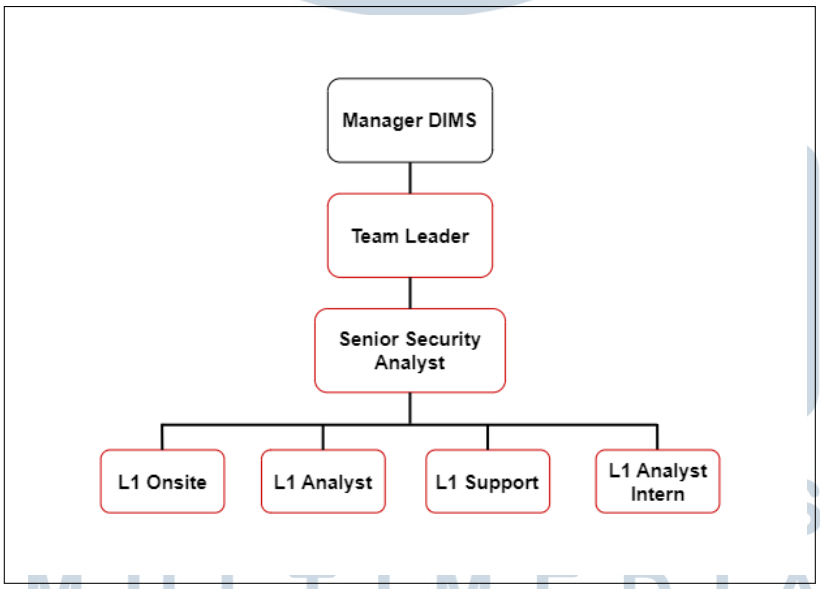
BAB 3
PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Selama pelaksanaan magang di Defenxor, posisi yang ditempati setara dengan L1 Security Analyst yang berperan dalam kegiatan operasional *Security Operations Center* (SOC). Tanggung jawab utama meliputi pelaksanaan security monitoring dan incident handling untuk mendukung keberlangsungan sistem keamanan informasi perusahaan, serta pengimplementasian project yang diberikan.

Kegiatan operasional SOC dilakukan dalam tim yang terdiri atas *security analyst*, *system administrator* (sysadmin), dan staf *security device management* (SDM) *engineer*. Seluruh anggota tim berada di bawah divisi *Defenxor Intelligence Managed Security* (DIMS) yang dipimpin oleh Andi Wahyudi selaku *Team Leader SOC Operation*. Struktur organisasi DIMS secara umum dapat dilihat pada Gambar 3.1, yang menampilkan hubungan koordinasi antara masing-masing peran dalam tim analis keamanan.



Gambar 3.1. Struktur DIMS

Sumber: Dokumen Pribadi [7]

Dalam pelaksanaan koordinasi kerja, aktivitas diawali oleh tim *security analyst* yang melakukan pemantauan dan analisis insiden secara bergantian sesuai jadwal shift. Setiap anggota tim berkolaborasi dalam mendeteksi, menginvestigasi, serta melaporkan potensi ancaman yang muncul. Apabila ditemukan kendala teknis

atau kebutuhan sistem, koordinasi dilakukan dengan *sysadmin*. Sementara itu, kebutuhan yang berkaitan dengan komunikasi kepada pihak klien dikoordinasikan melalui L2.

Apabila suatu insiden tidak dapat diselesaikan di tingkat analis, dilakukan proses eskalasi, yaitu dari L1 *security analyst* kepada L2 (*senior security analyst*), kemudian ke L3 (*Team Leader*), dan apabila masih memerlukan keputusan lebih lanjut, diteruskan kepada Manager DIMS. Proses koordinasi ini memastikan setiap insiden tertangani secara sistematis dan efisien sesuai tingkat urgensi dan kompleksitasnya.

3.2 Tugas yang Dilakukan

Pelaksanaan program magang pada tahap kedua difokuskan pada kegiatan yang berkaitan langsung dengan pengoperasian dan pengembangan sistem keamanan informasi di lingkungan SOC Defenxor. Tanggung jawab utama mencakup aktivitas security monitoring, pelaporan hasil analisis, serta implementasi *host intrusion detection system*.

Kegiatan pertama adalah melakukan monitoring terhadap sistem *Security Information and Event Management* (SIEM) yang digunakan dalam operasional SOC. Proses ini dilakukan untuk mengamati event log dari berbagai sumber sistem klien secara *real time*, mengidentifikasi aktivitas mencurigakan, dan memastikan tidak terjadi anomali yang berpotensi menjadi insiden keamanan. Hasil dari analisis alert kemudian disusun dalam bentuk notifikasi yang dikirimkan kepada pihak klien sebagai bentuk pelaporan dan rekomendasi tindak lanjut.

Selain aktivitas harian dalam *monitoring*, dilakukan pula penyusunan laporan bulanan (*monthly report*) yang berisi rekapitulasi temuan *security event* pada sistem klien. Laporan ini mencakup statistik jumlah *alert*, kategori ancaman, pola serangan yang teridentifikasi, serta rekomendasi langkah mitigasi yang dapat diterapkan untuk meningkatkan keamanan sistem klien. Penyusunan laporan ini berfungsi sebagai evaluasi rutin terhadap efektivitas mekanisme deteksi dan pencegahan ancaman.

Sebagai bagian dari pengembangan kemampuan teknis dan pemahaman terhadap sistem deteksi intrusi, dilakukan juga kegiatan implementasi dan konfigurasi Wazuh HIDS. Proses ini meliputi instalasi agent HIDS, pengaturan kebijakan deteksi, serta integrasi dengan platform SIEM yang digunakan di lingkungan simulasi SOC. Implementasi ini bertujuan untuk mensimulasikan

operasional sistem deteksi intrusi berbasis *host* dan memahami alur korelasi data *log* dari *endpoint* ke SIEM.

3.3 Uraian Pelaksanaan Magang

Kegiatan magang tahap kedua menjalankan implementasi sistem *Host Intrusion Detection System* (HIDS) menggunakan platform Wazuh yang terintegrasi dengan Elastic Stack (ELK) sebagai sarana simulasi operasional Security Information and Event Management (SIEM) di lingkungan SOC. Implementasi ini bertujuan untuk memahami mekanisme deteksi intrusi pada tingkat host serta bagaimana data log yang dihasilkan dapat dikorelasikan dan divisualisasikan melalui dashboard SIEM untuk mendukung analisis keamanan.

Wazuh digunakan karena merupakan platform keamanan bersifat open source yang menyediakan fungsi deteksi intrusi, pemantauan integritas sistem, analisis log, serta deteksi ancaman berbasis kebijakan. Dalam implementasi ini, HIDS berperan untuk memonitor aktivitas sistem secara *real-time* melalui agent yang dipasang pada host target, sementara data hasil deteksi dikirim ke server Wazuh yang kemudian diolah dan ditampilkan menggunakan Kibana sebagai tampilan visualisasi utama.

Selain proses implementasi, kegiatan magang juga meliputi monitoring terhadap security events dan agent status pada lingkungan simulasi. Melalui aktivitas ini, diperoleh pemahaman tentang bagaimana sistem HIDS bekerja dalam mengidentifikasi anomali serta bagaimana data dari host dikonsolidasikan dalam platform SIEM untuk mendukung proses analisis insiden keamanan.

Tabel 3.1. Pekerjaan yang dilakukan tiap minggu selama pelaksanaan kerja magang

Minggu Ke -	Pekerjaan yang dilakukan
1-4	Melakukan aktivitas harian berupa pemantauan sistem keamanan melalui platform SIEM. Mengamati dan menganalisis security logs untuk mendeteksi ancaman serta anomali yang terjadi pada sistem klien. Menyusun notifikasi insiden berdasarkan hasil temuan dan mengoordinasikan laporan dengan tim SOC.
Lanjutan pada halaman berikutnya	

Tabel 3.1. Pekerjaan yang dilakukan tiap minggu selama pelaksanaan kerja magang (Lanjutan)

Minggu Ke -	Pekerjaan yang dilakukan
4-8	Melanjutkan kegiatan monitoring harian, serta mulai mempelajari konsep dan arsitektur HIDS menggunakan Wazuh dan ELK Stack. Mempelajari alur komunikasi antar-komponen seperti Wazuh Agent, Manager, dan Elasticsearch, serta meninjau dokumentasi instalasi dan konfigurasi.
8-12	Melakukan proses implementasi sistem HIDS di lingkungan virtual, mencakup instalasi Wazuh Manager, Wazuh Agent, Filebeat, serta integrasi dengan Elasticsearch dan Kibana. Melakukan konfigurasi awal untuk pengumpulan log, verifikasi konektivitas antar-komponen, serta pengujian fungsi deteksi terhadap aktivitas sistem.
12-16	Melanjutkan tahap penyempurnaan dan pengujian HIDS, termasuk integrasi dengan VirusTotal API dan konfigurasi *security alerts* serta *active response*. Melakukan dokumentasi hasil pengujian, pembuatan laporan akhir, serta evaluasi efektivitas sistem dalam mendeteksi ancaman. Monitoring SOC tetap dilaksanakan sebagai tugas rutin harian.

3.3.1 Perancangan

Perancangan sistem menjadi tahap awal dalam pembangunan prototipe Host Intrusion Detection System (HIDS), karena seluruh proses implementasi dan pengujian sangat bergantung pada kejelasan arsitektur yang dirancang sejak awal. Pada tahap ini, ditentukan bagaimana komponen-komponen utama—seperti Wazuh Agent, Wazuh Manager, Filebeat, Elasticsearch, dan Kibana—akan saling berkomunikasi untuk membentuk alur kerja deteksi intrusi yang utuh. Selain itu, perancangan dilakukan untuk memastikan prototipe yang dibangun benar-benar mampu merepresentasikan mekanisme HIDS dalam lingkungan Security Operations Center (SOC), baik dari sisi pengumpulan log, korelasi event, hingga penyajian alert kepada analis.

A Tujuan dan Fokus Perancangan

Perancangan sistem HIDS ini difokuskan untuk menyediakan kemampuan deteksi intrusi pada tingkat *host* guna mendukung operasional SIEM dalam lingkungan SOC. Fokus utama pengembangan diarahkan pada kemampuan memonitor aktivitas *endpoint* secara mendalam, seperti perubahan *file system*, *command*, *authentication*, dan kejadian lain yang berpotensi mengindikasikan ancaman. Dengan pendekatan *host-based detection*, prototipe ini bertujuan memberikan visibilitas yang lebih granular terhadap aktivitas internal perangkat, sehingga dapat melengkapi analisis berbasis log jaringan yang biasanya membentuk inti dari sistem SIEM.

Tujuan lain dari perancangan ini adalah mengilustrasikan alur kerja deteksi intrusi mulai dari pengumpulan log di sisi *endpoint*, pemrosesan dan korelasi oleh Wazuh Manager, hingga visualisasi dan analisis melalui dashboard SIEM. Dengan demikian, rancangan ini tidak hanya berfungsi sebagai fondasi implementasi HIDS, tetapi juga sekaligus menjadi sarana pembelajaran untuk memahami bagaimana data diolah dan ditransformasikan menjadi alert keamanan yang siap ditindaklanjuti oleh analis SOC.

