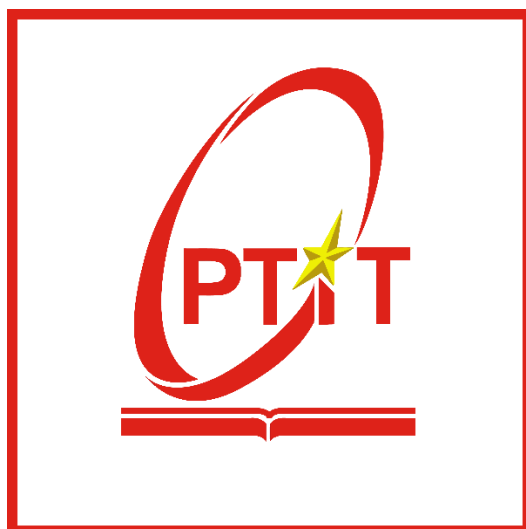


HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA AN TOÀN THÔNG TIN



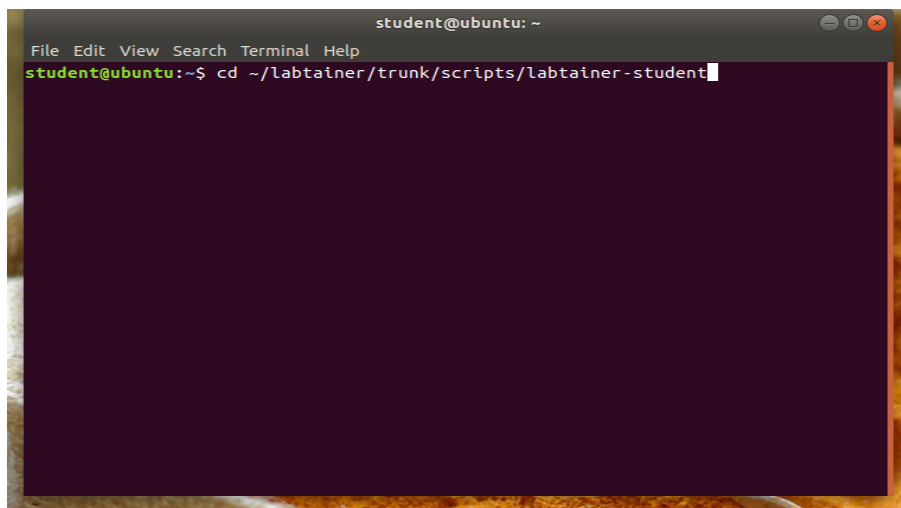
BÀI THỰC HÀNH 03
MÔN HỌC: CƠ SỞ AN TOÀN THÔNG TIN

Giảng viên : Nguyễn Ngọc Diệp
Sinh viên : Lê Anh Tuấn
Lớp : D21CQAT01-B
Nhóm : 03
Mã sinh viên : B21DCAT205
Số điện thoại : 0369288612

Tháng 11/2023

I. Bài thực hành: Danh sách điều khiển truy cập trên Linux

- Bật terminal và gõ dòng lệnh `cd ~/labtainer/trunk/scripts/labtainer-student` để chuyển vào thư mục labtainer-student



- Trong thư mục labtainer-student, ta tiếp tục gõ câu lệnh `labtainer -r acl`
- Tiếp tục ta được yêu cầu nhập email, ta nhập mã sinh viên

```
student@ubuntu:~/labtainer/labtainer-student$ labtainer -r acl
latest: Pulling from labtainers/acl.acl.student
0570a56d4eb9: Pull complete
dbb921ae0acf: Pull complete
7f3b93fa3fec: Pull complete
29300ef5b2f9: Pull complete
7a097d42bedc: Pull complete
7ee6dd60ef07: Pull complete
93a9868b3fcd: Pull complete
41ef24b6fc41: Pull complete
d6e06ef07515: Pull complete
1484f13502bb: Pull complete
c88256823f1a: Pull complete
05bfb7e0c334: Pull complete
a5f9318f4119: Pull complete
3352e201306f: Pull complete
a65f14a2a0ce: Pull complete
1dfc9b5a4a2d: Pull complete
Digest: sha256:dac6694c72f86aea073585f861a6fee33351a26048470a7777e1f1df0bee0fe4
Status: Downloaded newer image for labtainers/acl.acl.student:latest
non-network local connections being added to access control list

Please enter your e-mail address: [B21DCAT205]
```

- Ấn enter

```
student@ubuntu: ~/labtainer/trunk/scripts/labtainer-student
File Edit View Search Terminal Help
student@ubuntu:~$ cd ~/labtainer/trunk/scripts/labtainer-student
student@ubuntu:~/labtainer/trunk/scripts/labtainer-student$ labtainer -r telnetlab
non-network local connections being added to access control list

Please enter your e-mail address: [B21DCAT205]B21DCAT205
Starting the lab, this may take a moment...
Started 2 containers, 2 completed initialization. Done.

The lab manual is at
  file:///home/student/labtainer/trunk/labs/telnetlab/docs/telnet.pdf
You may open this manual by right clicking
and select "Open Link".

Press <enter> to start the lab

student@ubuntu:~/labtainer/trunk/scripts/labtainer-student$
```

- Sau khi khởi động bài lab, 3 thiết bị đầu cuối ảo sẽ được bật chế độ login, hãy đăng nhập theo các tài khoản dưới đây:

User	Password
bob	password4bob
alice	password4alice
harry	password4harry

The image shows three terminal windows side-by-side, each representing a different user logging into a system. The top window is for 'harry@acl: /'. It shows a list of IP addresses and their status (Pull complete), followed by 'acl login:' prompts, 'Login timed out after 60 seconds.' messages, and successful logins for 'harry' and 'harry@acl ~]\$'. The bottom-left window is for 'bob@acl ~]\$'. It shows 'acl login:' prompts, 'Login timed out after 60 seconds.' messages, and successful logins for 'bob' and 'bob@acl ~]\$'. The bottom-right window is for 'alice@acl:~'. It shows 'acl login:' prompts, 'Login timed out after 60 seconds.' messages, and successful logins for 'alice' and 'alice@acl ~]\$'.

Nhiệm vụ 1: Xem lại các quyền trên các file hiện có

- Trên terminal “Alice”, đến thư mục /shared data và liệt kê các quyền trên file, thư mục:

```
cd /shared_data
```

```
ls -l
```

```
[alice@acl ~]$ cd /shared_data
[alice@acl shared_data]$ ls -l
total 24
-rw-rw----+ 1 root  root   13 Jan 27  2020 accounting.txt
drwxr-xr-x  1 alice alice 4096 Jan 27  2020 alice
drwxr-xr-x  1 bob  bob   4096 Jan 27  2020 bob
[alice@acl shared_data]$
```

- Chúng ta sẽ thấy các quyền trên file accounting.txt và 2 thư mục. Kiểm tra xem “Alice” có thể xem nội dung file accounting.txt hay không, ta sử dụng lệnh *cat* để đọc file. Ta sẽ thấy Alice có quyền xem nội dung file.

```
cat accounting.txt
```

```
[alice@acl shared_data]$ cat accounting.txt
some numbers
```

- Ta kiểm tra quyền ACL của file này sử dụng lệnh:

```
getfacl accounting.txt
```

```
[alice@acl shared_data]$ getfacl accounting.txt
# file: accounting.txt
# owner: root
# group: root
user::rw-
user:alice:r--
user:harry:rw-
group::r--
mask::rw-
other::---
```

- Ta thấy rằng harry có quyền thực thi nên ta chuyển tới terminal của harry và thực hiện câu lệnh sau để kiểm tra điều đó.

```
echo "more stuff" >> /shared_data/accounting.txt
```

```
[harry@acl shared_data]$ echo "more stuff" >> /shared_data/accounting.txt
[harry@acl shared_data]$
```

