Trường Đại học Thủy Lợi
Bộ môn Công nghệ thông tin

Đề tài: Tìm hiểu và triển khai cấu hình bảo mật SSH server trên Linux

Môn: Linux và mã nguồn mở
Giảng viên hướng dẫn : Kiều Tuấn Dũng
Nhóm : L25

1, Danh sách thành viên và công việc.

Họ và tên	**Mã SV**	**Công việc**	**Tiến độ**
Nguyễn Duyên Mạnh	175A071275	-Tìm hiểu về cấu hình bảo mật SSH server trên Linux. -Hướng dẫn cài đặt và bảo mật.	Hoàn thành
Nguyễn Thùy Linh	175A071621	-Cài đặt demo.	Hoàn thành

2, Nội dung nghiên cứu(tài liệu được thao khảo qua trang digitalocean.com, server-world.info):

a. Khái niệm về giao thức SSH:

Secure Shell (SSH) là một giao thức mạng mật mã cung cấp chức năng mã hóa giữa và máy khách và máy chủ. Nó thay thế các máy khách mạng không bảo mật trước đây trong môi trường mạng.

SSH dùng mật khẩu để xác thực người dùng trong một phiên kết nối giữa client và server.
SSH server là máy chủ của chúng ta, với máy chủ chạy hệ điều hành Linux ta cần cìa openssh-server
SSH Client : là máy muốn truy cập vào SSH server của chúng ta, trên Linux có openssh-client

b. Chức năng cụ thể của giao thức SSH:

Dịch vụ được tạo ra nhằm thay thế cho trình Telnet vốn không có mã hóa và sử dụng kỹ thuật cryptographic để đảm bảo tất cả giao tiếp gửi tới và gửi từ server từ xa diễn ra trong tình trạng mã hóa. Nó cung cấp thuật toán để chứng thực người dùng từ xa, chuyển input từ client tới host, và relay kết quả trả về tới khách hàng.

c. Cách SSH hoạt động:

SSH làm việc thông qua 3 bước đơn giản:

- Định danh host: Xác định danh tính của hệ thống tham gia phiên làm việc SSH
- Mã hóa: Thiết lập kênh làm việc mã hóa
- Chứng thực: Xác thực người sử dụng có quyền đăng nhập hệ thống

Cách SSH hoạt động:

Để thiết lập kết nối SSH, bạn cần hai thành phần: máy khách và thành phần phía máy chủ tương ứng. Máy khách SSH là một ứng dụng bạn cài đặt trên máy tính mà bạn sẽ sử dụng để kết nối với máy tính khác hoặc máy chủ. Máy khách sử dụng thông tin máy chủ từ xa được cung cấp để bắt đầu kết nối và nếu thông tin đăng nhập được xác minh, sẽ thiết lập kết nối được mã hóa.

Về phía máy chủ, có một thành phần được gọi là SSH daemon liên tục lắng nghe một cổng TCP / IP cụ thể cho các yêu cầu kết nối máy khách có thể. Khi khách hàng khởi tạo kết nối, trình nền SSH sẽ phản hồi với phần mềm và các phiên bản giao thức mà nó hỗ trợ và cả hai sẽ trao đổi dữ liệu nhận dạng của họ. Nếu thông tin đăng nhập

được cung cấp là chính xác, SSH sẽ tạo một phiên mới cho môi trường phù hợp.

Lệnh SSH có 3 phần:
ssh {user}@{host}

SSH key command cho hệ thống biết là bạn muốn mở một kết nối được mã hóa Secure Shell Connection. **{user}** đại diện cho tài khoản người dùng bạn muốn dùng để truy cập. Ví dụ, bạn muốn truy cập user **root**, thì thay root tại đây. User root là user quản trị hệ thống với toàn quyền để chỉnh sửa bất kỳ điều gì trên hệ thống. **{host}** đại diện cho máy tính bạn muốn dùng để truy cập. Nó có thể là một địa chỉ IP **(ví dụ 244.235.23.19)** hoặc một tên miền (ví dụ, www.xyzdomain.com)

Khi bạn nhấn enter, nó sẽ hỏi bạn nhập mật khẩu tương ứng cho tài khoản. Khi bạn gõ, bạn sẽ không thấy bất kỳ dấu hiệu nào trên màn hình, nhưng nếu bạn gõ đúng mật khẩu và nhấn enter, bạn sẽ vào được hệ thống và nhận thông báo đăng nhập thành công.

d. Hướng dẫn cài đặt SSH server trên Ubuntu Desktop

- Để cài đặt openssh-server, ta chạy dòng lệnh:

```
Sudo apt-get install openssh-server
```

```
linh@linh:~$ sudo apt-get install openssh_server
[sudo] password for linh:
Reading package lists... Done
Building dependency tree
Reading state information... Done
E: Unable to locate package openssh_server
```

- Sau khi tải về xong chúng ta dùng lệnh dưới để chạy SSH server:

```
Sudo service ssh start
```

```
linh@linh:~$ sudo service ssh start
linh@linh:~$ _
```

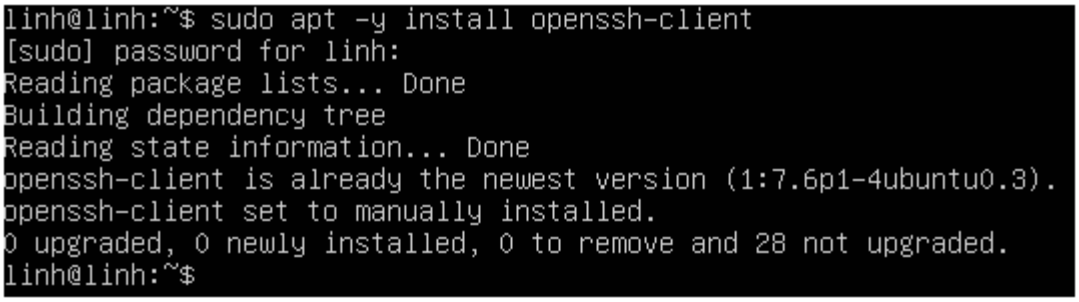
- Để kiểm tra dịch vụ SSH đã chạy hay chưa, ta dùng lệnh:

