

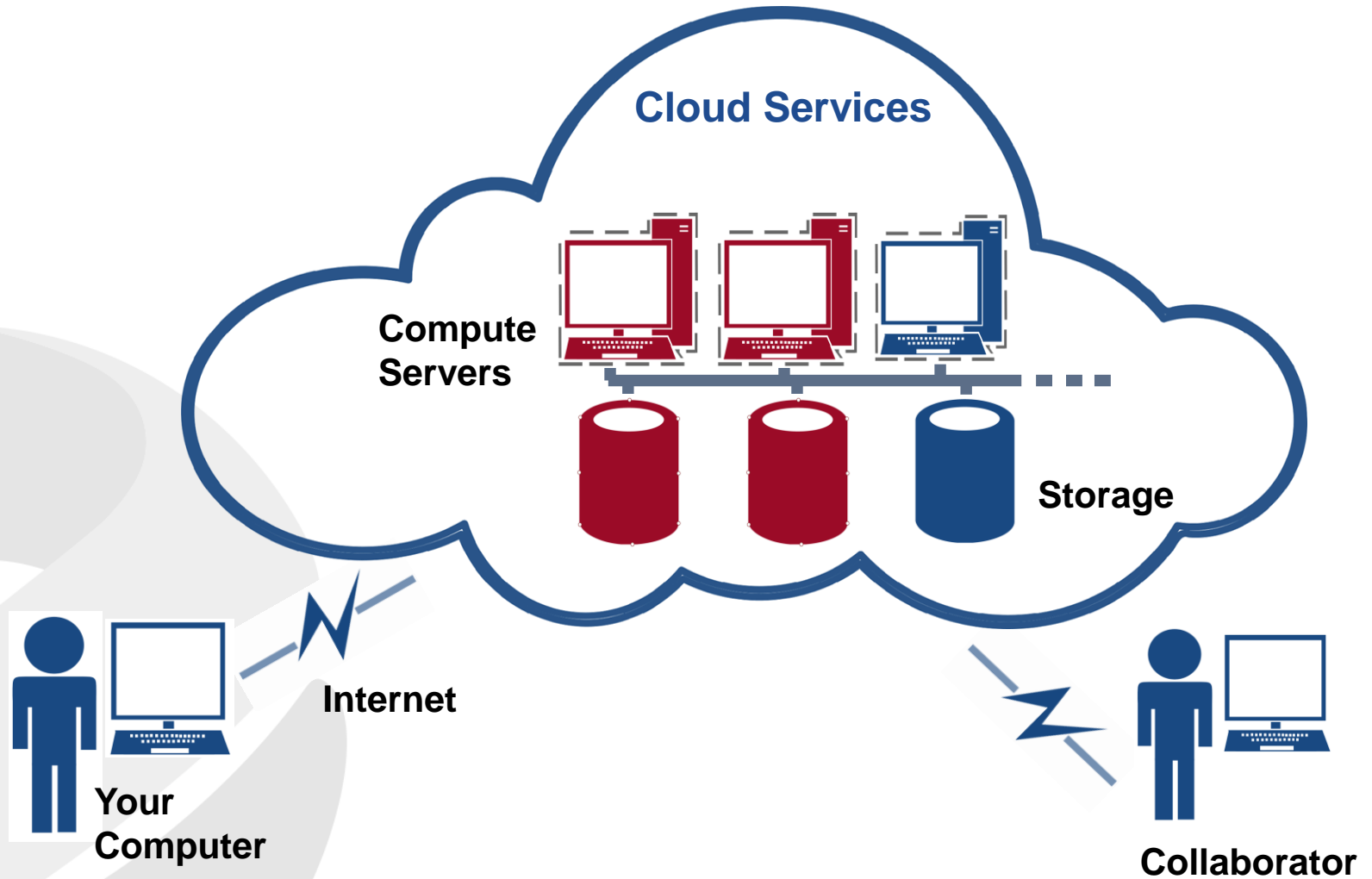
# WORKSHOP:

## Running a Virtual Machine in the Cloud

Joey Gerlach  
25 February 2016

WORKSHOP: Running a Virtual Machine in the Cloud

# WHAT IS CLOUD COMPUTING?



# Let's brainstorm use case scenarios for the Australian research cloud....

- As an alternative to using your desktop computer
- As an alternative to setting up a server
- As an alternative to using a supercomputer

## When to use the Cloud

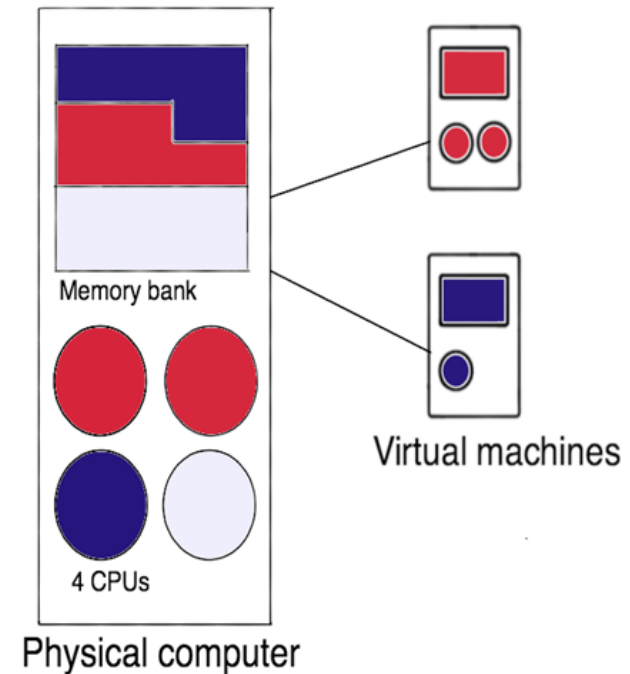
- You want **instant availability** of large-scale computing resources.
- Possibility of **software choice**: design virtual machines to suit your need, incl. choice of OS.
- The simple case: you need **easy access to computing infrastructure**.

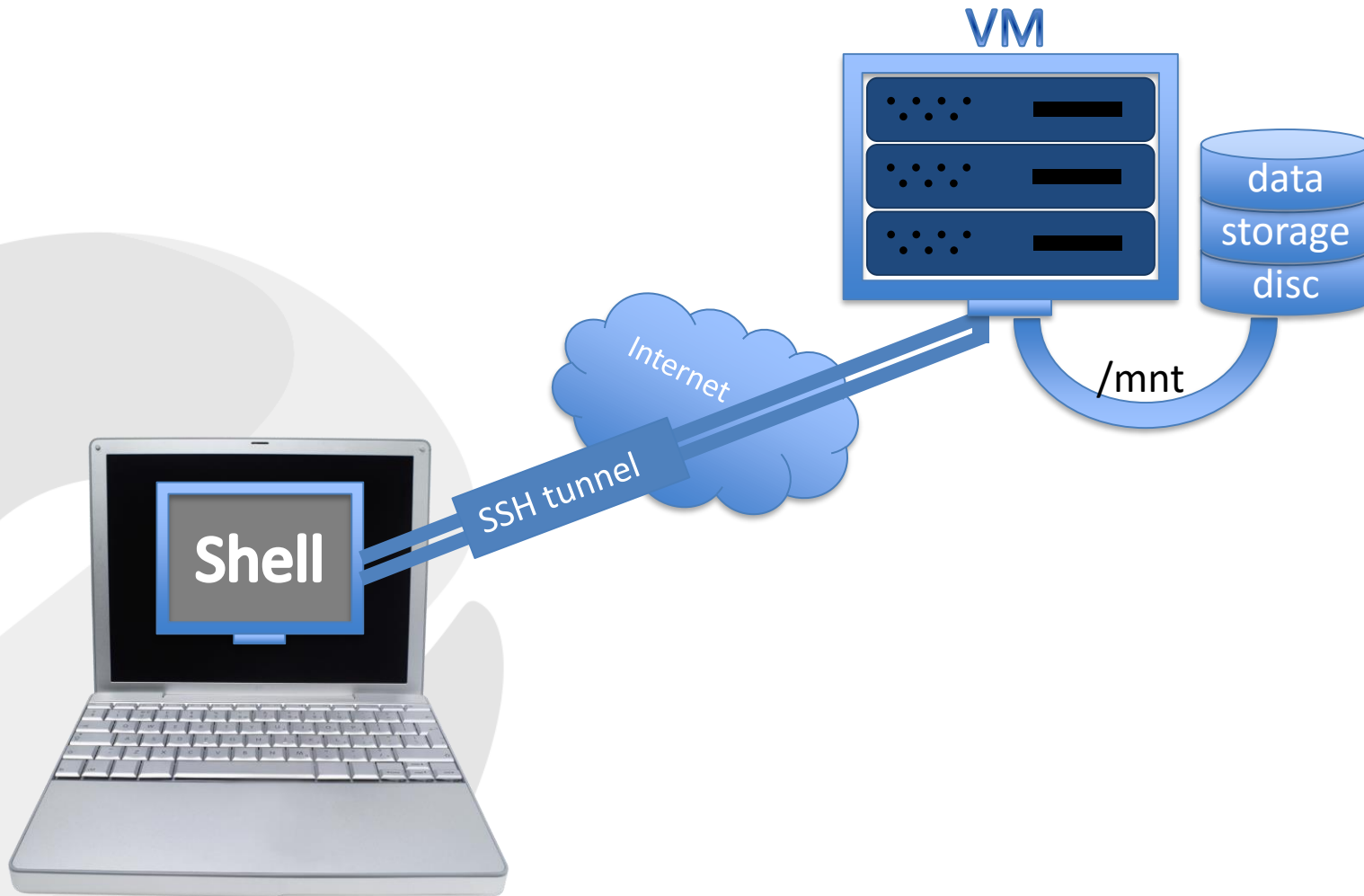
# THE STRUCTURE OF THE VIRTUAL MACHINE

- Discs
- File Structure
- Users

# Virtualization

- **Virtualization** basically means that the hardware is “simulated”.
- We can simulate a whole computer including the Operating System.
- Several virtual computers can run on one larger, more powerful computer



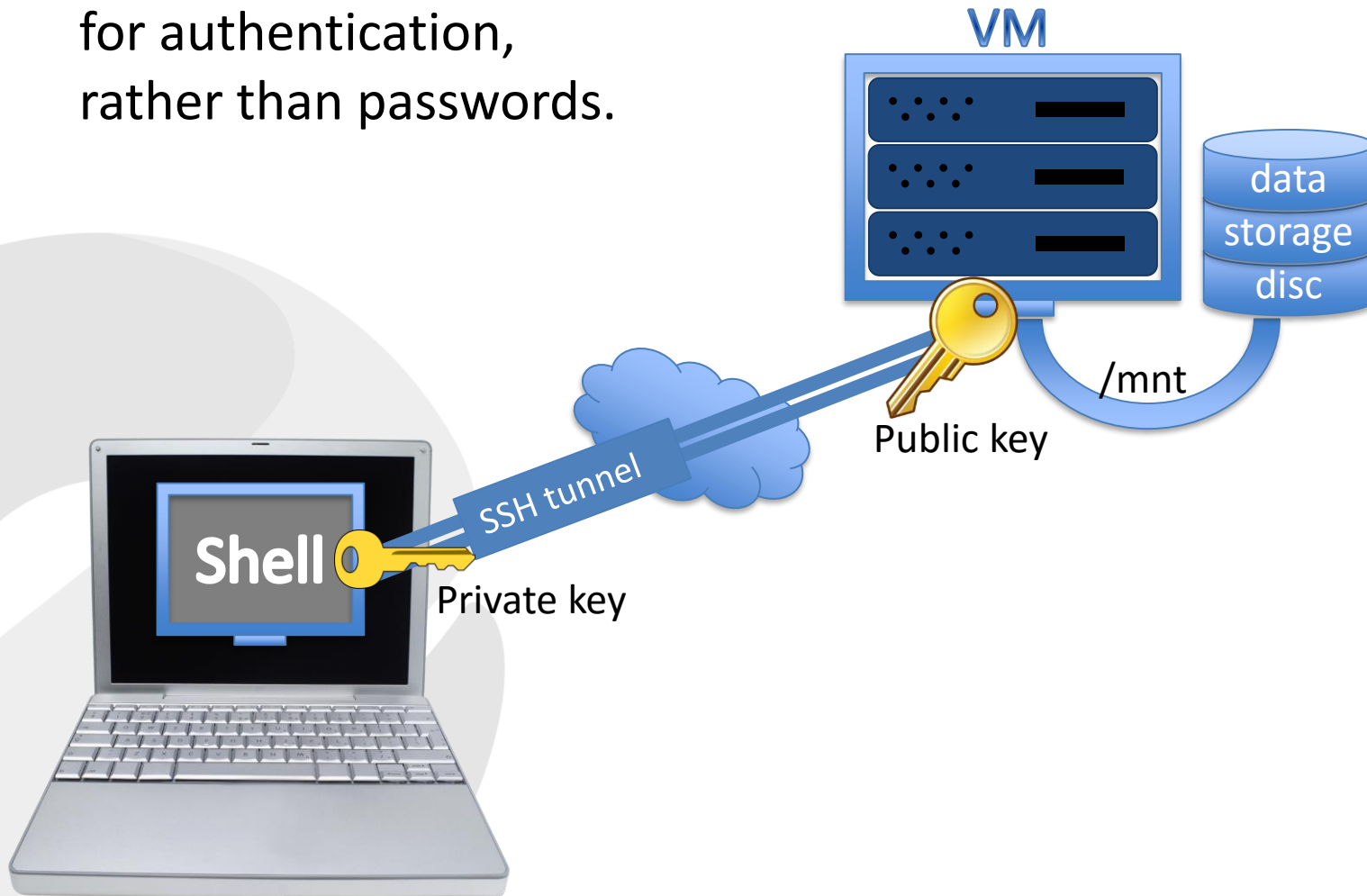


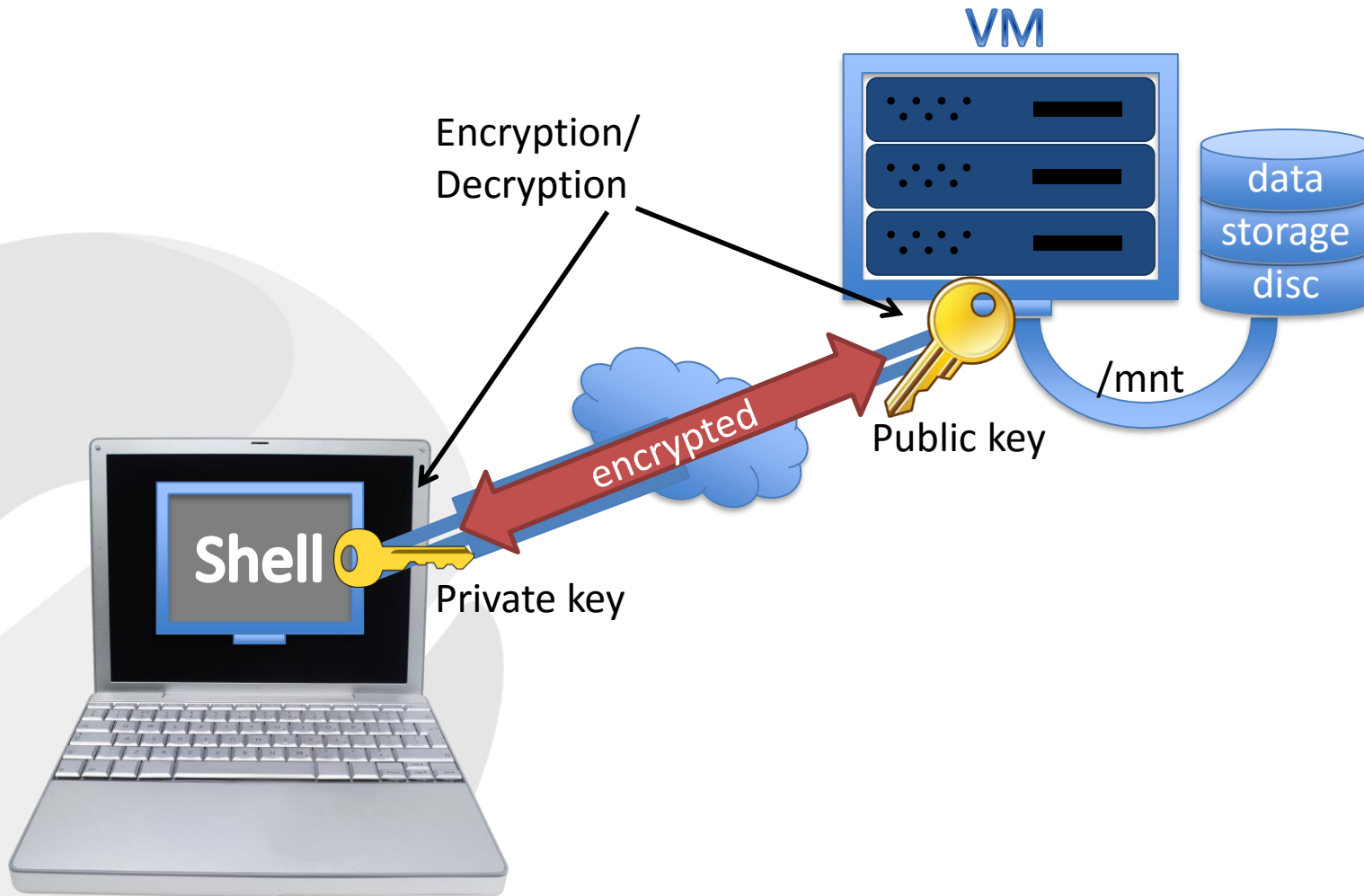


# Access and Security tab

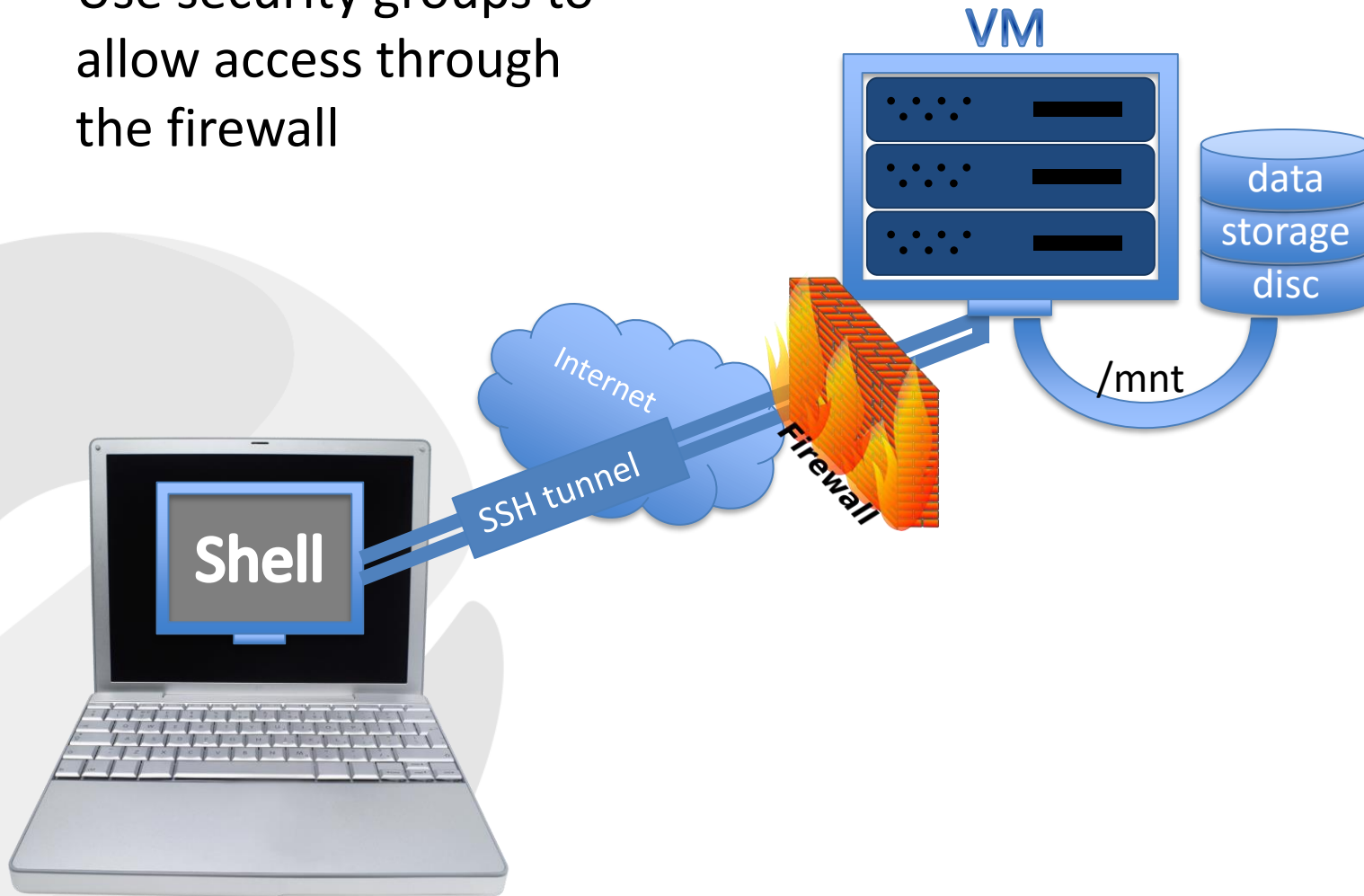
Key pairs and Security groups

Nectar uses “keypairs”  
for authentication,  
rather than passwords.





