



DevOps external course

Remote access. Client-server implementation of a secure terminal

Lecture 4.6

Module 4 Linux Essentials

Serge Prykhodchenko



<epam>

Services



System-V

The Services

- Background processes that provide services
 - A.k.a daemon process (usually with name xxx**d**)
 - Automatically started during boot
- Example
 - Standard services
 - NTP (ntpd)
 - Secure Shell (sshd)
 - Commonly known services
 - Web (httpd)
 - DNS (named)

Startup Scripts

● SystemV-style startup scripts

- Sun
- /etc/init.d/
- /etc/rc.d/rcn.d/
- Each script is responsible for one daemon or one aspect of system.



Example: sshd in SunOS

```
case "$1" in
  'start')
    if [ -x /usr/local/sbin/sshd ]; then
      echo "Starting the secure shell daemon"
      /usr/local/sbin/sshd &
    fi
    ;;

  'stop')
    echo "Stopping the secure shell daemon"
    pkill -TERM sshd
    ;;
  *)
    echo "Usage: /etc/init.d/sshd { start | stop }"
    ;;
esac
exit 0
```

Startup Scripts – SystemV-style startup scripts (1)

- /etc/rc.d/rcn.d/
 - When init transitions from lower run level to higher one,
 - It runs all the scripts that start with "S" in ascending order with "start" argument
 - When init transitions from high run level to lower one,
 - It runs all the scripts that start with "K" in descending order with "stop" argument

```
❶ linux4 ➤ fyli ... > rc.d ➤ cd /etc/rc.d/
❶ linux4 ➤ fyli ... > rc.d ➤ ls
init.d rc0.d rc1.d rc2.d rc3.d rc4.d rc5.d rc6.d rc.local
❶ linux4 ➤ fyli ... > rc.d ➤ cd rc0.d/
❶ linux4 ➤ fyli ... > rc0.d ➤ ls
K50netconsole K90network
❶ linux4 ➤ fyli ... > rc0.d ➤ █
```

Startup Scripts –SystemV-style startup scripts (2)

- If you write a daemon and want init to start/stop it,
 - write a script and put in /etc/init.d
 - make suitable symbolic link in rcn.d
- ln -s /etc/init.d/initiaId /etc/rc2.d/S61initiaId
- ln -s /etc/init.d/initiaId /etc/rc0.d/K33initiaId

Common Flow of Running a Service

1. Installation

- Through ports, packages, or source tarballs

2. Configuration

- Service specific configuration file(s)
- Make it start on boot
- Or using service(8) command

3. Start the service now

- Calling script under rc.d/*
- Or using "service"

4. Maintenance

- Updating、Restarting

Configuration Files (BSD)

- /usr/local/etc
 - Configuration files of local installed programs are located there.

Daemon	User Program
/usr/local/etc/pure-ftpd.conf	/usr/local/etc/vim/vimrc
/usr/local/etc/apache24/httpd.conf	/usr/local/etc/screenrc

- Default config file usually installed with **.sample** suffix, **.default** suffix, or different suffix for different purpose. (**Copy and rename before using.**)
 - pure-ftpd.conf.sample
 - php.ini-dist
 - php.ini-recommended

Configuration Files

- Most popular styles
 - key <space> value
 - key = value
- Some with local effectiveness (e.g. http server)
 - Markup language-like:

```
<directory /path>
    setting-for-this-path...
</directory>
```

- Samba、rsync、devfs...

```
[xxxx]
settings...
[yyyy]
settings...
```

```
<VirtualHost _default_:443>
ServerAdmin lctseng@cs.nctu.edu.tw
DocumentRoot "/usr/local/www/nic2015/"
ServerName nic2015.nctucs.tw:443
ErrorLog "/var/log/www.nic2015.error"
CustomLog "/var/log/www.nic2015.common" common
TransferLog "/var/log/www.nic2015.access"
<Directory "/usr/local/www/nic2015/">
    AllowOverride All
    Require all granted
</Directory>
```

```
[system=10]
add path 'usb/*' mode 0660 group operator
```

What does RC means?

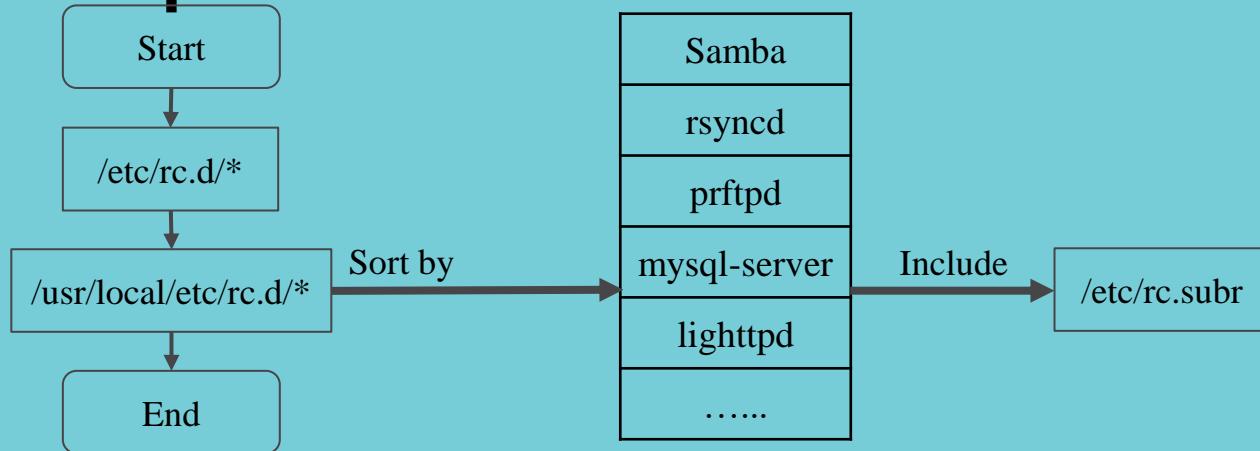
- Run Commands (RunCom)
- Command scripts for auto-reboot and daemon startup

Why do we need RC Script?

