|  |  |  |
| --- | --- | --- |
| Topik Capstone | **Topik Capstone** | |
| Siklus / Tahun | **\*Gasal atau Genap / (tahun)** | |
| Judul Dokumen | **Capstone TA**  Judul Capstone Proyek kelompok | |
| Jenis Dokumen | **SPESIFIKASI** Catatan: Penggunaan dan penyebaran dokumen ini dikendalikan oleh Departemen Teknik Komputer Universitas Diponegoro | |
| Nomor Dokumen | **(C200.[NoRev]TA[tahun].[1/2].[KodeKelompok]** | |
| Nomor Revisi | **NoRev** | |
| Nama File | **Kode Kelompok.pdf** | |
| Tanggal Penerbitan | **Tanggal Penerbitan** | |
| Unit Penerbit | **Departemen Teknik Komputer Universitas Diponegoro** | |
| Jumlah Halaman | **Jumlah Halaman** | Tidak termasuk sampul |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Pengusul** | | | | |
| Pengusul | Nama |  | Jabatan | Anggota |
|  | NIM |  |  |  |
|  | Tanggal |  | Tanda Tangan |  |
|  | Nama |  | Jabatan | Anggota |
|  | NIM |  |  |  |
|  | Tanggal |  | Tanda Tangan |  |
|  | Nama |  | Jabatan | Anggota |
|  | NIM |  |  |  |
|  | Tanggal |  | Tanda Tangan |  |
| Pembimbing 1 (Utama) | Nama |  | Tanda Tangan |  |
|  | NIP. |  |  |  |
|  | Tanggal |  |  |  |
| Pembimbing 2 | Nama |  | Tanda Tangan |  |
|  | NIP. |  |  |  |
|  | Tanggal |  |  |  |

|  |  |
| --- | --- |
| Versi, Tanggal, Oleh | Perbaikan |
|  |  |

**Daftar Isi**

[1. Pendahuluan 6](#_Toc188031104)

[1.1. Ringkasan isi Dokumen 6](#_Toc188031105)

[1.2. Aplikasi Dokumen 6](#_Toc188031106)

[1.3. Referensi 7](#_Toc188031107)

[1.4. Daftar Singkatan 7](#_Toc188031108)

[2. Spesifikasi Sistem dan Pengguna 7](#_Toc188031109)

[2.1. Gambaran Sistem 7](#_Toc188031110)

[2.2. Batasan Sistem 8](#_Toc188031111)

[2.3. Kebutuhan Fungsional 9](#_Toc188031112)

[2.3.1. Deskripsi Fitur Utama 9](#_Toc188031113)

[2.3.2. Alur Proses Utama 10](#_Toc188031114)

[2.4. Kebutuhan Data 15](#_Toc188031115)

[2.5. Kebutuhan Non Fungsional 16](#_Toc188031116)

[2.5.1. Kinerja Sistem 17](#_Toc188031117)

[2.5.2. Keamanan Sistem 18](#_Toc188031118)

[2.5.3. Keandalan Sistem 18](#_Toc188031119)

[2.5.4. Kebutuhan Skalabilitas 19](#_Toc188031120)

[2.5.5. Kebutuhan Teknologi dalam Lingkungan Operasional 20](#_Toc188031121)

[2.6. Kebutuhan Arsitektur Sistem 22](#_Toc188031122)

[2.7. Karakteristik Pengguna / Role & Permission 22](#_Toc188031123)

[2.8. Target Sistem 23](#_Toc188031124)

[3. Kebutuhan Antarmuka Sistem 24](#_Toc188031125)

[3.1. Antarmuka Perangkat Keras 24](#_Toc188031126)

[3.2. Antarmuka Perangkat Lunak 25](#_Toc188031127)

[3.3. Antarmuka Komunikasi 25](#_Toc188031128)

[3.4. Kebutuhan Integrasi dengan Sistem Lain (*Opsional*) 25](#_Toc188031129)

# Pendahuluan

## Ringkasan isi Dokumen

*Jelaskan dengan singkat mengapa dokumen ini dibuat. Sertakan pernyataan tujuan spesifik, seperti mendefinisikan ruang lingkup sistem, mendokumentasikan kebutuhan pengguna, dan memastikan pemahaman yang seragam di antara tim pengembang dan pemangku kepentingan.*

**Contoh:**

Dokumen ini disusun untuk memberikan deskripsi mendetail tentang spesifikasi sistem "Sistem Monitoring dan Deteksi Serangan Slow DoS/DRDoS Berbasiskan Ensemble Learning menggunakan ELK Stack". Sistem ini dirancang untuk Ditreskrimsus Polda Jateng Semarang dengan tujuan:

* Mengidentifikasi dan memitigasi serangan Slow DoS dan DRDoS pada infrastruktur jaringan.
* Memberikan informasi real-time mengenai aktivitas mencurigakan.
* Menghasilkan laporan analitik yang dapat digunakan untuk investigasi lebih lanjut.

Dokumen ini bertujuan untuk memastikan keselarasan antara kebutuhan pengguna dan implementasi sistem melalui spesifikasi yang terperinci.

