BỘ CÔNG THƯƠNG

TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP. HCM
KHOA CÔNG NGHỆ THÔNG TIN

ĐỒ ÁN 1

XÂY DỰNG HỆ THỐNG IPS DÙNG HIDS-OSSEC

OSSEC

Open Source HIDS SECurity

GVHD: Nguyễn Hoà

Sinh viên thực hiện:

1. Phạm Văn Duẩn (13037321)
2. Võ Thanh Luân(13017701)
3. Mai Quốc Quy (13032721)
4. Nguyễn Hữu Vinh (13021241)

TP. Hồ Chí Minh, tháng 5 năm 2016

Mục lục

I. Đặt vấn đề

Khi mạng Internet ngày càng phát triển và các mạng nội bộ xuất hiện nhiều ở khắp mọi nơi, thách thức về các vấn đề xâm phạm và bảo mật mạng được đặt ra.

Nhiều biện pháp đã được đưa ra để bảo mật cơ sở hạ tầng mạng và truyền thông trên Internet. IDS hoặc HIDS là phương pháp bảo mật có khả năng chống lại các cuộc tấn công, các hoạt động trái phép trên hệ thống.

1. Tầm quan trọng của IDS

IDS (Intrusion Detection System – Hệ thống phát hiện xâm nhập) là một hệ thống giám sát hoạt động trên hệ thống mạng và phân tích để tìm ra các dấu hiệu vi phạm đến các quy định bảo mật máy tính, chính sách sử dụng và các tiêu chuẩn an toàn thông tin. Các dấu hiệu này xuất phát từ rất nhiều nguyên nhân khác nhau, như lây nhiễm malwares, hackers tấn công, người dùng truy nhập tráo phép vào các tài nguyên..v.v

Sử dụng IDS giúp nâng cao khả năng quản lí và bảo vệ mạng. Nó giúp hệ thống an toàn trước những nguy cơ tấn công, nó cũng cho phép nhà quản trị nhận dạng và phát hiện những nguy cơ tiềm ẩn dựa trên những phân tích và báo cáo được IDS cung cấp. Từ đó, IDS có thể góp phần giảm thiểu đáng kể những lỗ hổng bảo mật trong môi trường mạng.

2. Vai trò của HIDS

HIDS thường được cài đặt trên một máy tính nhất định. Thay vì giám sát hoạt động của một network segment, HIDS chỉ giám sát các hoạt động trên một máy tính. Được trển khai trên từng host, thông thường là một software hoặc một agent, mục tiêu là giám sát các tính chất cơ bản, các sự kiện liên quan đến các thành phần này nhằm nhận diện các hoạt động khả nghi. Host-based IDS/IPS thường được một dịch vụ quan trọng (gọi là Application-based IDS/IPS).

Quá trình triển khai các agent HIDS/IPS thường đơn giản do chúng là một phần mềm được cài đặt trực tiếp lên host. Application-based agent thường được triển khai

thẳng hàng ngay phía trước host mà chúng bảo vệ. Một trong những lưu ý quan trọng trong việc triển khai hệ thống Host-based IDS/IPS là cân nhắc giữa việc cài đặt agent lên host hay sử dụng agent-based appliances. Trên phương diện phát hiện và ngăn chặn xâm nhập, việc cài đặt agent lên host được khuyến khích với 1 số hệ điều hành nhất định nên trong trường hợp người ta sử dụng thiết bị. Một lý do khác để sử dụng thiết bị là việc cài đặt agent lên host có thể ảnh hưởng đến performance của host.

Hệ thống HIDS/ IPS cung cấp các khả năng bảo mật:

- Khả năng ghi log.
- Khả năng phát hiện.
    + Phân tích mã
    + Phân tích và lọc lưu lượng mạng.
    + Giám sát filesystem.
    + Phân tích log.
    + Giám sát cấu hình mạng.
- Khả năng ngăn chặn.

Ưu, nhược điểm của HIDS

Ưu điểm:

- Có khả năng xác định người dùng liên quan tới một sự kiện.
- HIDS có khả năng phát hiện các cuộc tấn công diễn ra trên một máy.
- Có thể phân tích các dữ liệu mã hoá.
- Cung cấp các thông tin về host trong lúc tấn công diễn ra.

Nhược điểm:

- Thông tin từ HIDS là không đáng tin cậy ngay khi sự tấn công vào host này thành công.

- Khi hệ điều hành bị hạ do tấn công, đồng thời HIDS cũng bị hạ.

- HIDS phải được thiết lập trên từng host giám sát.

- HIDS không có khả năng phát hiện các cuộc dò mạng.

- HIDS cần tài nguyên Host để hoạt động.

- Đa số chạy trên hệ điều hành window. Tuy nhiên cũng đã có 1 số chạy trên linux như trên Ubuntu.

3. Bài toán đặt ra của đề tài

Trong một mạng LAN có thể có rất nhiều máy Workstation. Việc quản lí các Worktation gặp phải rất nhiều khó khăn. Việc người dùng làm gì trên máy tính của mình là chuyện khó quản lí, hoặc bị tấn công mà không hề hay biết.

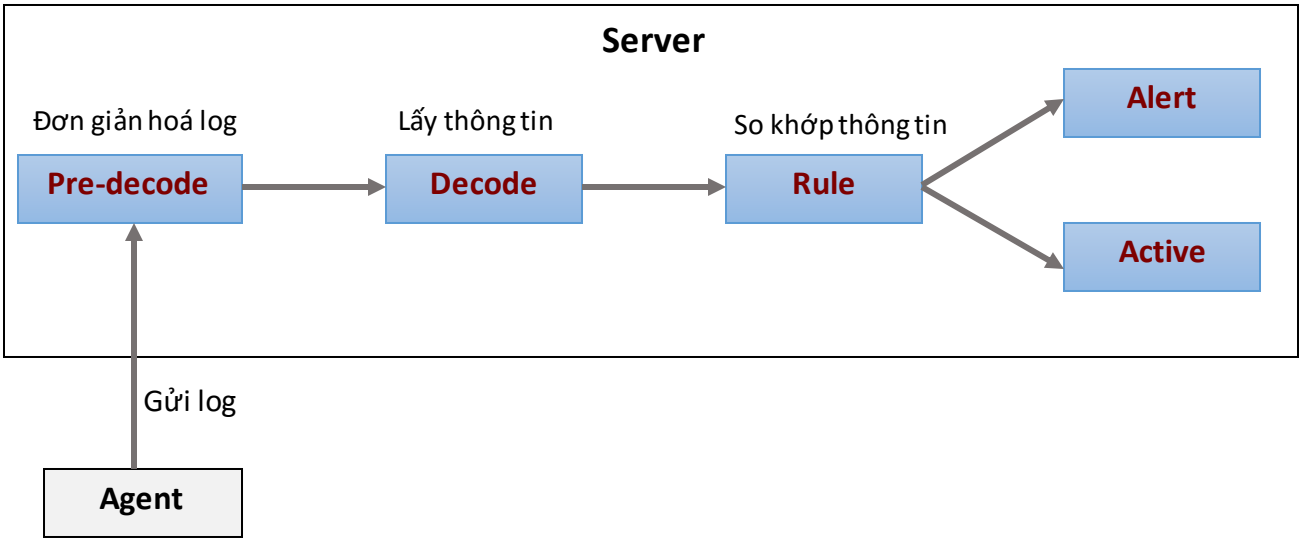
Vì vậy việc quản trị an ninh, an toàn mạng: bao gồm các công tác quản lý, giám sát mạng lưới, các hệ thống để đảm bảo phòng tránh các truy nhập trái phép. Việc phòng chống, ngăn chặn sự lây lan của các loại virus máy tính, các phương thức tấn công như Dos làm tê liệt hoạt động của mạng cũng là một phần rất quan trọng trong công tác quản trị, an ninh, an toàn mạng.

