Perancangan dan Analisis Crowdsec Sebagai *Intrusion Prevention System* pada Infrastruktur Server

1st Muhammad Thariq Sunu Rendratama
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
sunumd@student.telkomuniversity.ac.id

2nd Parman Sukarno
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
parmansukarno@telkomuniversity.ac.id

3rd Aulia Arif Wardana
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
auliawardan@telkomuniversity.ac.id

Abstrak-Belakangan ini perkembangan teknologi semakin membesar, industri - industri *Technology Information* (IT) banyak menerapkan dan membangun teknologi - teknologi baru tersebut ke dalam bisnis perusahaannya, salah satunya bisnis *Financial Technology* atau biasa disebut Fintech. Fintech ini merupakan sebuah industri perusahaan yang berbasis finansial atau keuangan, dimana mereka ingin memberikan layanan dalam mengelola uang kepada masyarakat agar lebih efisien dan juga mudah. Banyak dari Fintech ini memberikan layanan keuangan misalkan *payment*, *assest management*, dan juga *financing* [2] secara daring tersebut. Dari banyak hal Fintech tersebut maka akan semakin banyak juga kejahatan siber di internet yang mengincar perusahan tersebut, untuk itu orang - orang yang berada di dalam sektor tersebut perlu meningkatkan sistem keamanan yang sangat ekstra guna menangkal serangan - serangan yang ada. Dan salah satu cara yang paling ampuh digunakan adalah memblokir akses pengguna ke dalam sistem bagi pengguna yang dinilai bersifat membayakan, dan *tool* yang akan digunakan pada penelitian ini yaitu Crowdsec. Crowdsec merupakan sebuah *Intrusion Detection and Prevention System* (IDPS) untuk mendeteksi dan juga mencegah sistem dari ancaman siber yang ada, teknisnya adalah dia akan mem-*ban* atau memblokir akses pengguna melalui IP *address* yang dibacanya sebagai akses mencurigakan atau gagal.

Kata Kunci- fintech, blockchain, cryptocurrentcy crowdsec, pengguna, sistem.

Abstract-Lately, technological developments are getting bigger, the Information Technology (IT) industry is applying and building these new technologies into their company's business, one of which is the Financial Technology business or commonly called Fintech. Fintech is a financial or financial-based company industry, where they want to provide services in managing money to the public to be more efficient and easy. Many of these Fintechs provide financial services, such as online payments, asset management, and financing [2]. From these many Fintech things, there will be more and more cyber crimes on the internet that are targeting these companies, for that people in the sector need to improve a very extra security system to ward off existing attacks. And one of the most effective ways to use it is to block user access to the system for users who are considered dangerous, and the tool that will be used in this research is Crowdsec. Crowdsec is an Intrusion Detection and Prevention System (IDPS) to detect and prevent the system from existing cyber threats, technically it will ban or block user access via IP addresses which it reads as suspicious or failed access.

Keywords: fintech, blockchain, cryptocurrency, crowdsec, users, systems

I. PENDAHULUAN

A. Latar Belakang

Disamping besar fungsionalitas Fintech dalam industri, sebenarnya ada banyak sekali bahaya kejahatan siber yang mengancamnya. Berdasarkan hasil penelitian di Kenya pada tahun 2020 - 2021 ada beberapa serangan yang paling populer di dunia *Information Technology* (IT) yaitu *Malware*, DDoS, *Web Application Attack*, *System Vulnerability* [3]. Dari beberapa daftar serangan tersebut merupakan serangan yang paling banyak diperoleh saat ini, dan dampak dari serangan tersebut sangatlah begitu besar sehingga jika perusahaan terkena dampak serangan tersebut akan dapat mendapatkerugian yang sangat besar. Dilansir dari Trustwave's 2015 Global Security Report, ada sekitar 98% *vulnerability* dari pengujian *web application*, serta rata - rata ada di basis *Department of Business*. Kemudian dari Innovation and Skills'2015 Security Survey ada skitart 90% di organisasi atau perusahaan besar, 74% dari organisasi atau perusahaan kecil [4].

Maka dari itu peran keamanan siber sangat diperlukan untuk melindungi aset perusahaan yang di miliki dari serangan siber yang ada, dan sebagai salah satu bentuk implementasi, pada dari Tugas Akhir ini saya mendapatkan riset peneilitian untuk Tugas Akhir pada perusahaan Fintech dengan basisnya di Blockchain. Faktor riset yang saya lakukan disini adalah untuk mengimplementasikan sistem keamanan baru pada perusahaan yaitu Crowdsec sebagai *Intrusion Detection/Prevention System* (IDS/IPS).

