

**Tugas EC 7010  
Keamanan Sistem Lanjut**

**Host-Based Intrusion Detection Systems  
(HBIDS)**

Oleh:  
I Ketut Suparsa  
NIM: 23203110



**PROGRAM MAGISTER TEKNIK ELEKTRO  
BIDANG KHUSUS TEKNOLOGI INFORMASI DIKMENJUR  
INSTITUT TEKNOLOGI BANDUNG  
2004**

## **KATA PENGANTAR**

Dengan mengucapkan puji syukur kehadapan Tuhan Yang Maha Esa, penulis berhasil menyelesaikan laporan tugas akhir mata kuliah Sistem Keamanan Lanjut ini dengan baik.

Pada kesempatan ini penulis ingin mengucapkan terima kasih dan penghargaan kepada :

- Bapak Ir . Budi Raharjo, Phd., yang telah memberikan tugas, sehingga penulis dapat belajar tentang sistem keamanan pada jaringan komputer.
- Teman-teman seangkatan yang telah memberikan masukan terhadap penyelesaian laporan ini.

Tentunya laporan ini sangat jauh dari sempurna sebagai suatu karya tulis, untuk itu besar harapan penulis agar segala kekurangan tulisan ini dapat dilengkapi dan diperbaiki untuk dapat digunakan oleh yang memerlukan

Penyusun

## DAFTAR ISI

<b>KATA PENGANTAR.....</b>	<b>II</b>
<b>DAFTAR ISI .....</b>	<b>III</b>
<b>I. PENDAHULUAN.....</b>	<b>1</b>
1.1    LATAR BELAKANG .....	1
1.2    TUJUAN .....	1
1.3    PEMBATASAN MASALAH .....	2
<b>II. TINJAUAN TENTANG HOST BASED INTRUSION DETECTION SYSTEMS.....</b>	<b>2</b>
2.1    PERKEMBANGAN HOST-BASED INTRUSION DETECTION SYSTEMS.....	2
2.2    PENERAPAN SECURE COPROCESSOR PADA HBIDS .....	3
2.3    VULNERABILITY SECURITY SCANNER .....	5
2.4    KEMAMPUAN TASK YANG DIBEBANKAN PADA HOST-BASED IDS .....	6
2.5    STRUKTUR DAN ARSITEKTUR PADA HOST-BASED IDS.....	8
2.6    KLASIFIKASI DARI HOST-BASED IDS .....	9
2.7    KEUNGGULAN-KEUNGGULAN DARI HOST-BASED INTRUSION DETECTION SYSTEMS.....	11
2.8    SERANGAN–SERANGAN PADA HOST BASED IDS .....	13
2.8.1 <i>Taksonomi Serangan Dan Intrusion</i> .....	13
2.8.2 <i>Anomaly versus Signature Detection</i> .....	15
<b>III. PERTIMBANGAN-PERTIMBANGAN IMPLEMENTASI HBIDS .....</b>	<b>17</b>
3.1    KEBIJAKAN PERUSAHAAN TENTANG KEAMANAN SISTEM .....	17
3.2    PERLUNYA IMPLEMENTASI HOST-BASED IDS .....	18
<b>IV. PENUTUP .....</b>	<b>18</b>
<b>REFERENSI .....</b>	<b>19</b>

## I. PENDAHULUAN

### 1.1 Latar Belakang

Era komunikasi dengan menggunakan fasilitas internet memberikan banyak kemudahan dalam mendapatkan informasi yang dikehendaki. Dengan demikian semakin banyak orang, perusahaan-perusahaan, institusi pendidikan maupun instansi pemerintah yang menghubungkan jaringan komputernya ke jaringan layanan internet.

Pada awal penggunaan layanan internet ini belum banyak dilakukan transaksi-transaksi yang bersifat rahasia dan bernilai penting, tujuannya hanya menampilkan bahwa perusahaannya telah ada informasinya di internet. Namun demikian pada perkembangan berikutnya internet difungsikan sebagai sarana untuk melakukan transaksi yang mengandung informasi yang lebih kompleks diantaranya : e-commerce, e-government, e-learning dan e-bussiness. Tentunya pada informasi seperti ini masalah keamanan menjadi hal yang sangat penting diperhatikan, supaya informasi yang disampaikan akan dapat diterima oleh pihak yang berhak saja.

Dengan semakin banyaknya manfaat dan benefit yang diperoleh melalui penggunaan jaringan internet, maka semakin banyak pula pemakai komputer yang mengubungkan komputernya dengan internet saat ini, tentunya masalah keamanan menjadi semakin rumit dalam penanganannya, sehingga sistem keamanan ini seharusnya menjadi pertimbangan untuk sebuah perusahaan ataupun institusi yang akan menggunakan internet sebagai media koneksinya. Masalah yang sering terjadi pada akses informasi ini, yaitu pihak yang tidak berhak mendapatkan akses, dapat melakukan transaksi secara penuh dengan mengelabui sistem database pada komputer servernya. Mereka ini sering disebut dengan *hackers*, *cracker*, *carder*. Hal ini tentunya dapat terjadi karena adanya kelemahan-kelemahan pada sistem yang digunakan dan serangan dari dalam (*disgruntled employee*) diantaranya berupa : kelemahan aplikasi-aplikasi yang digunakan, banyaknya *debug* pada software sistem operasi dan penyalahgunaan *username* dan *password* oleh user sendiri. Untuk menghindari serangan dan melindungi server dari penyusup ini, maka diperlukan alat deteksi yang dapat mendeteksi terjadinya intrusi pada sistem kita. Alat deteksi ini disebut : *Intrusion detection systems (IDS)*.

### 1.2 Tujuan

Pada saat ini ada beberapa *Intrusion Detection Systems (IDS)* yang umum digunakan pada jaringan. Adapun tujuan dari *tools* ini diantaranya :

- a. mengawasi dan mencegah jika terjadi penetrasi kedalam sistem,
- b. mengawasi traffic yang terjadi pada jaringan,
- c. mendeteksi anomali, terjadinya penyimpangan dari sistem yang normal atau tingkah laku user;
- d. mendeteksi signature, membedakan pola antara *signature user* dengan *attacker*.

### 1.3 Pembatasan Masalah

Pada IDS yang yang disajikan akan ditekankan pembahasannya pada sistem deteksi intrusi berbasis *host* (*Host-Based Intrusion Detection Systems*)-HBIDS yang meliputi:

- a. pembahasan sains tentang HBIDS,
- b. keunggulan-keunggulan HBIDS,
- c. serangan-serangan pada HBIDS,
- d. Pertimbangan-pertimbangan penerapan HBIDS.

## II. TINJAUAN TENTANG HOST BASED INTRUSION DETECTION SYSTEMS

### 2.1 Perkembangan Host-Based Intrusion Detection Systems

*Host based intrusion detection* dimulai pada awal tahun 1980-an sebelum jaringan dikenal secara umum, dan tidak sekompleks dan terkoneksi secara luas seperti saat ini. Pada situasi yang masih sederhana itu umumnya digunakan *audit log* untuk melihat kembali aktivitas yang mencurigakan. Penyusupan dirasakan masih jarang sampai ditemukan fakta dari analisis yang membuktikan perlunya pencegahan terhadap serangan di masa mendatang.

Saat ini HBIDS tetap sebagai *tool* yang mampu memahami serangan-serangan yang terjadi sebelumnya dan menentukan metoda yang sesuai untuk mengatasi jika mereka melakukan serangan lagi. HBIDS masih menggunakan *audit logs*, tetapi menjadi lebih otomatis, dengan peningkatan teknik deteksi yang lebih responsif dan lebih canggih. HBIDS khususnya memonitor sistem, kejadian-kejadian, dan *log* keamanan pada windows NT dan *syslog* pada lingkungan UNIX. Bilamana beberapa dari *file* berubah, maka IDS membandingkan masuknya *log* yang baru dengan *attack signatures* jika terjadi kecocokan pada tanda tangan, maka sistem akan menanggapi dengan memberikan tanda bahaya kepada administrator dan panggilan yang lain untuk memberikan tindakan segera.