## Aplikasi Dokumen

*Dokumen ini berlaku sebagai technical specification design yang berfungsi untuk menjelaskan:*

1. *Gambaran proses rekayasa atau proses bisnis yang diakomodasi oleh system, termasuk ruang lingkup sistem.*
2. *Target efisiensi/efektifitas yang ingin dicapai.*
3. *Dokumentasi dari spesifikasi sistem yang dibangun*

## Referensi

1. P. Citra, “Kombinasi Sobel , Canny Dan Otsu Untuk Segmentasi Citra,” Technologia, vol. 13, no. 2, hal. 102–107, 2022.
2. S. C. Re, K. Daya, D. Tanggung, dan J. Perusahaan, “Keselamatan dan Kesehatan Kerja,” 2013, Diakses: Feb 10, 2023 Tersedia pada: www.ifrro.org.
3. “Raspberry Pi 3 Model B+,” Diakses: Feb 10, 2023. Tersedia pada: [www.raspberrypi.org/products/raspberry](https://www.raspberrypi.org/products/raspberry).
4. IEA Constituent Agreements, 2021. Graduate Attributes and Professional Competences. Available at: https://www.ieagreements.org/assets/Uploads/IEA-Graduate-Attributes-and-Professional-Competencies-2021.1-Sept-2021.pdf [Accessed 13 January 2025].

## Daftar Singkatan

|  |  |
| --- | --- |
| **Akronim** | **Pengertian** |
| DoS | Denial of Service |
| DDoS | Distributed Denial of Service |
| IDS | Intrusion Detection System |
| IPS | Intrusion Prevention System |

# Spesifikasi Sistem dan Pengguna

## Gambaran Sistem

*Jelaskan secara singkat tujuan dari sistem yang akan dibangun. Nyatakan alasan utama mengapa sistem ini dirancang dan apa manfaat utama yang diharapkan dari penerapan sistem.*

**Contoh:**

Sistem Monitoring dan Deteksi Serangan Slow DoS/DRDoS berbasis Ensemble Learning menggunakan ELK Stack dirancang untuk meningkatkan keamanan jaringan Ditreskrimsus Polda Jateng. Sistem ini mengintegrasikan teknologi ELK Stack dengan model machine learning berbasis Ensemble Learning untuk memberikan kemampuan deteksi dini terhadap serangan jaringan. Data log jaringan dikumpulkan dari berbagai sumber, seperti firewall, router, dan server aplikasi, melalui Logstash. Logstash memproses dan memformat data sebelum diteruskan ke Elasticsearch, yang bertugas menyimpan log dalam indeks terstruktur untuk memudahkan pencarian dan analisis data dalam jumlah besar.

Proses deteksi ancaman dilakukan dengan menerapkan model Ensemble Learning yang menggabungkan algoritma seperti Random Forest dan Gradient Boosting. Model ini menganalisis log yang telah disimpan di Elasticsearch untuk mengidentifikasi pola-pola yang mencurigakan, seperti karakteristik serangan Slow DoS atau DRDoS. Hasil analisis ditampilkan melalui Kibana, yang menyediakan visualisasi interaktif dalam bentuk dashboard. Dashboard ini dirancang untuk menampilkan statistik aktivitas jaringan, grafik anomali, serta laporan serangan yang terdeteksi. Selain itu, sistem dilengkapi dengan mekanisme notifikasi real-time yang mengirimkan peringatan kepada admin Ditreskrimsus melalui email atau aplikasi pesan untuk memastikan respons cepat terhadap potensi ancaman.

Keunggulan sistem ini meliputi kemampuan deteksi ancaman dengan akurasi tinggi, skalabilitas yang mendukung pertumbuhan data log yang besar, dan kemudahan integrasi dengan infrastruktur jaringan yang sudah ada di Ditreskrimsus Polda Jateng. Sistem ini dirancang tidak hanya untuk memantau aktivitas jaringan secara real-time, tetapi juga untuk memberikan data analitik yang mendalam guna mendukung investigasi forensik digital. Dengan mengadopsi sistem ini, Ditreskrimsus Polda Jateng diharapkan dapat meningkatkan efektivitas operasional mereka dalam mendeteksi dan mengatasi serangan siber.

## Batasan Sistem

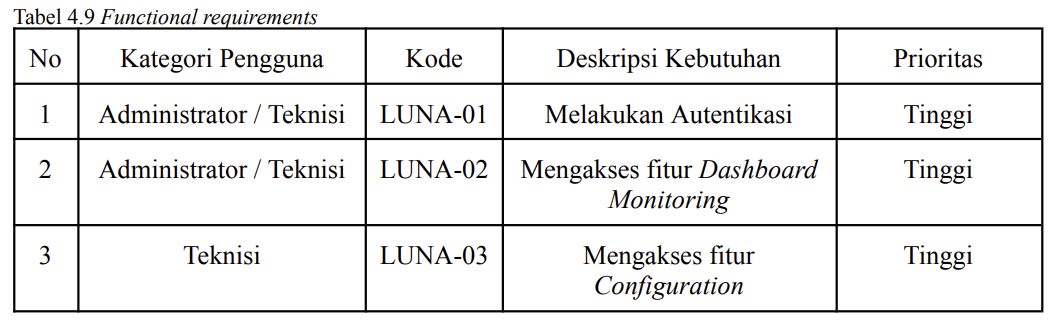
**Contoh:**

1. Pendeteksian DDoS Terbatas pada Layanan Tertentu: Sistem hanya akan mendeteksi serangan DDoS pada perangkat atau layanan IoT yang terhubung dengan jaringan dan tidak mencakup serangan pada infrastruktur jaringan secara umum.
2. Model AI Terbatas pada Jenis Serangan Tertentu: Sistem menggunakan model AI yang terlatih untuk mendeteksi serangan DDoS berbasis pada pola lalu lintas data. Model ini tidak akan mendeteksi jenis serangan lainnya, seperti serangan malware atau phishing.
3. Monitoring Perangkat IoT Tertentu: Hanya perangkat IoT yang terhubung dengan jaringan yang akan dimonitor. Perangkat yang tidak terhubung atau tidak mendukung protokol komunikasi yang kompatibel tidak akan dimonitor.