- Start services on system startup
- Starting and stopping services in a standard way
- Write configuration in the rc script for easy launch

with RC script	\$ service pure-ftpd start
without RC script	\$ /usr/local/sbin/pure-ftpd -g /var/run/pure-ftpd.pid -A -c50 -B -C8 -D -fftp -H -I15 -lpam -lunix -L10000:8 -m4 -s -U133:022 -u100 -k99 -Z

RC Script



- Dependency between each service is described in header of the script
- rcorder(8) is used to find out dependency ordering of each script
- Each rc script defines what to do when start / stop ...
- /etc/rc.subr defines what to do & check before / after start stop
 - rc.subr(8)

How to use rc script

- Start
 - Start the service and write PID files now
- Stop
 - Terminates the service by killing the process with PID recorded in pidfile
- Restart
 - Restart the service (or just start a new one if not running)
 - Some services implement 'restart' by 'stop + start'
- Enable/Disable
 - Edit /etc/rc.conf with XXX_enable="YES" or "NO"
 - Whether to automatically launch when booting

Systemd

Systemd

- Evolved from System-V
 - Backward compatibility
- Goal: provide a faster booting process
 - Less processes
 - Parallel launching
- Use "targets" replace run-levels
 - Easier to use
- Tool
 - systemctl (1)
 - Similar to "service" tool in FreeBSD

Flow of Running a Service with Systemd

Example with Ubuntu

- Installation
 - Through packages, or source tarballs
 - `apt install apache2`
- Configuration
 - Service specific configuration file(s)
 - `/etc/apache2/*`
 - `systemctl enable apache2`
- Start
 - `systemctl start apache2`
- Maintenance
 - Updating、Restarting

How to use systemctl

- Usage
 - `systemctl [OPTIONS...] {COMMAND} ...`
- Common commands
 - enable / disable
 - Enable/disable launch when booting
 - start / stop / reload / restart / status
 - condrestart
 - Restart only if service is running

Unit files

- Define services
 - /lib/systemd/system/*
 - Similar to /etc/rc.d/* in FreeBSD
- systemd.service (5)
- Example 1: (simple service)
 - Type
 - Simple: main process keeps running
 - Forking: main process forks and exits
 - ExecStart
 - Command to launch the service
 - WantedBy
 - Run this service at which target

[Unit]

Description=Some simple daemon

[Service]

Type=forking

ExecStart=/usr/sbin/my-simple-daemon -d

PIDFile=/var/run/my-daemon.pid

[Install]

WantedBy=multi-user.target

Unit files

- apache2.service

- After

- Dependency. Start service after dependency is fulfilled

- ExecStop / ExecReload

- Custom command to stop / reload the service

```
[Unit]
```

```
Description=The Apache HTTP Server
```

```
After=network.target remote-fs.target nss-lookup.target
```

```
[Service]
```

```
Type=forking
```

```
Environment=APACHE_STARTED_BY_SYSTEMD=true
```

```
ExecStart=/usr/sbin/apachectl start
```

```
ExecStop=/usr/sbin/apachectl stop
```

```
ExecReload=/usr/sbin/apachectl graceful
```

```
PrivateTmp=true
```

```
Restart=on-abort
```

```
[Install]
```

```
WantedBy=multi-user.target
```

Unit files

- When enable a service, it will create links from "lib/systemd/system/*.service" to "etc/systemd/system/XXX.target.wants/*"
- /etc/systemd/system/multi-user.target.wants/apache2.service
 - -> /lib/systemd/system/apache2.service

```
13:18 lctseng@lctseng-sa-ubuntu(10.0.2.15) [/etc/systemd/system/multi-user.target.wants]
[XD] % ll
total 8
drwxr-xr-x  2 root root 4096 Sep 30 12:21 .
drwxr-xr-x 15 root root 4096 Sep 28 23:25 ..
lrwxrwxrwx  1 root root   35 Sep 30 12:21 apache2.service -> /lib/systemd/system/apache2
lrwxrwxrwx  1 root root   31 Aug  5 19:24 atd.service -> /lib/systemd/system/atd.service
lrwxrwxrwx  1 root root   41 Aug  5 19:23 console-setup.service -> /lib/systemd/system/c
lrwxrwxrwx  1 root root   32 Aug  5 19:23 cron.service -> /lib/systemd/system/cron.servi
lrwxrwxrwx  1 root root   36 Aug  5 19:24 ebtables.service -> /lib/systemd/system/ebtabl
lrwxrwxrwx  1 root root   38 Aug  5 19:24 irqbalance.service -> /lib/systemd/system/irqb
lrwxrwxrwx  1 root root   33 Aug  5 19:24 lxcfs.service -> /lib/systemd/system/lxcfs.ser
lrwxrwxrwx  1 root root   42 Aug  5 19:24 lxd-containers.service -> /lib/systemd/system/
```

Basic Level Comparison Between SystemV & Systemd

Comments	SysVinit	Systemd
Start a service	service dummy start	systemctl start dummy.service
Stop a service	service dummy stop	systemctl stop dummy.service
Restart a service	service dummy restart	systemctl restart dummy.service
Reload a service	service dummy reload	systemctl reload dummy.service
Service status	service dummy status	systemctl status dummy.service
Restart a service if already running	service dummy conrestart	systemctl conrestart dummy.service
Enable service at startup	chkconfig dummy on	systemctl enable dummy.service
Disable service at startup	chkconfig dummy off	systemctl disable dummy.service
Check if a service is enabled at startup	chkconfig dummy	systemctl is-enabled dummy.service
Create a new service file or modify configuration	chkconfig dummy --add	systemctl daemon-reload
System halt	0	runlevel0.target, poweroff.target
Single user mode	1, s, single	runlevel1.target, rescue.target
Multi user	2	runlevel2.target, multi-user.target
Multi user with Network	3	runlevel3.target, multi-user.target
Experimental	4	runlevel4.target, multi-user.target
Multi user, with network, graphical mode	5	runlevel5.target, graphical.target
Reboot	6	runlevel6.target, reboot.target
Emergency Shell	emergency	emergency.target
Change to multi user runlevel/target	telinit 3	systemctl isolate multi-user.target (OR systemctl isolate runlevel3.target)
Set multi-user target on next boot	sed s/^id:.*:initdefault:/ id:3:initdefault:/	ln -sf /lib/systemd/system/multi-user.target /etc/systemd/system/default.target
Check current runlevel	runlevel	systemctl get-default
Change default runlevel	sed s/^id:.*:initdefault:/ id:3:initdefault:/	systemctl set-default multi-user.target

Basic Level Comparison Between SystemV & Systemd

System halt	halt	systemctl halt
Power off the system	poweroff	systemctl poweroff
Restart the system	reboot	systemctl reboot
Suspend the system	pm-suspend	systemctl suspend
Hibernate	pm-hibernate	systemctl hibernate
Follow the system log file	tail -f /var/log/messages or tail -f /var/log/syslog	journalctl -f

Systemd new commands

Execute a systemd command on remote host	systemctl dummy.service start -H user@host
Check boot time	systemd-analyze or systemd-analyze time
Kill all processes related to a service	systemctl kill dummy
Get logs for events for today	journalctl --since=today
Hostname and other host related information	hostnamectl
Date and time of system with timezone and other information	timedatectl