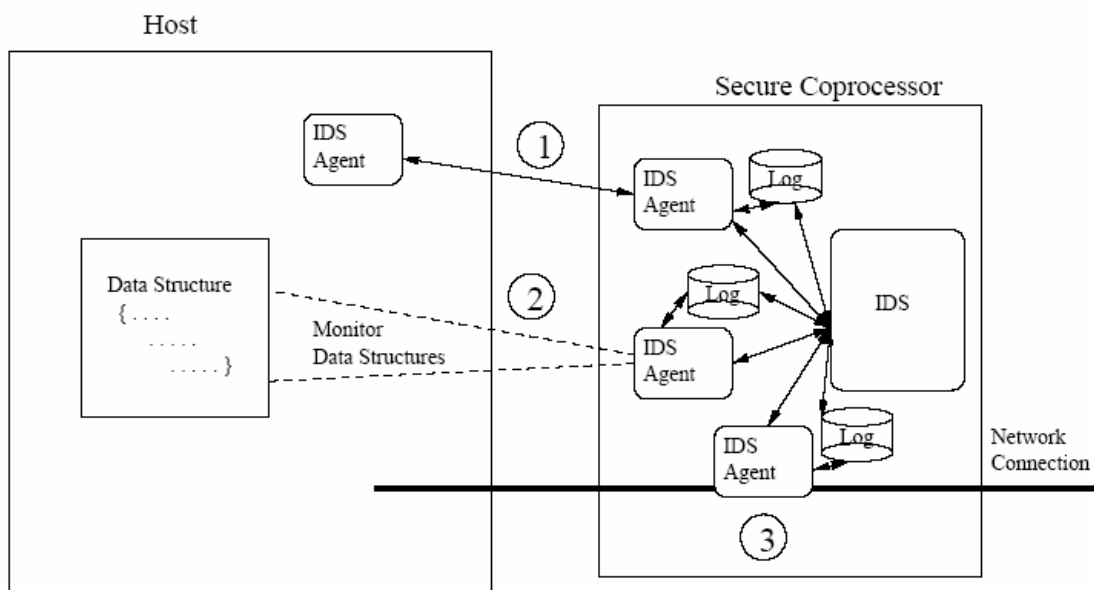
HBIDS telah tumbuh dan memasukkan unsur-unsur teknologi yang lain. Salah satu metoda yang populer untuk deteksi penyusupan dengan pemeriksaan *key system file* dan *executables file* dengan menguji beberapa bit yang ditransfer untuk mengetahui terjadinya *error* pada interval tertentu, untuk mengetahui perubahan yang tidak terduga. Ketepatan dari respon adalah berhubungan langsung kepada frekwensi dari interval pengujian. Terakhir hasil pengujian dari mendengar aktivitas dari *port* dan memberi tanda bahaya kepada administrator jika *port-port* yang khusus diakses. Type deteksi ini membawa unsur dari level mendasar dari teknologi *network-based intrusion detection* kedalam lingkungan *host-based*.

HBIDS menggunakan informasi yang dihasilkan oleh *host* untuk mendeteksi penyalahgunaannya. Sumber informasi berbeda juga dapat digunakan, seperti *event log* dari sistem dan aplikasi, statistik penggunaan *account*, akses data genting, dan modifikasinya.

## 2.2 Penerapan Secure Coprocessor pada HBIDS

HBIDS umumnya berbasis *agent*, terdiri dari file yang *executable* yang berjalan pada *host* dan berkomunikasi dengan beberapa sentral sistem kendali[2]. Data yang belum diolah dikumpulkan dan dikirim ke sentral sistem kontrol untuk dianalisis. Jika analisis mengindikasikan adanya potensi serangan, maka sentral sistem kendali akan menjadi yang pertama mengeluarkan rekomendasi respon atau membetulkan. Data tadi juga dapat dianalisis secara lokal oleh *agent* dengan mengirim status dan memberi tanda bahaya ke sentral sistem kendali sebagai pencocokan, dengan cara ini dapat mengurangi beban pada sentral sistem kendali dan dapat mendekati hasil yang *real time* atau tepat waktu pada deteksi dan responnya. Kombinasi dari pendekatan ini sering digunakan. Gambar 2.1 mengilustrasikan :

- 1) *Host-based* IDS dengan posisi *agent* pada *host*, dalam *secure coprocessor* atau keduanya.
- 2) Penyederhanaan sistem pemantau host.
- 3) *Host-based network* IDS



Gambar 2.1 *Host Based* IDS dengan *Secure Coprocessor*

Keuntungan dari aplikasi *secure coprocessor* pada *host-based* IDS hampir sama dengan *service processor* pada beberapa *server* dan *mission critical systems* saat ini. Artinya kemampuan untuk fungsi independen dari sistem *host* dan *software* aplikasi serta untuk perluasan, independen untuk komponen *hardware* yang berbeda dan subsistem dari sistem *host*. Bagaimanapun *service processor* sendiri tidak kebal secara fisik ataupun *logic* sebagai pertimbangan penggunaan. Dalam kebanyakan kasus *service processor* tidak didesain dengan pertimbangan keamanan, sehingga

tidak dapat dipercaya untuk perluasan penggunaan, dimana *secure processor* dapat melakukan fungsi tersebut (setelah mengalami pengembangan, review, inspeksi, dan evaluasi keamanan). Kebanyakan *service processor* juga tidak menyediakan fungsi untuk lingkungan komputasi yang general, penyimpanan yang aman, manajemen data, atau kemampuan untuk menampilkan operasi kriptografi dengan cara tepat waktu.

IDS *agents* dan beberapa kebijakannya memaksa dapat ditempatkan dan dieksekusi secara internal untuk dapat menjamin kerja yang baik dari *secure coprocessor*. *Agent* ini harus dapat mengakses data disisi *host* pada perangkat penyimpanan, atau pada main memori untuk analisis *event log*, pencocokan pola/*signature* dan sejenisnya. Pada beberapa kejadian, data mentah seperti sistem dan aplikasi *event log*, tingkah laku dan statistik penggunaan, dan yang lainnya dapat berdampingan dalam *secure processor* dengan baik untuk menjamin integritas dan kerahasiaan dari data tersebut (pengujian dari beberapa bit untuk integritas dapat dipelihara pada *secure processor* untuk data yang ditempatkan pada *host* barangkali dapat digunakan untuk tujuan ini dengan baik), juga pengontrolan akses ke data ini. Dikarenakan kemampuan untuk menjamin integritas data, penempatan data mentah pada *secure coprocessor* dengan kemampuan tepat waktu yang telah diuji, bahkan yang bersifat sementara juga harus difasilitasi dengan analisis (*offline*), sebaik data *forensic* dan mendukung untuk beberapa potensi yang dapat digugat secara hukum seperti, penuntutan dan pertanggungjawaban.

*Secure coprocessor* dapat juga digunakan untuk mengamati data yang ditargetkan, target program, target sumber daya sistem, atau *account* bilamana penyalahgunaannya sudah dicurigai. Pengamatan sama seperti mengumpulkan data general dan menganalisis untuk mendapatkan deskripsi yang terjadi sebelumnya. Bagaimanapun pengamatan lebih baik dipakai untuk serangan yang spesifik. Dengan menggunakan *secure coprocessor* menjadikan *host* secara fakta telah melakukan pengamatan, atau menerapkan beberapa fungsi IDS selain sebagai batas dari apa yang ditampilkan secara normal.

*Secure coprocessor* juga dapat digunakan dalam koneksi dengan software sistem pada *host*, melalui perintah tidak langsung atau kolaborasi, untuk mendapat hak akses dan hak istimewa lainnya, atau untuk memantau panggilan sistem yang khusus dan penggunaan sumber daya sistem. *Secure coprocessor* juga merupakan *platform* ideal untuk respon otomatis. Seperti untuk merespon terhadap serangan yang telah terantisipasi (dimana satu set aktivitas cocok dengan *signature* dari pola awal serangan), sehingga dapat menghentikan serangan atau penyalahgunaan sebelum hal itu terjadi.