- Quay trở lại terminal “alice”, thực hiện lệnh sửa đổi file ở trên để xác nhận rằng “alice” không có quyền sửa đổi file này.

```
[alice@acl shared_data]$ echo "more stuff" >> /shared_data/accounting.txt
-bash: /shared_data/accounting.txt: Permission denied
```

Nhiệm vụ 2: Cài đặt ACL trên một file

- Với tư cách là người dùng Bob, sử dụng lệnh setfacl để cho phép Alice đọc file /shared_data/bob/bobstuff.txt.

```
setfacl -m "u:alice:r" /shared_data/bob/bobstuff.txt
```

```
[bob@acl shared_data]$ setfacl -m "u:alice:r" /shared_data/bob/bobstuff.txt
[bob@acl shared_data]$
```

- Xác nhận khả năng đọc file của alice sử dụng câu lệnh *cat bob/bobstuff.txt*

```
[alice@acl shared_data]$ cat bob/bobstuff.txt
bob's stuff
[alice@acl shared_data]$
```

- Harry không có khả năng đọc file này

```
[harry@acl shared_data]$ cat bob/bobstuff.txt
cat: bob/bobstuff.txt: Permission denied
```

- Checkwork sau khi hoàn thành nhiệm vụ 1 và nhiệm vụ 2.

```
student@ubuntu:~/labtainer/labtainer-student$ checkwork acl
Results stored in directory: /home/student/labtainer_xfer/acl
Labname acl

Student      |      did_trojan |      bob_stuff_acl |      alice_default |
===== | ===== | ===== | ===== |
B21DCAT205   |              |              |              |
What is automatically assessed for this lab:
  bob_stuff_acl: Changed ACL so alice can read bob's stuff
  alice_default: Bob got default read access to newly created alice file
  did_trojan: Does not check that result is readable, but does confirm fun altered
               to read the accounting.txt file, and was run by alice.
```

Nhiệm vụ 3: Cài đặt ACL mặc định cho 1 thư mục

- Ta nhận thấy bob không có một quyền nào trong đây

```
[alice@acl shared_data]$ cd /shared_data/alice/
[alice@acl alice]$ ls
alicestuff.txt
[alice@acl alice]$ getfacl /shared_data/alice/
getfacl: Removing leading '/' from absolute path names
# file: shared_data/alice/
# owner: alice
# group: alice
user::rwx
group::r-x
other::r-x
```

- Ta cấp quyền cho bob có quyền đọc sử dụng câu lệnh
setfacl -dm "u:bob:r" /shared_data/alice/

```
[alice@acl alice]$ setfacl -dm "u:bob:r" /shared_data/alice/
[alice@acl alice]$ getfacl /shared_data/alice/
getfacl: Removing leading '/' from absolute path names
# file: shared_data/alice/
# owner: alice
# group: alice
user::rwx
group::r-x
other::r-x
default:user::rwx
default:user:bob:r--
default:group::r-x
default:mask::r-x
default:other::r-x
```

- Ta tạo 1 file mới và alice có quyền đọc file đó sử dụng câu lệnh
echo "B21DCAT205" > test.txt

```
[alice@acl alice]$ echo "B21DCAT205" > test.txt
[alice@acl alice]$ cat test.txt
B21DCAT205
[alice@acl alice]$
```

- Bob cũng có quyền đọc file

```
[bob@acl ~]$ cat /shared_data/alice/test.txt
B21DCAT205
[bob@acl ~]$
```

- Vì *default:other::r-x* nên harry cũng có quyền đọc file này

```
[harry@acl ~]$ cat /shared_data/alice/test.txt
B21DCAT205
```

- Để xóa quyền của harry ta sử dụng câu lệnh
setfacl -dm "u:bob:r" /shared_data/alice/

- Sau đó sử dụng câu lệnh để thiết lập other có quyền được thực thi(x).
setfacl -m "o:--x" /shared_data/alice/

```
[alice@acl alice]$ setfacl -dm "o:--x" /shared_data/alice/
[alice@acl alice]$ setfacl -m "o:--x" /shared_data/alice/
[alice@acl alice]$ getfacl /shared_data/alice/
getfacl: Removing leading '/' from absolute path names
# file: shared_data/alice/
# owner: alice
# group: alice
user::rwx
group::r-x
other::--x
default:user::rwx
default:user:bob:r--
default:group::r-x
default:mask::r-x
default:other:---
```

- Kiểm tra lại bằng cách tạo một file mới test2.txt.
echo "Le Anh Tuan" > test2.txt

```
[alice@acl alice]$ echo "Le Anh Tuan" > test2.txt
[alice@acl alice]$
```

- Ta thấy bob có quyền đọc file này.

```
[bob@acl ~]$ cat /shared_data/alice/test2.txt
Le Anh Tuan
[bob@acl ~]$
```

- Nhưng Harry không có quyền đọc file này.

```
[harry@acl ~]$ cat /shared_data/alice/test2.txt
cat: /shared_data/alice/test2.txt: Permission denied
[harry@acl ~]$
```

- Checkwork sau khi hoàn thành nhiệm vụ 3

```
student@ubuntu:~/labtainer/labtainer-student$ checkwork acl
Results stored in directory: /home/student/labtainer_xfer/acl
Labname acl

Student      |      did_trojan |    bob_stuff_acl |    alice_default |
===== | ===== | ===== | ===== |
B21DCAT205   |              |              Y   |              Y   |
What is automatically assessed for this lab:
  bob_stuff_acl: Changed ACL so alice can read bob's stuff
  alice_default: Bob got default read access to newly created alice file
  did_trojan: Does not check that result is readable, but does confirm fun altered
               to read the accounting.txt file, and was run by alice.
```

Nhiệm vụ 4: Trojan Horses

- Đọc thử file fun của bob ở máy alice

```
[alice@acl alice]$ cd /shared_data/bob/  
[alice@acl bob]$ ls  
bobstuff.txt fun  
[alice@acl bob]$ ./fun
```

A detailed ASCII art illustration of a motorcycle, viewed from the side. The drawing uses various symbols like dots, pipes, and dashes to create texture and shading. It shows the front headlight, handlebars, fuel tank, engine, exhaust pipe, and rear wheel.

```
[alice@acl bob]$
```

- Kiểm tra lại Bob không có quyền đọc file accounting.txt

```
[bob@acl ~]$ cat /shared_data/accounting.txt
cat: /shared_data/accounting.txt: Permission denied
[bob@acl ~]$
```

- Sử dụng câu lệnh *vim fun* để tiến hành sửa file script

```
[bob@aql ~]$ cd /shared_data/bob/
[bob@aql bob]$ ls
bobstuff.txt  fun
[bob@aql bob]$ vim fun
```

- Thêm vào một hàm có tên là trojan, sau đó định nghĩa nó như hình dưới với việc khi mà alice mở file thì nội dung của accounting.txt sẽ được copy sang tệp có tên trojan.txt và cấp quyền bob có thể đọc được tệp đó


```
[bob@acl bob]$ cat trojan.txt
some numbers
more stuff
[bob@acl bob]$
```

- Xóa đi file để tiến hành thử nghiệm với harry vì harry cũng có quyền đọc file accounting.txt

```
rm trojan.txt
```

```
[bob@ac1 bob]$ rm trojan.txt
[bob@ac1 bob]$
```

- Thử lại với harry.

```
[harry@acl ~]$ cd /shared_data/bob
[harry@acl bob]$ ls
bobstuff.txt  fun
[harry@acl bob]$ ./fun
      .-.
     /:.  ' \
    /:::-. \.-" ""-;  .-:::..  :::\
   /:::' \  \ \      \ '  \::'  :::\
  /  _-'  '  |  /  (o|o)  \  \::'  :::\
 |  :::.'  |  /  (.-.)  .::\
 \  _-'  '  |  /  |I|  \  \::'  :::\
  \  _-'  '  |  /  |I|  \  \::'  :::\
   \  _-'  '  |  /  |I|  \  \::'  :::\
    \  _-'  '  |  /  |I|  \  \::'  :::\
     \  _-'  '  |  /  |I|  \  \::'  :::\
      .-.
[harry@acl bob]$ ls
bobstuff.txt  fun  trojan.txt
[harry@acl bob]$
```

- Vì harry có quyền đọc file accounting nên bob có thể đọc được file trojan.txt tương tự.

```
[bob@acl bob]$ cat trojan.txt
some numbers
more stuff
[bob@acl bob]$
```

- Checkwork sau khi hoàn thành nhiệm vụ 4

```

student@ubuntu:~/labtainer/labtainer-student$ checkwork acl
Results stored in directory: /home/student/labtainer_xfer/acl
Labname acl

Student          |      did_trojan |    bob_stuff_acl |    alice_default |
=====          | =====          | =====          | =====          |
B21DCAT205       |                  Y |                  Y |                  Y |
What is automatically assessed for this lab:
  bob_stuff_acl: Changed ACL so alice can read bob's stuff
  alice_default: Bob got default read access to newly created alice file
  did_trojan: Does not check that result is readable, but does confirm fun altered
               to read the accounting.txt file, and was run by alice.