B Lingkup Prototipe

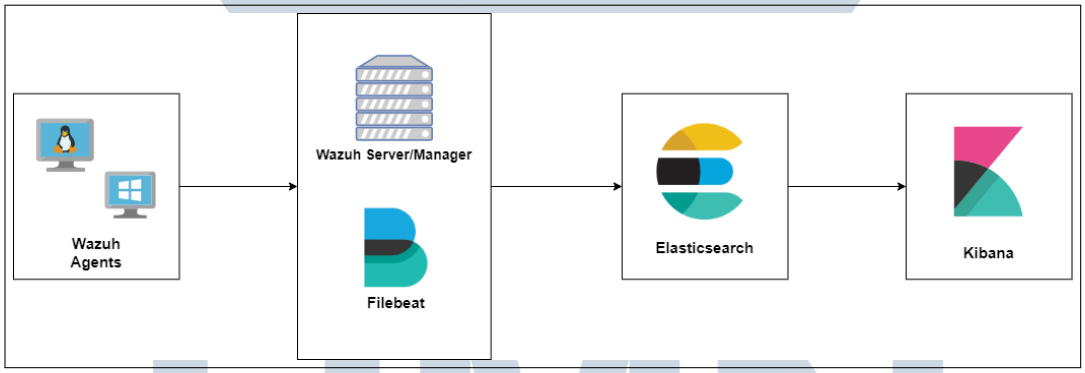
Lingkup pengembangan prototipe ini dibatasi pada skenario implementasi HIDS dalam lingkungan simulasi SOC dengan fokus pada deteksi dan pemantauan aktivitas pada host. Sistem dirancang sebagai prototipe fungsional, bukan sebagai deployment skala produksi, sehingga tidak mengimplementasikan *clustering* dan *load balancing* agar sistem menjadi seringan mungkin. Pembatasan ini dilakukan agar pengembangan tetap terfokus pada alur teknis inti, yaitu pengumpulan log, pemrosesan event, korelasi data, dan visualisasi hasil deteksi.

Prototipe ini hanya mencakup komponen utama untuk operasional HIDS, yaitu Wazuh Agent sebagai sumber data log, Wazuh Manager sebagai pusat pemrosesan dan korelasi, Filebeat sebagai modul pengirim log ke Elasticsearch, serta Elasticsearch dan Kibana sebagai komponen penyimpanan dan visualisasi. Dengan ruang lingkup yang lebih terarah, prototipe ini mampu menunjukkan fungsi fundamental HIDS secara jelas dan terukur tanpa kompleksitas tambahan yang belum dibutuhkan dalam konteks pengembangan sistem skala kecil untuk tujuan pembelajaran.

C Komponen dan Topologi Sistem

Implementasi sistem HIDS ini dilakukan menggunakan platform Wazuh, yang diintegrasikan dengan Elastic Stack (ELK) sebagai sarana penyimpanan, pengolahan, dan visualisasi data log. Sistem ini dirancang untuk mensimulasikan lingkungan SOC dalam mendeteksi, mengelola, serta menganalisis peristiwa keamanan yang terjadi pada tingkat host.

Topologi sistem terdiri atas beberapa bagian utama, yaitu Wazuh Agent, Wazuh Manager/Server, Filebeat, Elasticsearch, dan kibana, Wazuh Agent berfungsi sebagai pengumpul data log dari host, kemudian data tersebut dikirim ke Wazuh Manager untuk dianalisis. Filebeat bertugas meneruskan hasil analisis log ke Elasticsearch, yang kemudian divisualisasikan melalui Kibana dalam bentuk dashboard yang interaktif. Gambar 3.2 adalah topologi dari sistem HIDS diimplementasikan.



Gambar 3.2. Topologi Wazuh HIDS

Sumber: Dokumen Pribadi [7]

C.1 Wazuh Agent

Komponen ini berfungsi sebagai pengumpul informasi dari host tempat agent dipasang. Wazuh Agent memantau aktivitas sistem seperti perubahan file, eksekusi proses, koneksi jaringan, dan kejadian login. Data hasil pemantauan dikirim secara terenkripsi ke Wazuh Manager menggunakan protokol komunikasi yang aman. Dengan mekanisme ini, setiap aktivitas pada host dapat tercatat dan dianalisis secara *real-time* oleh sistem pusat. Agent juga dapat dikonfigurasi untuk memantau direktori atau layanan tertentu sesuai kebutuhan simulasi keamanan.

C.2 Wazuh Manager / Server dan Filebeat

Wazuh Manager bertugas sebagai pusat pengelolaan dan analisis data log yang diterima dari seluruh agent. Manager melakukan proses normalisasi, deteksi pola ancaman, serta mengelompokkan event berdasarkan *severity level*nya. Di sisi lain, Filebeat digunakan untuk menyalurkan *data log* hasil analisis dari Wazuh Manager ke Elasticsearch. Integrasi antara kedua komponen ini memastikan aliran data log berjalan efisien dan terstruktur, sehingga sistem dapat menampilkan alert dan notifikasi secara akurat.

C.3 Elasticsearch

Komponen ini berfungsi sebagai basis penyimpanan data log yang terpusat. Semua *alert* dan *event* yang dikirim dari Filebeat disimpan dalam index Elasticsearch, sehingga dapat diakses dan dicari dengan cepat menggunakan *search querry*. Elasticsearch juga berperan dalam pengelolaan data dalam jumlah besar dan memberikan fondasi bagi analisis data yang mendalam di lingkungan SOC.

C.4 Kibana

Kibana menjadi tampilan visual dari keseluruhan sistem. Melalui dashboard Wazuh, *user* dapat memantau status keamanan sistem, meninjau alert berdasarkan tingkat risiko, serta mengamati tren aktivitas host dalam jangka waktu tertentu. Visualisasi ini membantu dalam melakukan analisis cepat terhadap peristiwa keamanan, memudahkan proses investigasi insiden, serta meningkatkan efektivitas pengambilan keputusan dalam operasional keamanan.

3.3.2 Hasil Implementasi Prototipe HIDS

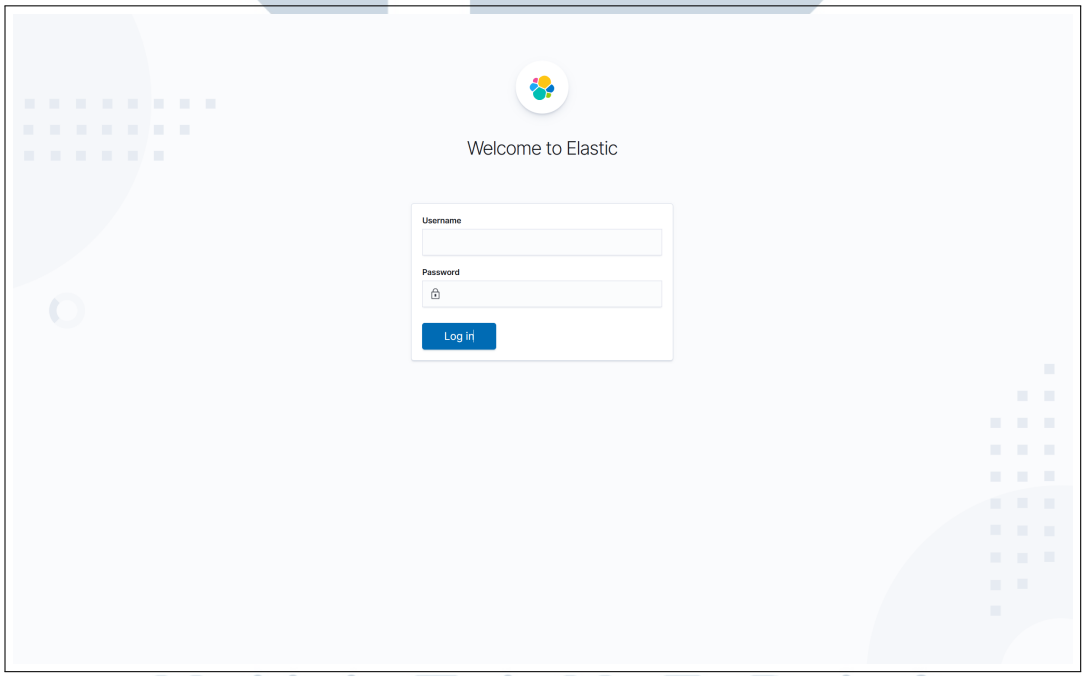
A Wazuh ELK SIEM

Implementasi Wazuh yang terintegrasi dengan Elastic Stack (ELK) menghasilkan sebuah platform pemantauan keamanan yang mampu menampilkan data log, event, dan aktivitas sistem secara terpadu melalui *interface* SIEM. Integrasi tersebut memungkinkan proses korelasi log antara Wazuh Manager, Filebeat, Elasticsearch, dan Kibana berjalan secara konsisten sehingga seluruh informasi yang dikirim oleh agent dapat divisualisasikan dengan jelas. Penjelasan

berikut menyajikan hasil implementasi antarmuka mulai dari proses masuk ke dashboard, eksplorasi log, hingga akses menu Wazuh untuk kebutuhan analisis keamanan.

A.1 Login Page

Halaman login merupakan tahap awal dalam proses autentikasi *user* untuk mengakses tampilan Wazuh yang terintegrasi dengan Elastic Stack (ELK). Tampilan halaman ini merupakan bagian dari tampilan Kibana yang berfungsi sebagai front-end visualisasi data log hasil deteksi HIDS. Pada halaman login, *user* diminta untuk memasukkan kredensial yang telah ditentukan untuk memastikan bahwa hanya pihak yang memiliki hak akses yang dapat masuk ke dalam sistem. Dari sisi keamanan, proses autentikasi ini telah diamankan dengan protokol HTTPS menggunakan self-signed certificate yang dihasilkan saat proses deployment, sehingga seluruh komunikasi antara *user* dan server terenkripsi dengan baik.



Gambar 3.3. *Login page SIEM*
Sumber: Dokumen Pribadi [7]

Pada konfigurasi awal sistem, akses login dilakukan menggunakan akun bawaan yang telah diset selama proses instalasi dan pengaturan awal komponen Elastic Stack. Username dan password ini kemudian diatur melalui modul keamanan yang tersedia di dalam Elastic Stack untuk membatasi akses hanya

kepada *user* tertentu. Implementasi keamanan ini memastikan bahwa setiap *user* yang mengakses sistem telah terverifikasi dengan benar.