Điều đó có nghĩa là người quản trị phải quản lý tất cả các hành vi từ bên trong lẫn bên ngoài của hệ thống mạng . Thông qua việc server phải quản lý được tất cả các log event của các thiết bị mạng trong hệ thống mạng. Bài toán đặt ra ở đây là làm thế nào để người quản trị có thể quản lý chặt chẽ các sự kiện trong hệ thống mạng và xử lý cảnh báo hoặc ngăn chặn hành động đó với các mức độ quan trọng khác nhau. Đề tài này thực hiện công việc phát hiện và ngăn chặn xâm nhập bằng sự hỗ trợ của hệ thống OSSEC HIDS.

II. Mô hình hệ thống và phương pháp thực hiện

Trong phần này, sẽ làm rõ quá trình thu thập và phân tích log. Cùng với việc giới thiệu công nghệ OSSEC và quá trình phân tích log của nó. Nêu rõ quy trình thực hiện của đề tài.

1. Mô hình hệ thống



Hình 1. Mô hình hoạt động của hệ thống

- **Agent**

Agent chính là các máy workstation. Agent được cấu hình để kiểm quản lí file (tra tính toàn vẹn của file), kiểm tra rootkit, phân tích ventlog, quản lí Registry, đọc log, có logcollector để thu thập và gửi log về cho server.

- **Pre-decode**

Pre-decode là giai đoạn đầu tiên của quá trình phân tích log. Log nhận từ các Agent gửi đến hoặc trên local.

Những log có định dạng mà pre-decode nhận ra, sẽ được trích xuất ra các thông tin mặc định (hostname, program_name, time, date và log message,). Pre-decode làm cho log trông đơn giản hơn và việc decode cũng sẽ đơn giản, nhanh hơn.

Pre-decode gửi thông tin đã trích xuất từ log ban đầu sang cho decode, hoặc gửi tất cả đoạn log vì nó không đúng định dạng.

- **Decode**

Tuỳ vào mỗi loại file log sẽ có một decoder (mặc định hoặc do người dùng định nghĩa) riêng tương ứng. Decoder trích xuất các thông tin khác quan trọng (user, ip,

port, protocol, …) mà pre-decode không thể làm. Các thông tin này được dùng trong rule, cây rule.

- **Rule**

Các thông tin từ việc decode, nhanh chóng được so khớp với rule, cây rule. Tuỳ theo mức độ cảnh báo mà alert hay active response được sinh ra.

- **Alert**

Là kết quả của việc áp dụng rule, alert có thể được lưu vào database hoặc gửi cho người dùng qua mail.

- **Active**

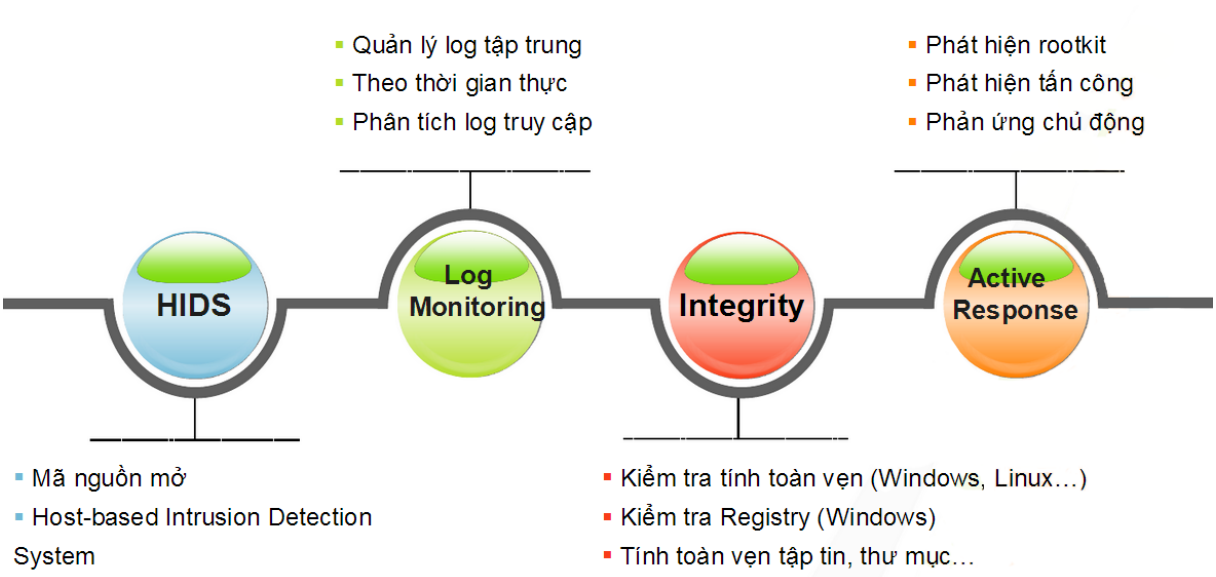
Thực hiện active response để ngăn chặn các hành vi trái phép ngay lập tức.

2. Công nghệ thực hiện

Như đã ghi ở tên đề tài, công nghệ thực hiện ở đây là HIDS-OSSEC. Trong phần này, giới thiệu OSSEC là gì và làm rõ quá trình phân tích log của OSSEC.