Other init systems

Different Init Systems

	BSD rc	sysvinit (System V)	systemd	launchd	OpenRC
Operating Systems	NetBSD, FreeBSD, OpenBSD	Debian (< 8.0)	Ubuntu, Debian (>= 8.0)	Mac OS X	Gentoo Linux, FreeBSD, NetBSD
Example commands	service	init, telinit, runlevel	systemctl	launchctl	rc-service, rc-update
Service config files	/etc/rc.d	/etc/init.d/	/lib/systemd/ system/	/System/Library/LaunchD aemons	/etc/init.d/

QUOTING

Quotas

- https://access.redhat.com/knowledge/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Storage_Administration_Guide/ch-disk-quotas.html
- Quotas give us the ability to keep track of users' disk usage: both blocks (disk space) and inodes (number of files)
- `quota rpm` must be installed
- For both blocks and inodes, we quotas allow hard limits and soft limits:
 - Soft limit: user is allowed to exceed a soft limit, but they will be warned, and after a grace period, they cannot increase usage
 - Hard limit: user is never allowed to exceed the hard limit
- We enable quotas for a filesystem
- Quotas can be applied to users and/or groups
- System administrator can report on all users' disk usage status
- Each user can see their own disk usage status (quota information)

Turning quotas on (and off)

- Example: enabling quotas on /home (separate /home filesystem)

- In /etc/fstab, add the `usrquota`, `grpquota` mount options for the file system mounted on the /home mount point
- Initialize the quota database files for /home with the command

```
quotacheck -cug /home
```

- c: don't read quota files, create new quota database files
- u: do user quotas
- g: do group quotas

- Turn quotas on
- quotaon -vaug # turn quotas on
 - v: display a message for each filesystem affected
 - a: turn quotas on for all automatically mounted file systems according to /etc/fstab
 - u: user quotas
 - g: group quotas
- repquota -a # report on quotas
- Turn quotas off
- quotaoff -vaug # turn quotas off
- quotaoff -vaug; quotacheck -vaug; quotaon -vaug #single user mode

Setting Quotas

- To set a quota for a user, as root

```
edquota username
```

- where
 - you'll see (example) DO NOT edit blocks or inodes, just soft and hard limits!

Disk quotas for user tgk (uid 107):

Filesystem	blocks	soft	hard	inodes	soft	hard
/dev/sda8	108	1000	2000	1	0	0

or this command can be used in scripts

```
setquota -u username soft hard isoft ihard fs
```

- where
 - username is the name of the user
 - soft is the block soft limit
 - hard is the block hard limit
 - isoft is the inode soft limit
 - ihard is the inode hard limit
 - fs is the file system mount point (e.g. /home)

Quota Grace Period

- To set the grace period for all users

```
edquota -t      # edit grace period
```

- where you'll see something like this (note units)

```
Grace period before enforcing soft limits for users:
```

```
Time units may be: days, hours, minutes, or seconds
```

Filesystem	Block grace period	Inode grace period
/dev/mapper/VolGroup00-LogVol00	8days	8days

- To set the grace period for an individual user

```
edquota -T tgk
```

- where you'll see something like this (note units)

```
Times to enforce softlimit for user tgk (uid 498):
```

```
Time units may be: days, hours, minutes, or seconds
```

Filesystem	block grace	inode grace
/dev/mapper/VolGroup00-LogVol00	unset	unset

quota and repquota commands

- individual users can check their individual quota status with `quota` command:
 - shows
 - block usage and limits
 - inode usage and limits
 - remainder on grace period if over soft limit
- System administrator can print report of all users quota status (see also `warnquota`):
 - `repquota -a`
 - shows for each user what they've used, soft limits, hard limits, and remainder of grace periods if that user has entered one of their grace periods

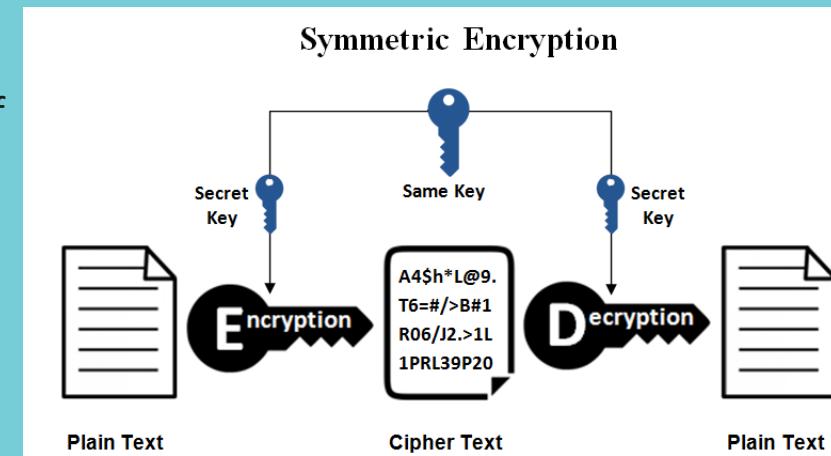
CHALLENGES

1. Secure client-server connection
2. Automation

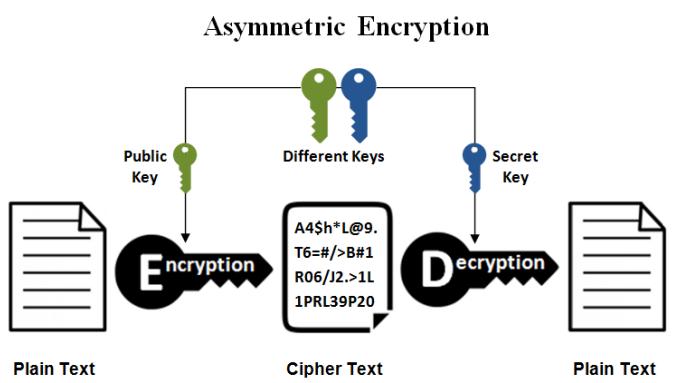
Secure client-server connection

Asymmetric cryptography vs symmetric

- This is the simplest kind of encryption that involves only one secret key to cipher and decipher information. Symmetrical encryption is an old and best-known technique. It uses a secret key that can either be a number, a word or a string of random letters. It is blended with the plain text of a message to change the content in a particular way. The sender and the recipient should know the secret key that is used to encrypt and decrypt all the messages.
- Blowfish, AES, RC4, DES, RC5, and RC6 are examples of symmetric encryption. The most widely used symmetric algorithm is AES-128, AES-192, and AES-256.
- The main disadvantage of the symmetric key encryption is that all parties involved have to exchange the key used to encrypt the data before they can decrypt it.



Secure client-server connection



- Asymmetrical encryption is also known as public key cryptography, which is a relatively new method, compared to symmetric encryption. Asymmetric encryption uses two keys to encrypt a plain text. Secret keys are exchanged over the Internet or a large network. It ensures that malicious persons do not misuse the keys. It is important to note that anyone with a secret key can decrypt the message and this is why asymmetrical encryption uses two related keys to boosting security. A public key is made freely available to anyone who might want to send you a message. The second private key is kept a secret so that you can only know.