Systemctl status ssh

Hiển thị active (running) có nghĩa là nó đang chạy

- Cài đặt cấu hình SSH client cho Ubuntu:

Sudo apt –y install openssh-client

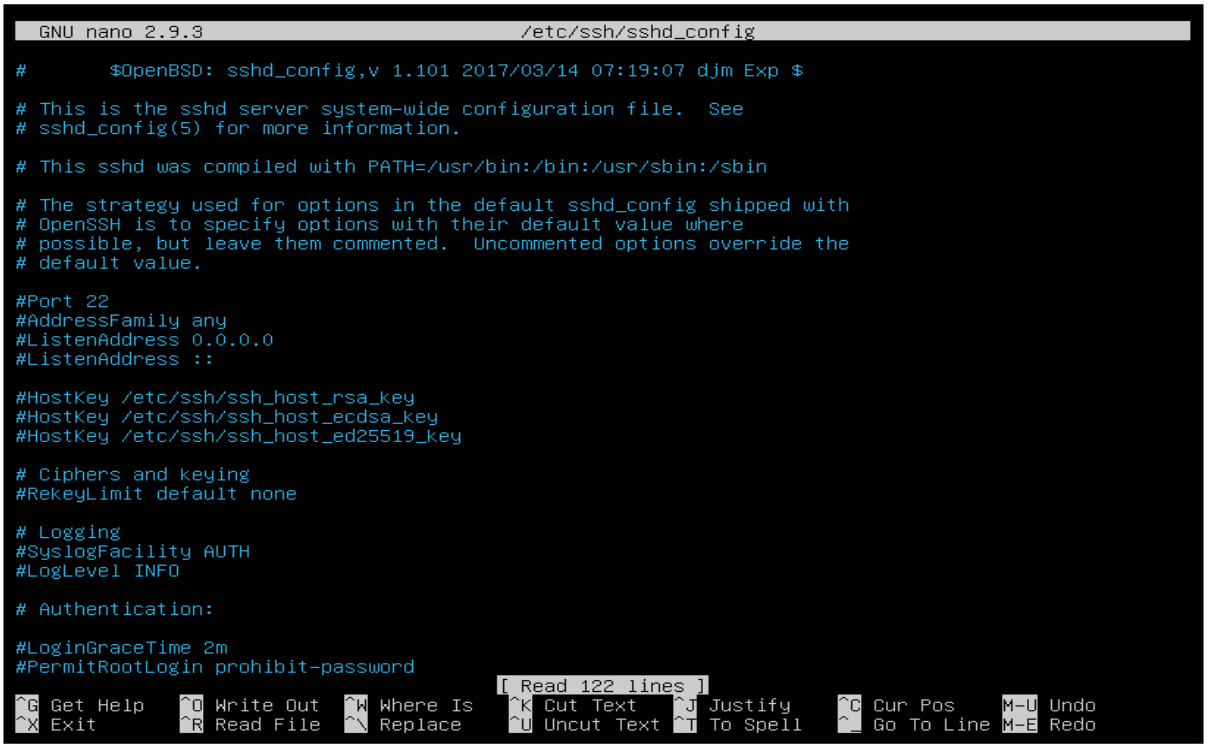


Việc cài đặt hoàn tất.

e. Hướng dẫn sử dụng / quản trị:

e.1. Thay đổi cấu hình ssh:

Sudo nano /etc/ssh/sshd_config



e.2. Kết nối SSH với window và với Ubuntu Server qua putty.exe.

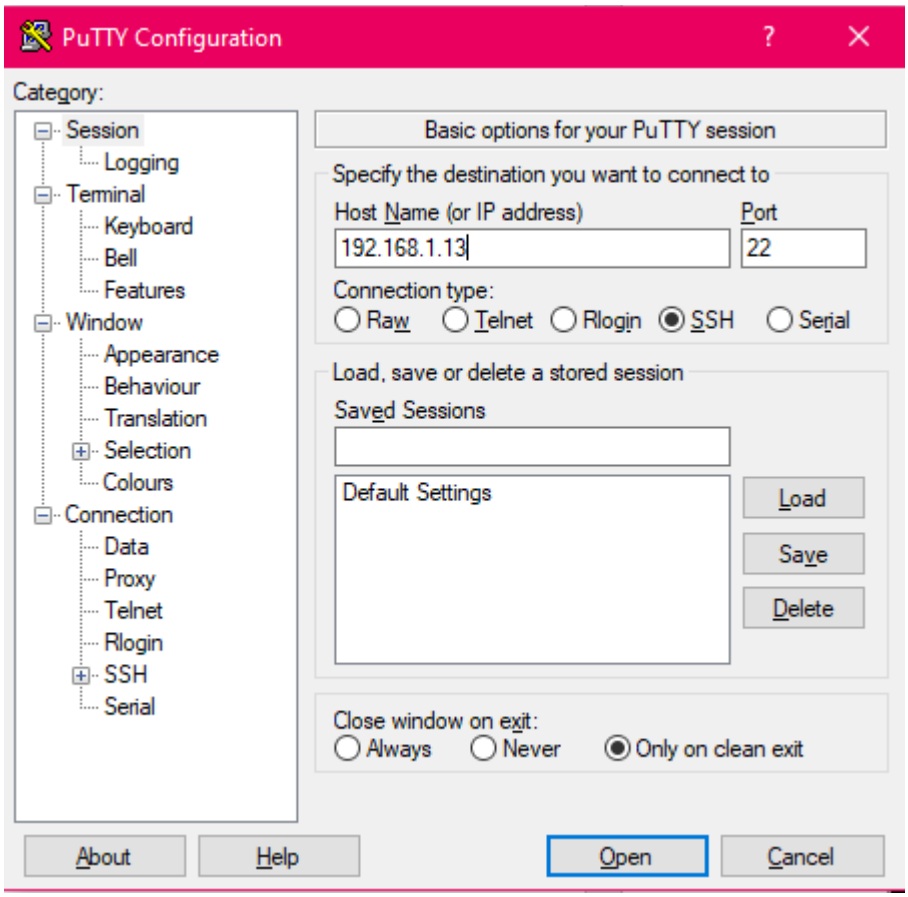
- Trước hết ta kiểm tra IP của máy:

Ifconfig

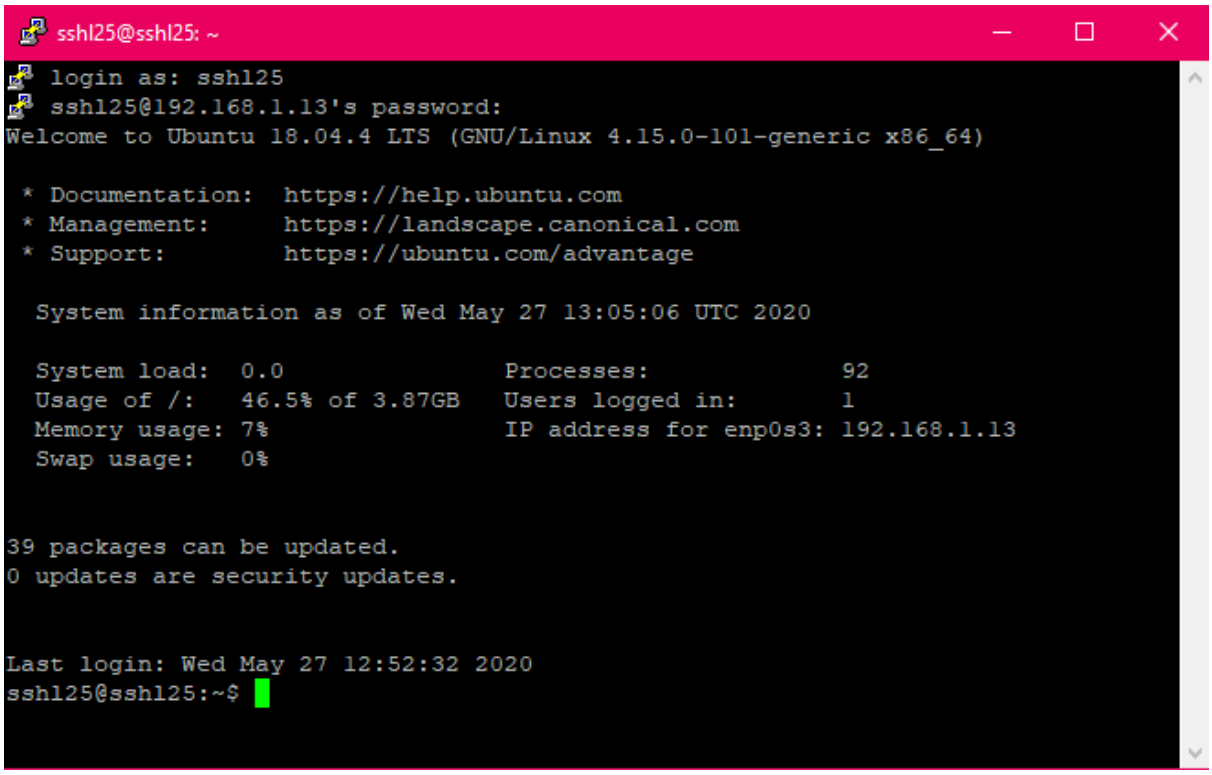
```
ssh125@ssh125:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.13  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::a00:27ff:fe37:5bb2  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:37:5b:b2  txqueuelen 1000  (Ethernet)
        RX packets 151  bytes 21706 (21.7 KB)
        RX errors 0  dropped 1  overruns 0  frame 0
        TX packets 67  bytes 7488 (7.4 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 92  bytes 7036 (7.0 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 92  bytes 7036 (7.0 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

ssh125@ssh125:~$
```

- Kết nối với server thao tác như dưới:

Nhập địa chỉ IP vào phần host name:



Sau đó bạn đăng nhập:



e.3. Bảo mật tệp cấu hình SSH

- Để tăng tính năng bảo mật chúng ta truy cập vào cấu hình ssh để chỉnh sửa:

```
Sudo nano /etc/ssh/sshd_config
```

- Sử dụng SSH2:

Giao thức SSH 1 (SSH1) chứa nhiều lỗ hổng bảo mật. Thay vào đó, sử dụng giao thức 2 (SSH2) được khuyến khích.Theo mặc định, SSH2 nên được đặt. Nếu không thì thay đổi dòng Giao thức để sử dụng SSH2.

```
#protocol 2_
```

Một cuộc tấn công phổ biến là cố gắng sử dụng root để đăng nhập vào máy chủ bằng SSH. Vì đây là một rủi ro bảo mật lớn, hãy vô hiệu hóa đăng nhập SSH gốc bằng cách thay đổi PermitRootLogin từ không có mật khẩu thành:

```
#PermitRootLogin no_
```

- Ẩn lần đăng nhập cuối cùng:

Bạn có thể ẩn người dùng đăng nhập cuối cùng bằng cách chỉnh sửa dòng sau:

```
#PrintLastLog no
```

- Hạn chế đăng nhập SSH vào địa chỉ IP cụ thể:

Theo mặc định, SSH sẽ chấp nhận các kết nối từ bất kỳ địa chỉ IP bên ngoài nào. Nếu bạn muốn hạn chế SSH chỉ cho phép kết nối từ một địa chỉ IP cụ thể, bạn có thể thêm một dòng ListenAddress.

Ví dụ: nếu bạn muốn chỉ chấp nhận kết nối SSH từ địa chỉ IP 192.168.1.2, bạn sẽ thêm dòng:

- Vô hiệu hóa xác thực mật khẩu:

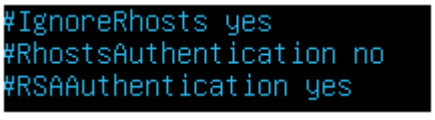
Xác thực mật khẩu trong SSH là một rủi ro bảo mật lớn nếu người dùng của bạn đặt mật khẩu yếu. Xem phần này để biết hướng dẫn về cách thiết lập xác thực khóa SSH.

Để tắt xác thực mật khẩu, hãy thay đổi dòng PasswordAuthentication:

- Vô hiệu hóa Rhosts:

Theo mặc định, SSH không cho phép rhosts. Các tệp .rhosts chỉ định người dùng nào có thể truy cập các lệnh r (chẳng hạn như RCp và rsh) trên hệ thống cục bộ mà không cần mật khẩu.