Kemampuan untuk menyesuaikan konfigurasi dan ukuran pada kondisi yang baru secara terus menerus (serangan lambat) dan kapasitas ruangan (jumlah *host* yang besar), dimana sentral sistem kendali tidak dapat mencukupi pengalamatan dari jumlah data yang membutuhkan untuk dianalisa dalam waktu yang cepat, dan itu merupakan masalah dari kebanyakan solusi IDS saat ini. Pada lingkungan dimana *secure coprocessor* diterapkan untuk tujuan-tujuan IDS, IDS kolektif, atau terakhir untuk fungsionalitas analisis, barangkali dapat diimplementasikan dalam sebuah cara

terdistribusi tingkat tinggi, dalam rangkaian untuk memperoleh keuntungan dari pemrosesan sumber daya pada kolektif sistem dalam *secure coprocessor*, masing-masing dengan kemampuan untuk berkomunikasi secara aman dan independen satu dengan yang lain.

Kehandalan sumber daya informasi (seperti *agent* dan *sensor*) sama baiknya dengan kehandalan dari *engine* analisis dan mekanisme respon, kelambatan respon juga menjadi masalah bagi kebanyakan solusi IDS saat ini. Dengan kemampuan untuk mengamankan eksekusi program khusus dan menjamin integritas dari data kolektif, *secure processor* menawarkan keunggulan unik untuk menangani seluruh problem kehandalan

Pola/*signature* dan aturan/kebijakan yang ditentukan pada *host-based* IDS menunggu laporan, dan pemberian tanggapan/respon harus dirahasiakan. Kalau tidak, penyerang akan dengan mudah dapat menghindari pola-pola yang spesifik tadi, mengatur waktu penyerangan untuk menghindari *event timing windows*, atau bereaksi pada peluncuran serial atau serangan yang saling terkait sebelum IDS dapat melawan serangan yang asli. Sebagaimana IDS dan deteksi pola/*patern* sepertinya tidak pernah secara menyeluruh dapat melindungi informasi dari para penyerang.

Mempunyai kontrol fisik dan perlindungan pada IDS adalah juga perlu untuk melindungi IDS dari miskonfigurasi oleh *host* administrator/operator. Hal ini menjadi isu sangat penting sebagai fungsionalitas IDS agar lebih dapat tampil dimanapun pada saat yang sama.

Menggunakan keamanan berbentuk fisik (*hardware*) yang disediakan oleh *secure coprocessor* dan IPsec atau SSL untuk membangun koneksi dari pusat keamanan perusahaan ke IDS, memungkinkan dilakukannya manajemen jarak jauh seperti, meng-update, dan pemantauan. Menggunakan *secure coprocessor* yang bersertifikat memungkinkan pemakai untuk membangun kepercayaan terhadap *secure coprocessor* itu sendiri.

Serupa pada banyak aplikasi, *host-based* IDS dengan penerapan *secure coprocessor* memerlukan fungsi-fungsi umum yang cukup dan kemampuan memproses kriptografi untuk memenuhi penyelesaian *task* yang perlu dengan tepat waktu. Sejak IDS menjadi bagian dari intensif data, diperlukan sistem yang cukup dan local bus *bandwidth* yang sama baiknya dengan *bandwidth* jaringan, dan kapasitas keamanan pada penyimpanan data, yang merupakan faktor penting pada penerapan *host-based* IDS.

### 2.3 Vulnerability Security Scanner

Melakukan *scan* terhadap kerentanan sistem merupakan pendekatan yang diambil secara langsung pada sistem keamanan berbasis *host* (*host-based*). Dimana terkenal dengan istilah "*black box*" atau metodologi buta. Solusi ini melakukan *scan* pada jaringan terhadap perlakuan *up to date* basis data dari kerentanan yang telah dikenali. *Host* dan layanan-layanannya adalah yang pertama diidentifikasi sebagai yang terbaik untuk memungkinkan diperluas dan kemudian saling dihubungkan.

Perlakuan *scanner* ini dilakukan sampai ditemui kelemahan-kelemahan versi *script* dari eksploitasi yang telah dikenal dengan harapan dapat dikirim kepada administrator secara komprehensif sebagai laporan daripada penerobosan sistem keamanan melewati infrastruktur yang ada. Efek dari pemindahan ini adalah pertanggungjawaban oleh vendor dari pengenalan penemuan baru yang telah dieksplotasi dan dilaksanakan, dan untuk mengurangi beban administratif daripada pemeliharaan profil sistem keamanan yang tinggi.

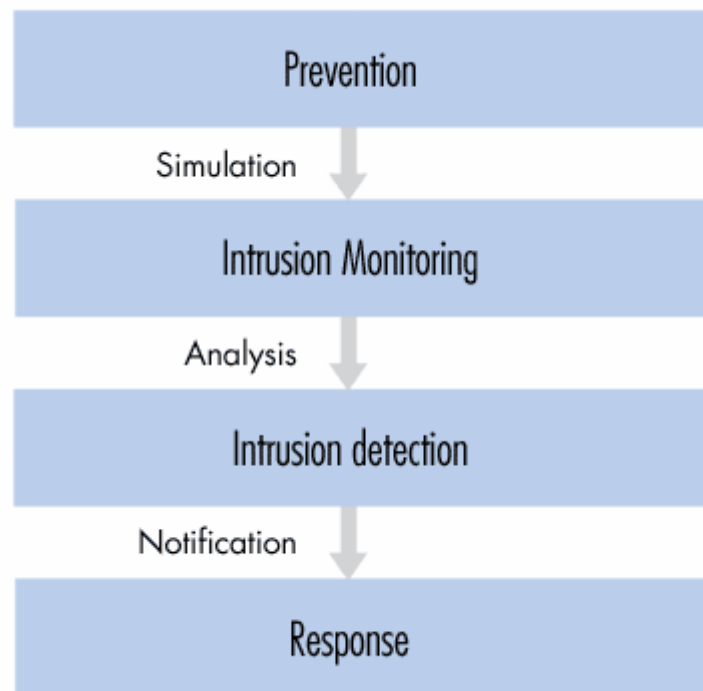
Salah satu produk komersial terkemuka pada area ini adalah [Founstone Enterprise™](#). Software manajemen vulnerabilitas dari founstone, memiliki keahlian pada keamanan strategis. Foundstone tumbuh besar dengan cepat dan merupakan salah satu solusi yang efisien dengan kemampuan scan ribuan host dapat dilakukan dalam waktu relatif pendek. Aset jaringan mengidentifikasi dengan *scan* yang diikuti oleh pemeriksaan kerentanan dengan menggunakan simulasi *script* dari eksploitasi *hacker* untuk menemukan kerentanan-kerentanan yang terbuka.

Sebagai tambahan pada keunggulan tools yang tercepat dan terakurat dalam solusi scanning komersial yang ada saat ini, Foundstone Enterprise™ mempunyai beberapa fitur yang sekaligus dapat membuat laporan kerja *scanning* yang sangat berguna untuk administrator dan disukai para manajer.