2.1. Giới thiệu OSSEC

Là một chương trình để theo dõi và kiểm soát hệ thống của bạn. Nó là một phần mềmcủa HIDS (dựa trên máy chủ phát hiện xâm nhập – Host IDS), giám sát đăng nhập và các tác vụ của hệ thống với một giải pháp mã nguồn đơn giản, mạnh mẽ. Nó cũng được hỗ trợvà hỗ trợ đầy đủ bởi Trend Micro.



Hình 2. Tổng quan về OSSEC

2.1.1. Lợi ích của OSSEC

➢ Đáp ứng được các tiêu chuẩn quốc tế

OSSEC cho phép bạn phát hiện và cảnh báo những thay đổi hệ thống tập tin và nhữnghành vi nguy hiểm nhúng trong các file log của các sản phẩm, cũng như các ứng dụng tùychỉnh. Nó bao gồm các phần giám sát tính toàn vẹn file kiểm tra đăng nhập,

giám sát vàthực thi chính sách kiểm tra. Tuân thủ các tiêu chuẩn như PCI (Payment Card Industry),HIPAA (Health Insurance Portability and Accountability Act)

- Là một hệ thống đa nền tảng

  Có thể thực hiện một hệ thống IPS dựa trên các chính sách riêng trên nhiều nền tảng khác nhau. Tuy nhiên, chỉ cần quan tâm Windown.

- Xây dựng cảnh báo theo thời gian thực

Cho phép khách hàng cấu hình những cái họ muốn được cảnh báo, họ có thể tập trungvào các sự cố quan trọng trong hệ thống hơn là những thứ khác. Tích hợp với SMTP, sms và syslog cho phép khách hàng có các cảnh báo trên bằng việc gửi trên e-mail và cách thiết bị cầm tay như điện thoại di động. Các hoạt động active response cũng có sẵn để ngăn chặn một cuộc tấn công ngay lập tức.

- Quản lí tập trung

OSSEC cung cấp một server chính quản lý tập trung các chính sách cho nhiều máy chủ sử dụng nhiều hệ điều hành khác nhau. Ngoài ra nó còn cho phép khách hàng có thể định nghĩa những chính sách riêng cho từng server cụ thể nhằm tối ưu hóa cho các chính sách.

2.1.2. Các chức năng chính của OSSEC

- Kiểm tra tính toàn vẹn của tập tin

Sử dụng MD5/SHA1 checksums, hỗ trợ cấu hình file được check, tần suất, thời gian,… Mục đích của việc kiểm tra tính toàn vẹn là để phát hiện và cảnh báo các thay đổi bất thường trong hệ thống khi có một cuốc tấn công nào đó và các mạng và máy tính.

- Phát hiện rootkit

Chúng là công cụ phần mềm được cài đặt bởi Hacker để ẩn giấu một số file và tiến trình chạy ngầm khi tấn công vào hệ thống. Việc sử dụng OSSEC sẽ phát hiện rootkit, tự động kiểm tra sau một khoảng thời gian nhất định do người dùng định nghĩa, dựa trên việc so sánh với cơ sở dữ liệu mà các rootkit bị phát hiện và được thông báo.

- Kiểm tra Registry (Windows)

Registry trên Windows là một phần khá nhạy cảm vì dễ bị tấn công. Việc kiểm tra sẽ giúp phát hiện những thay đổi nhỏ đôi khi có thể gây nguy hiểm toàn hệ thống.

2.1.3. Kiến trúc hệ thống

Hình 3. Kiến trúc hệ thống của OSSEC

Cài đặt và hoạt động trực tiếp trên một máy local

Dạng Client-Server gồm:

- Phần quản lý cài ở máy chủ

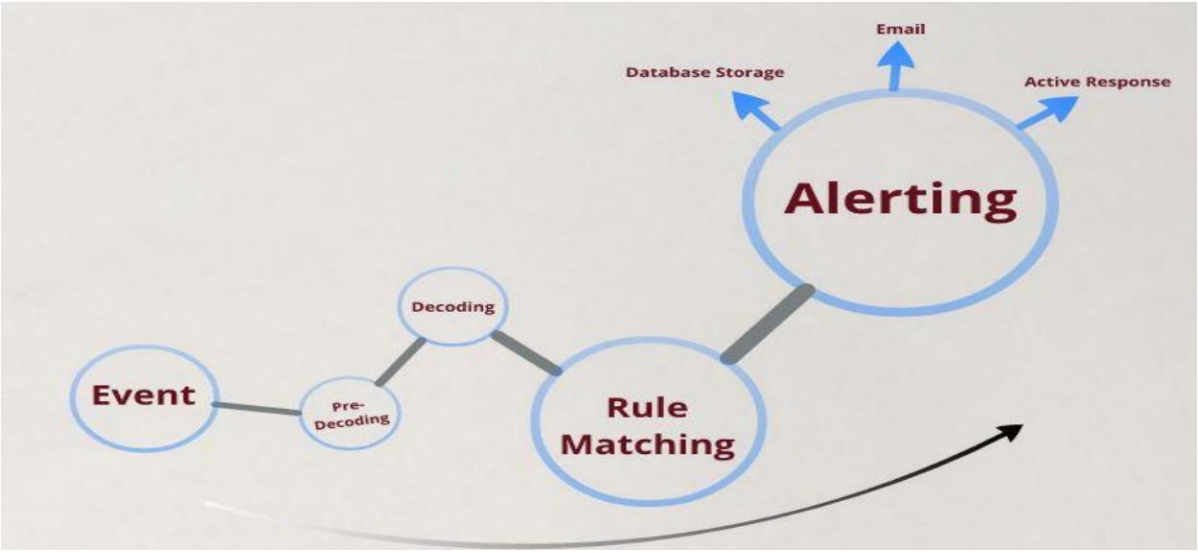
  Có thể nhận log từ Firewalls, switches and routers

- Agent:

  Agent có thể cài ở máy client hoặc cũng hỗ trợ tính năng agentless

  Agent có thể cài được trên máy ảo

2.2. Quá trình phân tích log của OSSEC



Hình 4. Quá trình phân tích log của OSSEC

2.2.1. Giai đoạn Pre-Decoding

Quá trình pre-decoding là rất đơn giản, nó chỉ trích xuất thông tin tĩnh từ các thuộc tínhcủa một sự kiện. Các thông tin được trích xuất trong giai đoạn này là time, date, hostname,program name, và log message, vv. Những thông tin tĩnh quan trọng từ log được lấy ra và được thực hiện sau đó ở trong phần decoding của quá trình.