- A message that is encrypted using a public key can only be decrypted using a private key, while also, a message encrypted using a private key can be decrypted using a public key. Security of the public key is not required because it is publicly available and can be passed over the internet. Asymmetric key has a far better power in ensuring the security of information transmitted during communication.
- Asymmetric encryption is mostly used in day-to-day communication channels, especially over the Internet. Popular asymmetric key encryption algorithm includes ElGamal, RSA, DSA, Elliptic curve techniques, PKCS

Secure client-server connection

Asymmetric cryptography vs symmetric

- Symmetric encryption uses a single key that needs to be shared among the people who need to receive the message while asymmetrical encryption uses a pair of public key and a private key to encrypt and decrypt messages when communicating.
- Symmetric encryption is an old technique while asymmetric encryption is relatively new.
- Asymmetric encryption was introduced to complement the inherent problem of the need to share the key in symmetrical encryption model, eliminating the need to share the key by using a pair of publicprivate keys.
- Asymmetric encryption takes relatively more time than the symmetric encryption.

Secure client-server connection

telnet, rlogin, RDS vs ssh

Key Differences Between Telnet and SSH

- Telnet and SSH both serve the same purpose and provides the connectivity to the remote server but Telnet is conventional protocol, although it is still in use in the various application. SSH is the replacement for Telnet and has some enhanced features too.
- Telnet doesn't provide any security mechanism whereas SSH is more secure and provides security measures.
- In Telnet transmits data in plain text that is the reason it is vulnerable to security attacks. On the other hand, SSH uses encryption for transmitted data and security breach does not likely occur. SSH can withstand eavesdropping, man in the middle and insertion/ replay attacks.
- Telnet doesn't provide authentication facility while SSH provides user authentication.
- Telnet works with a private network. In contrast, SSH works with a public network.
- Telnet communicates via port number 23 over TCP/IP. As against, SSH uses port number 22 for communication.

Secure client-server connection

telnet, rlogin, RDS vs ssh

Key differences between Rlogin and SSH

- SSH traffic is encrypted while Rlogin traffic is not
- SSH authenticates the user while Rlogin does not
- SSH can be used for automation while Rlogin cannot
- Rlogin is no longer being used in favor of SSH

Secure client-server connection

telnet, rlogin, RDS vs ssh

Key differences between RDS and SSH

- + The Remote Desktop Protocol is solely used for accessing Windows virtual machines (VMs) and physical Windows servers (as opposed to Linux® servers). From a user perspective, RDP provides a Windows Graphical User Interface (GUI) experience, making servers more accessible to a wider range of employees — with or without a technical background.
- Because RDP ports often need to be connected to the internet for remote access, for security purposes, admins should protect their RDP instances with a virtual private network (VPN) and/or a form of multi-factor authentication (MFA). RDP ports can be vulnerable to attacks when exposed to the internet.

Automation

- Automated your SSH login with Public Key Authentication
- Using native Linux-based OS shell scripting language (sh, bash, etc)
- Automated scripts over SSH
- <https://www.openssh.com/>

Description

- Completely open source project with free licensing
- Strong cryptography (AES, ChaCha20, RSA, ECDSA, Ed25519...)

Encryption is started before authentication, and no passwords or other information is transmitted in the clear.

Encryption is also used to protect against spoofed packets. A number of different ciphers and key types are available, and legacy options are usually phased out in a reasonable amount of time.

- X11 forwarding (which also encrypts X Window System traffic)

X11 forwarding allows the encryption of remote X windows traffic, so that nobody can snoop on your remote xterms or insert malicious commands. The program automatically sets DISPLAY on the server machine, and forwards any X11 connections over the secure channel. Fake Xauthority information is automatically generated and forwarded to the remote machine; the local client automatically examines incoming X11 connections and replaces the fake authorization data with the real data (never telling the remote machine the real information).

- Port forwarding (encrypted channels for legacy protocols)

Port forwarding allows forwarding of TCP/IP connections to a remote machine over an encrypted channel.

Insecure internet applications like POP can be secured with this.

- Strong authentication (public keys, one-time passwords)

Strong authentication protects against several security problems: IP spoofing, fakes routes and DNS spoofing.

Some authentication methods include public key authentication, one-time passwords with s/key and authentication using Kerberos (only in-portable)

Description

Agent forwarding

An authentication agent, running in the user's laptop or local workstation, can be used to hold the user's authentication keys. OpenSSH automatically forwards the connection to the authentication agent over any connections, and there is no need to store the authentication keys on any machine in the network (except the user's own local machine). The authentication protocols never reveal the keys; they can only be used to verify that the user's agent has a certain key. Eventually the agent could rely on a smart card to perform all authentication computations.

Interoperability

Interoperability between implementations is a goal, but not a promise. As OpenSSH development progresses, older protocols, ciphers, key types and other options that have known weaknesses are routinely disabled.

SFTP client and server support.

Complete SFTP support is included.

Optional data compression

Data compression before encryption improves the performance for slow network links

Realization. Prerequisites

- VirtualBox ver. 5+
 - Installed Ubuntu 16.04 with SSH server (server edition) with
“Network”-> “Adapter” set to “Bridged Adapter”
 - Clone(-s) of Installed Ubuntu 16.04 with SSH server (server edition) with
“Network”-> “Adapter” set to “Bridged Adapter” (full clone with reinitialized MAC-addresses)
- SSH client :

<https://mobaxterm.mobatek.net/download.html> (for Windows)

or

<https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html> (Windows, Linux, but simple)

or

• <https://www.termius.com/> (Windows, Linux, Mac)