Pemain lain pada arena ini termasuk [Internet Security Scanner](#) dari ISS dan [Retina Network security Scanner](#) dari Eeye

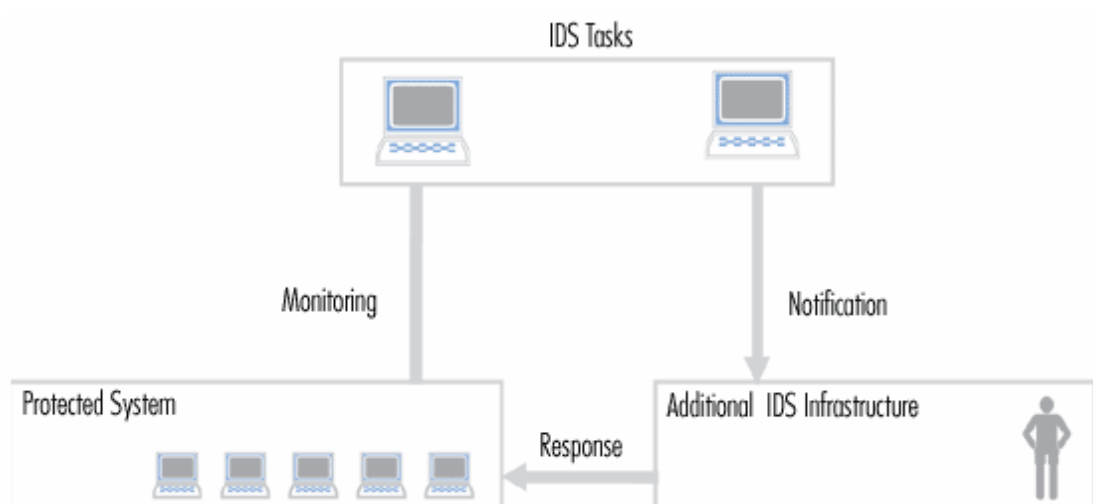
#### **2.4 Kemampuan Task Yang Dibebankan Pada Host-Based IDS**

Kemampuan utama dari *host-based* IDS adalah mempertahankan sistem komputer dari serangan dengan cara mendeteksi serangan dan memungkinkan mengusirnya. Mendeteksi serangan yang bermusuhan tergantung pada jumlah dan tipe dari aksi yang tepat (Gambar 2.2). Pencegahan intrusi memerlukan kombinasi pilihan yang baik dari *baiting and trapping* yang keduanya bertujuan untuk menginvestigasi daripada tantangan.



Gambar 2.2 Aktivitas IDS[1]

Pengalihan perhatian intruder dari sumber daya yang dilindungi adalah tugas lain dari *host-based* IDS. Keduanya merupakan sistem nyata dan sistem perangkat yang memungkinkan secara konstan dipantau. Data yang dibawa lewat *host-based* IDS diperiksa dengan hati-hati (hal ini merupakan tugas utama dari setiap IDS) untuk deteksi dari kemungkinan serangan (intrusi)



Gambar 2.3 infrastruktur IDS [1]

Sekali sebuah penyusup telah dideteksi, IDS menerbitkan tanda bahaya, mengingatkan administrator pada fakta yang terjadi. Langkah berikutnya mengatasi intrusi yang dapat juga dilakukan oleh administrator atau IDS itu sendiri. Dengan mengambil keunggulan dari tambahan nilai untuk melawan ( fungsi blok khusus untuk mematikan sesi, sistem backup, koneksi routing pada perangkat sistem,

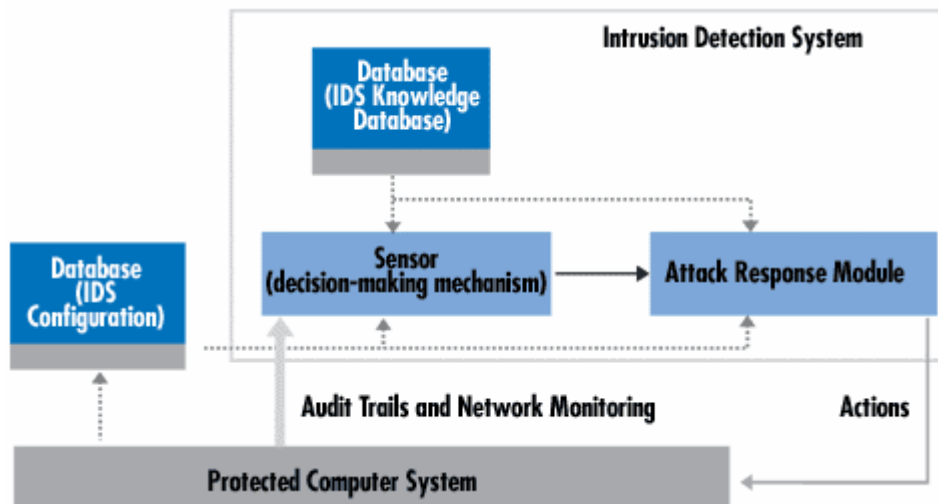
infrastruktur legal dan lainnya). Kebijakan keamanan organisasi (Gambar 2.3). dimana sebuah IDS adalah merupakan elemen dari kebijakan keamanan.

Diantara variasi tugas IDS, identifikasi intruder adalah salah satu dari yang paling fundamental. Ini sangat berguna dalam riset forensik pada kecelakaan atau bencana. Penginstalan dengan penambalan yang sesuai untuk dapat mendeteksi pada percobaan serangan yang akan datang yang menargetkan person yang spesifik atau pada sumber daya server.

Deteksi intrusi dapat pula kadang-kadang menghasilkan *alarm* yang salah, sebagai contoh hasil dari kesalahan fungsi dari antar muka jaringan, atau mengirim deskripsi serangan atau signature lewat email.

## 2.5 Struktur Dan Arsitektur Pada Host-Based IDS

Sebuah sistem deteksi intrusi selalu mempunyai elemen inti sebuah sensor (sebuah analisis engine) yang bertanggungjawab untuk mendeteksi intrusi. Sensor ini terdiri dari mekanisme pengambilan keputusan mengenai intrusi-intrusi. Sensor menerima data mentah dari tiga sumber informasi utama. Hal ini ditunjukkan pada gambar 2.4 berikut.



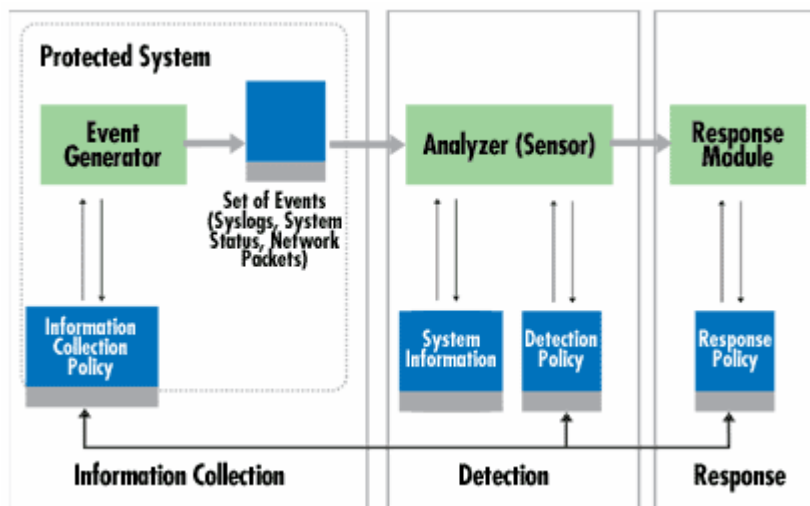
Gambar 2.4 Contoh IDS[1]

(lebar panah mewakili jumlah informasi yang mengalir diantara komponen sistem)

IDS memiliki knowledge base, syslog dan audit jejak(foot print). Syslog mungkin termasuk yang ditangani, sebagai contoh, konfigurasi pada sistem file, otorisasi pemakai dan lain-lainnya. Informasi ini menciptakan dasar untuk proses pengambilan keputusan lebih jauh.

Sensor diintegrasikan dengan komponen yang bertanggungjawab untuk koleksi data (Gambar 2.5) merupakan sebuah generator kejadian. Tingkah laku dari kumpulan ditentukan oleh kebijakan generator kejadian yang membatasi mode *filter* dari informasi tentang peringatan kejadian. Generator kejadian ( sistem operasi, jaringan, aplikasi) menghasilkan satu set kebijakan yang konsisten pada kejadian yang

barangkali sebuah *log* (atau audit) dari kejadian sistem, atau paket jaringan. Hal ini merupakan satu set yang dikirim bersama informasi kebijakan yang dapat disimpan bahkan dalam sistem yang terproteksi atau sama sekali diluar. Dalam beberapa kasus, tidak terdapat tempat penyimpanan data yang tersedia, sebagai contoh, bilamana aliran data kejadian sedang ditransfer secara langsung kedalam *analyzer*. Dalam urusan ini paket jaringan dikhususkan.



