MÔN AN TOÀN MẠNG

BÁO CÁO ĐỀ TÀI 1:

NGHIÊN CỨU PHÒNG CHỐNG THÂM NHẬP TRÁI PHÉP IDS, IPS (TRÊN LINUX)

Giáo Viên Hướng Dẫn:
THẦY VÕ ĐỖ THẮNG
Sinh Viên Thực Hiện:
0512176 - NGUYỄN ĐĂNG KHOA
0512207 - PHAN HUỲNH LUÂN
0512231 - LÊ BẢO NGHI
0512281 - NGUYỄN THANH QUÂN
0512300 - LÊ GIANG THANH

THÀNH PHỐ HỒ CHÍ MINH
--2009--

Mục lục:

1 . IDS

1.1 Giới thiệu sự ra đời.

- Cách đây khoảng 25 năm, khái niệm phát hiện xâm nhập xuất hiện qua một bài báo của James Anderson. Khi đó người ta cần IDS với mục đích là dò tìm và nghiên cứu các hành vi bất thường và thái độ của người sử dụng trong mạng, phát hiện ra các việc làm dụng đặc quyền để giám sát tài sản hệ thống mạng. Các nghiên cứu về hệ thống phát hiện xâm nhập được nghiên cứu chính thức từ năm 1983 đến năm 1988 trước khi được sử dụng tại mạng máy tính của không lực Hoa Kỳ. Cho đến tận năm 1996, các khái niệm IDS vẫn chưa được phổ biến, một số hệ thống IDS chỉ được xuất hiện trong các phòng thí nghiệm và viện nghiên cứu. Tuy nhiên trong thời gian này, một số công nghệ IDS bắt đầu phát triển dựa trên sự bùng nổ của công nghệ thông tin. Đến năm 1997 IDS mới được biết đến rộng rãi và thực sự đem lại lợi nhuận với sự đi đầu của công ty ISS, một năm sau đó, Cisco nhận ra tầm quan trọng của IDS và đã mua lại một công ty cung cấp giải pháp IDS tên là Wheel.
- Hiện tại, các thống kê cho thấy IDS/IPS đang là một trong các công nghệ an ninh được sử dụng nhiều nhất và vẫn còn phát triển.
- Tại sao Gartner nói: IDS is dead? Vào năm 2003, Gartner- một công ty hàng đầu trong lĩnh vực nghiên cứu và phân tích thị trường công nghệ thông tin trên toàn cầu- đã đưa ra một dự đoán gây chấn động trong lĩnh vực an toàn thông tin : "Hệ thống phát hiện xâm nhập (IDS) sẽ không còn nữa vào năm 2005". Phát biểu này của xuất phát từ một số kết quả phân tích và đánh giá cho thấy hệ thống IDS khi đó đang đối mặt với các vấn đề sau:
    - ❖ IDS thường xuyên đưa ra rất nhiều báo động giả (False Positives).
    - ❖ Là gánh nặng cho quản trị an ninh hệ thống bởi nó cần được theo dõi liên tục (24 giờ trong suốt cả 365 ngày của năm).
    - ❖ èm theo các cảnh báo tấn công là một quy trình xử lý an ninh rất vất vả.
    - ❖ Không có khả năng theo dõi các luồng dữ liệu được truyền với tốc độ lớn hơn 600 Megabit trên giây.Nhìn chung Gartner đưa ra nhận xét này dựa trên nhiều phản ánh của những khách hàng đang sử dụng IDS rằng quản trị và vận hành hệ thống IDS là rất khó khăn, tốn kém và không đem lại hiệu quả tương xứng so với đầu tư.
- Sau khi phát biểu này được đưa ra, một số ý kiến phản đối cho rằng, việc hệ thống IDS không đem lại hiệu quả như mong muốn là do các vấn đề còn tồn tại trong việc quản lý và vận hành chứ không phải do bản chất công nghệ kiểm soát và phân tích gói tin của IDS. Cụ thể, để cho một hệ thống IDS hoạt động hiệu quả, vai trò của các công cụ, con người quản trị là rất quan trọng, cần phải đáp ứng được các tiêu chí sau:
    - ❖ Thu thập và đánh giá tương quan tất cả các sự kiện an ninh được phát hiện bởi các IDS, tường lửa để tránh các báo động giả.
    - ❖ Các thành phần quản trị phải tự động hoạt động và phân tích.

Kết hợp với các biện pháp ngăn chặn tự độngKết quả là tới năm 2005, thế hệ sau của IDS-hệ thống tự động phát hiện và ngăn chặn xâm nhập IPS- đã dần khắc phục được các mặt còn hạn chế của IDS và hoạt động hiệu quả hơn nhiều so với thế hệ trước đó.

1.2 Khái niệm

Intrusion detection system (IDS) - Hệ thống phát hiện xâm phạm: là một hệ thống phòng chống, nhằm phát hiện các hành động tấn công vào một mạng. Mục đích của nó là phát hiện và ngăn ngừa các hành động phá hoại đối với vấn đề bảo mật hệ thống, hoặc những hành

động trong tiến trình tấn công như sưu tập, quét các cổng. Một tính năng chính của hệ thống này là cung cấp thông tin nhận biết về những hành động không bình thường và đưa ra các báo cảnh thông báo cho quản trị viên mạng khóa các kết nối đang tấn công này. Thêm vào đó công cụ IDS cũng có thể phân biệt giữa những tấn công bên trong từ bên trong tổ chức (từ chính nhân viên hoặc khách hàng) và tấn công bên ngoài (tấn công từ hacker).

1.3 Chức năng.

- Chức năng quan trọng nhất : giám sát - cảnh báo - bảo vệ
  - ❖ Giám sát : lưu lượng mạng + các hoạt động khả nghi.
  - ❖ Cảnh báo : báo cáo về tình trạng mạng cho hệ thống + nhà quản trị.
  - ❖ Bảo vệ : Dùng những thiết lập mặc định và sự cấu hình từ nhà quản trị mà có những hành động thiết thực chống lại kẻ xâm nhập và phá hoại.
- Chức năng mở rộng :
  - ❖ Phân biệt : "thù trong giặc ngoài"
  - ❖ Phát hiện : những dấu hiệu bất thường dựa trên những gì đã biết hoặc nhờ vào sự so sánh thông lượng mạng hiện tại với baseline

1.4 Phân biệt IDS.

Các thiết bị bảo mật dưới đây không phải là IDS:

- Hệ thống đăng nhập mạng được sử dụng để phát hiện lỗ hổng đối với vấn đề tấn công từ chối dịch vụ (DoS) trên một mạng nào đó. Ở đó sẽ có hệ thống kiêm tra lưu lượng mạng.
- Các công cụ đánh giá lỗ hổng kiểm tra lỗi và lỗ hổng trong hệ điều hành, dịch vụ mạng (các bộ quét bảo mật).
- Các sản phẩm chống virus đã thiết kế để phát hiện phần mềm mã nguy hiểm như virus, Trojan horse, worm... Mặc dù những tính năng mặc định có thể rất giống hệ thống phát hiện xâm phạm và thường cung cấp một công cụ phát hiện lỗ hổng bảo mật hiệu quả.
- Tường lửa
- Các hệ thống bảo mật/mật mã, ví dụ như VPN, SSL, S/MIME, Kerberos, Radius

1.5 Các loại tấn công.

Các loại tấn công được phân thành hai loại như sau:

- Bị động (được trang bị để tăng mức truy cập làm cho có thể thâm nhập vào hệ thống mà không cần đến sự đồng ý của tài nguyên CNTT)
- Tích cực (các kết quả gây ra thay đổi trạng thái không hợp lệ của tài nguyên CNTT)

Dưới dạng mối quan hệ giữa nạn nhân và người xâm phạm, các tấn công được chia thành:

- Bên trong, những tấn công này đến từ chính các nhân viên của công ty, đối tác làm ăn hoặc khách hàng
- Bên ngoài, những tấn công đến từ bên ngoài, thường thông qua Internet.

Các loại tấn công có thể bị phát hiện bởi công cụ IDS. Các loại tấn công dưới đây có thể được phân biệt:

- Những tấn công này liên quan đến sự truy cập trái phép đến tài nguyên.
  - ❖ Việc bẻ khóa và sự vi phạm truy cập
  - ❖ Trojan horses
  - ❖ Đánh chặn; hầu hết kết hợp với việc lấy cắp TCP/IP và sự đánh chặn thường sử dụng các cơ chế bổ sung để thỏa hiệp hệ thống
  - ❖ Sự giả mạo
  - ❖ Quét cổng và dịch vụ, gồm có quét ICMP (ping), UDP, TCP

- ❖ Lấy dấu OS từ xa, ví dụ như việc kiểm tra phản ứng đối với các gói cụ thể, các địa chỉ cổng, phản ứng của ứng dụng chuẩn, các tham số ngăn xếp IP,…
- ❖ Nghe gói tin mạng (một tấn công thụ động rất khó khăn phát hiện nhưng đôi khi vẫn có thể)
- ❖ Lấy cắp thông tin, ví dụ như trường hợp bị lộ thông tin về quyền sở hữu.
- ❖ Lạm dụng tính xác thực; một loại hình tấn công bên trong, ví dụ: nghi ngờ sự truy cập của một người dùng xác thực có thuộc tính kỳ lạ (đến từ một địa chỉ không mong muốn)
- ❖ Các kết nối mạng trái phép
- ❖ Sử dụng tài nguyên CNTT cho các mục đích riêng, ví dụ như truy cập vào các trang có hoạt động không lành mạnh
- ❖ Lợi dụng điểm yếu của hệ thống để truy cập vào tài nguyên hoặc các quyền truy cập mức cao.

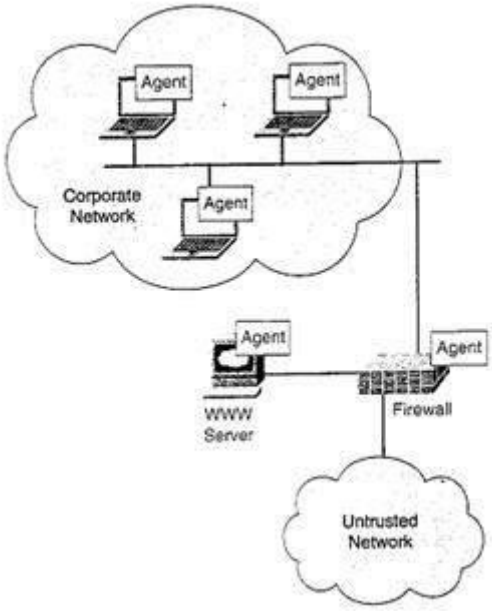
- Sự thay đổi tài nguyên trái phép (sau khi đã chiếm được quyền truy cập)
  - ❖ Xuyên tạc tính đồng nhất, ví dụ: để lấy được các quyền quản trị viên hệ thống.
  - ❖ Thay đổi và xóa thông tin
  - ❖ Truyền tải và tạo dữ liệu trái phép, ví dụ: lập một cơ sở dữ liệu về các số thẻ tín dụng đã bị mất cắp trên một máy tính của chính phủ.
  - ❖ Thay đổi cấu hình trái phép đối với hệ thống và các dịch vụ mạng (máy chủ)
- Từ chối dịch vụ (DoS)
  - ❖ Làm lụt (Flooding) – thỏa hiệp một hệ thống bằng việc gửi đi một số lượng lớn các thông tin không giá trị để làm tắc nghẽn lưu lượng hạn chế dịch vụ.
    - ▪ Ping (Smurf) – một số lượng lớn các gói ICMP được gửi đến một địa chỉ quảng bá.
    - ▪ Gửi mail – làm lụt với hàng trăm hoặc hàng nghìn các message trong một thời điểm ngắn.
    - ▪ SYN – khởi tạo một số lượng lớn các yêu cầu TCP và không tiến hành bắt tay hoàn toàn như được yêu cầu đối với một giao thức.
    - ▪ Hạn chế dịch vụ phân tán; đến từ nhiều nguồn khác nhau
  - ❖ Gây tổn hại hệ thống bằng việc lợi dụng các lỗ hổng của nó
    - ▪ Tràn bộ đệm (ví dụ: “Ping of Death” – gửi một số lượng lớn ICMP (vượt quá 64KB))
    - ▪ Tắt hệ thống từ xa
- Tấn công ứng dụng web; các tấn công lợi dụng lỗi ứng dụng có thể gây ra như đã nói ở phần trên.

1.6 Phân Loại IDS.

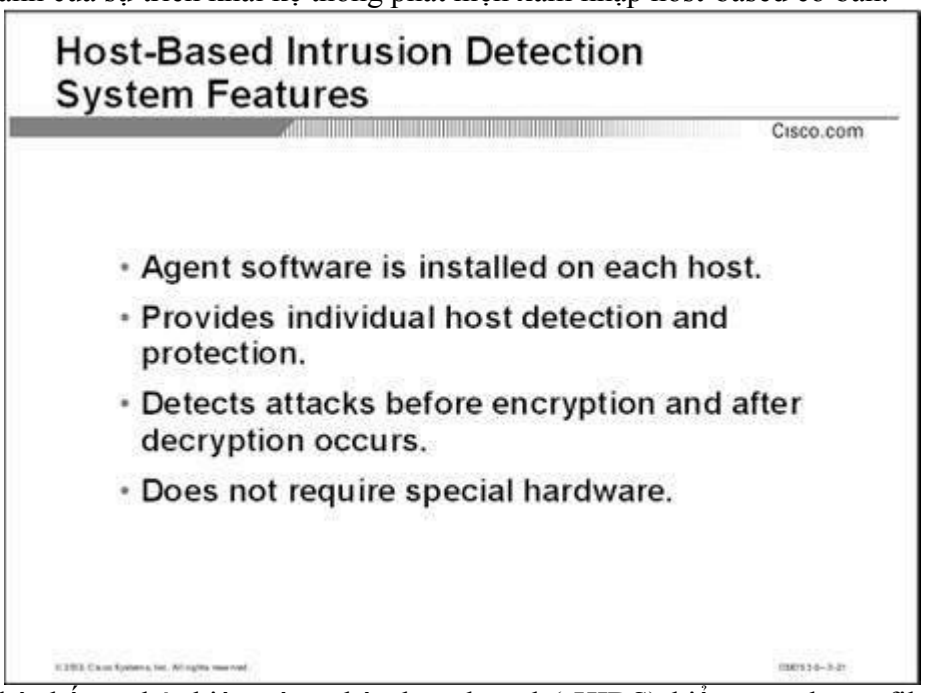
Có 2 loại IDS là Network Based IDS(NIDS) và Host Based IDS (HIDS):

1.6.1 Hệ thống phát hiện xâm nhập Host-Based(Host-based IDS).

- Host-based IDS kiểm tra sự xâm nhập bằng cách kiểm tra thông tin ở host hay mức hệ điều hành. Hệ thống IDS này kiểm tra nhiều diện mạo host của bạn, như hệ thống những cuộc gọi(system call), bản ghi kiểm toán(audit log), thông điệp lỗi(error message),…Hình dưới đây minh họa cho một sự miệu tả host-based IDS tiêu biểu.



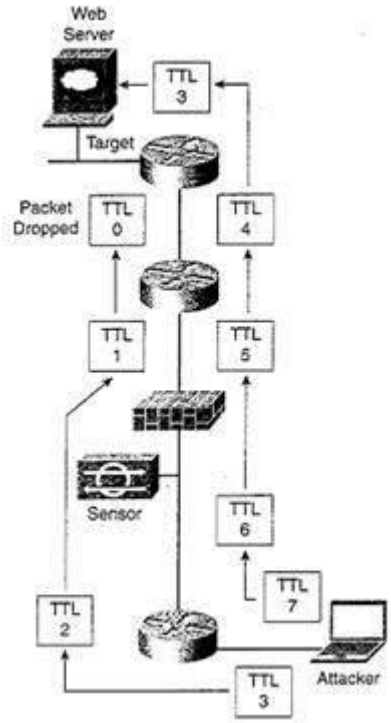
- Mục này miêu tả những đặc điểm của hệ thống phát hiện Host-Based bao gồm một hình ảnh của sự triển khai kệ thống phát hiện xâm nhập host-based cơ bản.