Use security groups to  
allow access through  
the firewall



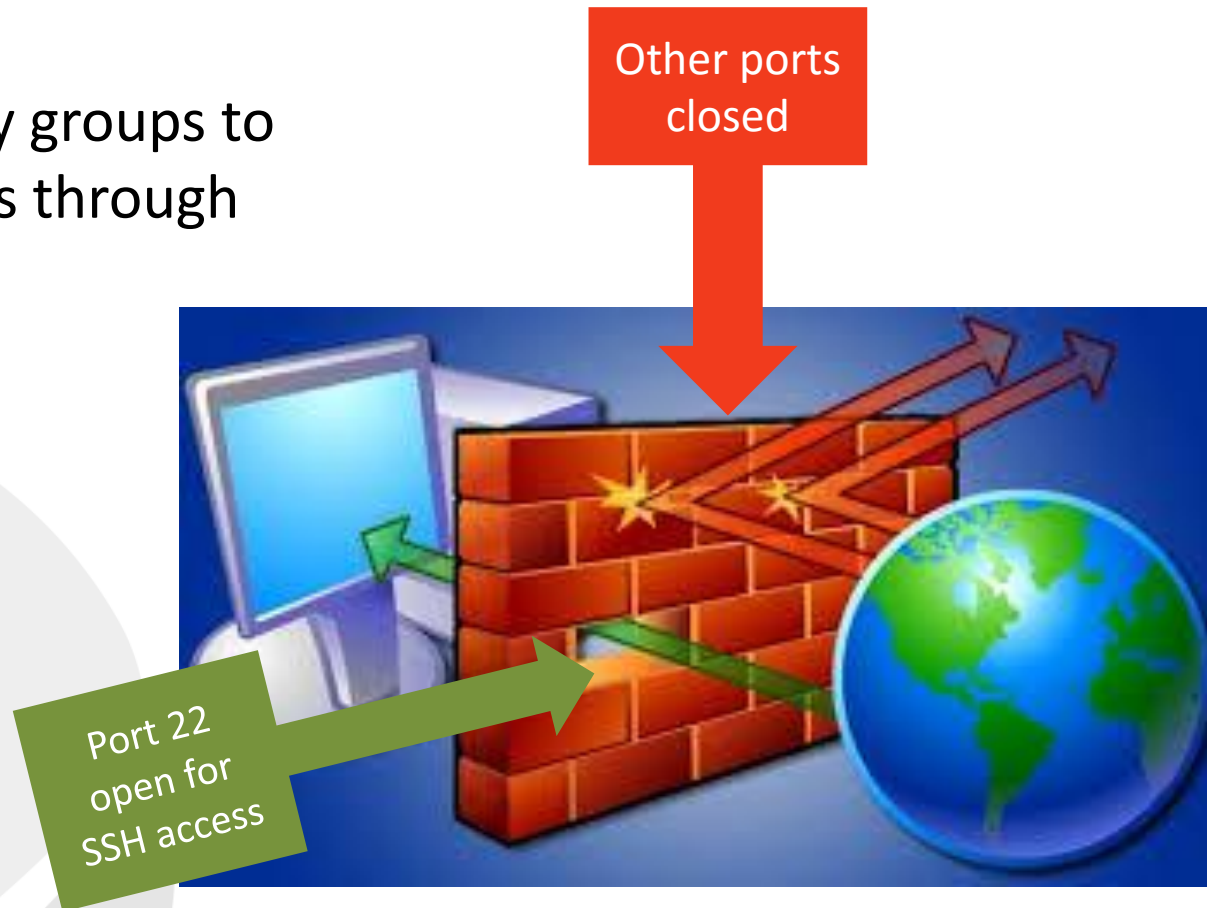
Use security groups to  
allow access through  
the firewall



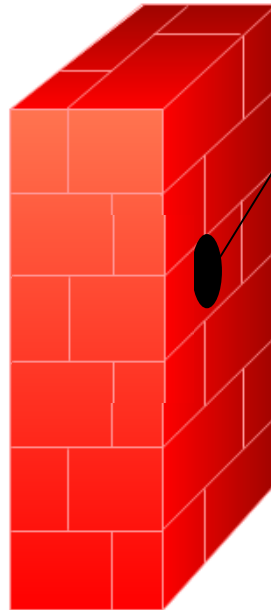
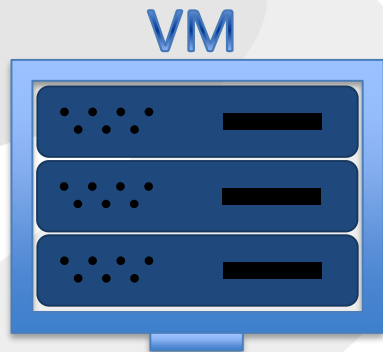
Use security groups to  
allow access through  
the firewall



Use security groups to  
allow access through  
the firewall



Access through an open port can be restricted to a range of IP addresses

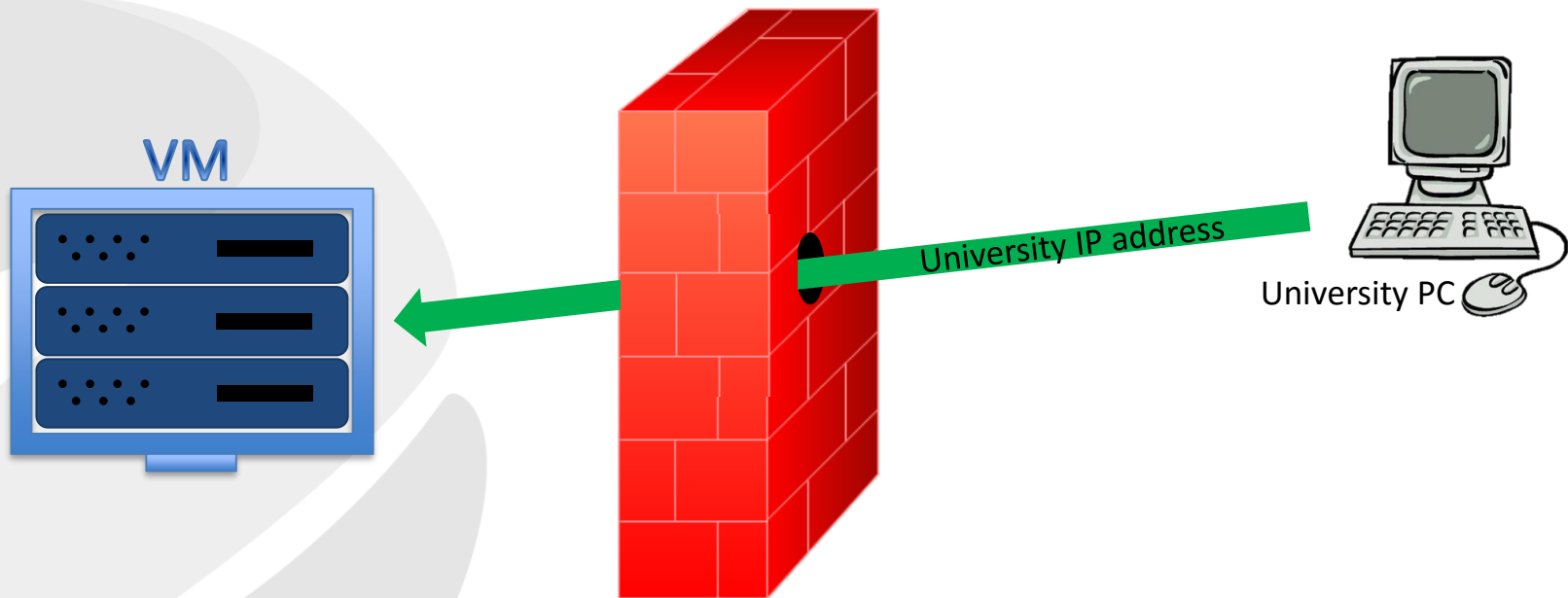


Port open for  
University IP  
addresses only

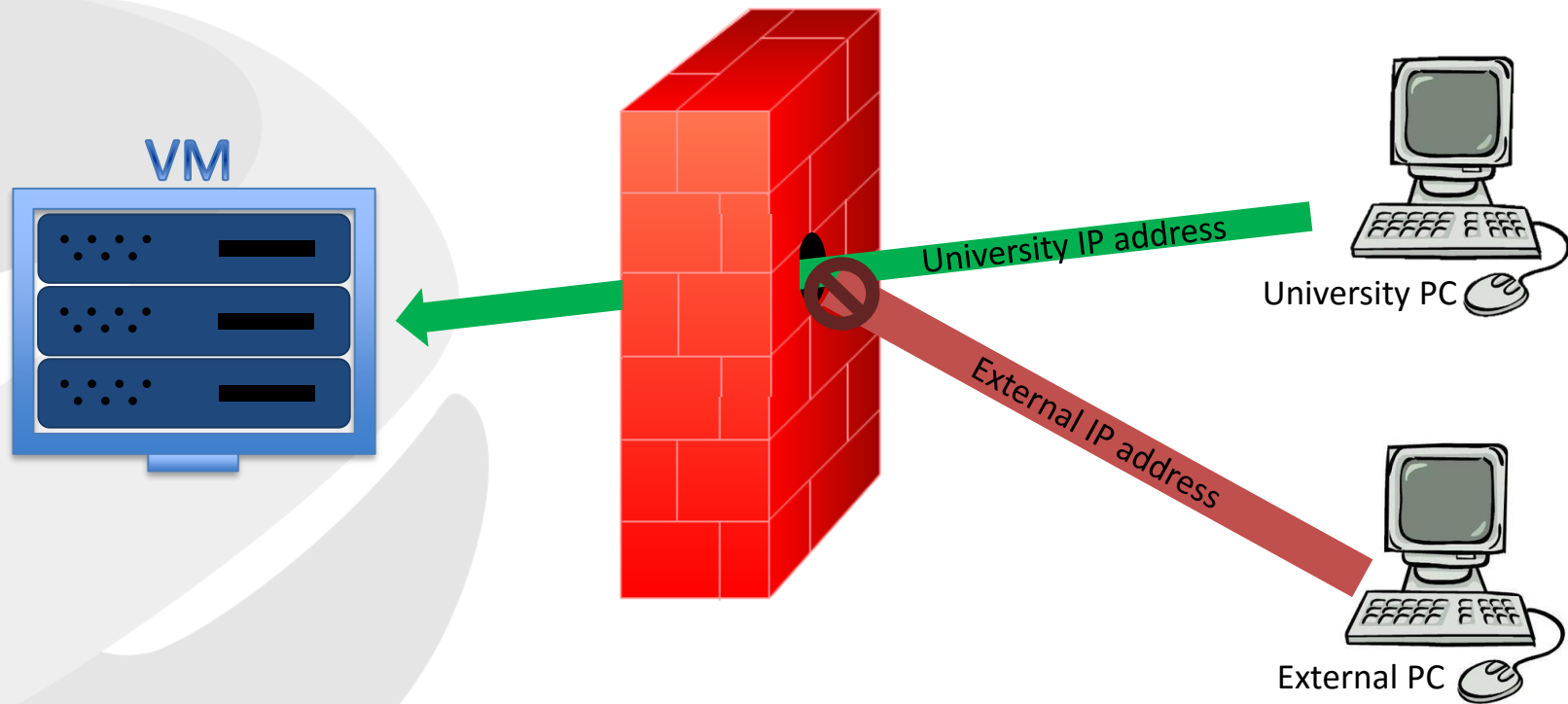




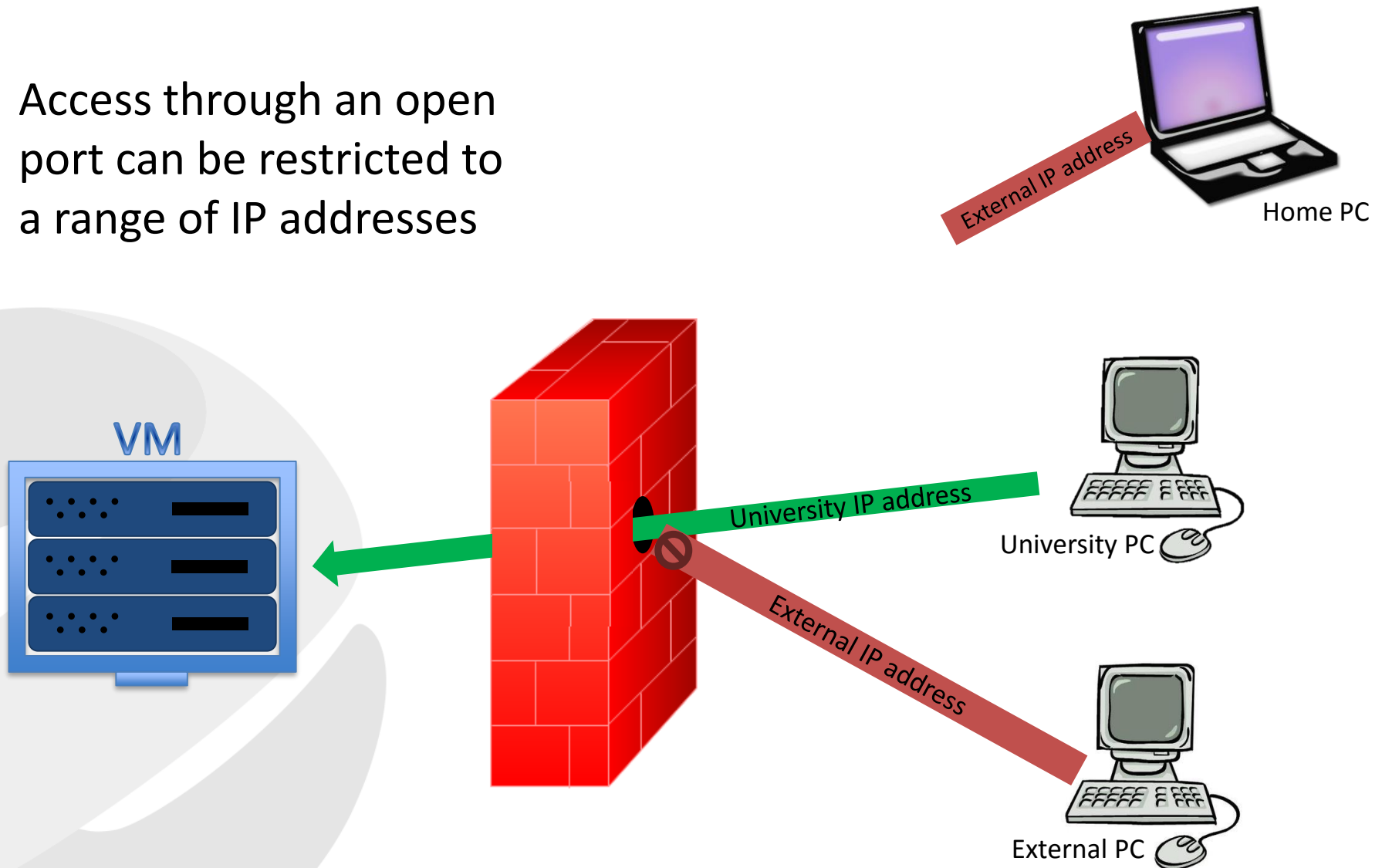
Access through an open port can be restricted to a range of IP addresses