```

Kiểm tra kết quả

ID	Thời gian	Bài tập	Kết quả
15312	2023-11-23 11:28:28	Danh sách điều khiển truy cập	3/3 (AC)

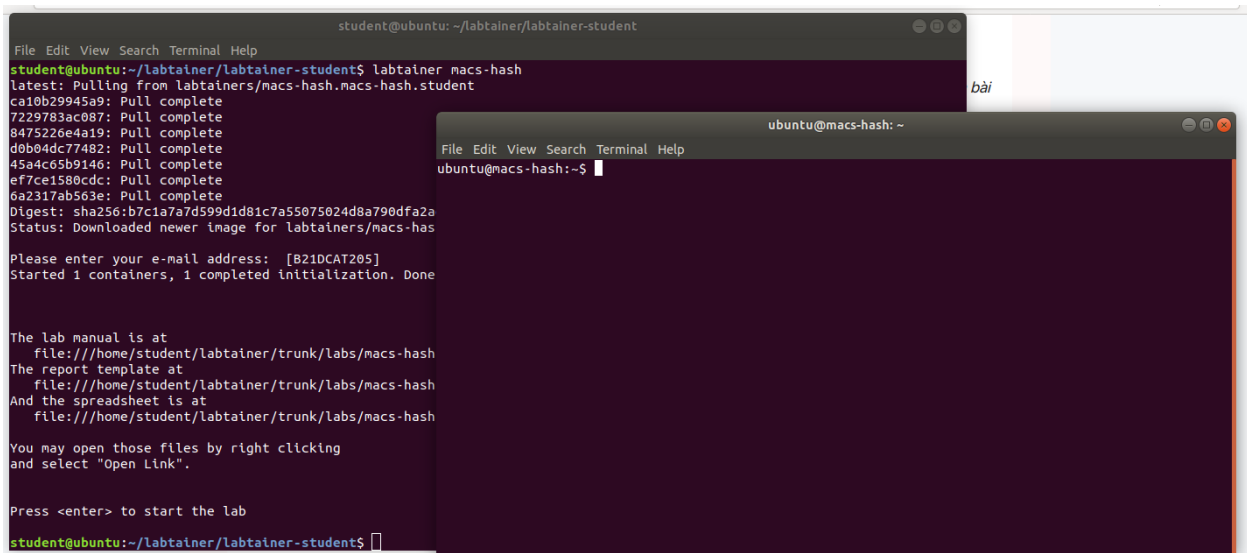
Kết luận

- Ta đã biết cách xem các quyền ACL trên file và thư mục
- Cài đặt thành công ACL trên một file
- Cài đặt mặc định ACL trên thư mục
- Biết cách sử dụng ACL để cấp quyền, xóa bỏ quyền cho file, thư mục
- Cần lưu ý một số tình huống đọc file lạ có thể sẽ bị rò rỉ thông tin.

II. Bài thực hành: Tìm hiểu về hàm băm và mã xác thực thông điệp MACs

Khởi động bài lab:

labtainer -r macs-hash



```
student@ubuntu: ~/labtainer/labtainer-student
student@ubuntu:~/labtainer/labtainer-student$ labtainer macs-hash
latest: Pulling from labtainers/mac-hash:macs-hash.student
ca10b29945a9: Pull complete
7229783ac087: Pull complete
8475226e4a19: Pull complete
d0b04dc77482: Pull complete
45a4c65b9146: Pull complete
ef7ce1588cdc: Pull complete
6a2317ab563e: Pull complete
Digest: sha256:b7c1a7a74599d1d01c7a55075024d8a790dfa2a
Status: Downloaded newer image for labtainers/mac-hash
Please enter your e-mail address: [B21DCAT205]
Started 1 containers, 1 completed initialization. Done

The lab manual is at
file:///home/student/labtainer/trunk/labs/mac-hash
The report template at
file:///home/student/labtainer/trunk/labs/mac-hash
And the spreadsheet is at
file:///home/student/labtainer/trunk/labs/mac-hash

You may open those files by right clicking
and select "Open Link".

Press <enter> to start the lab
student@ubuntu:~/labtainer/labtainer-student$
```

Nhiệm vụ 1: Trong nhiệm vụ này, ta sẽ không sử dụng OpenSSL để tạo bản tóm lược (digest) vì có nhiều cách tạo dễ dàng hơn trên Unix. Thay vào đó, ta sử dụng lệnh Shasum mặc định hỗ trợ SHA-1 với đầu ra 160 bit, cũng như các đầu ra SHA-2 dưới các tùy chọn khác. Ta có thể xem các tùy chọn bằng lệnh dưới:

shasum --help / less

```
ubuntu@macs-hash: ~
File Edit View Search Terminal Help
ubuntu@macs-hash:~$ shasum --help
Usage: shasum [OPTION]... [FILE]...
Print or check SHA checksums.
With no FILE, or when FILE is -, read standard input.

-a, --algorithm 1 (default), 224, 256, 384, 512, 512224, 512256
-b, --binary      read in binary mode
-c, --check       read SHA sums from the FILES and check them
-t, --text       read in text mode (default)
-U, --UNIVERSAL   read in Universal Newlines mode
                  produces same digest on Windows/Unix/Mac
-0, --01         read in BITS mode
                  ASCII '0' interpreted as 0-bit,
                  ASCII '1' interpreted as 1-bit,
                  all other characters ignored
-p, --portable    read in portable mode (to be deprecated)

The following two options are useful only when verifying checksums:
-s, --status      don't output anything, status code shows success
-w, --warn        warn about improperly formatted checksum lines

-h, --help        display this help and exit
-v, --version      output version information and exit

When verifying SHA-512/224 or SHA-512/256 checksums, indicate the
```

- Tạo ra một bản tóm lược 160 bit SHA-1 với tên foo.txt

shasum -a 1 foo.txt

```
ubuntu@macs-hash:~$ shasum -a 1 foo.txt
e225a942b33cebf9ebdedc405622298882dea22d foo.txt
```

- Tạo một tệp văn bản test.txt trên hệ thống với nội dung bất kỳ sử dụng câu lệnh *echo "B21DCAT205" > test.txt*

```
ubuntu@macs-hash:~$ echo "B21DCAT205" > test.txt
ubuntu@macs-hash:~$ ls
collide1.sh collide2.py declare.txt foo.txt test.txt
ubuntu@macs-hash:~$ cat test.txt
B21DCAT205
ubuntu@macs-hash:~$
```

