|  |  |  |
| --- | --- | --- |
| Topik Capstone | **Topik Capstone** | |
| Siklus / Tahun | **\*Gasal atau Genap / (tahun)** | |
| Judul Dokumen | **Capstone TA**  Judul Capstone Proyek kelompok | |
| Jenis Dokumen | **PROPOSAL**  Catatan: Penggunaan dan penyebaran dokumen ini dikendalikan oleh Departemen Teknik Komputer Universitas Diponegoro | |
| Nomor Dokumen | **C100.[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 5](#_Toc187828014)

[1.1. Ringkasan isi dokumen 5](#_Toc187828015)

[1.2. Aplikasi Dokumen 5](#_Toc187828016)

[1.3. Referensi 5](#_Toc187828017)

[1.4. Daftar Singkatan 6](#_Toc187828018)

[2. Proposal Pengembangan Produk 6](#_Toc187828019)

[2.1. Latar belakang masalah 6](#_Toc187828020)

[2.2. Rumusan masalah 8](#_Toc187828021)

[2.3. Tujuan 8](#_Toc187828022)

[2.4. Pemilihan Solusi dan Teknik 9](#_Toc187828023)

[2.5. Skenario pemanfaatan produk oleh stakeholder 18](#_Toc187828024)

[3. Usaha pengembangan 21](#_Toc187828025)

[3.1. *Man*-*month* 21](#_Toc187828026)

[3.2. *Machine-month* 21](#_Toc187828027)

[3.3. *Development tools* 23](#_Toc187828028)

[3.4. *Test equipment* 25](#_Toc187828029)

[3.5. Kebutuhan expert 26](#_Toc187828030)

[4. Kesimpulan 27](#_Toc187828031)

# **Pendahuluan**

## **Ringkasan isi dokumen**

Dokumen ini memaparkan informasi mengenai alasan pembuatan, analisis sistem kebutuhan dan usulan proses pengembangan dari sistem. Dokumen ini juga membahas seluruh aspek dari pembuatan sistem, mulai dari sudut pandang pihak pengguna hingga dari sudut pandang pihak penulis. Dokumen ini sendiri dapat membantu menjelaskan sistem yang dibuat kepada pengguna. Dengan demikian, pembaca diharapkan dapat mengerti proses pengembangan dari solusi yang ditawarkan. Kemudian, dokumen ini dapat digunakan sebagai dasar proses pengembangan dan evaluasi selama tahap pengembangan proyek berlangsung. Selain itu, dokumen ini juga dapat digunakan sebagai acuan teknis untuk

## **Aplikasi Dokumen**

Dokumen ini berlaku untuk digunakan dalam pengembangan produk proyek capstone:

1. Sebagai gambaran umum dari segi teknis maupun non-teknis tugas akhir yang akan dikerjakan.
2. Memastikan kelayakan tugas akhir, baik dari segi teknik, waktu, biaya/ekonomis, maupun strategis.
3. Menjadi catatan proses pengerjaan dan revisi yang dilakukan.

Proposal ini diajukan kepada dosen pembimbing tugas akhir dan tim capstone tugas akhir Program Studi Sarjana Teknik Komputer Undip sebagai bahan penilaian tugas akhir.

## **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 |

# **Proposal Pengembangan Produk**

## **Latar belakang masalah**

*Masalah dilihat dari kacamata customer dan dapat menunjukkan di mana terjadinya masalah, merupakan complex engineering problem. Definisi complex engineering problem mengacu pada buku pedoman capstone project.*

**Contoh:**

Keamanan jaringan menjadi semakin krusial mengingat perkembangan pesat teknologi informasi dan internet pada era digital saat ini. Keamanan jaringan menjadi fokus utama dalam menghadapi tantangan berbagai ancaman siber yang terus berkembang, termasuk serangan peretasan, virus, *malware*, dan pencurian data sensitif. Dengan semakin kompleksnya infrastruktur jaringan dan ketergantungan masyarakat terhadap teknologi, organisasi, perusahaan, serta lembaga pemerintahan perlu memastikan integritas, kerahasiaan, dan ketersediaan data yang berada dalam jaringan mereka.

Salah satu jenis serangan pada jaringan adalah DoS/DDoS. *Denial* *of* *Service* (DoS) atau *Distributed* *Denial* of *Service* (DDoS) merupakan serangan yang membanjiri server dengan mengirimkan permintaan yang sangat banyak sehingga menghabiskan sumber daya pada server tersebut sampai server tersebut tidak dapat menjalankan fungsi dan tugasnya dengan benar [1]. Serangan DoS/DDoS mempunyai beberapa bentuk berdasarkan *OSI layer* yang diserang seperti, serangan *slow* HTTP DoS, DRDoS (*Distributed* *Reflection* *Denial* *of* *Service*), dan *DNS* *amplification* DoS/DdoS.

Serangan *slow* HTTP DoS adalah salah satu metode serangan DoS yang menargetkan server HTTP. Metode ini menghambat layanan dengan menjenuhkan kumpulan koneksi dengan permintaan yang lambat dan banyak. Laporan bahwa serangan DoS “*slow* *read*” semacam serangan HTTP DoS dari hanya satu penyerang dapat dicegah secara efektif dengan membatasi jumlah koneksi untuk setiap alamat IP [2]. Sementara itu ada juga serangan DoS lainnya, yaitu *DNS* *amplification*. Serangan amplifikasi *Domain* *Name* *Server* (DNS) adalah bentuk populer dari *Denial* *Distribution* *of* *Service* (DDoS), di mana penyerang menggunakan server DNS terbuka yang dapat diakses publik untuk membanjiri sistem target dengan lalu lintas respons DNS [3]. DRDoS (*Distributed Reflection Denial of Service*) adalah sejenis serangan DoS (*Denial* *of* *Service*). Pada serangan ini pihak ketiga ditipu atau dibohongi untuk mengirimkan data dalam jumlah besar ke para korban. Artinya, penyerang menggunakan spoofing IP alamat sumber untuk menyembunyikan identitas mereka dan menyebabkan pihak ketiga mengirim data ke korban.