## Kebutuhan Fungsional

*Sub-bab ini akan menjelaskan apa yang harus dilakukan oleh sistem untuk memenuhi kebutuhan yang telah dipaparkan pada Gambaran Sistem. Spesifikasi ini berfokus pada fungsi yang akan disediakan oleh sistem, yaitu tindakan atau layanan yang harus mampu dilakukan oleh sistem yang akan dibangun, tanpa membahas secara detail bagaimana fungsi tersebut akan diimplementasikan secara teknis.*

*Lengkap dengan list tabel Kebutuhan Fungsional.*

**

### Deskripsi Fitur Utama

*Deskripsikan / gambarkan secara jelas tentang fitur inti pada sistem yang akan dibangun. Lebih baik jika disertai dengan manfaatnya masing-masing.*

**Contoh:**

Luna System digunakan untuk mendeteksi serangan Slow DoS/DRDoS menggunakan pembelajaran ensemble dan Elk Stack. Adapun fungsi utama di dalamnya adalah sebagai berikut

* 1. Dashboard dan Visualisasi Data

Deskripsi: Dashboard akan disajikan dalam bentuk website. Dashboard ini akan berisi data atau log riwayat serangan DoS/DRDoS serta beberapa visualisasi data yang akan mempermudah administrator dalam melakukan monitoring sistem.

Manfaat: …

* 1. Notifikasi real-time

Deskripsi: Sistem ini akan menggunakan aplikasi pesan yaitu Telegram untuk mengirimkan notifikasi kepada administrator. Ketika terjadi serangan Slow DoS/DRDoS, sistem akan mendeteksinya dan akan langsung mengirimkan notifikasi ke aplikasi tersebut.

Manfaat: …

### Alur Proses Utama

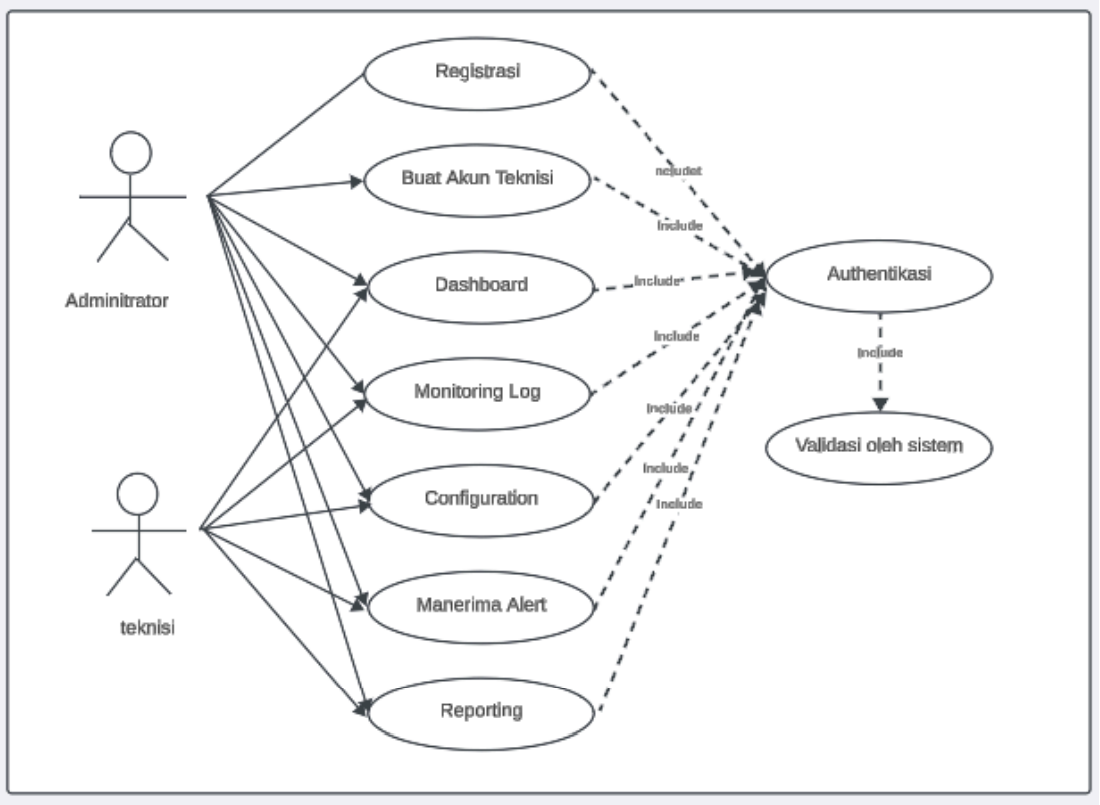
*Jelaskan alur bisnis bagaimana sistem bekerja, dilengkapi dengan flowchart secara detail, lebih baik lengkap dengan alur negatifnya. Dapat diperjelas menggunakan Usecase Diagram, Usecase Scenario, dan Activity Diagram/Business Process Model and Notation (BPMN) untuk tiap bisnis proses utama.*

*Berlaku juga bagi yang memiliki project hardware terkait embedded system / IoT.*

Contoh:

#### Usecase Diagram

Dalam pengembangan Luna System ini terdapat enam Diagram Use Case sesuai dengan enam fitur utama yang dikembangkan yaitu Monitoring Dashboard, Monitoring Log, Configuration Device, Menerima Alert, dan Reporting.



Gambar 1. Usecase Diagram

#### *Usecase scenario*

*Usecase scenario* dari Diagram Use Case Luna System memiliki total 5 bagian untuk setiap fitur utama yaitu Monitoring Dashboard, Monitoring Log, Configuration Device, Menerima Alert, Reporting.