Để vô hiệu hóa rhosts:



- Vô hiệu hóa xác thực dựa trên máy chủ:

Xác thực dựa trên máy chủ của SSH an toàn hơn xác thực rhosts. Tuy nhiên, các máy chủ đáng tin cậy vẫn được coi là một rủi ro bảo mật.

Theo mặc định, tùy chọn HostbasingAuthentication bị tắt, nếu không thì thay đổi dòng sau:

```
# HostbasedAuthentication no
```

- Đặt thời gian chờ LoginGraceTime:

"LoginGraceTime" chỉ định khoảng thời gian sau khi yêu cầu kết nối, máy chủ SSH sẽ đợi trước khi ngắt kết nối. Giá trị được đề xuất cho thời gian chờ đăng nhập là 60 giây.

Bạn có thể thay đổi giá trị này bằng cách chỉnh sửa dòng sau:

```
#LoginGraceTime 60
```

- Đặt kết nối khởi động tối đa:

Giới hạn số lượng kết nối đồng thời tối đa vào daemon SSH có thể giúp bảo vệ máy chủ SSH của bạn khỏi một cuộc tấn công. Bạn có thể đặt giá trị này bằng cách chỉnh sửa dòng sau thành số lượng kết nối đồng thời bạn muốn cho phép. Trong ví dụ này, chúng tôi đã chọn 2:

```
#MaxStartups 2
```

- Vô hiệu hóa chuyển tiếp:

Tin tặc có thể sử dụng kỹ thuật chuyển tiếp cổng tới các kết nối mạng đường hầm thông qua phiên SSH để đăng nhập vào hệ thống.

Để vô hiệu hóa thay đổi này, các dòng sau:

```
#AllowTcpForwarding no
#GatewayPorts no
#X11Forwarding no
```

- Đăng nhập thêm thông tin:

Theo mặc định, SSH ghi lại mọi thứ. Nếu bạn muốn đăng nhập thêm thông tin như các lần đăng nhập thất bại. bạn có thể thay đổi giá trị từ INFO sang VERBOSE

Thao tác thay đổi dòng dưới:

```
#LogLevel VERBOSE
```

- Vô hiệu hóa mật khẩu trống:

Bạn sẽ muốn từ chối đăng nhập cho người dùng bằng mật khẩu trống (trống).

Theo mặc định, tùy chọn này bị vô hiệu hóa, nếu không thì thay đổi dòng sau:

```
#PermitEmptyPasswords no
```

SSH cho phép người dùng đặt khoảng thời gian chờ không hoạt động. Sau khi khoảng thời gian này trôi qua, người dùng nhàn rỗi sẽ tự động đăng xuất.

Bạn có thể đặt số giây bằng cách thêm dòng sau:

```
#ClientAliveInterval 300
#ClientAliveCountMax 0
```

- Khi bạn đã hoàn tất chỉnh sửa tệp /etc/ssh/sshd_config, hãy lưu và thoát , sau đó khởi động lại máy chủ SSH:

```
ssh125@ssh125:~$ sudo service ssh start
ssh125@ssh125:~$ _
```

e.3. Bảo mật kết nối SSH bằng cách trao đổi khóa để tăng tính bảo mật:

- *Các thành phần của SSH Key:*

- Gồm 3 thành phần:

Private Key: có dạng file chứ chuỗi mã hóa được lưu trên Client, cần phải bảo mật file này cẩn thận để lấy lại Public Key
Public Key: cũng là dạng file chứ chuỗi mã hóa được lưu trên Server
Passphrase: là mật khẩu dùng để nhận diện Public Key và Private Key khi tạo SSH connection, cũng như để lấy lại Public Key

- *Cách làm việc của SSH Key:*

- Quá trình hình thành kết nối SSH sử dụng Key sẽ trải qua nhiều lớp xác thực khác nhau.
- Đầu tiên, khi SSH Client khởi tạo kết nối bạn phải nhập vào Passphrase để kiểm tra xem Private Key & Public Key có phải là một cặp hay không.
- Tiếp theo chúng sẽ được so sánh với nhau theo thuật toán riêng xem có khớp hay không. Nếu ok người dùng phải nhập đúng tài khoản được cấp trên SSH Server tương ứng với cặp khóa này. Khi đó kết nối mới được khởi tạo để bắt đầu phiên làm việc.