Mục đích của pre-decoding là làm cho việc decoding trờ nên đơn giản hơn.

Ví dụ về pre-decoding qua đoạn log được tạo bởi chương trình *sshd*

"Apr 14 17:32:06 linux_server sshd[1025]: Accepted password for dcid from 192.168.2.180 port 1618 ssh2"

Thuộc tính	Kết quả
hostname	linux_server
program_name	sshd
log	Accepted password for dcid from 192.168.2.180 port 1618 ssh2
time/data	Apr 14 17:32:06

Bảng 1. Ví dụ pre-decode

Tuy nhiên, các loại log không đúng định dạng thì quá trình pre-decoding trở nên vô nghĩa.

2.2.2. Giai đoạn Decoding

Là bước tiếp theo trong quá trình, sau giai đoạn pre-decoding. Mục tiêu của decoding là trích xuất thông tin động, quan trọng từ các log để có thể sử dụng trong các rules sau này. Decoder sẽ trích xuất thông tin như địa chỉ IP, username, port …

Từ ví dụ trên ta có được log như sau:

Log: ”Accepted password for dcid from 192.168.2.180 port 1618 ssh2”

Sau quá trình Decoding ta được:

Thuộc tính	Kết quả
user	dcid
srcip	192.168.2.180

Bảng 2. Ví dụ decode

Decoder có thể được cấu hình để lấy được tất cả những thông tin quan trọng, không phụ thuộc vào định dạng của log, từ nhiều nguồn log khác nhau.

Tất cả các decoder được cấu hình trong file */var/ossec/etc/decoder.xml*

Sử dụng Decoder với các thẻ ở bảng bên dưới

Bảng 3. Các thẻ trong decoder

Tag	@	Value	Mô tả
decoder			
	@name		Tên decoder duy nhất
parent			Tên của decoder cha
program_name			Tên chương trình phải giống như ở pre-decoding

prematch			Thực hiện decode nếu trùng khớp
	@offset	after_parent	Bắt đầu lấy thông tin tại nơi decoder cha dừng
regex			Biểu thức dùng để trích xuất thôn tin
	@offset	after_parent after_prematch after_regex	Bắt đầu sau parent, prematch hoặc regex của decoder cha
oder		srcip, dstip,srcport, dstport, protocol, action, user, id, status, command, url, data, system_name	Các thuộc tính tương ứng với các biểu thức trong regex theo thứ tự lần lượt
type		firewall, ids, syslog, web-log, squid, windows, ossec	Xác định từng loại log

Ví dụ với đoạn log sau

"Apr 14 17:32:06 linux_server sshd[1025]: Accepted password for dcid from 192.168.2.180 port 1618 ssh2"

Decoder:

```
<decoder name="sshd-success">
    <parent>sshd</parent>
    <prematch>^Accepted</prematch>
    <regex offset="after_prematch">^ \S+ for (\S+) from (\S+) port</regex>
    <order>user, srcip</order>
</decoder>
```

Mỗi decode được giới hạn trong thẻ <decoder></decoder> và tên của decoder cũng được định nghĩa trong thẻ này. Ví dụ: <decoder name="sshd-success"></decoder>.

Trong các decoder chúng ta cần phải xác định thẻ <prematch></ prematch>. Khi prematch phù hợp với những gì trong file log, các regex sẽ được gọi là để trích xuất user, ip. Ví dụ: <prematch>^Accepted</prematch>

Để trích xuất thông tin về user, ip chúng ta phải sử dụng thẻ <regex></ regex>. Để đảm bảo các biểu thức được được so sánh sau những cái gì đã được đọc từ prematch, giúp tiết kiệm rất nhiều thời gian. Các đối tượng cần được trích xuất nằm trong thẻ ( ). Ví dụ: <regex offset="after_prematch">^ \S+ for (\S+) from (\S+) port</regex>

Các biểu thức chính quy được liệt kê trong bảng dưới

Bảng 4. Các biểu thức chính quy

Biểu thức	Ý nghĩa
\w	A - Z, a - z, 0 - 9, và các kí tự: - @
\d	0 – 9
\s	Khoảng trắng
\t	Phím tab
\p	Dấu câu
\W	Mọi thứ trừ \w
\D	Mọi thứ trừ \d
\S	Mọi thứ trừ \s
\.	Mọi thứ
+	Lặp lại 1 hoặc nhiều lần
*	Lặp lại 0 hoặc nhiều lần
^	Xác định nơi bắt đầu
$	Xác định nơi kết thúc
|	Phép “OR” giữa hai hoặc nhiều biểu thức

Chúng ta cần phải xác định order để nới với HIDS OSSEC mỗi field nào của thông báo cần phải parsing. Thứ tự các đối tượng trong order là thứ tự thông tin xuất hiện trong regex. Ví dụ <order>user, srcip</order>. Lấy ra user và ip.

2.2.3. Rule Matching

Rule và decoder kết hợp với nhau giúp cho OSSEC phát huy được các tính năng mạnh mẽ của mình. Khi kết hợp chúng với nhau cho phép cấu hình và điều chỉnh các cảnh báo từ OSSEC, bao gồm kiểm tra tính toàn vẹn, syslog, caclog events của agent và cảnh báo phát hiện rootkit.

Rule của OSSEC được lưu trữ bên trong thư mục */var/ossec/rule*. Mỗi rule được xác định với phần mở rộng là XML riêng biệt và được đặt tên phù hợp. XML được sử dụng thay vì file cấu hình (dạng text) bởi vì XML dễ đọcvà hiểu được nó dễ dàng. Có 43 file rule được cài đặt mặc định trong HIDS OSSEC.

Mỗi các rule có một Rule_ID duy nhất, Rule_ID do người dùng tự định nghĩa một rules nằm trong khoảng 100000 đến 119999.

Có hai loại rule trong OSSEC: atomic và composite. Atomic rule dựa trên các sự kiện đơn lẻ, không có bất kì mối quan hệ nào. Ví dụ, một thông báo có thể được sinh ra với một lần đăng nhập thất bại duy nhất. Composite rule kết hợp nhiều sự kiện với nhau. Ví dụ với 10 lần đăng nhập thất bại, từ một IP và trong khoảng thời gian 180s thì lúc này cần tới một composite rule.