Pada kantor Ditreskrimsus Polda Jateng belum ada sistem yang dapat mengenali serangan DoS/DDoS yang terjadi pada server. Oleh karena itu dibutuhkan sistem yang dapat digunakan untuk melakukan monitoring dan memberikan peringatan ketika terjadi suatu serangan DoS/DDoS pada server tersebut. Sebagai bentuk upaya meningkatkan pengamanan pada server kantor Subdit V Ditreskrimsus Polda Jateng dari serangan *Denial of Service/Distributed Denial of Service (DoS/DDoS)*, maka dapat diterapkan teknologi *Intrusion Detection System/Intrusion Prevention System* (IDS/IPS). *Intrusion Detection System* (IDS) adalah sebuah aplikasi perangkat lunak atau perangkat keras yang dapat mendeteksi aktivitas yang mencurigakan dalam sebuah sistem atau jaringan [5]. Dalam melakukan pendeteksian, IDS menggunakan beberapa metode, diantaranya yaitu, *signatures dan anomaly detection*. Teknik IDS saat ini masih banyak menggunakan metode tradisional, sehingga masih jauh dari sempurna dibandingkan dengan teknik dan alat yang digunakan penyerang, karena IDS dengan metode tradisional hanya menggunakan deteksi berbasis signature atau model deteksi berbasis anomali dan menyebabkan banyak kesalahan [6]. Oleh karena itu *Model-Based Feature Selection* diusulkan sebagai cara untuk menutup kesenjangan dan meningkatkan efektivitas IDS. *Smart Intrusion Detection System* (SIDS) ini akan diintegrasikan dengan *log management system* yang akan mempermudah admin dari *stakeholder* untuk memonitoring perangkat dari serangan DoS/DDoS.

## **Rumusan masalah**

*Berisi permasalah stakeholder yang dipilih untuk diselesaikan dengan produk yang diusulkan oleh Tim Capstone. Dapat dinyatakan dalam minimal dua kalimat pernyataan tanpa jargon.*

**Contoh:**

Ditreskrimsus Polda Jateng belum memiliki sistem yang dapat mengenali serangan DoS/DDoS yang terjadi pada server sehingga dibutuhkan sistem yang dapat digunakan untuk memonitor dan memberikan peringatan ketika terjadi serangan DoS/DdoS.

*Intrusion Detection System* (IDS) dengan metode tradisional hanya menggunakan deteksi berbasis *signature* atau model deteksi berbasis anomali dan menyebabkan banyak kesalahan.

## **Tujuan**

*Tujuan umum menjelaskan visi besar atau hasil utama yang ingin dicapai dari proyek diikuti dengan tujuan khusus yang merinci langkah-langkah spesifik atau aspek teknis untuk mendukung pencapaian tujuan umum.*

**Contoh:**

1. Tujuan Umum

"Mengembangkan *Smart Intrusion Detection System* (SIDS) untuk memonitor perangkat dari serangan DoS/DdoS dengan menambahkan *Model-Based Feature Selection* untuk meningkatkan efektivitas IDS tradisional.

1. Tujuan Khusus

*Tujuan khusus ini memuat tujuan proyek capstone sesuai dengan pembagian pekerjaan pada masing-masing anggota tim Capstone.*

**Contoh:**

1. "Membuat dan melakukan pelatihan *machine learning* *Model-Based Feature Selection* untuk mengidentifikasi serangan DoS/DDoS."
2. "Mengintegrasikan *Model-Based Feature Selection* untuk meningkatkan efektivitas IDS pada Smart Intrusion Detection System (SIDS).”
3. " Mengintegrasikan Smart Intrusion Detection System (SIDS) dengan *log management system* dan *distributed streaming system* yang akan mempermudah admin dari *stakeholder* untuk memonitor perangkat dari serangan DoS/DDoS."

## **Pemilihan Solusi dan Analisis**

*Gambarkan beberapa alternatif solusi yang bisa dipilih untuk menyelesaikan masalah yang ada (minimal 3), kemudian lakukan analisis berikut untuk memilih solusi yang ada. Alternatif solusi yang diberikan merupakan tiga contoh alternatif solusi secara umum yang dapat digunakan untuk menyelesaikan permasalahan yang ada. Jika kasus yang digunakan tidak bisa memberikan alternatif solusi secara umum, maka tim capstone bisa memberikan alternatif terhadap fitur yang ada di dalam sistem. Contoh pemilihan alternatif solusi bisa dipisahkan berdasarkan perangkat keras, perangkat lunak, atau hal lain sesuai dengan proyek yang dikembangkan. Analisis aspek terkait yang dilkakukan minimal sebanyak tiga aspek dengan aspek wajib yaitu: ekonomis, manufakturabilitas, dan sustainabilitas.*

**Contoh:**

## **Alternatif Solusi**

* 1. **Alternatif Solusi untuk Deteksi DoS/DdoS**

Berdasarkan usulan solusi yang ditunjukkan oleh tabel 1, *machine learning* dipilih untuk menjadi solusi dalam deteksi DoS/DDoS. Hal ini dikarenakan dalam proses deteksi menggunakan pendekatan berbasis *signature*, IDS tidak dapat mendeteksi serangan yang belum dikenal karena basis data *signature* yang telah kadaluwarsa atau karena *signature* memang belum tersedia. Pendekatan kedua adalah deteksi berbasis anomali dimana pada deteksi ini, perlu diciptakan suatu profil perilaku khusus dari fitur aktivitas jaringan dalam taraf tertentu. Profil ini kemudian dijadikan sebagai dasar untuk mendefinisikan aktivitas jaringan normal [10].