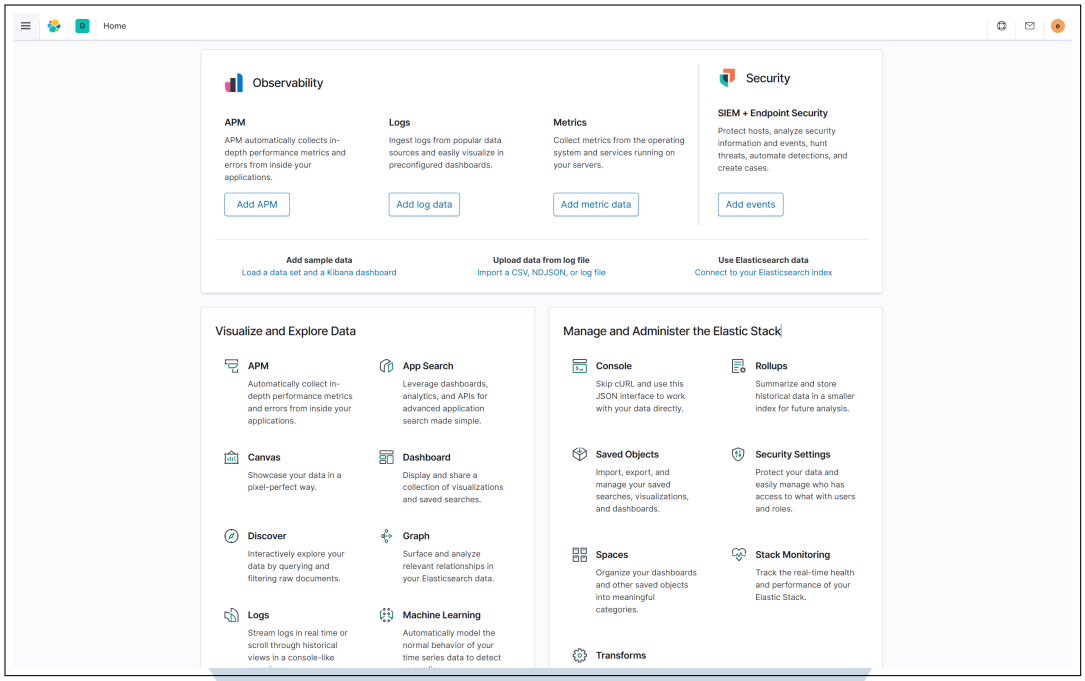
Mekanisme autentikasi pada sistem Wazuh ELK ini diperkuat menggunakan X-Pack Security Module yang merupakan fitur bawaan dari Elastic Stack untuk melindungi akses *user* serta memastikan keamanan data. Modul ini mengaktifkan autentikasi berbasis sertifikat dan *role-based access control* (RBAC), di mana setiap *user* memiliki hak akses tertentu sesuai perannya dalam sistem. Konfigurasi X-Pack dilakukan melalui file konfigurasi elasticsearch.yml dan kibana.yml, dengan parameter penting seperti `xpack.security.enabled: true`, `xpack.security.http.ssl.enabled: true`, dan path sertifikat yang digunakan pada direktori certs.

Selain mengaktifkan autentikasi dan enkripsi, X-Pack juga menyediakan fitur otorisasi yang memungkinkan administrator membatasi ruang lingkup akses *user* hanya pada indeks atau dashboard tertentu. Sertifikat yang digunakan untuk mengamankan komunikasi antar-komponen, seperti antara Elasticsearch dan Kibana, dihasilkan secara *self-signed* menggunakan utilitas bawaan Elastic Stack. Hal ini memastikan bahwa seluruh pertukaran data antar-komponen terjadi melalui koneksi yang aman dan terenkripsi.

Dengan adanya konfigurasi keamanan melalui X-Pack, sistem Wazuh ELK yang diimplementasikan dapat menjamin integritas dan kerahasiaan data log yang dikumpulkan oleh HIDS. Selain itu, penggunaan *role-based access control* dan autentikasi berbasis sertifikat memperkuat keandalan sistem dengan memastikan bahwa hanya *user* yang sah dan terotorisasi yang dapat mengakses dashboard serta melakukan tindakan administratif di dalam lingkungan SIEM.

A.2 Home Page SIEM

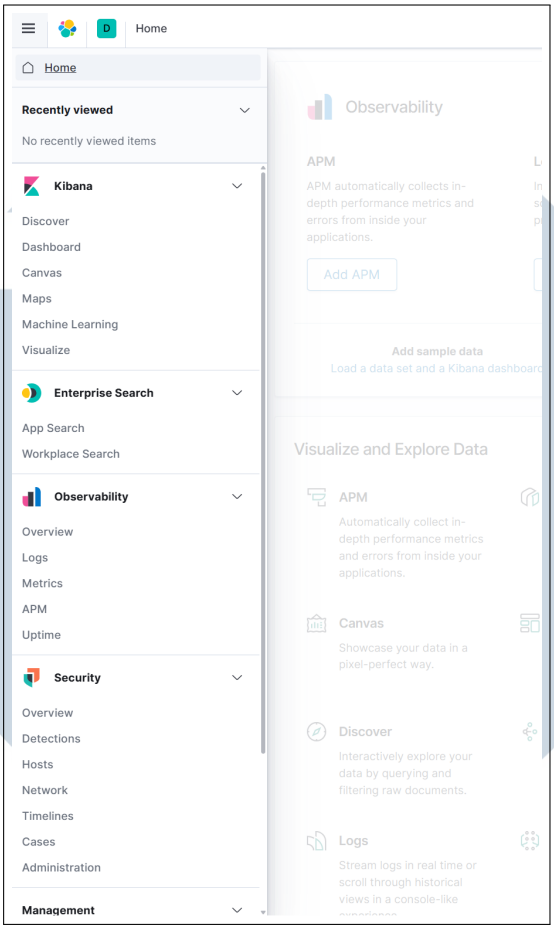
Setelah authentication, akan diarahkan menuju halaman utama atau home page dari sistem Wazuh–Kibana SIEM. Tampilan awal ini berfungsi sebagai titik masuk utama untuk seluruh aktivitas pemantauan dan analisis keamanan. Pada halaman ini, dapat dilihat berbagai menu dan fitur yang tersedia dalam sistem, termasuk akses menuju Discover, Dashboard, Visualize Library, dan Wazuh App. Tampilan yang terorganisasi dengan baik memungkinkan *user* untuk dengan mudah mengakses area yang diinginkan tanpa perlu melakukan navigasi yang kompleks.



Gambar 3.4. *Home Page*

Sumber: Dokumen Pribadi [7]

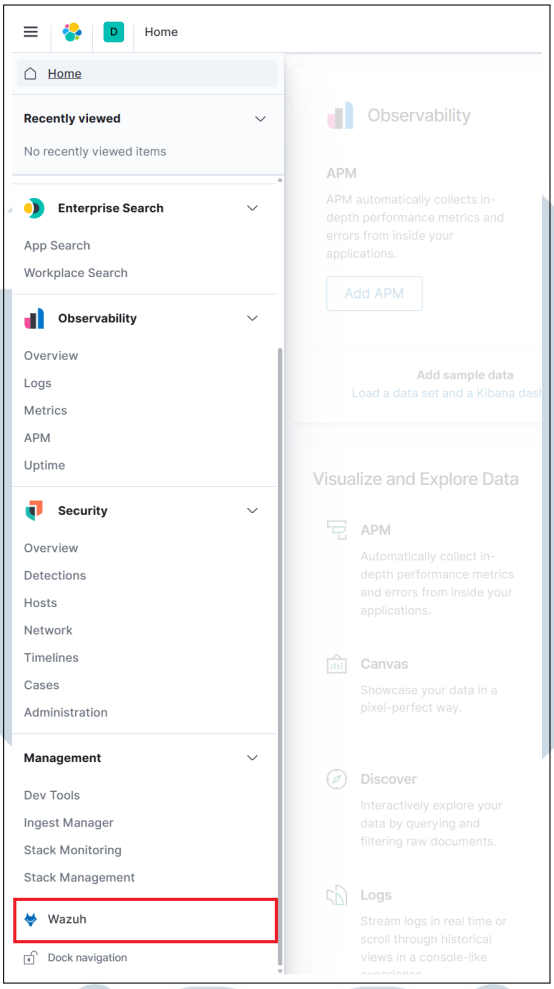
Seperti terlihat pada Gambar 3.4, halaman utama menampilkan tata letak tampilan yang sederhana dan fungsional. Panel navigasi di sisi kiri berisi menu utama untuk menjelajahi berbagai fitur analisis data, sementara area tengah berfungsi sebagai ruang kerja utama. Terdapat juga bagian *quick access panel* yang menyediakan tautan menuju *recent dashboards* dan *saved visualizations*.



Gambar 3.5. *Sidebar*

Sumber: Dokumen Pribadi [7]

Gambar 3.5 memperlihatkan bagian menu navigasi yang menjadi pusat pengendalian dalam sistem. Melalui menu ini, *user* dapat memilih untuk melihat data log pada halaman Discover, menampilkan data visual melalui Dashboard, atau mengakses halaman khusus Wazuh untuk melihat hasil deteksi HIDS. Menu ini juga menyediakan akses ke Management Section untuk melakukan pengaturan indeks dan *user*. Keberadaan menu navigasi yang terintegrasi mempermudah *user* untuk berpindah antarfitur tanpa perlu keluar dari tampilan utama.



Gambar 3.6. Wazuh App

Sumber: Dokumen Pribadi [7]

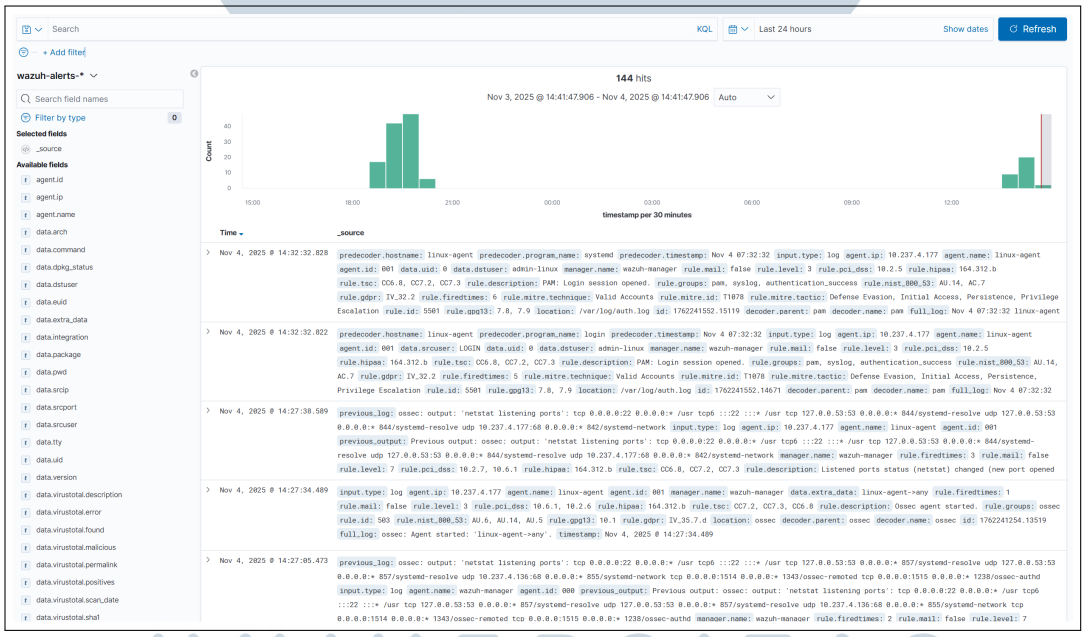
Sebagaimana terlihat pada Gambar 3.6, salah satu integrasi antara Wazuh dan Elastic Stack adalah menu Wazuh di sidebar Kibana. Integrasi ini memungkinkan sistem untuk menampilkan data hasil deteksi HIDS secara langsung di dalam tampilan SIEM tanpa memerlukan aplikasi tambahan. Wazuh App berfungsi sebagai modul visualisasi khusus yang menampilkan berbagai informasi keamanan seperti *security events*, *integrity monitoring*, *agent management*, dan fitur lainnya. Integrasi ini menjadikan Wazuh sebagai platform SIEM yang komprehensif dan mudah digunakan oleh tim SOC.