Gambar 2.5 IDS komponen [1]

Fungsi dari sensor adalah untuk menyaring informasi dan mengabaikan beberapa data yang tidak relevan, informasi ini diamankan dari set kejadian yang disatukan dengan perlindungan sistem, sehingga mampu mendeteksi aktivitas yang mencurigakan. *Analyzer* menggunakan database kebijakan deteksi untuk fungsi ini. Belakangan yang termasuk elemen ikutan adalah seperti : signature serangan, profil tingkah laku normal, parameter yang diperlukan (thresholds). Sebagai tambahan, database dapat memelihara/menahan parameter konfigurasi IDS , termasuk mode-mode komunikasi dengan modul respon. Sensor juga mempunyai database sendiri berisikan history dinamis dari kompleksitas intrusi yang potensial (diciptakan dari multiple aksi).

## 2.6 Klasifikasi dari Host-Based IDS

Prinsipnya sebuah sistem deteksi intrusi melibatkan diri pada aksi yang saling bermusuhan. Tool-tool keamanan jaringan umumnya menggunakan salah satu dari dua teknik yaitu[4] :

- *Anomaly detection*, meneliti pendapat dalam deteksi intrusi diasosiasikan dengan penyimpangan dari sistem normal atau tingkah laku pemakai,
- *Signature detection*, membedakan antara anomali atau pola (*signature*) serangan dan dikenal sebagai deteksi intrusi menggunakan *signature*.

Kedua metoda ini mempunyai beberapa keunggulan-keunggulan dan kekurangan-kekurangan sesuai dengan area kerja aplikasi deteksi intrusi.

Ketika mempertimbangkan sebuah area sebagai sumber data yang digunakan untuk deteksi intrusi, pertimbangan perbedaan klasifikasi dari sistem deteksi intrusi dapat digunakan dalam hubungannya dengan tipe-tipe sistem yang akan dilindungi.

Terdapat kelompok dari tool sistem deteksi intrusi yang menggunakan informasi yang diambil dari sebuah *single host (system)* yang disebut *host-based IDS*, dan yang lain menggunakan penelitian informasi yang datang dari seluruh segmen dari sebuah *local network* disebut *network-based IDS*.

Terdapat dua tipe utama dari HBIDS yang dapat dikenalkan[4] :

- Sistem yang memonitor kedatangan sebuah koneksi (RealSecure Agent, PostSentry). Yaitu, memeriksa kedatangan pada host dan keluaran melalui koneksi jaringan. Hal ini terutama berhubungan dengan percobaan koneksi dari yang tidak berhak melalui port TCP atau UDP dan dapat juga mendeteksi kedatangan *portscans*.
- Sistem yang memeriksa lalu lintas jaringan (paket) yang mencoba mengakses *host*. Sistem ini melindungi *host* dengan memotong paket yang mencurigakan dan mencari sesuatu yang tidak biasanya (inspeksi paket)
- Sistem yang memonitor aktivitas *login* melalui layer jaringan dari *host* yang dilindungi (*HostSentry*). Mereka berperan untuk memonitor *login* dan *logout*, mencari aktivitas yang tidak biasanya terjadi pada sistem dalam waktu yang tidak terduga, terutama pada lokasi jaringan atau mendeteksi percobaan *login* berkali-kali (khususnya salah satu keagalannya).
- Sistem yang memantau tingkah sebuah *super-user (root)* yang memiliki hak istimewa tertinggi (*logCheck*). IDS melakukan scan untuk aktivitas yang tidak biasanya, aktivitas *super-user* ditingkatkan, atau aksi yang ditampilkan pada waktu-waktu utama, dan lainnya.
- Sistem yang memantau integritas sistem file (Tripwire, AIDE). Tool yang mempunyai kemampuan ini (*integrity checker*) diijinkan untuk mendeteksi beberapa perubahan pada file yang sangat penting untuk sistem operasi.
- Sistem yang memantau status register sistem (hanya *Windows platform*). Mereka didesain untuk mendeteksi perubahan yang illegal dalam register sistem dan memberikan tanda waspada kepada sistem administrator untuk kejadian ini.
- Sistem deteksi intrusi berbasis *kernel* (Els00). Hal ini khususnya disediakan pada Linux (LIDS, Openwall). Sistem ini memeriksa status dari file kunci sistem operasi dan aliran-aliran, mencegah *buffer overflow*, memblokir komunikasi antar proses yang tidak biasanya, pencegahan sebuah *intruder* dari penyerangan ke sistem. Kemampuan tambahan lain yaitu mereka dapat memblokir sebagian dari tindakan yang diambil oleh *super-user (restricting privilege)*.

Produk-produk Host-based IDS seperti : [Snort](#), [Dragon Squire](#), [Emerald eXpert-BSM](#), [NFR HID](#), [Intruder Alert](#) seluruhnya dapat menampilkan kemampuan untuk tipe pemantauan diatas.

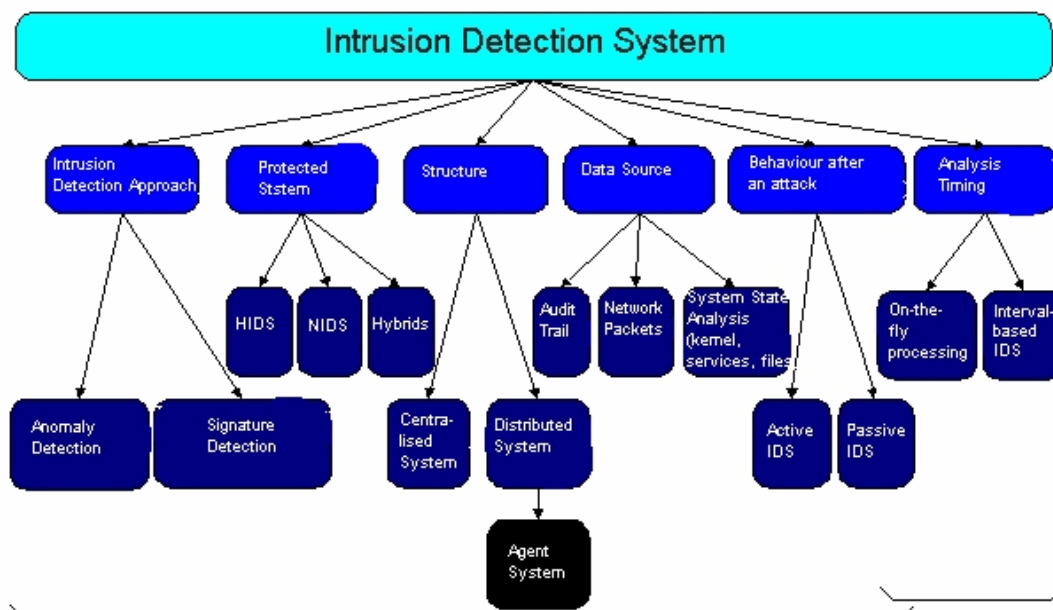
Pertimbangan mutu dari *host-based IDS* dan *network-based IDS* sebagai klasifikasi terpisah dari *network node IDS (NNIDS)*, dimana mempunyai *agent* sendiri yang diterapkan pada setiap *host (server)* dengan perlindungan meliputi seluruh jaringan (khusus NIDS menggunakan *agent* jaringan untuk memantau seluruh segmen LAN). Secara fakta NNIDS beroperasi mirip seperti hybrid *host-network IDS* dimulai sebuah *agent* yang biasanya memproses lalu lintas jaringan langsung ke *host*

(pendekatan tingkah laku manusia). Alasan utama untuk pengenalan hybrid IDS adalah kebutuhan untuk kerja secara *online* dengan jaringan yang terenkripsi dan datanya ditempatkan pada sebuah *single host* (hanya sumber dan tujuan dapat melihat lalu lintas jaringan yang didekripsi). Kebanyakan tawaran dari tool komersial yang besar pada sistem deteksi intrusi adalah sebuah *shim-hybrid*, yang menggabungkan keunggulan dari HBIDS dan NIDS dalam konsep yang unik.

