Bài thực hành:

Đảm bảo an toàn an ninh với Ossec HIDS

1. **Mục đích**

Giúp sinh viên hiểu về khái niệm HIDS, cách thức hoạt động và cách sử dụng các chức năng của Ossec-HIDS.

2. **Yêu cầu đối với sinh viên**

Có kiến thức cơ bản về hệ điều Linux, mô hình Manager-Agent.

Nên thực hiện trước bài thực hành ossec trong labtainer.

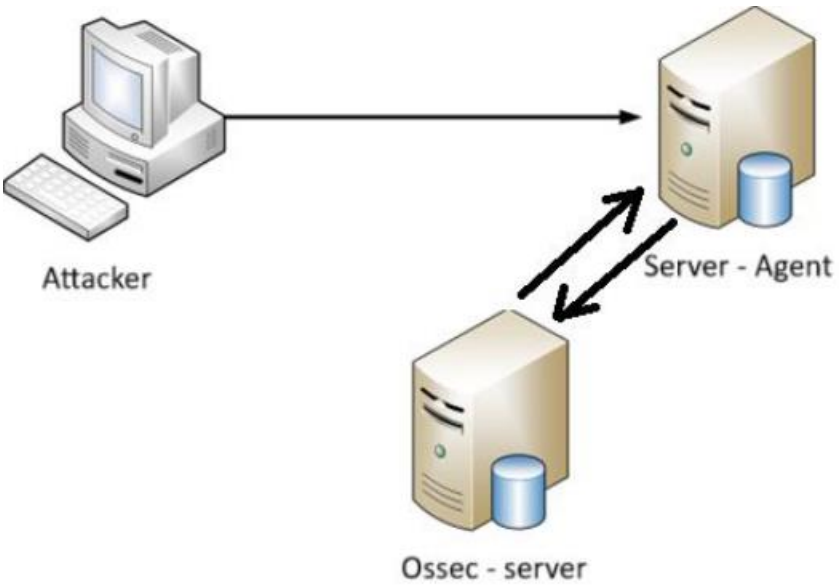
3. **Nội dung thực hành**

- **Khởi động bài lab**: *labtainer -r ossec-hids*

(chú ý: sinh viên sử dụng <TÊN_TÀI_KHOẢN> của mình để nhập thông tin người thực hiện bài lab khi có yêu cầu, để sử dụng khi chấm điểm)

Sau khi khởi động xong hai terminal ảo sẽ xuất hiện, một cái gồm 2 tab là đại diện cho máy ossec-manager: ***manager***, một cái gồm 2 tab là đại diện cho ***server*** và ***attacker***. Biết rằng 3 máy nằm cùng mạng LAN.

- **Mô hình bài lab như sau**:



Tất cả chức năng của ossec-manager và ossec-agent đều nằm trong thư mục /var/ossec.

Theo dõi cảnh báo được sinh ra: *tail -f /var/ossec/logs/alerts/alerts.log*

Nhiệm vụ 1: Kết nối máy server tới máy manager. Ossec-HIDS hoạt động theo mô hình Manager-Agent, Ossec-Manager sẽ nhận thông tin từ các file log mà Agent gửi tới, tiến hành phân giải thông tin (decode), khớp các thông tin này với các quy tắc (rules) được viết trước, từ đó tạo các cảnh báo hoặc các active-response (như chặn ip thực hiện ddos). Do vậy, agent cần được cấu hình và thêm vào danh sách theo dõi của manager. Việc giao tiếp này sử dụng khoá được sinh ra trên manager. Thực hiện như sau:

Trên manager, sử dụng */var/ossec/bin/manage_agents.* Đăng ký thông tin cho agent (ở đây là máy server), sau đó sinh khoá cho agent này.

Trên server, sử dụng */var/ossec/bin/manage_agents.* Thêm khoá được sao chép từ bước trên.

Khởi động lại dịch vụ ossec: */var/ossec/bin/ossec-control restart*

Chú ý: thẻ <server-ip> trong file cấu hình ossec: */var/ossec/etc/ossec.conf* cần là ip của server.

Kiểm tra kết nối bằng câu lệnh: */var/ossec/bin/agent_control -l* hoặc theo dõi trong file alerts.log

Nhiệm vụ 2: Phát hiện chèn mã sql.

Kiểm tra tình trạng dịch vụ apache2 trên máy server.