Các mức độ cảnh báo của OSSEC

Bảng 5. Các mức độ cảnh báo trong OSSEC

Level	Định nghĩa
0	Bỏ qua, không quan tâm
2	Thông báo của hệ thống mức thấp
3	Sự kiện thành công/uỷ quyền
4	Lỗi hệ thống mức thấp
5	Lỗi do người dùng tạo ra
6	Tấn công cấp thấp
7	So sánh với những từ khoá xấu
8	Sự kiện xuất hiện lần đầu
9	Lỗi từ nguồn không hợp lệ
10	Nhiều người dùng tạo ra lỗi
11	Cảnh báo kiểm tra tính toàn vẹn
12	Sự kiện có mức quan trọng cao
13	Lỗi bất thường (quan trọng cao)
14	Sự kiện bảo mật có tầm quan trong cao
15	Tấn công thành công

Rule được sử dụng với các thẻ ở bảng bên dưới

Bảng 6. Các thẻ sử dụng trong rule

Tag	@	Value	Mô tả
group			Thẻ nhóm các rule
	@name		Tên theo ngữ cảnh
rule			
	@id		Duy nhất
	@level	(number)	Từ 0 tới 15
	@maxsize	(number)	Kích thước tối đa của log
	@frequency	(number)	Số lần rule trùng khớp
	@accuracy	0/1	0 = no, 1 = yes
	@noalert	0/1	
	@ignore	(number)	Bao nhiêu lần rule xảy ra được bỏ qua
	@overwrite	yes/no	Chỉnh sửa lại rule mặc định
	@timefarm	(seconds)	
description			Bắt buộc
cve/infor			
action/status			Decode bởi decoder
group		invalid_login, authenication_success, authentication_failed, attacks, sshd, ids,(…)	Nhóm ứng dụng độc lập
decoded_as			Khớp với tên decoder
match			So sánh nhanh chuỗi với mẫu bất kì
regex			Giống như ở decoder

if_sid *if_group* *if_level*			Rule chỉ đúng nếu rule có cùng id/group/level
hostname			Giống với decode ở pre-decoder
srcip/dstip/ *srcport/dstport/* *user/id/url*			Giống với decode bằng decoder
program_name			Giống với decode ở pre-decoder
category		firewall, ids, syslog, web-log, squid, windows, ossec	Giống với kiểu đã khai báo ở decoder
weekday		sunday, sun, monday, mon, tuesday, tue, wednesday, wed, thursday, thu, friday, fri, saturday, sat, weekdays, weekends	Dành cho chính sách. Ngày trong tuần
time		hh:mm – hh:mm or hh:mm am – hh:mm pm	Dành cho chính sách
options		alert_by_email, no_email_alert, no_log	
extra_data			Được trích xuất bởi decode (data)
if_matched_sid *if_matched_group* *if_matched_regex*			Composite rule: rule chắc chắn trùng khớp với id/group/regex
same_source_ip *same_src_port* *same_dst_port* *same_user/same_location* *same_id/different_url*			Composite rule: quy định các giá trị trong từng thuộc tính phái giống nhau (ip đến từ 1 nguồn,…)

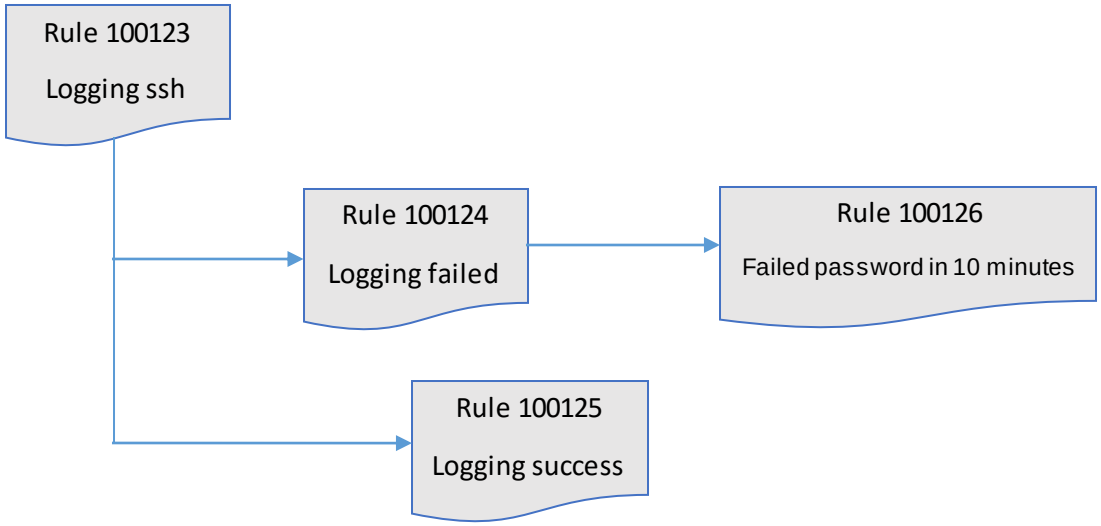
Ví dụ rule:

```
<group name="syslog,sshd,">

        <rule id="100123" level="2">
                <decoded_as>sshd</decoded_as>
                <description>Logging every decoded sshd message</description>
        </rule>
        <rule id="100124" level="7">
                <if_sid>100123</if_sid>
                <match>^Failed password</match>
                <group>authentication_failure</group>
                <description>Failed SSHD password attempt</description>
        </rule>
        <rule id="100125" level="3">
                <if_sid>100123</if_sid>
                <match>^Accepted password</match>
                <group>authentication_success</group>
                <description>Successful SSHD password attempt</description>
        </rule>
        <rule id="100126" level="10" frequency="5" timeframe="600">
                <if_matched_sid>100124</if_matched_sid>
                <same_source_ip />
                <description>5 Failed passwords within 10 minutes</description>
        </rule>

</group>
```

Cây rule của ví dụ trên



Hình 5. Ví dụ cây rule đăng nhập ssh

Atomic Rules

Mỗi rules, hoặc nhóm các rules, phải được xác định trong một thẻ <group></group>. Thuộc tính tên của nhóm cần được định nghĩa trong thẻ này.

Ví dụ: <group name="syslog,sshd,"> </group>

Dấu *(,)* để phân biệt hai nhóm khác nhau trong cùng một thẻ group .

Các quy tắc được xác định bằng cách sử dụng thẻ <rule> </ rule> và phải có ít nhất hai thuộc tính là id và level. Id là một định danh duy nhất và level để chỉ mức độ nghiêm trọng của các cảnh báo.