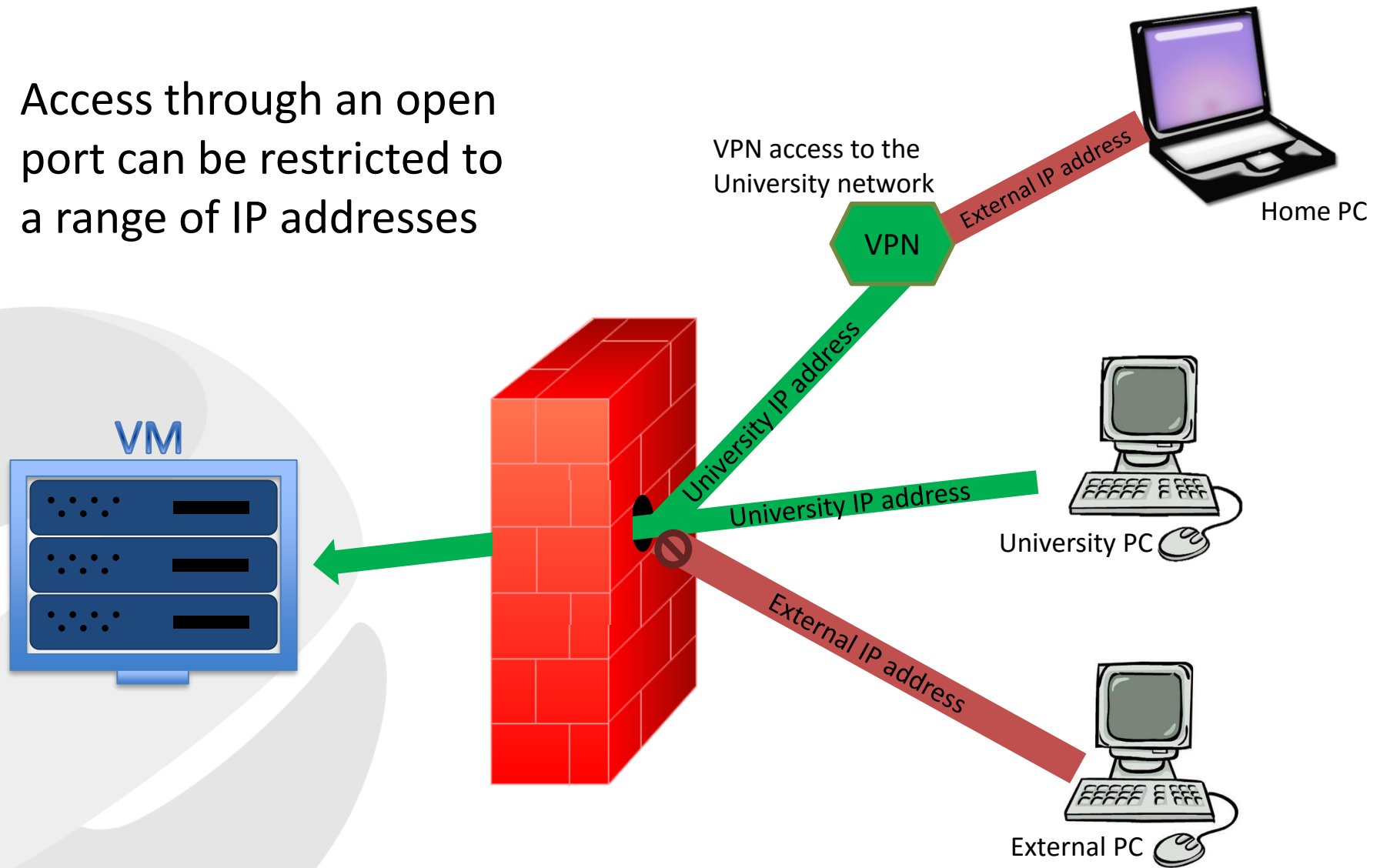
Access through an open port can be restricted to a range of IP addresses



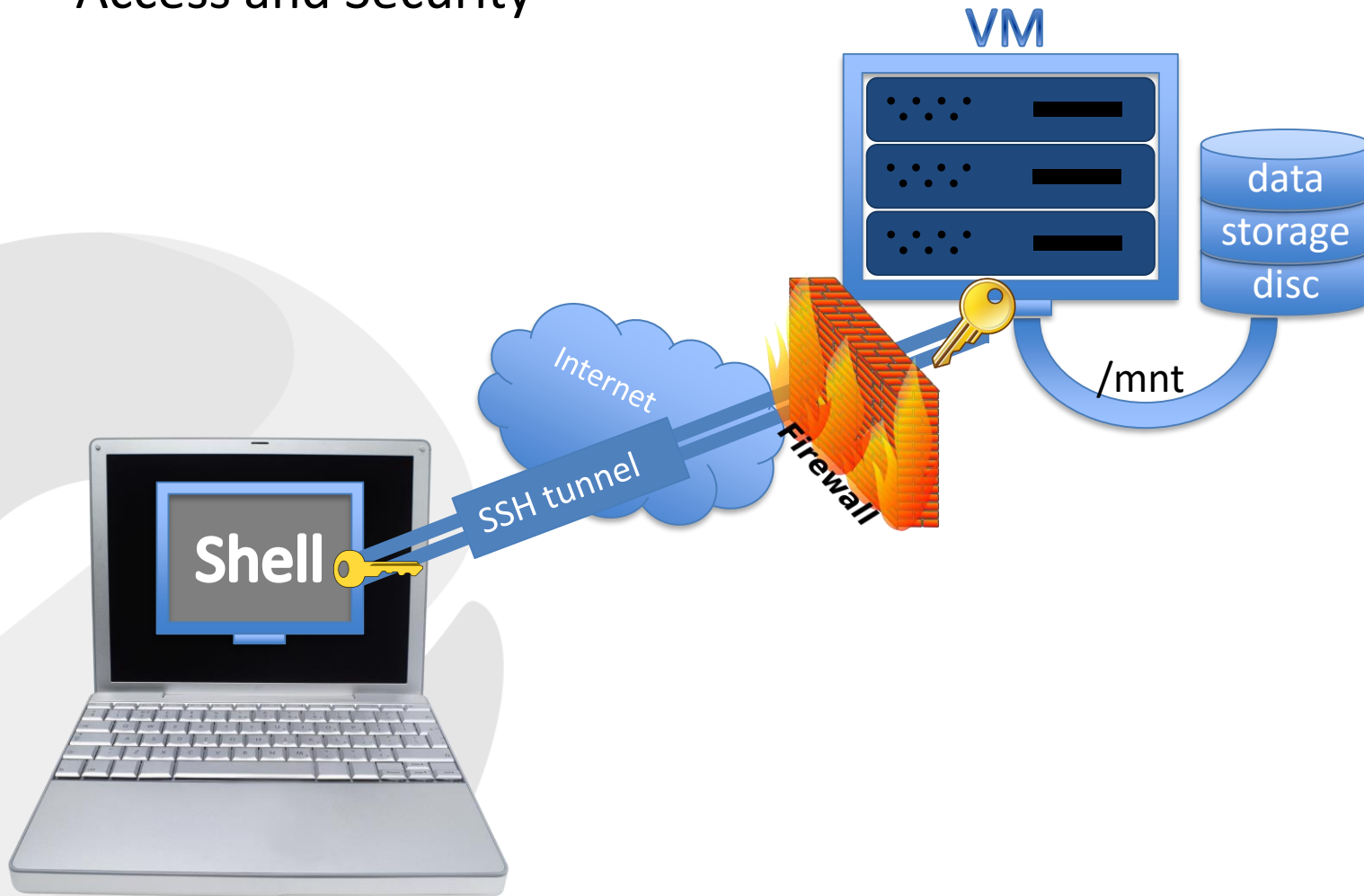
Access through an open port can be restricted to a range of IP addresses



Access through an open port can be restricted to a range of IP addresses



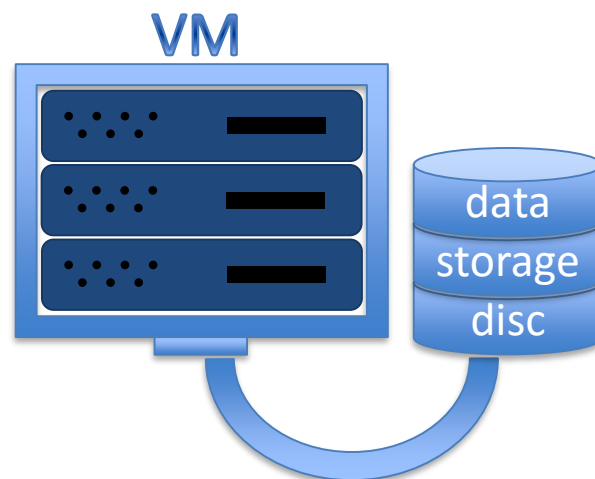
## Access and Security

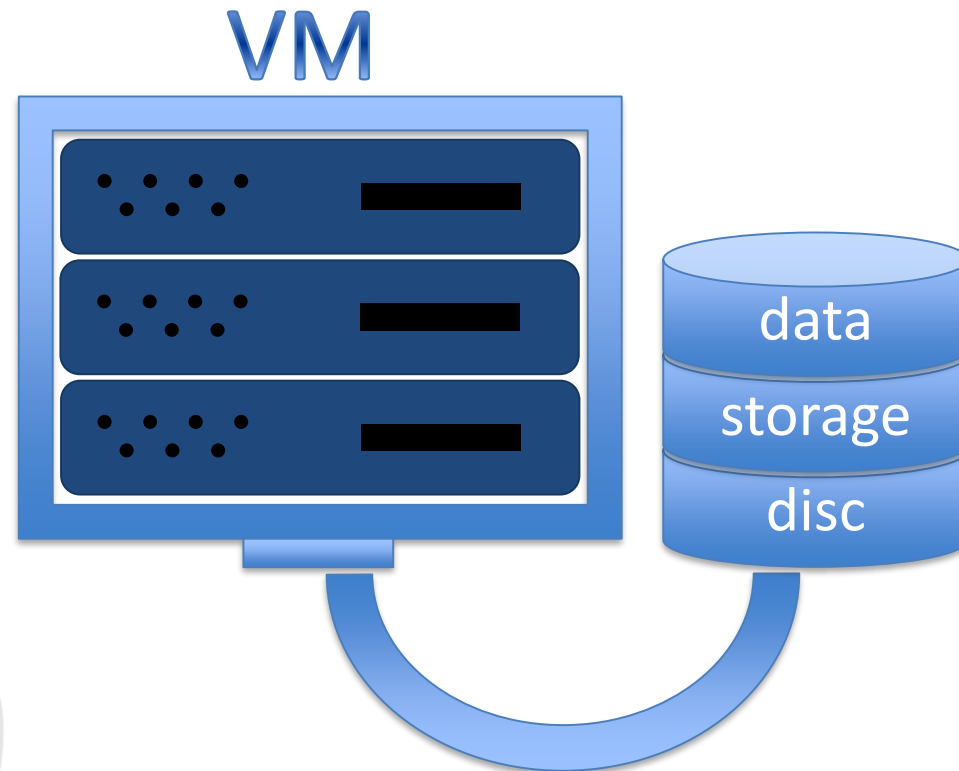


# Instances Tab

Design your Virtual Machine

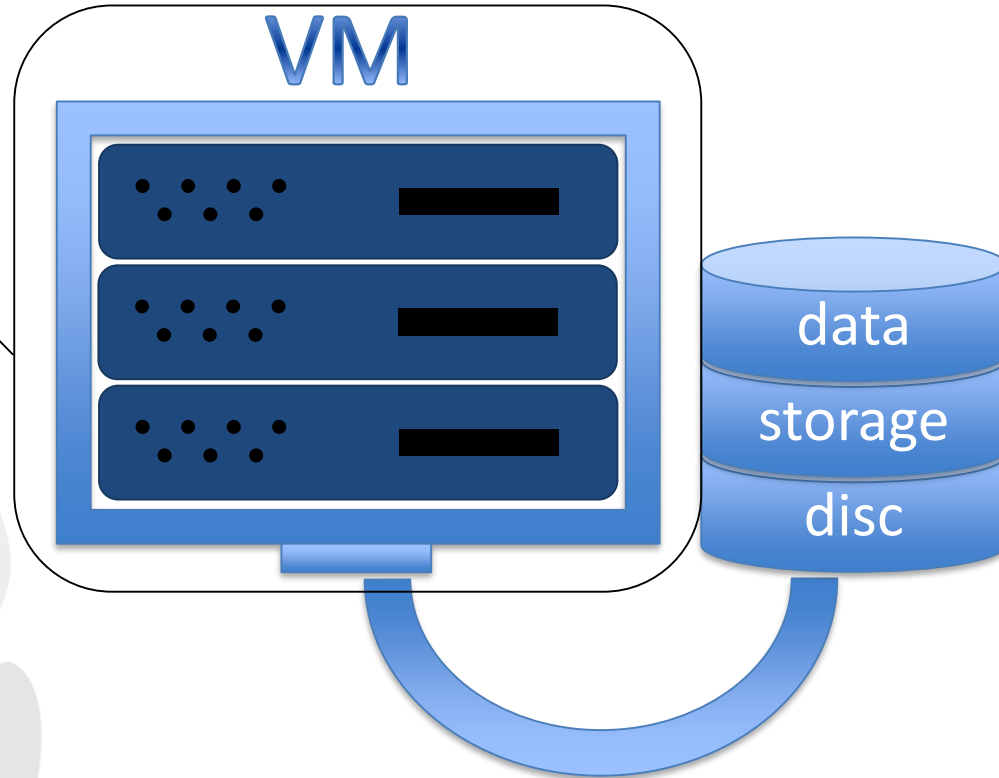
Understand the Flavors







**/dev/vda**  
Root (primary)  
disc  
5-30 GB

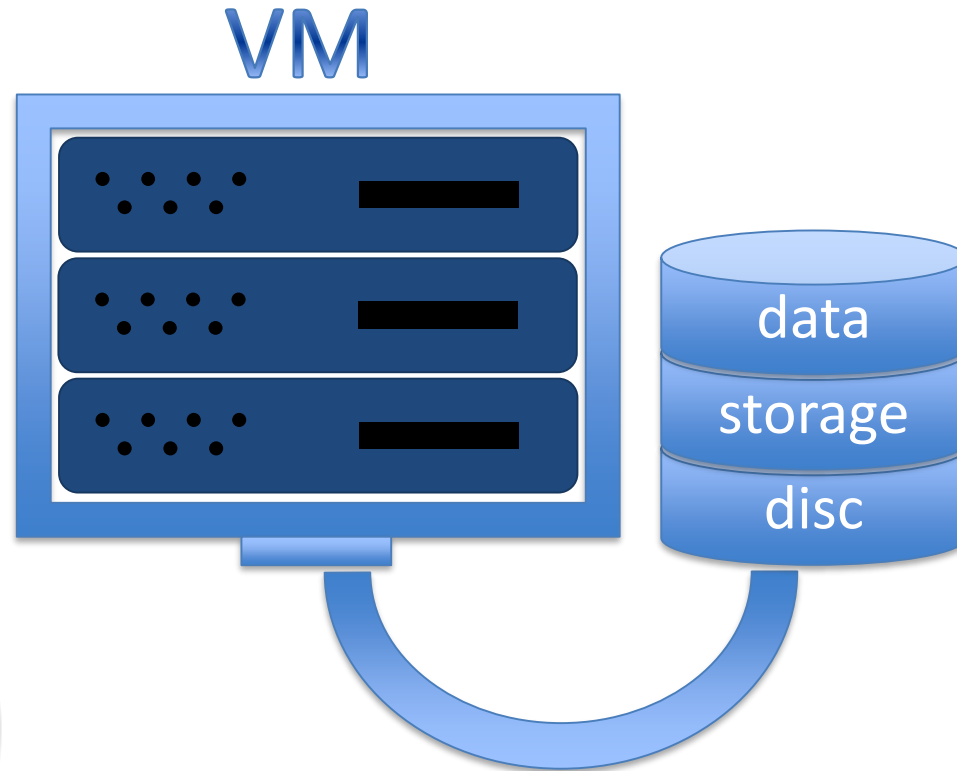


**/dev/vda**

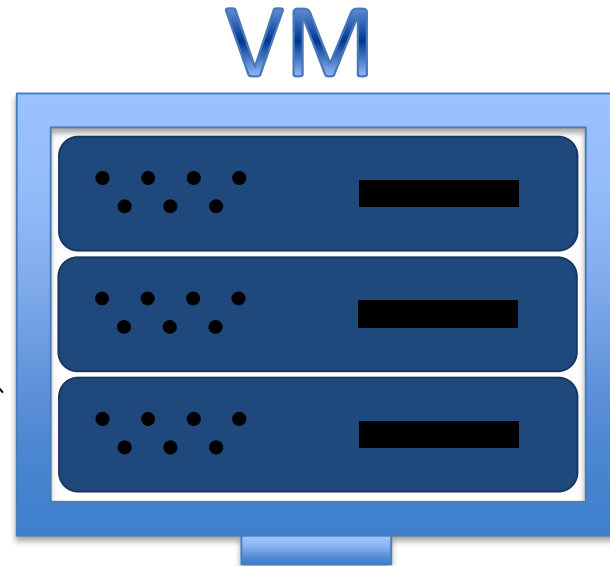
Root disc

5-30 GB

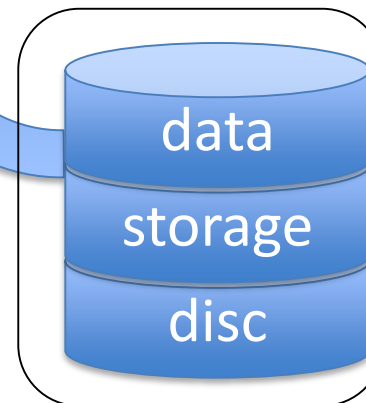
For the operating  
system and  
software  
applications