Host-based IDS hanya melihat lalu-lintas pada host yang mendeteksi serangan dari lokal ke lokal lainnya, atau serangan dari lokal ke *root*, dimulai dengan konsep yang jelas dari informasi yang tersedia secara lokal, contoh : mereka dapat meneliti pemakai IDS. Juga fitur *tools* deteksi anomali memberi jangkauan lebih baik dari problem internal. Sejak Host-based memiliki kemampuan deteksi yang didasari pada pola-pola tingkah laku yang normal dari pemakai

Sistem deteksi intrusi dapat beroperasi *standalone*, aplikasi terpusat, atau aplikasi terpadu yang membuatnya menjadi sistem terdistribusi. Belakangan IDS mempunyai sebuah arsitektur utama dengan agent otomatis yang dapat menyela dan mengukur reaksi bahkan memindahkan melalui jaringan.

Klasifikasi sistem deteksi intrusi ditunjukkan oleh Gambar 2.6 berikut.



Gambar 2.6 Klasifikasi Sistem Deteksi Intrusi[4]

### 2.7 Keunggulan-Keunggulan dari Host-Based Intrusion Detection Systems

Kalaupun HBIDS tidak secepat *network-based* IDS, tetapi tetap menawarkan keunggulan-keunggulan yang tidak didapatkan pada NIDS. Keunggulan ini termasuk analisis forensik yang lebih kuat, lebih fokus pada data kejadian yang khusus di host dan biaya awal yang lebih murah.

Beberapa keunggulan HBIDS sebagai berikut.

- 1. Memastikan berhasil atau gagalnya dari sebuah serangan.** Dengan menggunakan *log audit* HBIDS dapat mengukur apabila sebuah serangan berhasil atau gagal dengan akurasi yang tinggi.
- 2. Memonitor aktivitas sistem khusus,** HBIDS memonitor *user* dan aktivitas akses file, termasuk akses-akses file, perubahan pada ijin penggunaan file, percobaan untuk instalasi file *executable* yang baru, dan percobaan untuk mengakses layanan khusus. Teknologi HBIDS juga dapat memonitor aktivitas yang normalnya dieksekusi hanya oleh administrator, seperti sistem operasi melakukan *log* beberapa *event* dimana *user account* ditambahkan, dihapus, atau dimodifikasi. Hal ini dapat dideteksi terutamerubahan yang tidak sepatutnya pada saat perubahan itu dieksekusi. HBIDS juga dapat melakukan audit perubahan kebijakan yang berpengaruh pada cara sistem *log*-nya. Terakhir HBIDS dapat memonitor perubahan pada file *key system* dan file *executable*. Percobaan untuk melakukan *overwrite* file sistem vital, atau untuk menginstall *trojan horse* atau *backdoors*, dapat dideteksi dan dihentikan.
- 3. Mendeteksi serangan yang tidak terdeteksi oleh *network-based*.** Seperti serangan dari keyboard pada server yang penting yang tidak melewati jaringan, dan sudah pasti tidak dapat dilihat oleh *network-based*.
- 4. Sangat sesuai untuk lingkungan terenkripsi dan menggunakan Switch.** Sejak HBIDS dipasang pada enterprise yang berbeda, mereka dapat mengatasi beberapa tantangan pada penerapan dibandingkan dengan *network-based* pada lingkungan terenkripsi dan menggunakan switch. Switch memudahkan mengatur jaringan yang besar menjadi beberapa segmen jaringan yang lebih kecil. Dengan teknik ini dapat membantu melihat lalu lintas dan mengadministrasi *port*. HBIDS menyediakan sudut pandang yang lebih besar dalam lingkungan sebuah switch dengan pemasangan pada beberapa *host* yang vital bila diperlukan. HBIDS dapat melihat serangan jikalau terdapat pengiriman melalui protokol yang terenkripsi oleh waktu datang sebuah sistem operasi.
- 5. Deteksi dan respon mendekati waktu nyata.** Meskipun *host-based intrusion detection* tidak benar-benar *real time* pada responnya, ini dapat memperoleh hasil yang sangat mendekati jika diimplementasikan dengan benar. Tidak seperti sistem sebelumnya dimana membutuhkan proses untuk memeriksa status dan isi dari *log file* pada awal interval, sehingga dapat mengintrupsi sistem operasi bila sebuah *log file* baru masuk. Tetapi pada sistem HBIDS yang baru hal seperti ini tidak terjadi sehingga *log file* yang masuk dapat di proses langsung, dengan demikian dapat mengurangi waktu antara mengenali serangan dan respon
- 6. Hardware tambahan tidak dibutuhkan.** HBIDS dipasang pada infrastruktur jaringan yang sudah ada termasuk *server file*, *web server*, dan sumber daya terbagi lainnya. Dengan efisiensi ini HBIDS dapat mengurangi biaya karena tidak membutuhkan instalasi yang baru pada jaringan yang disyaratkan untuk pengalamanan, pemeliharaan, dan manajemen.

7. **Biaya lebih murah.** Karena tidak membutuhkan ifrastruktur baru HBIDS memiliki biaya pemasangan lebih murah dibandingkan dengan network-based

## 2.8 Serangan–Serangan Pada Host Based IDS

### 2.8.1 Taksonomi Serangan Dan Intrusion

Sejak *intrusion detection system* menangani pelanggaran akses yang illegal(hacking), mari kita melihat lebih dekat tentang bahaya aktivitas ini. Untuk membantu dalam pemahaman tentang hal tersebut terdapat beberapa taksonomi, denisi-definisi yang akan membantu walaupun masih bervariasi[1].

- *Intrusion*, deretan aktifitas yang tergabung yang diposisikan untuk melawan keamanan dari sumber daya IT dari akses yang tidak syah kedalam komputer yang spesifik dan atau ke alamat domain.
- *Incident*, pelanggaran pada aturan kebijakan keamanan sistem yang boleh dikenali sebagai penyusupan yang sukses.
- *Attack*, percobaan memasuki sistem yang gagal ( belum terjadi pelanggaran).
- *Modeling of intrusion*, pemodelan berbasis waktu dari aktivitas yang mengubah sebuah intrusi. Intruder memulai penyerangannya dengan aksi perkenalan diikuti dengan suatu bantuan (atau penghindaran) untuk memproses akses yang sukses, dalam prakteknya beberapa percobaan untuk garansi selama serangan oleh seorang; sebagai contoh oleh manajer sumber daya IT dapat diidentifikasi sebagai serangan

Secara umum serangan dapat dikategorikan dalam dua area yaitu :

- Pasif, bertujuan untuk mendapatkan keuntungan akses untuk penetrasi kedalam sistem tanpa meminta persetujuan sumber daya IT,
- Aktif, memperoleh hasil dalam status perubahan yang tidak syah dari sumber daya IT

Dalam hubungan istilah intruder-victim, serangan dapat dikategorikan sebagai :

- Internal, datang dari dalam pegawai perusahaan sendiri atau partner bisnisnya atau dari pelanggan.
- Eksternal, datang dari luar, biasanya lewat internet.

Serangan juga diidentifikasi oleh kategori sumber daya, dengan nama yang ditampilkan dari dalam sistem (jaringan lokal), dari internet atau dari *dial in* jarak jauh. Sekarang mari kita lihat tipe apa pada serangan dan penyalahgunaan yang dapat dideteksi ( kadang dteksinya sulit) oleh *tool* IDS untuk menempatkan mereka dalam kategorisasi sementara. Tipe serangan yang dapat diidentifikasi sebagai berikut.