Realization (1)

ubuntu16srvr Clone1 [Running] - Oracle VM VirtualBox

```
File Machine View Input Devices Help
student@ubuntu16srvr:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: emp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:3e:2e brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.104/24 brd 192.168.0.255 scope global emp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe3e:c32e/64 scope link
        valid_lft forever preferred_lft forever
student@ubuntu16srvr:~$ ifconfig
emp0s3    Link encap:Ethernet HWaddr 08:00:27:3e:2e
          inet addr:192.168.0.104 Bcast:192.168.0.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe3e:c32e/64 Scope:Link
            UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
            RX packets:7815 errors:0 dropped:0 overruns:0 frame:0
            TX packets:1917 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:11073677 (11.0 MB) TX bytes:160983 (160.9 KB)

lo       Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
            UP LOOPBACK RUNNING MTU:65536 Metric:1
            RX packets:176 errors:0 dropped:0 overruns:0 frame:0
            TX packets:176 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1
            RX bytes:13296 (13.2 KB) TX bytes:13296 (13.2 KB)

student@ubuntu16srvr:~$ _
```



ubuntu16srvr [Running] - Oracle VM VirtualBox

```
File Machine View Input Devices Help
student@ubuntu16srvr:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: emp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:e6:24:b0 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.103/24 brd 192.168.0.255 scope global emp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fee6:24b0/64 scope link
        valid_lft forever preferred_lft forever
student@ubuntu16srvr:~$ ifconfig
emp0s3    Link encap:Ethernet HWaddr 08:00:27:e6:24:b0
          inet addr:192.168.0.103 Bcast:192.168.0.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fee6:24b0/64 Scope:Link
            UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
            RX packets:6108 errors:0 dropped:0 overruns:0 frame:0
            TX packets:1614 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:8475352 (8.4 MB) TX bytes:121933 (121.9 KB)

lo       Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
            UP LOOPBACK RUNNING MTU:65536 Metric:1
            RX packets:176 errors:0 dropped:0 overruns:0 frame:0
            TX packets:176 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1
            RX bytes:13296 (13.2 KB) TX bytes:13296 (13.2 KB)

student@ubuntu16srvr:~$ _
```



Realization (2)

ubuntu16srvr Clone1 [Running] - Oracle VM VirtualBox

```
File Machine View Input Devices Help
student@ubuntu16srvr:~$ ssh student@192.168.0.103
The authenticity of host '192.168.0.103 (192.168.0.103)' can't be established.
ECDSA key fingerprint is SHA256:/z3Q8P7qBEewbHT2topjj/Vgfz9w4mT4idf+gx+jK7E.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.0.103' (ECDSA) to the list of known hosts.
student@192.168.0.103's password:
Welcome to Ubuntu 16.04.5 LTS (GNU/Linux 4.4.0-131-generic x86_64)

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

196 packages can be updated.
143 updates are security updates.

New release '18.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Tue Aug 18 16:48:56 2020
student@ubuntu16srvr:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host
                valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:e6:24:b0 brd ff:ff:ff:ff:ff:ff
        inet 192.168.0.103/24 brd 192.168.0.255 scope global enp0s3
            valid_lft forever preferred_lft forever
            inet6 fe80::a00:27ff:fe6:24b0/64 scope link
                valid_lft forever preferred_lft forever
student@ubuntu16srvr:~$
```

ubuntu16srvr [Running] - Oracle VM VirtualBox

```
File Machine View Input Devices Help
student@ubuntu16srvr:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host
                valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:e6:24:b0 brd ff:ff:ff:ff:ff:ff
        inet 192.168.0.103/24 brd 192.168.0.255 scope global enp0s3
            valid_lft forever preferred_lft forever
            inet6 fe80::a00:27ff:fe6:24b0/64 scope link
                valid_lft forever preferred_lft forever
student@ubuntu16srvr:~$ ifconfig
enp0s3    Link encap:Ethernet HWaddr 08:00:27:e6:24:b0
          inet addr:192.168.0.103 Bcast:192.168.0.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe6:24b0/64 Scope:Link
            UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
            RX packets:6108 errors:0 dropped:0 overruns:0 frame:0
            TX packets:1614 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:8475352 (8.4 MB) TX bytes:121933 (121.9 KB)

lo       Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
            UP LOOPBACK RUNNING MTU:65536 Metric:1
            RX packets:176 errors:0 dropped:0 overruns:0 frame:0
            TX packets:176 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1
            RX bytes:13296 (13.2 KB) TX bytes:13296 (13.2 KB)
student@ubuntu16srvr:~$
```

Realization (3)

ubuntu16srvr Clone1 [Running] - Oracle VM VirtualBox

```
File Machine View Input Devices Help
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
inet 127.0.0.1/8 scope host lo
    valid_lft forever preferred_lft forever
inet6 ::1/128 scope host
    valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:e6:24:b0 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.103/24 brd 192.168.0.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe6:24b0/64 scope link
        valid_lft forever preferred_lft forever
student@ubuntu16srvr:~$ exit
logout
Connection to 192.168.0.103 closed.
student@ubuntu16srvr:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/student/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/student/.ssh/id_rsa.
Your public key has been saved in /home/student/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:ugRy4ADiE4VScJmmJpE0hMLatMt9/M1SnCBsdISg5mo student@ubuntu16srvr
The key's randomart image is:
+---[RSA 2048]---+
|00+= o. |
|X+B . o. |
|*0 . = . |
|*o+ o + . |
|+oo= o .So . |
| oo.o o. + |
| ..... + |
| E. . o o |
| . . . |
+---[SHA256]---+
student@ubuntu16srvr:~$ _
```



ubuntu16srvr [Running] - Oracle VM VirtualBox

```
File Machine View Input Devices Help
student@ubuntu16srvr:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:e6:24:b0 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.103/24 brd 192.168.0.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe6:24b0/64 scope link
        valid_lft forever preferred_lft forever
student@ubuntu16srvr:~$ ifconfig
enp0s3    Link encap:Ethernet HWaddr 08:00:27:e6:24:b0
          inet addr:192.168.0.103 Bcast:192.168.0.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe6:24b0/64 Scope:Link
            UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
            RX packets:6108 errors:0 dropped:0 overruns:0 frame:0
            TX packets:1614 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:8475352 (8.4 MB) TX bytes:121933 (121.9 KB)

lo      Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
            UP LOOPBACK RUNNING MTU:65536 Metric:1
            RX packets:176 errors:0 dropped:0 overruns:0 frame:0
            TX packets:176 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1
            RX bytes:13296 (13.2 KB) TX bytes:13296 (13.2 KB)

student@ubuntu16srvr:~$
```



Realization (4)

ubuntu16srvr Clone1 [Running] - Oracle VM VirtualBox

```
File Machine View Input Devices Help
student@ubuntu16srvr:~$ ssh-copy-id student@192.168.0.103
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/student/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install all the new keys
student@192.168.0.103's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'student@192.168.0.103'"
and check to make sure that only the key(s) you wanted were added.

student@ubuntu16srvr:~$ ssh student@192.168.0.103
Welcome to Ubuntu 16.04.5 LTS (GNU/Linux 4.4.0-131-generic x86_64)

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

196 packages can be updated.
143 updates are security updates.

New release '18.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Tue Aug 18 17:00:17 2020 from 192.168.0.104
student@ubuntu16srvr:~$ _
```