- Một hệ thống phát hiện xâm nhập host-based (HIDS) kiểm tra những file log vào host,những hệ thống và tài nguyên host file. Một sự tiện lợi của hệ thống HIDS là những gì mà nó có thể xem xét tiến trình của hệ điều hành và bảo vệ những tài nguyên hệ thống đặc biệt bao gồm những tập tin mà có thể chỉ tồn tại trên những host đặc biệt.
- Một hình thức đơn giản của HIPS là có khả năng đang nhập vào một host. Tuy nhiên nó có thể trở thành nhân sự đắc lực để chuyển đổi và phân tích những log này. Phần mềm HIPS ngày nay yêu cầu phần mềm Agent phải được cài đặt trên mỗi host để xem xét những hoạt động thực thi trên nó và chông lại những host. Phần mềm Agent thực thi những phân tích và bảo vệ phát hiện xâm nhập vào host.
- **Những thuận lợi:**

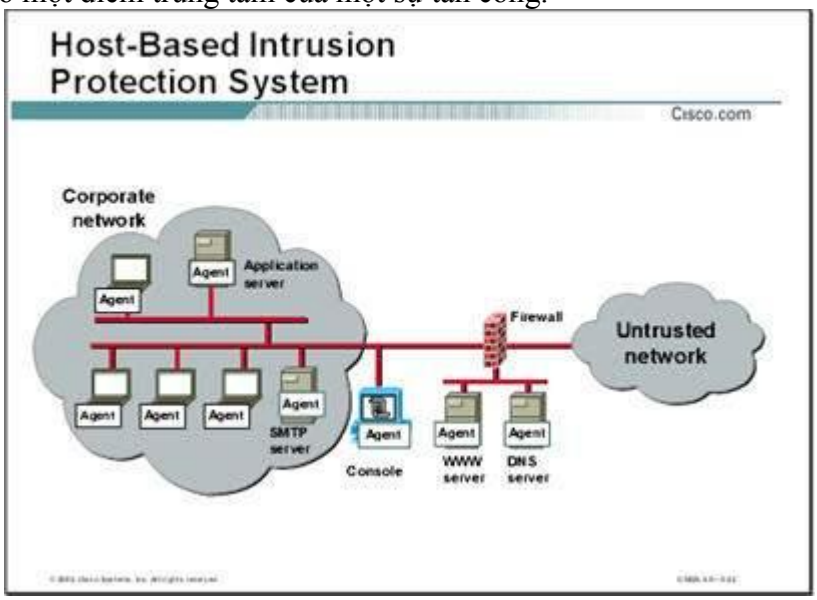
- ❖ Bởi vì một host-based IDS kiểm tra đường đi sau khi nó tiến tới đích(target) của cuộc tấn công(việc thừa nhận host là một đích), nó có thông tin trực tiếp trên sự thành công của những tấn công. Với một networ-based IDS, chuông báo được tạo ra trên những hoạt động xâm nhập biết trước, nhưng chỉ một host-based IDS có thể xác định sự thành công hay thất bại thật sự của những cuộc tấn công.
- ❖ Vấn đề khác như những mảnh vỡ ráp lại và những cuộc tấn công Time-To-Live có thể thay đổi(TTL) thì khó để nhận biết việc ử dụng network-based IDS. Tuy nhiên, một host-based IDS có thể sử dụng cụm IP riêng của host để dễ thỏa thuận với những vấn đề này.

- **Những khó khăn:**
  - ❖ Host-based IDS có một vài trở ngại hay khó khăn: Giới hạn tầm nhìn mạng, Phải xử lí mỗi hệ điều hành trên mạng.
  - ❖ Khó khăn đầu tiên đối với host-based IDS là giới hạn tầm nhìn mạng với sự liên quan tới sự tấn công. Ví dụ,hầu hết hệ thống IDS này không phát hiện những cú quét port chống lại những host. Vì vậy, nó thì cũng không thể làm được với host-based IDS để phát hiện những cú quét dọ thám chống lại mạng của bạn. Những cú quét này cho thấy một đồng hồ chỉ thị cho nhiều tấn công khác chống lại mạng của bạn.
  - ❖ Khó khăn khác của host-based IDS đó là phần mềm phải chạy trên mỗi host của mạng. Điều này miêu tả vấn đề phát triển mới cho những mạng hỗn tạp được soan với một số hệ điều hành. Đôi khi, đại lí host-based IDS có thể chọn để hỗ trợ nhiều hệ điều hành bởi vì những vấn đề hỗ trợ này. Nếu phần mềm host-based IDS của bạn không hỗ trợ tất cả hệ điều hành trên mạng, mạng của bạn không bảo vệ toàn vẹn để chống lại những xâm nhập.



- ❖ Sự khó khăn cuối cùng là khi host-based IDS phát hiện một sự tấn công, nó phải truyền thông tin này tới một vài loại phương tiện quản lí trung tâm. Một sự tấn công có thể lấy những truyền thông ngoại tuyến của host. Khi đó host này không thể truyền thông bất kì thông tin nào đến phương tiện truyền thông

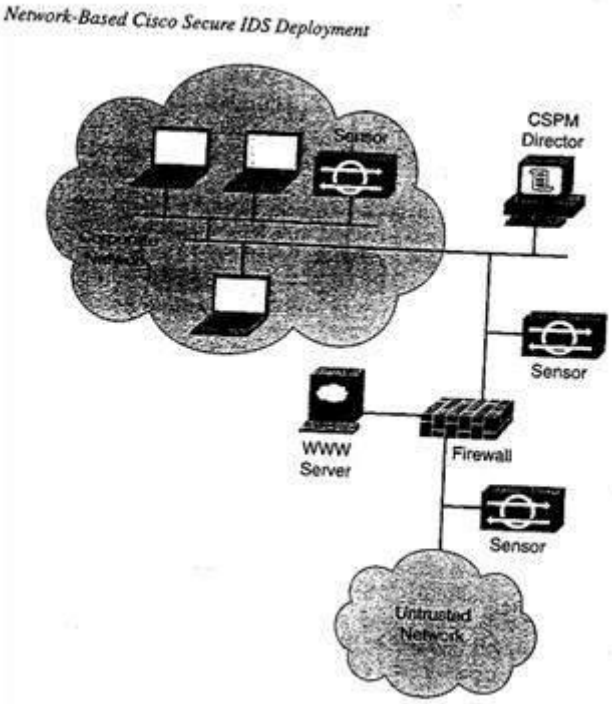
trung tâm. Hơn nữa, đường đi mạng tới sự quản lí trung tâm có thể thực hiện cho nó một điểm trung tâm của một sự tấn công.



Hình này minh họa cho sự trình bày HIPS cơ bản. Agent được cài đặt không chỉ trên những server truy cập công cộng, những tập đoàn mail server, những server ứng dụng, mà còn máy tính cá nhân của người sử dụng. Agent báo cáo những sự kiện tới một server điều khiển trung tâm đặt bên cạnh tập đoàn firewall.

1.6.2 Hệ thống phát hiện xâm nhập Network-Based(Network-based IDS).

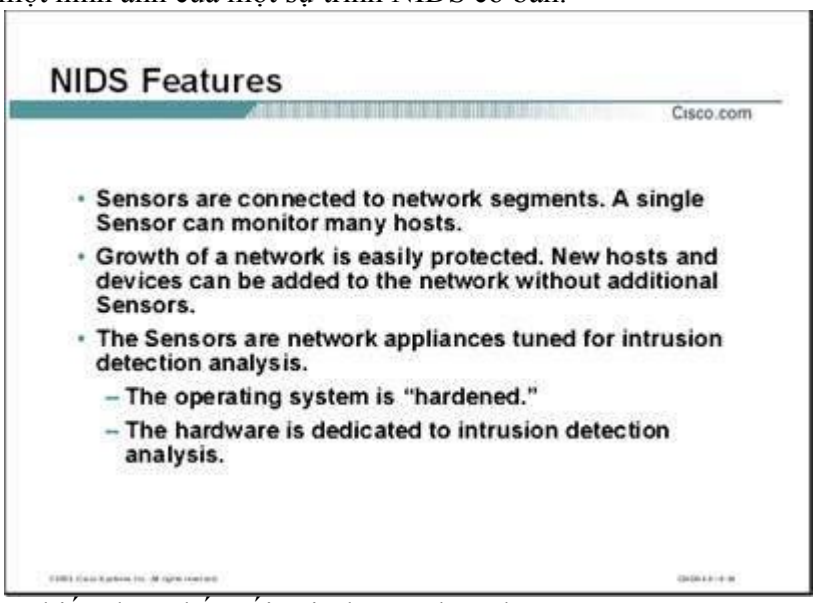
- Một network-based IDS kiểm tra những gói dữ liệu tới những sự tấn công định vị chống lại mạng. IDS đánh hơi(sniff) những gói mạng và so sánh đường đi chống lại những signature cho những hoạt động xâm nhập.



- Hệ thống phát hiện xâm nhập bảo mật của Cisco(CSIDS) là một network-based IDS. Bằng việc sử dụng signature, CSIDS quan tâm đến mỗi gói đi vào mạng và tạo ra chuông báo khi những sự xâm nhập được phát hiện. Bạn có thể cấu hình CSIDS để

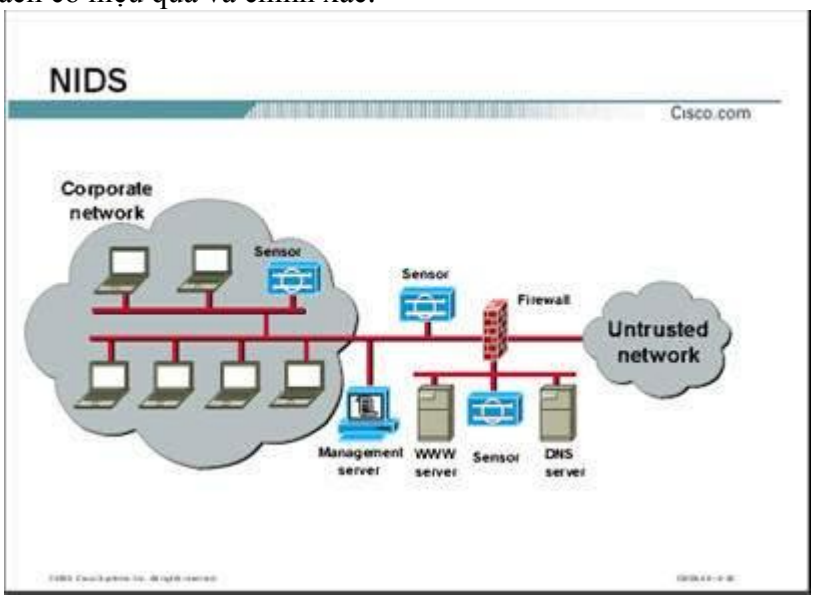
không cho những signature và những chỉnh sửa thông số signature vào làm việc một cách tốt nhất trong môi trường mạng của bạn. Hình trên cho thấy sự phát triển của CSIDS.

- Mục này miêu tả đặc điểm của hệ thống phát hiện xâm nhập Network-Based (NIDSs), bao gồm một hình ảnh của một sự trình NIDS cơ bản.



- Những cảm biến được kết nối tới những phân đoạn mạng. Một sensor đơn lẻ có thể kiểm xét nhiều host.
- Sự phát triển của một mạng được bảo vệ một cách dễ dàng. Những host và dịch vụ mới có thể được thêm vào mạng mà không có những sensor thêm vào.
- Những sensor là những ứng dụng mạng được hòa hợp vào những sự phân tích
  - ❖ Hệ điều hành thì "được làm cứng"
  - ❖ Phần cứng được thiết kế chuyên dụng cho sự phân tích phát hiện xâm nhập.
- Một NIDS bao gồm sự trình bày của những thiết bị kiểm duyệt hay "những sensor" thông qua mạng, mà bắt lại và phân tích lưu lượng khi nó đi ngang qua mạng. Những sensor phát hiện những hoạt động không cho phép và nguy hiểm trong thời gian thực và có thể tham gia hoạt động khi được yêu cầu.
- Những Sensor có thể được trình bày ở một thời điểm mạng được qui định rõ mà có thể là những người quản trị bảo mật để kiểm duyệt những hoạt động mạng trong khi nó đang xảy ra, bất chấp vị trí đích của sự tấn công.
- NIDS cho những nhà quản trị bảo mật nhìn thấy bên trong việc bảo mật thời gian thật của mạng bất chấp sự phát triển của nó. Sự phát triển mạng có thể xảy ra bằng việc thêm vào những host truyền thống hay những mạng mới.Những mạng truyền thống thêm vào sự tồn tại những mạng được bảo vệ sẽ được bao bọc mà không có bất kì sensor mới nào. Những sensor truyền thống có thể dễ dàng được triển khai để bảo vệ những mạng mới. Một vài nhân tố mà bao gồm sự thêm vào những sensor như sau :
  - ❖ Ngoại trừ những công suất lưu lượng – ví dụ , việc thêm vào những phân đoạn gigabit mới đòi hỏi một sensor công suất cao.
  - ❖ Khả năng thực thi của Sensor – những sensor hiện tại có thể không được thi hành việc cho một traffic capacity mới.
  - ❖ Sự bổ sung mạng – Chính sách bảo mật hay thiết kế mạng có thể yêu cầu những sensor truyền thống để giúp việc thúc ép ranh giới bảo mật.
- Những sensor NIDS được chỉnh một cách tiêu biểu cho sự phân tích phát hiện xâm nhập. Hệ điều hành cơ bản là " trần trụi" về những dịch vụ mạng không cần thiết và những dịch vụ chủ yếu được bảo mật.

- Phần cứng được chọn cung cấp sự phân tích phát hiện xâm nhập cực đại có khả năng cho những mạng đa dạng khác nhau. Phần cứng bao gồm những phần sau đây :
    - ❖ Card giao tiếp mạng (NIC) – NIDS phải có khả năng kết nối vào bất kí mạng nào. Card giao tiếp mạng NIDS chungbao gồm Ethernet, Fast Ethernet, GigEthernet, Token Ring và FDDI.
    - ❖ Bộ xử lí – Thiết bị phát hiện xâm nhập đòi hỏi khả năng của CPU để thực thi sự phân tích giao thức phát hiện xâm nhập và làm khớp mẫu.
    - ❖ Bộ nhớ -- Sự phân tích phát hiện xâm nhập là một bộ nhớ chuyên sâu. Bộ nhớ va chạm với khả năng của một NIDS một cách trực tiếp để phát hiện tấn công một cách có hiệu quả và chính xác.

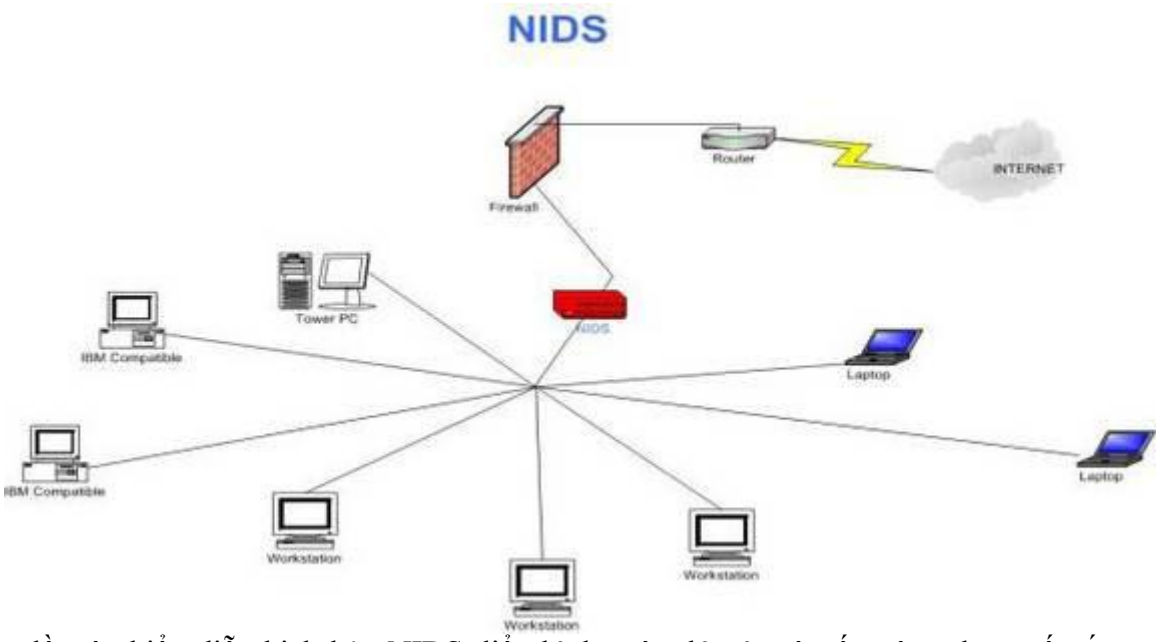


- **Sự thuận lợi:** Một network-based IDS có một vài sự thuận lợi như sau
    - ❖ Hình phối cảnh toàn mạng: Bằng việc thấy đường đi đến đích với nhiều host, một bộ phận cảm biến nhận một mạng mà cân nhắc trong mối liên hệ với những sự tấn công chống lại mạng của bạn.Nếu một ai đó đang quét nhiều host trên mạng của bạn, những thông tin này thì hiển nhiên sẵn sàng vào bộ cảm biến.
    - ❖ Không phải chạy trên mỗi hệ điều hành mạng: Sự thuận lợi khác với network-based IDS đó là không cần chạy trên mỗi hệ điều hành của mạng. Một network-based IDS chạy trên một số bộ cảm biến giới hạn và những nền tảng của người quản lí. Những nền tảng này có thể được chọn để tiếp xúc với những yêu cầu thực thi đặc biệt. Bên cạnh việc ẩn trên mạng đang bị giám sát, những dịch vụ này có thể dễ dàng được làm cứng để bảo vệ chúng từ những tấn công bởi vì chúng phục vụ một mục đích đặc biệt trên mạng. Ngay cả CSIDS hỗ trợ một bộ cảm biến mà là một lá trong gia đình 6000 chất xúc tác(xem chương 14.”catalyst 6000 IDS Module Configuration”).
- **Những khó khăn:** Một network-based IDS đối diện một vài khó khăn sau
    - ❖ Băng thông-Bandwidth: Khó khăn lớn nhất đối với network-based IDS là băng thông.Như những ông dẫn mạng phát triển ngày càng lớn, nó thì khó để giám sát thành công tất cả đường đi thông qua mạng ở một thời điểm đơn lẻ trong thời gian thực, mà không bỏ sót những packet. Thay vì bạn cần cài đặt nhiều sensor một cách thông thường thông qua mạng ở những vị trí mà những sensor có thể giữ băng thông đường đi.
    - ❖ Những mảnh vỡ ráp- Fragment reassembly: Những gói mạng có kích thước cực đại. Nếu một kết nối cần gửi dữ liệu mà vượt quá giới hạn cực đại này, dữ