- Yang berhubungan dengan akses yang illegal ke sumber (sering hanya sebagai perkenalan selanjutnya aksinya lebih canggih) :
  - Pemecahan *password* dan pelanggaran akses,
  - *Trojan horses*,
  - *Interception*; kebanyakan bersama-sama dengan pencurianTCP/IP dan mengintrupsi yang sering menggunakan mekanisme tambahan untuk

- kompromi operasi dari sistem yang terserang ( contohnya dengan *flooding*); orang yang ditengah-tengah serangan),
- *Spoofing*; dengan tujuan mengubah arah informasi atau membungkus identitas host dengan menempatkan data ditempatkan dalam cache yang bernama server, contohnya DNS spoofing,
  - *Scanning port* dan layanan, termasuk menscan ICMP (ping), UDP, TCP, stealth Scanning TCP yang mengambil keuntungan dari koneksi partial pada pembangunan protokol, dan lainnya,
  - *Remote OS fingerprinting* , sebagai contoh dengan uji tipe respon pada paket yang spesifik, pengalamatan dari *port* yang terbuka, respon aplikasi standar, parameter stack IP dan lainnya,
  - *Network packet listening*; sebuah seranga pasif yang sulit dideteksi tetapi kadang-kadang dimungkinkan,
  - *Stealing information* , contoh penyingkapan dari informasi pribadi,
  - *Koneksi jaringan tidak legal*,
  - Penggunaan sumber daya IT untuk penggunaan pribadi, contoh mengakses situs pornografi,
  - Mengambil keuntungan dari kelemahan sistem untuk meningkatkan akses ke sumber atau hak-hak istimewa.
- Perubahan yang tidak legal dari sumber (setelah memperoleh akses ilegal) :
    - Memalsukan identitas, sebagai contoh untuk mendapatkan hak sistem administrator,
    - Perubahan informasi dan penghapusan informasi,
    - Pelanggaran transmisi dan membuat data set, contoh mengatur database dari nomor kartu kredit pada komputer pemerintah (pencurian yang spektakuler dari beberapa ribu nomor kartu kredit pada tahun 1999),
    - Penukaran konfigurasi yang tidak syah pada sistem dan layanan jaringan (server).
  - Denial of Service (DoS) :
    - Flooding, mengkompromikan sebuah sistem dengan mengirim sejumlah besar informasi yang tidak berguna untuk memacetkan lalu lintas dan menolak layanan :
      - Ping flood (Smurf), jumlah yang banyak dari paket ICMP dikirim ke alamat pemancaran,
      - Mengirim mail flood, membanjiri dengan ratusan ribu pesan dalam periode waktu yang pendek, juga tergantung pada POP dan SMTP,
      - SYN Flood, menginisialisasi jumlah permintaan TCP yang besar dan tidak dilengkapi handshake sebagai disyaratkan oleh protokol,
      - Distributed Denial of Service (DdoS); datang dari berbagai sumber,
    - Mengkompromikan sistem dengan mengambil keuntungan dari kerentanannya :
      - Buffer Overflow, contoh Ping of Death ; mengirim ICMP sangat besar (melampaui 64 KB),
      - Remote sistem Shutdown,
-

- Serangan Aplikasi Web ; serangan yang mengambil keuntungan dari *bugs* aplikasi yang dapat menimbulkan masalah seperti diatas.

Hal penting yang perlu diingat, bahwa kebanyakan serangan tidak hanya aksi tunggal, lebih banyak berupa seri dari kejadian individual dikembangkan dalam tingkah laku yang terkoordinasi.

### 2.8.2 Anomaly versus Signature Detection

Sistem deteksi intrusi harus berkemampuan untuk membedakan antara pemakai yang normal dan aktivitas pemakai yang tidak normal, untuk menemukan percobaan pelanggaran tepat waktu. Bagaimanapun menterjemahkan tingkah laku pemakai dalam keputusan berkaitan dengan keamanan yang konsisten seringkali tidak begitu mudah. Beberapa pola tingkah laku tidak dapat diduga dan tidak jelas (Gambar 2.7). Untuk penggolongan aksi-aksi, sistem deteksi intrusi mengambil keuntungan pada pendekatan deteksi anomali, yang sering dirujuk sebagai *behavior based* [Deb99] atau signature serangan misalnya deskripsi materi pada tingkah laku abnormal yang dikenal (*signature detection*), [Axe00, Jon00, Kum95] juga disebut *knowledge based*.



Gambar 2.7 Tingkah Laku Pemakai Pada Sitem[4]

- **Pola tingkah laku normal menggunakan deteksi anomali**, sangat berguna untuk memprediksi tingkah laku pemakai dan tingkah sistem. Disini detektor anomali membuat profil yang mewakili penggunaan normal dan kemudian data itu untuk mendeteksi kemungkinan tidak cocok antara profil dan mengenali kemungkinan ada percobaan serangan.

Untuk mencocokkan profil *event*, sistem disyaratkan untuk menghasilkan inisial profil pemakai untuk melatih sistem dengan menghormati pada tingkah laku pemakai yang syah. Masalah yang berhubungan dengan pembuatan profil diantaranya : kapan sistem diijinkan untuk ,belajar' sendiri, intruder berpengalaman (atau user) dapat melatih sistem pada poin ini dimana tingkah laku intrusif sebelumnya menjadi tingkah laku normal. Sebuah profil yang tidak sesuai akan dapat mendeteksi seluruh kemungkinan aktivitas intrusif. Lebih jauh terdapat sebuah kebutuhan nyata untuk *updating* profil dan pelatihan sistem yang sangat sulit dan makan waktu lama.

Diberikan satu set profil tingkah laku normal, semuanya yang tidak cocok dengan profil yang disimpan akan dipertimbangkan sebagai aksi mencurigakan. Kemudian sistem ini dikarakterisasi oleh efisiensi deteksi yang sangat tinggi (mereka dapat mengenali banyak serangan yang baru ke dalam sistem), tetapi tedensi untuk membuat alarm palsu biasanya merupakan sebuah masalah.

Keuntungan dari metoda deteksi anomali adalah : kemungkinan daripada deteksi pada serangan novel sebagai intrusi; anomali dikenali tanpa harus masuk kedalam penyebab dan karakteristik; kelemahan pertahanan dari IDS pada lingkungan operasi; kemampuan untuk mendeteksi penyalahgunaan dari hak istimewa pemakai.

- **Misbehavior Signature**

Sistem mengontrol informasi pada tingkah laku yang abnormal, tingkah laku tidak aman ( serangan *signature-based systems*) sering digunakan pada real-time IDS

Terdapat dua kategori *misbehavior signature* diantaranya :

- Serangan Signature, mereka menggambarkan pola aksi yang berposisi sebagai ancaman keamanan. Khususnya mereka hadir bersamaan dengan waktu hubungan antara urutan aktivitas yang barangkali saling silang dengan salah satu yang netral.
- String teks terseleksi, signature mencocokkan string teks yang digunakan untuk mencari aksi yang mencurigakan.

Beberapa aksi yang tidak dipertimbangkan dengan jelas dilarang diijinkan. Jadi keakuratannya sangat tinggi (timbulnya alarm palsu sangat kecil). Khususnya mereka tidak mencapai keseluruhannya dan tidak kebal terhadap serangan novel.

Terdapat dua pendekatan utama yang berhubungan dengan deteksi signature diantaranya :

- Verifikasi dari pathologi pada paket layer bawah, beberapa tipe serangan (*Ping of Death* atau *TCP Stealth Scanning*) mengeksploitasi kerusakan dalam IP, TCP, UDP, atau paket ICMP. Dengan pemeriksaan yang sederhana pada *flag set* pada paket yang spesifik menjadi mungkin untuk menentukan apakah paket tersebut legitimat atau tidak. Kesulitan barangkali terjadi dengan kemungkinan fragmentasi paket dan kebutuhan untuk merakit kembali. Kemiripan dari beberapa masalah ditimbulkan pada layer TCP/IP dari sistem yang diproteksi. Hal ini dikenal dengan baik oleh para hacker yang menggunakan fragmentasi paket untuk melakukan bypass pada tools IDS[4].
- Verifikasi dari protokol layer aplikasi, beberapa tipe serangan (*WinNuke*) mengeksploitasi kerusakan program, contohnya pengiriman data diluar jangkauan ke sebuah koneksi jaringan mapan. Untuk mendeteksi secara efektif sebuah serangan IDS seharusnya mengimplementasikan beberapa protokol layer aplikasi.