ubuntu16srvr [Running] - Oracle VM VirtualBox

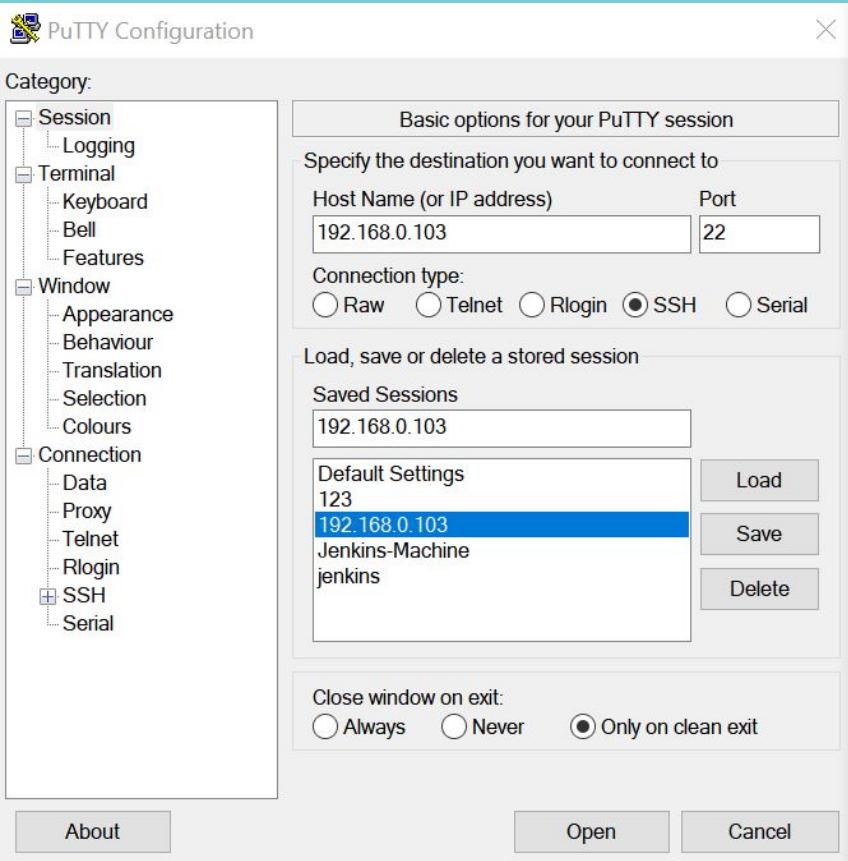
```
File Machine View Input Devices Help
inet6 ::1/128 scope host
      valid_lft forever preferred_lft forever
2: emp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
      link/ether 08:00:27:e6:24:b0 brd ff:ff:ff:ff:ff:ff
      inet 192.168.0.103/24 brd 192.168.0.255 scope global emp0s3
          valid_lft forever preferred_lft forever
      inet6 fe80::a00:27ff:fe6:24b0/64 scope link
          valid_lft forever preferred_lft forever
student@ubuntu16srvr:~$ ifconfig
emp0s3  Link encap:Ethernet HWaddr 08:00:27:e6:24:b0
        inet addr:192.168.0.103 Bcast:192.168.0.255 Mask:255.255.255.0
        inet6 addr: fe80::a00:27ff:fe6:24b0/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:6108 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1614 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:8475352 (8.4 MB) TX bytes:121933 (121.9 KB)

lo    Link encap:Local Loopback
        inet addr:127.0.0.1 Mask:255.0.0.0
        inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:65536 Metric:1
          RX packets:176 errors:0 dropped:0 overruns:0 frame:0
          TX packets:176 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:13296 (13.2 KB) TX bytes:13296 (13.2 KB)

student@ubuntu16srvr:~$ who
student  tty1          2020-08-18 16:48
student  pts/0          2020-08-18 17:05 (192.168.0.104)
student@ubuntu16srvr:~$ w
 17:06:43 up 18 min,  2 users,  load average: 0.00,  0.02,  0.00
USER   TTY     FROM             LOGIN   IDLE   JCPU   PCPU WHAT
student  tty1          16:48   2.00s  0.06s  0.00s w
student  pts/0          192.168.0.104 17:05   1:08  0.03s  0.03s -bash
student@ubuntu16srvr:~$ _
```



Realization (5)



A terminal window titled 'ubuntu16srver [Running] - Oracle VM VirtualBox' is displayed. The window title bar includes standard icons for minimize, maximize, and close. The terminal menu bar includes File, Machine, View, Input, Devices, and Help. The terminal window displays the output of several commands:

```
student@ubuntu16srver:~$ ifconfig
enp0s3    Link encap:Ethernet HWaddr 08:00:27:e6:24:b0
          inet 192.168.0.103  Bcast 192.168.0.255  Mask 255.255.255.0
          inet6 fe80::a00:27ff:fe6:24b0/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:6108 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1614 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:8475352 (8.4 MB)  TX bytes:121933 (121.9 KB)

lo       Link encap:Local Loopback
          inet 127.0.0.1  Mask 255.0.0.0
          inet6 ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:65536 Metric:1
          RX packets:176 errors:0 dropped:0 overruns:0 frame:0
          TX packets:176 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:13296 (13.2 KB)  TX bytes:13296 (13.2 KB)

student@ubuntu16srver:~$ who
student  pts/0     2020-08-18 16:48
student  pts/0     2020-08-18 17:05 (192.168.0.104)
student@ubuntu16srver:~$ w
17:06:43 up 18 min, 2 users, load average: 0.00, 0.02, 0.00
USER   TTY      FROM          LOGIN@  IDLE   JCPU   PCPU WHAT
student  pts/0          16:48   2.00s  0.06s  0.00s w
student  pts/0     192.168.0.104  17:05   1:08   0.03s  0.03s -bash
student@ubuntu16srver:~$ _
```

Realization (6)

```
student@ubuntu16srvr: ~
login as: student
student@192.168.0.103's password:
Welcome to Ubuntu 16.04.5 LTS (GNU/Linux 4.4.0-131-generic x86_64)

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

196 packages can be updated.
143 updates are security updates.

New release '18.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Tue Aug 18 17:05:35 2020 from 192.168.0.104
student@ubuntu16srvr:~$
```

```
ubuntu16srvr [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
valid_ift forever preferred_ift forever
inet6 fe80::a00:27ff:fe6:24b0/64 scope link
valid_ift forever preferred_ift forever
student@ubuntu16srvr:~$ ifconfig
ens3 Link encap:Ethernet HWaddr 08:00:27:e6:24:b0
      inet addr:192.168.0.103 Bcast:192.168.0.255 Mask:255.255.255.0
      inet6 addr: fe80::a00:27ff:fe6:24b0/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:6108 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1614 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:8475352 (8.4 MB) TX bytes:121933 (121.9 KB)

lo Link encap:Local Loopback
      inet addr:127.0.0.1 Mask:255.0.0.0
      inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:65536 Metric:1
          RX packets:176 errors:0 dropped:0 overruns:0 frame:0
          TX packets:176 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:13296 (13.2 KB) TX bytes:13296 (13.2 KB)

student@ubuntu16srvr:~$ who
student    tty1          2020-08-18 16:48
student    pts/0          2020-08-18 17:05 (192.168.0.104)
student@ubuntu16srvr:~$ w
 17:06:43 up 18 min,  2 users,  load average: 0.00, 0.02, 0.00
USER   TTY     FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
student  tty1          16:48   2.00s  0.06s  0.00s w
student  pts/0          192.168.0.104 17:05   1:08  0.03s  0.03s -bash
student@ubuntu16srvr:~$ w
 17:20:06 up 31 min,  3 users,  load average: 0.00, 0.00, 0.00
USER   TTY     FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
student  tty1          16:48   2.00s  0.07s  0.00s w
student  pts/0          192.168.0.104 17:05   4:15  0.03s  0.03s -bash
student  pts/1          192.168.0.102 17:19   4.00s  0.04s  0.04s -bash
student@ubuntu16srvr:~$
```