**/dev/vda**  
Root (primary)  
disc  
5-30 GB

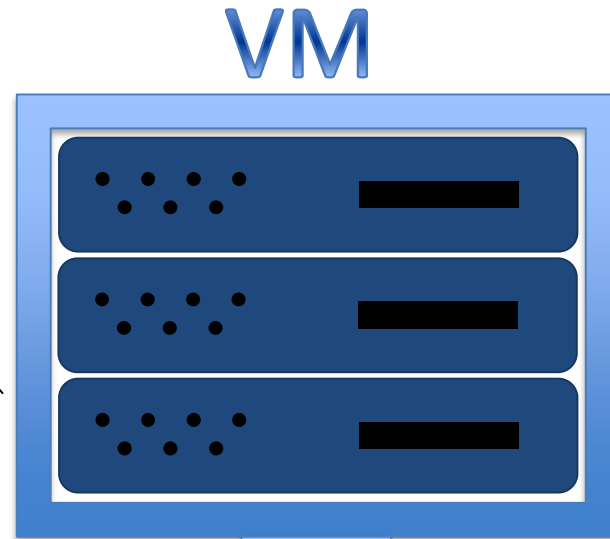


**/dev/vdb**  
Secondary  
storage  
0 – 480 GB



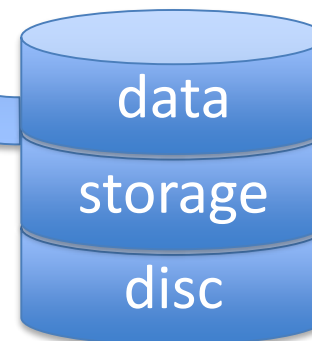
**/dev/vda**

Root (primary)  
disc  
5-30 GB



**/dev/vdb**

Secondary  
storage  
0 – 480 GB

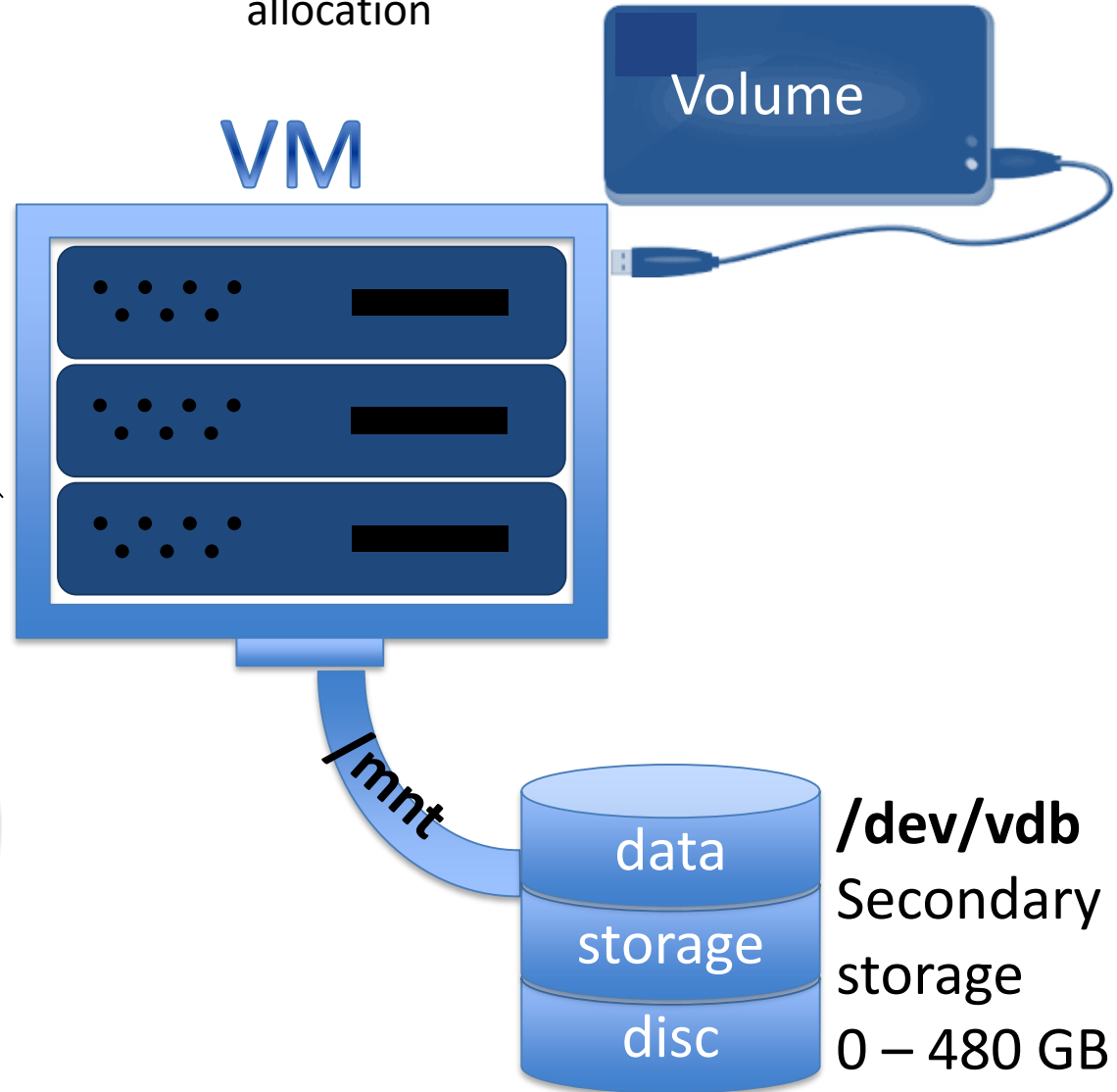


For storage of your  
data. Input and  
Output files.

# **/dev/vdc** Volume storage

- available by  
allocation

**/dev/vda**  
Root (primary)  
disc  
5-30 GB



# Folder Structure

Looking around you VM

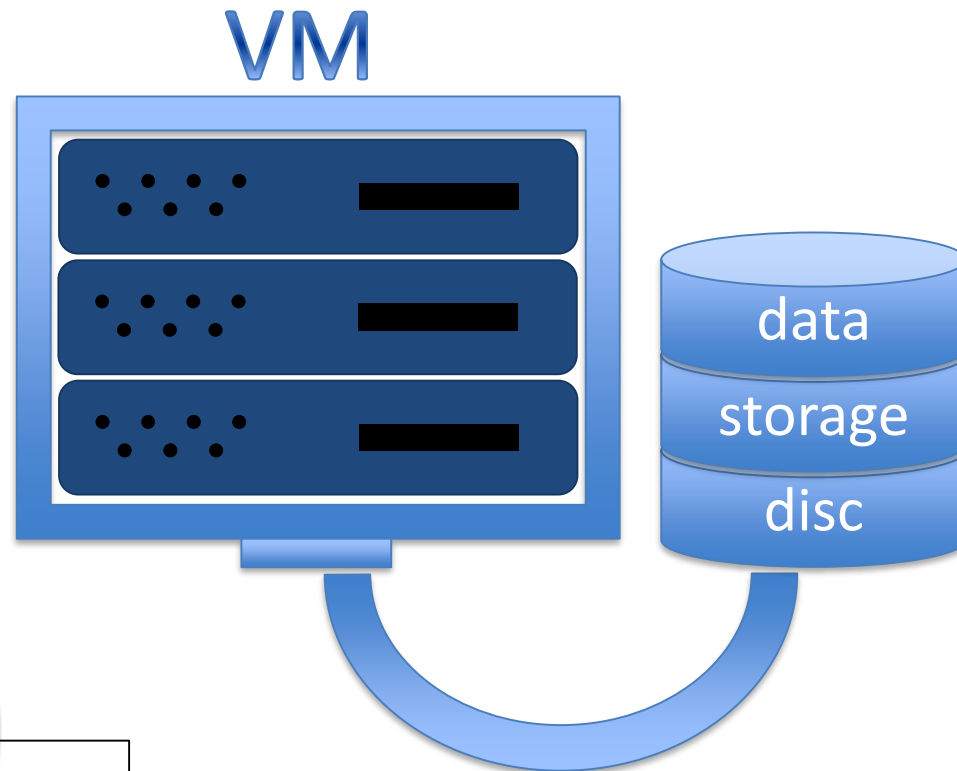
**/dev/vda**

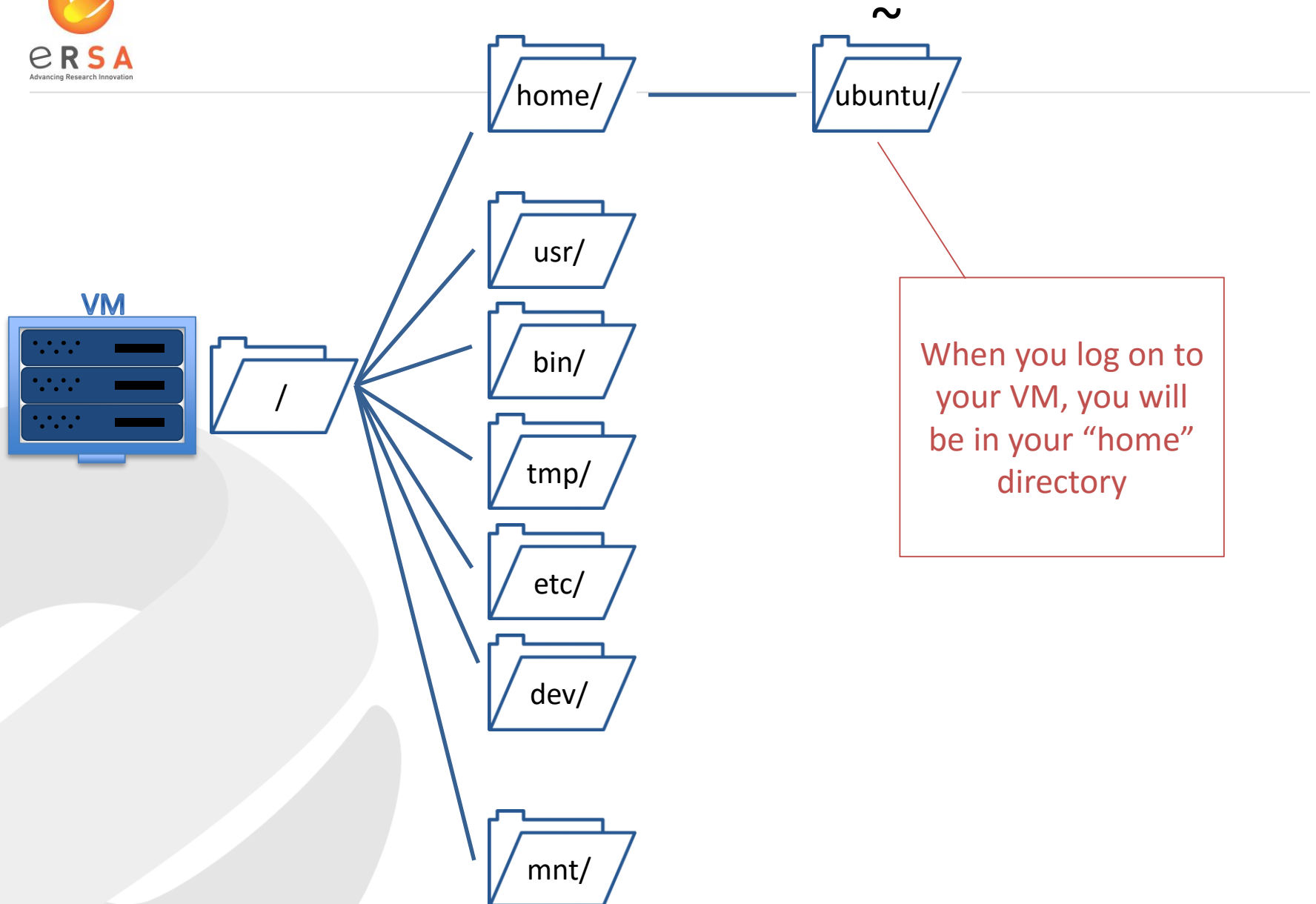
Root disc  
5-30 GB

For the operating  
system and  
software  
applications

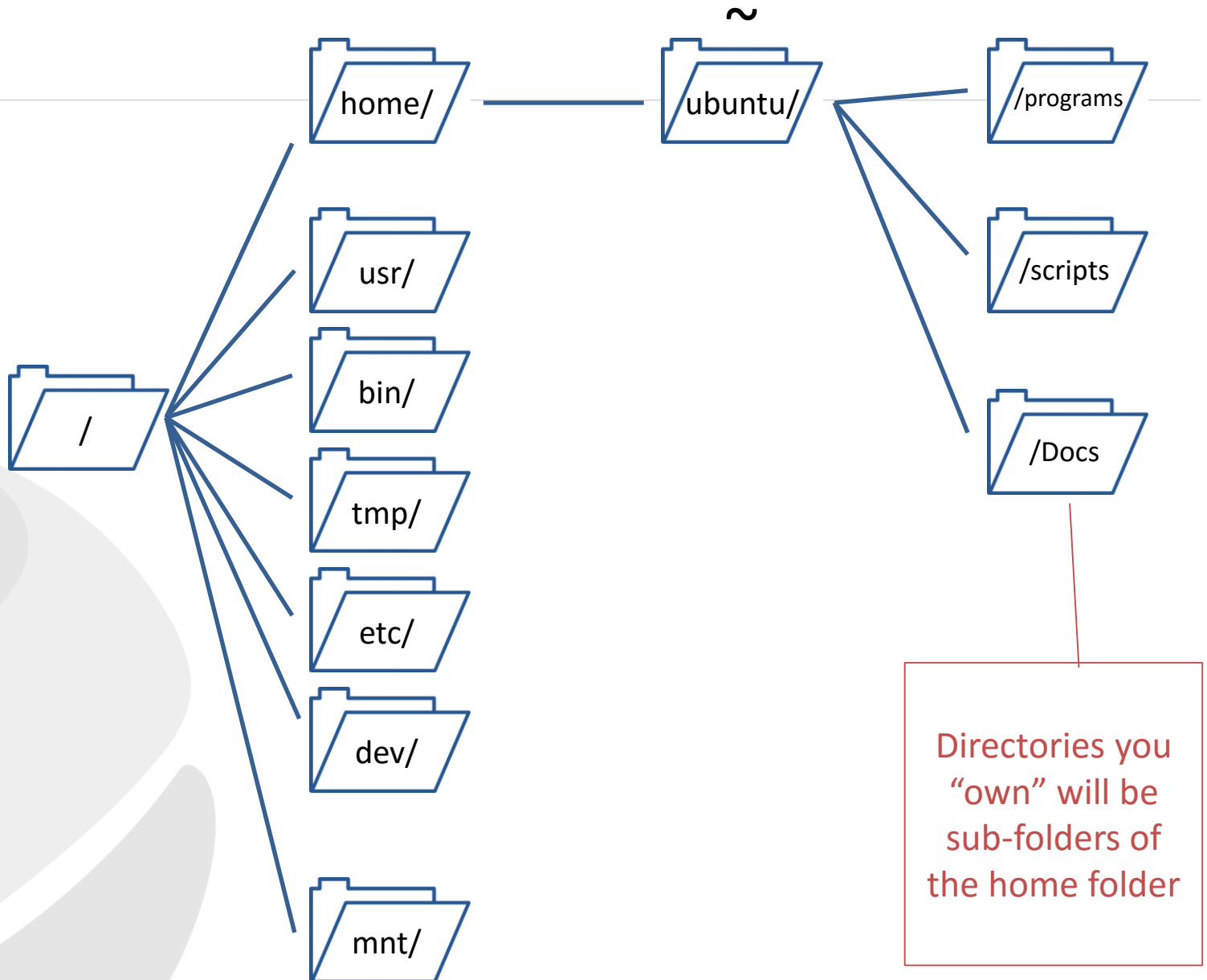
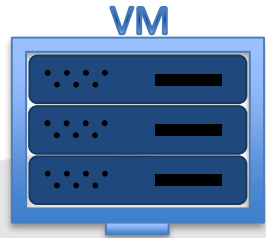
**/**

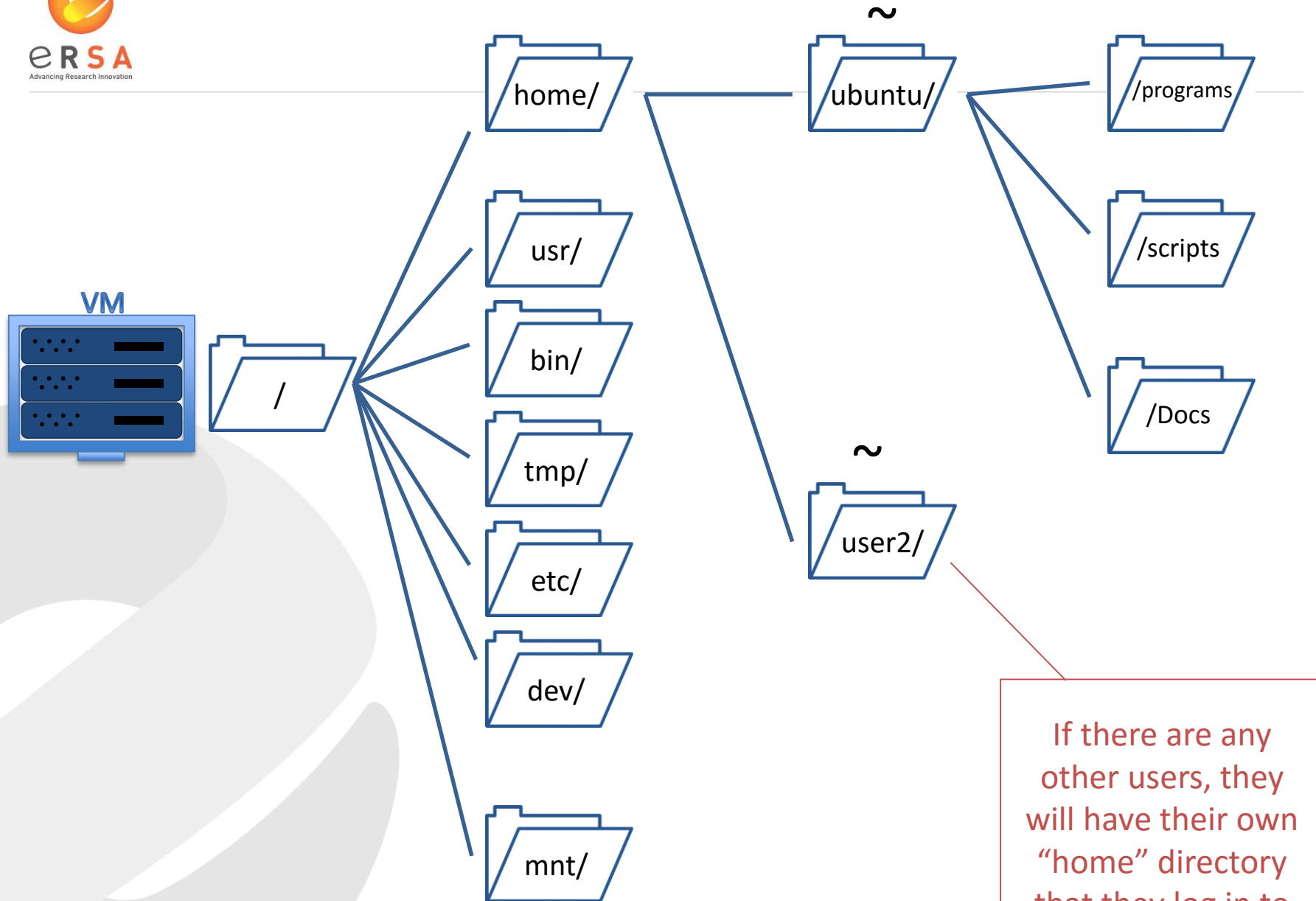
Root Directory  
Includes user directories  
(\$HOME directories)