Tabel 1. Alternatif solusi untuk deteksi DoS/DdoS

|  |  |
| --- | --- |
| **Solusi** | **Deskripsi** |
| *Signature Based Detection* | Signature Based Detection adalah salah satu metode IDS yaitu, dengan cara menyadap paket data kemudian membandingkannya dengan database rute IDS yang berisi signature-signature paket serangan. Berdasarkan pembandingan tersebut, jika paket data mempunyai pola yang sama dengan setidaknya salah satu pola di dalam database rule IDS, maka paket tersebut dapat dianggap sebagai serangan, dan demikian juga sebaliknya. Apabila paket data tersebut sama sekali tidak mempunyai pola yang sama dengan pola di database rule IDS. maka paket data tersebut dianggap bukan serangan [7]. |
| *Anomaly Based Detection* | IDS jenis ini dapat mendeteksi adanya penyusupan dengan mengamati adanya kejanggalan pada sistem, atau adanya penyimpangan-penyimpangan dari kondisi normal. Sebagai contoh, apabila terdapat penggunaan memori yang melonjak secara terus menerus atau terdapat koneksi paralel dari sebuah IP Address dalam jumlah yang banyak dalam waktu yang bersamaan, maka kondisi tersebut dapat dianggap sebagai sebuah kejanggalan, yang kemudian oleh IDS dianggap sebagai serangan [7]. |
| *Machine Learning* | Machine learning adalah bidang ilmu komputer yang menggunakan teknik statistika untuk memberi kemampuan sistem komputer agar dapat belajar dari data tanpa diprogram secara eksplisit [8]. Untuk IDS, machine learning dapat digunakan untuk mendeteksi serangan yang diketahui atau serangan yang tidak diketahui jika model telah cukup terlatih [9]. |

* 1. **Alternatif solusi untuk *Intrusion Detection System/Intrusion Prevention System* (IDS/IPS)**

Berdasarkan pada tiga usulan solusi yang ditunjukkan di tabel 2, Snort dipilih untuk menjadi solusi dan teknik untuk *Intrusion Detection System/Intrusion Prevention System* (IDS/IPS). Hal ini dikarenakan Snort lebih mudah untuk diimplementasikan dan memiliki *ruleset* yang dapat dikustomisasi sesuai kebutuhan [13].

Tabel 2. Alternatif solusi untuk *Intrusion Detection System/Intrusion Prevention System* (IDS/IPS)

|  |  |
| --- | --- |
| **Solusi** | **Deskripsi** |
| *Suricata* | Suricata adalah IDS, IPS dan monitoring engine untuk jaringan yang berkinerja tinggi. Suricata adalah open source dan dimiliki oleh masyarakat yang dikelola yayasan non-profit, Open Information Security Foundation (OISF). Suricata dikembangkan oleh OISF dan vendor pendukungnya [11]. |
| *Snort* | Snort merupakan salah satu contoh program Network-based Intrusion Detection System, yaitu sebuah program yang dapat mendeteksi suatu usaha penyusupan pada suatu sistem jaringan komputer. Snort bersifat open source dengan lisensi GNU General Purpose License sehingga software ini dapat dipergunakan untuk mengamankan sistem server tanpa harus membayar biaya lisensi [7]. |
| *Security Onion* | Security Onion merupakan salah satu distro dari sistem operasi berbasis Linux. Security Onion umumnya digunakan sebagai tools atau bundle packet untuk Network Security Monitoring (NSM). Security Onion dapat difungsikan menjadi dua jenis sistem operasi, pertama sebagai sistem operasi standalone dimana Security Onion akan berfungsi sebagai penyaji data sedangkan jenis kedua sebagai server untuk merekam, mengelola dan menyajikan data yang didapat dari sistem sensornya [12]. |

* 1. **Alternatif solusi untuk *Log Management System***

Dari tiga pilihan solusi yang ditunjukkan dalam tabel 3, ELK Stack dipilih karena *platform Log Management System* lainnya mematok harga yang cukup tinggi untuk penyimpanannya. ELK Stack merupakan *Log Management System* berbasis open source dan gratis serta memiliki kemampuan untuk menjadi *Log Management System* yang baik.

Tabel 3. Alternatif solusi untuk *Log Management System*

|  |  |
| --- | --- |
| **Solusi** | **Deskripsi** |
| ELK Stack | ELK Stack terdiri dari 4 buah aplikasi yang berbeda yaitu Elasticsearch, Logstash, Kibana, dan Beats. Masing-masing aplikasi memiliki fungsi yang berbeda-beda untuk menunjang dalam manajemen log [14]. Elasticsearch Logstash Kibana (ELK Stack) merupakan komponen yang tepat dalam membangun *log event management* yang dapat memberi insight kepada sistem administrator mengenai tren, statistik, dan anomali yang terjadi [15]. Sebuah *platform* perangkat lunak yang digunakan untuk memantau, mencari, menganalisis dan memvisualisasikan data yang dihasilkan mesin. |
| Splunk | Splunk melakukan penangkapan, pengindeksan, dan menghubungkan data secara realtime ke dalam sebuah wadah yang dapat dengan mudah dicari dan menghasilkan suatu grafik, peringatan, *dashboard* serta visualisasi agar data dapat dengan mudah dibaca dan dianalisis [16]. Menghadirkan pengumpulan dan normalisasi log terpusat, deteksi ancaman dan respons otomatis, visualisasi yang intuitif, dan antarmuka pengguna, serta korelasi waktu nyata dan pencarian *log* untuk mendukung penyelidikan [17]. |
| *SolarWinds Security Event Manager* | Menghadirkan pengumpulan dan normalisasi *log* terpusat, deteksi ancaman dan respon otomatis, visualisasi yang intuitif, dan antarmuka pengguna, serta korelasi waktu nyata dan pencarian *log* untuk mendukung penyelidikan [17]. |

* 1. **Alternatif solusi untuk algoritma klasifikasi**

Dari usulan solusi untuk algoritma machine learning pada tabel 4, Algoritma Naive Bayes dan KNN dipilih. Pada penelitian ini digunakan dua dataset yaitu, CICDoS2017 dan CICDDoS2019. Algoritma Decision Tree memberikan akurasi 60,23% untuk UNSW-NB-15 dan 99,08% untuk dataset CICDDoS2019 [22]. Sementara itu, pada dataset CICDoS2017 Algoritma KNN memberikan akurasi 99,36% [23].