Tabel 1. Tabel *Use Case Scenario* Autentikasi

|  |  |  |
| --- | --- | --- |
| *Use Case* ID *Number* | 1 | |
| *Use Case Name* | Autentikasi | |
| *Use Case Description* | Menggambarkan proses autentikasi. | |
| *Primary Actor* | Teknisi, Administrator. | |
| *Pre*-*Condition* | *User* belum *login* ke sistem. | |
| *Primary Flow of Events* | *User Action* | *System Response* |
| 1. *1. User* masuk ke halaman |  |
|  | 2. Sistem menampilkan *form login* |
| *3. User* memasukkan username dan *password* |  |
|  | 4. Sistem melakukan validasi data |
| 5. Setelah data valid, *User* melakukan *login* |  |
|  | 6. Sistem menampilkan halaman *dashboard*. |
| *7. User* berada di halaman *monitoring dashboard*. |  |
| *Error Flow of Events* | 4a. *User* memberikan data *login* yang tidak sesuai. |  |
|  | 4b. Sistem menolak proses *login* dan menampilan pesan *error*. |
| *Post Condition* | *User* berhasil *login*. | |

Tabel 2. Tabel *Use Case Scenario* Membuat Akun Teknisi

|  |  |  |
| --- | --- | --- |
| *Use Case ID Number* | *2* | |
| *Use Case Name* | *Buat akun teknisi* | |
| *Use Case Description* | *Menggambarkan  proses dibuatnya akun teknisi* | |
| *Primary Actor* | *Administrator* | |
| *Secondary Actor* | *Teknisi* | |
| *Pre-Condition* | *Administrator telah login ke sistem.* | |
| *Primary Flow of Events* | *User Action* | *System Response* |
| *1. Admin masuk ke menu user management* |  |
|  | *2. Sistem menampilkan menu teknisi* |
| *3. Admin memilih menu “tambahkan teknisi”* |  |
|  | *4. Sistem mengarahkan ke halaman buat teknisi baru.* |
| *5. Admin memasukkan data teknisi (username, password)* |  |
|  | *6. Sistem menyimpan data dan menambahkan menjadi teknisi baru. Teknisi siap menggunakan akun.* |
| *Error Flow of Events* | *5a. Admin memberikan data yang tidak sesuai.* |  |
|  |  | *5b. Sistem menolak proses pembuatan akun teknisi dan menampilkan pesan error.* |
| *Post Condition* | *Akun teknisi bisa digunakan* | |

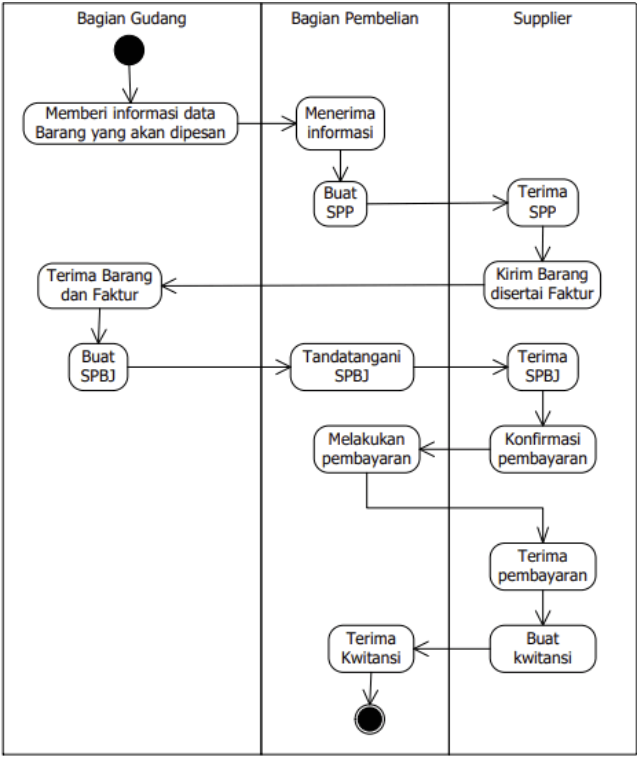
Tabel 3. Tabel *Use Case Scenario* Menampilkan Dashboard

|  |  |  |
| --- | --- | --- |
| *Use Case* ID *Number* | 3 | |
| *Use Case Name* | Menampilkan Dashboard | |
| *Use Case Description* | Menggambarkan proses *monitoring dashboard* | |
| *Primary Actor* | Administrator, Teknisi | |
| *Secondary Actor* |  | |
| *Pre-Condition* | *User* telah *login* ke sistem | |
| *Primary Flow of Events* | *User Action* | *System Response* |
| *1. User* memilih menu *Dashboard* |  |
|  | 2. Sistem menampilkan menu untuk melihat  visual dari data-data hasil deteksi seperti grafik dan *activity log*. |
| *Error Flow of Events* |  | 2b.  *Dashboard* tidak menampilkan data karena belum dilakukan  deteksi pada  *traffic* jaringan. |
| *Post Condition* | *User* dapat melihat tampilan visual dari data-data hasil deteksi seperti grafik dan *activity log*. | |

Tabel 4. Tabel *Use Case Scenario Monitoring Log*

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Monitoring Log* | |
| *Use Case Description* | Meanggambarkan proses *monitoring Log* | |
| *Primary Actor* | Teknisi | |
| *Secondary Actor* | Administrator | |
| *Pre-Condition* | Teknisi telah *login* ke sistem | |
| *Primary Flow of Events* | *User Action* | *System Response* |
|  | 1. Sistem melakukan pengamatan pada lalu lintas jaringan, permintaan HTTP, atau kejadian keamanan. |
|  | 2. Sistemmelakukan Analisis pola lalu lintas *log* |
|  | 3. Sistemmenerapkan teknik deteksi anomali untuk membandingkan perilaku lalu lintas saat ini dengan *baseline* normalitas. |
| *4. User* mengamati proses *monitoring* dan melakukan penanganan apabila ditemukan serangan keamanan. |  |
| *Error Flow of Events* |  | 1a.  Sistemtidak menampilkan proses pengamatan pada lalu lintas jaringan, permintaan HTTP, atau kejadian keamanan dikarenakan belum berjalannya proses deteksi. |
| *Post Condition* | *User* mengamati proses, melakukan penanganan, dan melakukan perbaikan pada infrastruktur jika diperlukan. | |

#### Activity Diagram



Gambar 2. Contoh Activity Diagram

*Activity diagram,* dalam bahasa Indonesia diagram aktivitas, yaitu diagram yang dapat memodelkan proses-proses yang terjadi pada sebuah sistem. Runtutan proses dari suatu sistem digambarkan secara vertikal. *Activity diagram* merupakan pengembangan dari *Use Case* yang memiliki alur aktivitas.