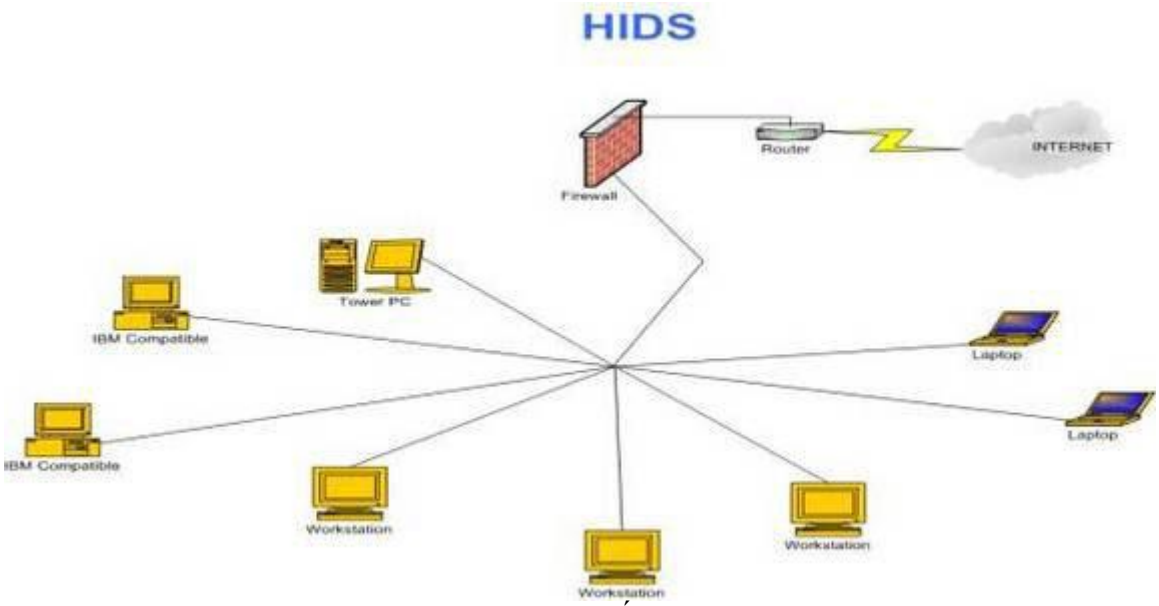
liệu phải được gửi trong nhiều gói. Điều này đượ xem như là fragmentation. Khi việc nhận host lấy những gói fragmantation, nó phải tập hợp lại dữ liệu lại. Không phải tất cả host thi hành những tiến trình một cách tập hợp trong bậc(order) giống nhau. Một vài hệ điều hành bắt đầu với fragment cuối cùng và làm việc thông qua cái đầu tiên. Những cái khác bắt đầu ở cái đầu tiên và làm việc thông qua cái cuối cùng. Order không làm sự kiện nếu những fragment không chồng lên nhau. Nếu chúng đè lên nhau, những kết quả khác nhau cho mỗi tiến trình không tập hợp với nhau. Để kiểm tra những gói fragmentation, một sensor mạng cũng phải tập hợp những fragment lại. Vấn đề bao gồm việc chọn những order đúng một cách tập hợp. Những kẻ tấn công tấn công trên những fragment lapping để thử phá hỏng hệ thống network-based IDS.

- Sự mã hóa- Encryption: cố gắng bảo vệ sự tách biệt của những kết nối dữ liệu của họ. Khi nhiều mạng và người sử dụng cung cấp sự mã hoá cho những sessor người dùng, những thông tin ẩn có sẵn vào để giảm bớt một sensor network-based IDS.Khi đường mạng được mã hoá, sensor mạng không thể đối chọi với dữ liệu được mã hoá nhằm chống lại cơ sở dữ liệu signature của nó

1.6.3 So sánh HIDS và NIDS.



Sơ đồ trên biểu diễn kịch bản NIDS điển hình, trên đó có một tấn công đang cố gắng tạo đường lưu lượng thông qua thiết bị NIDS trên mạng. Thiết bị màu đỏ biểu thị nơi NIDS được cài đặt.



HIDS là một giải pháp toàn diện hơn và cho thấy sự mạnh mẽ hơn trong các môi trường mạng. Nó không quan tâm đến vị trí các máy tính đặt ở đâu và được bảo vệ mọi lúc. Các máy màu vàng thể hiện nơi HIDS được cài đặt.

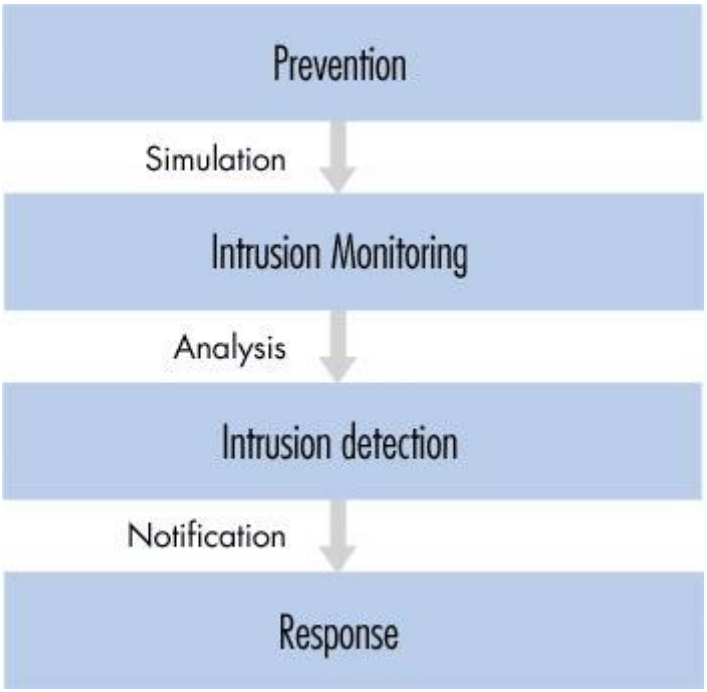
- Phân tích so sánh giữa HIDS và NIDS

Chức năng	HIDS	NIDS	Các đánh giá
Bảo vệ trong mạng LAN	****	****	Cả hai đều bảo vệ bạn khi trong mạng LAN
Bảo vệ ngoài mạng LAN	****	-	Chỉ có HIDS
Dễ dàng cho việc quản trị	****	****	Tương đương như nhau xét về bối cảnh quản trị chung
Tính linh hoạt	****	**	HIDS là hệ thống linh hoạt hơn
Giá thành	***	*	HIDS là hệ thống ưu tiết kiệm hơn nếu chọn đúng sản phẩm
Dễ dàng trong việc bổ sung	****	****	Cả hai tương đương nhau
Đào tạo ngắn hạn cần thiết	****	**	HIDS yêu cầu việc đào tạo ít hơn NIDS
Tổng giá thành	***	**	HIDS tiêu tốn của bạn ít hơn
Băng tần cần yêu cầu trong LAN	0	2	NIDS sử dụng băng tần LAN rộng, còn HIDS thì không
Network overhead	1	2	NIDS cần 2 yêu cầu băng tần mạng đối với bất kỳ mạng LAN nào
Băng tần cần yêu cầu (Internet)	**	**	Cả hai đều cần băng tần Internet để cập nhật kịp thời các file mẫu
Các yêu cầu về cổng mở rộng	-	****	NIDS yêu cầu phải kích hoạt mở rộng cổng để đảm bảo lưu lượng LAN của

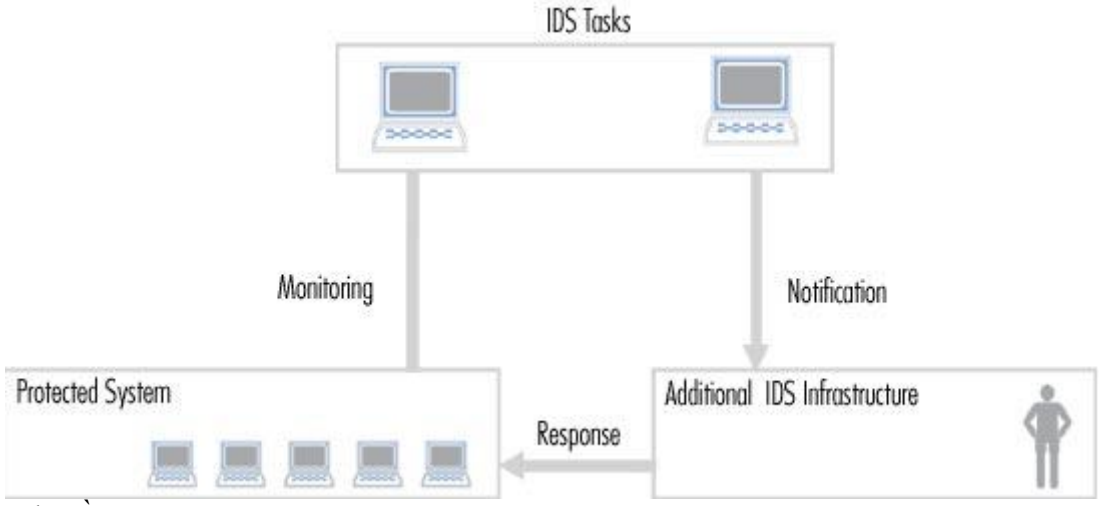
			bạn được quét
Chu kỳ nâng cấp cho các client	****	-	HIDS nâng cấp tất cả các client với một file mẫu trung tâm
Khả năng thích nghi trong các nền ứng dụng	**	****	NIDS có khả năng thích nghi trong các nền ứng dụng hơn
Chế độ quét thanh ghi cục bộ	****	-	Chỉ HIDS mới có thể thực hiện các kiểu quét này
Bản ghi	***	***	Cả hai hệ thống đề có chức năng bản ghi
Chức năng cảnh báo	***	***	Cả hai hệ thống đều có chức năng cảnh báo cho từng cá nhân và quản trị viên
Quét PAN	****	-	Chỉ có HIDS quét các vùng mạng cá nhân của bạn
Loại bỏ gói tin	-	****	Chỉ các tính năng NIDS mới có phương thức này
Kiến thức chuyên môn	***	****	Cần nhiều kiến thức chuyên môn khi cài đặt và sử dụng NIDS đối với toàn bộ vấn đề bảo mật mạng của bạn
Quản lý tập trung	**	***	NIDS có chiếm ưu thế hơn
Khả năng vô hiệu hóa các hệ số rủi ro	*	****	NIDS có hệ số rủi ro nhiều hơn so với HIDS
Khả năng cập nhật	***	***	Rõ ràng khả năng nâng cấp phần mềm là dễ hơn phần cứng. HIDS có thể được nâng cấp thông qua script được tập trung
Các nút phát hiện nhiều đoạn mạng LAN	****	**	HIDS có khả năng phát hiện theo nhiều đoạn mạng toàn diện hơn

1.7 Nhiệm Vụ Của IDS.

- Nhiệm vụ chính của các hệ thông phát hiện xâm phạm là phòng chống cho một hệ thống máy tính bằng cách phát hiện các dấu hiệu tấn công và có thể đẩy lùi nó. Việc phát hiện các tấn công phụ thuộc vào số lượng và kiểu hành động thích hợp. Để ngăn chặn xâm phạm tốt cần phải kết hợp tốt giữa "bả và bẫy" được trang bị cho việc nghiên cứu các mối đe dọa. Việc làm lệnh hướng sự tập trung của kẻ xâm nhập vào tài nguyên được bảo vệ là một nhiệm vụ quan trọng khác. Cả hệ thống thực và hệ thống bẫy cần phải được kiểm tra một cách liên tục. Dữ liệu được tạo ra bằng các hệ thống phát hiện xâm nhập được kiểm tra một cách cẩn thận (đây là nhiệm vụ chính cho mỗi IDS) để phát hiện các dấu hiệu tấn công (sự xâm phạm).



Quá trình của IDS

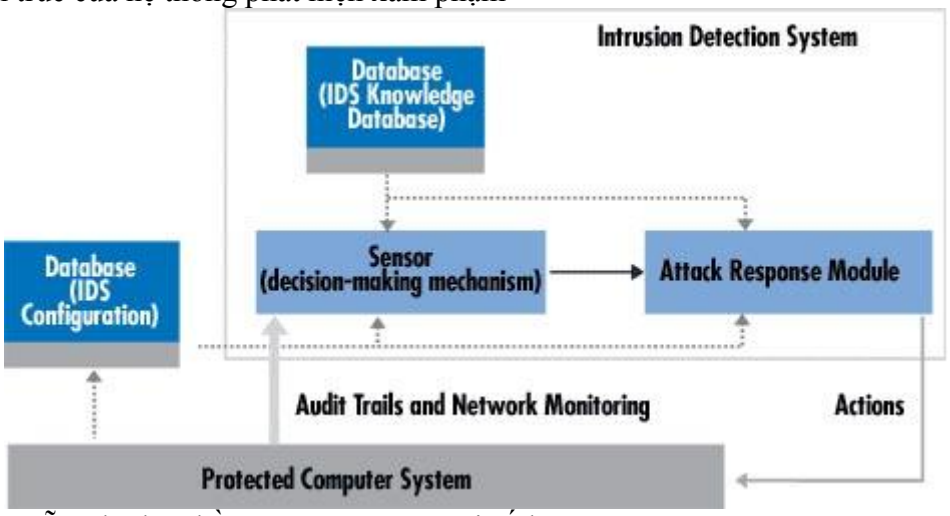


Cơ sở hạ tầng IDS

- Khi một sự xâm nhập được phát hiện, IDS đưa ra các cảnh báo đến các quản trị viên hệ thống về sự việc này. Bước tiếp theo được thực hiện bởi các quản trị viên hoặc có thể là bản thân IDS bằng cách lợi dụng các tham số đo bổ sung (các chức năng khóa để giới hạn các session, backup hệ thống, định tuyến các kết nối đến bẫy hệ thống, cơ sở hạ tầng hợp lệ,…) – theo các chính sách bảo mật của các tổ chức. Một IDS là một thành phần nằm trong chính sách bảo mật.
- Giữa các nhiệm vụ IDS khác nhau, việc nhận ra kẻ xâm nhập là một trong những nhiệm vụ cơ bản. Nó cũng hữu dụng trong việc nghiên cứu mang tính pháp lý các tình tiết và việc cài đặt các bản vá thích hợp để cho phép phát hiện các tấn công trong tương lai nhằm vào các cá nhân cụ thể hoặc tài nguyên hệ thống.
- Phát hiện xâm nhập đôi khi có thể đưa ra các báo cảnh sai, ví dụ những vấn đề xảy ra do trục trặc về giao diện mạng hoặc việc gửi phần mô tả các tấn công hoặc các chữ ký thông qua email.

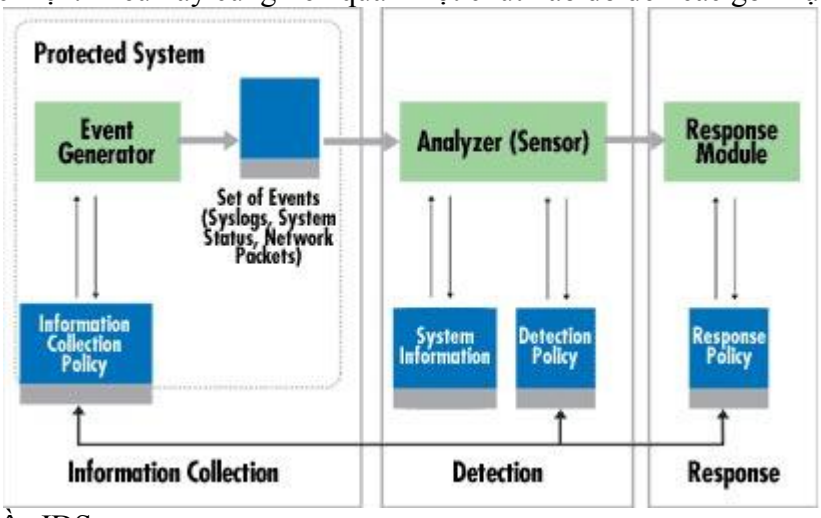
1.8 Kiến trúc IDS.