Halaman utama SIEM dalam aktivitas operasional harian SOC menjadi titik awal dalam menentukan area analisis atau pemantauan yang akan dilakukan. Dari halaman ini, analis dapat langsung berpindah ke Discover Page untuk meninjau log secara keseluruhan, atau menuju Security Events Dashboard untuk melakukan investigasi terhadap aktivitas yang mencurigakan. Dengan demikian, home page

berperan sebagai pusat navigasi strategis yang menghubungkan berbagai komponen dalam sistem SIEM.

A.3 Log Discover Page

Halaman Discover digunakan untuk melakukan analisis awal terhadap data log yang dikumpulkan oleh sistem. Melalui halaman ini, seluruh log hasil pemantauan dari Wazuh agents yang terhubung akan dikirim ke Elasticsearch dan dapat diakses secara langsung dalam bentuk data mentah. Halaman ini berfungsi sebagai titik awal bagi analis keamanan untuk menelusuri aktivitas sistem, mengidentifikasi anomali, serta melakukan pencarian terhadap kejadian tertentu sebelum data tersebut divisualisasikan dalam dashboard. Dengan demikian, Discover Page menjadi komponen penting dalam mendukung proses investigasi keamanan secara menyeluruh.

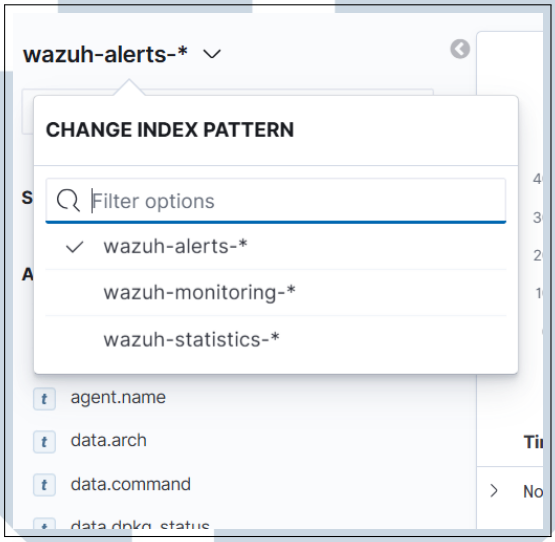


Gambar 3.7. *Discover Page*

Sumber: Dokumen Pribadi [7]

Seperti terlihat pada Gambar 3.7, halaman Discover menampilkan tampilan yang dirancang untuk memudahkan proses eksplorasi data. Bagian sebelah kiri berisi daftar field yang tersedia pada log, di mana *user* dapat memilih atribut tertentu untuk ditampilkan atau disembunyikan. Di bagian tengah, data log yang sesuai dengan filter pencarian ditampilkan secara real-time dengan informasi seperti timestamp, nama agent, jenis event, dan level risiko. Sementara itu, bagian atas

menyediakan search bar dan opsi rentang waktu yang dapat diatur sesuai kebutuhan analisis. Dengan tampilan yang interaktif ini, *user* dapat melakukan pencarian mendetail terhadap ribuan baris log dalam waktu singkat tanpa perlu menulis query manual yang kompleks.



Gambar 3.8. *Index Pattern*

Sumber: Dokumen Pribadi [7]

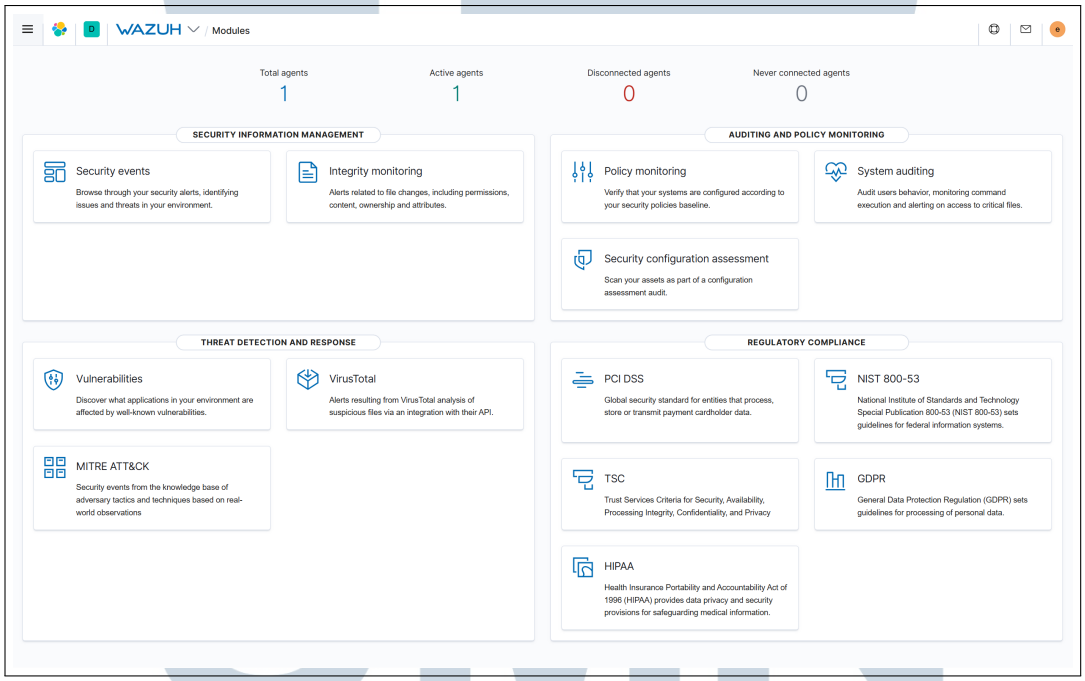
Saat dibuka, halaman Discover akan langsung menampilkan data log berdasarkan index pattern yang aktif, namun dapat dilakukan perpindahan ke index lain sesuai kebutuhan analisis, seperti yang ditunjukkan pada Gambar 3.8. Setiap log yang dikumpulkan oleh Wazuh akan tersimpan ke dalam index Elasticsearch dengan pola nama tertentu, misalnya wazuh-alerts-* atau wazuh-monitoring-*. Pemilihan index pattern ini memungkinkan sistem untuk menampilkan dataset yang relevan dengan kebutuhan analisis, misalnya hanya log dari agent tertentu atau dari kategori event tertentu. Pengaturan ini menjadi dasar penting dalam memastikan hasil pencarian tetap akurat dan efisien.

Pemilihan index pattern yang tepat juga berpengaruh terhadap performa sistem serta kecepatan pencarian data. Dalam implementasi ini, struktur index dikelola secara otomatis oleh Wazuh, sehingga pembaruan data log dari setiap agent akan langsung masuk ke index yang sesuai tanpa intervensi manual. Selain itu, dapat dilakukan pembuatan custom index pattern untuk memisahkan sumber data tertentu, seperti log dari server Windows, Linux, atau modul integrasi lain seperti VirusTotal dan Syscheck. Setelah memahami cara kerja Discover Page dan struktur index pattern ini, proses analisis dapat dilanjutkan melalui menu Wazuh yang menampilkan hasil deteksi dan aktivitas HIDS secara terstruktur dan mudah

dipahami.

A.4 Wazuh Menu

Menu Wazuh merupakan tampilan khusus yang muncul pada sidebar Kibana setelah proses instalasi dan integrasi plugin Wazuh berhasil dilakukan. Melalui menu ini, *user* dapat mengakses seluruh komponen dan fitur yang berkaitan dengan sistem HIDS yang telah terpasang. Menu ini juga menjadi bagian penting dari integrasi antara Wazuh dan Elastic Stack, karena berfungsi sebagai wadah yang menampilkan data hasil analisis dan deteksi ancaman yang dikumpulkan oleh Wazuh Manager dari berbagai agent yang terhubung.



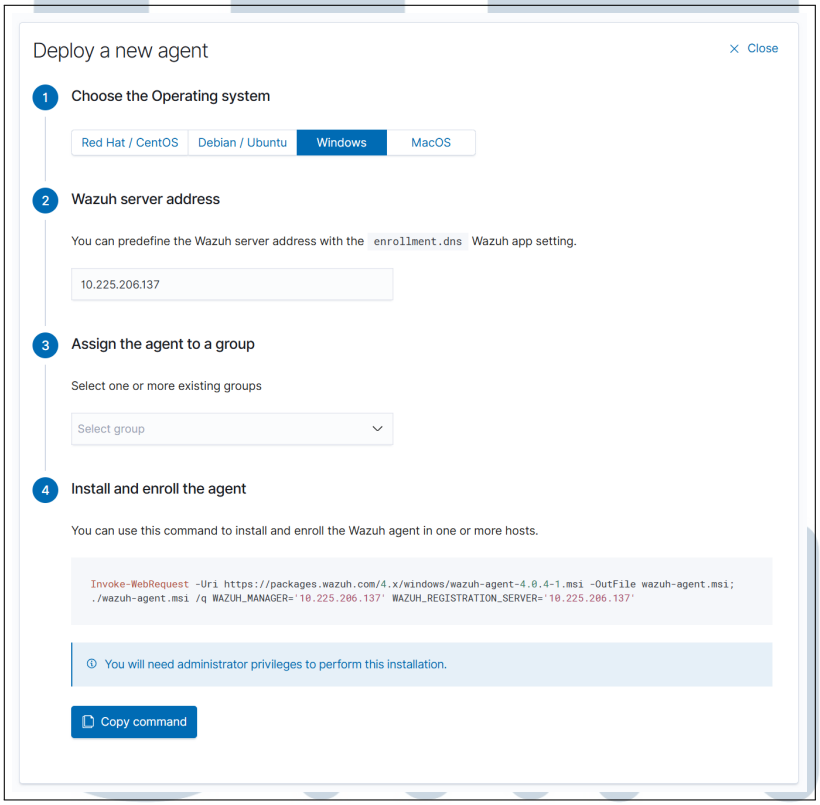
Gambar 3.9. Tampilan menu Wazuh

Sumber: Dokumen Pribadi [7]

Seperti terlihat pada Gambar 3.9, halaman utama Wazuh menampilkan ringkasan status dari seluruh agent yang telah terdaftar di sistem. Informasi yang disajikan meliputi jumlah agent yang connected dan disconnected, serta daftar fitur utama seperti Security Events, Integrity Monitoring, Agent Management, dan Vulnerability Detection. Tampilan ini memberikan gambaran umum mengenai kondisi terkini sistem HIDS, termasuk jumlah endpoint yang aktif dalam proses pemantauan. Selain itu, menu Wazuh juga menyediakan akses langsung ke berbagai modul pemantauan, memungkinkan *user* untuk berpindah antarfitur tanpa perlu

keluar dari tampilan utama Kibana.