B. Informasi Data Serangan

Berdasarkan perolehan data internal perusahaan mengenai serangan yang pernah terjadi pada perusahaan, disini saya memperoleh informasi jenis serangan yang paling dominan sebanyak dua kali dalam 3 (tiga) tahun terakhir. Dari perolehan dominan yang dimaksud ini adalah meruapakn serangan yang paling sering terjadi dan memiliki dampak serangan yang cukup merugikan bagi perusahaan yang saya tempati saat ini. Dan unuk daftar serta jenis serangannya adalah sebagai berikut:

TABLE 1.
DAFTAR PEROLEHAN SERANGAN

No.	Serangan	Target
1.	Distributed Denial of Service (DDoS)	Network
2.	Login Enumeration	API Backend

Kemudian dari daftar serangan diatas dapat di deskipsikan untuk pemetaan resikonya sebagai berikut:

TABLE 2.
DAFTAR PEMETAAN RESIKO

No.	Resiko	Deskripsi
1.	Distributed Denial of Service (DDoS)	*Denial of Service* (DoS) merupakan jenis serangan yang sangat merugikan bagi layanan *server*, karena serangan tersebut membuat sistem dapat mengalami *overload* sistem akibat membanjiri jaringan *server* target dengan *request* yang bertubi-tubi. Kemudian bila di sandingkan dengan *distributed* DoS ini bisa menjadi lebih berbahaya karena menggunakan *bot* atau beberapa komputer *host* agar dapat melakukan serangan yang sama ke target tujuan, hasilnya serangan tersebut bisa mejadi dua kali lipat bahkan lebih banyak dari serangan DoS bisa dengan menggunakan *satu host* saja,
2.	Login Enumeration	*Enumeration* merupakan sebuah serangan *brute-force* yang memanfaatkan kumpulan data atau biasa yang disebut *wordlist* untuk mencocokan data maupun *credential* baik itu *username*, *password* atau data sensitif lainnya. Bisanya serangan ini paling banyak di lakukan untuk mencari *login* akses pada suatu layanan.

C. Tujuan

Tujuan penelitian ini adalah untuk mencari sistem keamanan yang kompetibel dari segi *Intrusion Prevention System* (IPS) pada perusahaan kerja yang penulis tempati saat ini, kemudian juga menjadi penangkal untuk serangan - serangan siber yang dapat meresat atau merusak baik itu infrastruktur ataupun aplikasi yang perusahaan miliki.

D. Organisasi Tulisan

Bagian selanjutnya pada penelitian ini adalah bagian 2 (dua) yang membahas studi terkait yang digunakan, selanjutnya yaitu bagian 3 (tiga) yang membahas perancangan sistem yang dibangun, kemudian bagian 4 (empat) membahas pengujian dan analisis penelitian, dan yang terakhir yaitu kesimpulan.

II. KAJIAN TEORI

A. Financial Technology

Financial Technology atau biasa disebut Fintech merupakan sebuah bidang industri dalam hal keuangan, dimana perusahaan atau organisasi memiliki produk berkaitan tentang jasa keuangan, Fintech sendiri merupakan industri yang baru lahir dengan perusahaan biasanya berdiri dengan nama *startup*. Fintech bertujuan untuk menarik pelanggan dengan produk dan layanan yang lebih ramah pengguna, efisien, transparan, dan otomatis daripada yang tersedia saat ini [2]. Dan dalam Fintech sendiri memiliki empat segmen atau bagian industri, diantaranya *Financing*, *Asset Management*, *Payments*, *Other Fintechs* [2].

Kemudian di Indonesia sendiri memiliki banyak basis industrinya, dan beberapa yang paling banyak digunakan adalah sebagai berikut:

1. *Peer-to-Peer Lending*

Jenis ini lebih di kenal dalam Fintech sebagai untuk peminjaman dana/uang, Fintech ini membantu masyarakat yang membuauhkan akses keuangan untuk memenuhi keutuhannya. Dengan begitu konsumen dapat meminjam uang sesuai yang diinginkan dengan mudah tanpa harus pergi ke kantor Bank.

2. *Microfinancing*

Fintech ini mememberikan layanan berupa keuangan bagi masyarakat kelas menengah kebawah dalam membantu kehidupannya sehari-hari seperti Usaha Mikro Kecil Menengah (UMKM). *Microfinanacing* yang diberikan seperti usaha modal bagi kelompokk UMKM tersebut.