- Sử dụng Shasum để thử 7 thuật toán mã hóa với tệp đã tạo ở trên lần lượt với các câu lệnh:

shasum -a 1 test.txt

shasum -a 224 test.txt

shasum -a 256 test.txt

shasum -a 384 test.txt

shasum -a 512 test.txt

shasum -a 512224 test.txt

shasum -a 512256 test.txt

```

ubuntu@macs-hash:~$ shasum -a 1 test.txt
e3a9c1def5cc297af8c21f0009a6e371df4291ff test.txt
ubuntu@macs-hash:~$ shasum -a 224 test.txt
c6aeaf7c13e2236cc52b5f026122399a3dcdbcdbb10c39e7271e605ce test.txt
ubuntu@macs-hash:~$ shasum -a 256 test.txt
c0b115ec747054e7cd208c85c7ee7a0220f5d8cbf0910528917a4ad63fd10d07 test.txt
ubuntu@macs-hash:~$ shasum -a 384 test.txt
24083a689e1d1e17e2052555b01e71f4ec324b20011cde6b149e7cea4aa499d8efb134b13dc4bd8aea596dd3f2ad9c0b test.txt
ubuntu@macs-hash:~$ shasum -a 512 test.txt
3d2e80f7d5729dd805758ab470b4feebdc6310ba61b8f8eee083e5c1792dbfff41fe7c3fcc3f7917175ea2110b048e2b80d77faff8e95e08db9d34462ec9484a test.txt
ubuntu@macs-hash:~$ shasum -a 512224 test.txt
538045d9289ba9373af431233de39b39fcc857f16078c0e3d94376a8 test.txt
ubuntu@macs-hash:~$ shasum -a 512256 test.txt
82bd0d6d886f550f801db741608fb8a2e57e217f4ac8f29aba44884cf3ba75a3 test.txt
ubuntu@macs-hash:~$

```

- Checkwork sau khi hoàn thành nhiệm vụ 1

```

student@ubuntu:~/labtainer/labtainer-student$ checkwork macs-hash
Results stored in directory: /home/student/labtainer_xfer/macs-hash
Labname macs-hash

Student | collide1_count | collide2_count | shasum_count | did_7_shasum | hashed_floppy | hashed_iou | openssl_key |
=====|=====|=====|=====|=====|=====|=====|=====|
B21DCAT205 | 0 | 0 | 8 | Y | | | |
What is automatically assessed for this lab:

did_7_shasum: Did at least 7 different sha checksums
hashed_floppy: hashed the floppy
hashed_iou: hashed the iou file
collide1_count, collide2_count, shasum_count: Count of program invocation
openssl_key: brute force ran openssl to discover the key

```

Nhiệm vụ 2: Kiểm tra bản tóm lược

Trong nhiệm vụ này, chúng ta sẽ học cách kiểm tra tính toàn vẹn của file tải xuống bằng cách sử dụng hàm băm. Sử dụng trình duyệt lynx để tải xuống 2 file.

- Gõ *"lynx verydodgy.com"*
- Chọn enter + d để tải file

```
ubuntu@macs-hash: ~  
File Edit View Search Terminal Help  
Directory listing for /  
-----  
* floppy57.fs  
* SHA256.sdx  
-----  
Commands: Use arrow keys to move, '?' for help, 'q' to quit, '<-' to go back.  
Arrow keys: Up and Down to move. Right to follow a link; Left to go back.  
H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list
```

- Tiếp tục ấn mũi tên xuống, sau đó ấn enter 2 lần

```
ubuntu@macs-hash: ~  
File Edit View Search Terminal Help  
<<< Download Options (Lynx Version 2.8.9dev.8), help  
Downloaded link: http://verydodgy.com/floppy57.fs  
Suggested file name: floppy57.fs  
Standard download options:  
  Save to disk  
Local additions:  
Commands: Use arrow keys to move, '?' for help, 'q' to quit, '<-' to go back.  
Arrow keys: Up and Down to move. Right to follow a link; Left to go back.  
H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list
```

- Sau đó ấn q và mũi tên quay lại để trở về.

```
ubuntu@macs-hash: ~  
File Edit View Search Terminal Help  
<<< Download Options (Lynx Version 2.8.9dev.8), help  
Downloaded link: http://verydodgy.com/floppy57.fs  
Suggested file name: floppy57.fs  
Standard download options:  
  Save to disk  
Local additions:  
  
Commands: Use arrow keys to move, '?' for help, 'q' to quit, '<-' to go back.  
  Arrow keys: Up and Down to move. Right to follow a link; Left to go back.  
  H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list
```

- Tiếp tục ấn mũi tên xuống và thực hiện tương tự

```
ubuntu@macs-hash: ~  
File Edit View Search Terminal Help  
Directory listing for /  
-----  
* floppy57.fs  
* SHA256.sdx  
-----
```

```
ubuntu@macs-hash: ~  
File Edit View Search Terminal Help  
<<< Download Options (Lynx Version 2.8.9dev.8), help Download Options  
Downloaded link: http://verydodgy.com/SHA256.sdx  
Suggested file name: SHA256.sdx  
Standard download options:  
  Save to disk  
Local additions:  
  
Enter a filename: SHA256.sdx  
Arrow keys: Up and Down to move. Right to follow a link; Left to go back.  
H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list
```

- Kiểm tra các file vừa cài đặt sử dụng câu lệnh `ls`

```
ubuntu@macs-hash:~$ ls  
SHA256.sdx collide1.sh collide2.py declare.txt floppy57.fs foo.txt test.txt  
ubuntu@macs-hash:~$
```

- Tạo bản tóm lược SHA256 với file vừa tải xuống. Kiểm tra bản tóm lược tạo ra và file *SHA256.sdx* ta thấy khớp nhau.

```
ubuntu@macs-hash:~$ shasum -a 256 SHA256.sdx  
1d5134ed8e8f8279554d9ed423502cfb8f0594e5a58ef98a06214737c7d1bb10 SHA256.sdx
```



```
ubuntu@macs-hash:~$ cat SHA256.sdx
SHA256 (INSTALL.amd64) = a4604b4982cb2d1546fcef31c351f97d7203538d77fca11012b857410c8d2260
SHA256 (base57.tgz) = e48291b0fe2ec9965ed0d574d96bc8fd8a075236be8d191538aaadc8a9c8bbcc
SHA256 (bsd) = 6382680ee4bd05dcaccc7b3c165a7644ca65cd0a1fa2670a9c7860cadcb8f7e2
SHA256 (bsd.mp) = 7386dada2a05298d6c76f642859bd06ca6f1f604049b18080bac9cc2d4be7643
SHA256 (bsd.rd) = a964fde0855e3585113f4a92627699a697d0e89d2c9a0cba4b7b0c04b77a5101
SHA256 (cd57.iso) = 691868e505aadde6feba0c0ba530bb99aad86e62c998a914e03e84c9bfb3b9e5
SHA256 (cdboot) = 12a00c426830f5fdc3afc8a6809cfec238d4fc245d57a4724a26bb545d671449
SHA256 (cldr) = 4e1953342e0e620e375a9404ee8bc419596e0a4f64a3d9a6237e3bd6fe3f4ad4
SHA256 (comp57.tgz) = de232f696d9d310aa80951ed97006f9c2bb03500822fffc76c6f642171827876e
SHA256 (floppy57.fs) = 91dbf055b06c2c54ad54b6cc83c3fb2c1f2944ae732f8075cd7bd39340e5ca50
SHA256 (game57.tgz) = 414679b2ff204f23ae1be683189563f0d868ecda59d321eff74a444ab170eff0
SHA256 (install57.fs) = d61f5d8bcd8c860c2dbe92c827be515f5d34fb8ab31884145e5dce535a5a73b3
SHA256 (install57.iso) = 3f714d249a6dc8f40c2fc2fccea8ef9987e74a2b81483175d081661c3533b59a
SHA256 (man57.tgz) = 051f7a2ddc35a5cdddf8df1441bf85f982c0dd829e6e43d1716497bfe02f74c
SHA256 (miniroot57.fs) = 9d14ecfbc4cb4fdbc9e8d6db8d2711c3a5704806e47e49bfa729138089bc4c97
SHA256 (pxeboot) = 7bc2cb6401cb8cf6add64414743eaa9df77d63831d78afb4d13d17f2b6eb6d90
SHA256 (xbase57.tgz) = bf54381e494ed249dbefdbf4bb13223c3985200ff2db974298cd8bdc15e38e3e
SHA256 (xfont57.tgz) = 938014c3ae1bbfbb3404aa2f02e5bc77724ffdd6a53926f58d8b2b1cfd580283
SHA256 (xserv57.tgz) = cc3a8aa01b0361d9633a12ad3b2b4209756ac40bd9f6c02d12bcd2489aadf8d4
SHA256 (xshare57.tgz) = 0cecc83cc2c826b09a4aea07bde8a0001917c4d261c1994a5af056014394e984
ubuntu@macs-hash:~$
```

Nhiệm vụ 3: Tìm hiểu về “Avalanche Effect”

“Avalanche Effect” mô tả hiện tượng một thay đổi nhỏ ở file đầu vào cũng sẽ làm thay đổi hoàn toàn đến bản tóm lược đầu ra.