- Kiến trúc của hệ thống phát hiện xâm phạm



Một IDS mẫu.Thu hẹp bề rộng tương ứng với số lượng
luồng thông tin giữa các thành phần hệ thống

- Bộ cảm biến được tích hợp với thành phần sưu tập dữ liệu – một bộ tạo sự kiện. Cách sưu tập này được xác định bởi chính sách tạo sự kiện để định nghĩa chế độ lọc thông tin sự kiện. Bộ tạo sự kiện (hệ điều hành, mạng, ứng dụng) cung cấp một số chính sách thích hợp cho các sự kiện, có thể là một bản ghi các sự kiện của hệ thống hoặc các gói mạng. Số chính sách này cùng với thông tin chính sách có thể được lưu trong hệ thống được bảo vệ hoặc bên ngoài. Trong trường hợp nào đó, ví dụ, khi luồng dữ liệu sự kiện được truyền tải trực tiếp đến bộ phân tích mà không có sự lưu dữ liệu nào được thực hiện. Điều này cũng liên quan một chút nào đó đến các gói mạng.

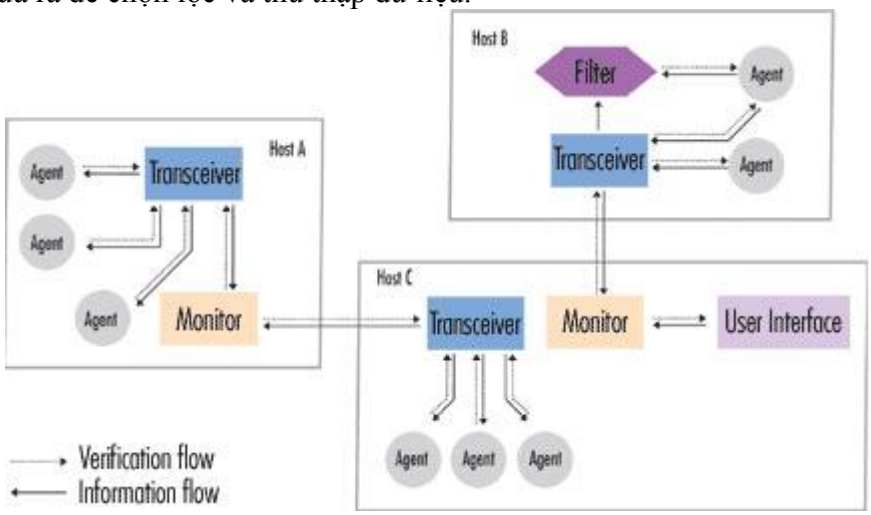


Các thành phần IDS

- Vai trò của bộ cảm biến là dùng để lọc thông tin và loại bỏ dữ liệu không tương thích đạt được từ các sự kiện liên quan với hệ thống bảo vệ, vì vậy có thể phát hiện được các hành động nghi ngờ. Bộ phân tích sử dụng cơ sở dữ liệu chính sách phát hiện cho mục này. Ngoài ra còn có các thành phần: dấu hiệu tấn công, profile hành vi thông thường, các tham số cần thiết (ví dụ: các ngưỡng). Thêm vào đó, cơ sở dữ liệu giữ các tham số cấu hình, gồm có các chế độ truyền thông với module đáp trả. Bộ cảm biến cũng có cơ sở dữ liệu của riêng nó, gồm dữ liệu lưu về các xâm phạm phức tạp tiềm ẩn (tạo ra từ nhiều hành động khác nhau).
- IDS có thể được sắp đặt tập trung (ví dụ như được tích hợp vào trong tường lửa) hoặc phân tán. Một IDS phân tán gồm nhiều IDS khác nhau trên một mạng lớn, tất cả

chúng truyền thông với nhau. Nhiều hệ thống tinh vi đi theo nguyên lý cấu trúc một tác nhân, nơi các module nhỏ được tổ chức trên một host trong mạng được bảo vệ.

- Vai trò của tác nhân là để kiểm tra và lọc tất cả các hành động bên trong vùng được bảo vệ và phụ thuộc vào phương pháp được đưa ra – tạo phân tích bước đầu và thậm chí đảm trách cả hành động đáp trả. Mạng các tác nhân hợp tác báo cáo đến máy chủ phân tích trung tâm là một trong những thành phần quan trọng của IDS. DIDS có thể sử dụng nhiều công cụ phân tích tinh vi hơn, đặc biệt được trang bị sự phát hiện các tấn công phân tán. Các vai trò khác của tác nhân liên quan đến khả năng lưu động và tính roaming của nó trong các vị trí vật lý. Thêm vào đó, các tác nhân có thể đặc biệt dành cho việc phát hiện dấu hiệu tấn công đã biết nào đó. Đây là một hệ số quyết định khi nói đến nghĩa bảo vệ liên quan đến các kiểu tấn công mới. Các giải pháp dựa trên tác nhân IDS cũng sử dụng các cơ chế ít phức tạp hơn cho việc nâng cấp chính sách đáp trả.
- Giải pháp kiến trúc đa tác nhân được đưa ra năm 1994 là AAFID (các tác nhân tự trị cho việc phát hiện xâm phạm). Nó sử dụng các tác nhân để kiểm tra một khía cạnh nào đó về các hành vi hệ thống ở một thời điểm nào đó. Ví dụ: một tác nhân có thể cho biết một số không bình thường các telnet session bên trong hệ thống nó kiểm tra. Tác nhân có khả năng đưa ra một cảnh báo khi phát hiện một sự kiện khả nghi. Các tác nhân có thể được nhái và thay đổi bên trong các hệ thống khác (tính năng tự trị). Một phần trong các tác nhân, hệ thống có thể có các bộ phận thu phát để kiểm tra tất cả các hành động được kiểm soát bởi các tác nhân ở một host cụ thể nào đó. Các bộ thu nhận luôn luôn gửi các kết quả hoạt động của chúng đến bộ kiểm tra duy nhất. Các bộ kiểm tra nhận thông tin từ các mạng (không chủ từ một host), điều đó có nghĩa là chúng có thể tương quan với thông tin phân tán. Thêm vào đó, một số bộ lọc có thể được đưa ra để chọn lọc và thu thập dữ liệu.



1.9 Kỹ thuật xử lý dữ liệu của IDS.

Phụ thuộc vào kiểu phương pháp được sử dụng để phát hiện xâm nhập, các cơ chế xử lý khác nhau (kỹ thuật) cũng được sử dụng cho dữ liệu đối với một IDS. Dưới đây là một số hệ thống được mô tả vắn tắt:

- **Hệ thống Expert:** hệ thống này làm việc trên một tập các nguyên tắc đã được định nghĩa từ trước để miêu tả các tấn công. Tất cả các sự kiện có liên quan đến bảo mật đều được kết hợp vào cuộc kiểm định và được dịch dưới dạng nguyên tắc if-then-else. Lấy ví dụ Wisdom & Sense và ComputerWatch (được phát triển tại AT&T).
- **Phân tích dấu hiệu** giống như phương pháp hệ thống Expert, phương pháp này dựa trên những hiểu biết về tấn công. Chúng biến đổi sự mô tả về ngữ nghĩa từ của mỗi

tấn công thành định dạng kiểm định thích hợp. Như vậy, dấu hiệu tấn công có thể được tìm thấy trong các bản ghi hoặc đầu vào của luồng dữ liệu theo một cách dễ hiểu. Một kịch bản tấn công có thể được mô tả, ví dụ như một chuỗi sự kiện kiểm định đối với các tấn công hoặc mẫu dữ liệu có thể tìm kiếm đã lấy được trong cuộc kiểm định. Phương pháp này sử dụng các từ tương đương trừu tượng của dữ liệu kiểm định. Sự phát hiện được thực hiện bằng cách sử dụng chuỗi văn bản chung hợp với các cơ chế. Điển hình, nó là một kỹ thuật rất mạnh và thường được sử dụng trong các hệ thống thương mại (ví dụ như Stalker, Real Secure, NetRanger, Emerald eXpert-BSM).

- Phương pháp Colored Petri Nets thường được sử dụng để tổng quát hóa các tấn công từ những hiểu biết cơ bản và để thể hiện các tấn công theo đồ họa. Hệ thống IDIOT của đại học Purdue sử dụng Colored Petri Nets. Với kỹ thuật này, các quản trị viên sẽ dễ dàng hơn trong việc bổ sung thêm dấu hiệu mới. Mặc dù vậy, việc làm cho hợp một dấu hiệu phức tạp với dữ liệu kiểm định là một vấn đề gây tốn nhiều thời gian. Kỹ thuật này không được sử dụng trong các hệ thống thương mại.
- **Phương pháp Colored Petri Nets** thường được sử dụng để tổng quát hóa các tấn công từ những hiểu biết cơ bản và để thể hiện các tấn công theo đồ họa. Hệ thống IDIOT của đại học Purdue sử dụng Colored Petri Nets. Với kỹ thuật này, các quản trị viên sẽ dễ dàng hơn trong việc bổ sung thêm dấu hiệu mới. Mặc dù vậy, việc làm cho hợp một dấu hiệu phức tạp với dữ liệu kiểm định là một vấn đề gây tốn nhiều thời gian. Kỹ thuật này không được sử dụng trong các hệ thống thương mại.
- **Phân tích trạng thái phiên:** một tấn công được miêu tả bằng một tập các mục tiêu và phiên cần được thực hiện bởi một kẻ xâm nhập để gây tổn hại hệ thống. Các phiên được trình bày trong sơ đồ trạng thái phiên.
- **Phương pháp phân tích thống kê:** đây là phương pháp thường được sử dụng. Hành vi người dùng hoặc hệ thống (tập các thuộc tính) được tính theo một số biến thời gian. Ví dụ, các biến như là: đăng nhập người dùng, đăng xuất, số file truy nhập trong một chu kỳ thời gian, hiệu suất sử dụng không gian đĩa, bộ nhớ, CPU,… Chu kỳ nâng cấp có thể thay đổi từ một vài phút đến một tháng. Hệ thống lưu giá trị có nghĩa cho mỗi biến được sử dụng để phát hiện sự vượt quá ngưỡng được định nghĩa từ trước. Ngay cả phương pháp đơn giản này cũng không thể hợp được với mô hình hành vi người dùng điển hình. Các phương pháp dựa vào việc làm tương quan profile người dùng riêng lẻ với các biến nhóm đã được gộp lại cũng ít có hiệu quả. Vì vậy, một mô hình tinh vi hơn về hành vi người dùng đã được phát triển bằng cách sử dụng profile người dùng ngắn hạn hoặc dài hạn. Các profile này thường xuyên được nâng cấp để bắt kịp với thay đổi trong hành vi người dùng. Các phương pháp thống kê thường được sử dụng trong việc bổ sung trong IDS dựa trên profile hành vi người dùng thông thường.
- **Neural Networks** sử dụng các thuật toán đang được nghiên cứu của chúng để nghiên cứu về mối quan hệ giữa các vector đầu vào - đầu ra và tổng quát hóa chúng để rút ra mối quan hệ vào/ra mới. Phương pháp neural network được sử dụng cho phát hiện xâm nhập, mục đích chính là để nghiên cứu hành vi của người tham gia vào mạng (người dùng hay kẻ xâm phạm). Thực ra các phương pháp thống kê cũng một phần được coi như neural networks. Sử dụng mạng neural trên thống kê hiện có hoặc tập trung vào các đơn giản để biểu diễn mối quan hệ không tuyến tính giữa các biến và trong việc nghiên cứu các mối quan hệ một cách tự động. Các thực nghiệm đã được tiến hành với sự dự đoán mạng neural về hành vi người dùng. Từ những kết quả cho thấy rằng các hành vi của siêu người dùng UNIX (root) là có thể dự đoán. Với một số ít ngoại lệ, hành vi của hầu hết người dùng khác cũng có thể dự đoán. Neural networks vẫn là một kỹ thuật tính toán mạnh và không được sử dụng rộng rãi trong cộng đồng phát hiện xâm nhập.

- **Phân biệt ý định người dùng:** Kỹ thuật này mô hình hóa các hành vi thông thường của người dùng bằng một tập nhiệm vụ mức cao mà họ có thể thực hiện được trên hệ thống (liên quan đến chức năng người dùng). Các nhiệm vụ đó thường cần đến một số hoạt động được điều chỉnh sao cho hợp với dữ liệu kiểm định thích hợp. Bộ phân tích giữ một tập hợp nhiệm vụ có thể chấp nhận cho mỗi người dùng. Bất cứ khi nào một sự không hợp lệ được phát hiện thì một cảnh báo sẽ được sinh ra.
- **Computer immunology Analogies** với sự nghiên cứu miễn dịch được chủ định để phát triển các kỹ thuật được xây dựng từ mô hình hành vi thông thường trong các dịch vụ mạng UNIX hơn là người dùng riêng lẻ. Mô hình này gồm có các chuỗi ngắn cuộc gọi hệ thống được tạo thành bởi các quá trình. Các tấn công khai thác lỗ hổng trong mã ứng dụng rất có khả năng gây ra đường dẫn thực thi không bình thường. Đầu tiên, một tập dữ liệu kiểm định tham chiếu được sưu tập để trình bày hành vi hợp lệ của các dịch vụ, sau đó kiến thức cơ bản được bổ sung thêm với tất cả các chuỗi được biết rõ về cuộc gọi hệ thống. Các mẫu đó sau đó được sử dụng cho việc kiểm tra liên tục các cuộc gọi hệ thống, để xem chuỗi được tạo ra đã được liệt kê trong cơ sở kiến thức chưa; nếu không, một báo cảnh sẽ được tạo ra. Kỹ thuật này có tỉ lệ báo cảnh sai rất thấp. Trở ngại của nó là sự bất lực trong việc phát hiện lỗi trong cấu hình dịch vụ mạng.
- **Machine learning** (kỹ thuật tự học). Đây là một kỹ thuật thông minh nhân tạo, nó lưu luồng lệnh đầu ra người dùng vào các biểu mẫu vector và sử dụng như một tham chiếu của profile hành vi người dùng thông thường. Các profile sau đó được nhóm vào trong một thư viện lệnh người dùng có các thành phần chung nào đó.

2 IPS.

2.1 Khái niệm.

- Một hệ thống chống xâm nhập (Intrusion Prevention System –IPS) được định nghĩa là một phần mềm hoặc một thiết bị chuyên dụng có khả năng phát hiện xâm nhập và có thể ngăn chặn các nguy cơ gây mất an ninh. IDS và IPS có rất nhiều điểm chung, do đó hệ thống IDS và IPS có thể được gọi chung là IDP-Intrusion Detection and Prevention.
- IPS ra đời khi nào, tại sao lại cần IPS chứ không phải là IDS?
    - ❖ Trước các hạn chế của hệ thống IDS, nhất là sau khi xuất hiện các cuộc tấn công ồ ạt trên quy mô lớn như Code Red, NIMDA, SQL Slammer, một vấn đề được đặt ra là làm sao có thể tự động ngăn chặn được các tấn công chứ không chỉ đưa ra các cảnh báo nhằm giảm thiểu công việc của người quản trị hệ thống. Hệ thống IPS được ra đời vào năm 2003 và ngay sau đó, năm 2004 nó được phổ biến rộng rãi.
    - ❖ Kết hợp với việc nâng cấp các thành phần quản trị, hệ thống IPS xuất hiện đã dần thay thế cho IDS bởi nó giảm bớt được các yêu cầu tác động của con người trong việc đáp trả lại các nguy cơ phát hiện được, cũng như giảm bớt được phần nào gánh nặng của việc vận hành. Hơn nữa trong một số trường hợp đặc biệt, một IPS có thể hoạt động như một IDS bằng việc ngắt bỏ tính năng ngăn chặn xâm nhập. Ngày nay các hệ thống mạng đều hướng tới sử dụng các giải pháp IPS thay vì hệ thống IDS cũ.
- Nhìn bề ngoài, các giải pháp phát hiện xâm nhập và ngăn ngừa xâm nhập xuất hiện theo kiểu cạnh tranh nhau. Rốt cuộc, chúng chia sẻ một danh sách các chức năng giống nhau như kiểm tra gói tin, phân tích có trạng thái, ráp lại các đoạn, ráp lại các TCP-segment, kiểm tra gói tin sâu, xác nhận tính hợp lệ giao thức và thích ứng chữ ký. Một IPS hoạt động giống như một người bảo vệ gác cổng cho một khu dân cư,

cho phép và từ chối truy nhập dựa trên cơ sở các uỷ nhiệm và tập quy tắc nội quy nào đó. Một IDS (hệ thống phát hiện xâm nhập) làm việc giống như một xe tuần tra bên trong khu dân cư, giám sát các hoạt động và tìm ra những tình huống bất bình thường. Dù mức độ an ninh tại cổng vào khu dân cư mạnh đến mức nào, xe tuần tra vẫn tiếp tục hoạt động trong một hệ thống giám sát và sự cân bằng của chính nó.