Metoda deteksi signature mempunyai beberapa keunggulan-keunggulan berikut : rasio alarm palsu sangat kecil, algoritmanya sederhana, mudah mengkreasi database serangan signature, mudah diimplementasikan dan khususnya minimal dalam penggunaan sumber daya sistem.

Secara komersial penawaran produk IDS seringkali menggunakan metoda deteksi signature untuk dua alasan. Pertama, adalah lebih mudah untuk diberikan signature yang akan dihubungkan dengan abstraksi serangan yang telah dikenal dan memberikan nama pada sebuah serangan, misalnya *Ping of Death*. Kedua, database

signature serangan seharusnya di update secara teratur (dengan menambah signature dari serangan yang baru ditemukan dan dieksploitasi), dimana dapat menjadi sumber masukan bagi vendor dari tool IDS. Update database pada waktu bersamaan mengurangi ketidak praktisan daripada yang yang dihubungkan dengan perubahan dari tipe profil tingkah laku user. Pada kasus belakangan penutupan sementara pada sistem barangkali diperlukan, bahwa beberapa aplikasi tidak dapat mentolerir dilakukan bersamaan.

Contoh dibawah mempresentasikan sebuah signature serangan diambil dari program snort yang mendeteksi ping paket ICMP lebih besar dari 800 bytes, kedatangannya dari jaringan luar dan terkoneksi dengan beberapa port[4] :

```
Alert icmp $EXTERNAL_NET any->  
$HOME_NET any (msg:"MISC large ICMP"; dsize: >800;  
Reference:arachnids,246;  
Classtype:bad-unknown; sid:499;)
```

### III. PERTIMBANGAN-PERTIMBANGAN IMPLEMENTASI HBIDS

#### 3.1 Kebijakan Perusahaan Tentang Keamanan Sistem

Jaringan perusahaan tidak menjadi aman karena seseorang telah menginstall sebuah firewall. Keamanan jaringan dimulai sebelum sistem itu pernah diluncurkan. Organisasi seharusnya memahami bagaimana menginstalasi dengan aman sebelum terkoneksi ke jaringan perusahaan. Kebijakan seperti itu harus turun dai organisasi dengan program proteksi informasi dan diarahkan oleh tujuan perusahaan dan budaya pada perusahaan[5].

Kebijakan keamanan melindungi informasi perusahaan dari aktivitas yang tidak etis, seperti pencurian pada rahasia dagang oleh individu yang tidak berhak. Sebuah program pelaksanaan membutuhkan untuk memasukkan pemantauan untuk mengetahui bila kebijakan sistem dan jaringan disalahgunakan. Tanpa pemantauan ditempat, para eksekutif tidak akan pernah mengetahui jika kebijakan keamanan mereka dilanggar.

Perusahaan mempunyai urusan penting dan alasan hukum untuk menyiapkan kebijakan yang sesuai dan rencana pelaksanaan yang cukup. Perusahaan yang gagal memonitor penyalahgunaan kebijakan akan kehilangan komponen penting pada keamanan. Jadi software IDS seharusnya diinstall pada jaringan, karena software ini akan memantau penyalahgunaan dari kebijakan. Jika perusahaan tidak menginvestasikan software IDS, ini merupakan syarat. Penundaan pada investasi tentang kebijakan keamanan menempatkan perusahaan pada resiko tidak hanya dari pengubahan, perusakan, dan pencurian data, tetapi juga pada tindakan hukum, jika sesuatu yang buruk terjadi dan *lawyer* dilibatkan, satu pertanyaan yang akan ditanyakan dalam pengadilan sekarang pada kasus yang melibatkan perusahaan kartu

kredit misalnya adalah, “apakah anda mempunyai kebijakan keamanan dan pemantauan di tempat?”[5]

### 3.2 Perlunya Implementasi Host-Based IDS

Sebuah sistem deteksi intrusi memantau dan menganalisis *event* yang terjadi pada jaringan atau sistem, menemukan percobaan intrusi (*event* yang mencoba untuk mengakses tanpa ijin tentang kerahasiaan, integritas, dan ketersediaan data).

Meningkatnya kebringasan serangan saat ini membuat IDS bagian yang perlu pada keamanan. Semenjak kebanyakan jaringan mensyaratkan deteksi intrusi. Perusahaan seharusnya memahami apa tipe IDS yang menyediakan fungsi yang diperlukan untuk melindungi infrastrukturnya. Kadang-kadang perusahaan menginvestasikan IDS yang sulit, laporan terlalu banyak *false positive*, dan tidak dapat menjaga kehandalan sesuai dengan kecepatan dari jaringan.

*False positive* adalah munculnya serangan yang dibuat oleh aktivitas yang legitimat. Sistem administrator boleh percaya bahwa sebuah serangan telah terjadi padahal faktanya tidak pernah terjadi. IDS yang melaporkan banyak *false positive* menjadi sulit dan tidak mungkin di manaj. Sesudah beberapa saat sistem administrator boleh jadi mengabaikan bahaya karena kelihatannya hanya seperti *false positive*, atau administrator berhenti melihat tanda bahaya dan data dikumpulkan, karena sangat sulit untuk menggambarkan jika sebuah serangan sebenarnya sedang terjadi.

Pada penerapan HBIDS, *software* membutuhkan untuk dipasang langsung pada host yang akan dimonitor. Sekali mendeploy pada host, *software* akan memantau file sistem, memantau proses-proses, dan memantau *log files* untuk aktivitas yang mencurigakan. Sebagai tambahan beberapa *host-based* IDS dapat memantau perubahan hak-hak istimewa pemakai. memperoleh hak istimewa level lebih tinggi atau menyiapkan *account* pemakai baru adalah pendekatan digunakan oleh pesaing pada jaringan internal. Untuk mendeteksi penyalahgunaan semacam ini pada *server* yang genting adalah penting dan perlu dipantau langsung pada *host*. Karena itu para ahli merekomendasikan sebuah kombinasi dari deteksi *host-based* dan *network-based* pada jaringan besar. Pemahaman tentang tingginya resiko pada jaringan adalah kunci suksesnya penerapan untuk kedua sistem deteksi *host-based* and *network-based*.

## IV. PENUTUP

Kebijakan tentang keamanan pada perusahaan memberi jaminan terhadap akibat penyalahgunaan hak istimewa dan perlakuan yang tidak etis pada sistem yang ada, seperti pencurian data transaksi, data rahasia perusahaan. Dengan penerapan kombinasi antara *host-based* IDS dan *Network-based* IDS, maka secara efektif dapat : mengawasi dan mencegah penetrasi kedalam sistem, memantau lalu lintas pada jaringan dan mengusir jika ada aktivitas yang mencurigakan, memantau jika terdapat tingkah laku dari pemakai ataupun sistem yang anomali, juga dapat memantau pola signature antara user yang legitimat dan serangan.

## REFERENSI

- [1] Przemyslaw Kazienko & Piotr Dorosz, Intrusion detection Systems (IDS) Part I, [www.WindowSecurity.com](http://www.WindowSecurity.com), 17 september 2004, 11.34 WIB
- [2] Joan Dyer; Ronald Perez; Rainer Sailer; Leendert Van Doorn, Personal Firewall and Intrusion Detection Systems, [www.ibm.com](http://www.ibm.com), 10 April 2004, 13.22 WIB
- [3] Internet Security Systems, Network VS Host-Based Intrusion Detection , [www.iss.net](http://www.iss.net), 22 September 2004, 8.54 WIB
- [4] Przemyslaw Kazienko & Piotr Dorosz, Intrusion detection Systems (IDS) Part II, [www.WindowSecurity.com](http://www.WindowSecurity.com), 23 September 2004, 14.56 WIB
- [5] Symantec, Justification for intrusion Detection, [www.symantec.com/symantec/Advantage-issue.htm](http://www.symantec.com/symantec/Advantage-issue.htm), 10 September 2004, 11.19 WIB