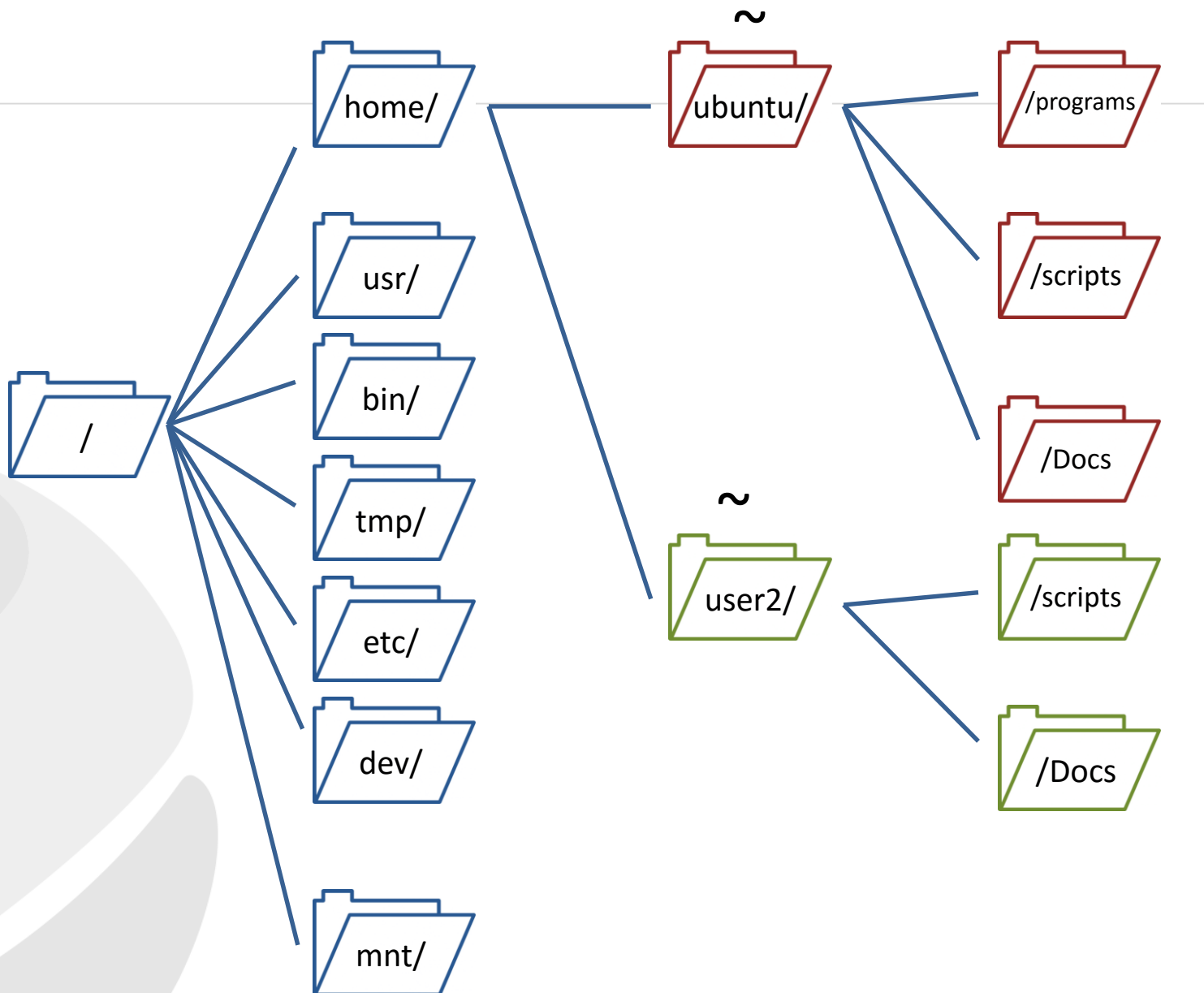
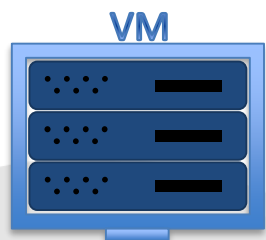








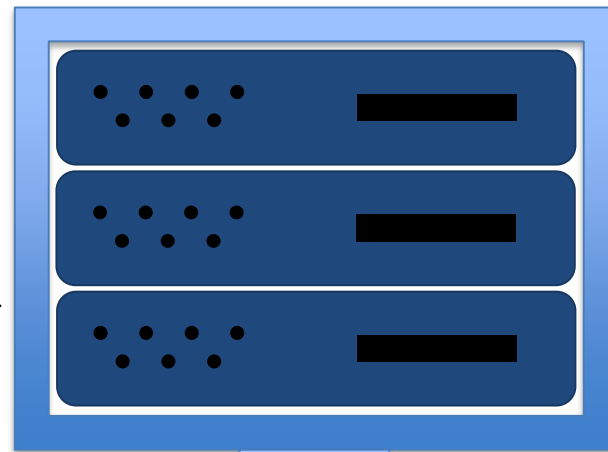
If there are any other users, they will have their own “home” directory that they log in to



**/dev/vda**

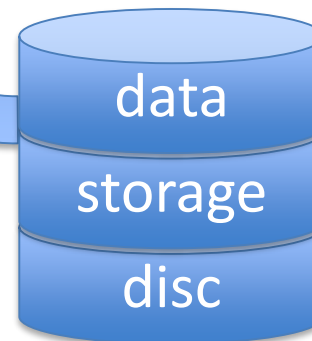
Root (primary)  
disc  
5-30 GB

VM



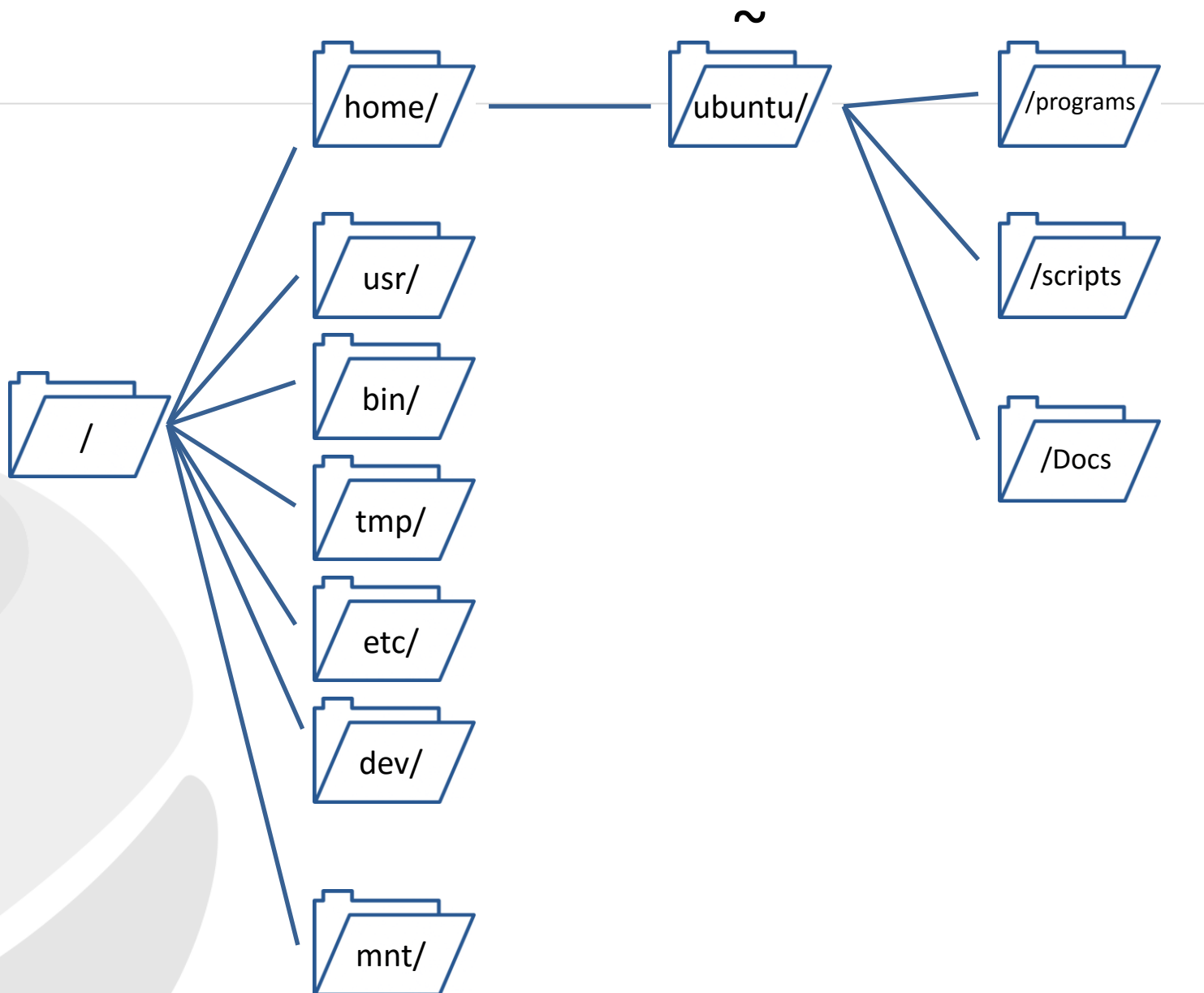
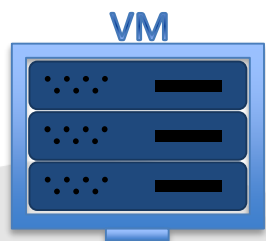
**/mnt**

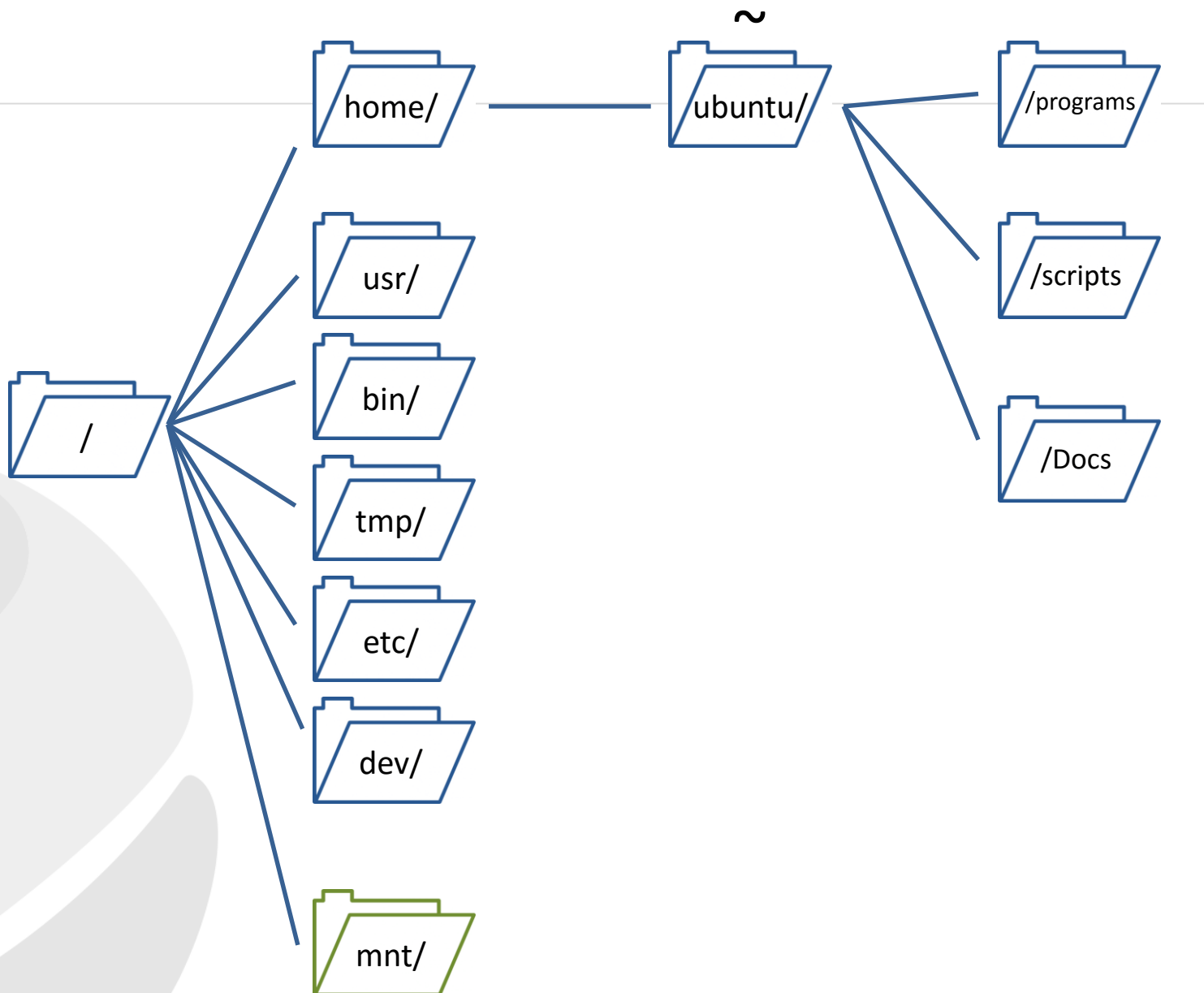
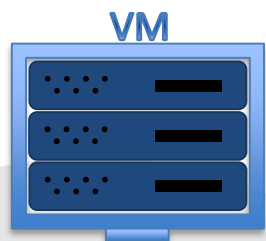
The storage disc is  
“mounted” in the  
**/mnt** directory.