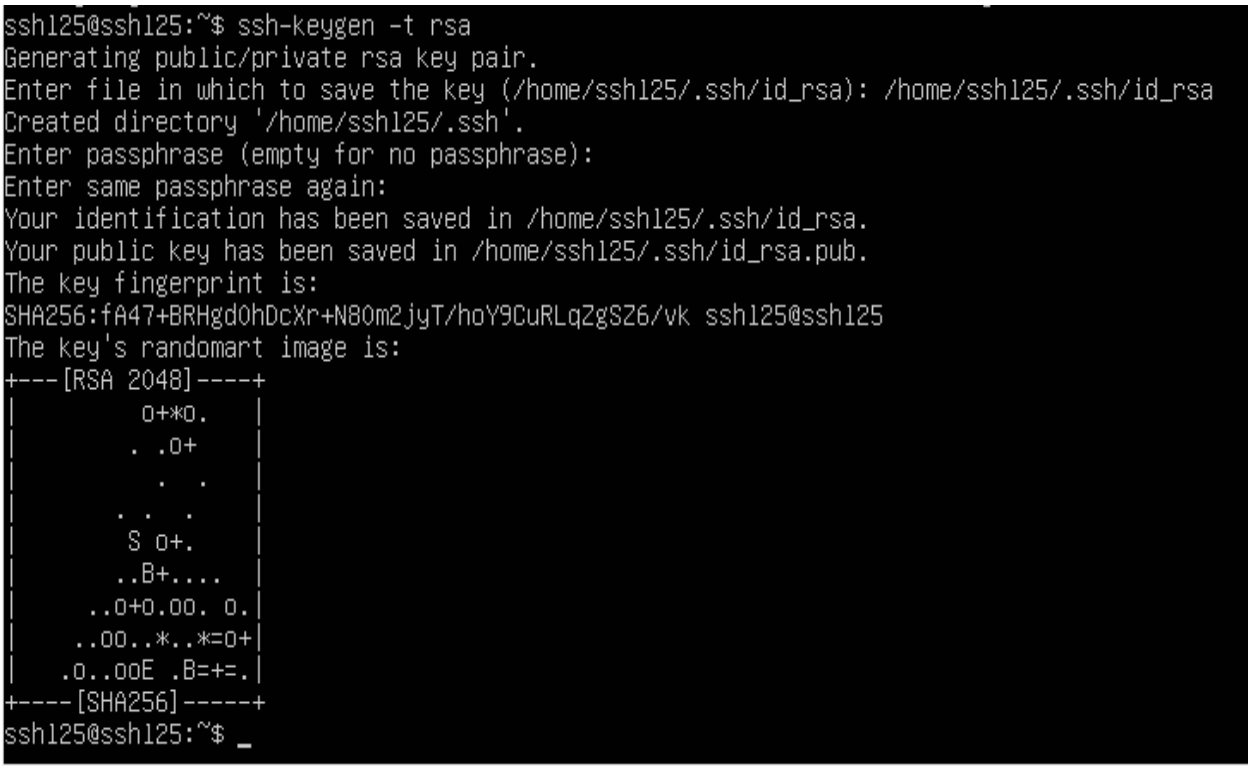
-*Tạo private key và public key:*

Linux: dùng ssh-keygen tạo khóa, công cụ này có sẵn nên chúng ta có thể sử dụng luôn

- Tạo cặp khóa cho mỗi người dùng, vì vậy hãy đăng nhập với một người dùng chung và làm việc như sau:

  Ssh-keygen –t rsa

  -t rsa : chỉ định thuật toán mã hóa là RSA



- Nó sẽ hỏi bạn taọ Passphrase, nhập vào pasphrase muốn dùng. Quá trình tạo SSH keys trên Linux hoàn tất ta sẽ được hai file:
  Id_rsa: đây là file chứa Private Key
  Id_rsa.pub: đây là chứa file Public Key

- Mặc định mỗi user trên Linux có một Home Directory (profile) có đường dẫn home/user . Khi đăng nhập bằng user nào để tạo sshKey thì cặp khóa sinh ra nằm trong thư mục /home/user/.ssh với dấu chấm đằng trước chỉ định .ssh là thư mục ẩn.

- Cách sử dụng Public Key:

Theo mô hình ở trên, ta di chuyển Public Key và0 đường dẫn là : home/sshl25/.ssh/authorized_keys trên server:

```
sshl25@sshl25:~$ mv ~/.ssh/id_rsa.pub ~/.ssh/authorized_keys
sshl25@sshl25:~$
```

Phân quyền Người sở hữu chỉ có quyền viết và thực thi:

```
sshl25@sshl25:~$ chmod 600 ~/.ssh/authorized_keys
sshl25@sshl25:~$ _
```

Ta cho quyền người sở hữu thư mục có quyền đọc nghe viết. Còn những người cùng nhóm hoặc những người còn lại không có quyền:

```
sshl25@sshl25:~$ chmod 700 ~/.ssh
sshl25@sshl25:~$
```

Ta chuyển tệp tin từ xa đến cục bộ:

```
sshl25@sshl25:~$ scp sshl25@192.168.1.13:/home/sshl25/.ssh/id_rsa ~/.ssh
Enter passphrase for key '/home/sshl25/.ssh/id_rsa':
id_rsa                                        100% 1766     1.0MB/s   00:00
sshl25@sshl25:~$
```

```
linh@linh:~$ ssh linh@192.168.138.133
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-99-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Thu Apr 30 17:11:59 UTC 2020

  System load:  0.0               Processes:           156
  Usage of /:   23.8% of 19.56GB  Users logged in:     1
  Memory usage: 11%               IP address for ens33: 192.168.138.133
  Swap usage:   0%

 * Ubuntu 20.04 LTS is out, raising the bar on performance, security,
   and optimisation for Intel, AMD, Nvidia, ARM64 and Z15 as well as
   AWS, Azure and Google Cloud.

     https://ubuntu.com/blog/ubuntu-20-04-lts-arrives


28 packages can be updated.
0 updates are security updates.


Last login: Thu Apr 30 17:02:52 2020
linh@linh:~$
```

- Cấu hình SSH Key trên Server:

Khởi động lại hệ thống:

```
ssh125@ssh125:~$ systemctl restart ssh
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ===
Authentication is required to restart 'ssh.service'.
Authenticating as: ssh125
Password:
==== AUTHENTICATION COMPLETE ===
ssh125@ssh125:~$
```

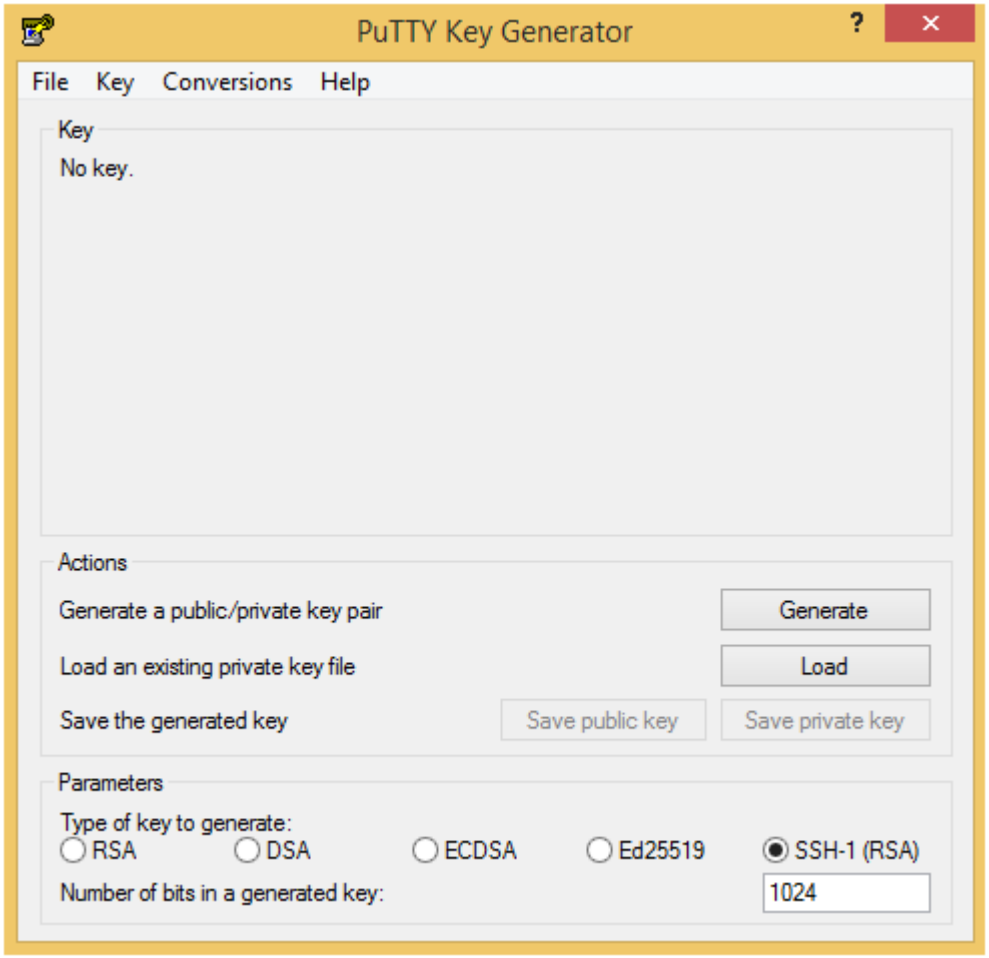
- *Sử dụng Putty để thử đăng nhập bằng SSH Key trên Linux:*