Komunikasi antara komponen Wazuh Manager dan seluruh agent yang terhubung dilakukan melalui Wazuh API. API ini menggunakan protokol HTTPS dan mekanisme autentikasi berbasis token untuk menjamin keamanan pertukaran data. Fungsi utamanya mencakup pendaftaran agent baru, pengambilan data log, serta pengiriman *alert* dari agent ke manager. Dengan adanya Wazuh API, sistem dapat beroperasi secara terdistribusi namun tetap terpusat dalam manajemen data dan konfigurasi. Salah satu implementasi langsung dari API ini dapat dilihat pada fitur penambahan *agent* melalui tampilan Wazuh.



Gambar 3.10. Penambahan agent

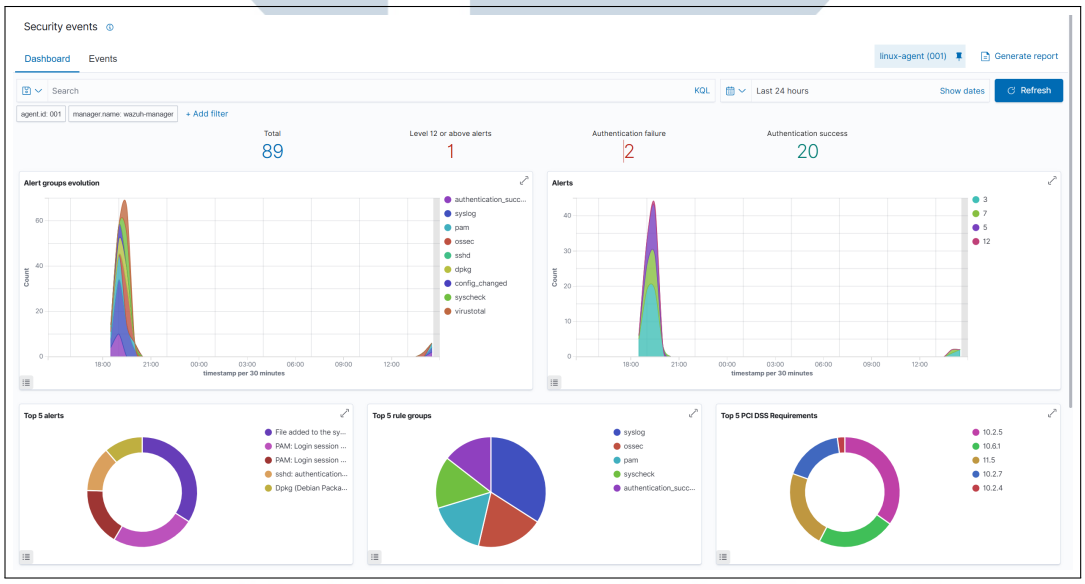
Sumber: Dokumen Pribadi [7]

Seperti yang ditunjukkan pada Gambar 3.10, halaman penambahan agent menampilkan perintah instalasi yang telah disesuaikan dengan sistem operasi yang digunakan, seperti Linux, Windows, atau macOS. Setiap perintah mencakup informasi penting seperti manager address, authentication key, serta instruksi konfigurasi agar agent dapat terhubung dengan benar ke Wazuh Manager. Mekanisme ini memperlihatkan efisiensi proses integrasi endpoint baru, karena *user* tidak perlu melakukan konfigurasi manual pada sisi server. Setelah proses

penambahan selesai, agent yang baru ditambahkan akan muncul pada daftar agent aktif dan mulai mengirimkan data log secara otomatis ke sistem untuk dianalisis melalui Security Events Dashboard yang dibahas pada bagian selanjutnya.

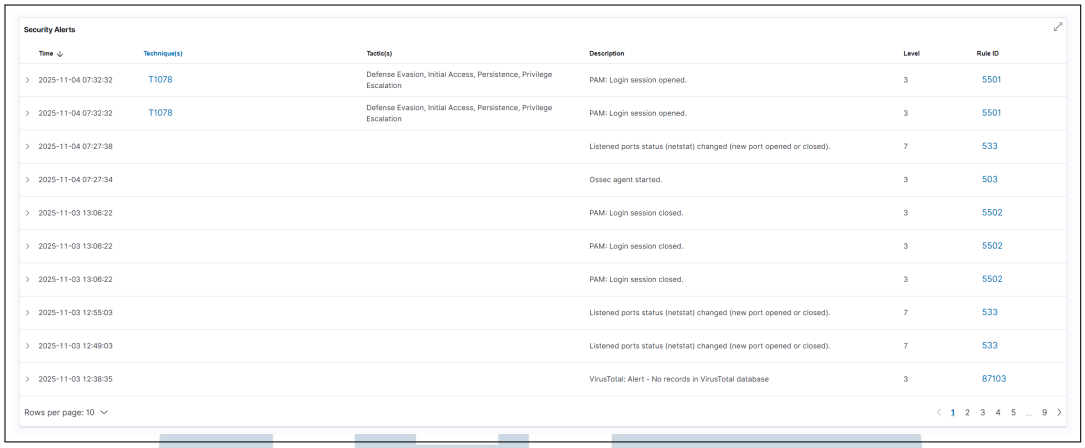
B *Security Events Monitoring*

Proses security events monitoring merupakan salah satu bagian penting dalam implementasi sistem Wazuh HIDS. Melalui fitur ini, seluruh aktivitas dan kejadian yang terjadi pada host dapat terpantau secara *real-time* menggunakan dashboard Wazuh yang terintegrasi dengan Kibana. Dashboard ini menampilkan berbagai informasi penting, seperti jenis event keamanan yang terdeteksi, tingkat risiko (*risk level*), serta sumber *host* atau agent yang memicu peristiwa tersebut. Dengan adanya sistem ini, *user* dapat memantau kondisi keamanan dari setiap agent secara terpusat dan cepat. Gambar 3.11 adalah tampilan dari *dashboard security events*.



Gambar 3.11. *Dashboard security events*
Sumber: Dokumen Pribadi [7]

Dashboard security events juga menyajikan tren aktivitas keamanan berdasarkan periode waktu tertentu. Melalui grafik tren ini, dapat diketahui pola peningkatan jumlah event pada jam atau hari tertentu, yang memberikan gambaran mengenai potensi lonjakan aktivitas mencurigakan. Fitur ini sangat membantu dalam analisis threat hunting, karena mempermudah identifikasi perilaku tidak normal yang mungkin menandakan serangan terkoordinasi.



Gambar 3.12. *Security alerts*

Sumber: Dokumen Pribadi [7]

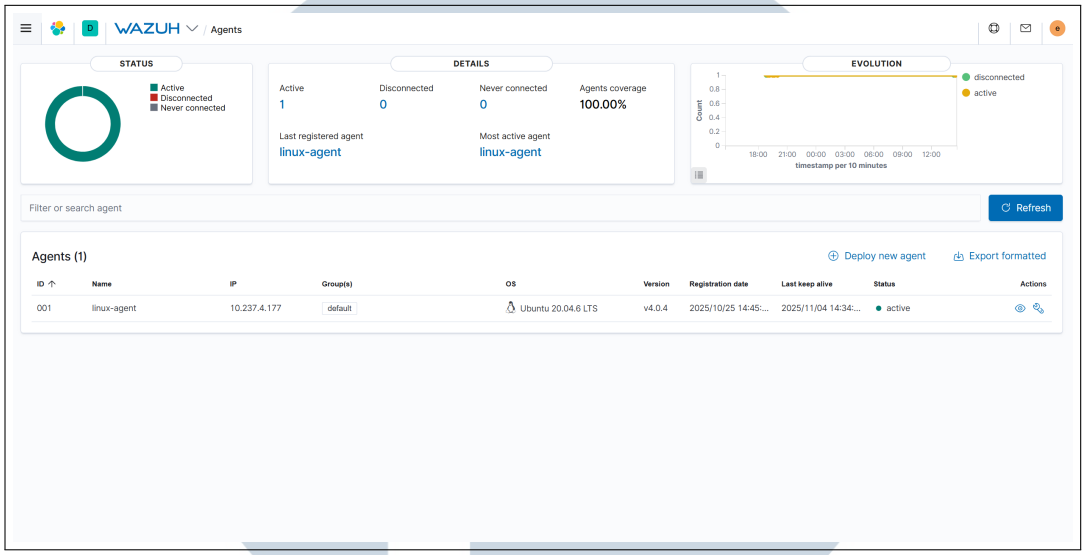
Gambar 3.12 adalah security alerts yang muncul dari event yang terjadi. Setiap event yang dikirim oleh Wazuh Agent akan diproses oleh Wazuh Manager, dikategorikan berdasarkan jenis ancaman, lalu ditampilkan dalam bentuk security alerts di dashboard. Contohnya, event dapat berupa aktivitas login yang gagal, deteksi perubahan file sistem, atau upaya akses tidak sah terhadap layanan tertentu. Masing-masing event memiliki nilai risk level yang dihitung berdasarkan *severity level* ancaman. Nilai ini membantu analis keamanan dalam menentukan prioritas penanganan insiden. Semakin tinggi nilai risk level, semakin besar potensi ancaman yang perlu segera diinvestigasi.

Secara keseluruhan, *dashbaord security events monitoring* berfungsi sebagai tulang punggung sistem deteksi intrusi berbasis host. Dengan kemampuannya mengumpulkan, menganalisis, dan memvisualisasikan data log dari berbagai sumber, Wazuh membantu menciptakan lingkungan simulasi SOC yang menyerupai kondisi operasional. Hasil pemantauan ini menjadi dasar dalam melakukan evaluasi keamanan, menyusun laporan bulanan, serta meningkatkan pemahaman terhadap proses korelasi data dan penanganan insiden pada tingkat host.

C *Agent Monitoring*

Agent monitoring pada Wazuh digunakan untuk memantau status konektivitas dan aktivitas dari setiap agent yang terhubung dengan Wazuh Manager. Melalui dashboard agent yang tersedia di tampilan Kibana, *user* dapat melihat daftar semua agent yang aktif maupun tidak aktif, lengkap dengan detail informasi seperti sistem operasi, alamat IP, versi agent, serta waktu terakhir agent melakukan

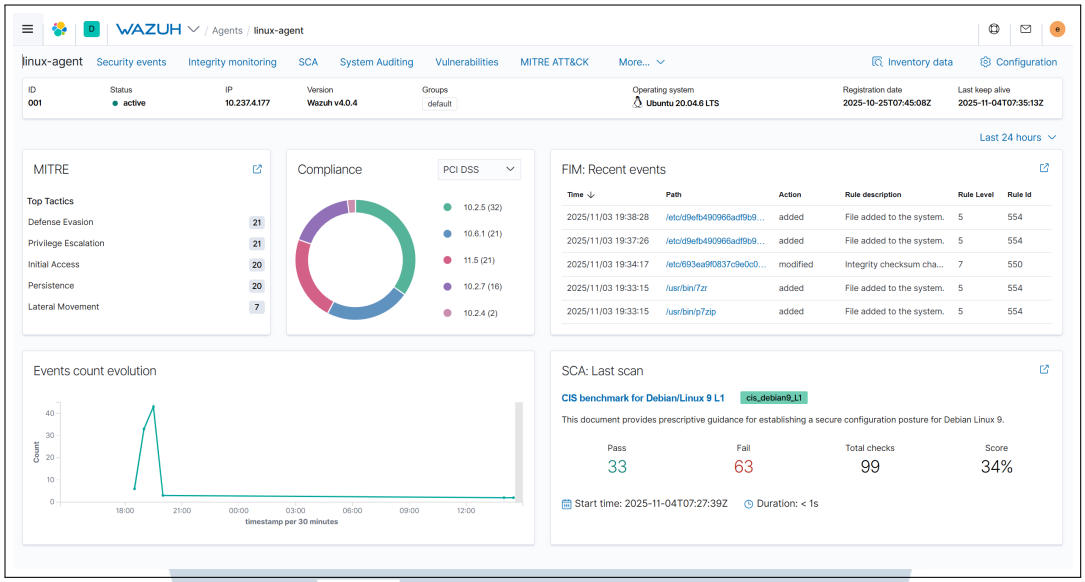
komunikasi dengan server. Fungsi ini memungkinkan proses pengawasan sistem dilakukan secara menyeluruh, terutama untuk memastikan bahwa seluruh host yang diawasi tetap berada dalam kondisi aktif dan terkendali.