Jelaskan alur secara deskriptif beserta aturannya jika ada.

## Kebutuhan Data

*Kebutuhan data merujuk pada semua jenis data yang diperlukan oleh sistem untuk berfungsi dengan baik. Bagian ini biasanya menjelaskan secara rinci tentang data apa saja yang harus dikumpulkan, diolah, disimpan, dan dikelola oleh sistem untuk memenuhi tujuan dan persyaratan yang ditetapkan.*

1. *Jenis Data: Mengidentifikasi tipe data yang akan digunakan, misalnya log jaringan, metadata, atau informasi konfigurasi.*

*Contoh: Data log dari firewall, server, dan perangkat jaringan lainnya.*

1. *Sumber Data: Menjelaskan dari mana data tersebut berasal.*

*Contoh: Data log berasal dari perangkat seperti router, switch, atau server.*

1. *Format Data: Format atau struktur data yang harus digunakan untuk memastikan kompatibilitas.*

*Contoh: JSON, CSV, atau format log lainnya.*

1. *Volume Data: Estimasi jumlah data yang akan dikelola oleh sistem, baik dalam ukuran file (MB, GB) maupun jumlah record.*

*Contoh: Sistem harus mampu menangani minimal 1.000 log per detik.*

1. *Frekuensi Pengumpulan Data: Menjelaskan seberapa sering data akan diperbarui atau dikumpulkan.*

*Contoh: Data dikumpulkan secara real-time atau dalam interval tertentu (misalnya, setiap 5 menit).*

1. *Penyimpanan Data: Kebutuhan penyimpanan, termasuk kapasitas penyimpanan dan durasi retensi data.*

*Contoh: Data log harus disimpan selama 6 bulan untuk keperluan audit dan analisis.*

1. *Keamanan Data: Persyaratan perlindungan terhadap data untuk memastikan kerahasiaan, integritas, dan ketersediaannya.*

*Contoh: Semua data harus dienkripsi saat transit dan saat disimpan.*

1. *Kebutuhan Pemrosesan Data: Menjelaskan bagaimana data akan diolah, seperti analisis, transformasi, atau agregasi.*

*Contoh: Data log diproses melalui Logstash untuk parsing dan dikirim ke Elasticsearch untuk analisis lebih lanjut.*

**Contoh:**

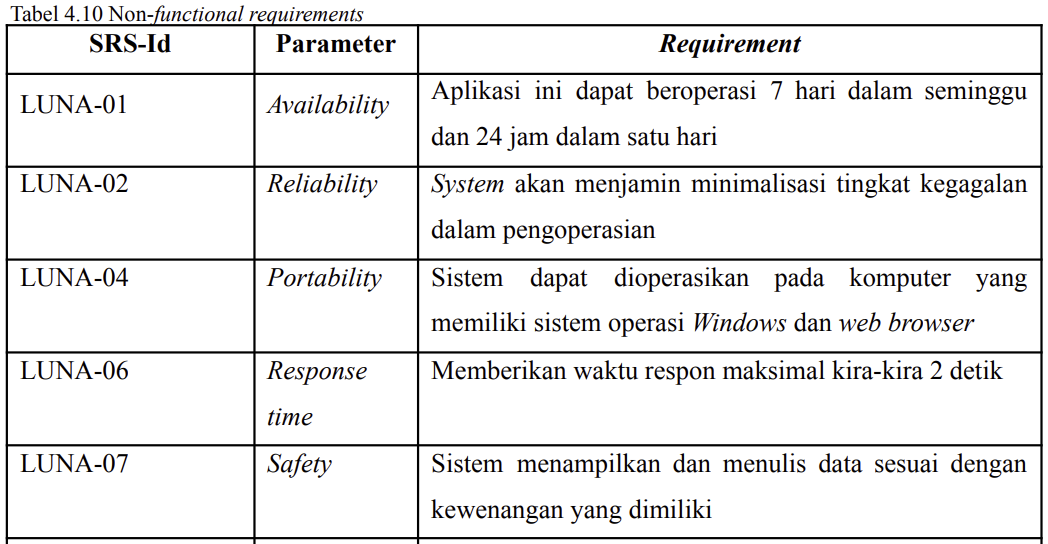
Dalam kasus Sistem Monitoring dan Deteksi Serangan Slow DoS/DRDoS, kebutuhan data mencakup:

* Data log yang mencakup informasi tentang alamat IP sumber, waktu akses, jenis permintaan, dan status respons.
* Data pelatihan untuk model Ensemble Learning, seperti pola serangan sebelumnya atau data normal dari aktivitas jaringan.
* Informasi konfigurasi, seperti aturan firewall atau daftar putih/daftar hitam IP.