SSH sẽ tự biết lấy private key trong /home/linh/.ssh/id_rsa trên Client để so sánh tạo kết nối với server

```
ssh125@ssh125:~$ ssh ssh125@192.168.1.13
Enter passphrase for key '/home/ssh125/.ssh/id_rsa':
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-101-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Wed May 27 13:55:27 UTC 2020

  System load:  0.0               Processes:             93
  Usage of /:   46.5% of 3.87GB   Users logged in:       1
  Memory usage: 8%                IP address for enp0s3: 192.168.1.13
  Swap usage:   0%


39 packages can be updated.
0 updates are security updates.


ssh125@ssh125:~$ _
```

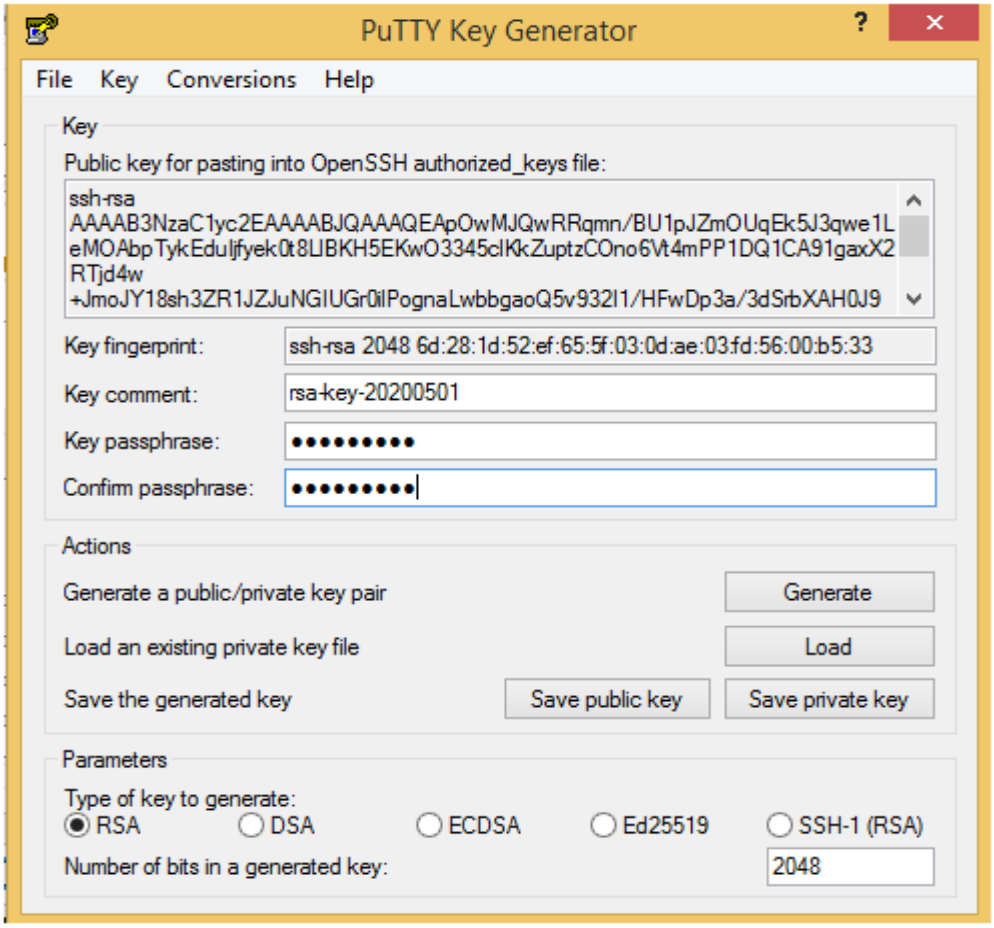
- Sử dụng PuTTy để thử đăng nhập bằng SSH Key trên Window (mở tìm kiếm gõ puttygen):

- Chọn generate để tạo key:

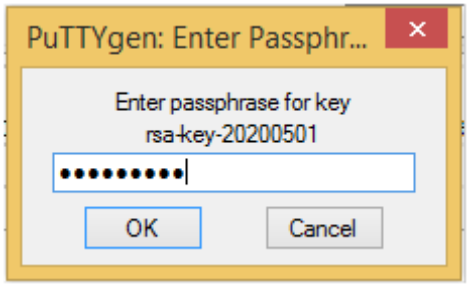


- Nhấp vào nút "Save Private Key" để lưu nó trong thư mục bạn thích với bất kỳ tên tệp nào bạn thích.