2.2 Phát hiện và ngăn ngừa xâm nhập.

2.2.1 Phát hiện xâm nhập.

- Mục đích của "phát hiện xâm nhập" là cung cấp sự giám sát, kiểm tra, tính pháp lý và báo cáo về các hoạt động của mạng. Nó hoạt động trên các gói tin được cho phép thông qua một thiết bị kiểm soát truy nhập. Do những hạn chế về độ tin cậy và những đe doạ bên trong, "Ngăn ngừa Xâm nhập" phải cho phép một số "vùng xám" (gray area) tấn công để tránh các trường hợp báo động giả. Mặt khác, những giải pháp IDS được "nhồi" trí thông minh có sử dụng nhiều kỹ thuật khác nhau để nhận biết những cuộc xâm nhập, những khai thác, lạm dụng bất chính và các cuộc tấn công tiềm tàng. Một IDS có thể thực hiện các hoạt động mà không làm ảnh hưởng đến các kiến trúc tính toán và kết nối mạng.
- Bản chất bị động của IDS nằm ở chỗ cung cấp sức mạnh để chỉ đạo phân tích thông minh các lưu lượng gói tin. Những vị trí IDS này có thể nhận ra :
  - ❖ Các cuộc tấn công quen biết theo đường chữ ký (signature) và các quy tắc.
  - ❖ Những biến thiên trong lưu lượng và phương hướng sử dụng những quy tắc và phân tích thống kê phức tạp.
  - ❖ Những biến đổi mẫu lưu lượng truyền thông có sử dụng phân tích luồng.
  - ❖ Phát hiện hoạt động bất bình thường có sử dụng phân tích độ lệch đường cơ sở (baseline deviation analysis).
  - ❖ Phát hiện hoạt động đáng nghi nhờ phân tích luồng, các kỹ thuật thống kê và phát hiện sự bất bình thường.

2.2.2 Ngăn ngừa xâm nhập.

- Như được đề cập trước đây, các giải pháp "Ngăn ngừa Xâm nhập" nhằm mục đích bảo vệ tài nguyên, dữ liệu và mạng. Chúng sẽ làm giảm bớt những mối đe doạ tấn công bằng việc loại bỏ những lưu lượng mạng có hại hay có ác ý trong khi vẫn cho phép các hoạt động hợp pháp tiếp tục. Mục đích ở đây là một hệ thống hoàn hảo – không có những báo động giả nào làm giảm năng suất người dùng cuối và không có những từ chối sai nào tạo ra rủi ro quá mức bên trong môi trường. Có lẽ một vai trò cốt yếu hơn sẽ là cần thiết để tin tưởng, để thực hiện theo cách mong muốn dưới bất kỳ điều kiện nào. Điều này có nghĩa các giải pháp "Ngăn ngừa Xâm nhập" được đặt vào đúng vị trí để phục vụ với:
  - ❖ Những ứng dụng không mong muốn và những cuộc tấn công "Trojan horse" nhằm vào các mạng và các ứng dụng cá nhân, qua việc sử dụng các nguyên tắc xác định và các danh sách điều khiển truy nhập (access control lists).
  - ❖ Các gói tin tấn công giống như những gói tin từ LAND và WinNuke qua việc sử dụng các bộ lọc gói tốc độ cao.
  - ❖ Sự lạm dụng giao thức và những hành động lảng tránh – những thao tác giao thức mạng giống như Fragroute và những khảo sát lấn TCP (TCP overlap exploits) – thông qua sự ráp lại thông minh.
  - ❖ Các tấn công từ chối dịch vụ (DOS/DDOS) như "lụt" các gói tin SYN và ICMP bởi việc sử dụng các thuật toán lọc dựa trên cơ sở ngưỡng.

- ❖ Sự lạm dụng các ứng dụng và những thao tác giao thức – các cuộc tấn công đã biết và chưa biết chống lại HTTP, FTP, DNS, SMTP .v.v. – qua việc sử dụng những quy tắc giao thức ứng dụng và chữ ký.
- ❖ Những cuộc tấn công quá tải hay lạm dụng ứng dụng bằng việc sử dụng các hữu hạn tiêu thụ tài nguyên dựa trên cơ sở ngưỡng.

- Tất cả các cuộc tấn công và trạng thái dễ bị tấn công cho phép chúng tình cờ xảy ra đều được chứng minh bằng tài liệu. Ngoài ra, những khác thường trong các giao thức truyền thông từ mạng qua lớp ứng dụng không có chỗ cho bất cứ loại lưu lượng hợp pháp nào, làm cho các lỗi trở thành tự chọn lọc trong ngữ cảnh xác định.

2.3 Yêu cầu tương lai của IPS.

- Trong tương lai, một giải pháp cổng an ninh nội tuyến (inline) phải đạt được các mục tiêu này :
  - ❖ Khả năng phát hiện và ngăn chặn tấn công dựa trên cơ sở sử dụng lôgic và vật lý của nhiều công nghệ ép buộc. Rộng hơn, điều này còn bao gồm cả khả năng ngăn ngừa cả hai dạng tấn công đã biết và chưa biết có sử dụng các biện pháp phòng thủ ứng dụng (Application Defenses).
  - ❖ Khả năng cùng nhau hoạt động với cơ sở hạ tầng an ninh được triển khai cho những mục đích hỗ trợ tập hợp dữ liệu, bằng chứng điện tử, giám sát theo dõi và phục tùng điều chỉnh khi cần.
  - ❖ Khả năng không phá vỡ những hoạt động kinh doanh do thiếu tính sẵn sàng, hiệu năng kém, những khẳng định sai hay không có khả năng hoạt động cùng nhau với các cơ sở hạ tầng chứng thực quy định.
  - ❖ Khả năng hỗ trợ các chuyên gia an ninh CNTT trong việc chuyển giao kế hoạch quản lý rủi ro của tổ chức của họ bao gồm chi phí cho thực hiện, hoạt động và những kết quả làm việc từ các cảnh báo và báo cáo từ hệ thống.
- Những thách thức để đạt được mục đích
  - ❖ Hiện thời không có các nghiên cứu của đối tác thứ ba có thể chấp nhận được tính hiệu quả của IPS như là một giải pháp. Sự quảng cáo thổi phồng xung quanh "Ngăn ngừa Xâm nhập" đang làm lẫn lộn giữa những gì công nghệ này có thể cung cấp và những gì nó hứa hẹn.
  - ❖ Cách tiếp cận nhiều lớp cho an ninh CNTT tiếp tục có giá trị trong khi công nghiệp phát triển. Nó không có vẻ là sự di trú ra xa khỏi phòng thủ chiều sâu phân lớp đúng như nó được tổ chức.
  - ❖ Nhiều giải pháp IPS sẽ đòi hỏi những yêu cầu giống IDS để điều chỉnh, giám sát và báo cáo.
- Một cách nhìn thực dụng trong tương lai: Hiện tại không có sản phẩm nào thích hợp cho tất cả có thể làm việc phù hợp với nhu cầu thị trường rộng lớn tại mức mà nó có thể thay thế tường lửa hiện tại, NIDS (Network Intrusion Detection System), các bộ chuyển mạch lớp 7 và các thành phần khác có thể hay không thể trở thành các cổng an ninh nội tuyến của ngày mai. Tuy vậy, nếu một sản phẩm như vậy xuất hiện, nó sẽ phải phù hợp với những mục tiêu được thảo luận trước đây trong tài liệu này, bao gồm cả khả năng "phòng thủ ứng dụng" (Application Defenses). Tiếp theo là gì? Một cuộc cách mạng không phải là cái gì đó có thể đoán trước được và nói chung gồm nhiều bước trong tương lai. Những mối đe doạ trong tương lai mà ngày hôm nay chúng ta chưa biết sẽ điều khiển phương hướng của những giải pháp của chúng ta trong tương lai. Có thể có những mối đe doạ mới và tính dễ bị tổn thương mới được phát hiện tác động đến các khái niệm an ninh "Ngăn ngừa Xâm nhập" của ngày hôm nay theo những cách cơ bản. Nhưng sự phát triển các "Hệ thống Ngăn ngừa Xâm

nhập" phần nhiều giống như sự hoà trộn từng bước một qua thời gian của các khái niệm an ninh khác nhau vào trong một mô hình phòng thủ ứng dụng đích thực.

3 So sánh IDS và IPS.

- Hiện nay, Công nghệ của IDS đã được thay thế bằng các giải pháp IPS. Nếu như hiểu đơn giản, ta có thể xem như IDS chỉ là một cái chuông để cảnh báo cho người quản trị biết những nguy cơ có thể xảy ra tấn công. Dĩ nhiên ta có thể thấy rằng, nó chỉ là một giải pháp giám sát thụ động, tức là chỉ có thể cảnh báo mà thôi, việc thực hiện ngăn chặn các cuộc tấn công vào hệ thống lại hoàn toàn phụ thuộc vào người quản trị. Vì vậy yêu cầu rất cao đối với nhà quản trị trong việc xác định các lưu lượng cần và các lưu lượng có nghi vấn là dấu hiệu của một cuộc tấn công. Và dĩ nhiên công việc này thì lại hết sức khó khăn. Với IPS, người quản trị không những có thể xác định được các lưu lượng khả nghi khi có dấu hiệu tấn công mà còn giảm thiểu được khả năng xác định sai các lưu lượng. Với IPS, các cuộc tấn công sẽ bị loại bỏ ngay khi mới có dấu hiệu và nó hoạt động tuân theo một quy luật do nhà Quản trị định sẵn.
- IDS hiện nay chỉ sử dụng từ một đến 2 cơ chế để phát hiện tấn công. Vì mỗi cuộc tấn công lại có các cơ chế khác nhau của nó (Có thể tham khảo thêm các bài viết về DoS của tui), vì vậy cần có các cơ chế khác nhau để phân biệt. Với IDS, do số lượng cơ chế là ít nên có thể dẫn đến tình trạng không phát hiện ra được các cuộc tấn công với cơ chế không định sẵn, dẫn đến khả năng các cuộc tấn công sẽ thành công, gây ảnh hưởng đến hệ thống. Thêm vào đó, do các cơ chế của IDS là tổng quát, dẫn đến tình trạng báo cáo nhầm, cảnh báo nhầm, làm tốn thời gian và công sức của nhà quản trị. Với IPS thì được xây dựng trên rất nhiều cơ chế tấn công và hoàn toàn có thể tạo mới các cơ chế phù hợp với các dạng thức tấn công mới nên sẽ giảm thiểu được khả năng tấn công của mạng, thêm đó, độ chính xác của IPS là cao hơn so với IDS.
- Nên biết rằng với IDS, việc đáp ứng lại các cuộc tấn công chỉ có thể xuất hiện sau khi gói tin của cuộc tấn công đã đi tới đích, lúc đó việc chống lại tấn công là việc nó gửi các yêu cầu đến các máy của hệ thống để xoá các kết nối đến máy tấn công và máy chủ, hoặc là gửi thông tin thông báo đên tường lửa (Firewall) để tường lửa thực hiện chức năng của nó, tuy nhiên, việc làm này đôi khi lại gây tác động phụ đến hệ thống. Ví dụ như nếu Attacker giả mạo (sniffer) của một đối tác, ISP, hay là khách hàng, để tạo một cuộc tấn công từ chối dịch vụ thì có thể thấy rằng, mặc dù IDS có thể chặn được cuộc tấn công từ chối dịch vụ nhưng nó cũng sẽ Block luôn cả IP của khách hàng, của ISP, của đối tác, như vậy thiệt hại vẫn tồn tại và coi như hiệu ứng phụ của DoS thành công mặc dù cuộc tấn công từ chối dịch vụ thất bại. Nhưng với IPS thì khác nó sẽ phát hiện ngay từ đầu dấu hiệu của cuộc tấn công và sau đó là khoá ngay các lưu lượng mạng này thì mới có khả năng giảm thiểu được các cuộc tấn công.

4 Snort.

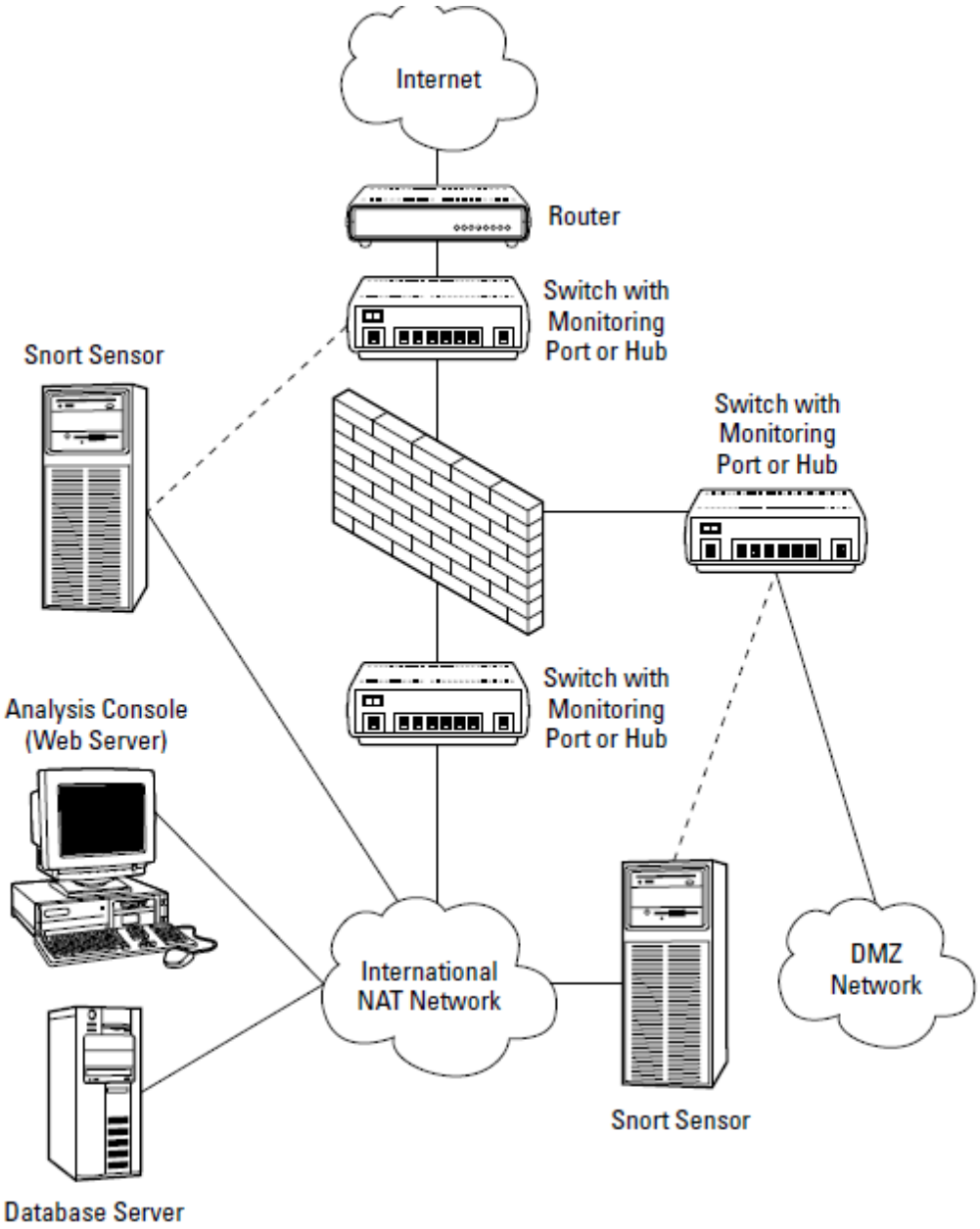
4.1 Giới Thiệu.

- SNORT là một hệ thống mã nguồn mở phát hiện, ngăn chặn xâm nhập trên mạng, có khả năng phân tích lưu thông trên mạng thời gian thực và ghi log các gói tin trên mạng dùng giao thức IP. Hệ thống có thể thực hiện phân tích giao thức, so khớp nội dung và có thể được sử dụng để phát hiện các kiểu tấn công khác nhau, như là: tràn bộ nhớ đệm, quét port, tấn công CGI …Sử dụng SNORT không khó, nhưng hệ thống này có nhiều tùy chọn khi dùng dòng lệnh.
- SNORT có thể được cấu hình để chạy trong các chế độ sau:
  - **Sniffer:** lắng nghe gói tin trên mạng và hiển thị chúng theo một luồng liên tiếp nhau lên màn hình console.
  - **Packet Logger:** ghi log các gói tin và bộ nhớ.
  - **Network Intrusion Detection System (NIDS):** nhận các gói tin từ libpcap/winpcap, phân tích lưu thông trên mạng để so khớp với tập các luật do người dùng định nghĩa và có nhiều hành động tương ứng.
  - **Inline:** nhận các gói tin từ bảng iptables, sau đó làm cho iptables cho phép hay bỏ gói tin dựa vào các luật của SNORT.
- *SNORT được phát triển bởi Sourcefire Inc. Eric Raymond đã phổ biến và sử dụng để mở đường cho Linux thành công trên thị trường hệ điều hành, mọi người trên cộng đồng mã nguồn mở SNORT có thể phát hiện và phản hồi các lỗi và những mối nguy hiểm về bảo mật một cách nhanh hơn và hiệu quả hơn là môi trường mã nguồn đóng.*
- *SNORT sử dụng các rules chứa trong các tập tin dạng văn bản text bình thường và các tập tin này có thể được sửa và bổ sung bởi người dùng. Các rules này được tập hợp lại thành các mục riêng biệt và được chứa trong các tập tin riêng biệt (ví dụ như web-attack.rule, misc.rule…). Các tập tin này sau đó sẽ được khai báo trong tập tin cấu hình, gọi là **SNORT.conf**. SNORT sẽ đọc các rules này lúc khởi động và xây dựng các cấu trúc dữ liệu bên trong nó hoặc kết nối các rules lại để bắt các gói tin.*

4.2 Mô hình hoạt động.

4.2.1 Network Intrusion Detection Systems (NIDS).

- NIDS Thường được đặt trong hệ thống mạng để giám sát các giao dịch giữa các thiết bị. Chúng ta có thể quét tất cả các thông tin vào và ra hệ thống.