3. *Digital Payment System*

Digital Payment System atau biasa disebut DPS ini bergerak pada bidang penyediaan layanan pembayaran, contohnya seperti tagihan - tagihan listrik, pulsa, air, kredit dll. Fintech ini bertujuan

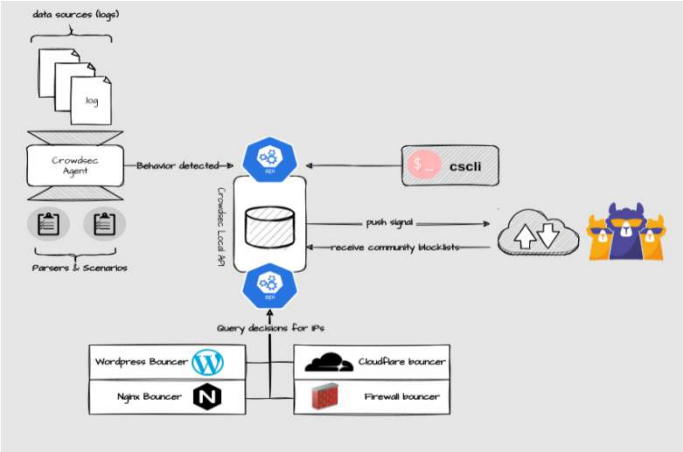
sama yaitu meringankan operasional masyarakat dalam melakukan pembayaran tagihan tersebut tanpa perlu menemui kantor pihak yang bersangkutan.

4. *Equity Crowdfunding*

Crowdfunding atau biasa disebut dengan penggalangan dana merupakan model bisnis Fintech yang membuat program seperti penggalangan dana atau donasi untuk masyarakatt ataupun acara yang ditujukan, contoh donasi terhadap masyarakat Palestina dll.

B. IDPS Crowdsec

Crowdsec adalah sebuah alat atau aplikasi *Intrusion Detection and Prevention System* (IDPS), Crowdsec ini bersifat *opensource* dan bisa digunakan secara bebas bagi pengguna manapun dan digunakan untuk analisis perilaku *user* dalam mengakses jaringan ke *server*, kemudian nantinya IP tersebut akan di blokir secara otomatis oleh sistem dan IP akan di jadikan *blocklist* yang disimpan ke dalam database [5].



GAMBAR 1
CROWDSEC WORKFLOW

Berbeda dengan temannya yaitu Fail2ban, Crowdsec ini memiliki keunggulan lain yaitu mampu untuk memonitoring banyak server dalam satu dashboard, jadi tidak perlu banyak melakukan konfigurasi dalam tiap servernya [5]. Adapun poin fungsi penting yang ada di dalam Crowdsec sendiri yaitu:

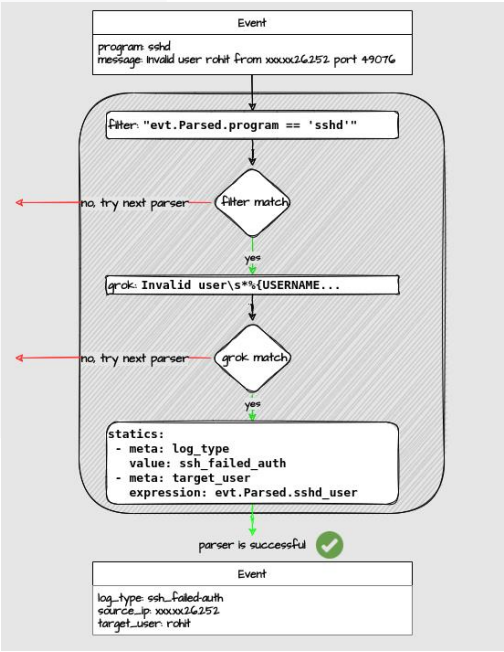
1. Alerts

   Alerts adalah representaasi *runtime* dari *bucker overflow* selama diproses oleh Crowdsec yang disematkan dalam *event* [5].
2. Bouncers

   Bouncers merupakan bagian kecil *softawre* yang bertugas untuk bertindak pada aktor yang memicu sebuah *alerts*, untuk melakukannya *bouncers* meminta konfirmasi dahulu ke LAPI untuk mengetahui apakah ada *decisions* berdasarkan IP *range*, *username*, dll [5].
3. Collections