## Kebutuhan Non Fungsional

*Spesifikasi non-fungsional berfokus pada "****bagaimana sistem beroperasi****" dan kualitas yang harus dimiliki oleh sistem untuk memenuhi kebutuhan pengguna dan stakeholder.*

*Lengkap dengan list tabel kebutuhan Non Fungsional.*

**

**Contoh:**

Spesifikasi non-fungsional berfokus pada bagaimana sistem beroperasi serta kualitas yang harus dimiliki oleh sistem untuk memenuhi kebutuhan pengguna dan stakeholder. Pada sistem monitoring dan deteksi serangan Slow DoS/DRDoS berbasis Ensemble Learning menggunakan ELK Stack, beberapa spesifikasi non-fungsional yang perlu dipertimbangkan adalah:

### Kinerja Sistem

*Deskripsi seperti waktu respons maksimum, throughput, atau kebutuhan bandwidth.*

**Contoh:**

* Waktu Respons Maksimum: Sistem harus mampu memberikan respons dalam waktu kurang dari 5 detik untuk setiap deteksi serangan yang terjadi pada jaringan yang dipantau. Waktu respons ini penting agar petugas di Ditreskrimsus Polda Jateng dapat segera melakukan tindakan mitigasi terhadap serangan yang terdeteksi.
* Throughput: Sistem diharapkan mampu memproses data log jaringan yang masuk dengan throughput minimal 1.000 log per detik. Hal ini memastikan bahwa sistem dapat menangani volume data yang tinggi dari berbagai sumber perangkat dan server yang ada di lingkungan Ditreskrimsus Polda Jateng.
* Kebutuhan Bandwidth: Sistem monitoring ini memerlukan bandwidth minimal 100 Mbps untuk memastikan data yang dikirimkan dari berbagai perangkat jaringan dapat diterima dan dianalisis secara real-time tanpa menyebabkan latensi atau penurunan kualitas deteksi serangan.

### Keamanan Sistem

Persyaratan keamanan seperti enkripsi data atau otentikasi multi-faktor.

**Contoh:**

Keamanan sistem sangat penting, mengingat sistem ini beroperasi dalam konteks jaringan yang sensitif dan rentan terhadap serangan. Beberapa persyaratan keamanan yang harus dipenuhi oleh sistem adalah:

* Enkripsi Data: Semua data yang dikirimkan antara perangkat, server, dan aplikasi monitoring harus dienkripsi menggunakan protokol TLS/SSL untuk menjaga kerahasiaan dan integritas data. Hal ini untuk melindungi data yang dapat berisi informasi sensitif, seperti log dan metadata terkait serangan.
* Otentikasi Multi-Faktor (MFA): Untuk meningkatkan keamanan akses ke antarmuka administrasi sistem, sistem harus menggunakan otentikasi multi-faktor, yang mengharuskan pengguna untuk memasukkan kombinasi informasi yang lebih dari sekadar username dan password (misalnya, kode OTP atau autentikasi berbasis biometrik).
* Pengelolaan Akses Berbasis Peran (RBAC): Sistem harus mengimplementasikan pengelolaan akses berbasis peran untuk memastikan bahwa hanya pengguna yang memiliki izin yang tepat yang dapat mengakses fungsionalitas sensitif, seperti pengaturan deteksi serangan atau analisis data.

### Keandalan Sistem

*Tingkat ketersediaan atau waktu kerja (uptime) yang diharapkan.*

**Contoh:**

Keandalan sistem berfokus pada ketersediaan dan waktu operasi sistem yang tinggi, yang penting untuk memastikan sistem dapat berfungsi secara terus-menerus tanpa gangguan signifikan.

* Uptime: Sistem harus memiliki tingkat ketersediaan atau uptime sebesar 99.9%, yang berarti sistem hanya boleh mengalami downtime maksimal 8 jam per tahun. Hal ini untuk memastikan bahwa sistem monitoring selalu tersedia untuk mendeteksi dan menangani serangan secara real-time.
* Pemulihan dari Gangguan (Disaster Recovery): Sistem harus dilengkapi dengan rencana pemulihan bencana yang memungkinkan pemulihan dalam waktu maksimal 2 jam jika terjadi kegagalan sistem besar atau kehilangan data.

### Kebutuhan Skalabilitas

Penjelasan kemampuan sistem untuk berkembang dengan meningkatnya jumlah pengguna.

**Contoh:**

Sistem ini harus mampu berkembang seiring dengan meningkatnya jumlah perangkat yang terhubung ke jaringan dan volume data yang dihasilkan. Kebutuhan skalabilitas mencakup beberapa aspek:

* Skalabilitas Horizontal: Sistem harus dirancang agar dapat melakukan skalabilitas horizontal, memungkinkan penambahan server atau node baru ke dalam klaster ELK Stack jika volume data meningkat atau jumlah perangkat yang dipantau bertambah. Dengan demikian, sistem dapat menangani beban yang lebih tinggi tanpa penurunan kinerja.
* Skalabilitas Proses Deteksi: Sistem harus dapat menyesuaikan jumlah dan jenis model machine learning dalam ensemble learning berdasarkan jumlah data dan kompleksitas serangan. Sebagai contoh, jika data log yang masuk meningkat, maka jumlah model ensemble yang digunakan dapat ditingkatkan untuk menjaga akurasi deteksi.
* Integrasi dengan Sistem Eksternal: Sistem harus mampu berintegrasi dengan sistem keamanan eksternal lainnya, seperti firewall, IDS/IPS, dan sistem manajemen jaringan lainnya, agar dapat menangani peningkatan jumlah perangkat atau perubahan kebijakan keamanan tanpa gangguan operasional.