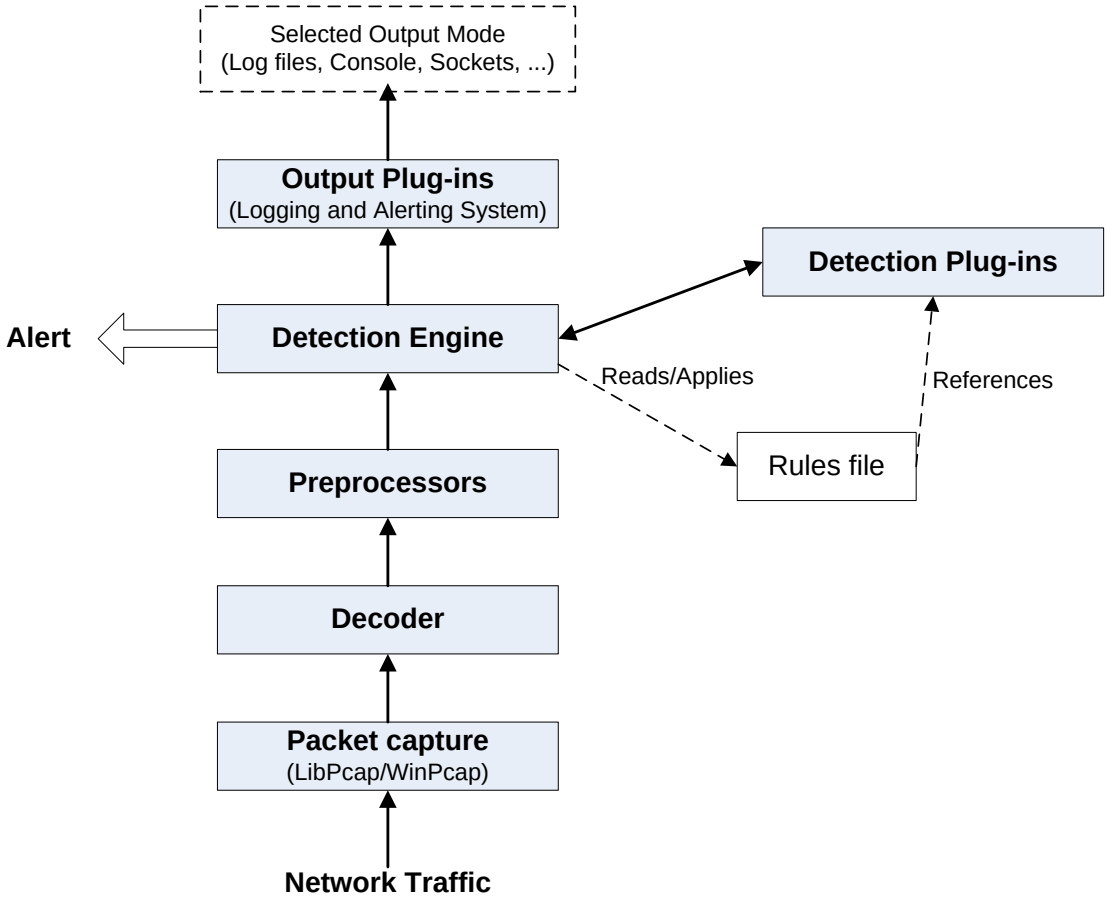


Hình 1: Mô hình triển khai của NIDS (hình được trích từ sách Snort for dummies)

4.2.2 Host Intrusion Detection Systems (HIDS)

- HIDS chạy trên một máy riêng biệt hoặc các thiết bị trên mạng, nhằm phát hiện ra sự tấn công vào chính các thiết bị đó

4.3 Cấu trúc Snort.



SNORT sử dụng pcap để bắt đọc các gói tin trên mạng. Pcap dùng một hàm callback là *ProcessPacket* mỗi khi nó đọc một gói tin. Từ hàm này sẽ gọi đến bộ phân tích gói tin(decoder), sau quá trình phân tích, tùy theo cách cấu hình để SNORT khởi động, nó đi tiếp đến các thành phần trên đó. Dưới đây là mô tả ngắn gọn về các thành phần cơ bản của snort.

4.3.1 Decoder.

Decoder lấy các packet mà Pcap hay LibCap gửi lên chuẩn bị packet đó cho quá trình preprocessor hoặc được gửi thẳng tới detection engine

4.3.2 Preprocessor (Input Plugin).

Preprocessor hay Input plugin có thể hoạt động dưới nhiều mục đích khác nhau

- Nó có thể là nơi để chỉnh hoặc sắp xếp lại dữ liệu trong payload của packet trước khi gửi lên cho dectection engine thực hiện detect, đây là tính năng khá quan trọng giúp chống lại các kỹ thuật mà các hacker thường dùng để qua mặt IDS.
- Nó có thể thực hiện việc detect một packet.
- Nó cũng có thể thực hiện chức năng defragment một gói tin đây cũng là một phần quan trọng trong quá trình detect.

4.3.3 Detection Engine.

- Đây là thành phần quan trọng là trái tim của snort nó thực hiện quá trình detect intrusion với các gói tin dựa trên các rule được định nghĩa. Nó tiến hành đọc các rule được định nghĩa trong file snort.conf và xây dựng một cấu trúc để thực hiện việc detect. Nếu quá trình detect phát hiện packet trùng với một rule nào đó thì một action tương ứng sẽ được tạo ra.

- Quá trình detect có thể thực hiện trên Header của gói tin như IP Header, ICMP Header, TCP Header, … hoặc trong payload của gói tin.

4.3.4 Logging và Alert:

Tùy vào cái gì được Detection engine phát hiện trong packet mà packet có thê được log trước khi active một rule hoặc tạo ra một alert. Đây là nơi tạo ra các message và alert.

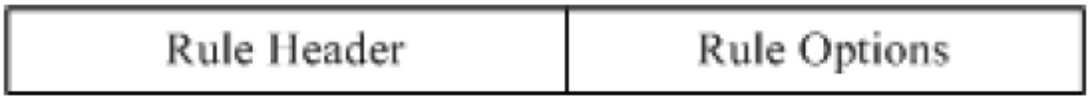
4.3.5 Output Plugin.

Đây là nơi xử lý các output được sinh ra bởi thành phần Logging và Alert, tùy vào việc cấu hình mà nó có thể thực hiện các chức năng sau:

- Loging gói tin vào một file nào đó.
- Loging vào trong cơ sở dữ liệu
- Tạo ra một file output xml
- Điều chỉnh lại cấu hình của tường lửa
- Gửi một SMB message tới một máy window nào đó.
- Ngoài ra còn nhiều thao tác khác như gửi email, SMNP, ….

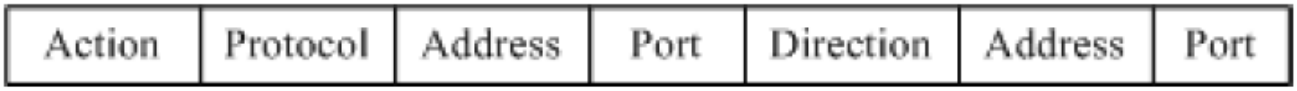
4.4 Cấu Trúc Rule.

Một rule trong Snort được chia thành hai phần rule header và rule options



Hình: Cấu trúc rule (trích từ IDS with snort)

4.4.1 Rule Header.



Hình: Cấu trúc rule Header (trích từ IDS with snort)

- **Action:** đây là trường xác định dạng hành động khi rule được Detection engine so khớp với một gói tin đến. Dưới đây là mô tả các action được định nghĩa sẵn trong snort
  - ❖ Alert: tạo ra một alert message và log lại gói tin
  - ❖ Log: log lại gói tin
  - ❖ Dynamic: luật được thực hiện dựa vào một lệnh khác active nó.
  - ❖ Pass: pass gói tin
  - ❖ Drop: drop một gói tin tạo một alert message và log lại gói tin đó.
  - ❖ Reject: drop gói tin log lại tạo một alert message và gửi thong báo về nguồn.
  - ❖ SDrop: drop một gói tin và không tạo một alert message không log lại gói tin đó không gửi lại thông báo cho nguồn.
- **Protocol:** đây là trường định nghĩa protocol của packet mà rule này được áp dụng, hiện tại snort hộ trợ các loại protocol sau: IP, ICMP, TCP, UDP.
- **Address:** đây là địa chỉ IP nguồn và đích của pakcet mà rule này được áp dụng
- **Port:** đây là port nguồn và đích của packet mà rule này được áp dụng chỉ có giá trị khi protocol là TCP hay UDP.
- **Direction:** dùng để xác định Address và Port nào là nguồn và cái nào là đích, có 3 loại direction như sau: →, ←, <>.
- Ví dụ: *alert icmp any any → any any (msg: "Example"; sid: 1000001)*

4.4.2 Rule option.

Về cơ bản rule option có thể được chia thành các loại như sau: Rule content, IP Option, ICMP Option, TCP Option, Meta Data Option, Miscellaneous Rule Option.

a. *Rule content:*

Đây là option rất mạnh và quan trọng nó cho phép bạn viết những luật dùng để phân tích payload của một gói tin thông qua giá trị binary hay ASCII, ngoài ra chúng ta có thể phối hợp với nhiều option khác để xác định chính xác malilous code trong nội dung của packet.

- ***Content option***: đây là option cho phép bạn định nghĩa một chuỗi ASCII hoặc một chuỗi binary mang những dấu hiệu tấn công thuộc về rule cụ thể tồn tại trong packet.

Ví dụ: `alert tcp any any -> any any (content: "|0101 FFFF|/etc/passwd|E234|"; msg:"Searching for Ascii and Binary stuff!";)` → trong luật này content option vừa chứa giá trị ASCII vừa chứa giái trị binary, giá trị binary được kẹp giữa dấu ||.

- ***Depth option***: đây là option cho phép chỉ rõ số lượng byte tối đa trong payload sẽ được so khớp với content string được định nghĩa trong content option
- ***Offset option***: đây là option định nghĩa vị trí trong payload bắt đầu thực hiện so khớp với content string được định nghĩa trong content option

Ví dụ: `alert tcp any any -> any 80 (content: "GET"; depth: 10; offset: 0 msg:"Searching for Ascii with offset and depth stuff!";)` → trong luật này chỉ rõ chỉ thực hiện so khớp chuỗi 'GET' bắt đầu ở vị trí thứ 0 trong payload của gói tin và thực hiện tìm kiếm tối đa tới vị trí thứ 10 của payload.

- ***Nocase option***: chỉ rõ là khi thực hiện so khớp content string với payload của gói tin thì không cần thực hiện phân biệt chữ hoa hay chữ thường.

b. *Ip options:*

Đây là option thao tác trên nội dung header của gói IP, nó thường được dùng để xác định các cuộc tấn công vào thiết bị, các hành động quét mạng, ….

- ***Equivalent Source and Destination option***: option dùng để detect những gói tin có soucer ip và destination ip trùng nhau

Ví dụ: `alert ip any any -> any any (msg:" Same Source and Destination IP Address"; sameip;)`

- ***IP option***: thực hiện detect trên trường option trong header của gói tin ip, cú pháp *ipopts: <IP_OPTION>;*.

Ip Options	Mô tả tổng quát
Eol	Sử dụng để chỉ sự kết thúc của một ip list
Lsrr	Ip loose soure routing
Nop	Sử dụng để chỉ gói tin có cờ option không được set
Rr	Record route
Satid	Ip stream identifier
Sec	
Ssrr	Ip strict source routing
Ts	Time stamp field

- ***TOS option***: thực hiện detect trên trường TOS trong header của gói tin ip, cú pháp *tos: "value";*.

Ví dụ: `alert tcp $EXTERNAL any -> $CISCO any (msg:" Cisco TOS Example"; tos:!"0";)` → chỉ ra những gói có tos khác 0.

- ***TTL option****:* thực hiện detect trên trường TTL trong header của gói tin ip, cú pháp *ttl: "value";*.
- ***ID option****:* thực hiện detect trên trường ID trong header của gói tin ip, cú pháp *id: "value";*.

c. *Tcp options:*

Cũng giống như ip option, tcp option dùng để detect các giá trị trong trường header của gói tin tcp. Trong tcp options chỉ có 3 options là Seq, ack và flag trong đó seq và ack rất hiếm khi sài ta chủ ý sài option flag.

- ***seq option****:* dùng để chỉ cụ thể một con số trong trường seq ở header của gói tin tcp mà ta muốn detect, cú pháp: *seq: <sequence_number_value>;*
- ***ack option****:* dùng để chỉ cụ thể một con số trong trường ack ở header của gói tin tcp mà ta muốn detect, cú pháp: *ack: <ack_numerical_value>;*
- ***flag option****:* dùng để chỉ ra các cờ được thiết lập hoặc không thiết lập, hoặc sử dụng kết hợp với các cờ khác trong gói tcp mà ta muốn detect, *cú pháp: flags: <TCP_VALUES>;* . Các giá trị hiện nay mà snort hộ trợ là:

TCP Flags	Mô tả
A	Đây là tùy chọn để check cờ ACK có bật
F	Đây là tùy chọn để check cờ FIN có bật
P	Đây là tùy chọn để check cờ PSH có bật
R	Đây là tùy chọn để check cờ RST có bật
U	Đây là tùy chọn để check cờ URG có bật
S	Đây là tùy chọn để check cờ SYN có bật
0	Đây là tùy chọn để kiểm tra packet không có cờ nào được bật
1	
2	
+	Dấu cộng được dung để diễn tả cờ được chỉ rõ đi kèm với bất kỳ một cờ nào khác, ví dụ: A+ nghĩa là luật sẽ được thực hiện nếu gói tin có bật cờ ACK và kèm thêm một cờ bất kỳ.
*	Dấu * dùng để chỉ bất kỳ cờ nào trong packet trùng với cờ được chỉ ra trong luật, ví dụ *AS luật sẽ được thực hiện nếu gói tin có cờ ACK hoặc SYN hoặc cả hai được bật
!	Dấu phủ định dùng để chỉ gói tin không có cờ được chỉ rõ trong luật, ví dụ: !S nghĩa là luật sẽ được thực hiện nếu gói có cờ SYN không được bật.

d. *ICMP options:*

Snort hiện nay hộ trợ 4 icmp options có thể được dùng trong rule options để tạo ra những rule cho việc detect những dấu hiệu tấn công cụ thể. 4 icmp option là *ICMP ID, ICMP SEQUENCE, ICODE, ITYPE*

- ***ID option****:* đây là trường khác với trường id của gói IP, luật này được thực hiện dựa trên giá trị được chỉ rõ trong trường ID của gói icmp, đây là trường còn có thể dùng để xác định các chương trình sử dụng giá trị ICMP ID tĩnh, cú pháp : *icmp_id : VALUE;* .
- ***Sequence option****:* tương tự như trường ID ICMP, cú pháp *icmp_seq :VALUE;*.
- ***icode option****:* cho phép chỉ rõ một giá trị trong trường code của gói tin icmp, trường này có hai cách sử dụng một là chỉ rõ code hợp lệ của gói icmp như vậy luật này sẽ

được thực hiện nếu gói tin icmp nào mà trường code có giá trị trùng với giá trị được định nghĩa, hai là chỉ ra một giá trị code không hợp lệ dùng để detect các gói icmp có giá trị trong trường code không hợp lệ, cú pháp *icode : VALUE;*.

- ***itype option****:* tương tự như trường icode, dùng để kiểm tra giá trị trong trường type của gói icmp, cú pháp *itype: VALUE;*.

e. *Metadata options:*

Đây là những option hộ trợ cho việc phân loại, định danh và tạo tài liệu cho các alert mà snort tạo ra, những option này nên được tạo một cách cẩn thận để hộ trợ quá trình report và cấu hình snort được thuận lợi.