Ví dụ

```
<group name="syslog,sshd,">

        <rule id="100123" level="2">
                <decoded_as>sshd</decoded_as>
                <description>Logging every decoded sshd message</description>
        </rule>
</group>
```

Một thẻ quan trọng là thẻ <decoded_as> </decoded_as>. Thẻ này chỉ ra các decoder nào sẽ được sử dụng trong rule đó.

Thẻ <description> </ description> để mô tả chi tiết về rules đó khi được giải mã. Sử dụng thẻ <match> </ match> để đánh giá các phần của log. Dùng thẻ match chúng ta có thể tìm kiếm thông qua log và sử dụng Accepted password là một phần của thông báo như một chìa khóa để phát hiện tất cả các sự kiện thuộc loại này.

Thẻ <if_sid> <\if_sid> cho biết thêm một quy tắc để tạo cây rules.

Ví dụ:

```
<rule id="100125" level="3">
        <if_sid>100123</if_sid>
        <match>^Accepted password</match>
        <group>authentication_success</group>
        <description>Successful SSHD password attempt</description>
</rule>
```

Composite Rules

Nếu chúng ta muốn các sự kiện có rự liên quan với nhau, có một vài tùy chọn mà chúng ta cần phải hiểu rõ các quy định của Composite rules. Composite rules phải phù hợp với các events hiện tại với những gì đã nhận được OSSEC HIDS. Trong Composite Rules có hai tùy chọn bổ sung thứ nhất là tùy chọn frequency xác định bao nhiêu lần một event phải xảy ra trước khi các rule sinh ra cảnh báo. Thứ hai là tùy chọn timeframe cho HIDS OSSEC bao lâu thì trở lại, xác định trong khoảng thời gian nào đó nó sẽ đưa ra cảnh báo.

Ví dụ:

```
<rule id="100126" level="10" frequency="5" timeframe="600">
        <if_matched_sid>100124</if_matched_sid>
        <same_source_ip />
        <description>5 Failed passwords within 10 minutes</description>
</rule>
```

Ví dụ này tạo ra một rule kiểm tra đăng nhập nếu đăng nhập sai 5 lần trong khoảng thời gian 10 phút (600s) nó sẽ đưa ra một cảnh báo. Thẻ <if_matched_sid>< /if_matched_sid > đưa ra các quy tắc cần được nhìn thấy trong frequency và timeframe mong muốn cho các rule mới để tạo ra một cảnh báo. Thẻ <same_source_ip /> để chỉ việc đăng nhập được thực hiện trên một nguồn duy nhất.

2.3. Quy trình thực hiện của đề tài

Phương pháp thực hiện bao gồm các giai đoạn trong quá trình phân tích log của HIDS-OSSEC, và bổ sung thêm quá trình nhận dạng log và lấy log từ workstation.

2.3.1. Nhận dạng log

Bước đầu tiên của quá trình là nhận diện log. Nó bao gồm việc chọn sự kiện an ninh nào trên Windows được quan tâm và phân tích log được sinh ra. Các sự kiện an ninh trên Windows như đăng nhập, chia sẻ file, cài đặt phần mềm, … Xác định được những trường hợp sinh ra log có biểu hiện bình thường/bất thường, từ đó quyết định xem những thuộc tính nào nên được trích xuất trong quá trình Decode.

Các chính sách (trong Group policy) trên Windows nên được thiết lập để ghi lại log vào EventLog.

2.3.2. Lấy log từ workstation

OSSEC agent sẽ được cấu hình để log (toàn bộ hoặc chỉ những log quan tâm) gửi tới server.

2.3.3. Decode log

Việc decode log dựa trên kết quả của nhận dạng log. Từ những thuộc tính quan trọng cần được trích xuất, các decoder được viết để lấy ra các thông tin đó.

2.3.4. Rule

Các trường hợp bình thường/bất thường được sử dụng để viết rule và cây rule.

Các thông tin từ quá trình decode được so khớp với cây rule để sinh ra alert và active response.

2.3.5. Alert và Active Reponse

Alert được sinh ra từ việc so khớp rule. Tuỳ vào mức độ quan trọng, alert được cấu hình để thông báo cho từng người cụ thể bằng mail hoặc chỉ lưu vào database.

Trong đồ án một Active Response không được nhắc đến.

III. Các module phân tích và xử lý log của đề tài

Trong phần này sẽ có các module phân tích, xử lí log và lấy ví dụ minh hoạ cụ thể về File Share trên Windows.

1. Nhận dạng log

Trên Windows có rất nhiều log khác nhau phát sinh từ nhiều sự kiện khác nhau. Mỗi log có một ID duy nhất. Tuy nhiên, chúng ta chỉ quan tâm tới log được phát sinh từ sự kiện mang tính an ninh. Nhóm các sự kiện cần quan tâm được liệt kê trong bảng bên dưới.

Bảng 7 . Ví dụ pre-decode

STT	Nhóm các sự kiện	Mô tả
1	Accout Logon	Các sự kiện đăng nhập bằng tài khoản
2	Accout Management	Các sự kiện quản lí tài khoản
3	Directory Servive	Các sự kiện dịch vụ thư mục
4	Logon/Logoff	Các sự kiện đăng nhập/đăng xuất
5	Non Audit (Event Log)	Các sự kiện trên eventlog
6	Object Access	Các sự kiện quyền truy cập đối tượng
7	Policy Change	Các sự kiện khi thay đổi chính sách
8	Privilege Use	Các sự kiện quyền người dùng
9	Process Tracking/ Detailed Tracking	Các sự kiện theo dõi tiến trình hoặc chi tiết
10	System	Các sự kiện từ hệ thống

Ví dụ: Chia sẻ file ở trên Windows (*Share File* trong nhóm Object Access) sẽ có các bản ghi log như sau:

```
Windows 5140 A network share object was accessed
Windows 5142 A network share object was added.
Windows 5143 A network share object was modified
Windows 5144 A network share object was deleted.
```

Windows	5145	A network share object was checked to see whether client can be granted desired access
Windows	5168	Spn check for SMB/SMB2 fails.

Trong ví dụ này chúng ta sẽ quan tâm tới hai sự kiện chính: khi có một đối tượng được chia sẻ và bị xoá đi. Những log này có id tương ứng là 5140 và 5144.