- Tạo một file với tên *iou.txt* có nội dung “*Bob owes me 200 dollars*”.

```
ubuntu@macs-hash:~$ echo "Bob owes me 200 dollars" >iou.txt
ubuntu@macs-hash:~$ cat iou.txt
Bob owes me 200 dollars
ubuntu@macs-hash:~$
```

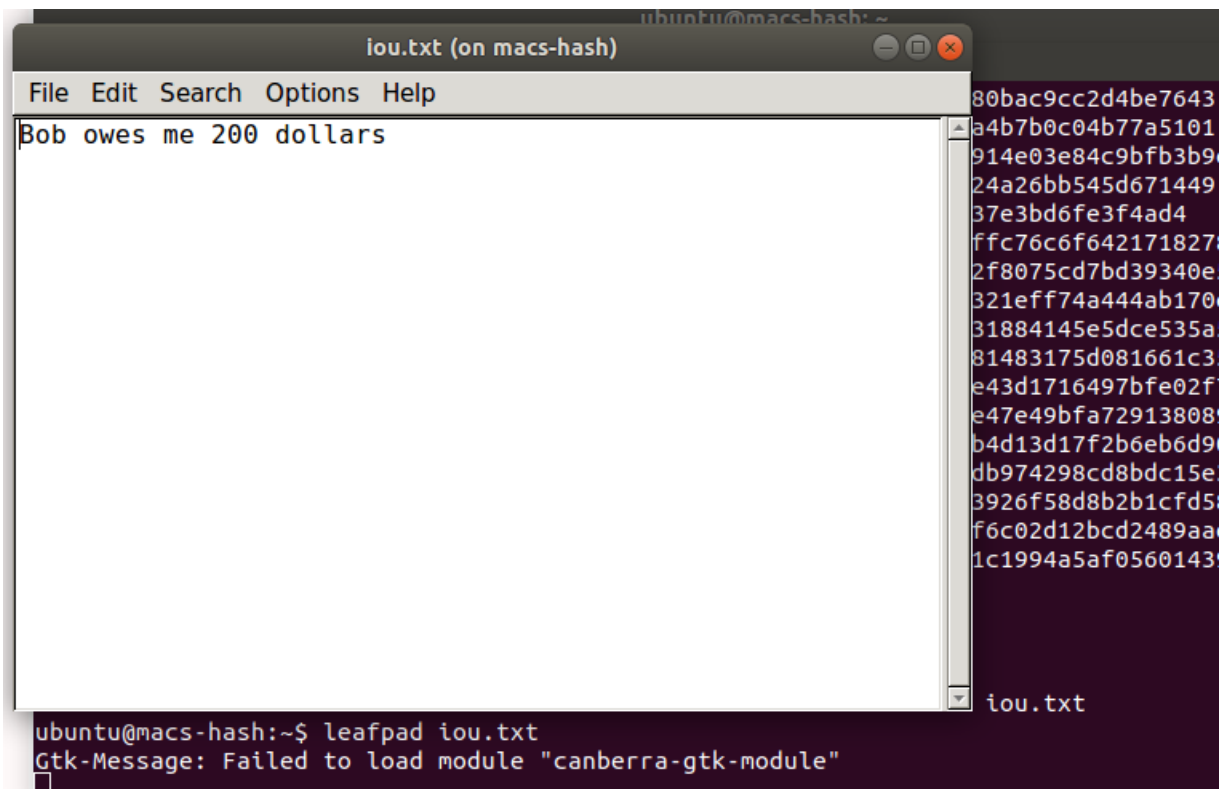
- Tạo bản tóm lược SHA256 của file *iou.txt*

shasum -a 256 iou.txt

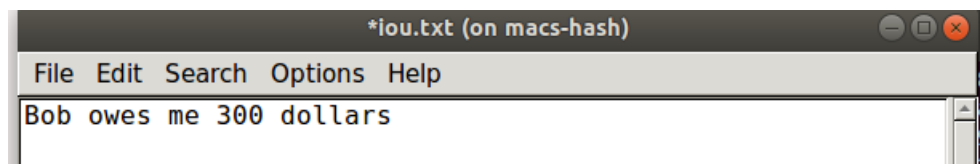
```
ubuntu@macs-hash:~$ shasum -a 256 iou.txt
5c1929627f046090aaf1f11058456d9203cd5415cb09cf08382f5caf59ae9bb1 iou.txt
ubuntu@macs-hash:~$
```

- Mở file *iou.txt* bằng chương trình chỉnh sửa, ví dụ leafpad

leafpad iou.txt



- Thay đổi số “2” thành “3”. Điều này sẽ làm 1 bit bị thay đổi và lưu các thay đổi và thoát khỏi chương trình chỉnh sửa.



- Kiểm tra lại file vừa sửa

```

ubuntu@macs-hash:~$ cat iou.txt
Bob owes me 300 dollars
ubuntu@macs-hash:~$

```

- Tạo bản tóm lược SHA256 khác đối với tệp *iou.txt* đã sửa đổi.
shasum -a 256 iou.txt

```

ubuntu@macs-hash:~$ shasum -a 256 iou.txt
337826a142761d83997b150c0090bc7a72ac6f4318bfacdeaf1986c3b704ae1b iou.txt
ubuntu@macs-hash:~$

```

- Tương tự đối với file floppy

```
ubuntu@macs-hash:~$ shasum -a 256 floppy57.fs
91dbf055b06c2c54ad54b6cc83c3fb2c1f2944ae732f8075cd7bd39340e5ca50 floppy57.fs
ubuntu@macs-hash:~$
```

Ta thấy rằng khi chỉ thay đổi 1 ký tự, bản tóm lược sẽ thay đổi hoàn toàn, để bản tóm lược trùng khớp với bản gốc thì ta chỉ cần thay đổi đoạn văn bản trở lại như ban đầu sẽ khôi phục được bản tóm lược.

- Checkwork sau khi hoàn thành nhiệm vụ 3

```
student@ubuntu:~/labtainer/labtainer-student$ checkwork macs-hash
Results stored in directory: /home/student/labtainer_xfer/macs-hash
Labname macs-hash
```

Student	collide1_count	collide2_count	shasum_count	did_7_shasum	hashed_floppy	hashed_tou	openssl_key
B21DCAT205	0	0	14	Y	Y	Y	

Nhiệm vụ 4: Tìm hiểu về Second Pre-Image Resistance

Trong phần này, chúng ta sẽ tìm hiểu các thuộc tính Second Pre-Image Resistant của hàm băm SHA256.

- Đầu tiên, tạo bản tóm lược SHA256 của file *declare.txt*
- Bốn chữ số thập lục phân cuối cùng của bản tóm lược được hiển thị là “7bc8”

```
ubuntu@macs-hash:~$ shasum -a 256 declare.txt
f98b1c252622e4a4f6edae5bdd135deb554693f6b3e5b4c35a1f30497bd77bc8 declare.txt
ubuntu@macs-hash:~$
```

- Thực hiện lệnh sau để tìm dữ liệu ngẫu nhiên mà khi băm, 6 số cuối dạng hex của bản tóm lược sẽ trùng với 6 số cuối dạng hex của bản tóm lược file *declare.txt*

./collide1.sh declare.txt 1

```
ubuntu@macs-hash:~$ ./collide1.sh declare.txt 1
8
1 hash was performed to find a match.
```

- Bây giờ, hãy thực hiện lệnh sau để so khớp hai chữ số hex cuối cùng của bản tóm lược file *declare.txt*.