- ***Snort id options****:* đây là options dùng để phân loại, định danh một rule cụ thể, cú pháp sử dụng là *sid:VALUE. Bảng dưới đây liệt kê các snort id.*

Tầm giá trị	Ý nghĩa
< 100	Dùng cho mục đích lưu trữ
100 – 1000000	Dùng cho snort phân phối các tập luật của mình
> 1000000	Dùng cho người dùng định nghĩa các luật của mình

- ***Rule revision number****:* đây là trường dùng trong trường hợp bạn thay đổi một luật nào đó và muốn phân biệt với các lần thay đổi khác hoặc với luật gốc, cú pháp *rev : REVISION_NUMBER;* . Ví dụ dưới đây cho thấy một luật có sid là 10000001 và rev là 3

```
Alert tcp any any -> any any (sid: 10000001; rev:3; msg: "sid and revision")
```

- ***Serverity Identifier options****:* đây là option cho phép ghi đè lên giá trị priority mặc định của rule được thiết lập bởi rule classtification, ở đây chúng ta có thể tăng hoặc giảm prioprity bằng cách sử dụng cú pháp như sau *priority: <PRIORITY_VALUE>* ví dụ: `alert udp any any -> $INTERNAL 21974 (priority:1; msg: "Bad Worm Backdoor";)`
- ***Classification Identifier options****:* đây là options cho phép bạn phân loại các rule dựa trên những những dạng tấn công khác nhau, các classtification này sẽ được định nghĩa trong một file config, cú pháp *classtype: <NAME_OF_CLASSTIFICATION>*, dưới đây là một số classtification đã được snort định nghĩa sẵn:

Table 5.5 Critical Classifications (Priority 1)

Classtype	Brief Description
attempted-admin	Attempted administrator privilege gain
attempted-user	Attempted user privilege gain
shellcode-detect	Executable code was detected
successful-admin	Successful administrator privilege gain
successful-user	Successful user privilege gain
trojan-activity	A network Trojan was detected
unsuccessful-user	Unsuccessful user privilege gain
web-application-attack	Web application attack

Table 5.6 Intermediate Classifications (Priority 2)

Classtype	Brief Description
attempted-dos	Attempted DoS
attempted-recon	Attempted information leak
bad-unknown	Potentially bad traffic
denial-of-service	Detection of DoS attack
misc-attack	Miscellaneous attack
non-standard-protocol	Detection of a nonstandard protocol or event
rpc-portmap-decode	Decode of an RPD query
successful-dos	Denial of service
successful-recon-largescale	Large-scale information leak
successful-recon-limited	Information leak
suspicious-filename-detect	A suspicious filename was detected
suspicious-login	An attempted login using a suspicious user-name was detected
system-call-detect	A system call was detected
unusual-client-port-connection	A client was using an unusual port
web-application-activity	Access to a potentially vulnerable Web appli-cation

các hình trên được trích từ sách snort 2.1

- ***External reference****:* đây là option dùng để thêm các tham chiếu tới các tài liệu cho một alert khi nó được tạo ra bởi snort phục vụ cho mục đích báo cáo, sắp xếp và làm tư liệu về các alert của snort, cú pháp: *reference: <SYSTEM>,<ID VALUE>*.

4.5 Cài Đặt.

Chúng ta cần chuẩn bị tất cả những phần mềm sau để tiến hành cài đặt snort:

- snort-2.4.2.tar.gz
- snortrules.tar.gz
- snort-1.0.wbm
- Net_SSLeay.pm-1.2.0.tar.gz
- webmin-1.230-1.noarch.rpm
- acid-0.9.6b23.tar.gz
- adodbb461.tar.gz
- gd-2.0.33.tar.gz
- phplot-4.4.6.tar.gz

4.5.1 Cài đặt snort.

Cài đặt snort bằng gói snort-2.4.2.tar.gz, với các lệnh sau:

```
# cp snort-2.4.2.tar.gz /usr/
# cd /usr/
# tar –xzvf snort-2.4.2.tar.gz
# cd snort-2.4.2
# ./configure --with-mysql
# make
```

```
# make install
```

Cập nhật tập luật cho snort, snort có thể hoạt động ở mode **NIDS** là nhờ vào tập luật này.

```
# mkdir /etc/snort
# cp snortrules.tar.gz /etc/snort
# cd /etc/snort
# tar –xzvf snortrules.tar.gz
```

Vì các đường dẫn đã được cấu hình mặc định trong snort, để snort có thể chạy đúng, chuẩn xác cần di chuyển tất cả các luật từ thư mục: **/etc/snort/snortrules** đến thư mục **/etc/snort**. Xóa thư mục **/etc/snort/snortrules**.
Sửa file **/etc/snort/snort.conf** ở những dòng sau: (815)

```
# output database: log, mysql, user=root password=test dbname=db host=localhost
sửa thành
  output database: log, mysql, user=snort password=123456 dbname=snort
host=000.000.000.000
```

```
var RULE_PATH ../rules
sửa thành
#var RULE_PATH ../rules
```

Bỏ tất cả "**$RULE_PATH**" ở những dòng include như sau: (73)

```
# include $RULE_PATH/bad-traffic.rules
……
sửa thành
#include bad-traffic.rules
……
```

Tạo thư mục để ghi lại các sự kiện log

```
# mkdir /var/log/snort
```

Tạo file để khởi động snort và thử start, stop snort

```
# cp snortd /etc/rc.d/init.d
# cd /etc/rc.d/init.d
# chmod 755 snortd
# chkconfig –level 2345 snortd on
#. /etc/rc.d/init.d/snortd start
#./etc/rc.d/init.d/snortd start
```

4.5.2 Cài đặt Webmin.

Trước khi cài đặt webmin, chúng ta phải cài đặt SSL để đảm bảo dữ liệu được truyền bảo mật). Theo các câu lệnh sau để cài đặt SSL

```
# cp Net_SSLeay.pm-1.20.tar.gz /usr
# cd /usr/Net_SSLeay.pm-1.20.tar.gz
# tar –xzvf Net_SSLeay.pm-1.20.tar.gz
# cd Net_SSLeay.pm-1.20
#perl Makefile.PL
#make install
```

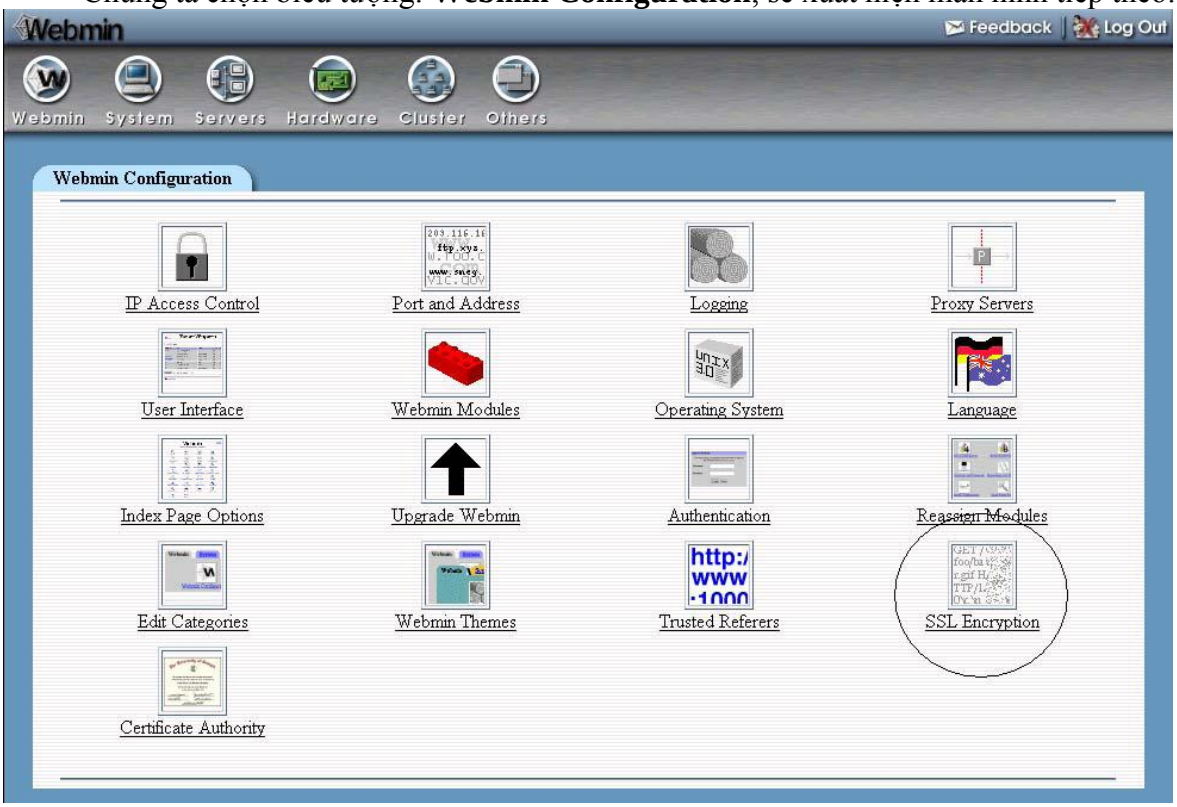
Cài đặt webmin bằng gói rpm

```
# rpm –ivh webmin-1.230-1.noarch.rpm
```

Sau khi cài đặt webmin, chúng ta tiến hành cấu hình SSL. Mở trình duyệt Mozilla. gõ vào địa chỉ: **http://localhost:10000**. Sau đó login vào với quyền ROOT, chúng ta sẽ thấy xuất hiện màn hình sau:

Hình 4-1: Webmin Configuration

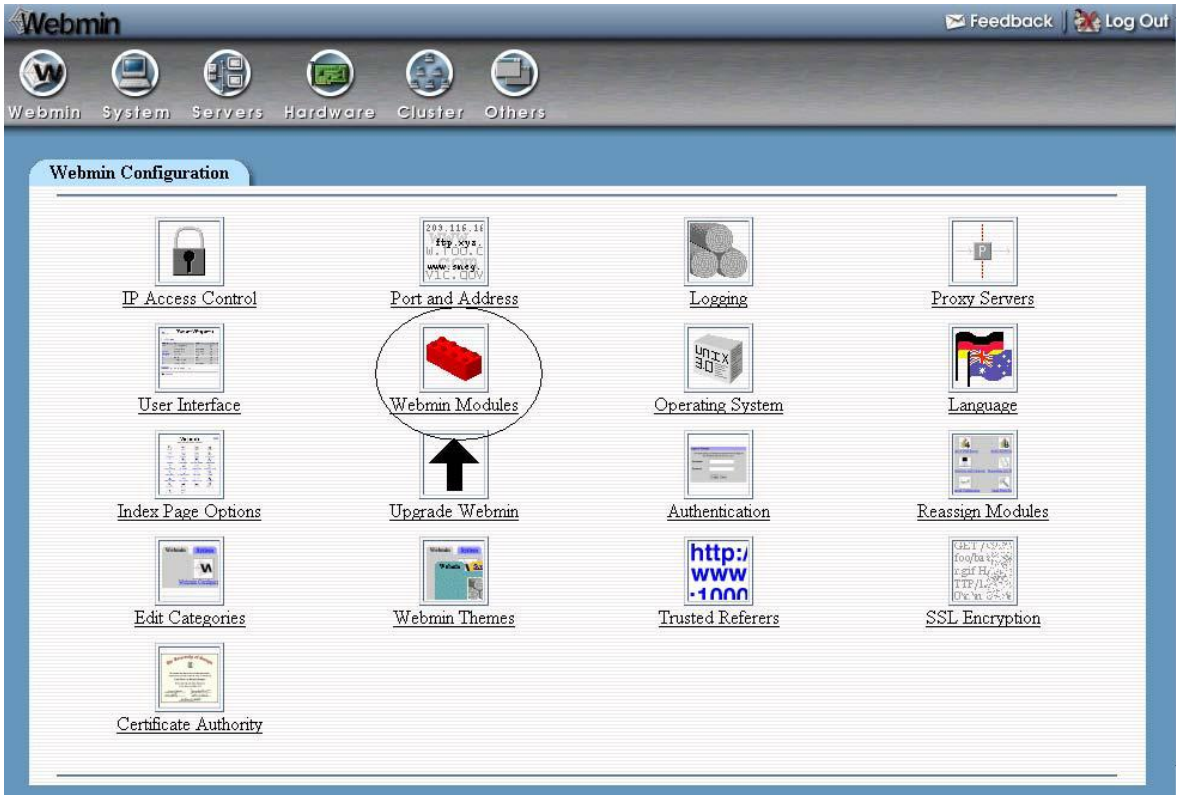
Chúng ta chọn biểu tượng: **Webmin Configuration**, sẽ xuất hiện màn hình tiếp theo:



Hình 4-2: SSL encrytion

Chúng ta sẽ chọn chức năng **SSL Encryption**, sau đó chọn tiếp tùy chọn: "**Enable SSL support if available**" để kích hoạt chức năng của SSL. Từ lúc này, chúng ta sẽ phải log vào trang này ở địa chỉ: **https://localhost:10000**.

Trở lại với hình 2.1.Webmin Configuration, chúng ta chọn chức năng **Webmin Modules** để cài đặt webmin cho snort.



Hình 4-3: Webmin Modules

Tiếp theo chọn cài đặt với local file, chọn đường dẫn đến **snort-1.0.wbm** để tiến hành cài đặt.

4.5.3 Cài đặt adodb, acid, gd, phplot.

Trước khi cài đặt các gói adodb, acid, gd, phplot, chúng ta phải tiến hành cấu hình mysql theo các bước như sau:

```
# mysql –u root
mysql> set password for ‘root’@’localhost’=password(‘123456’);
mysql> create database snort;
mysql> exit;
# chkconfig –level mysqld on
# mysql –u root –p
mysql> connect snort;
mysql> source create_mysql;
mysql> grant CREATE,INSERT,SELECT,DELETE,UPDATE on snort.* to snort;
mysql> grant CREATE, INSERT, SELECT, DELETE, UPDATE on snort.* to
snort@localhost;
mysql> grant CREATE,INSERT,SELECT,UPDATE on snort.* to acidviewer;
mysql> grant CREATE, INSERT, SELECT, UPDATE on snort.* to
acidviewer@localhost;
mysql> connect mysql;
mysql> set password for ‘snort’@’localhost’=password(‘123456’);
mysql> set password for ‘snort’@’%’=password(‘123456’);
mysql> set password for ‘acidviewer’@’localhost’=password(‘123456’);
mysql> set password for ‘acidviewer’@’%’=password(‘123456’);
mysql> flush privileges;
mysql> exit;
```

Tiếp theo chúng ta tiến hành cài đặt acid, adodb, gd, phplot

```
# tar –xzvf acid-0.9.6b23.tar.gz –C /var/www/html
# tar –xzvf adodb461.tar.gz –C /var/www/html
# tar –xzvf gd-2.0.33.tar.gz –C /var/www/html
# tar –xzvf phplot-4.4.6.tar.gz –C /var/www/html
```

Đổi tên các thư mục gd-2.0.33 và phplot-4.4.6 thành gd và phplot. Copy thư mục acid thành một thư mục khác là acidviewer.
Sửa file **/var/www/html/acid/acid_conf.php** và file **/var/www/html/acidviewer/ acid_conf.php** ở các dòng sau:

```
$DBlib_path=”../adodb”;
$alert_dbname=”snort”;
$alert_user=”snort”; (hoặc acidviewer)
$alert_password=”123456”;
$Chartlib_path=”../phplot”;
```

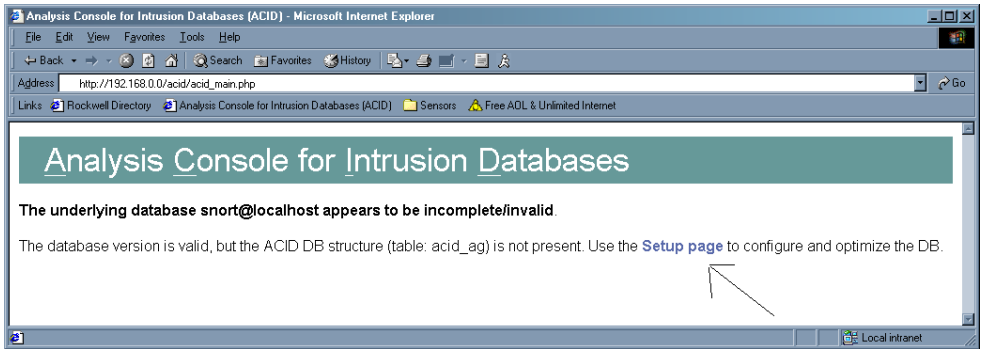
Tiếp tục cấu hình các bước sau:

```
# mkdir /usr/lib/apache
# mkdir /usr/lib/apache/passwords
# htpasswd –c /usr/lib/apache/passwords/passwords snort
# htpasswd /usr/lib/apache/passwords/passwords acidviewer
```