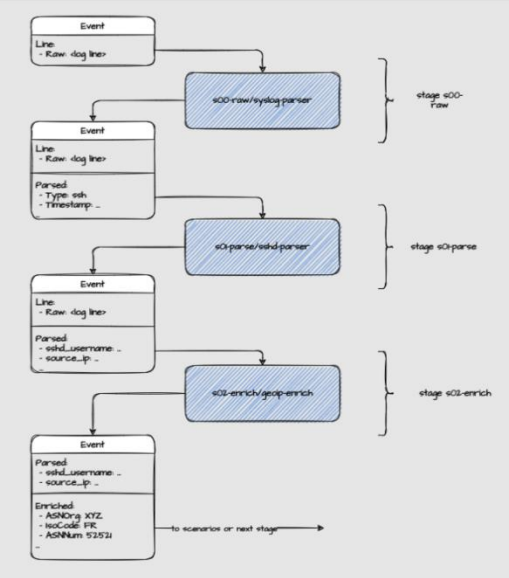
   Collections merupakan kumpulan - kumpulan *parsers*, *scenarios*, dan *post-overflow* yang membentuk paket koheren yang telah dibuat sebelumnya [5].
4. Parsers

   Merupakan file konfig YAML yang untuk mendefinisikan berapa banya *string* pada *log file* yang harus di *parsing*, dikatakan *sting* bila bermuat *log line* atau bagian terekstrak dari *parser* sebelumnya. Contoh aturannya seperti berikut:



GAMBAR 1
PARSERS WORKFLOW

Parser diatur dalam tahapan proses yang disebut dengan *stages*, seperti metode *queue* atau antrian yang mengharuskan *parser* tersebut sukses di eksekusi baru dapat melanjutkan kepada *stages* lainnya sampai akhir. Untuk gambarannya seperti berikut:



GAMBAR 2
STAGES WORKFLOW

5. Decisions

   Decisions adalah sebuah respresentasi *runtime* untuk mengatur atau membuat sebuah aturan atau *rule* dalam mem-*ban* seuatu IP *address* secara manual.

6. LAPI

   Local API (LAPI) adalah *service agent* yang berperan untuk mengelola dan mengatur *push signal* dari Crowdsec *client* lainnya, dia berperan juga sebagai *central* dan juga *controller* untuk setiap data yang disebar pada setiap Crowdsec *client*-nya [5].

7. CAPI

   Central API (CAPI) meruapakan pusat service dari setiap LAPI yang mem-*push* meta-data dari daerah manapun kepada *community blocklist*.

8. Dashboard

   Dashboard disini berperan sebagai media visual untuk menampilkan grafik data dari setiap *alerts, decisions, bouncers, ip address*.

C. DDoS dan Enumeration Attack

1. Distributed Denial of Service Attack

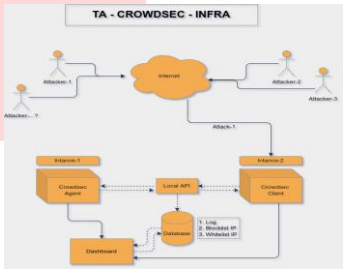
Pada serangan ini merupakan salah satu serangan yang paling sulit untuk di pertahankan sampai saat ini [6], serangan ini lebih mengganggu pada aktifitas yang sah dengan memakan sumber daya komputasi dan jaringan [6]. Penyerang akan mengirimkan paket secara bertubi - tubi ke target tujuannya, total pake yang dikirmkan tersebut biasanya bisa mencapai ribuan atau bahkan jutaan pake dalam satuan waktu. Dampak dari target serangan tersebut akan membuat sistem mengalami peninggian penerimaan data, sehingga sistem tidak mampu lagi dalam mengontrol atau mengelola paket yang masuk dan akhirnya sistem menjadi *down*. Untuk menanggunangi *down* dari sistem tersebut kita harus sesegera mungkin memutuskan koneksi *address* dari penyerang agar terputus secara cepat sehingga dampak yang di akibatkan tidak menjadi lebih besar.

2. Enumeration Attack

Enumeration atau serangan enumerasi merupakan sebuah metode serangan yang menggunakan sebuah kumpulan kosa kata yang memiliki puluhan atau bahkan ribuan kosa kata yang nantinya akan digunakan untuk mencocokan data dengan target layanan yang dilakukan. Contoh serangannya seperti mencari *user* dan *password login* aplikasi ataupun mencari *subdomain* dari sebuah *website.*

III. METODE

A. Gambaran sistem



GAMBAR 3
CROWDSEC FLOWCHART

Pada gambaran sistem diatas adalah bentuk gambaran kecil dari infrastruktur perusahaan yang saya tempati saat ini, untuk implementasinya disini Crowdsec berperan sebagai *agent* atau *central service* dari tiap - tiap Crowdsec *client* lainnya. Kemudian Crowdsec tersetbut akan di install pada tiap - tiap *service* yang ada, setalah itu akan dihubungkan kepada LAPI agar *meta-data* yang ada dapat disimpan dalam satu database saja dan di kontrol oleh LAPI/*agent* saja.