Realization (7)

The image shows two windows side-by-side. The left window is a Windows PowerShell session with a dark blue background. The right window is a terminal session on an Oracle VM VirtualBox running Ubuntu 16.04 LTS (64-bit), with a light gray background.

Windows PowerShell:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\User> ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\User/.ssh/id_rsa):
C:\Users\User/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\User/.ssh/id_rsa.
Your public key has been saved in C:\Users\User/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:7Ny0qsvAd28SgX1QEvndpUgAT6+DsNLqsmWlUwFYrU user@PC-Lite
The key's randomart image is:
+---[RSA 2048]----+
|   o.  ..o.=.+|
| . . .  .. =o|
|   E.  o o . =|
|   . . . o o ..o|
|   o . S..  o . |
|   +.+ o... |
|   o  =*.+ .. |
| .o.  o+=o .... |
| .+.oo...  o. |
+---[SHA256]----+
PS C:\Users\User>
```

ubuntu16srvr [Running] - Oracle VM VirtualBox:

```
File Machine View Input Devices Help
valid_lft forever preferred_lft forever
inet6 fe80::a00:27ff:fee6:24b0/64 scope link
valid_lft forever preferred_lft forever
student@ubuntu16srvr:~$ ifconfig
enp0s3    Link encap:Ethernet HWaddr 08:00:27:e6:24:b0
          inet addr:192.168.0.103  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fee6:24b0%enp0s3/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6108 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1614 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:8475352 (8.4 MB)  TX bytes:121933 (121.9 KB)

lo       Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:65536  Metric:1
          RX packets:176 errors:0 dropped:0 overruns:0 frame:0
          TX packets:176 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:13296 (13.2 KB)  TX bytes:13296 (13.2 KB)

student@ubuntu16srvr:~$ who
student  ttu1      2020-08-18 16:48
student  pts/0      2020-08-18 17:05 (192.168.0.104)
student@ubuntu16srvr:~$ w
17:06:43 up 18 min,  2 users,  load average: 0.00, 0.02, 0.00
USER     TTY     FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
student  ttu1      16:48    2.00s  0.06s  0.00s  u
student  pts/0      192.168.0.104 17:05    1:08  0.03s  0.03s -bash
student@ubuntu16srvr:~$ w
17:20:06 up 31 min,  3 users,  load average: 0.00, 0.00, 0.00
USER     TTY     FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
student  ttu1      16:48    2.00s  0.07s  0.00s  u
student  pts/0      192.168.0.104 17:05    4:15  0.03s  0.03s -bash
student  pts/1      192.168.0.102 17:19    4.00s  0.04s  0.04s -bash
student@ubuntu16srvr:~$ _
```

Realization (8)

```
student@ubuntu16srvr: ~
| E. o o . =
| . . . o o ..o |
| o . S.. o . |
| +.+ o... |
| o =*.+ .. |
| .o .o+=o .... |
| .+.oo.. o. |
+---[SHA256]----+
PS C:\Users\User> ssh student@192.168.0.103
The authenticity of host '192.168.0.103 (192.168.0.103)' can't be established.
ECDSA key fingerprint is SHA256:/z3Q8P7qBewbHT2topjj/Vgfz9w4mT4idf+gx+jK7E.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.0.103' (ECDSA) to the list of known hosts.
student@192.168.0.103's password:
Welcome to Ubuntu 16.04.5 LTS (GNU/Linux 4.4.0-131-generic x86_64)
Welcome to Ubuntu 16.04.5 LTS (GNU/Linux 4.4.0-131-generic x86_64)

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

196 packages can be updated.
143 updates are security updates.

New release '18.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Tue Aug 18 17:19:36 2020 from 192.168.0.102
student@ubuntu16srvr: $
```

```
ubuntu16srvr [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
inet6 addr: fe80::a00:27ff:fe6:24b0/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:6108 errors:0 dropped:0 overruns:0 frame:0
TX packets:1614 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:8475352 (8.4 MB) TX bytes:121933 (121.9 KB)

lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:65536 Metric:1
RX packets:176 errors:0 dropped:0 overruns:0 frame:0
TX packets:176 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1
RX bytes:13296 (13.2 KB) TX bytes:13296 (13.2 KB)

student@ubuntu16srvr:~$ who
student  pts/0     2020-08-18 16:48
student  pts/0     2020-08-18 17:05 (192.168.0.104)
student@ubuntu16srvr:~$ w
17:06:43 up 18 min, 2 users,  load average: 0.00, 0.02, 0.00
USER   TTY      FROM              LOGIN@  IDLE   JCPU   PCPU WHAT
student  tty1          16:48   2.00s  0.06s  0.00s w
student  pts/0   192.168.0.104  17:05   1:08   0.03s  0.03s -bash
student@ubuntu16srvr:~$ w
17:20:06 up 31 min, 3 users,  load average: 0.00, 0.00, 0.00
USER   TTY      FROM              LOGIN@  IDLE   JCPU   PCPU WHAT
student  tty1          16:48   2.00s  0.07s  0.00s w
student  pts/0   192.168.0.104  17:05   4:15   0.03s  0.03s -bash
student  pts/1   192.168.0.102  17:19   4.00s  0.04s  0.04s -bash
student@ubuntu16srvr:~$ w
17:23:43 up 35 min, 3 users,  load average: 0.00, 0.00, 0.00
USER   TTY      FROM              LOGIN@  IDLE   JCPU   PCPU WHAT
student  tty1          16:48   1.00s  0.07s  0.00s w
student  pts/0   192.168.0.104  17:05   7:52   0.03s  0.03s -bash
student  pts/1   192.168.0.102  17:23  12.00s  0.04s  0.04s -bash
student@ubuntu16srvr:~$ _
```

Establish SSH connection without login/password (using MobaXTerm)

192.168.88.214 (student)

Terminal Sessions View X server Tools Games Settings Macros Help

Quick connect...