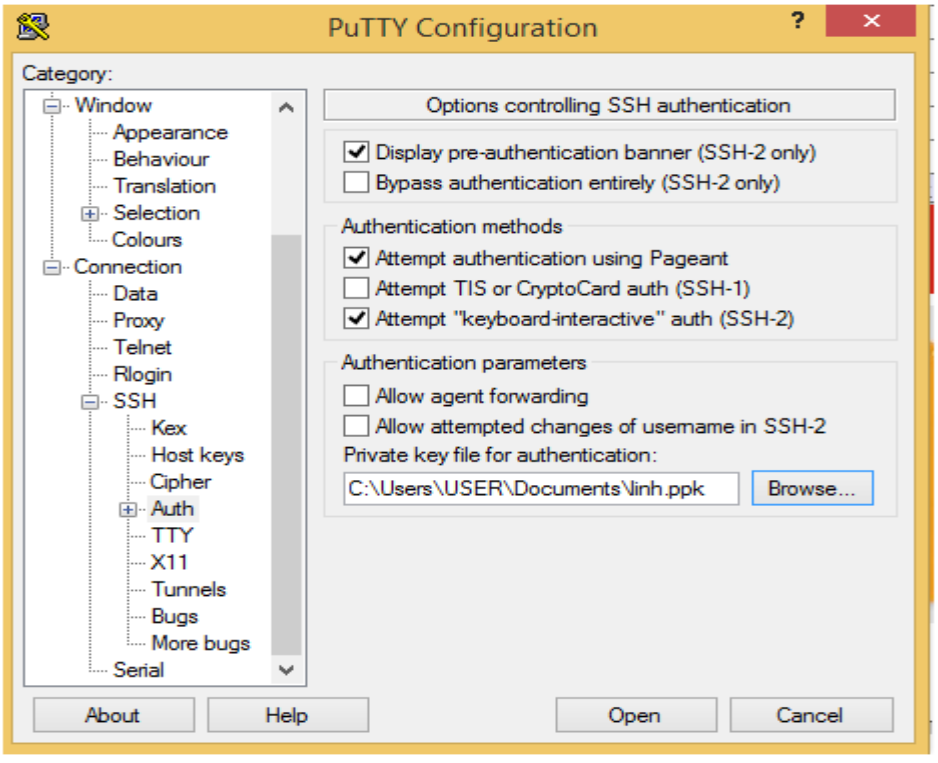
  Chọn save private key để lưu.



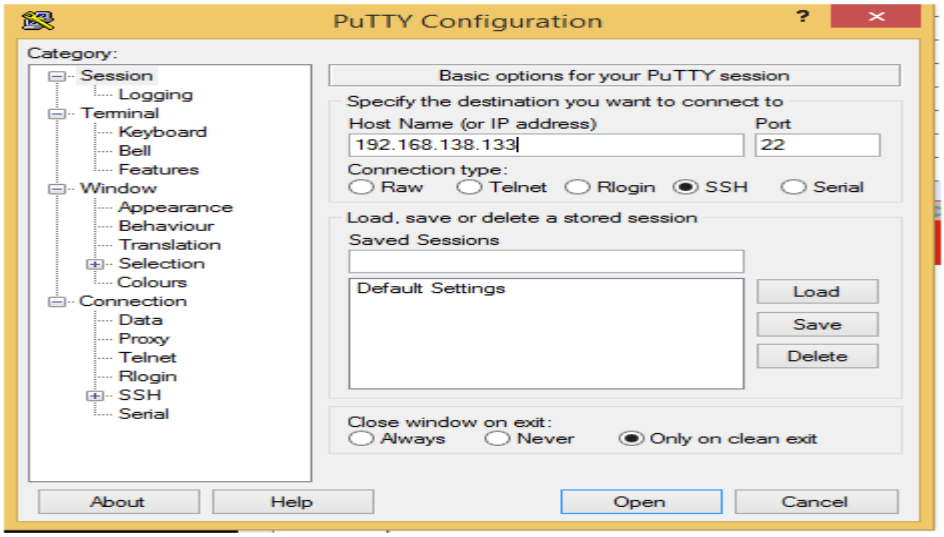
- Chỉ định khóa bí mật mà bạn đã tải xuống, sau đó cụm mật khẩu được yêu cầu như sau, hãy trả lời nó.



Bắt đầu Putty và mở [Kết nối] - [SSH] - [Auth] trên menu bên trái, sau đó chọn "private_key" vừa được lưu ở trên.



- Quay lại [Session] trên menu bên trái và kết nối với máy chủ SSH.

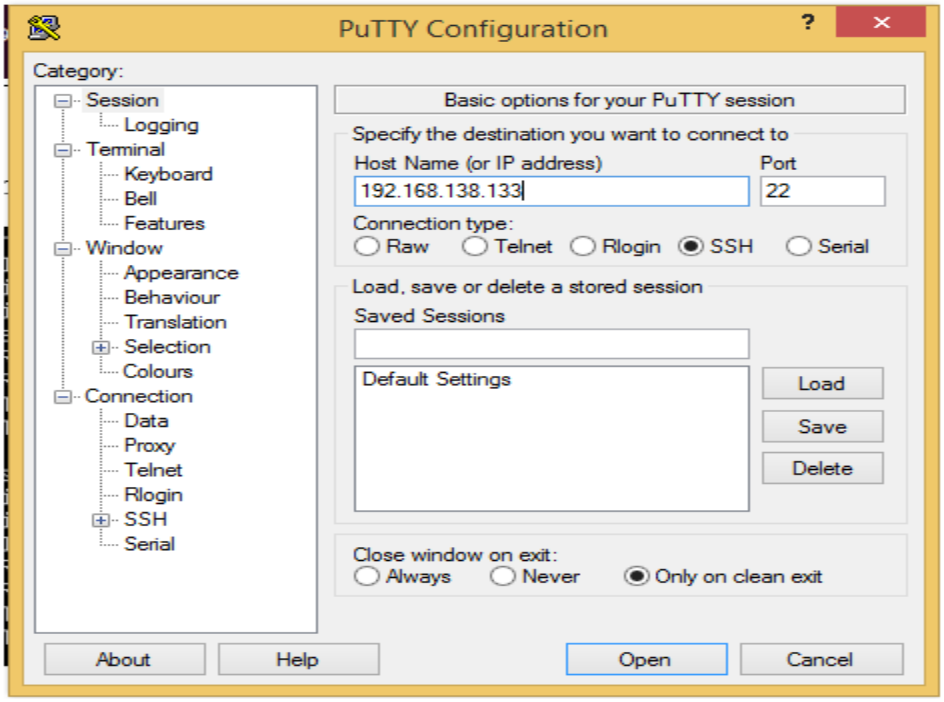


- Cụm mật khẩu được yêu cầu để đăng nhập, sau đó trả lời nó. Nếu đúng, bạn có thể đăng nhập bình thường như sau:

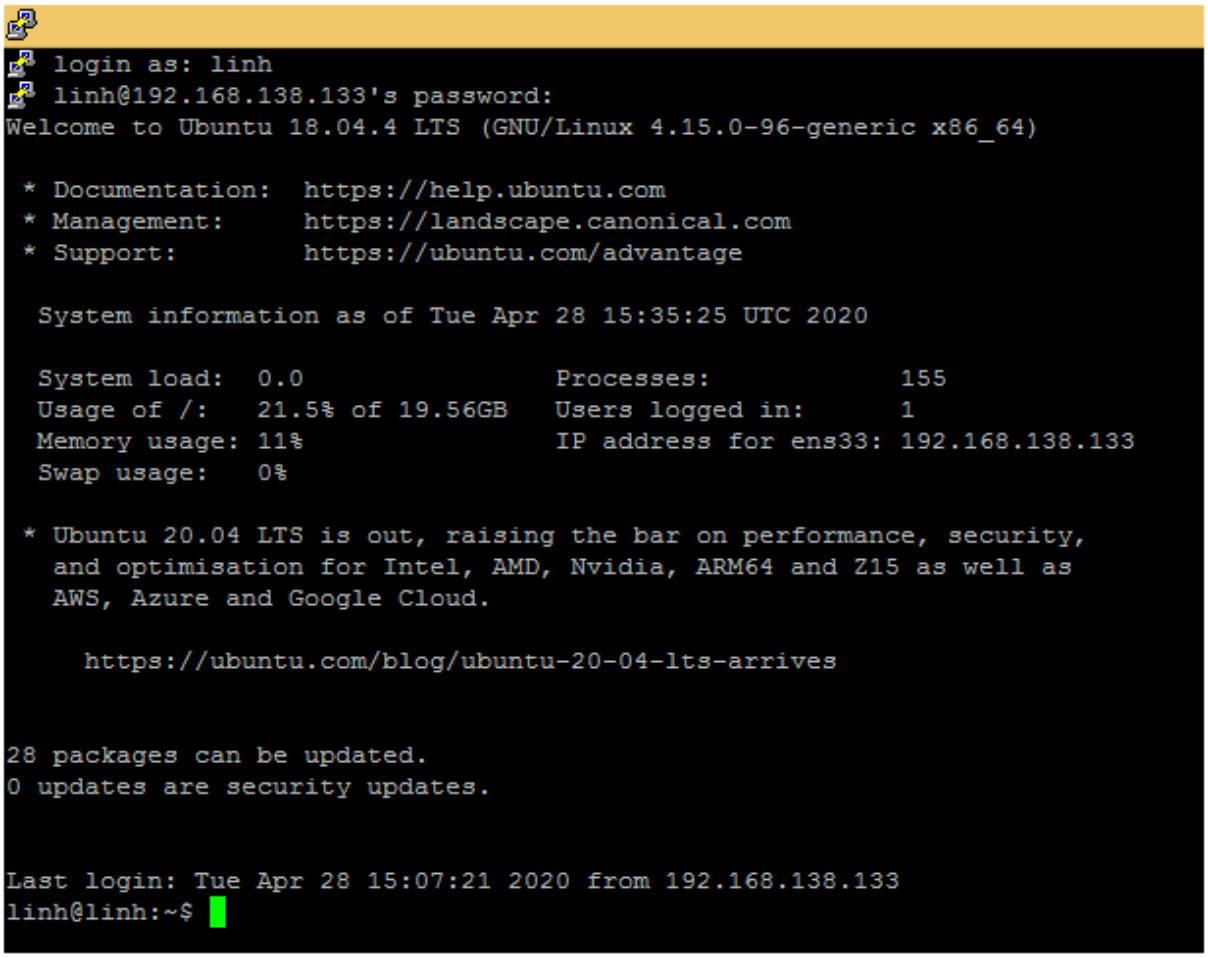
```
login as: linh
linh@192.168.138.133's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-99-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Thu Apr 30 17:51:23 UTC 2020

  System load:  0.0                Processes:           155
  Usage of /:   23.8% of 19.56GB   Users logged in:     1
  Memory usage: 11%                IP address for ens33: 192.168.138.133
  Swap usage:   0%

 * Ubuntu 20.04 LTS is out, raising the bar on performance, security,
   and optimisation for Intel, AMD, Nvidia, ARM64 and Z15 as well as
   AWS, Azure and Google Cloud.

     https://ubuntu.com/blog/ubuntu-20-04-lts-arrives


28 packages can be updated.
0 updates are security updates.


Last login: Thu Apr 30 17:26:45 2020 from 192.168.138.1
linh@linh:~$
```

e.4. Kết nối máy thật với máy ảo qua giao thức SSH:

```
linh@linh:~$ ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.138.133  netmask 255.255.255.0  broadcast 192.168.138.255
        inet6 fe80::20c:29ff:feaa:b982  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:aa:b9:82  txqueuelen 1000  (Ethernet)
        RX packets 2902  bytes 3786716 (3.7 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1607  bytes 130596 (130.5 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 403  bytes 46496 (46.4 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 403  bytes 46496 (46.4 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```



Nếu hiện ra màn hình đen xì là bạn đã kết nối thành công:


login as: linh
linh@192.168.138.133's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-96-generic x86_64)
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage
System information as of Tue Apr 28 15:35:25 UTC 2020
System load: 0.0 Processes: 155
Usage of /: 21.5% of 19.56GB Users logged in: 1
Memory usage: 11% IP address for ens33: 192.168.138.133
Swap usage: 0%
* Ubuntu 20.04 LTS is out, raising the bar on performance, security,
and optimisation for Intel, AMD, Nvidia, ARM64 and Z15 as well as
AWS, Azure and Google Cloud.
https://ubuntu.com/blog/ubuntu-20-04-lts-arrives
28 packages can be updated.
0 updates are security updates.
Last login: Tue Apr 28 15:07:21 2020 from 192.168.138.133
linh@linh:~$