### Kebutuhan Teknologi dalam Lingkungan Operasional

*Bab ini menggambarkan berbagai* ***teknologi*** *yang dibutuhkan untuk mendukung pengembangan dan operasi sistem yang sedang dirancang pada lingkungan operasional.* ***Kebutuhan teknologi*** *mencakup perangkat lunak, perangkat keras, dan lingkungan operasional yang diperlukan agar sistem dapat berfungsi dengan baik.*

*Penjelasan lingkungan operasional*

#### Perangkat Lunak

Perangkat lunak yang diperlukan untuk mendukung pengembangan, operasi, dan pemeliharaan sistem. Ini mencakup sistem operasi, perangkat lunak aplikasi, pustaka, framework, serta alat bantu lain yang diperlukan untuk mengembangkan dan menjalankan fungsionalitas sistem.

* Sistem Operasi:
  + Linux (Ubuntu, CentOS) digunakan sebagai sistem operasi untuk server pusat yang menjalankan sistem monitoring dan analitik.
  + Windows Server bisa digunakan jika ada kebutuhan tertentu untuk aplikasi berbasis Windows.
* Perangkat Lunak AI:
  + TensorFlow atau PyTorch untuk membangun model AI yang digunakan untuk mendeteksi pola serangan DDoS.
  + Scikit-learn untuk implementasi algoritma machine learning lainnya, seperti analisis trafik jaringan.
* Perangkat Lunak Jaringan:
  + Wireshark untuk pemantauan dan analisis trafik jaringan.
  + Zabbix atau Nagios untuk pemantauan jaringan secara real-time.
* Database:
  + MySQL atau PostgreSQL untuk menyimpan data yang berkaitan dengan status perangkat IoT, serangan DDoS, dan log lainnya.
  + InfluxDB atau Elasticsearch untuk menyimpan data time-series dari perangkat IoT yang terus-menerus mengirim data.
* Alat Analitik & Visualisasi:
  + Grafana untuk visualisasi data real-time dari perangkat IoT dan status DDoS.
  + Kibana (untuk Elasticsearch) untuk menganalisis dan menampilkan log serta data yang relevan dengan serangan DDoS

#### Perangkat Keras

Perangkat keras yang diperlukan untuk menjalankan dan mendukung perangkat lunak dan fungsi sistem. Ini mencakup server, perangkat IoT, serta perangkat pendukung lain seperti sensor, dan perangkat jaringan.

* Server:
  + Server pusat untuk menjalankan aplikasi monitoring dan model AI. Server ini bisa menggunakan spesifikasi seperti:
    - CPU: Intel Xeon atau AMD EPYC untuk pemrosesan data besar dan pembelajaran AI.
    - RAM: 32GB atau lebih, tergantung pada kebutuhan pemrosesan.
    - Penyimpanan: SSD untuk kecepatan akses data tinggi, 1TB atau lebih.
* Perangkat IoT:
  + Sensor jaringan dan perangkat IoT yang digunakan untuk mengumpulkan data trafik dari perangkat yang terhubung di jaringan, seperti sensor IoT Edge Devices atau IoT Gateways yang terhubung ke perangkat lain di jaringan.
  + Raspberry Pi atau ESP32 yang digunakan untuk mengumpulkan data dari perangkat IoT dan mengirimkannya ke server pusat.
* Perangkat Jaringan:
  + Switch dan Router untuk menghubungkan perangkat IoT ke server pusat dan mengelola lalu lintas data.
  + Firewall hardware atau IDS/IPS (Intrusion Detection/Prevention Systems) yang dapat diintegrasikan dengan sistem monitoring untuk melakukan mitigasi otomatis terhadap serangan DDoS yang terdeteksi.
* Perangkat AI dan Komputasi:
  + GPU (Graphics Processing Unit) untuk mempercepat proses pelatihan model AI jika diperlukan, terutama dalam penggunaan deep learning.
  + Edge computing devices untuk mengolah data lebih dekat ke sumbernya (di tempat perangkat IoT berada), mengurangi latensi.

## Kebutuhan Arsitektur Sistem

*Menyediakan gambaran umum komponen utama dan interaksi antar komponen sistem secara keseluruhan.*

**Contoh:**

Sistem ini dirancang untuk menyediakan gambaran umum tentang bagaimana sistem monitoring jaringan IoT dengan kemampuan deteksi DDoS. Sistem ini mengintegrasikan komponen utama jaringan IoT dengan algoritma kecerdasan buatan untuk mendeteksi dan merespons potensi ancaman secara otomatis.

## Karakteristik Pengguna / Role & Permission

*Penjelasan secara rinci siapa saja pengguna sistem yang diusulkan, peran mereka, dan izin akses yang dimiliki masing-masing peran. Informasi ini penting untuk memastikan bahwa setiap pengguna dapat mengakses fitur sistem sesuai dengan tanggung jawab dan kebutuhan mereka.*

* ***Identifikasi Pengguna***

*Sebutkan jenis-jenis pengguna sistem yang ada, berdasarkan peran atau fungsi mereka dalam organisasi. Jelaskan profil pengguna, seperti tingkat keahlian teknis, frekuensi penggunaan sistem, atau kebutuhan mereka terkait sistem.*

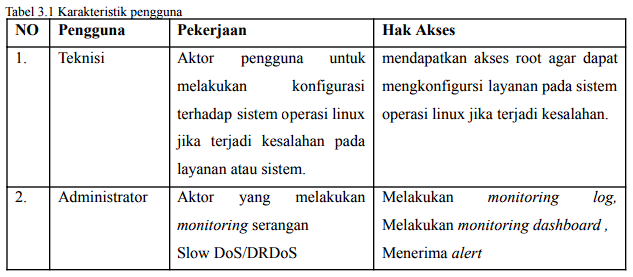
* ***Peran dan Tanggung Jawab (Role)***

*Uraikan peran masing-masing pengguna dalam sistem, termasuk tugas utama mereka. Gunakan istilah yang mudah dipahami untuk mendeskripsikan peran, seperti "Administrator", "Analyst", atau "Operator".*