LAPI tersebut akan mengontrol dan menyimpan *meta-data* yang diperoleh dari serangan siber dari tiap - tiap *service* ke dalam database, dan akan membuat sebuah

B. Perangkat yang digunakan

Pada bagian ini perangkat yang digunakan dibagi menjadi dua bagian, yang pertama perangkat utama dan yang kedua adalah perangkat cadangan apabila perangkat pertama sampai saat waktu sidang ini tidak bisa dipakai dari izin perusahaan. Maka dari itu perangkat kedua ini berbasis jaringan lokal saja dan menggunakan *virtual machine* saja. Kemudian untuk perangkat yang digunakan akan disebutkan dibawah berikut:

Perangkat utama:

1. Hardware
   a. Amazon Web Service (EC2 Instance - RAM 2GB, SWAP 1GB)

2. Software
    a. Ubuntu Server 20.04 (Focal)
    b. Docker 20.10.17
    c. Openvpn 2.5.7
    d. Crowdsec 1.3.4 (alphaga)
    e. Metabase 0.41.5

Perangkat cadangan:

1. Hardware
    a. Laptop Infinix Inbook X1 (Intel i3-1005G1 RAM 8GB)
2. Software
    a. Ubuntu Server 22.04 LTS (jammy)
    b. Virtualbox 6.1.34
    c. Crowdsec 1.3.4 (alphaga)
    d. Metabase 0.41.5

C. Metode Pengujian

Dalam tahapan pengujian ini, saya melakukannya pada jaringan publik dengan 2 (dua) *instance* dari AWS. Kemudian pada tiap bagian pengujiannya nanti akan dilakukan sebanyak 6 (enam) kali dari masing - masing pengujian serangan, untuk pengujian 6 (enam) kali tersebut di definisikan menggunakan tiga *tools* serangan dan tiap *tools* itu akan di ujikan sebanyak dua kali untuk memastikan fungsional *tool* berjalan dengan baik atau tidak dan nilai akurasi deteksi dari Crowdsec. Adapun tahap pengujiannya sebagai berikut:

1. Pengujian pertama mengunakan teknik DoS *attack*, dari pengujian ini akan dilihat apakah serangan dapat membuat sistem menjadi tidak bekerja atau malah sebaliknya sistem dapat langsung memblok koneksi serangan yang dilakukan.
2. Pengujian kedua menggunakan *Enumeration attack* dan juga dengan *login* ssh biasa, dari pengujian ini akan dilihat apakah saat melakukan *enumeration* itu terdeteksi semua atau tidak serangan dalam mencari daftar direktori aplikasi.

Setelah proses pengujian berakhir, selanjutnya untuk tahap analisis hasil pengujian dengan menggunakan metode algoritma *Adaptive Agent-based Profiling* [7]. Analisis dilkakukan untuk menghitung *false alarm rate* pada sistem Crowdsec.

IV. HASIL DAN PEMBAHASAN

A. Tahapan Instalasi

Pada bagian tahapan instalasi ini, saya hanya akan memaparkan bagian instalasi pada Crowdsec-nya saja. Karena pada instalasi *operating system* yang digunakan alurnya sudah umum diketahui, dan juga untuk membuat penulisan menjadi lebih fokus kepada proses dan fungsi utama Crowdsec. Kemudian

1. Setup Crowdsec Service
    a. Tahap pertama kita harus menambahkan dahulu *repository* dari Crowdsec, karena Crowdsec bukan merupakan aplikasi resmi dari distribusi Linux jadi harus menambahkan secara manual dengan perintah
    curl -s https://packageloud.io/install/repositories/crowdsec/crowdsec/script.deb.sh | sudo bash
    b. Tahap kedua *update repository* dengan
    sudo apt update
    c. Setelah itu tahap ketida *install* Crowdsec *package*-nya dengan
    sudo apt install crowdsec
    d. Kemudian *install* bouncers untuk *firewall*, karena di Ubuntu ini menggunakan Iptables maka kita perlu *install* boucner untuk *firewall-iptables*-nya dengan
    sudo apt install crowdsec-firewall-bouncer-iptables
    e. Tahap kelima jalankan **crowdsec.service** dengan
    sudo enable crowdsec.service
    agar ketika *instance* mengalami *reboot* atau *shutdown* dan memulai ulang Crowdsec dapat jalan secara otomatis di latar belakang sistem.