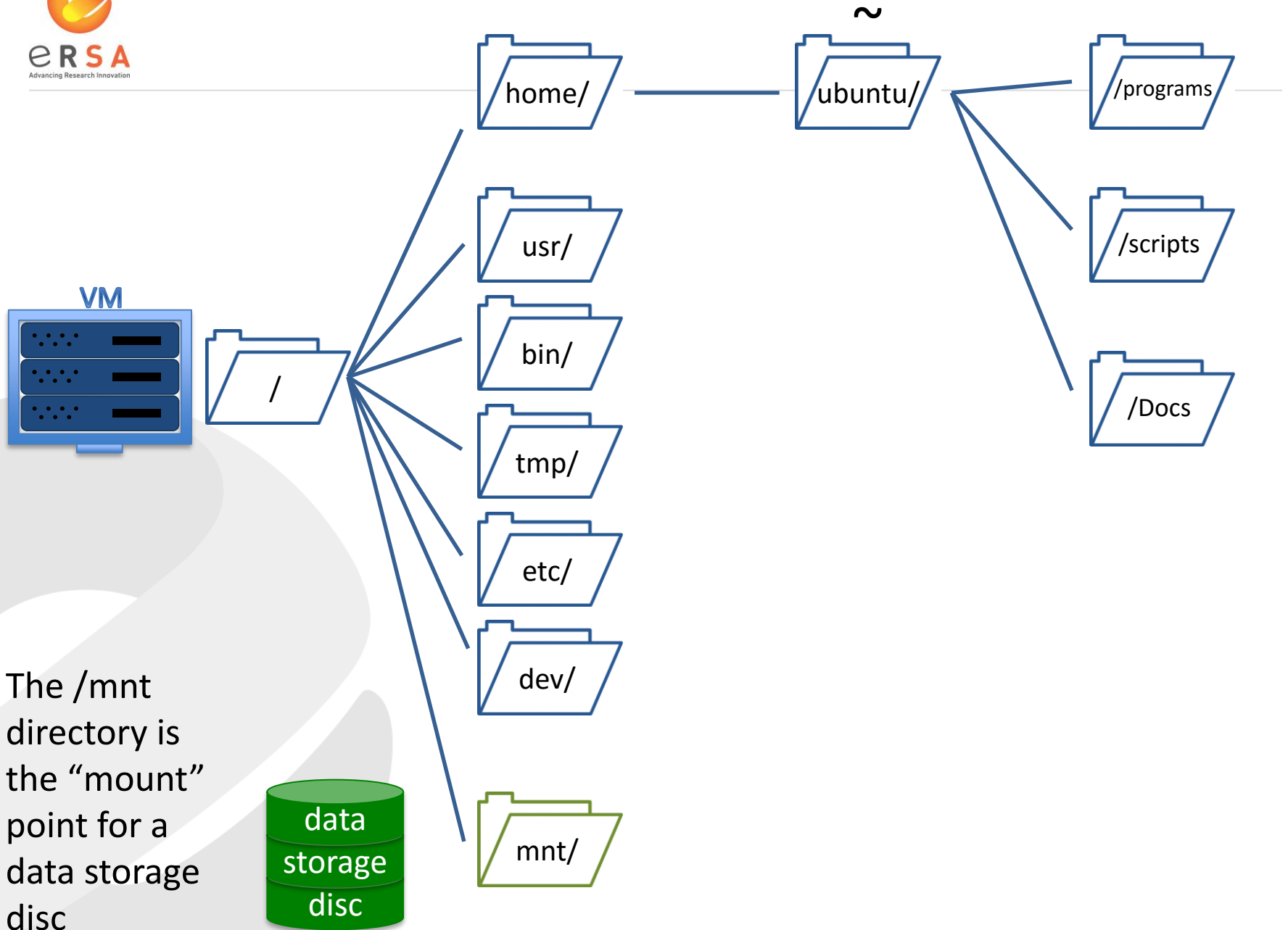


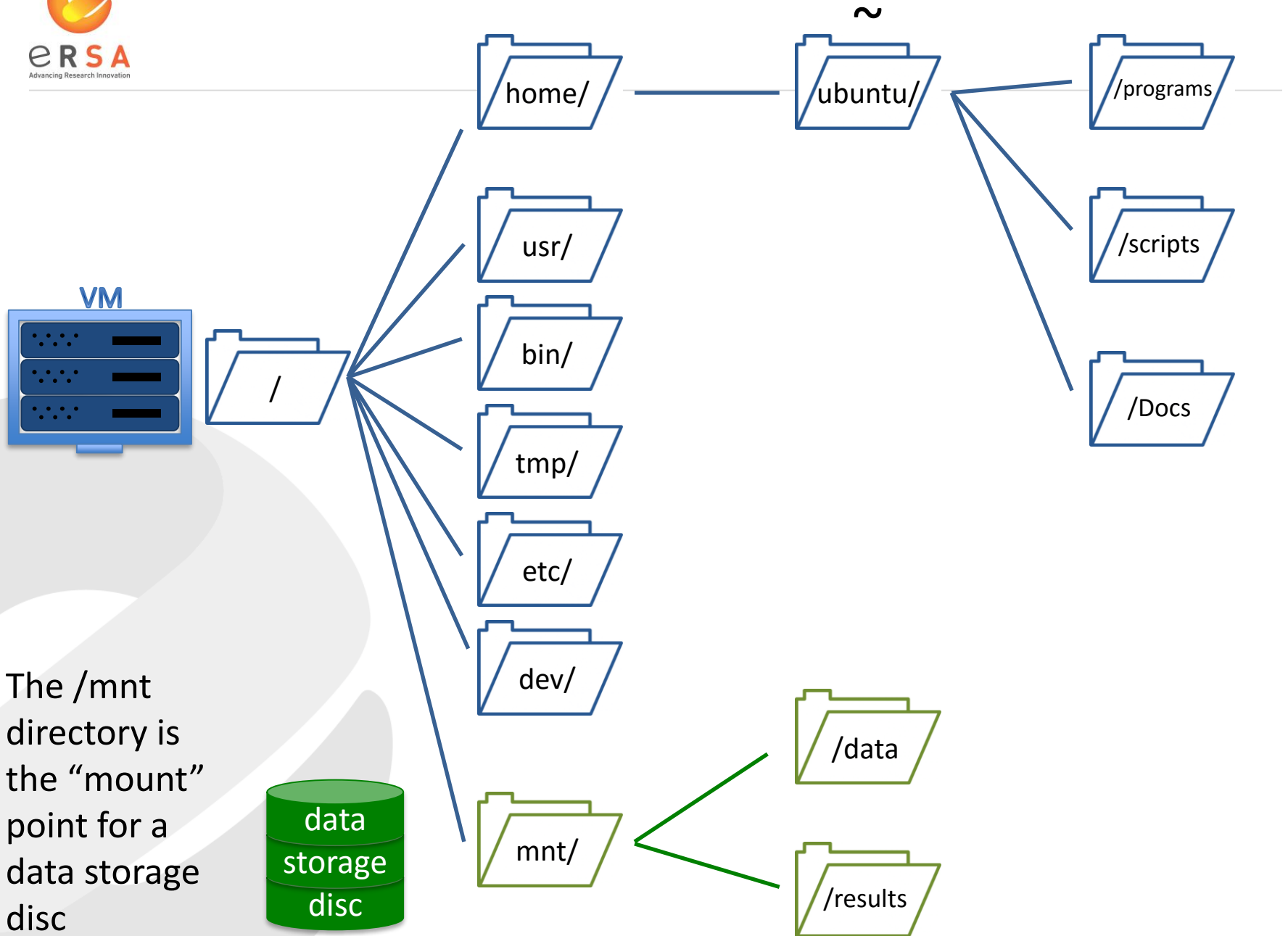
**/dev/vdb**

Secondary  
storage  
0 – 480 GB







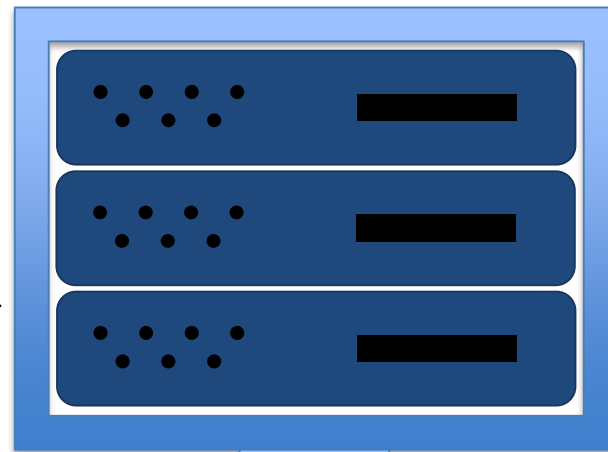




**/dev/vda**

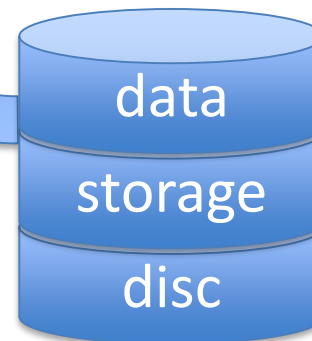
Root (primary)  
disc  
5-30 GB

VM



**/mnt**

The storage disc is  
“mounted” in the  
**/mnt** directory.



**/dev/vdb**

Secondary  
storage  
0 – 480 GB

**/dev/vdc**

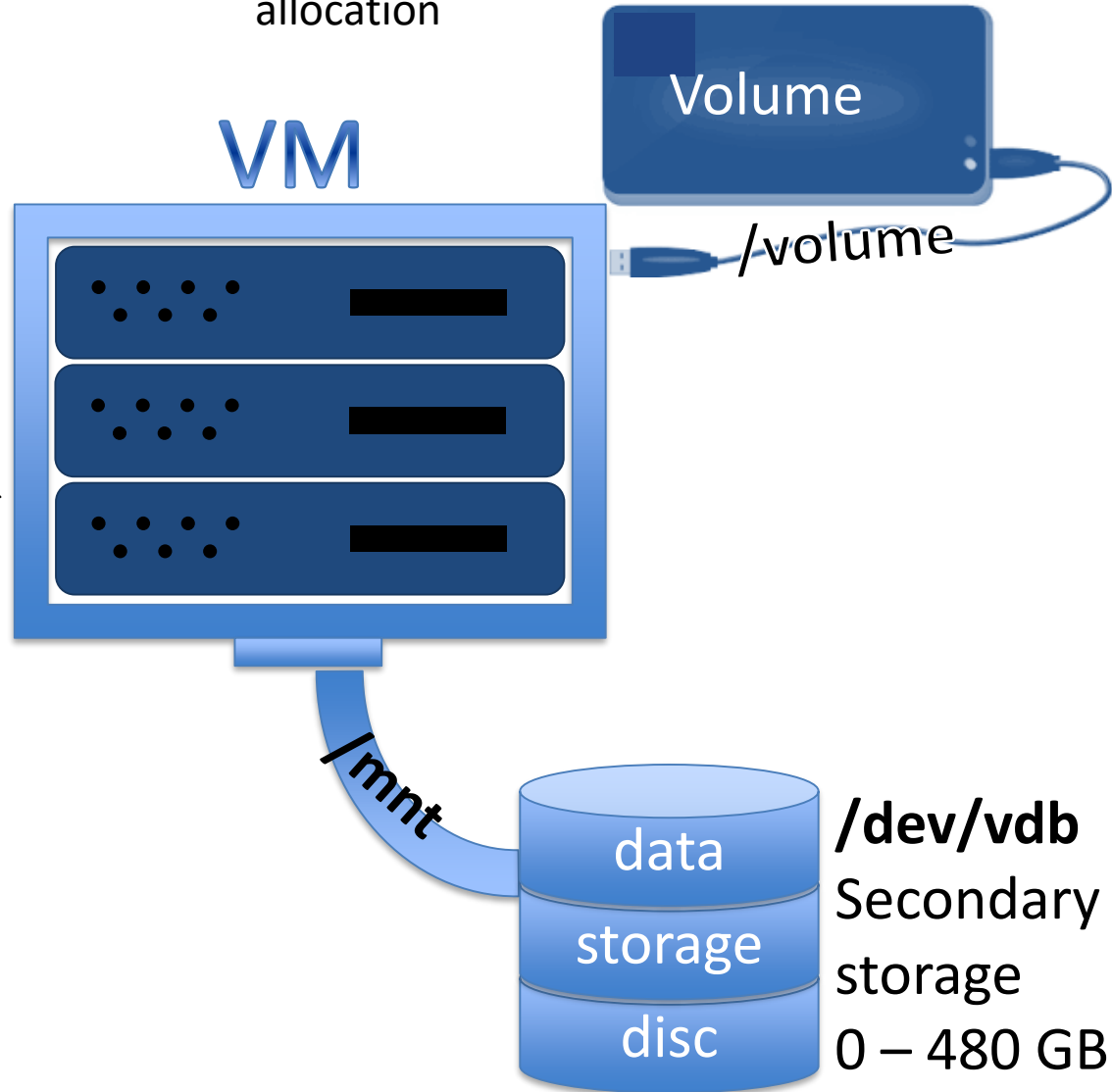
Volume storage

- available by  
allocation

Volume storage also  
must be mounted in a  
root directory

**/dev/vda**

Root (primary)  
disc  
5-30 GB



VM

Volume

/volume

/mnt

data

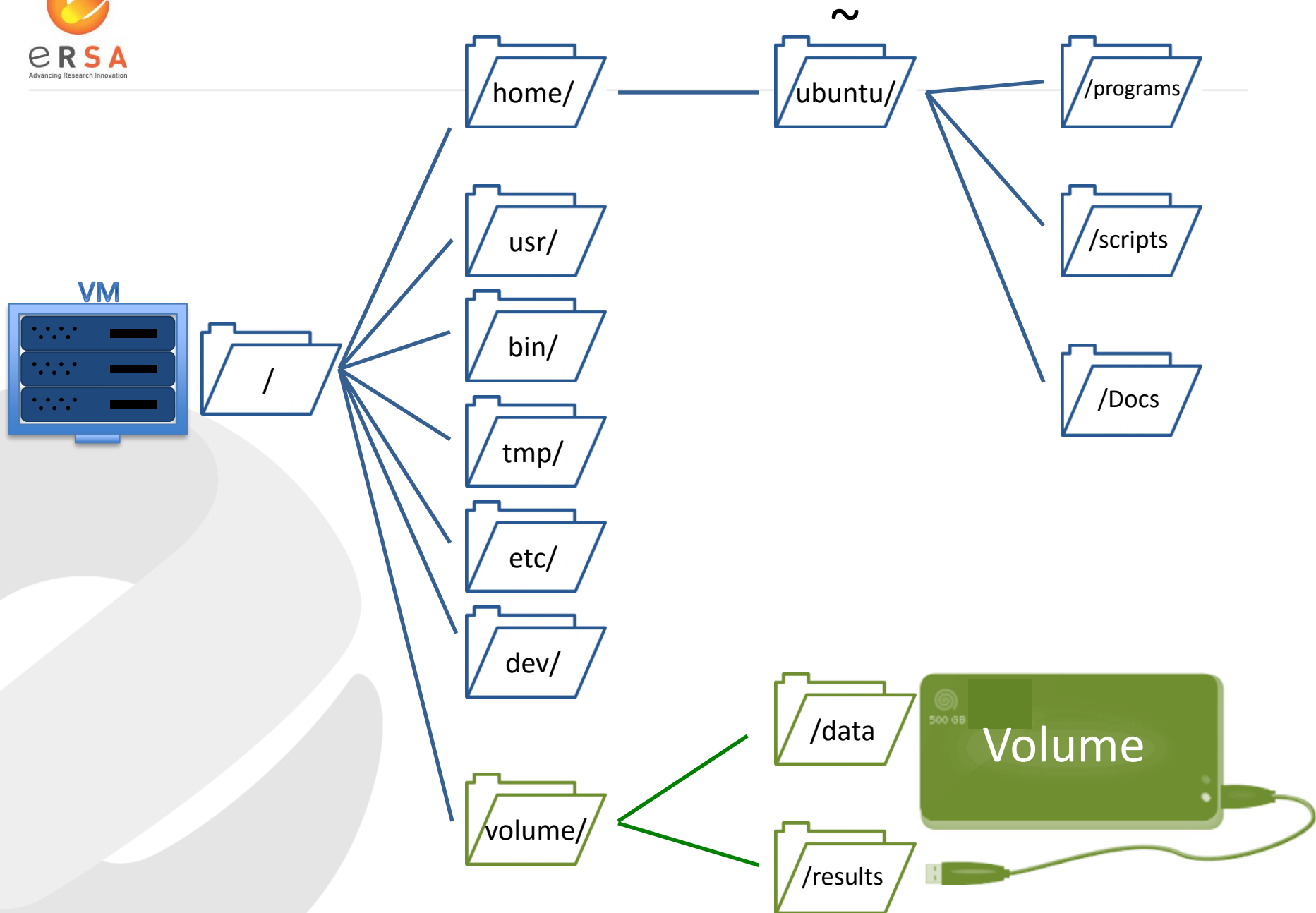
storage

disc

**/dev/vdb**

Secondary  
storage

0 – 480 GB



WORKSHOP: Running a Virtual Machine in the Cloud

# GNU PARALLEL

## GNU Parallel

- A major advantage of NeCTAR cloud computing, is the power to launch VM's with multiple CPUs
- Multiple CPUs can efficiently process more jobs simultaneously.
- GNU parallel is a command-line utility to manage the distribution of a list of jobs to the available CPU cores.

## GNU Parallel

- The GNU parallel utility will allow the user to simultaneously run as many processes as there are CPUs.
- If there are 32 jobs to do and 4 CPUs, parallel will send the first 4 to be done, and as each job finishes a new one will commence.

time



CPU 1



CPU 2



CPU 3



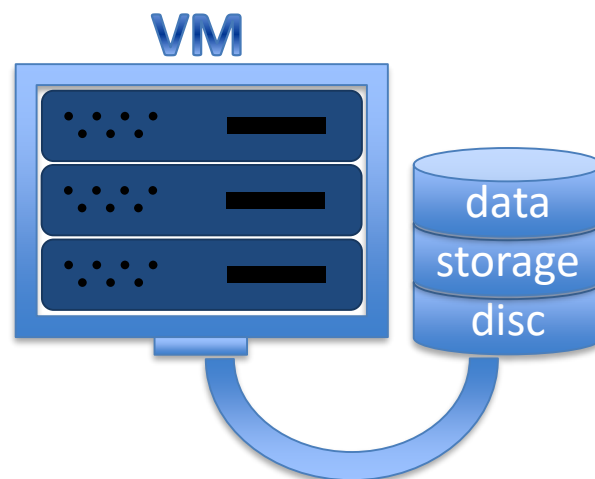
CPU 4

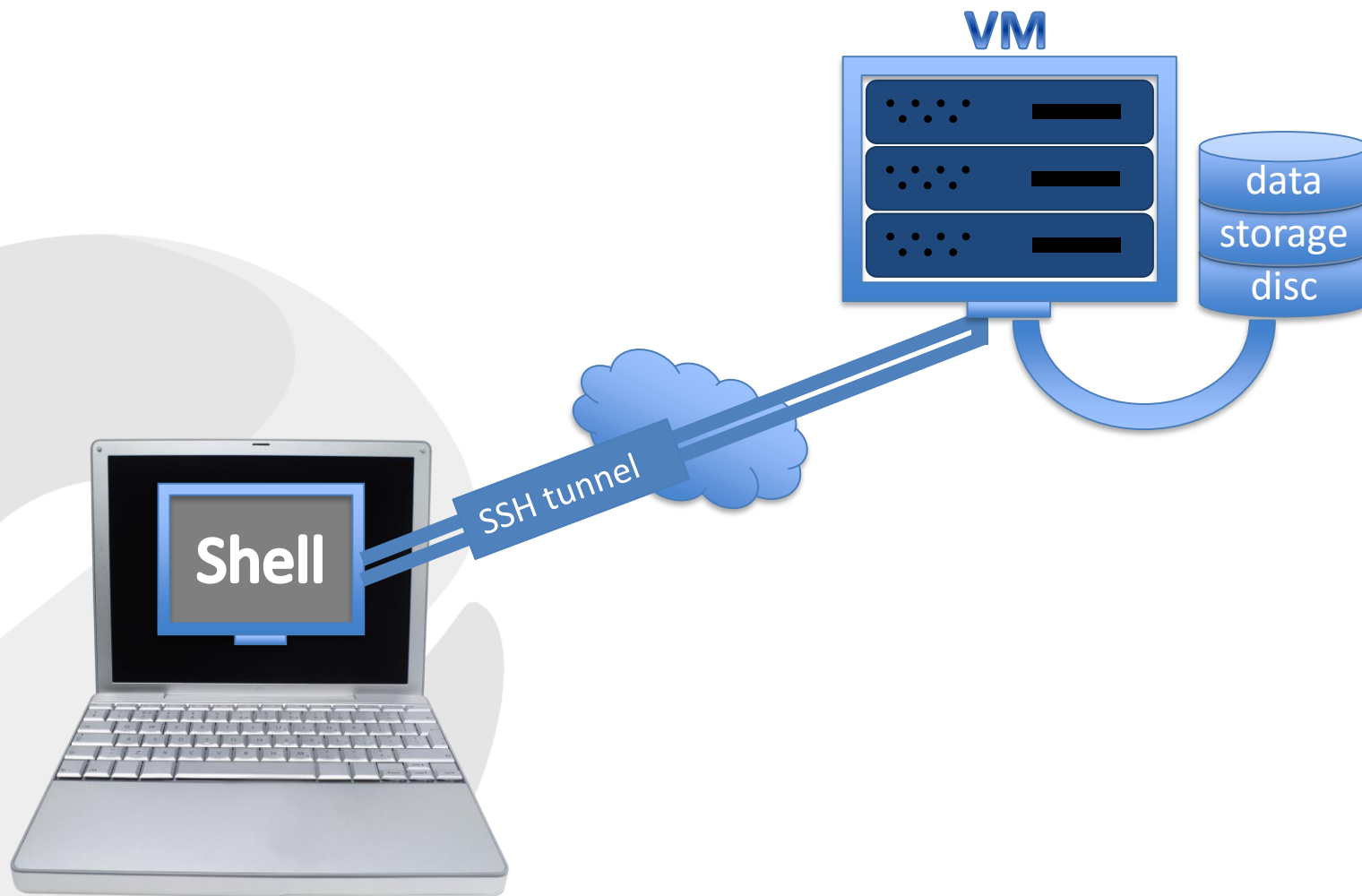


# Running Jobs

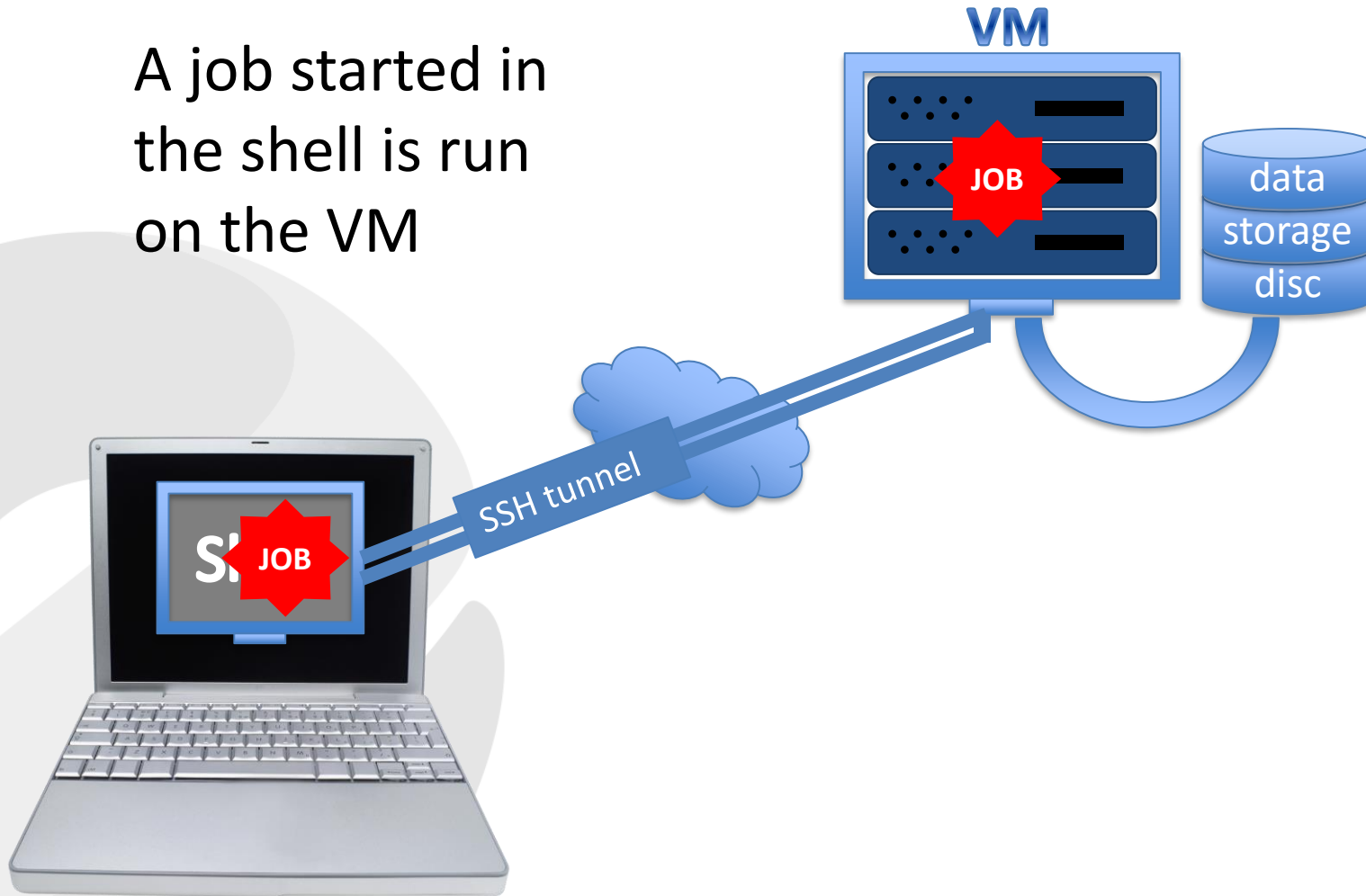
Making sure your computing jobs keep running on your virtual machine when you are no longer connected to it