Như đã nói, Manager theo dõi Agent dựa trên các file log, hay nói cách khác, máy server sẽ gửi các file log này tới manager.

Tìm đường dẫn file access.log của dịch vụ apache2 trên máy server: */var/log*

Trong file cấu hình ossec của máy server: */var/ossec/etc/ossec.conf* , thay thế đường dẫn file access.log sai trong thẻ <location></location>. Khởi động lại dịch vụ ossec trên máy agent.

Trên máy attacker, sử dụng: curl -XGET http://IP_SERVER /users/?id=SE!!!T+*+F!!M+users. Thay thế các dấu ! thành các từ phù hợp. Thông tin attacker gửi một yêu cầu như trên tới máy server sẽ được ghi lại vào file access.log -> chuyển tới manager -> khớp luật -> cảnh báo.

Quan sát file alerts.log để nhận biết cảnh báo đã thành công hay chưa.

Nhiệm vụ 3: Tạo cảnh báo khi có một user cố đăng nhập sai 3 lần trong 120 giây vào một user khác trên máy server.

Trên máy attacker, thực hiện ssh vào user ubuntu trên máy server.

Từ user ubuntu này, thực hiện đăng nhập sai vào một user khác (ngoài root) trên máy: *su* ...

Quan sát file alerts.log, nhận thấy một cảnh báo về việc user ubuntu đăng nhập sai vào một user khác. Quy tắc này có id là: 550! (tìm !).

Từ rule 550! này, ta cấu hình một rule mới. Rule này thực hiện cảnh báo khi rule 550! Lặp lại 3 lần trong vòng 120 giây.

Đây là một rule custom, do đó nên viết rule này vào file /rules/local_rules.xml trên máy manager.

<group name="pam,syslog,">

<rule id="100xxx" level="10" frequency="3" timeframe="1">

<if_matched_sid>x

Possible password guess: 3 failed logins in a short period of time</description>

</group>

Thực hiện điền x (biết các rule từ 100000 trở đi chưa từng tồn tại và các rule id không được trùng nhau). Viết lại các thẻ còn thiếu và các thông số sai bên trên (lưu ý không thay đổi "*Possible password guess: 3 failed logins in a short period of time"*

Bạn vừa có được một rule mới, đây là một rule custome dựa trên rule 550!. Khởi động lại dịch vụ ossec trên manager.

Trên máy attacker, thực hiện ssh vào user ubuntu trên máy server.

Đăng nhập sai vào một user khác (ngoài root) 3 lần trong 120 giây.

Quan sát cảnh báo trên file alerts.log

Nhiệm vụ 4: Gửi thông tin về mức sử dụng CPU tới cho Manager. Sử dụng chức năng command của ossec. Giải thích: máy agent sẽ thực hiện một command, gửi

output của command này tới manager -> khớp luật -> cảnh báo. Nhiệm vụ 4 gồm các bước: xác định câu lệnh để lấy được thông tin cần gửi, cấu hình để gửi output của câu lệnh này tới manager, tạo một rule trên manager để cảnh báo.

Dùng câu lệnh: *top -bn1 | grep Cpu*, sẽ nhận được các thông số của cpu

Tổng của các tham số trên chính là mức sử dụng cpu mà chúng ta cần tìm kiếm.

Dùng tiện ích awk: *awk '{print $2+$4+$6+$12+$14+$16}'*

Kết hợp 2 câu lệnh trên với 2 pipeline, ta được thông tin về cpu cần thiết.

Cấu hình cho server gửi thông tin của output trên tới manager trong file ossec.config

<localfile>

<log_format>full_command</log_format>

<command> </command>

<alias></alias>

<frequency></frequency>

</localfile>

Thêm thẻ trên vào file ossec.config. Điền các thông tin còn thiếu.

Frequency: tần số gửi, tính bằng s

Alias: nhãn, nghĩa là thông về output của command này được gán một nhãn, manager sẽ nhận ra nhãn này trong các rule của nó. Tự chọn nhãn.

Command: câu lệnh đã viết phía trên.