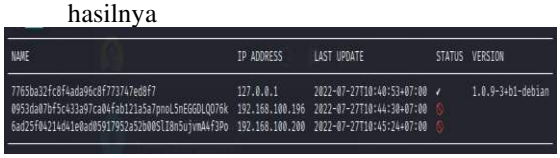
2. Setelah menginstall Crowdsec *service*, kita perlu menghubungkan tiap *instance* atau *server* menjadi satu pusat kendali. Dan cara yang perlu dilakukan adalah dengan mengatur LAPI tiap *instance* ke pusat *instance*, nanti segala bentuk data yang dikirim dan diperoleh tiap *instance* akan di atur oleh *instance* pusat ini atau disebut *agent*. *Agent* tersebut juga akan mengelola penyimpanan data yang telah diperoleh pada seluruh *instance* ke dalam database, sehingga perolehan data dari serangan akan bisa disebar ke seluruh *instance* sebagai daftar *decisions* barunya. Untuk cara mengaturnya sebagai berikut:

    a. Mengatur *address* pada LAPI terlebih dahulu di **/etc/crowdsec/local_api_credentials.yaml**, atur *address* URL dann port sesuai yang kamu inginkan. Namun dalam pengujian ini saya menggunakan *localhost* dengan *address* http://127.0.0.1:8080. Untuk *address* tersebut harus disesuaikan dengan *instance* - *instance* lainnya, karna *address* tersebut yang akan menjadi saluran *collaborative* setiap *instance*-nya.
    b. Kemudian lakukan *register* untuk LAPI ke *instance* pusat atau *instance-1* saya sebutkan disini, dengan cara
    sudo cscli lapi register -u http://127.0.0.1:8080.
    Untuk (**-u**) inisiasi URL yang telah diatur sebelumnya. Disini status keseluruhan

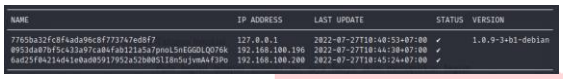
instance telah terdaftar dengan baik, hanya saja belum tervalidasi *id machines* yang ada, dengan perintah berikut

sudo cscli machines validate [id_name_machine]

jika telah melakukan validasi status *machine* akan berubah menjadi centang, dan bahwa *machine* telah benar di akui keasliannya. Seperti gambar berikut hasilnya



GAMBAR 4
STATUS MACHINES SETELAH TEREGISTRASI



GAMBAR 5
STATUS MACHINES SETELAH TERVALIDASI

3. Setup Crowdsec Bouncer Middleware

a. Tahap pertama *install* Crowdsec *bouncer-module* pada aplikasi NodeJS yang dibuat dengan
npm install @crowdsec/express-bouncer.

b. Setelah itu tahap kedua konfigurasi *bouncer* dalam *source code* tersebut sebagai *middleware* di dalam aplikasi seperti berikut:

```
const express = require("express");
const bodyParser =
require("body-parser");
const expressCrowdsecBouncer =
require("@crowdsec/express-bounc
er");
const crowdsecMiddleware =
require("./index.js");

(async () => {
// Configure CrowdSec
Middleware.
const crowdsecMiddleware = await
expressCrowdsecBouncer({
url: "http://127.0.0.1:2727",
apiKey:
"ef42f78726fe02bdfbcb934d137e5
93c",
});
```

```
  // Configure Express server.
  const app = express();

app.use(bodyParser.urlencoded({ ex
tended: true }));
  app.use(crowdsecMiddleware);

  // Create an example route.
  app.all("/", function (req, res) {
    res.status(200).send(`The way
is clear!`);
  });
```

c. Tahap ketiga perlu buat baru *bouncer* manual dengan perintah

```
  app.get("/api", function (req, res)
{
    res.status(200).send("API
page");
  });

  // Start server.
  app.listen(9000);
  console.log(`Server Up! );
})();

);
```

sudo cscli bouncers add [nama-bouncer] setelah itu ada dibuat juga API_KEY tersebut, lalu *copy-paste* pada variable `apiKey`.

d. Tahap keempat setelah *middleware* berhasil mengatur *middleware* selanjutnya menjalankan aplikasi yang kita buat dengan **pm2**, **pm2** sendiri merupakan sebuah *package manager* untuk men-*deployment* aplikasi yang di *development* dengan *framework* dari NodeJS, tujuannya agar memudahkan aplikasi dapat secara otomatis berjalan di latar belakang, sehingga tidak perlu lagi bagi pada *developer* menjalannya aplikasinya secara manual.

e. Tahap kelima yaitu *setup* Apache *proxy* agar aplikasi kita yang dibangun dari NodeJS dapat diakses secara otomatis melalui *port* 80 dari *browser*. Namun sebelum mengkonfigurasi *file* nya kita perlu install *bouncer-module* lagi untuk Apacher *service* ini dengan