./collide1.sh declare.txt 2

```
ubuntu@macs-hash:~$ ./collide1.sh declare.txt 2
03
e5
7d
3a
e1
5b
c8
7 hashes were performed before a match was found.
```

- Thực hiện lệnh sau để so khớp 3 chữ số hex cuối cùng của bản tóm lược file *declare.txt*.

./collide1.sh declare.txt 3

```
ubuntu@macs-hash:~$ ./collide1.sh declare.txt 3
The progress will not be shown -- only the final results.
2200 hashes were performed before a match was found.
ubuntu@macs-hash:~$
```

- *collide2.py* để giải quyết vấn đề xung đột. Nó tạo dữ liệu ngẫu nhiên và băm nó, sau đó lưu mỗi digest trong một bảng. Nó tiếp tục làm điều này cho đến khi nó tìm thấy hai xâu dữ liệu ngẫu nhiên có digest khớp nhau ở byte cuối cùng (tức là hai chữ số hex cuối cùng).

```
ubuntu@macs-hash:~$ ./collide2.py declare.txt 1
message 0 hashes to 60
message 1 hashes to 22
message 2 hashes to 51
message 3 hashes to 91
message 4 hashes to de
message 5 hashes to 5b
message 6 hashes to 9d
message 7 hashes to f4
message 8 hashes to 28
message 9 hashes to f6
message 10 hashes to f9
message 11 hashes to fc
message 12 hashes to a3
message 13 hashes to f7
message 14 hashes to a4
message 15 hashes to d5
message 16 hashes to 83
message 17 hashes to 52
message 18 hashes to 6e
message 19 hashes to d5
found after 19 tries:
0x82 0x32 0x3e 0xe0 = 0x25 0x12 0xcf 0xa8
ubuntu@macs-hash:~$
```

```

ubuntu@macs-hash:~$ ./collide2.py declare.txt 2
message 0 hashes to 26
message 1 hashes to 50
message 2 hashes to bf
message 3 hashes to 09
message 4 hashes to 11
message 5 hashes to c3
message 6 hashes to 05
message 7 hashes to 55
message 8 hashes to 1c
message 9 hashes to 52
message 10 hashes to 7f
message 11 hashes to 2d
message 12 hashes to 4e
message 13 hashes to 9d
message 14 hashes to fc
message 15 hashes to 59
message 16 hashes to 45
message 17 hashes to fe
message 18 hashes to da
message 19 hashes to 81
message 20 hashes to 9a
message 21 hashes to 0e
message 22 hashes to fb
message 23 hashes to 2c
message 24 hashes to 8d
message 25 hashes to a3
message 26 hashes to c0
message 27 hashes to b8
message 28 hashes to 85
message 29 hashes to 88
message 30 hashes to 36
message 31 hashes to 2b
message 32 hashes to c0
found after 32 tries:
0xe7 0x45 0x55 0xa7 = 0xc5 0x77 0x8a 0xfd

```

```

ubuntu@macs-hash:~$ ./collide2.py declare.txt 3
message 0 hashes to fc
message 1 hashes to 5c
message 2 hashes to dc
message 3 hashes to fa
message 4 hashes to 16
message 5 hashes to c7
message 6 hashes to 2e
message 7 hashes to 1c
message 8 hashes to 0b
message 9 hashes to 05
message 10 hashes to f6
message 11 hashes to a1
message 12 hashes to 44
message 13 hashes to 32
message 14 hashes to 55
message 15 hashes to da
message 16 hashes to 2b
message 17 hashes to b0
message 18 hashes to 91
message 19 hashes to 2e
found after 19 tries:
0x5d 0x3b 0x3e 0x3d = 0x9b 0xdc 0xb9 0x2b
ubuntu@macs-hash:~$

```

- Để tạo một HMAC dựa trên SHA1 bằng OpenSSL, bạn có thể nhập lệnh sau:

openssl dgst -sha1 -hmac KEY FILENAME

Với

- KEY bằng bất kỳ chuỗi nào bạn chọn, miễn là nó không có khoảng trắng.
- FILENAME bằng tên của một tệp trên hệ thống của bạn.

```
ubuntu@macs-hash:~$ openssl dgst -sha1 -hmac "2" declare.txt
HMAC-SHA1(declare.txt)= 4059519c75af436d508bf7e385c5f916af38ffb6
ubuntu@macs-hash:~$ openssl dgst -sha1 -hmac "mykey" declare.txt
HMAC-SHA1(declare.txt)= 851faeda4451a7eb991a008156f0de99da6e6b5b
ubuntu@macs-hash:~$ openssl dgst -sha1 -hmac "mykey" foo.txt
HMAC-SHA1(foo.txt)= 6d5abc6f42608cc49b34f837598532407cbbff6b
```

```
ubuntu@macs-hash:~$ openssl dgst -sha1 -hmac 1 declare.txt
HMAC-SHA1(declare.txt)= 76ca653c023a7eafecbb89e5997b623982d6d95c
ubuntu@macs-hash:~$ openssl dgst -sha1 -hmac 2 declare.txt
HMAC-SHA1(declare.txt)= 4059519c75af436d508bf7e385c5f916af38ffb6
ubuntu@macs-hash:~$ openssl dgst -sha1 -hmac 3 declare.txt
HMAC-SHA1(declare.txt)= f08793849fbad551a200c8c3fcf1c86abc0b439b
ubuntu@macs-hash:~$ openssl dgst -sha1 -hmac 4 declare.txt
HMAC-SHA1(declare.txt)= cd849f0f19eec429bc711c894245cc69b9c20624
ubuntu@macs-hash:~$ openssl dgst -sha1 -hmac 5 declare.txt
HMAC-SHA1(declare.txt)= df0528072cb5edc14e2eba29047df026385e99aa
ubuntu@macs-hash:~$ openssl dgst -sha1 -hmac 6 declare.txt
HMAC-SHA1(declare.txt)= 986eb8a92e561f550a911352c8b2cf5fd0465342
ubuntu@macs-hash:~$
```

- Checkwork sau khi hoàn thành nhiệm vụ 4

```
student@ubuntu:~/labtainer/labtainer-student$ checkwork macs-hash
Results stored in directory: /home/student/labtainer_xfer/macs-hash
Labname macs-hash
```

Student	collide1_count	collide2_count	shasum_count	did_7_shasum	hashed_floppy	hashed_iou	openssl_key
B21DCAT205	3	3	18	Y	Y	Y	Y

Kết luận

- Các nhiệm vụ đã giúp chúng ta hiểu về quá trình tạo bản tóm lược, kiểm tra tính toàn vẹn của file tải xuống, hiệu ứng "Avalanche", và thuộc tính "Second Pre-Image Resistance" của hàm băm SHA256.
- Qua đó, chúng ta có cái nhìn sâu sắc hơn về tính an toàn và tính toàn vẹn của dữ liệu khi sử dụng các thuật toán băm trong bảo mật thông tin.