Tabel 4. Alternatif solusi untuk algoritma klasifikasi

|  |  |
| --- | --- |
| **Solusi** | **Deskripsi** |
| *Decision Tree* | *Decision Tree* adalah algoritma yang bisa menghasilkan keputusan dengan cara membentuk pohon keputusan [18]. Metode pohon keputusan mengubah fakta yang sangat besar menjadi pohon keputusan yang merepresentasikan aturan. Pohon keputusan juga berguna untuk mengekplorasi data, menemukan hubungan tersembunyi antara sejumlah calon variabel input dengan sebuah variabel target [19]. |
| *Naive Bayes* | Algoritma *Naive Bayes* merupakan algoritma yang menggunakan perhitungan probabilitas. Algoritma ini umumnya digunakan untuk menyelesaikan permasalahan prediksi berupa klasifikasi. Algoritma ini juga dikenal sebagai yang memiliki akurasi yang tinggi. Proses klasifikasi pada algoritma ini terdapat fase yakni fase *training* dan fase *testing*. Fase *training* atau bisa disebut sebagai fase *learning* adalah sebagian data telah diketahui kelas datanya untuk model perkiraan. Selanjutnya fase pengujian atau bisa disebut fase klasifikasi model yang sudah terbentuk diuji dengan sebagian data lainnya agar diketahui akurasi atas model yang sudah terbentuk [20]. |
| *KNN* | Algoritma KNN bersifat sederhana bekerja berdasarkan jarak terpendek dari data uji ke data latih untuk menentukakan kelas dari data tersebut Setelah mengumpulkan data-data pada kelompok k tertentu, kemudian diambil kelas data mayoritas untuk dijadikan sebagai kelas prediksi dari data uji. KNN memiliki beberapa kelebihan yaitu tahan terhadap data yang memiliki *noise* dan efektif terhadap data latih yang berjumlah besar dan memiliki performa cukup baik. Namun waktu yang digunakan untuk komputasi sangatlah lama jika data latihnya besar dan sangat sensitif dengan ciri yang redundan atau relevan [21]. |

## **Analisis Aspek Terkait**

1. **Analisis dari aspek ekonomis**

*Analisis aspek ekonomis menggunakan definisi sebagai berikut: (Hoffman, H.F. and Hoffman, H.F., 2014. Engineering and the capstone course (pp. 1-5). Springer International Publishing.)*

*Economic or financial: effect of this topic on the local economy, savings, possible cost of project development, material cost, labor issues, outsourcing needs, etc. Harus terdapat cost-benefit analysis yang mencakup hal-hal berikut.*

*Cost:*

* *Biaya operasional existing (sebelum adanya proposal solusi yang dikembangkan, jika ada).*
* *Biaya project development, yang juga mencakup setiap material cost.*
* *Biaya kebutuhan tenaga ahli dan operator (labor) dengan adanya solusi tersebut, sekaligus bagaimana efeknya terhadap ekonomi setempat, jika ada.*

*Benefit:*

* *Saving (berapa ekspektasi pengurangan biaya dengan adanya solusi).*
* *Aspek benefit yang intangible dengan adanya solusi yang ditawarkan. Contoh: solusi dapat memotong ekspektasi waktu pekerjaan tiga hari menjadi tiga jam.*

*Financial Metrics* (pilih salah satu):

* *Return of Investment (ROI)*
* *Payback period*
* *Net Present Value (NPV)*
* *Internal Rate of Return (IRR)*

**Contoh:**

***Cost*:**

|  |  |  |  |
| --- | --- | --- | --- |
| Kategori | Item | Biaya (Rp) | Keterangan |
| Biaya Pengembangan Sistem | Tenaga Kerja | 0 | Tim Capstone |
| Lisensi Software:  Suricata atau Snort,  ElkStack, Elasticsearch,  Logstast, dan Kibana),  Apache Kafka | 0 | Menggunakan perangkat *open source* |
| Infrastruktur:  Server internal Polda | 0 | Menggunakan server yang sudah ada |
| Dataset | 0 | Pemanfaatan dataset publik seperti CICDoS2017 dan CICDDoS2019. |
| Biaya Operasional Tahunan | Pemeliharaan server internal | 0 | Biaya listrik, pendinginan, dan perawatan server Polda Jateng sudah masuk ke anggaran tahunan Polda. |
| Sertifikat SSL/TLS | 0 | Keamanan enkripsi komunikasi data, sudah masuk ke anggaran tahunan Polda |

***Benefit*:**

* 1. ***Tangible Benefit***

Mengurangi biaya downtime akibat serangan DDoS:

* Asumsi kerugian akibat gangguan sistem Rp20.000.000/jam.
* Jika sistem mencegah downtime 5 jam/bulan:
  + Penghematan: Rp20.000.000 x 5 x 12 bulan = **Rp1.200.000.000/tahun**.

|  |  |  |  |
| --- | --- | --- | --- |
| Kategori | Item | Biaya (Rp) | Keterangan |
| Penghematan dan Efisiensi | Pencegahan downtime (5 jam/bulan) | 1.200.000.000 | Mengurangi kerugian akibat gangguan sistem layanan daring. |

* 1. ***Intangible Benefit***

Aspek ekonomis tidak harus selalu diukur dari sesuatu yang *tangible*, tetapi bisa juga dari aspek *intangible*. Misal: kecepatan proses pencarian data, kemampuan *real-time tracking*,

1. **Analisis dari aspek manufakturabilitas**

*Analisis dari aspek manufakturabilitas ini terkait dengan ketersediaan, skalabilitas dan maintanibilitas seperti: material availability, software availability, technology stack, use of commercial-off-the-shelf (COTS) versus custom components, special needs for hostile environments.*

**Contoh:**

1. *Material Availability:*

|  |  |
| --- | --- |
| Sensor | Sensor kualitas air (pH, suhu, oksigen terlarut, salinitas). |
| Ketersediaan | Tersedia secara luas di pasar dengan berbagai spesifikasi. |

1. Penggunaan *Commercial-Off-The-Shelf* dan *Custom Components*

|  |  |
| --- | --- |
| COTS Components | Sensor, microkontroler, modul komunikasi |
| Custom Components | jalur sensor, tempat sensor, dkk |