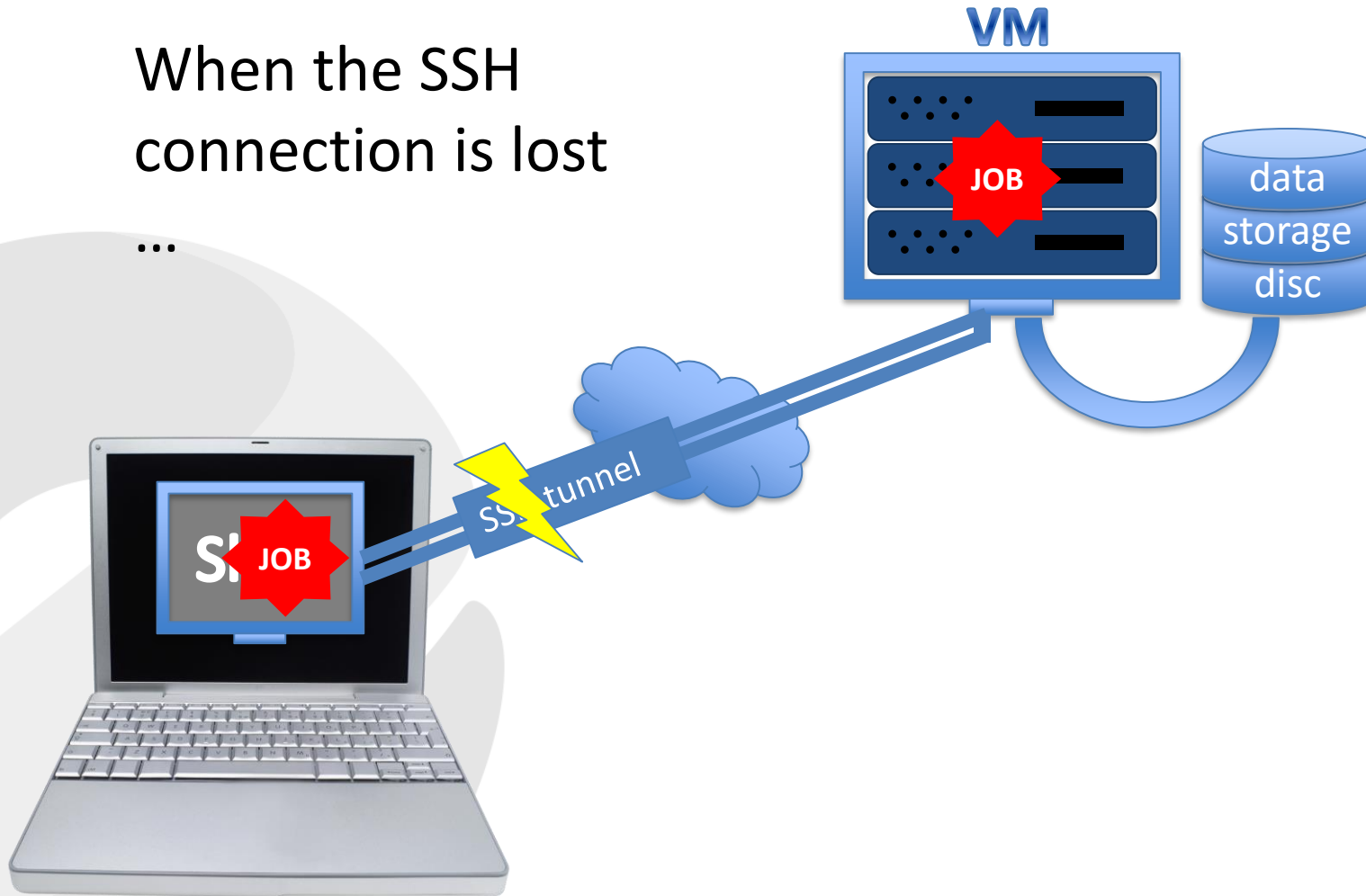


A job started in  
the shell is run  
on the VM

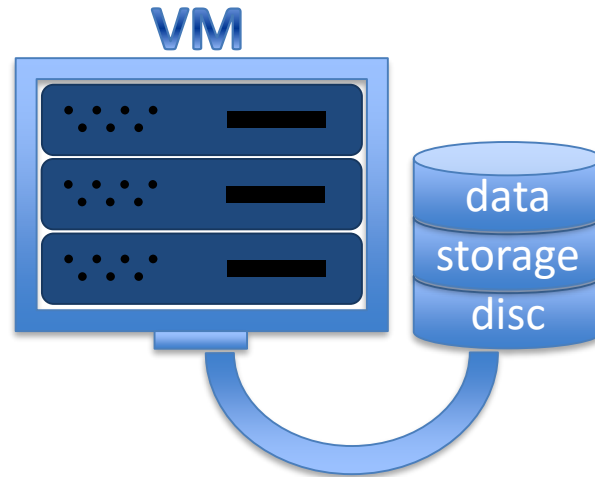


When the SSH  
connection is lost

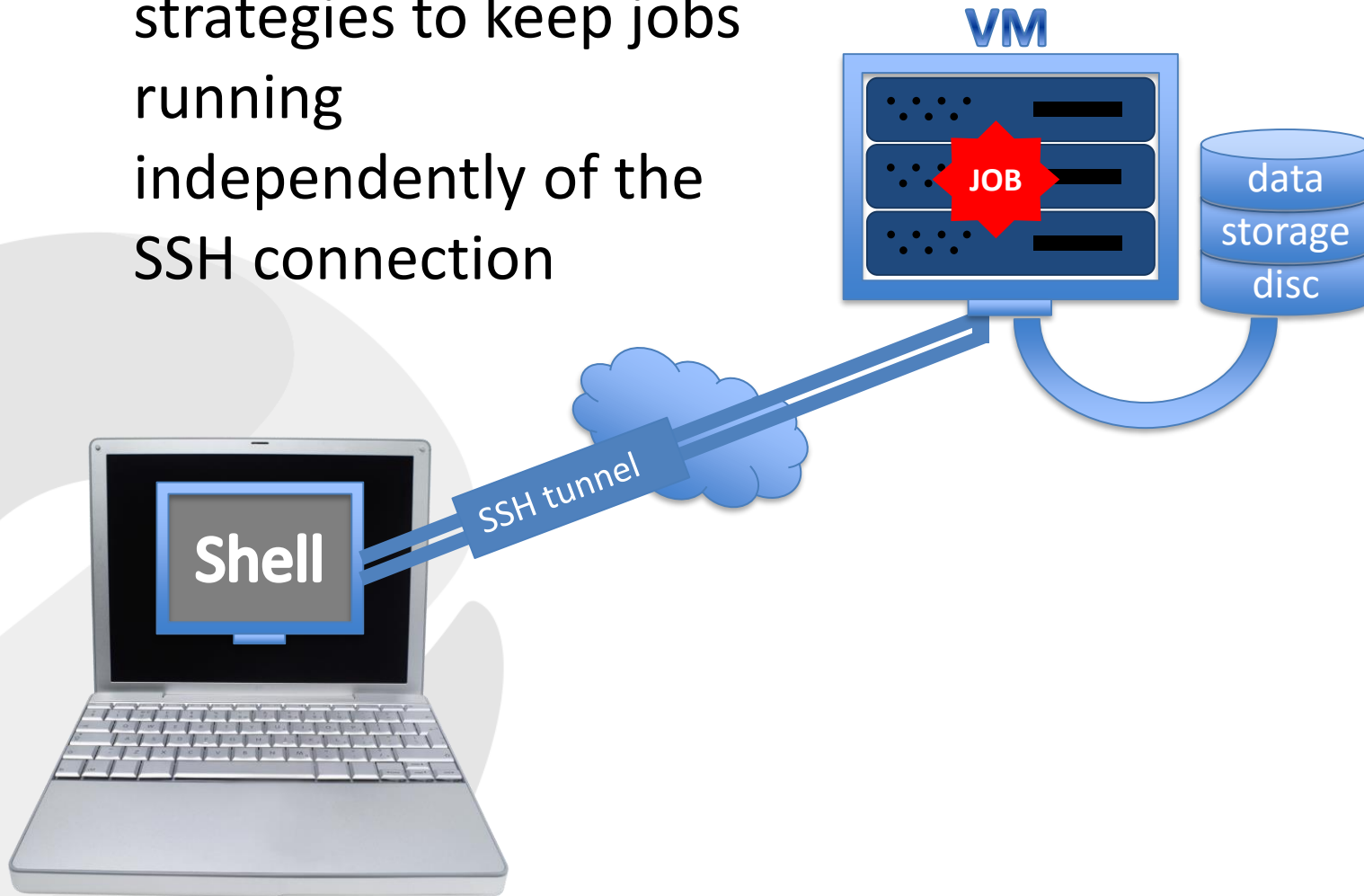
...



... the job will be  
discontinued



But there are  
strategies to keep jobs  
running  
independently of the  
SSH connection