Gambar 3.13. List agent yang terhubung

Sumber: Dokumen Pribadi [7]

Setiap agent berperan sebagai sensor keamanan yang memantau berbagai aktivitas di sisi host, seperti perubahan file, proses berjalan, dan percobaan akses tidak sah. Oleh karena itu, pemantauan status agent menjadi krusial agar data log yang dikirim ke Wazuh Manager tetap konsisten dan tidak terputus. Melalui dashboard ini, apabila sebuah agent tidak mengirimkan data dalam jangka waktu tertentu, sistem akan menampilkan status *disconnected*. Kondisi tersebut dapat menjadi indikator adanya gangguan pada konfigurasi jaringan, kesalahan autentikasi, atau permasalahan layanan agent pada host bersangkutan. Pada Gambar 3.13 ditampilkan terdapat satu agent yang terhubung dengan SIEM. Agent yang terhubunga memiliki operating system Ubuntu 20.04 yang berstatus connected, yang menunjukkan bahwa log dari agent tersebut diterima oleh wazuh manager.



Gambar 3.14. Detail agent

Sumber: Dokumen Pribadi [7]

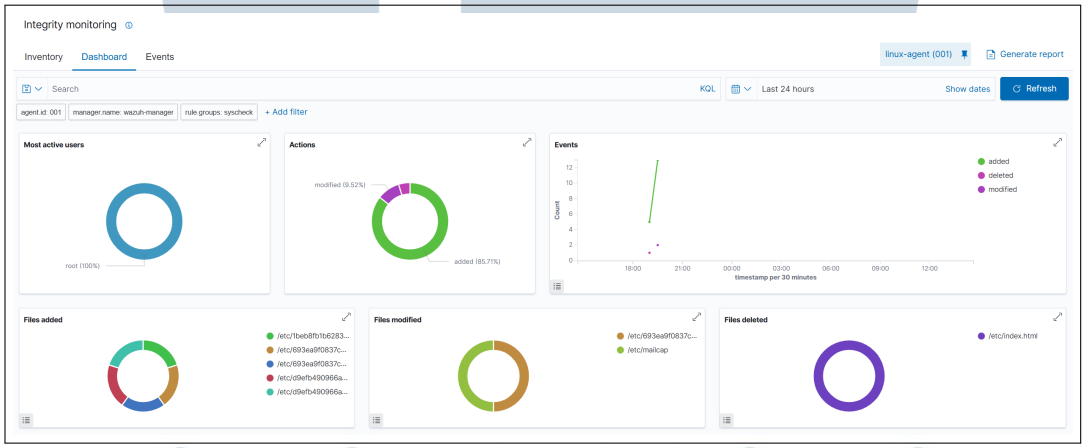
Pemantauan agent secara rutin berperan penting dalam menjaga reliabilitas sistem HIDS secara keseluruhan. Melalui integrasi antara Wazuh Manager dan Kibana, setiap perubahan status agent dapat segera terdeteksi dan ditindaklanjuti. Dengan demikian, proses deteksi intrusi dan analisis keamanan dapat berlangsung tanpa gangguan, serta memastikan bahwa setiap host yang terhubung tetap berfungsi optimal dalam memberikan data yang akurat untuk keperluan analisis keamanan.

D *Integrity Monitoring*

Fitur Integrity Monitoring merupakan salah satu komponen inti dalam sistem HIDS yang diimplementasikan menggunakan Wazuh. Tujuan utama dari dashboard ini adalah untuk mendeteksi setiap perubahan yang terjadi pada sistem berkas di tingkat host secara real-time. Proses pemantauan integritas file menjadi penting karena banyak serangan dilakukan dengan cara mengubah, menambahkan, atau menghapus file penting di sistem target, baik secara langsung maupun melalui eksploitasi proses aplikasi. Oleh sebab itu, pemantauan perubahan file secara berkelanjutan dapat membantu mengidentifikasi indikasi awal kompromi sistem (*system compromise*) bahkan sebelum dampak lebih besar terjadi.

D.1 Dashboard Integrity Monitoring

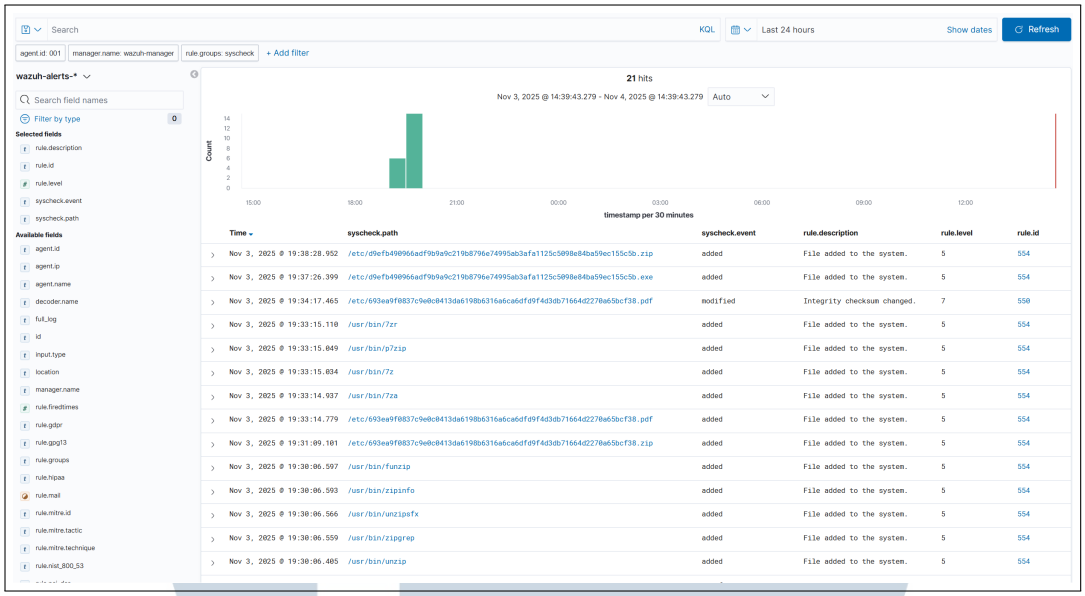
Integrity Monitoring dijalankan menggunakan komponen Wazuh, yaitu modul *Syscheck*. Modul ini berfungsi untuk mengaudit dan merekam setiap aktivitas yang terjadi pada sistem berkas *host*, termasuk operasi penambahan (*added*), penghapusan (*deleted*), dan perubahan isi file (*modified*). Wazuh Agent secara periodik melakukan pemindaian terhadap direktori dan file yang telah ditentukan dalam konfigurasi agent. Setiap perubahan yang terdeteksi akan menghasilkan event log yang dikirimkan ke Wazuh Manager, kemudian dikategorikan dan dikorelasikan sebelum akhirnya ditampilkan di dashboard Kibana.



Gambar 3.15. *Dashboard integrity monitoring*

Sumber: Dokumen Pribadi [7]

Salah satu keunggulan dari Integrity Monitoring pada Wazuh adalah kemampuannya menampilkan hasil pemantauan dalam bentuk visualisasi yang interaktif. Pada dashboard awal integritas sistem yang ditampilkan pada gambar 3.15, dapat dilihat beberapa metrik utama, seperti jumlah total file yang diubah (*modified*), file baru yang ditambahkan (*added*), serta file yang dihapus (*deleted*). Selain itu, tersedia juga grafik yang menampilkan jumlah *events* dan jenis *action* yang dilakukan terhadap file dalam jangka waktu tertentu. Tampilan ini membantu analis keamanan memahami pola perubahan sistem dan membedakan aktivitas normal dari aktivitas yang mencurigakan. Misalnya, lonjakan signifikan pada jumlah file yang dimodifikasi dalam waktu singkat dapat menjadi indikasi terjadinya *malware infection* atau *unauthorized script injection*.



Gambar 3.16. *Discover* dari *integrity monitoring*

Sumber: Dokumen Pribadi [7]

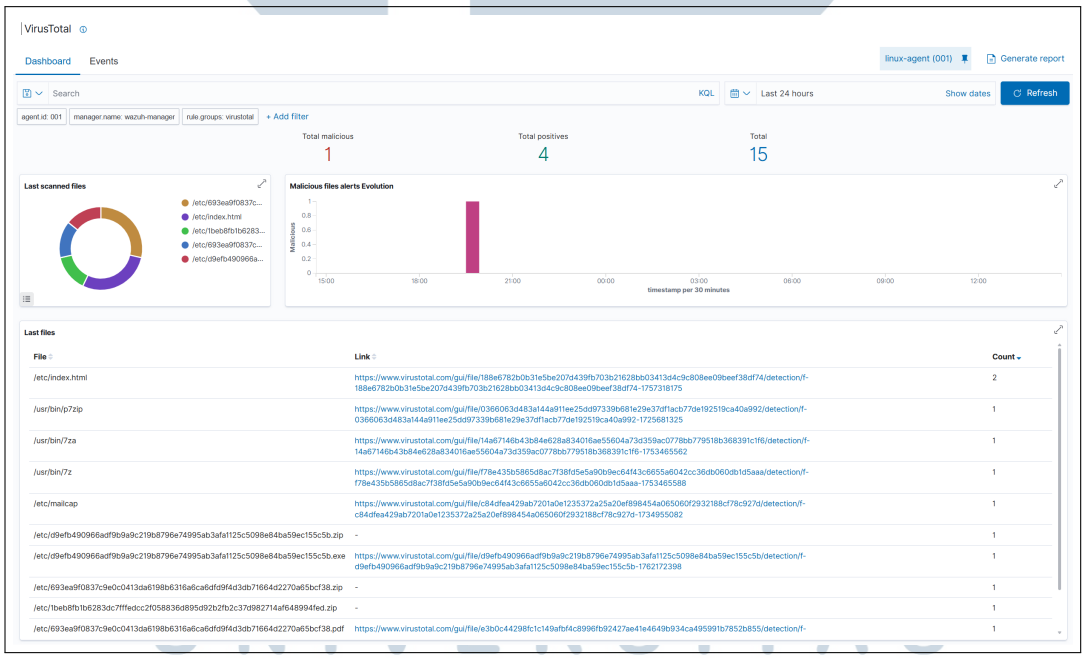
Pada halaman discover di Kibana yang ditampilkan pada gambar 3.16, data hasil pemantauan integritas dapat dianalisis lebih lanjut secara mendetail. Melalui fitur pencarian dan filter, analis dapat memfokuskan pencarian pada event tertentu seperti syscheck activity atau menelusuri file yang sering mengalami perubahan. Informasi yang ditampilkan meliputi nama file, path lokasi, hash MD5 dan SHA1, timestamp, serta status tindakan yang dilakukan terhadap file tersebut. Kombinasi antara data hash dan waktu kejadian memungkinkan analis melakukan audit forensik dengan lebih akurat, karena setiap perubahan yang tidak sah dapat dibandingkan dengan nilai hash sebelumnya untuk memastikan keaslian file. Dalam praktik keamanan informasi, proses ini disebut sebagai file baseline verification — yaitu membandingkan kondisi file terkini dengan kondisi file acuan yang dianggap aman.