* ***Hak Akses dan Izin (Permission)***

*Jelaskan hak akses yang diberikan kepada masing-masing peran. Deskripsikan fitur atau modul sistem yang dapat diakses oleh pengguna tertentu, serta batasan yang diterapkan untuk menjaga keamanan dan integritas data.*

Contoh:



1. Description of User : Pelaku

Role : Teknisi   
Prerequisite : Teknisi memerlukan akses root   
Task description : Melakukan konfigurasi layanan dan sistem jika terjadi kesalahan

## Target Sistem

*Jelaskan kondisi system existing seperti apa, lalu jelaskan target dari system yang dibuat, lebih baik jika dilengkapi dengan metric (service time) yang akan di optimalkan.*

**Contoh:**

Target service time pada projek Luna System yang sedang dikembangkan yaitu system monitoring yang dapat mengidentifikasi serangan Denial of Service/Distributed Denial of Service (Slow DoS/DRDoS) di Ditreskrimsus Polda Jateng yaitu:

1. Terbuatnya sistem untuk mendeteksi serangan Slow DoS/DRDoS yang sebelumnya belum ada pada Ditreskrimsus Polda Jateng.
2. Hasil deteksi serangan Slow DoS/DRDoS ke dashboard sistem dapat dilakukan dalam waktu kurang dari 10 detik dari waktu inisial serangan terjadi.
3. Dashboard sistem dapat memonitor log serangan Slow DoS/DRDoS secara real-time.
4. Dapat memberikan notifikasi serangan yang terjadi ke administrator secara real-time.

# Kebutuhan Antarmuka Sistem

*Merujuk pada deskripsi detail tentang bagaimana sistem akan berinteraksi dengan perangkat keras (hardware) dan perangkat lunak (software) lainnya. Bagian ini penting untuk memastikan bahwa sistem dapat diintegrasikan dengan lancar ke dalam lingkungan operasional yang ada.*

## Antarmuka Perangkat Keras

Pada sistem deteksi serangan Slow DoS/DRDoS dari Luna System yang diajukan, proses pengembang telah dirancang dengan mempertimbangkan beberapa aspek antarmuka perangkat keras yang diperlukan untuk memfasilitasi interaksi pengguna dengan sistem. Daftar beberapa hardware interface yang telah dipertimbangkan dalam perancangan sistem ini dijelaskan dalam Tabel dibawah

Tabel 4. Antarmuka Perangkat Keras

| **No** | **Antarmuka Pengguna** | **Fungsi** |
| --- | --- | --- |
| 1. | *Keyboard* | Antarmuka *keyboard* digunakan untuk memasukkan data ke dalam sistem. |
| 2. | *Mouse* | Antarmuka *mouse* digunakan untuk memindahkan objek dari satu tempat ke tempat lain. |
| 3. | *Monitor* | Antarmuka *monitor* digunakan untuk melihat tampilan dari sistem informasi. |

## Antarmuka **Perangkat Lunak**

Fokus pada kebutuhan yang berkaitan dengan interaksi antar perangkat lunak, library, atau database yang digunakan untuk pertukaran data antara aplikasi atau modul dalam sistem.

**Contoh:**

1. API RESTful untuk mengakses data log.
2. Integrasi dengan database (misalnya MySQL, Elasticsearch).
3. Kompatibilitas dengan system operasi tertentu (Linux, Windows).
4. Format data yang digunakan (JSON, XML).

## Antarmuka Komunikasi

*Fokus pada kebutuhan komunikasi di tingkat jaringan, seperti protokol, bandwidth, dan keamanan dalam pertukaran data antar perangkat atau sistem melalui jaringan.*

**Contoh:**

1. Protokol komunikasi yang digunakan, seperti HTTP, HTTPS, MQTT, atau WebSocket.
2. Keamanan komunikasi, seperti penggunaan TLS/SSL untuk enkripsi.
3. Kebutuhan bandwidth untuk mengelola volume data yang besar (misalnya, 1.000 log/detik).
4. Konektivitas antar perangkat melalui jaringan lokal (LAN) atau internet.

## Kebutuhan Integrasi dengan Sistem Lain (*Opsional*)

*Merujuk pada kemampuan sistem yang sedang dikembangkan untuk berinteraksi, berbagi data, atau bekerja sama dengan sistem atau perangkat lain dalam sebuah ekosistem teknologi. Tujuannya adalah memastikan interoperabilitas dan efisiensi dalam mencapai tujuan fungsional dan operasional.*

**Contoh:**

Dalam konteks *Sistem Monitoring dan Deteksi Serangan Slow DoS/DRDoS Berbasiskan Ensemble Learning*:

1. Integrasi dengan Sistem SIEM:
   * Sistem harus dapat mengirimkan data deteksi serangan ke platform SIEM (misalnya, Splunk atau QRadar) untuk korelasi dan pelaporan yang lebih luas.
2. Integrasi dengan Firewall atau IDS/IPS:
   * Sistem dapat mengirimkan perintah mitigasi serangan ke perangkat firewall untuk memblokir alamat IP yang mencurigakan.
3. Integrasi dengan Dashboard atau Platform Monitoring:
   * Hasil analisis dari sistem harus dapat diakses oleh pengguna melalui dashboard pihak ketiga, seperti Grafana atau Kibana.
4. Integrasi dengan Sistem Notifikasi:
   * Sistem perlu mengirimkan peringatan ke email gateway, SMS gateway, atau aplikasi pesan instan (misalnya, Slack atau WhatsApp).
5. Integrasi dengan Database:
   * Sistem menyimpan data log dan analisis di database yang ada, seperti Elasticsearch, MongoDB, atau PostgreSQL.