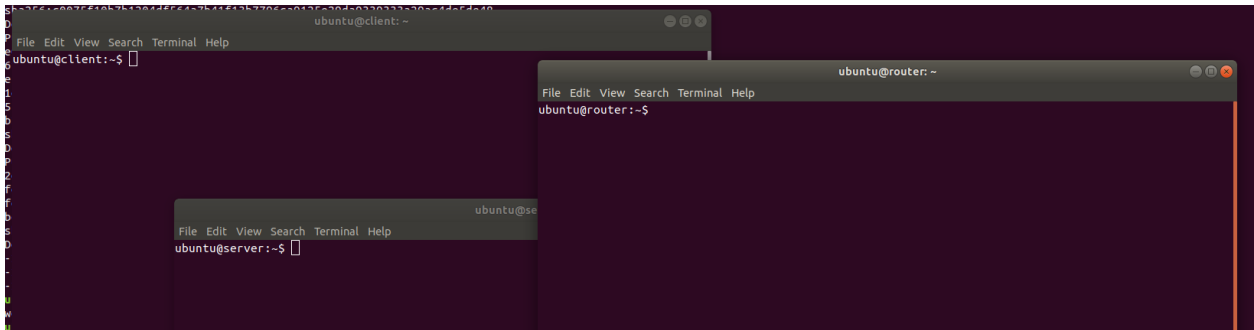
III. Bài thực hành: VPN host-to-host

Khởi động bài lab:

- Trên terminal, gõ lệnh:

```
labtainer -r vpnlab
```

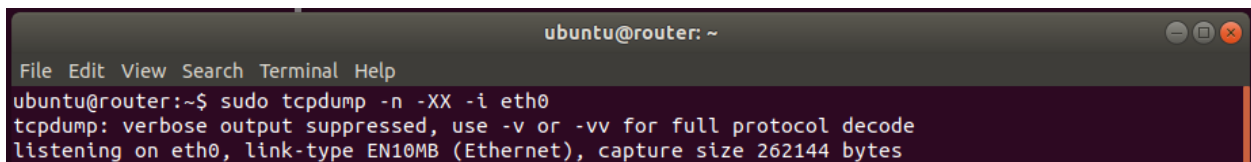
- Sau khi khởi động xong 3 terminal ảo sẽ xuất hiện: trong đó 1 máy ảo đóng vai trò client, 1 máy đóng vai trò server và 1 máy đóng vai trò router.



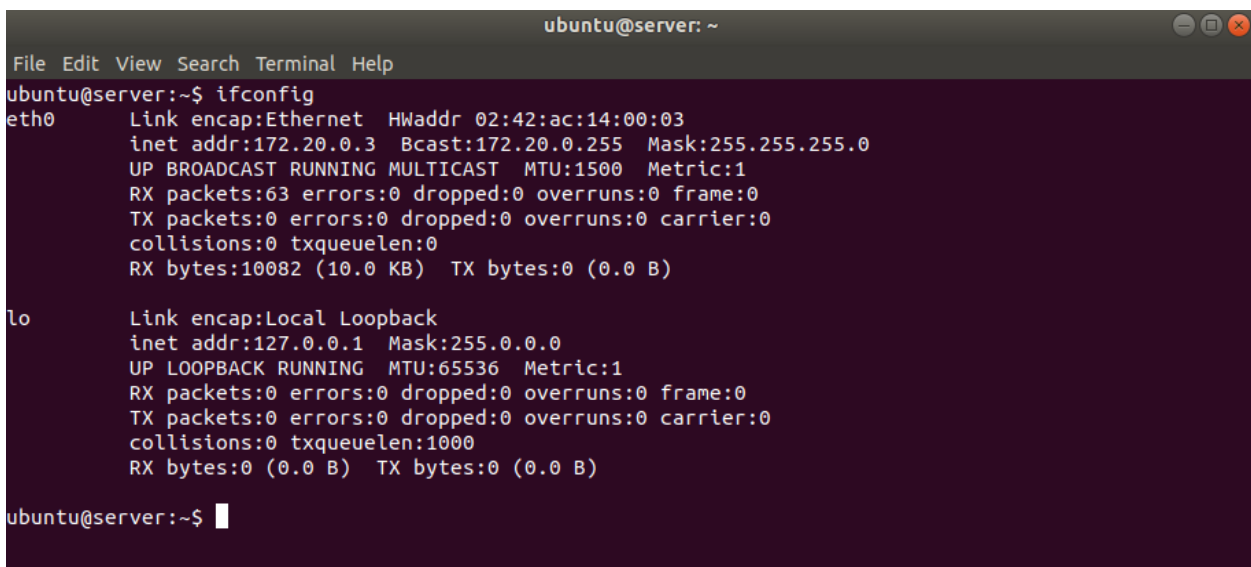
Các nhiệm vụ cần phải thực hiện:

- Khởi động tcpdump trên “router” để bắt các gói tin từ interface eth0.

sudo tcpdump -n -XX -i eth0



- Trong server sử dụng *ifconfig* để xem địa chỉ ip, ta thấy địa chỉ ip của server là 172.20.0.3



- Sử dụng wget trên máy khách để tìm nạp tệp index.html.
wget <http://172.20.0.3/index.html>

```

ubuntu@client: ~
File Edit View Search Terminal Help
ubuntu@client:~$ wget http://172.20.0.3/index.html
--2023-11-23 18:01:56-- http://172.20.0.3/index.html
Connecting to 172.20.0.3:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 223 [text/html]
Saving to: 'index.html'

index.html          100%[=====] 223 --.-KB/s  in 0.003s

2023-11-23 18:01:56 (74.3 KB/s) - 'index.html' saved [223/223]

ubuntu@client:~$

```

- Quan sát lưu lượng mạng từ tcpdump.

```

18:01:56.464381 IP 172.25.0.2.53812 > 172.20.0.3.80: Flags [P.], seq 1:148, ack 1, win 229, options [nop,nop,TS val 182964314 ecr 2723049842], length 147: HTTP: GET /index.html HTTP/1.1
0x0000: 0242 ac19 0001 0242 ac19 0002 0800 4500 .B....B.....E.
0x0010: 00c7 1a72 4000 4006 c78c ac19 0002 ac14 ...r@. ....
0x0020: 0003 d234 0050 c672 0354 fa7d 5713 8018 ...4.P.r.T.}W...
0x0030: 00e5 58ec 0000 0101 080a 0ae7 d05a a24e ..X.....Z.N
0x0040: 7172 4745 5420 2f69 6e64 6578 2e68 746d qrGET./index.htm
0x0050: 6c20 4854 5450 2f31 2e31 0d0a 5573 6572 l.HTTP/1.1..User
0x0060: 2d41 6765 6e74 3a20 5767 6574 2f31 2e31 -Agent:.Wget/1.1
0x0070: 372e 3120 286c 696e 7578 2d67 6e75 290d 7.1.(linux-gnu).
0x0080: 0a41 6363 6570 743a 202a 2f2a 0d0a 4163 .Accept:./*..Ac
0x0090: 6365 7074 2d45 6e63 6f64 696e 673a 2069 cept-Encoding:.i
0x00a0: 6465 6e74 6974 790d 0a48 6f73 743a 2031 dentity..Host:.1
0x00b0: 3732 2e32 302e 302e 330d 0a43 6f6e 6e65 72.20.0.3..Conne
0x00c0: 6374 696f 6e3a 204b 6565 702d 416c 6976 ction:.Keep-Aliv
0x00d0: 650d 0a0d 0a e....

```

- Xem html văn bản thuần túy trong luồng dữ liệu.

```

18:01:56.549308 IP 172.20.0.3.80 > 172.25.0.2.53812: Flags [P.], seq 187:410, ack 148, win 235, options [nop,nop,TS val 2723049928 ecr 182964395], length 223: HTTP
0x0000: 0242 ac19 0002 0242 ac19 0001 0800 4500 .B....B.....E.
0x0010: 0113 4e3e 4000 3f06 9474 ac14 0003 ac19 ..N>@.?.t.....
0x0020: 0002 0050 d234 fa7d 57cd c672 03e7 8018 ...P.4.}W..r....
0x0030: 00eb 5938 0000 0101 080a a24e 71c8 0ae7 ..Y8.....Nq...
0x0040: d0ab 3c68 746d 6c3e 0a3c 7469 746c 653e ..<html>.<title>
0x0050: 5361 6d70 6c65 2069 6e64 6578 2e68 746d Sample.index.htm
0x0060: 6c20 666f 7220 4d79 4854 5450 5365 7276 l.for.MyHTTPServ
0x0070: 6572 3c2f 7469 746c 653e 0a3c 626f 6479 er</title>.<body
0x0080: 3e0a 3c68 313e 5361 6d70 6c65 2069 6e64 >.<h1>Sample.ind
0x0090: 6578 2e68 746d 6c20 666f 7220 4d79 4854 ex.html.for.MyHT
0x00a0: 5450 5365 7276 6572 3c2f 6831 3e0a 3c62 TPServer</h1>.<b
0x00b0: 723e 0a3c 6832 3e4c 696e 6b73 3a3c 2f68 r>.<h2>Links:</h
0x00c0: 323e 0a3c 6272 3e0a 3c61 2068 7265 663d 2>.<br>.<a.href=
0x00d0: 6c69 6e6b 312e 6874 6d6c 3e6c 696e 6b31 link1.html>link1
0x00e0: 2e68 746d 6c3c 2f61 3e0a 3c62 723e 0a3c .html</a>.<br>.<
0x00f0: 6120 6872 6566 3d6c 696e 6b32 2e68 746d a.href=link2.htm
0x0100: 6c3e 6c69 6e6b 322e 6874 6d6c 3c2f 613e l>link2.html</a>
0x0110: 0a3c 2f62 6f64 793e 0a3c 2f68 746d 6c3e .</body>.</html>
0x0120: 0a

