahasiswa Menggunakan Algoritma Naïve Bayes (Studi Kasus: Fasilkom Unilak)." *Digital Zone: Jurnal Teknologi Informasi Dan Komunikasi* 6.2 (2015): 1-11.

Hardjianto, Mardi. "Sistem Monitoring Serangan Ssh Dengan Metode Intrusion Prevention System (IPS) Fail2ban Menggunakan Python Pada Sistem Operasi Linux." *Jurnal TICOM:*

Technology of Information and Communication 11.1 (2022): 33-38.

Nanlohy, Ovan Philipi, and Arif Faizin. "IMPLEMENTASI HONEYPOT UNTUK MENDETEKSI SERANGAN BRUTE FORCE PADA LAYANAN SSH." *JATI (Jurnal Mahasiswa Teknik Informatika)* 9.6 (2025): 9454-9457.

Simanjuntak, Erick, et al. "Analisis Kerja dan Keamanan Algoritma Enkripsi AES pada Protokol SSH di Debian Server." *JITKO: Jurnal Inovasi Teknologi dan Komputer* 2.01 (2024): 41-47.

Fitra, Muhammad Rizki Andrian, et al. "Uji Penetrasi Menggunakan Hydra dan Metasploit pada Protokol Secure Shell." *JATI (Jurnal Mahasiswa Teknik Informatika)* 9.1 (2025): 1017-1024.

Novianto, Fahmi. "Keamanan http dan https berbasis web menggunakan sistem operasi Kali Linux." *Komputa: Jurnal Ilmiah Komputer dan Informatika* (2015).

Lahagu, Delvin Krisnawati, et al. "Analisis Potensi Kerentanan Terhadap Serangan Phishing Pada Website sttsundermann. siakadcloud. com Menggunakan Simulasi Lingkungan Kali Linux dan Ngrok." *Jurnal Komputer Teknologi Informasi Sistem Informasi (JUKTISI)* 4.2 (2025): 390-397.

Desmira, Desmira, and Romi Wiryadinata. "Rancang Bangun Keamanan Port Secure Shell (SSH) Menggunakan Metode Port knocking." *Jurnal Ilmu Komputer dan Sistem Informasi (JIKOMSI)* 5.1 (2022): 28-33.

Ridho, M. Ridho M., et al. "Peningkatan Keamanan SSH Server Berbasis Linux melalui Implementasi Fail2Ban dan Uji Serangan Brute Force." *Jurnal Penelitian Multidisiplin Bangsa* 1.12 (2025): 2206-2214.

Usman, Ansharullah Patiroi, Rizki Yusliana Bakti, and Muhyiddin Am Hayat. "Optimalisasi Sistem Keamanan SSH dari Serangan Brute Force Mengunakan Intrusion Prevention System pada Mikrotik." *Arus Jurnal Sains dan Teknologi* 2.1 (2024): 116-122.

Cahyani, Ika Dwi. "Sistem keamanan enkripsi secure shell (ssh) untuk keamanan data." *Semarang: Universitas Pandanaran* (2010).

Waliana, Victor Gilbert Danielo, and Rissal Efendi. "KINERJA MODEL DEEP LEARNING UNTUK DETEKSI SERANGAN SSH BRUTE FORCE." *JATI (Jurnal Mahasiswa Teknik Informatika)* 10.3 (2026): 3783-3790.

Menteng, Constantin, Arief Setyanto, and Hanif Al Fatta. "MODEL DETEKSI SERANGAN SSH-BRUTE FORCE BERDASARKAN DEEP BELIEF NETWORK." *Jurnal Teknologi Informasi: Jurnal Keilmuan dan Aplikasi Bidang Teknik Informatika* 17.2 (2023): 101-110.

SUBHAN, AHMAD. *ANALISA POLA SERANGAN BRUTE FORCE ATTACK PADA PORT SSH*. Diss. UNIVERSITAS BINA DARMA, 2023.

Putra, Dewa Made Julijati, et al. "Analisis Perbandingan Serangan Hydra, Medusa dan Ncrack pada Password Attack." *Jurnal Informatika Teknologi dan Sains (JINTEKS)* 4.4 (2022): 461-466.