Cuối cùng, cần tạo một custom rule trên máy manager:

<group name="cpu_metric,">

<rule id="1000xx" level="3">

<if_sid>530</if_sid>

<mat>^ossec: output: 'alias'</ch>

<description>CPU usage metrics

</rule>

<rule id="100056" level="12">

<if_sid>1000xx</if_sid>

<regex>[8-9]\d|100</regex>

<description>CPU usage is high</description>

</rule>

Điền alias, các từ còn thiếu, không thay đổi nội dung thẻ <description>

Khởi động lại ossec trên cả 2 máy

Quan sát alerts.log

Nhiệm vụ 5: Cảnh báo khi một dịch vụ quan trọng không hoạt động. Ở đây dịch vụ quan trọng được đề xuất là apache2. Và chức năng ossec sử dụng cũng là command tương tự nhiệm vụ 4. Trình tự các bước không thay đổi.

Kiểm tra dịch vụ apache2 có đang chạy hay không, nếu không khởi động dịch vụ.

Sử dụng câu lệnh: *ps -auxw | grep "apache2"* để hiển thị thông tin về tiến trình liên quan tới dịch vụ apache2. Tìm một *từ khoá* quan trọng gắn với dịch vụ này (có thể là user chạy dịch vụ).

Sau khi cấu hình xong trên máy agent, chuyển sang máy manager, tạo một rule custom để cảnh báo khi apache2 không chạy.

<rule id="1000xx" level="6">

<if_sid>530</if_sid>

<description>Important process not running.</description>

</rule>

<rule id="100011" level="0">

<if_sid>1000xx</if_sid>

<match>từ khoá</match>

<description>Processes running as expected</description>

</rule>

Dựa trên nhiệm vụ 4, điền các thẻ còn thiếu, các thông tin còn thiếu. Không thay đổi thẻ *<description> </description>*

Khởi động lại ossec trên cả 2 máy

Tắt dịch vụ apache2 trên server

Quan sát alerts.log

Nhiệm vụ 6: Chặn user cố đăng nhập vào một user khác trên máy server. Đây là một tính năng IPS của Ossec được gọi là active-response giúp phản ứng tự động lại các cuộc tấn công, các lỗi, … Tuy vậy, sử dụng tính năng IPS này của Ossec cần thận trọng và hạn chế do việc tự động hoá có thể sai, gây ra nhiều vấn đề không đáng có.

Hoạt động của active-response được hiểu như sau: máy agent gửi thông tin (file log) tới máy manager, thông tin này được so sánh với các rule, nếu các rule được kích hoạt này là điều kiện kích hoạt cho một active-respone -> manager gửi lệnh thực thi một hành động nào đó tới agent. Agent thực hiện lệnh được chỉ định. Các lệnh này là các file script được viết sẵn lưu trên agent liên quan tới các tính năng chặn người dùng, chặn địa chỉ ip, …

Active-response trong ossec tồn tại 2 trạng thái: stateful-có giới hạn thời gian và stateless-vĩnh viễn. Trạng thái này được định nghĩa bằng thẻ <timeout_allowed></timeout_allowed> trong file ossec.config trên manager.

Nhiệm vụ 6 sẽ chặn user ở nhiệm vụ 3.

Trong file ossec.config, thêm các thẻ:

```
<active-response>
  <command></command>
  <location>local
  <rules_id>
  <timeout>
</active-response>
```

Thêm các thẻ, các thông tin còn thiếu

command: tìm kiếm trong chính file ossec.config command liên quan tới disable account

rules id: id của rule ở nhiệm vụ 3

timeout: thời gian giới hạn mà tài khoản bị vô hiệu hoá

Sau khi thêm xong, khởi động lại dịch vụ ossec trên máy manager

Thực hiện lại nhiệm vụ 3, quan sát file alerts.log

Xem output của câu lệnh *passwd --status ubuntu* có flag L (lock) hay không.

Nhiệm vụ 7: Chặn IP vét cạn ssh vào máy server

Trên máy attacker, sử dụng hydra để vét cạn ssh server (có thể thoát sau 10s):

sudo hydra -t 4 -l user -P pass.txt IP_SERVER ssh

Quan sát file alerts.log, tìm được rule có nội dung "Multiple SSHD authencation …". Ghi lại id của rule này

Thực hiện tương tự nhiệm vụ 6, thay thế rule id và và command được thực hiện liên quan tới firewall drop.

Khởi động lại dịch vụ ossec trên cả 2 máy.

Thực hiện lại bước 1. Thoát quá trình vet cạn, thực hiện ping tới server, kết quả thu được là gì ?

- **Kết thúc bài lab:** *stoplab ossec-hids*
- **Khởi động lại bài lab:** *labtainer -r ossec-hids*