B. Tahapan Pengujian

Dalam tahapan pengujian ini dilakukan dalam jaringan lokal saja dengan meunggunakan VPN, dan akses ini saya lakukan dalam *cloud instance* perusahaan. Sehingga

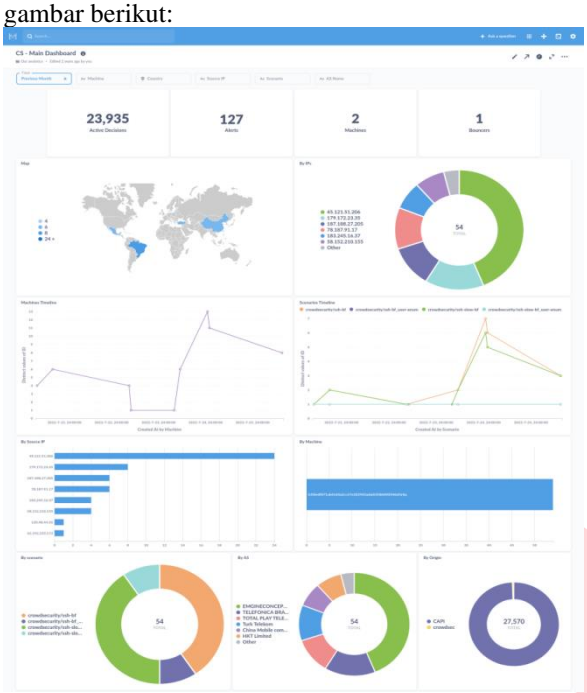
C. Analisis Hasil Pengujian

Pada bagian ini, untuk menganalisis pengujian dibagi menjadi dua bagian karena merujun pada tujuan penelitian dan juga implementasi berbeda. Bagi analisis pertama bersumber pada pengujian monitoring *multiple endpoint* dan analisis kedua pada *filtering endpoint* menggnakan *framework* ExpressJS.

a. Multiple Endpoint

Pada analisis berikut hasil uji yang dilakukan untuk *multiple endpoint* secara instalasi berhasil sepenuhnya, dan dapat sinkron antar satu *instance*

dengan *instance* lainnnya. Untuk dashboardnya pada gambar berikut:

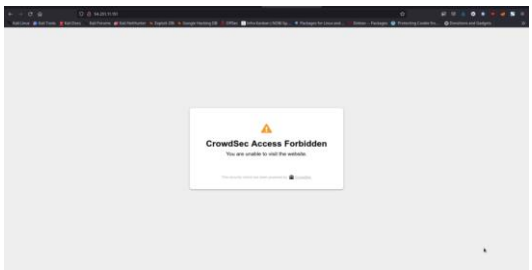


GAMBAR 6
DASHBOARD CROWDSEC

Kemudian untuk *active decisions* diatas meruapakan total daftar IP yang terblokir dan tersimpan pada database internal, dan *alerts* adalah sebagai alarm yang dibaca sebagai total serangan dari berbagai sumbernya, selanjutnya *bouncers* ini berperan sebagai eksekutornya, dia berkerja di dalam sistem dengan *firewall* untuk melakukan *blocking* pada *address* yang sudah ditandai blokir sebelumnya agar *address* tersebut tidak bisa mengakses ke sistem tujuannya kembali. Dan berikut tampilan *dashboard* lainnya yang menampilkan *timeline* serangan, total serangan yang dilakukan *address attacker*, dan juga jenis serangan yang mereka luncurkan pada sistem.

b. Filtering Endpoint

Pada bagian ini untuk hasil uji yang di lakukan sudah berhasil, disini saya membuat dua direcroty satu '/' dan kedua '/api'. Dan pengujian yang dilakukan menggunakan *tools* Nikto, Nikto sendiri merupakan tools untuk untuk *enumeration* alamat *host* yang diserang, dimana lebih untuk mendeteksi berapa alamat *host* yang tersedia. Berikut untuk hasil *screenshot output*-nya:



GAMBAR 7
HASIL BAN ACCESS DARI WEBSITE

Dari hasil blokir tersebut si *attacker* tidak akan dapat mengakses ke halaman web yang dituju.

c. Analisis dengan algoritma *Adaptive Agent-based Profiling*

Dari hasil pengujian serangan disini saya melakukan sebanyak 18 (delapan belas) kali pengujian serangan dan 5 (lima) hasil deteksi serangan yang berhasil dari kategori serangan *enumeration*. Kemudian menghitung *false alarm rate* [7] dari Crowdsec, apakah Crowdsec dapat benar - benar akurat dalam mendeteksi seluruh serangan yang ada.