1. *Special Needs for Hostile Environments*
2. Lingkungan basah dan korosif:

|  |  |
| --- | --- |
| Masalah | Air tambak sering kali memiliki kadar garam yang tinggi, yang dapat mempercepat korosi pada komponen logam. |
| Solusi | Gunakan bahan tahan korosi seperti stainless steel untuk rangka dan sensor dan melapisi PCB dengan bahan konformal untuk melindungi dari kelembapan dan salinitas. |
| Gunakan casing pelindung IP68 yang tahan air dan debu digunakan untuk perangkat elektronik utama. |

1. Paparan sinar matahari dan suhu ekstrem:

|  |  |
| --- | --- |
| Masalah | Perubahan suhu yang drastis dapat memengaruhi performa baterai dan sensor. |
| Solusi | Gunakan baterai lithium-ion yang dirancang untuk suhu tinggi dan menambahkan pelindung UV pada casing untuk mengurangi degradasi material. |

1. *Software availability:*

|  |  |  |
| --- | --- | --- |
| Jenis Perangkat Lunak | Nama Perangkat Lunak | Keterangan |
| *Intrusion Detection System/Intrusion Prevention System (IDS/IPS)* | Suricata | *Open source* |
| Snort | *Open source* |
| *Log Management System* | Elk Stack (Elasticsearch, Logstast, dan Kibana) | *Open source* |
| *Distributed Streaming System* | Apache Kafka | *Open source* |

Perangkat lunak yang digunakan yaitu Suricata atau Snort, Elk Stack (Elasticsearch, Logstast, dan Kibana) dan Apache Kafka merupakan perangkat lunak *open* *source* dan diperbaharui secara berkala. Forum penggunaan pun aktif, sehingga pencarian informasi mengenai perangkat lunak yang digunakan lebih mudah dilakukan. Selain itu aplikasi ini juga berjalan pada sistem operasi Linux, Linux sendiri merupakan sistem operasi yang bersifat *open source* yang cukup populer.

1. **Analisis dari aspek sustainibilitas**

*Dalam membahasa aspek sustainibilitas, Tim Capstone perlu memaparkan dampak analisis produk yang dipilih dalam hal: meminimalisasi pengaruh negatif terhadap lingkungan, memastikan kelangsungan ekonomi jangka panjang, penggunaan energi, dll.*

**Contoh:**

1. Efisiensi Energi: bagaimana meminimalisasi penggunaan energi akan produk yang dikembangkan.
2. Manajemen Sumber: bagiaman merancang suatu aplikasi yang meminimalkan penggunaan ruang penyimpanan, bagaimana meminimalkan proses data transfer.
3. Efisiensi desain: bagaimana merancang sistem yang *scalable.*
4. Dll.
5. **Analisis aspek tambahan**

*Bila diperlukan dalam perancangan produk yang dibuat dapat memilih salah satu proses analisis aspek yang ada (ethical, legal, environment, political, societal), yang berhubungan dengan produk yang akan dirancang*.

**Contoh:**

* + - * 1. Aspek Legal: Aspek legal terkait dengan hukum yang mungkin muncul pada produk yang akan dibuat, terkait dengan privasi data, hukum-hukum yang mengatur pengambilan dan penyimpanan data. Contoh: Bila pada program yang dibuat berisikan tentang data-data kesehatan, maka perlu melakukan analisis program berdasarkan Undang-Undang Nomor 36 Tahun 2009 tentang kerahasiaan data kesehatan, UU Nomor 19 Tahun 2016 tentang informasi dan transaksi elektronis, dll.
        2. Aspek Etika: Terkait isu etika yang muncul pada produk yang akan dibuat. Contoh: Isu etika yang dibahas dapat terkait pada tanggung-jawab moral terkait transparasi data, atau pada pembagian hak akses data pada masing-masing *role* dan kerahasiaannya.

## **Pemilihan Solusi**

Berdasarkan usulan solusi yang ditunjukkan oleh tabel 1, *machine learning* dipilih untuk menjadi solusi dalam deteksi DoS/DDoS. Hal ini dikarenakan dalam proses deteksi menggunakan pendekatan berbasis *signature*, IDS tidak dapat mendeteksi serangan yang belum dikenal karena basis data *signature* yang telah kadaluwarsa atau karena *signature* memang belum tersedia. Pendekatan kedua adalah deteksi berbasis anomali dimana pada deteksi ini, perlu diciptakan suatu profil perilaku khusus dari fitur aktivitas jaringan dalam taraf tertentu. Profil ini kemudian dijadikan sebagai dasar untuk mendefinisikan aktivitas jaringan normal [10].

Berdasarkan pada tiga usulan solusi yang ditunjukkan di tabel 2, Snort dipilih untuk menjadi solusi dan teknik untuk *Intrusion Detection System/Intrusion Prevention System* (IDS/IPS). Hal ini dikarenakan Snort lebih mudah untuk diimplementasikan dan memiliki *ruleset* yang dapat dikustomisasi sesuai kebutuhan [13].

Dari tiga pilihan solusi yang ditunjukkan dalam tabel 3, ELK Stack dipilih karena *platform Log Management System* lainnya mematok harga yang cukup tinggi untuk penyimpanannya. ELK Stack merupakan *Log Management System* berbasis open source dan gratis serta memiliki kemampuan untuk menjadi *Log Management System* yang baik.

Dari usulan solusi untuk algoritma machine learning pada tabel 4, Algoritma Naive Bayes dan KNN dipilih. Pada penelitian ini digunakan dua dataset yaitu, CICDoS2017 dan CICDDoS2019. Algoritma Decision Tree memberikan akurasi 60,23% untuk UNSW-NB-15 dan 99,08% untuk dataset CICDDoS2019 [22]. Sementara itu, pada dataset CICDoS2017 Algoritma KNN memberikan akurasi 99,36% [23].