```

- Khởi động chương trình openvpn trên máy chủ:

sudo openvpn --config server.conf --daemon

```
ubuntu@server:~$ sudo openvpn --config server.conf --daemon
ubuntu@server:~$
```

- Sau đó khởi động chương trình openvpn trên máy khách:

sudo openvpn --config client.conf --daemon

```
ubuntu@client:~$ sudo openvpn --config client.conf --daemon
ubuntu@client:~$
```

- Sử dụng lại ifconfig từ máy server để kiểm tra rằng có một “đường hầm” giúp đảm bảo an toàn khi truyền dữ liệu là tun0 với địa chỉ ip là 10.8.0.1

```
ubuntu@server:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 02:42:ac:14:00:03
          inet addr:172.20.0.3  Bcast:172.20.0.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:85 errors:0 dropped:0 overruns:0 frame:0
          TX packets:17 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:12176 (12.1 KB)  TX bytes:1635 (1.6 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

tun0      Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          inet addr:10.8.0.1  P-t-P:10.8.0.2  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

ubuntu@server:~$
```

- Sử dụng lại wget, nhưng lần này sử dụng địa chỉ đường hầm của máy chủ(địa chỉ hiển thị ở interface ”tun0” khi chạy lệnh ifconfig trên “server”).

wget <http://10.8.0.1/index.html>

```

ubuntu@client:~$ wget http://10.8.0.1/index.html
--2023-11-23 18:07:10-- http://10.8.0.1/index.html
Connecting to 10.8.0.1:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 223 [text/html]
Saving to: 'index.html.1'

index.html.1          100%[=====>]          223  --.-KB/s   in 0.001s

2023-11-23 18:07:10 (204 KB/s) - 'index.html.1' saved [223/223]

ubuntu@client:~$

```

- Quan sát lưu lượng mạng qua tcpdump, ta thấy tất cả thông tin đều được mã hóa và không thể đọc được.

```

18:07:10.538142 IP 172.20.0.3.1194 > 172.25.0.2.1194: UDP, length 484
    0x0000: 0242 ac19 0002 0242 ac19 0001 0800 4500 .B....B.....E.
    0x0010: 0200 b0e0 4000 3f11 30da ac14 0003 ac19 ....@.?..0.....
    0x0020: 0002 04aa 04aa 01ec 5a30 b016 6a03 8f06 .....Z0..j...
    0x0030: 1f7c 2c9a 188d 8b34 c0fd 8961 ad6d 35ea .|,....4...a.m5.
    0x0040: 4324 fa2d 86cc 4814 2136 44f4 3c04 e776 C$.~..H.!6D.<..v
    0x0050: 8c0a ed94 a54a 83d7 099c 2fe5 659c a2c3 .....J..../.e...
    0x0060: 0212 a976 8248 0987 bf33 ca08 f765 8e07 ...v.H...3...e..
    0x0070: 7343 c8d2 b5f7 a484 d3de 6645 c4a0 65d1 sC.....fE..e.
    0x0080: 9750 87c8 ad9f e2e5 65ce 8701 d433 0943 .P.....e....3.C
    0x0090: 70c4 9b54 0030 f4e3 8d1c cc0b 3235 e18e p..T.0.....25..
    0x00a0: 54a6 2b68 d8c5 b7ab 9491 2505 ab69 8d23 T.+h.....%.i.#
    0x00b0: b684 e426 2aff 72d0 b64e b2d4 27cf 14cd ...&*.r..N..'...
    0x00c0: cb09 1a90 c514 e0dd 04d8 5299 dcd5 6bbb .....R...k.
    0x00d0: c8fa 869a 56bd 709c 2151 ad1c f2b0 fe05 ....V.p.!Q.....
    0x00e0: 1934 05c5 749b 22b5 fcdd 82de 60d1 5297 .4..t.".....`R.
    0x00f0: b8fb c9f9 1b19 1a7c 0151 156d 0bbc 0549 .....|.Q.m...I
    0x0100: 961d 36b3 21d1 f9c9 0a26 a8c0 a241 448f ..6.!....&...AD.
    0x0110: a10e da35 e18e 2428 3af6 c2bd f5bf ea7b ...5..$(.....{
    0x0120: 1910 b3d6 29e0 5a7a 3029 e45a 860f 648d .....).Zz0).Z..d.
    0x0130: b655 b445 c7a4 3ed2 ae2f b919 57c4 8c07 .U.E...>../..W...
    0x0140: 8425 9840 efa6 4003 ef3e f6ed 84b2 2a31 .%.@...@...>...*1
    0x0150: ecd6 c7af a379 3808 7b33 4ef0 e9a6 3ec7 .....y8.{3N...>.
    0x0160: 7716 d6cf eeda 2afe 32a9 8f97 a084 7e4d w.....*.2.....~M
    0x0170: d3f2 019e 6316 abd8 cdf9 c26d f4c4 2e4d ....c.....m...M
    0x0180: 1acf b612 8854 0c27 328a 0bc8 5f84 a066 .....T.'2..._..f
    0x0190: a1b1 8fac e846 5c0c cd7c d4f9 301f 69db .....F\...|.0.i.
    0x01a0: c949 05b2 6843 5ac0 0c8c db1b 5c1a 2802 .I..hCZ.....\.(.
    0x01b0: dab1 2b53 f08f 9bc3 a3d6 81b5 f75e 9d18 ..+S.....^...
    0x01c0: 7745 f8de 6fe4 2b06 990b 4c5c d168 d780 wE..o.+...L\..h..
    0x01d0: d7d0 6a7c 2d63 7c89 8e82 cadd 8aaf a7fa ..j|-c|.....
    0x01e0: 9772 024f b2ee 9287 ef65 180f fa35 8f90 .r.O.....e...5..
    0x01f0: 6477 72bf 58ac a7fc f869 07ca d3ba d3f6 dwr.X....i.....
    0x0200: 222e 6bd3 9b04 1695 b37a 452e ef94 ".k.....zE...

```

- Checkwork kiểm tra bài làm

```
student@ubuntu:~/labtainer/labtainer-student$ checkwork vpnlab
Results stored in directory: /home/student/labtainer_xfer/vpnlab
Labname vpnlab

Student          |
=====         |
B21DCAT205      |
What is automatically assessed for this lab:

: was able to access http over the vpn
get_index_ok = matchany : string_equal : getindexhtml : answer=True
get_link1_ok = matchany : string_equal : getlink1html : answer=True
get_link2_ok = matchany : string_equal : getlink2html : answer=True
```

Kết luận

- Qua các nhiệm vụ, chúng ta đã thực hành việc bắt gói tin mạng, quan sát nội dung của lưu lượng mạng, triển khai và sử dụng OpenVPN để tạo một kênh kết nối bảo mật.
- Việc này giúp chúng ta hiểu rõ hơn về cách thức hoạt động của các công cụ mạng cũng như việc thiết lập và sử dụng VPN để bảo vệ dữ liệu truyền qua mạng.