Đối với việc chia sẻ file: Mặc định các thư mục ở trong ổ đĩa hệ thống bị cấm chia sẻ, ngoại trừ thư mục trong User (C:\User)

- *Bình thường*: Chia sẻ thư mục ngoài ổ đĩa hệ thống (khác ổ đĩa C) và thư mục cá nhân người dùng (bên trong C:\User)
- *Bất thường*: - Chia sẻ thư mục bên trong ổ đĩa hệ thống.
  - Chia sẻ thư mục bên trong C:\Windows (nguy hiểm)

Các thông tin nên được trích xuất ra để xác các trường hợp bình thường và bất thường khi chia sẻ trong giai đoạn decode: *status, id, user, hostname và path*

2. Lấy log từ workstation

Tiến hành lấy log:

- Cấu hình Group Policy để log được ghi lại trong EventLog và gửi tới Server.
- Cấu hình OSSEC Agent trong file *ossec.conf*

```
<localfile>
      <location>Security</location>
      <log_format>eventchannel</log_format>
      <query>Event/System[EventID=5140  or EventID=5144]</query>
</localfile>
```

Như vậy chỉ những log có ID 5140 hoặc 5144 sẽ được gửi tới OSSEC Server

- Chia sẻ file ta có được đoạn log:

```
WinEvtLog: Security: AUDIT_SUCCESS(5140): Microsoft-Windows-Security-Auditing: (no user): no domain: VanDuan-PC: A network share object was accessed.  Subject:  Security ID:  S-1-5-21-739373341-442988835-2982865765-1000  Account Name:  VanDuan  Account Domain:  VANDUAN-PC  Logon ID:  0x6D0388  Network Information:  Object Type:  File  Source Address:  fe80::9d4d:93cc:9f6d:c594  Source Port:  8545  Share Information:  Share Name:  \\*\Share Folder  Share Path:  \??\D:\Share Folder  Access
```

```
Request Information:  Access Mask:  0x1  Accesses:  ReadData (or
ListDirectory)
```

- Xoá một đối tượng chia sẻ ta được log:

```
WinEvtLog: Security: AUDIT_SUCCESS(5144): Microsoft-Windows-Security-
Auditing: (no user): no domain: VanDuan-PC: A network share object was
deleted.  Subject:  Security ID:  S-1-5-21-739373341-442988835-
2982865765-1000  Account Name:  VanDuan  Account Domain:  VANDUAN-
PC  Logon ID:  0x6D0353  Share Information:  Share Name:  \\*\Share
Folder  Share Path:  D:\Share Folder
```

(Các thông tin cần lấy được gạch chân)

3. Decode log

Viết decoder cho log trên với các thông tin cần trích xuất: *status, id, user, hostname và path*. Sau đó, lưu vào trong file *decoder.xml* trong thư mục: */var/ossec/etc/decoder.xml*

```
<!--File Share Create-->
<decoder name="windows-file-share-create">
    <prematch>\.(5140)\.</prematch>
</decoder>
<decoder name="windows-file-share-create-sub">
    <parent>windows-file-share-create</parent>
    <regex>^\.+:\.+:  (\.+)\((\d+)\): </regex>
    <regex>\.+Account Name:\s\s(\w+)\s\s\sAccount Domain:\s\s(\w+)\s\s\s</regex>
    <order>status, id, user, extra_data</order>
</decoder>
<decoder name="windows-file-share-create-sub">
    <parent>windows-file-share-create</parent>
    <regex>\.+Share Path:\s\s\\\p+\\(\.+)\s\s\s</regex>
    <order>url</order>
</decoder>
<!--File Share Delete-->
<decoder name="windows-file-share-delete">
    <prematch>\.(5144)\.</prematch>
</decoder>
<decoder name="windows-file-share-delete-sub">
    <parent>windows-file-share-delete</parent>
    <regex>^\.+:\.+:  (\.+)\((\d+)\): </regex>
    <regex>\.+Account Name:\s\s(\w+)\s\s\sAccount Domain:\s\s(\w+)\s\s\s</regex>
    <order>status, id, user, extra_data</order>
</decoder>
<decoder name="windows-file-share-delete-sub">
    <parent>windows-file-share-delete</parent>
```

```
    <regex>\.+Share Path:\s\s(\.+)</regex>
    <order>url</order>
</decoder>
```

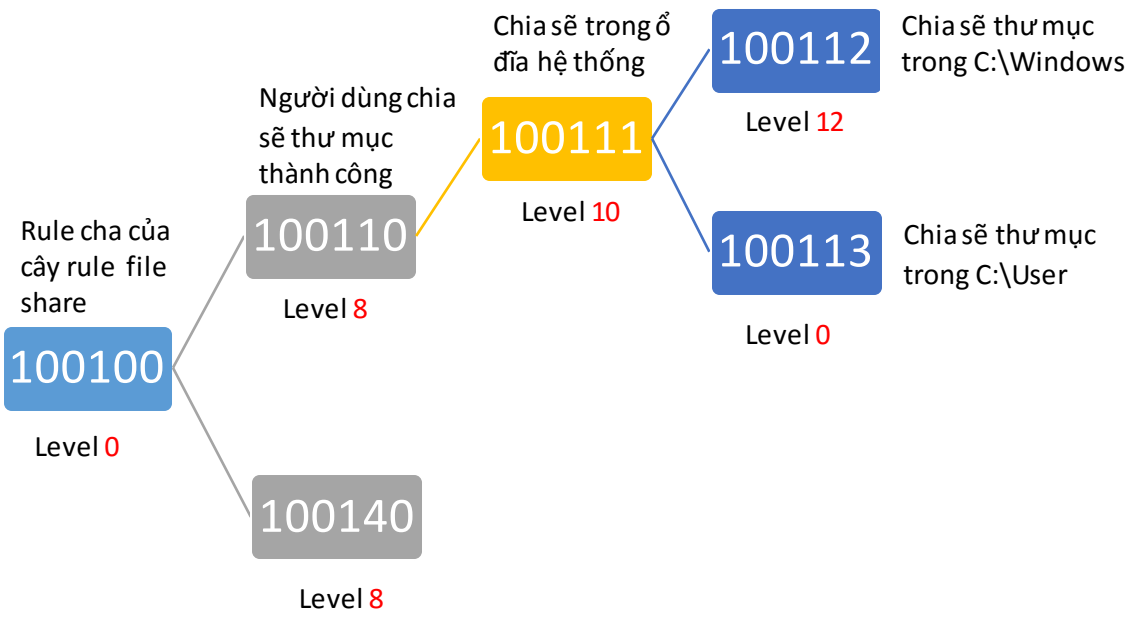
Ở đây decocer *windows-file-share-create* sẽ prematch để tim ra log có id bằng 5140. Tiếp theo hai decoder con sẽ trích xuất thông tin với regex. Tương tự như vậy với decoder *windows-file-share-delete.*