## **Skenario pemanfaatan produk oleh stakeholder**

*Mendeskripsikan bagaimana produk digunakan (dijelaskan sistem/ sub-sistem serta bagaimana masing-masing stakeholder memanfaatkan produk. Dalam bagian ini, tim capstone harus membuat: 1. Gambar rancangan arsitektur high-level dari produk (bagaimana setiap sistem/subsistem terhubung); 2. Penjelasan bagaimana produk yang dikembangkan digunakan per-stakeholder (misal dihubungkan antara fitur dengan stakeholder).*

**Contoh**:

Dalam penelitian ini, dilakukan penelitian untuk mengembangkan sebuah Sistem Monitoring dan Deteksi Serangan DoS/DDoS Menggunakan *Smart Intrusion Detection System* (SIDS) dengan menerapkan pendekatan *Model-Based Feature Selection* ke sistem yang sedang dibangun. Sistem monitoring ini memiliki fungsi utama, yaitu untuk mendeteksi beberapa jenis serangan DoS/DDoS seperti *slow* HTTP DoS, DNS amplification DoS/DDoS, dan DRDoS (*Distributed Reflection Denial of Service*).

Gambar 2.1 menunjukkan arsitektur sistem monitoring ini menggunakan beberapa perangkat lunak seperti IDS Snort, Kafka, dan Elasticsearch Logstash Kibana (ELK Stack). Setiap perangkat lunak memiliki fungsi masing-masing dalam pembangunan sistem ini dan akan dijelaskan pada bagian *development tools*. Berikut merupakan cara kerja Sistem Monitoring dan Deteksi Serangan DoS/DDoS Menggunakan Smart Intrusion Detection System (SIDS) :

1. IDS Snort melakukan Capture/Sniff trafik jaringan pada host/server kemudian menyimpannya dalam bentuk file pcap.
2. File pcap diubah ke bentuk file csv.
3. Setelah itu, dilakukan ekstraksi fitur terhadap file pcap yang sudah diubah ke bentuk file .csv berdasarkan format dari dataset yang digunakan.
4. Melakukan prediksi menggunakan model machine learning yang sudah dikembangkan.
5. Mengirimkan hasil prediksi ke Apache Kafka menggunakan Filebeat.
6. Apache Kafka mendistribusikan log yang dikirim dari Filebeat ke Log Management System.
7. Log Management System (ELK Stack) melakukan klasifikasi terhadap log yang diterima menjadi log trafik berjenis DoS/DDoS atau trafik normal.
8. ElastAlert dapat mengirimkan notifikasi ke aplikasi pesan seperti Whatsapp apabila dibutuhkan.

A diagram of a log management system

Description automatically generated

Gambar 2.1. Arsitektur Sistem Monitoring dan Deteksi Serangan DoS/DdoS

Sistem monitoring DoS/DDoS ini akan dioperasikan di kantor Ditreskrimsus Polda Jateng Subdit V Siber. Pada kantor tersebut diperlukan satu orang administrator yang dapat menjalankan fungsi sebagai berikut :

1. Melakukan Monitoring Log

Sistem monitoring ini dapat menampilan log dari berbagai sistem seperti sistem operasi, router, switch, server, dan lain-lain. Namun untuk proyek ini dikhususkan untuk dapat menampilkan log yang berasal dari IDS/IPS.

1. Melakukan Monitoring Dashboard

Sistem monitoring ini juga terdapat fitur dashboard yang dapat dimanfaatkan oleh administrator agar dapat melakukan monitoring sistem dengan lebih mudah dibandingkan hanya melakukan monitoring log. Hal ini dikarenakan data pada dashboard sudah divisualisasikan.

1. Menerima Alert

Alert atau peringatan pada sistem monitoring ini dikhususkan hanya untuk peringatan DoS/DDoS saja. Namun dapat diperluas lagi untuk memberikan peringatan lain sesuai kebutuhan. Alert atau peringatan ini akan diintegrasikan dengan E-mail dan Whatsapp sehingga dapat memudahkan Administrator jika sedang berada di luar kantor.