TABLE 3.
TOTAL SERANGAN YANG DIPEROLEH

No	Jenis Serangan	Jumlah Serangan									Total Serangan
		1	2	3	4	5	6	7	8	9	
1.	Denial of Service (DoS)	0	0	0	0	0	0	0	0	0	0
2.	Enumeration	1	1	1	0	0	1	1	0	0	5
											5

Keterangan:
Nilai 1 = Serangan terbaca dan terblokir
Nilai 0 = Serangan terbaca dan tidak terblokir

Kemudian untuk rumus yang dipakai yaitu sebagai berikut:

$$D = \frac{\sum_{i-1}^{n} dAi}{n}$$

Dengan:
D = Detection
n = Dimensi data
i = Representasi atribut
dAi = *Primary decision value* (dA1, dA2, dA2,...,dAn)
Maka data yang dipakai yaitu
Untuk kategori DDoS

$$D = \frac{0+0+0+0+0+0+0+0+0+0}{9} \times 100 = 0\%$$

Kemudian kategori *Enumeration*

$$D = \frac{1+1+1+0+0+1+1+0+0}{9} \times 100 = 55.55\%$$

Sehingga total persentase akurasi deteksi dari Crowdsec terhadap serangan yang diperoleh adalah sebesar 55.55 %.

V. KESIMPULAN

A. Kesimpulan

Dari hasil pengujian dan analisis pada Bab sebelumnya, dapat disimpulkan bahwa Crowdsec merupakan *tools* yang sangat efektif di implementasikan sebagai IDPS untuk memblokir pengguna yang mencurigakan secara otomatis sehingga pengguna tidak dapat akses *service* tujuan bahkan dengan di *refresh* halaman atau jaringan yang digunakannya selaman IP pengguna tersebut belum di

hapus. Kemudian teruntuk implementasi menjadi *bouncers* sebagai *middleware* di *framework* ExpressJS juga sangat membantu dalam *filtering* akses pengguna juga ke dalam halaman web yang bsersifat terbatas atau rahasia.

B. Saran

1. Saran pertama, dari hasil penelitian yang telah dilakukan ini, masih harus dapat dikembangkan kembali bagi karena terbatasnya penggunaan *device* dan juga modal penelitian alangkah lebih baiknya penggunaan Crowdsec pada *multiple endpoint* dilakukan menggunakan mininal lebih dari lima *instance*. Agar hasil yang diperoleh lebih jelas bagaimana *flow* kerja dari Crowdsec itu sendiri dalam mendeteksi, mengontrol, dan memblokir *malicious* IP yang ingin mengakses atau merusak sistem kita sendiri.
2. Saran kedua, dalam penelitian menggunakan *framework* NodeJS ini lebih baik lebih membangun aplikasi yang sedikit lebih besar lagi, agar *flow* kerja dalam uji *filtering endpoint* dalam dilihat lebih jauh kembali dan juga lebih banyak contoh hasil yang di dapatkan nantinya bila hasil yang dicari mengenai kualitas Crowdsec.

REFERENSI

[1] Gregor Dorfleitner, Lars Hornuf, Matthias Schmitt, Martina Weber. "*Definitions of Fintech and Description of the Fintech Industry*". Springer 2017.

[2] Michael Nofer, Peter Gomber, Oliver Hinz, Dirk Schiereck. "*Blockhain*". Springer Fachmedien Wiesbaden 2017, 20 March 2017.

[3] Dr. James Mwikya Reuben, Dr. Johnmark Obura. "*Intrusion Detection and Prevention of Cyber-threats using Open-Source Software for Fintech Startup Firms in Kenya"*. KyU 4th Annual Virtual International Conference, 2021.

[4] A. Saravanan, S. Sathya Bama. "*A Review on Cyber Security and The Fifth Generation Cyberattacks*". Oriental Journal of Computer Science and Technology (OJCST) Vol.12 No.2 2019.

[5] Crowdsec Documentation Setup. 2022. Available Online: https://doc.crowdsec.net/docs/.

[6] Prof. Bill Buchanan, Flavien Flandrin, Richard Macfarlen, Dr. Jamie Graves. "*A Methodology to Evaluate Rate-Based Intrusion Prevention System Against Distributed Denial-of-Service* (DDoS). Edinburgh Napier University.

[7] Jusia Amanda Ginting, Irwan Sembiring, S.T.,M.Kom. "Deteksi *False Alarm* Pada *Intrusion Detection System* (IDS) Menggunakan Algoritma *Adaptive Agent-Based Profiling*. Jurnal Ilmiah Tugas Akhir Universitas Kristen Satya Wacana.