# BAB I
# PENDAHULUAN

## 1.1 Latar Belakang

Serangan *Distributed Denial of Service* (DDoS) merupakan salah satu ancaman utama dalam keamanan jaringan modern karena mampu melumpuhkan ketersediaan layanan dengan membanjiri target menggunakan lalu lintas data dalam jumlah besar secara simultan. Serangan ini menyebabkan sumber daya sistem tidak mampu melayani pengguna yang sah, dan menjadi semakin sulit dideteksi seiring meningkatnya kompleksitas jaringan serta volume trafik. Berdasarkan laporan terbaru, tercatat lebih dari 20,5 juta serangan DDoS terjadi dalam satu kuartal pada tahun 2025, meningkat hingga 358% dibandingkan tahun sebelumnya [1]. Kondisi ini menunjukkan eskalasi ancaman yang signifikan, sementara pendekatan deteksi tradisional berbasis *signature* dan *rule-based* masih memiliki keterbatasan dalam mengenali pola serangan baru yang bersifat dinamis [2].

Penelitian mengenai pemanfaatan distributed stream processing dalam deteksi serangan DDoS terus berkembang seiring meningkatnya kebutuhan akan sistem keamanan jaringan yang responsif dan skalabel. Sejumlah studi mengusulkan kerangka kerja klasifikasi berbasis pemrosesan aliran terdistribusi menggunakan *framework* seperti Apache Spark, Apache Storm, dan Apache Flink [3]. Di antara *framework* tersebut, Apache Spark banyak digunakan karena mendukung *distributed in-memory processing* untuk pemrosesan *data streaming* secara efisien [4]. Namun demikian, sebagian besar penelitian masih berfokus pada peningkatan performa model klasifikasi melalui berbagai teknik *Machine Learning* seperti *Naive Bayes*, *Decision Tree*, *Logistic Regression*, dan *Random Forest*. Di antara teknik tersebut, *Random Forest* menunjukkan performa yang baik dalam menangani data berdimensi tinggi serta mampu mengurangi risiko *overfitting* melalui pendekatan *ensemble learning* [4], [5].

Keunggulan model klasifikasi saja belum cukup untuk menjamin keberhasilan penerapan dalam lingkungan nyata, terutama apabila tidak disertai implementasi *pipeline streaming* yang mampu menangani trafik dinamis secara terukur. Oleh karena itu, integrasi antara model *machine learning* dan arsitektur stream processing perlu dievaluasi secara sistematis untuk mencapai deteksi DDoS secara *near real-time* dengan *latency* yang terkendali. Namun, kajian mengenai implementasi *end-to-end* beserta evaluasi performa berdasarkan *latency* dan *throughput* masih terbatas [6].

© Karya Dilindungi UU Hak Cipta
1. Dilarang menyebarluaskan dokumen tanpa izin.
2. Dilarang melakukan plagiasi.
3. Pelanggaran diberlakukan sanksi sesuai peraturan UU Hak Cipta.

Untuk menjawab kebutuhan tersebut, penelitian ini mengusulkan pengembangan pipeline pemrosesan *data streaming end-to-end* berbasis integrasi Apache Kafka dan Apache Spark Structured Streaming yang terintegrasi dengan algoritma *Random Forest* untuk mendeteksi serangan DDoS secara *near real-time*, serta melakukan evaluasi performa sistem berdasarkan metrik *latency* pemrosesan dan *throughput* aktual.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, rumusan masalah pada penelitian ini adalah:

1. Proses deteksi serangan DDoS menggunakan *distributed stream processing* masih menghadapi tantangan dalam memberikan respon deteksi secara cepat terhadap trafik jaringan yang berubah secara terus-menerus. Kondisi ini menyebabkan serangan berpotensi terlambat terdeteksi sehingga dapat mengganggu ketersediaan layanan jaringan.
2. Belum diketahui kemampuan *pipeline data streaming* dalam menangani variasi *input rate* trafik jaringan. Kondisi ini menyebabkan belum dapat dipastikan apakah *pipeline* mampu memproses data secara stabil ketika jumlah trafik meningkat.

## 1.3 Tujuan

Tujuan dari penelitian ini adalah untuk menghasilkan *pipeline data streaming* yang mampu memproses data trafik jaringan secara stabil dan mendeteksi serangan DDoS dengan jeda waktu yang rendah.

## 1.4 Manfaat

Manfaat yang dapat diperoleh dari tugas akhir ini adalah:

1. Integrasi pipeline pemrosesan data streaming berbasis Apache Kafka dan Apache Spark Structured Streaming dengan algoritma *Random Forest* mendukung pengembangan metode deteksi serangan DDoS secara *near real-time*.
2. Implementasi pipeline pemrosesan *data streaming end-to-end* memberikan gambaran arsitektur sistem deteksi serangan DDoS yang dapat dijadikan referensi dalam pengembangan sistem berbasis *distributed stream processing*.
3. Hasil penelitian ini dapat menjadi dasar untuk pengembangan lebih lanjut, seperti peningkatan metode *machine learning*, pengujian pada skala infrastruktur yang lebih besar, serta integrasi dengan mekanisme mitigasi serangan otomatis.

© Karya Dilindungi UU Hak Cipta
1. Dilarang menyebarluaskan dokumen tanpa izin.
2. Dilarang melakukan plagiasi.
3. Pelanggaran diberlakukan sanksi sesuai peraturan UU Hak Cipta.

## 1.5 Ruang Lingkup

Agar penelitian ini tetap terarah dan sesuai dengan tujuan yang telah ditetapkan, maka ruang lingkup penelitian ini dibatasi pada beberapa hal sebagai berikut:

1. Penelitian ini difokuskan pada implementasi *pipeline* pemrosesan data *streaming end-to-end* berbasis Apache Kafka sebagai *message broker* dan Apache Spark Structured Streaming sebagai *engine* pemrosesan data.
2. Metode *machine learning* yang digunakan dalam penelitian ini adalah algoritma *Random Forest*.
3. Data yang digunakan dalam penelitian ini adalah dataset trafik jaringan CIC-DDoS 2019 yang dikembangkan oleh Canadian Institute for Cybersecurity [7], dengan fokus pada beberapa jenis serangan DDoS seperti UDP, SYN, dan LDAP, serta data trafik normal yang digunakan untuk proses pelatihan dan pengujian model.
4. Evaluasi sistem dilakukan melalui *performance benchmarking* terhadap arsitektur monitoring yang dibangun, dengan mengukur metrik *latency* dan *throughput* selama proses pemrosesan data streaming bekerja [3], [8].
5. Penelitian ini tidak membahas mekanisme mitigasi atau penanggulangan serangan DDoS, melainkan hanya berfokus pada proses deteksi serangan pada lingkungan jaringan, tanpa mengembangkan sistem monitoring secara utuh.
6. Algoritma *machine learning* digunakan sebagai komponen klasifikasi yang diintegrasikan ke dalam *pipeline*, bukan sebagai fokus utama untuk peningkatan akurasi.

© Karya Dilindungi UU Hak Cipta
1. Dilarang menyebarluaskan dokumen tanpa izin.
2. Dilarang melakukan plagiasi.
3. Pelanggaran diberlakukan sanksi sesuai peraturan UU Hak Cipta.