4. Rule Matching

Xây dựng cây rule *ms_file_share_rules.xml* và lưu vào */var/ossec/rules/*

```
<group name="ms-file-share,">
    <rule id="100100" level="0">
        <id>5140|5142|5143|5144</id>
        <description>WINDOWS FILE SHARE message grouped</description>
        </rule>
    <rule id="100110" level="8">
        <if_sid>100100</if_sid>
        <id>5140</id>
        <match>A network share object was accessed</match>
        <description>Nguoi dung chia se thu muc</description>
    </rule>
    <rule id="100111" level="10">
        <if_sid>100110</if_sid>
        <url>^C:</url>
        <description>Nguoi dung chia se thu muc trong o dia he thong</description>
    </rule>
    <rule id="100112" level="13">
        <if_sid>100111</if_sid>
        <url>^C:\Windows|^C:\windows|^C:\WINDOWS</url>
        <description>Nguoi dung chia se thu muc trong C:\Windows</description>
    </rule>
    <rule id="100113" level="0">
        <if_sid>100111</if_sid>
        <url>^C:\Users|^C:\users|^C:\USERS</url>
        <description>Nguoi dung chia se thu muc trong C:\Users</description>
    </rule>
    <rule id="100140" level="8">
        <if_sid>100100</if_sid>
        <id>5144</id>
        <match>A network share object was deleted</match>
        <description>Nguoi dung xoa thu muc chia se</description>
    </rule>
</group>
```

Mô hình cây rule



5. Alert và Active Reponse

Khi áp dụng file rule thành công các cảnh báo sẽ được lưu lại trong */var/ossec/logs/alerts/*.

Ở đây là 3 mẫu cảnh báo:

- Cảnh báo thứ nhất cho việc lần đầu tiên chia sẻ file, đây cũng có thể là lỗi của người dùng hoặc do từ tấn công tạo ra (level 8).
- Cảnh báo thứ hai cho việc chia sẻ file trong ổ đĩa hệ thống. Điều này sẽ đưa ra mức cảnh báo ở level cao và sẽ cảnh báo qua email tới người quản trị (level 10)
- Cảnh báo thứ ba cũng thông báo cho việc chia sẻ file nhưng đó là file trong thư mục cấm chia sẻ nên mức độ cảnh báo sẽ ở level 12 và bắt đầu thực hiện Active response ngăn chặn ngay việc chia sẻ đó (level 12)

Khi phát hiện được tấn công, OSSEC có phản ứng lại theo hành động mặc định là chặn kết nối của máy tấn công.

5.1. Thiết lập cảnh báo qua mail

Để hệ thống cảnh báo đến email của người quản trị (*root@mail.server.com*) thì chúng ta định nghĩa <email_to>*abc@mail.server.com*</email_to> trong tag <global> </global>.

```
<globle>
      <mail_notification> yes </mail_notification>
      <mail_from> root@mail.server.com </mail_from>
      <mail_to> abc@mail.server.com </mail_to>
      <smtp_server>mail.server.com</smtp_server>
</globle>
```

Hệ thống thực hiện gửi mail với giao thức smtp server có tên "mail.server.com".

Ví dụ thiết lập cảnh báo chi tiết cho file share

```
<email_alerts>
      <rules_id>100110|100140</rules_id>
      <mail_from> vinh@mail.server.com </mail_from>
      <level>8</level>
</email_alerts>
```

Khi người dùng thực hiện chia sẻ thì mail hệ thống sẽ tự gửi mail thông báo đến mail người dùng (mail local: "vinh@mail.sevrer.com) cho người dùng biết được việc chia sẻ file được diễn ra.

```
<email_alerts>
      <rules_id>100111</rules_id>
      <mail_from> abc@mail.server.com</mail_from>
      <level>10</level>
</email_alerts>
<email_alerts>
      <rules_id>100112|100113</rules_id>
      <mail_from>xyz@mail.server.com</mail_from>
      <level>12</level>
      <do_not_delay/>
</email_alerts>
```

Trong trường hợp này level 12 hệ thống sẽ gửi đến mail người quản trị và thực hiện Active -respone để ngăn chặn hành động gây nguy hiểm đến hệ thống.

Tóm lại: Ngay khi có cảnh báo từ hệ thống thì một email sẽ được gửi đến cho người quản trị biết. Dựa vào giao thức SMTP để gửi một email từ localmail qua một mailbox khác. Việc quản trị sẽ dễ dàng hơn nhờ vào thiết lập này từ OSSEC. Người

quản trị sẽ cập nhật được chính xác những thông tin gây ảnh hưởng đến các host trong hệ thống mạng.

IV. Kết quả thực hiện

1. Kết quả đạt được

Sau một thời gian thực hiện đồ án dưới sự hướng dẫn của thầy Nguyễn Hòa. Nhóm đã hiểu được cơ cấu tổ chức và cách thức hoạt động của hệ thống phát hiện và chống xâm nhập sử dụng công nghệ OSSEC trên nền HIDS. Đặc biệt nhóm đã biết cấu hình thực hiện việc gữi nhận log, phân loại log: lấy log nào và không lấy log nào. Đồng thời sau khi nhận được log nhóm đã biết viết các decoded để phân tích log lấy các dữ liệu cần thiết của từng loại log của các chương trình khác nhau. Sau đó thực hiện viết các quy tắc phức (cây rule) để kiểm tra và lọc các loại dữ liệu log để đưa chung vào các cảnh báo phù hợp, và nhóm cũng đã thực hiện được việc thực hiện cảnh báo của OSSEC vào Email của người quản trị để người quản trị có thể ngăn ngừa kịp thời.

2. Hạn chế

Vì đây là đồ án 1 của nhóm , nên nhóm cũng mới bắt đầu tìm hiểu về đề tài. Điều đó dẫn đến còn nhiều chức năng, cách quản lý, và cách phòng chống , ngăn chặn của hệ thống OSSEC mà nhóm chưa nắm bắt được. Nhóm chưa thực hiện được việc cấu hình kích hoạt Active Reponse của hệ thống. Việc này nhóm sẽ thực hiện được tốt trong việc tiếp tục tìm hiểu và phát triển hệ thống OSSEC của một hệ thống mạng thông qua đồ án 2.

V. Hướng phát triển

Xây dựng một hệ thống phát hiện và phòng chống trên mạng LAN. Sau đó phát triển hệ thống ngăn chặn các tấn công mạnh từ bên ngoài ví dụ như : rootkit, hacker... Sau đó sẽ thực hiện trên Cloud dưới sự hướng dẫn của giảng viên.