Dari hasil implementasi yang dilakukan, *Integrity Monitoring* terbukti efektif dalam memberikan notifikasi terhadap perubahan yang terjadi pada file sistem yang sedang diawasi. Ketika agent mendeteksi adanya file baru yang ditambahkan atau diubah, sistem segera mengirimkan alert ke Wazuh Manager dengan informasi detail mengenai sumber host, jenis perubahan, dan nilai hash file. Mekanisme ini sangat penting untuk mencegah terjadinya persistence attack, yaitu upaya penyerang untuk mempertahankan akses pada sistem dengan cara menyembunyikan file berbahaya di dalam host yang terkompromi. Dengan sistem audit yang berkelanjutan, setiap perubahan mencurigakan dapat segera

diidentifikasi sebelum berkembang menjadi ancaman yang lebih serius.

D.2 Integrasi VirusTotal

Untuk memperluas kemampuan deteksi ancaman, sistem Wazuh mendukung integrasi dengan layanan threat intelligence eksternal, salah satunya adalah VirusTotal. Integrasi ini dirancang untuk menambah konteks keamanan terhadap file yang terdeteksi oleh modul *Integrity Monitoring*. Setiap kali sistem mendeteksi perubahan atau penambahan file baru, Wazuh dapat secara otomatis mengirimkan hash file tersebut ke VirusTotal API untuk dilakukan pemeriksaan reputasi. Hasil analisis dari VirusTotal mencakup informasi apakah file tersebut pernah diidentifikasi sebagai berbahaya (*malicious*), serta jumlah mesin antivirus yang mengonfirmasi deteksi tersebut. Dengan mekanisme ini, proses threat validation dapat dilakukan secara otomatis tanpa perlu analisis manual yang memakan waktu lama.

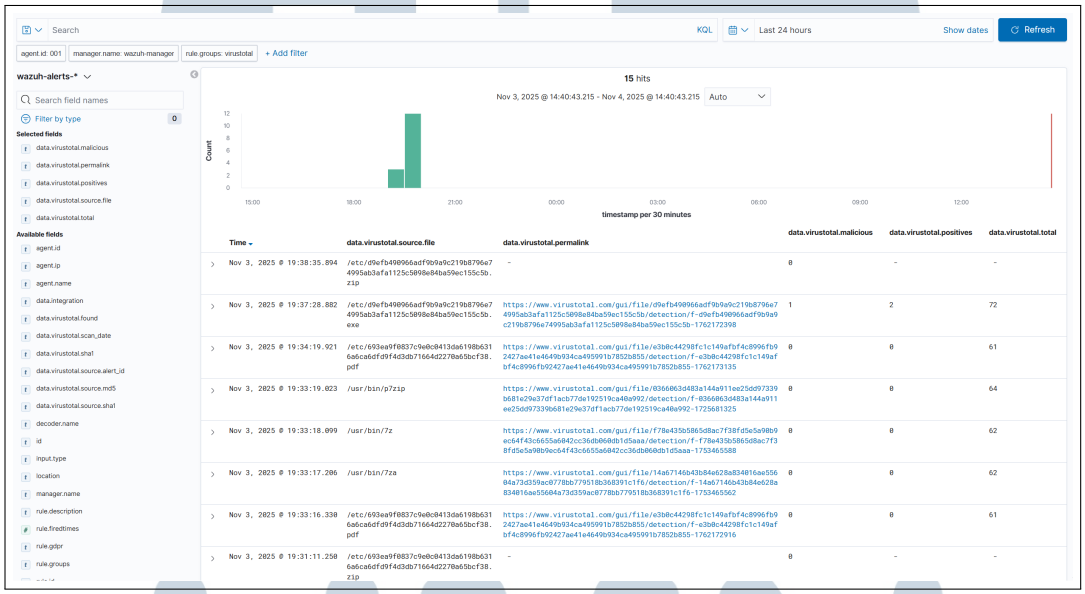


Gambar 3.17. *Dashboard VirusTotal*

Sumber: Dokumen Pribadi [7]

Gambar 3.17 menunjukkan tampilan dashboard dari pengecekan VirusTotal terhadap file yang terdapat pada agent. Dashboard akan menampilkan list dari file yang terdeteksi pada direktori yang masuk ke dalam list monitoring. Dalam list file, akan dicantumkan nama file, link hasil pengecekan VirusTotal, dan jumlah hasil vendor yang memberikan file dengan hash tersebut tag sebagai malicious.

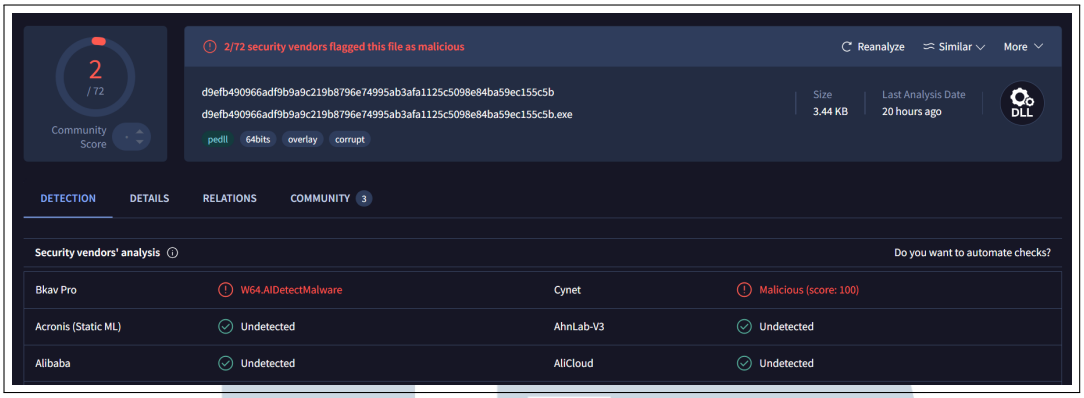
Integrasi dengan VirusTotal juga menghasilkan tampilan khusus pada dashboard Kibana yang menampilkan visualisasi hasil pemeriksaan reputasi file. Beberapa elemen visual utama meliputi daftar file terakhir yang dipindai (last scanned files), grafik evolusi *malicious files alert*, serta metrik statistik terkait tingkat deteksi file berbahaya dari seluruh agent yang terhubung. Tampilan ini memberikan gambaran menyeluruh kepada analis tentang tren kemunculan file berisiko di seluruh lingkungan pemantauan, serta memungkinkan identifikasi dini terhadap host yang paling sering menghasilkan file mencurigakan.



Gambar 3.18. *Discover* dari *VirusTotal events*

Sumber: Dokumen Pribadi [7]

Selain itu, tampilan discover juga mengalami penambahan field baru yang berasal dari hasil integrasi dengan VirusTotal. Gambar 3.18 menunjukkan field tambahan tersebut mencakup hasil pemeriksaan (scan result), tingkat kepercayaan deteksi (detection ratio), dan permalink yang mengarahkan langsung ke halaman hasil analisis di situs resmi VirusTotal. Dengan adanya informasi ini, analis dapat dengan cepat memverifikasi status keamanan file tertentu tanpa perlu berpindah platform. Apabila hasil pemeriksaan menunjukkan bahwa file memiliki reputasi malicious dari lebih dari satu mesin antivirus, maka Wazuh secara otomatis meningkatkan *severity level* alert yang dihasilkan ke level high atau critical, menandakan bahwa file tersebut perlu ditindaklanjuti segera. Permalink yang diberikan juga dapat digunakan sebagai bukti dalam analisis.



Gambar 3.19. Hasil pengecekan pada VirusTotal

Sumber: Dokumen Pribadi [7]

Gambar 3.19 adalah contoh dari tampilan hasil pengecekan pada website VirusTotal. Pada hasil pengecekan yang ditunjukan, hash yang dimiliki oleh file dianggap sebagai *malicious* oleh 2 vendor. Sebanyak dua tag malicious diberikan untuk hash tersebut, di mana tag tersebut diberikan oleh vendor Bkav Pro dan Cynet.

Integrasi Wazuh dengan VirusTotal tidak hanya memperkuat kemampuan deteksi sistem, tetapi juga meningkatkan efisiensi proses investigasi insiden. Dalam skenario nyata di SOC, fitur ini dapat membantu *security analyst* memprioritaskan penanganan insiden berdasarkan tingkat risiko aktual, bukan sekadar dari banyaknya event yang terjadi. Oleh karena itu, kolaborasi antara Integrity Monitoring dan VirusTotal Integration menjadikan sistem HIDS yang diimplementasikan lebih adaptif dan cerdas dalam mendeteksi serta mengonfirmasi keberadaan ancaman keamanan pada tingkat host.

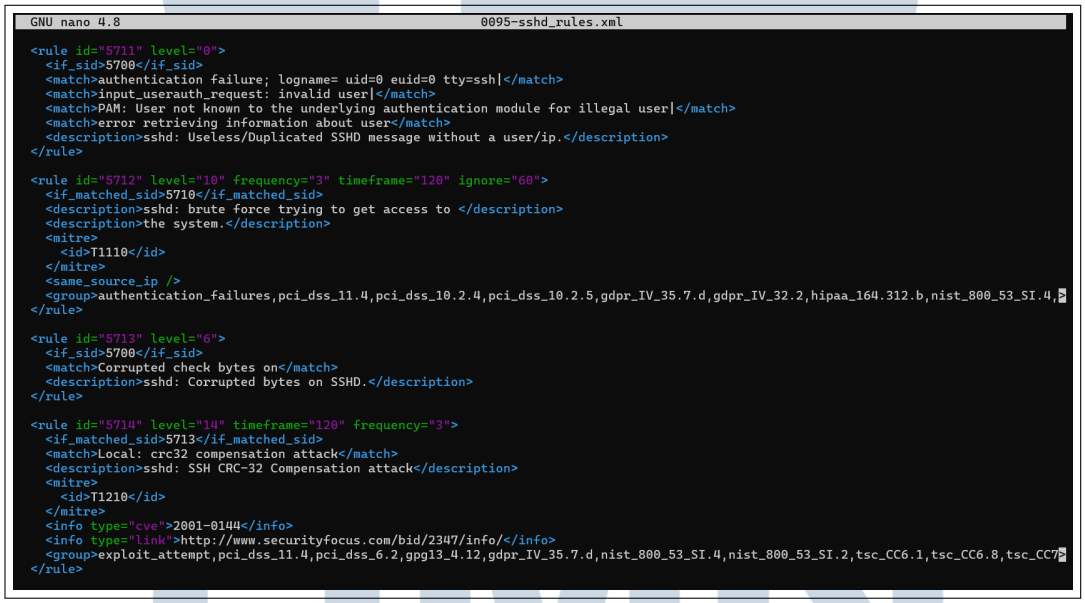
Gambar 3.10 adalah event-event yang terjadi terkait dengan suspicious file pada directory di wazuh agent. Berdasarkan event yang terjadi, dapat dilihat bahwa ditemukan event dengan file yang memiliki tag malicious dari vendor yagn terdaftar pada VirusTotal.