# **Usaha pengembangan**

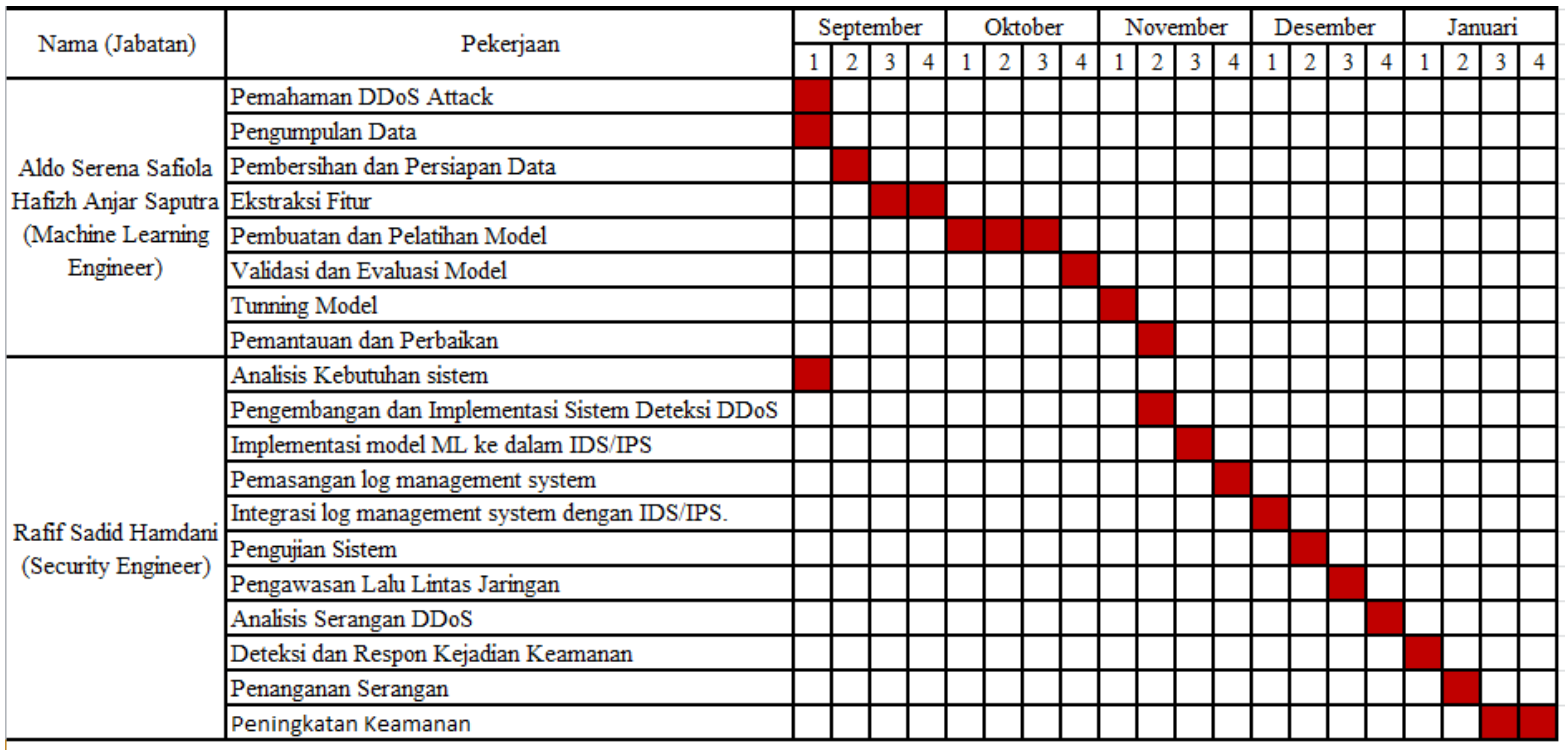
## ***Man*-*month***

*Detail dari analisis ekonomis yang sudah disebutkan dalam hal perhitungan biaya pekerja.*

**Contoh:**

Proyek pembuatan Sistem Monitoring dan Deteksi Serangan DoS/DDoS Menggunakan Model-Based Feature Selection ini, akan dikerjakan oleh satu tim Tugas Akhir Capstone program studi S1-Teknik Komputer periode 2023/2024 yang terdiri atas tiga orang mahasiswa Teknik Komputer angkatan 2020. Dua mahasiswa/pengembang akan bekerja menjadi *machine learning* *engineer* dan satu lainnya akan menjadi *security engineer* dengan tugas yang berbeda-beda. Berikut Gantt Chart pembagian waktu sesuai tugas masing-masing yang ditunjukkan pada Tabel 8 berikut.

Tabel 8. Gantt chart man month



## ***Machine-month***

Detail dari analisis ekonomis yang sudah disebutkan dalam hal perhitungan biaya mesin yang digunakan untuk mengembangkan produk. Sistem ini dibuat oleh tiga orang pengembang dengan setiap pengembang yang menggunakan laptop masing–masing. Dari pernyataan tersebut, berarti akan terdapat tiga buah perangkat yang digunakan untuk membuat sistem monitoring DoS/DDoS di mana setiap perangkat ini akan aktif atau digunakan sesuai dengan jam kerja penggunaannya.

Tabel 9. Tabel lama penggunaan mesin dan waktu pengerjaan

A table with a number of laptops

Description automatically generated with medium confidence

Tabel 9 menunjukkan total dari waktu dalam jam yang akan digunakan selama pembuatan Sistem Monitoring dan Deteksi Serangan DoS/DDoS menggunakan *Model-Based Feature Selection*.

*Machine-Month* merupakan metrik yang diperlukan untuk menentukan waktu/usaha yang

dibutuhkan untuk menyelesaikan satu proyek tugas berdasarkan waktu maksimal penggunaan

suatu mesin. Formula untuk menghitung Machine-Month didefinisikan sebagai berikut:

Machine − Month =

Sementara itu, untuk mendapatkan nilai “Hari kerja jika 24 jam bekerja” dapat menggunakan rumus:

Hari kerja jika 24 jam bekerja =

Kemudian, untuk nilai “Hari kerja dalam 1 bulan” diasumsikan sama dengan 20 hari, yang berarti 5 hari bekerja dalam seminggu selama 4 minggu. Dengan asumsi hari kerja dalam 1 bulan adalah 20 hari, maka dapat disimpulkan bahwa waktu kerja dalam 9 bulan ialah 9 x 20 hari = 180 hari. Perhitungan *Machine-Month* dari pembuatan sistem monitoring ini dapat dilihat pada Tabel 10.

Tabel 10. Tabel perhitungan Machine-Month

A table with numbers and text

Description automatically generated

## ***Development tools***

*Detail dari perangkat keras dan perangkat lunak yang digunakan dalam proses pengembangan.*

## **Perangkat Keras**

1. **PC/Komputer**

Pada proyek ini, komputer digunakan sebagai antarmuka untuk mengimplementasikan sistem monitoring dan deteksi DoS/DDoS. Hal ini dikarenakan sistem operasi yang digunakan pada server adalah ClearOS.

1. **Laptop**

Pada sistem monitoring dan deteksi DoS/DDoS ini, laptop digunakan sebagai tempat pemasangan VirtualBox. Setelah itu dilakukan pemasangan sistem operasi kali linux dan ubuntu pada VirtualBox tersebut.

1. **Server**

Server adalah sebuah sistem komputer yang terdapat pada jaringan komputer untuk menyediakan suatu layanan kepada pengguna yang disebut sebagai client [24]. Server digunakan sebagai tempat pemasangan perangkat lunak yang berkaitan dengan sistem monitoring DoS/DDoS ini. Selain itu semua log serangan DoS/DDoS juga akan disimpan di server.

## **Perangkat Lunak**

1. **VirtualBox**

VirtualBox adalah perangkat lunak virtualisasi, yang dapat digunakan untuk mengeksekusi sistem operasi tambahan di dalam sistem operasi utama. VirtualBox berfungsi untuk melakukan virtualisasi sistem operasi. VirtualBox juga dapat digunakan untuk membuat virtualisasi jaringan komputer [25]. Pada sistem yang dibuat, VirtualBox digunakan untuk untuk menjalankan sistem operasi Kali Linux dan Ubuntu.