Thêm đoạn sau vào file /etc/httpd/conf/httpd.conf

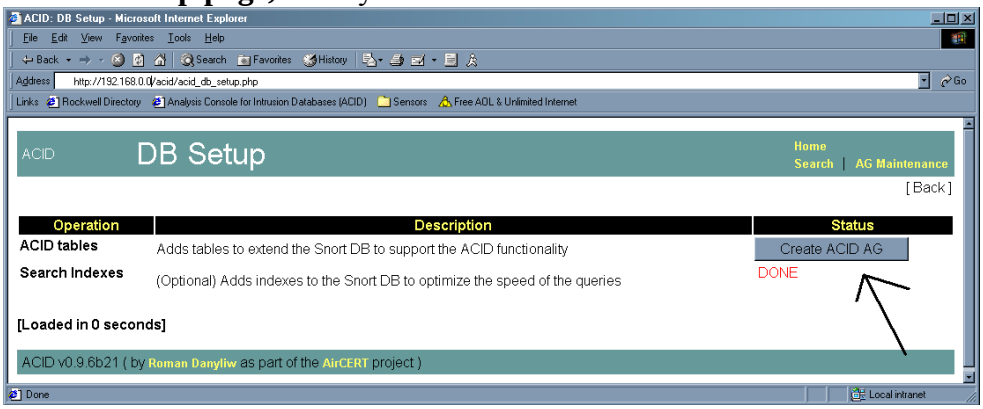
```
<Directory “/var/www/html/acid”>
        AuthType Basic
        AuthName “snort solution”
        AuthUserFile /usr/lib/apache/passwords/passwords
        Require user snort
        AllowOverride None
</Directory>

<Directory “/var/www/html/acid”>
        AuthType Basic
        AuthName “snort solution”
        AuthUserFile /usr/lib/apache/passwords/passwords
        Require user acidviewer
        AllowOverride None
</Directory>
```

Bây giờ chúng ta truy cập vào trang acid thông qua địa chỉ: **http://localhost/acid/,** sẽ thấy màn hình sau:



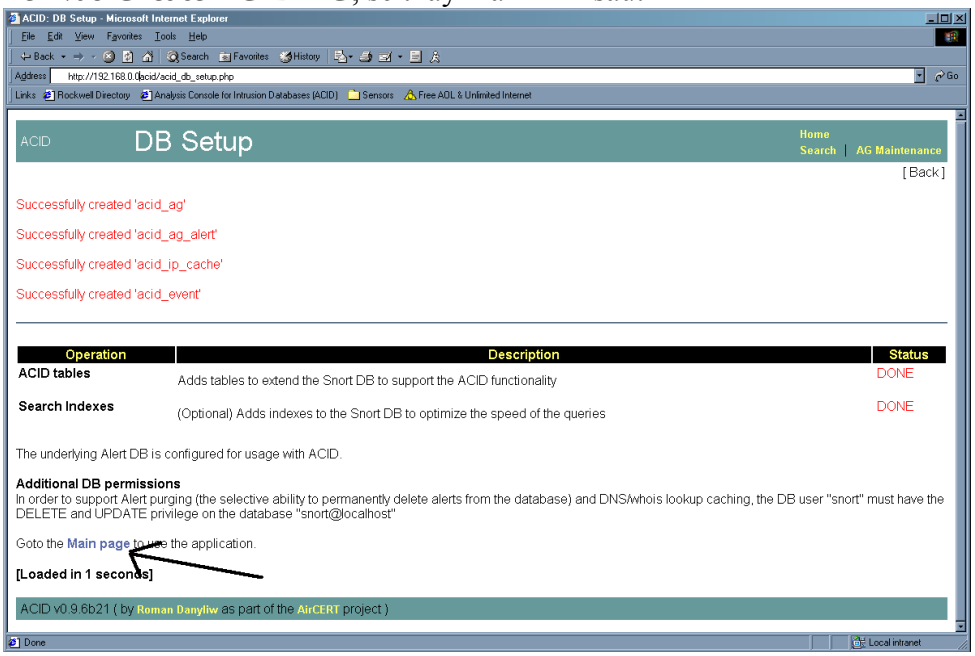
Hình 4-4: ACID setup

Click vào **Setup page,** sẽ thấy màn hình sau:



Hình 4-5: DB setup

Click vào **Create ACID AG**, sẽ thấy màn hình sau:



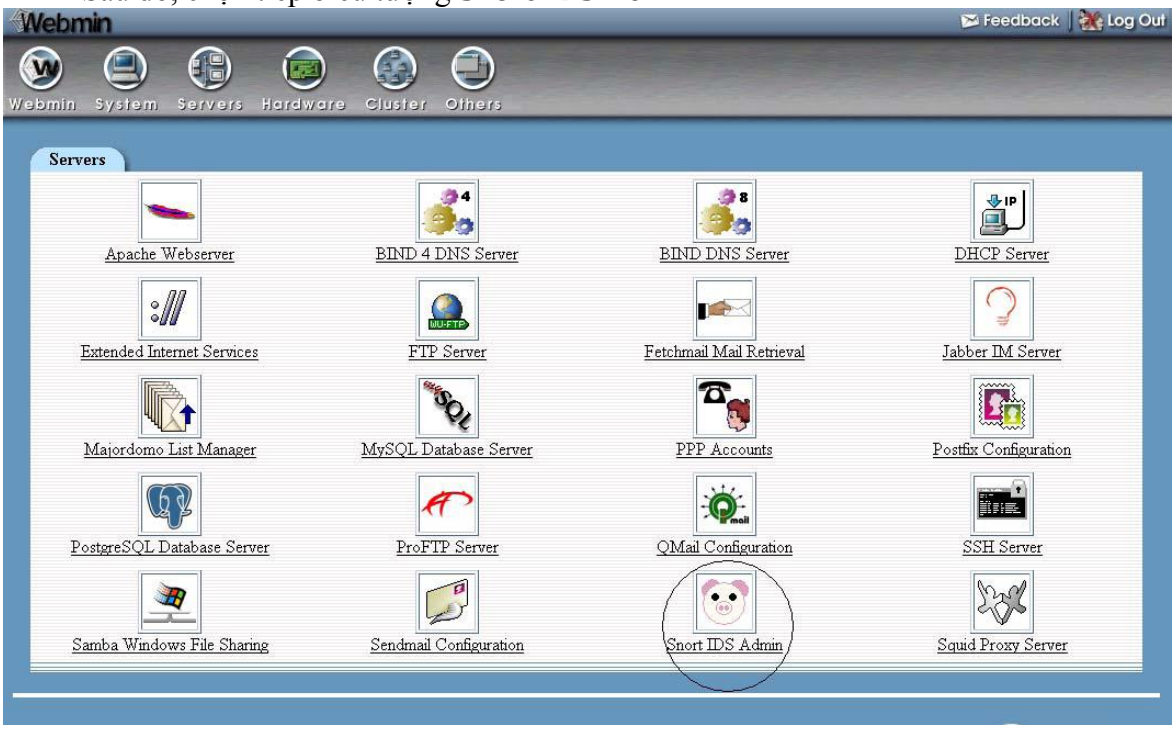
Hình 4-6: DB setup

Click **Main page** và tất cả hoàn tất.

Quá trình cài đặt của chúng ta đã hoàn tất, chúng ta có thể quản lí snort bằng cách log vào địa chỉ **https://localhost:10000** với quyền ROOT, chọn biểu tượng **Servers**:

Hình 4-7: Module Server

Sau đó, chọn tiếp biểu tượng **Snort IDS Admin**

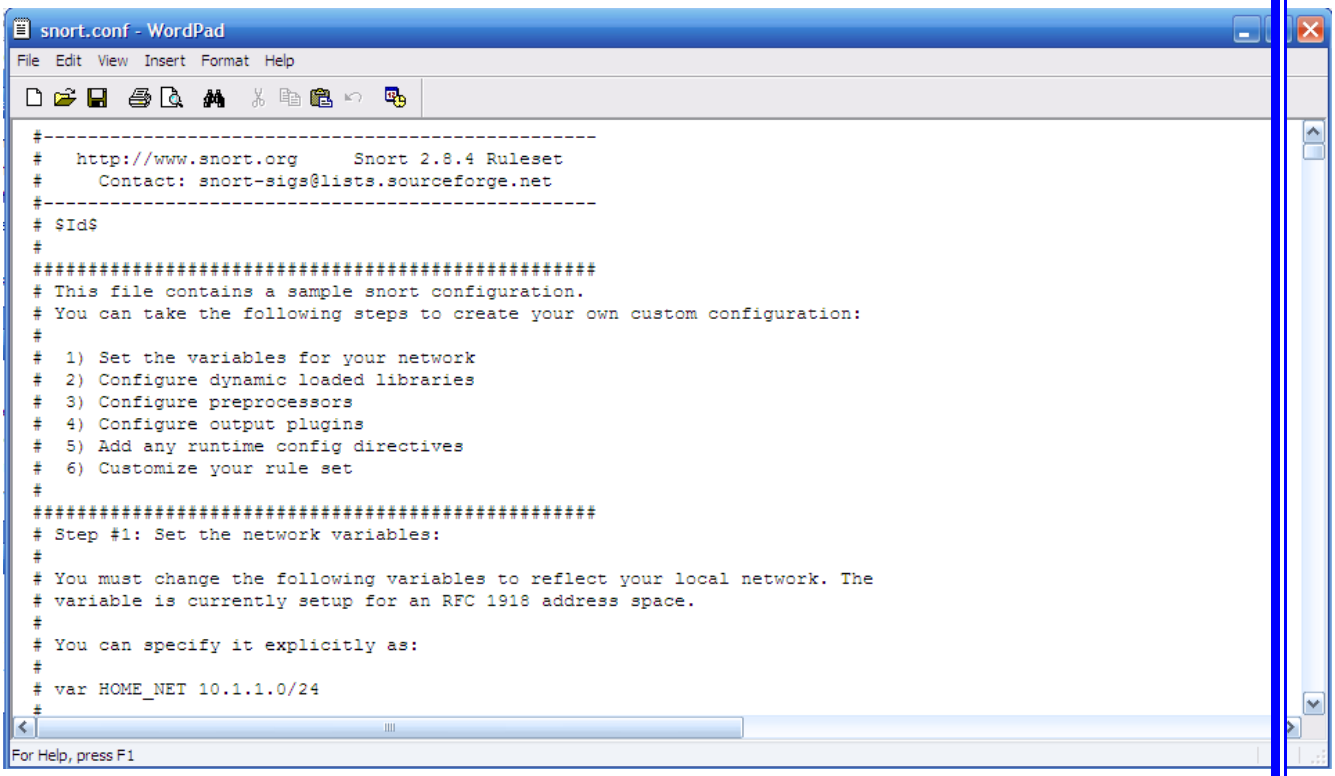


Hình 4-8: Snort IDS admin

Muốn xem thông tin về các packet mà snort log lại được thì vào địa chỉ **http://localhost/acid** với quyền của snort hoặc **http://localhost/acidviewer** với quyền của acidviewer.

4.6 Cấu Hình Snort:

Cấu hình snort được thực hiện chủ yếu trong file cấu hình có tên thường dùng là snort.conf, đây là một file có cấu trúc dạng text.



Hình: File Snort.conf

Dưới đây là các bước cấu hình cơ bản

- Thiết lập các biến: đây là nơi thiết lập các biến toàn cục xài cho rule, các chỉ lệnh như include, … Khai báo có dạng `var $Ten_Bien = gia tri.`
- Cấu hình preprocessor: đây là nơi cấu hình các preprocessor sẽ hoạt động, khai báo có dạng `preprocessor <name_of_processor>: <configuration_options>.`
- Cấu hình output plugin: đây là nơi cấu hình các output plugin sẽ hoạt động, khai báo có dạng `output <name_of_plugin>: <configuration_options>`
- Định nghĩa các rule type mới được sử dụng trong các định nghĩa rule dưới đây là ví dụ cụ thể về việc định nghĩa một rule type mới

```
ruletype suspicious
{
    type log
    output log_tcpdump: suspicious.log
}
```

- Tạo ra các rule: đây là nơi bạn sẽ định nghĩa ra các rule mà snort sẽ sử dụng để detect các gói tin, thường thì sẽ định nghĩa trên một file khác để tiện quản lý và ở đây bạn sẽ include vào.

Ngoài ra còn một số cấu hình khác bạn có thể xem chi tiết trong file snort.conf hoặc manual đi kèm.

4.7 Hướng Dẫn Sử Dụng Snort Trong Linux.

4.7.1 Sniffer mode.

Như đã giới thiệu, ở mode sniffer, snort đọc thông tin về các packet đang lưu chuyển trong mạng và hiển thị thông tin lên màn hình console.

Nếu chỉ muốn biết thông tin header của packet thì sử dụng:

./snort –v

Câu lệnh sau sẽ cung cấp nhiều thông tin hơn, ngoài thông tin về header, snort còn cho biết packet đang được ứng dụng nào lưu chuyển:

./snort –v –d

Nếu bạn muốn được cung cấp nhiều thông tin hơn nữa, muốn hiển thị các header của tầng datalink thì sử dụng câu lệnh:

./snort –vde hoặc **./snort –v –d –e**

4.7.2 Packet logger mode.

Packet logger mode là một mode hỗ trợ lưu thông tin packet vào đĩa cứng. Bạn chỉ cần đơn giản sử dụng câu lệnh sau, snort sẽ hiểu phải hoạt động ở mode Packet logger, và lưu thông tin xuống file đã được chỉ định:

./snort –dev –l ./log

Nếu bạn muốn log thông tin packet ở dạng phức tạp hơn để phục vụ cho việc phân tích sau này, bạn có thể log thông tin ở dạng binary. Khi đó, bạn sử dụng câu lệnh sau:

./snort –l ./log –b

Một khi đã lưu thông tin ở dạng binary, bạn cần một chương trình để dịch file binary ra dạng mà bạn có thể đọc được, như: tcpdump hoặc Ethereal. Snort cũng hỗ trợ việc đọc ngược này, đơn giản bằng cách sử dụng lệnh:

./snort –dv – r packet.log

Bạn có thể thêm tham số, để báo cho snort biết loại packet cần đọc. Ví dụ: bạn chỉ cần thông tin về các gói icmp, hãy sử dụng câu lệnh:

./snort –dv –r packet.log icmp

4.7.3 Network Intrusion Detection Mode (NIDS).

Đây là mode hoạt động phức tạp nhất của snort, có rất nhiều tham số để sử dụng. Tuy nhiên tham số quan trọng, bắt buộc phải có của mode hoạt động này là **"c".** Tham số này chỉ ra đường dẫn của các file luật, nhờ đó snort chỉ log lại những packet mà các file luật này yêu cầu.

./snort –u snort –g snort –d –D –c /etc/snort

4.7.3.1 Định dạng của một cảnh báo (alert).

Một cảnh báo (alert) có định dạng sau:

[**] [116:56:1] {snort_decoder}: T/TCP Detected [**]

Số đầu tiên là **Generator ID (GID),** GID sẽ cho biết cảnh báo này do thành phần nào của snort phát sinh (do lỗi nào phát sinh). Để có danh sách các GID tìm trong file **/etc/generators**.

Số thứ hai là **Snort ID (SID)**, SID sẽ cho biết cảnh báo này do preprocessor nào tạo ra. Để có danh sách các preprocessor xem trong file **/etc/gen-msg.map**.

Số thứ ba là revision ID, số này để phân biệt các cảnh báo.

4.7.3.2 Các tham số cảnh báo.

NIDS có rất nhiều tùy chọn để định nghĩa cách cảnh báo, cách ghi lại packet. Mặc định của mode này là cảnh báo **full alert** và log lại packet theo dạng **ASCII**. Sau đây là bảng các tham số để định nghĩa các cảnh báo:

Tham số	Cách cảnh báo
-A fast	NIDS sẽ đưa ra cảnh báo ở dạng đơn giản gồm có: thông điệp cảnh báo, địa chỉ IP nguồn và địa chỉ IP đích.
-A full	Đây là mode được sử dụng mặc định nếu bạn không dùng tham số.
-A unsock	Gởi cảnh báo đến một cổng UNIX để một chương trình khác có thể lắng

	nghe.
-A none	Tắt cảnh báo.
-A console	In những cảnh báo dưới dạng "fast" ra màn hình console.
-A cmg	Tạo những cảnh báo dạng "cmg".

4.7.4 Inline mode.

Khi hoạt động ở mode này, snort sẽ can thiệp trực tiếp vào iptables. Có 3 loại luật được sử dụng khi snort hoạt động ở mode Inline:

- **drop**: với kiểu luật drop, iptables sẽ bỏ qua packet và log lại sự kiện này.
- **reject**: với kiểu luật reject, iptables sẽ bỏ qua packet, log lại sự kiện, và thông báo đến máy tính rằng packet này sẽ không đến nơi.
- **sdrop**: với kiểu luật sdrop, iptables sẽ bỏ qua packet, không thông báo đến máy đích và cũng không log lại sự kiện.

Để snort hoạt động ở mode Inline, khi biên dịch snort cần chú ý các tham số như sau:

./configure –enable-inline
make
make install

Để gọi snort chạy ở mode inline, sử dụng câu lệnh sau:

snort_inline –QDc ../etc/drop.conf –l /var/log/snort

Các tham số có ý nghĩa như sau:

- **-Q:** lấy packet từ iptables.
- **-D**: chạy daemon snort_inline.
- **-c**: đọc file cấu hình.
- **-l**: ghi lại sự kiện vào thư mục.