E *Security Alerts*

Pada sistem HIDS yang diimplementasikan, *security alerts* merupakan hasil dari proses analisis terhadap berbagai aktivitas sistem yang terdeteksi oleh agent. Setiap aktivitas atau kejadian yang memenuhi pola tertentu berdasarkan aturan (rule) yang telah ditetapkan akan menghasilkan alert dengan *severity level* tertentu. Fitur ini memungkinkan sistem untuk secara otomatis mengenali potensi ancaman, mengklasifikasikannya berdasarkan tingkat risiko, serta memberikan respons yang sesuai.

E.1 *Alerts Rule*

Wazuh menggunakan sistem rule-based detection yang berbasis pada file konfigurasi berformat XML, terletak di direktori ruleset pada Wazuh Manager. Setiap rule didefinisikan dengan parameter tertentu seperti id, level, frequency, dan timeframe, yang bersama-sama menentukan kondisi kapan suatu event dianggap sebagai ancaman dan layak menghasilkan alert. Struktur ruleset ini memungkinkan sistem mendeteksi berbagai kategori serangan, mulai dari aktivitas login mencurigakan, perubahan konfigurasi sistem, hingga serangan *brute force* dan eksploitasi jaringan. Gambar 3.11 memperlihatkan contoh isi file XML rules pada direktori Wazuh Manager.



Gambar 3.20. Contoh konfigurasi *alert rules*

Sumber: Dokumen Pribadi [7]

Salah satu contoh rule yang digunakan dalam konfigurasi adalah pada ID 5712. Rule tersebut berfungsi untuk mendeteksi serangan *brute force* terhadap layanan SSH (sshd). Parameter `frequency="3"` dan `timeframe="120"` berarti bahwa apabila terdapat tiga kali kegagalan autentikasi dalam waktu dua menit, sistem akan menghasilkan alert dengan severity level=”10”.Selain itu, elemen `<same_source_ip />` memastikan bahwa deteksi dilakukan berdasarkan alamat IP yang sama, sehingga hanya aktivitas berulang dari sumber identik yang akan dianggap berisiko. Rule ini juga terhubung dengan standar keamanan internasional seperti MITRE ATT&CK (T1110) yang mengategorikan serangan *brute force* dalam fase Credential Access.

Ketika kondisi tersebut terpenuhi, Wazuh Manager akan menghasilkan alert yang dikirimkan ke Elastic Stack untuk kemudian divisualisasikan melalui Kibana Dashboard. Informasi yang ditampilkan meliputi nama rule, *severity level*, sumber host, alamat IP penyerang, serta timestamp kejadian. Dengan mekanisme ini, analis keamanan dapat segera mengidentifikasi adanya upaya login tidak sah dan menentukan langkah mitigasi yang diperlukan.

E.2 *Active Response*

Selain mendeteksi serangan, Wazuh juga memiliki kemampuan untuk memberikan tanggapan otomatis (*active response*) terhadap alert tertentu. Fitur ini memungkinkan sistem tidak hanya bersifat reaktif, tetapi juga melakukan tindakan pencegahan secara langsung terhadap ancaman yang sedang berlangsung. Pengaturan *active response* dilakukan di dalam berkas konfigurasi Wazuh Manager, pada bagian `<active-response>` yang menentukan jenis perintah, lokasi eksekusi, agent target, serta durasi tindakan.

```
281   <!-- Active Responses  -->
282   <!-- 5712 - sshd: brute force trying to get access to the system. -->
283   <active-response>
284     <command>firewall-drop</command>
285     <location>defied-agent</location>
286     <agent_id>001</agent_id>
287     <rules_id>5712</rules_id>
288     <timeout>60</timeout>
289     <repeated_offenders>30,60,120</repeated_offenders>
290   </active-response>
```

Gambar 3.21. Konfirgurasi *active-response*

Sumber: Dokumen Pribadi [7]

Gambar 3.21 memperlihatkan bagian konfigurasi *active response* pada file manager configuration.. Konfigurasi tersebut menunjukkan bahwa ketika *rule ID* 5712 (*brute force SSH*) aktif, sistem akan menjalankan perintah `firewall-drop`, yaitu instruksi untuk memblokir alamat IP penyerang melalui aturan *firewall* pada host target. Parameter `<timeout>60</timeout>` menandakan bahwa pemblokiran akan berlangsung selama 60 detik sebelum aturan *firewall* otomatis dilepas, sedangkan `<repeated_offenders>` menentukan peningkatan durasi blokir jika sumber yang sama melakukan pelanggaran berulang kali.

Fitur active response dalam konteks simulasi SOC merepresentasikan kemampuan otomatisasi dalam proses incident response. Dalam skenario nyata, fungsi ini membantu tim SOC mengurangi waktu reaksi terhadap serangan,

mencegah eskalasi insiden, dan menjaga stabilitas sistem dari upaya intrusi berulang. Integrasi antara *alert rule* dan *active response* menjadikan sistem Wazuh tidak hanya mendeteksi ancaman, tetapi juga mampu mengambil langkah proaktif dalam mempertahankan keamanan jaringan.

3.3.3 Hasil Pengujian Prototipe HIDS

Pengujian dilakukan untuk memastikan bahwa seluruh komponen prototipe HIDS berfungsi sesuai tujuan perancangannya, terutama dalam mendeteksi aktivitas host dan menampilkan hasilnya pada SIEM. Pengujian difokuskan pada empat aspek utama: konektivitas agent, pemicu rule alert, deteksi perubahan file melalui integrity monitoring, serta verifikasi reputasi file melalui integrasi VirusTotal. Setiap hasil pengujian disajikan dengan merujuk pada tampilan visual dan event yang telah dihasilkan pada tahap implementasi sebelumnya, sehingga memperlihatkan konsistensi antara konfigurasi dan output yang muncul pada dashboard.

A Konektivitas Agent

Pengujian konektivitas dilakukan untuk memastikan bahwa Wazuh Agent dapat terhubung dan mengirimkan data log ke Wazuh Manager secara stabil. Berdasarkan hasil implementasi, agent yang terpasang pada host uji berhasil muncul pada dashboard Agent Monitoring dengan status Active, sebagaimana terlihat pada Gambar 3.13, menandakan bahwa proses pendaftaran agent, verifikasi ID, dan koneksi TLS telah berfungsi sebagaimana mestinya. Konektivitas ini juga divalidasi melalui munculnya data log dari agent pada halaman Discover, yang menandakan bahwa pipeline pengiriman log telah berjalan end-to-end. Dengan demikian, seluruh alur komunikasi dasar antar komponen berhasil diuji dan bekerja sesuai topologi yang dirancang.

B *Rule Alert*

Pengujian rule alert bertujuan untuk memverifikasi bahwa Wazuh mampu menghasilkan alert berdasarkan aturan deteksi yang telah didefinisikan pada ruleset. Hasil pengujian menunjukkan bahwa ketika aktivitas tertentu dipicu pada host, Wazuh berhasil mengidentifikasi event tersebut dan mengonversinya menjadi alert yang tampil pada halaman Security Events. Alert yang dihasilkan menampilkan

informasi lengkap, seperti level severity, rule ID, deskripsi rule, serta korelasi dengan MITRE ATT&CK. Output yang dihasilkan tampak pada Gambar 3.12, yang menunjukkan bahwa sistem mampu menampilkan alert sesuai severity dan rule yang berlaku, dan pengiriman event ke Elasticsearch berfungsi secara konsisten, sesuai dengan yang ditampilkan pada bagian implementasi sebelumnya.

C *Integrity*

Pengujian fitur Integrity Monitoring dilakukan melalui simulasi perubahan struktur file pada host, seperti penambahan, penghapusan, dan modifikasi file. Hasil pengujian menunjukkan bahwa setiap perubahan berhasil terdeteksi oleh modul Syscheck dan divisualisasikan melalui dashboard Integrity Monitoring pada Wazuh. Event yang tercatat menampilkan detail seperti jenis aksi (*added, modified, deleted*), path file, dan hash file sebelum dan sesudah perubahan. Hasilnya terlihat pada Gambar 3.16, di mana perubahan file berhasil terdeteksi dan dicatat oleh modul *integrity monitoring*. Hasil ini menunjukkan bahwa proses pemindaian file, pembuatan hash, dan pengiriman data ke manager berfungsi dengan baik.

D *VirusTotal Integration*

Integrasi VirusTotal diuji dengan memonitor file yang di-scan oleh modul integrity, dan hasilnya menunjukkan bahwa hash file berhasil dikirim ke VirusTotal untuk pengecekan reputasi. Dashboard integrasi menampilkan daftar file terbaru yang dianalisis, beserta indikator reputasi file seperti malicious atau clean. Hasilnya diperlihatkan pada Gambar 3.18, yang menunjukkan bahwa sistem dapat mengirim hash file ke layanan VirusTotal dan menerima respons berupa informasi reputasi file tersebut. Hasil ini mengonfirmasi bahwa integrasi API berjalan dengan baik dan sistem mampu melakukan threat intelligence enrichment terhadap data yang diterima dari agent.

3.4 Kendala dan Solusi yang Ditemukan

Selama pelaksanaan kegiatan magang, terdapat sejumlah tantangan yang muncul dalam proses implementasi dan operasional sistem. Beberapa kendala yang ditemui selama kegiatan kerja magang antara lain sebagai berikut:

1. Kendala dalam proses instalasi dan integrasi antara Wazuh Manager, Filebeat,

dan Elasticsearch. Beberapa konfigurasi port dan parameter komunikasi menyebabkan koneksi antarkomponen gagal serta log tidak terdeteksi di dashboard Kibana.

2. Terbatasnya kuota permintaan (*request quota*) pada integrasi VirusTotal API menyebabkan beberapa hasil analisis reputasi file tidak muncul atau gagal diproses di dashboard.

Untuk mengatasi berbagai kendala yang muncul selama pelaksanaan magang, dilakukan sejumlah langkah penyelesaian sebagai berikut:

1. Melakukan penyesuaian pada konfigurasi dan melakukan optimasi alokasi *resource* pada *virtual machine* untuk memastikan sistem berjalan stabil dan data dapat terintegrasi secara *real-time*.
2. Melakukan pembatasan permintaan harian dan mengatur interval pengiriman permintaan agar sistem tidak melebihi batas kuota.