1. **ClearOS**

ClearOS adalah Linux yang di kostumasi khusus untuk keperluan server. Dengan berbagai fitur yang *powerfull* dan *setting* yang sederhana, ClearOS menjadi alternatif pilihan, baik untuk pemula yang tidak mengerti Linux sama sekali maupun untuk profesional yang memerlukan kemampuan terbaik dari OS Linux *server* [23].

1. **Ubuntu**

Ubuntu adalah sistem operasi turunan dari distro Linux jenis Debian *unstable* (sid), Ubuntu merupakan *project* untuk komunitas, yang bertujuan untuk menciptakan sebuah sistem operasi beserta dengan paket aplikasinya yang bersifat *free* dan *open* *source*, karena Ubuntu mempunyai prinsip untuk selamanya gratis dan tidak ada tambahan untuk versi *enterprise* *edition* [24]. Ubuntu merupakan salah satu sistem operasi yang digunakan dalam sistem ini. Ubuntu akan berperan sebagai pusat untuk melakukan pemasangan *tools* SIDS dan *log management system.*

1. **Kali Linux**

Kali Linux merupakan sistem operasi *open source* yang dapat digunakan untuk *penetration* *testing* terhadap suatu sistem dan jaringan komputer. Terdapat lebih dari 300 *tools* dengan fungsi masing-masing yang dapat digunakan untuk melakukan pengujian keamanan terhadap suatu sistem jaringan. Kali Linux dikembangkan dan didanai oleh Offensive Security [25]. Kali Linux merupakan salah satu sistem operasi yang digunakan dalam sistem, Sistem operasi Kali Linux akan digunakan sebagai *client* atau *attacker* yang akan melancarkan serangan DoS/DDoS.

1. **Snort**

Snort adalah sistem pendeteksi dan pencegahan intrusi jaringan sumber terbuka. Ini dapat menganalisis analisis lalu lintas dan aliran data secara real-time dalam jaringan. Snort mampu memeriksa analisis protokol dan dapat mendeteksi berbagai jenis serangan. Dalam NIDS, Snort pada dasarnya memeriksa paket melawan aturan yang ditulis oleh pengguna. Aturan Snort dapat

ditulis dalam bahasa apa pun, strukturnya juga baik dan mudah dibaca, serta aturan dapat dimodifikasi juga [26].

1. **Apache Kafka**

Apache Kafka adalah sebuah *platform open-source* yang digunakan untuk mengelola aliran data secara *real-time*. Apache Kafka pada awalnya dikembangkan di LinkedIn dan sekarang menjadi bagian dari yayasan Apache Software Foundation. Apache Kafka dirancang untuk menangani volume data yang besar dan memungkinkan pertukaran data yang efisien antara *producer* dan *consumer* [27].

1. **ELK Stack**

Elasticsearch Logstash Kibana (ELK Stack) merupakan komponen yang tepat dalam membangun log event management. ELK Stack dapat memberi informasi kepada sistem administrator mengenai tren, statistik, dan anomali yang terjadi. ELK Stack merupakan kumpulan dari tiga alat yaitu Elasticsearch, Logstash, dan Kibana [28].

## ***Test equipment***

*Gambaran pengujian yang akan dilakukan untuk menguji produk yang diusulkan beserta instrumen yang digunakan.*

**Contoh:**

Pengujian pada sistem ini akan dilakukan dengan menggunakan dataset dan pengujian secara langsung. Pengujian secara langsung akan digunakan dataset dari CICDoS2017 dan CICDDoS2019. Sementara itu, Untuk keperluan pengujian secara langsung digunakan beberapa perangkat lunak DoS dan DDoS yang dipasang pada sistem operasi kali linux. Perangkat lunak tersebut akan digunakan untuk menguji sistem yang sudah terpasang dengan *Smart Intrusion Detection System* (IDS). Perangkat lunak ini dibagi sesuai dengan jenis DoS/DDoS yang akan diuji. Berikut beberapa perangkat lunak DoS/DDoS yang akan digunakan untuk melakukan pengujian yang bisa dilihat pada Tabel 11.

Tabel 11. Perangkat lunak untuk pengujian DoS/DDoS

A table with text and images

Description automatically generated with medium confidence

## **Kebutuhan pakar**

*Jika membutuhkan saran ahli dalam mengembangkan sistem, maka tambahkan di sini judgment dari ahli tersebut.*

**Contoh:**

Pengembangan proyek Sistem Monitoring dan Deteksi Serangan DoS/DdoS menggunakan *Smart Intrusion Detection System* (SIDS) berbasiskan algoritma *machine learning* yang akan diimplementasikan di Ditreskrimsus Polda Jateng Semarang ini tidak memerlukan kebutuhan pakar dikarenakan pada proyek ini hanya diperlukan koordinasi antara stakeholder dan pengembang mengenai proyek agar sesuai dengan tujuan dan keinginan yang ditetapkan.

# **Kesimpulan**

Kesimpulan yang dapat diambil dari dokumen ini adalah:

1. Snort digunakan sebagai alat Intrusion Detection System (IDS) terhadap serangan DoS/DDoS.
2. Pendekatan yang digunakan adalah menggunakan algoritma machine learning.
3. Jenis serangan DoS/DDoS yang akan dideteksi yaitu, slow HTTP DoS, DNS amplification DoS/DDoS, dan DRDoS (Distributed Reflection Denial of Service). Tiga jenis serangan ini akan diuji menggunakan dataset CICDoS2017 dan CICDDoS2019. Selain itu dilakukan juga uji secara langsung menggunakan tools yang dipasang pada sistem operasi Kali Linux.
4. Perbedaan serangan DoS dan DDoS terletak pada subjek yang melakukan penyerangan. Serangan DoS biasanya dilakukan dari satu sumber atau satu lokasi. Sementara itu, serangan DDoS biasanya dilakukan oleh bot yang telah dikonfigurasi oleh penyerang.
5. Sistem operasi yang digunakan adalah berbasis Linux.