```
student@ubuntu16srvr:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/student/.ssh/id_rsa):
Created directory '/home/student/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/student/.ssh/id_rsa.
Your public key has been saved in /home/student/.ssh/id_rsa.pub.
The key's fingerprint is:
SHA256:d5ml+KpzYe7sBrCODRgoM88/l6frRIqf9vmnQuLLlGo student@ubuntu16srvr
The key's randomart image is:
+---[RSA 2048]---+
+ . .
* o .
o . + o . =
. .@S o =
.oE * .oo
. .*= + ..
. .o+...=.
. .oo++x=
+----[SHA256]----+
student@ubuntu16srvr:~$ sudo ssh student@192.168.88.211
[sudo] password for student:
The authenticity of host '192.168.88.211' (192.168.88.211) can't be established.
ECDSA key fingerprint is SHA256:/z3Q8P7qBEewbHT2topjj/Vgfz9w4mT4idf+gx+jK7E.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.88.211' (ECDSA) to the list of known hosts.
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

192.168.88.214 (student)

Terminal Sessions View X server Tools Games Settings Macros Help

Quick connect...

```
student@ubuntu16srvr:~$ sudo ssh student@192.168.88.211
[sudo] password for student:
The authenticity of host '192.168.88.211' (192.168.88.211) can't be established.
ECDSA key fingerprint is SHA256:/z3Q8P7qBEewbHT2topjj/Vgfz9w4mT4idf+gx+jK7E.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.88.211' (ECDSA) to the list of known hosts.
student@192.168.88.211's password:
Welcome to Ubuntu 16.04.5 LTS (GNU/Linux 4.4.0-131-generic x86_64)

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

186 packages can be updated.
117 updates are security updates.

New release '18.04.3 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Wed Nov 20 00:59:22 2019 from 192.168.88.254
student@ubuntu16srvr:~$ exit
logout
Connection to 192.168.88.211 closed.
student@ubuntu16srvr:~$ ssh-copy-id student@192.168.88.211
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/student/.ssh/id_rsa.pub"
The authenticity of host '192.168.88.211' (192.168.88.211) can't be established.
ECDSA key fingerprint is SHA256:/z3Q8P7qBEewbHT2topjj/Vgfz9w4mT4idf+gx+jK7E.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
student@192.168.88.211's password:

Number of key(s) added: 1

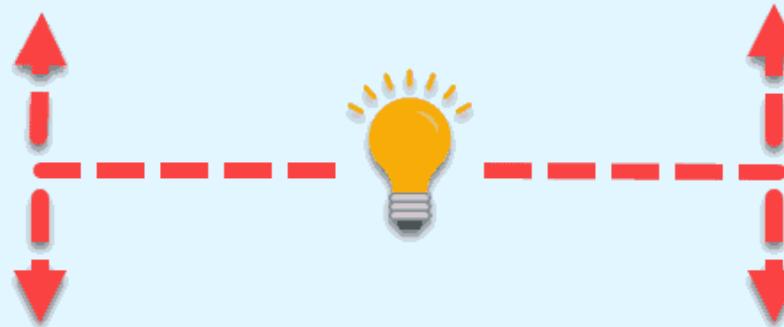
Now try logging into the machine, with: "ssh 'student@192.168.88.211'"
and check to make sure that only the key(s) you wanted were added.
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

Additional settings. Change the port

The port you choose must not be in conflict with another application. If you select a port that is reserved for another service, you can run into issues. To make a decision, refer to the list of TCP and UDP port numbers assigned by the Internet Assigned Numbers Authority (IANA).

Only root users can listen on ports below 1024.



Verify that the port you decide to use is not blocked.

Avoid the most common variations of port 22, such as **222**, **2222**, and **22222**.

Additional settings. Change the port

nano /etc/ssh/sshd_config

```
# Package generated configuration file
# See the sshd_config(5) manpage for details

# what ports, IPs and protocols we listen for
Port 22
# Use these options to restrict which interfaces/protocols sshd will bind to
#ListenAddress ::

#ListenAddress 0.0.0.0
Protocol 2
# HostKeys for protocol version 2  I
HostKey /etc/ssh/ssh_host_rsa_key
HostKey /etc/ssh/ssh_host_dsa_key
HostKey /etc/ssh/ssh_host_ecdsa_key
HostKey /etc/ssh/ssh_host_ed25519_key
#Privilege Separation is turned on for security
UsePrivilegeSeparation yes

# Lifetime and size of ephemeral version 1 server key
KeyRegenerationInterval 3600
ServerKeyBits 1024

# Logging
SyslogFacility AUTH
LogLevel INFO

# Authentication:
LoginGraceTime 120
PermitRootLogin yes
StrictModes yes
```

service sshd restart

Additional settings. Disable Server SSH Root Login

- Use SSH to log into the server as root.
- Use a text editor to open the main configuration file. This time, we will use the vi editor.
`vi /etc/ssh/sshd_config`
- Find the line that says “PermitRootLogin_yes” and change to PermitRootLogin_no. You may need to scroll down a few lines to find it.
- It is important to add the user account you will use to log in. Just add another line with the username in question: `AllowUsers your_username_here`
- Save the changes you made and then exit the text editor.
- Restart the SSH service but do not close the root session yet. For Ubuntu/Debian use `sudo service ssh restart` and for Fedora/CentOS use the `service ssh restart` command.
- Open a new terminal window and verify that you can now log in as the user you added. Once you confirm it works, exit the active root session.

Additional settings. Disable Password-Based Logins on Your Server

- Use SSH keys to log into the server as root or with sudo privileges.
- Use a text editor to open the sshd_config file. We will use vi:
`vi /etc/ssh/sshd_config`
- Look for the line that says PasswordAuthentication and change to PasswordAuthentication_no.

Make sure to uncomment the line if the # is present.

- Save the changes you've made and then exit the text editor.
- Restart the SSH service to apply the changes. For Ubuntu/Debian use sudo service ssh restart and for Fedora/CentOS use the service ssh restart command

Additional settings. Restrict SSH Access Using iptables

- Iptables is a Linux utility used for configuring firewall rules and monitoring/filtering incoming and outgoing traffic to your server. It is included by default with most Linux distributions.
- With iptables, you can define rules that limit or permit traffic for different kinds of services by IP address, port or network protocol and thus substantially improve the security of your server. In our case, we will set firewall rules to restrict the incoming SSH traffic for everyone but one IP address or subnet.
- This way, blocking port 22 will not only stop unauthorized access to your servers but can also stop or prevent DDoS attacks.
- While taking this step, you should make sure you do not lock yourself out by completely blocking SSH traffic. You will need to use only a few commands to allow a specific IP address or subnet for incoming SSH connections.

QUESTIONS & ANSWERS



THANK YOU!