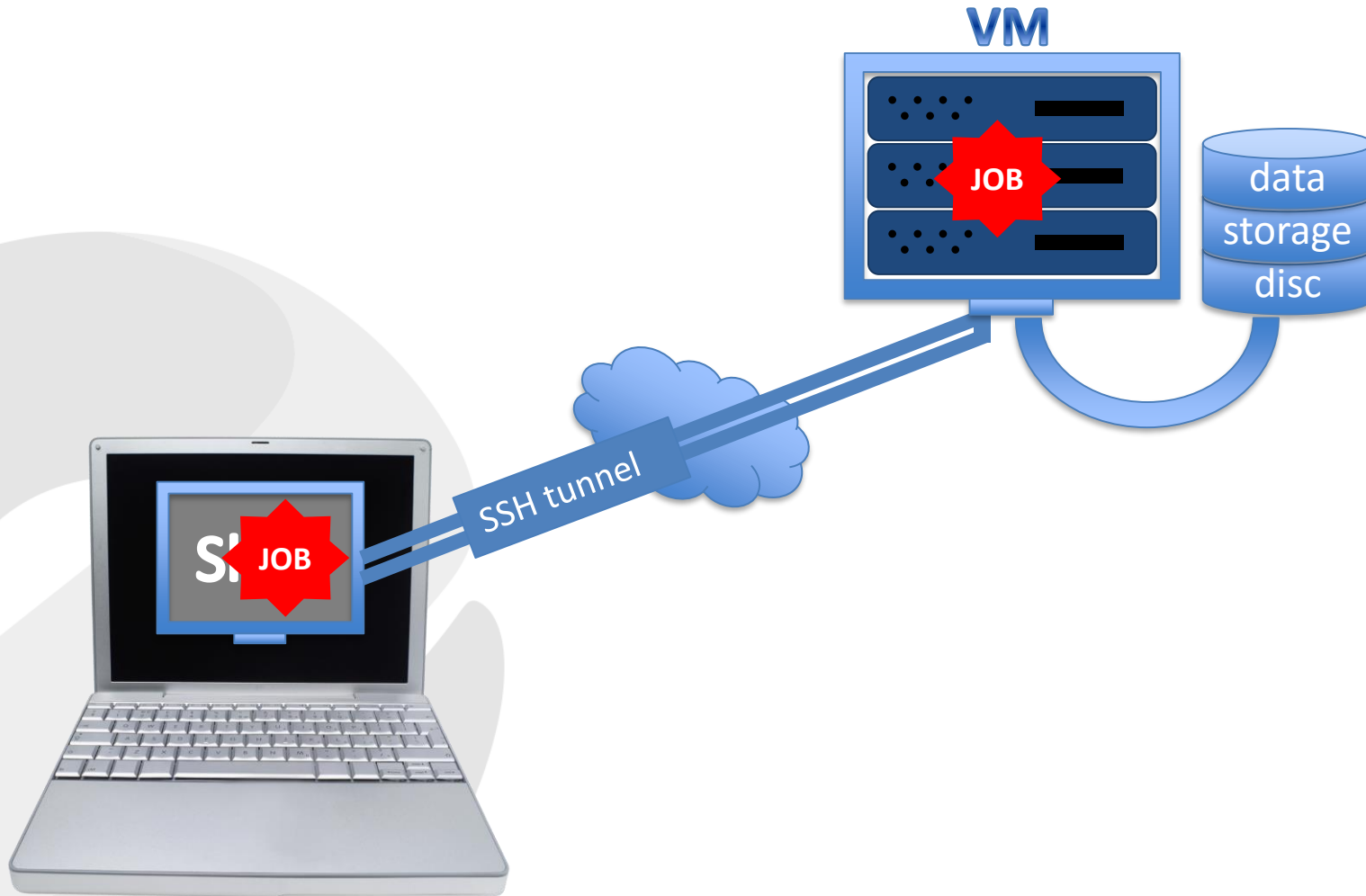
# Running Jobs

1. Detach the job from the shell
  - Nohup, background, disown
2. Persistent virtual consoles
  - Screen, Tmux, Byobu

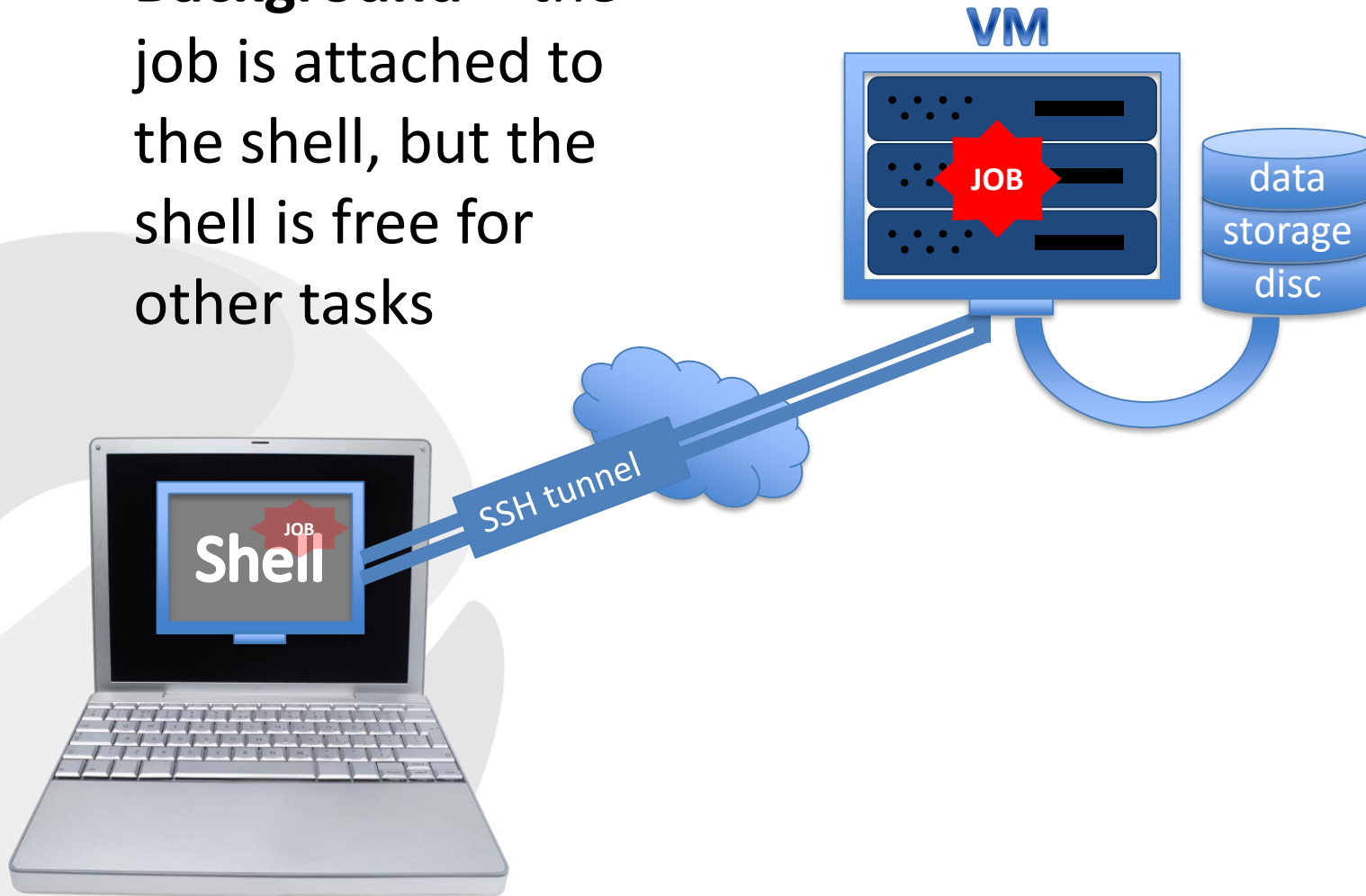
# Detach the job from the shell

nohup

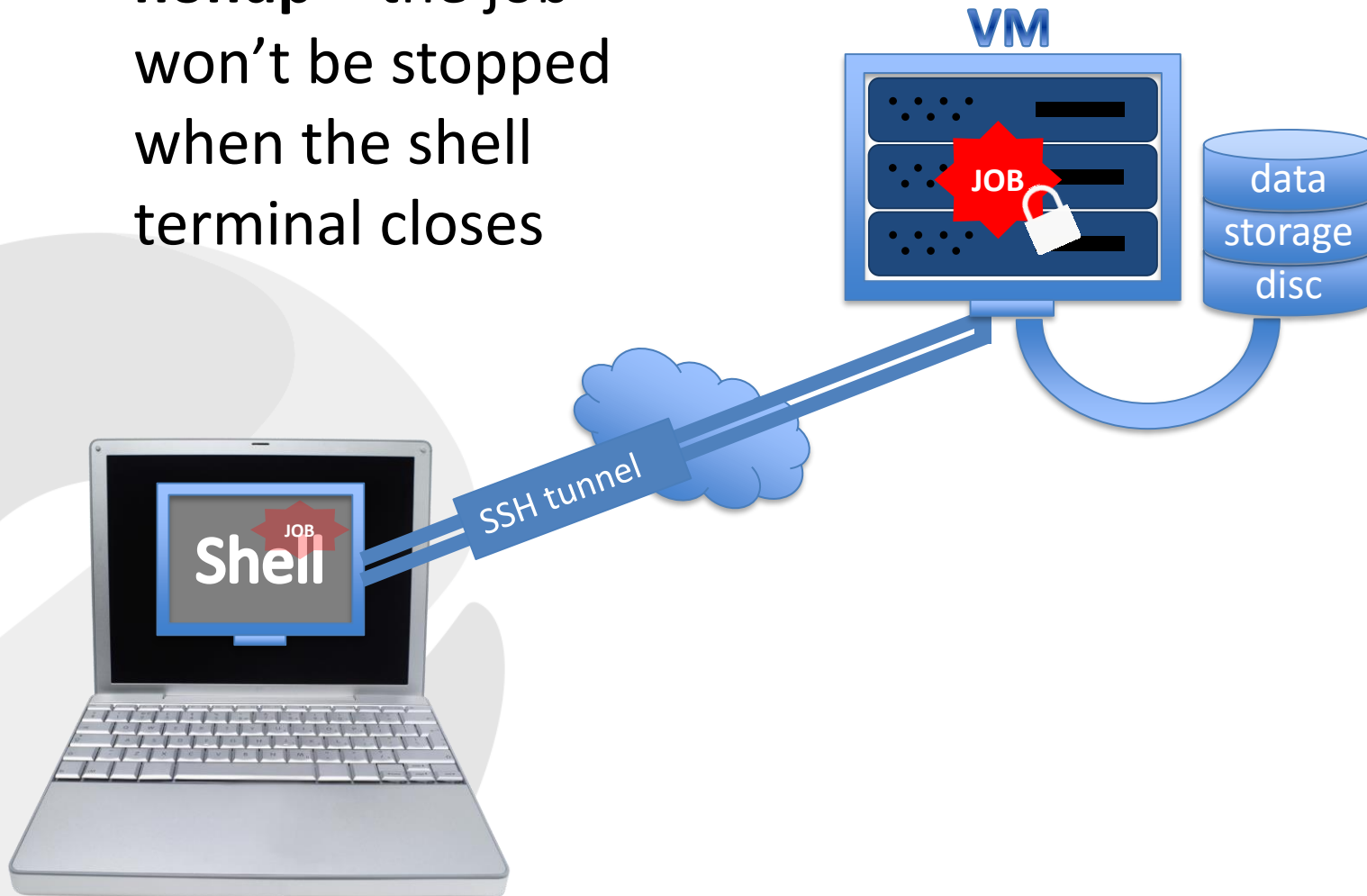




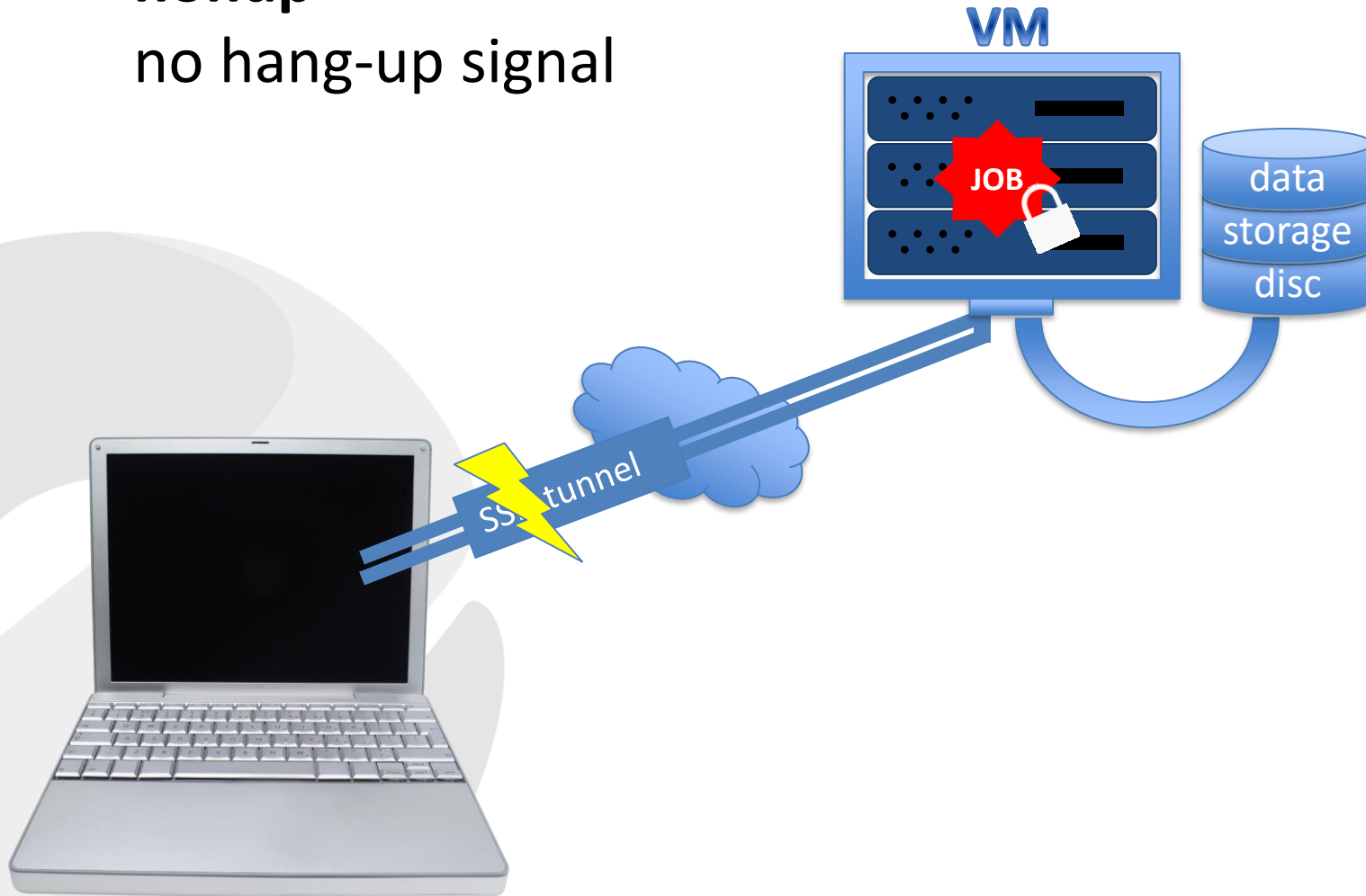
**Background** – the job is attached to the shell, but the shell is free for other tasks



**nohup** – the job  
won't be stopped  
when the shell  
terminal closes



**nohup** =  
no hang-up signal



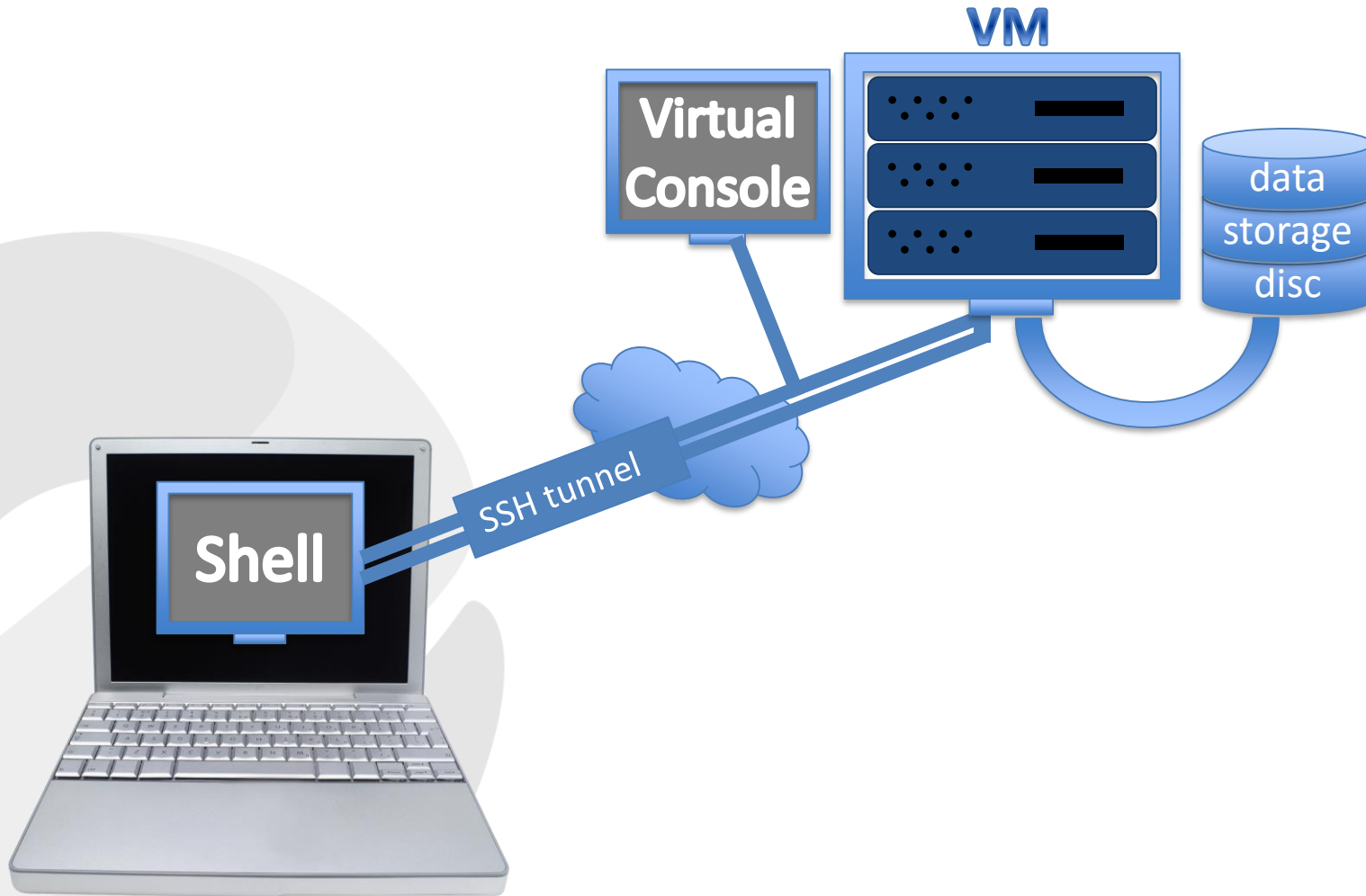
# Persistent virtual consoles

GNU Screen

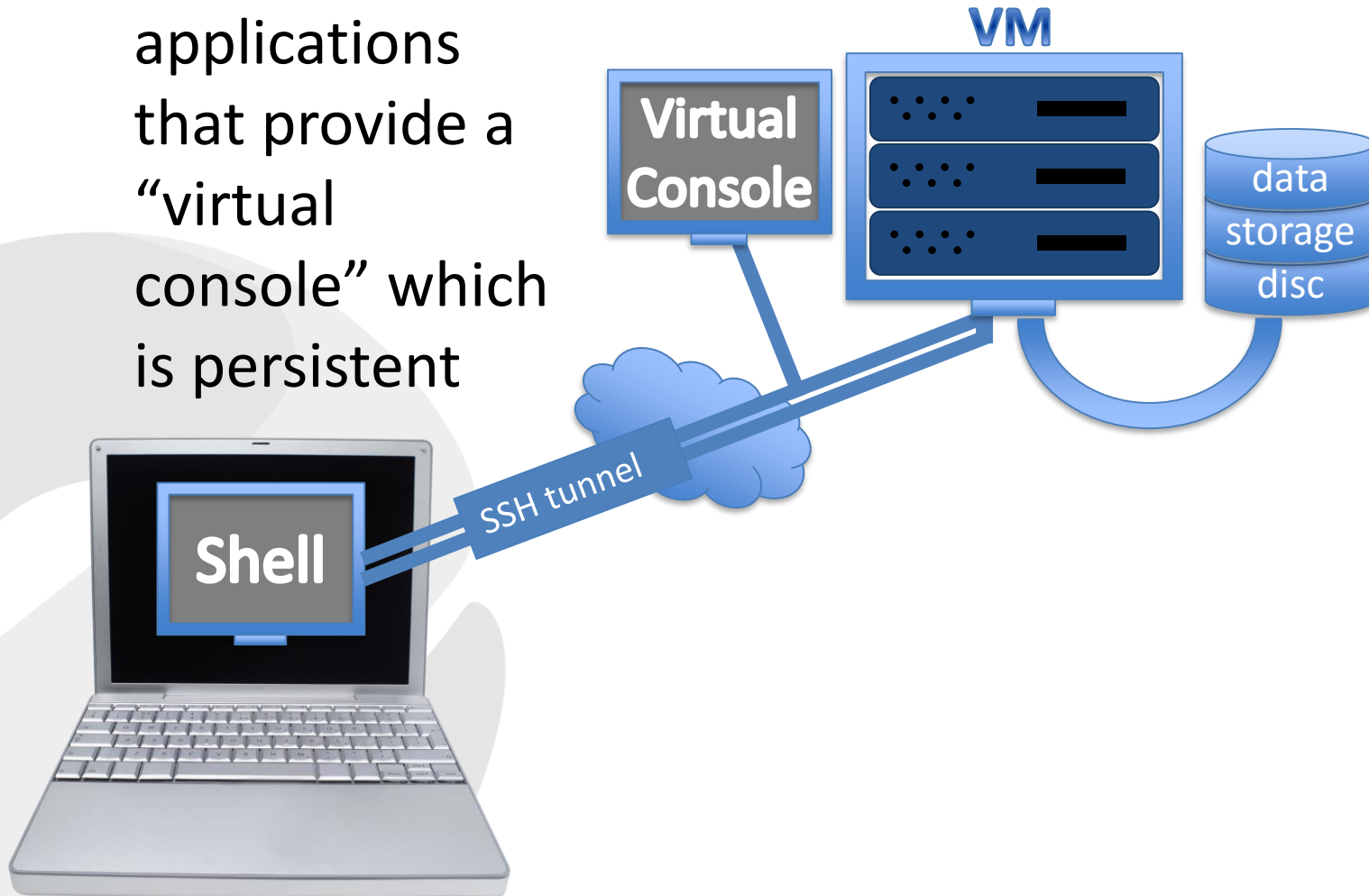
Tmux

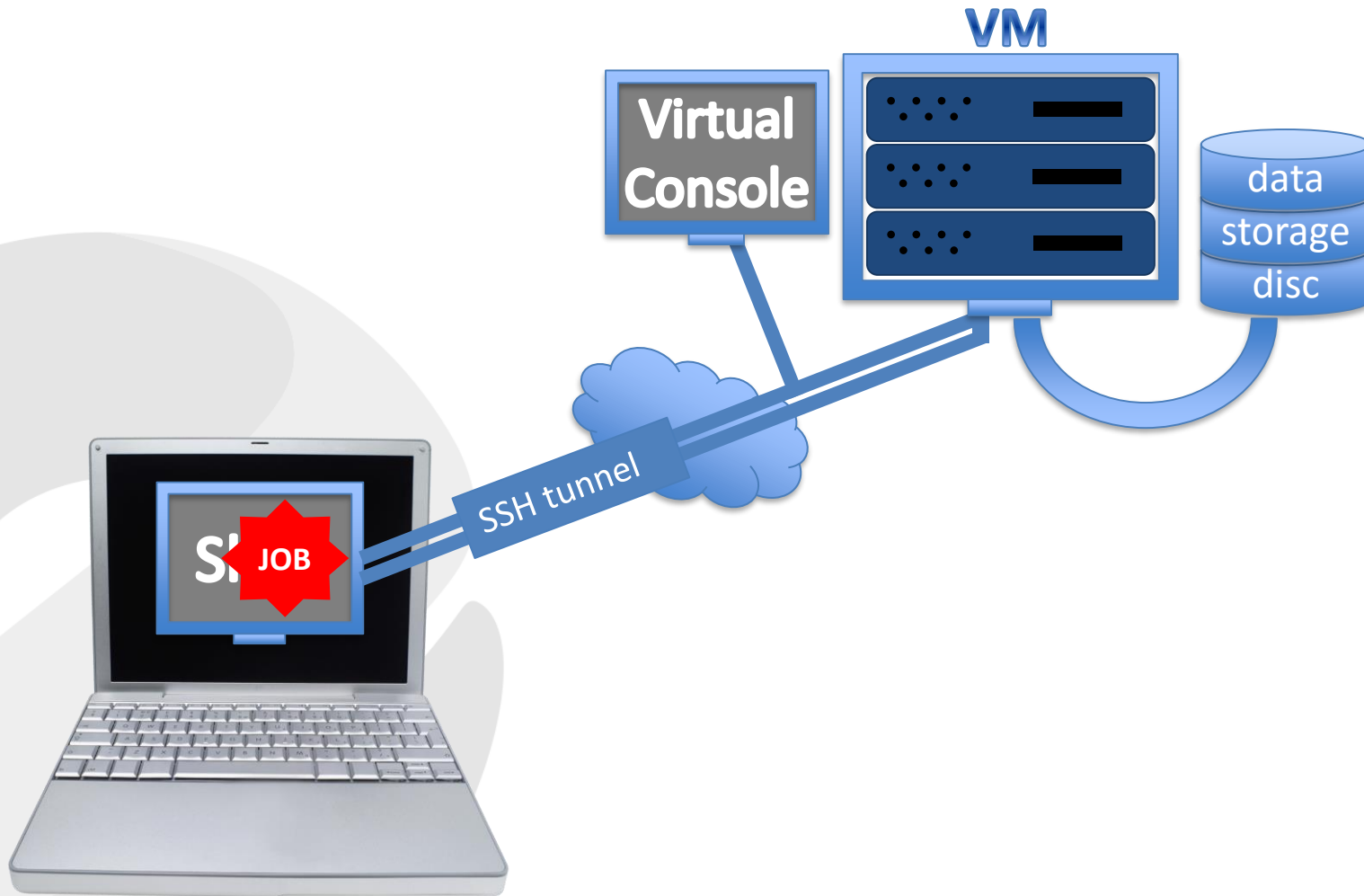
Byobu

# GNU Screen and Tmux

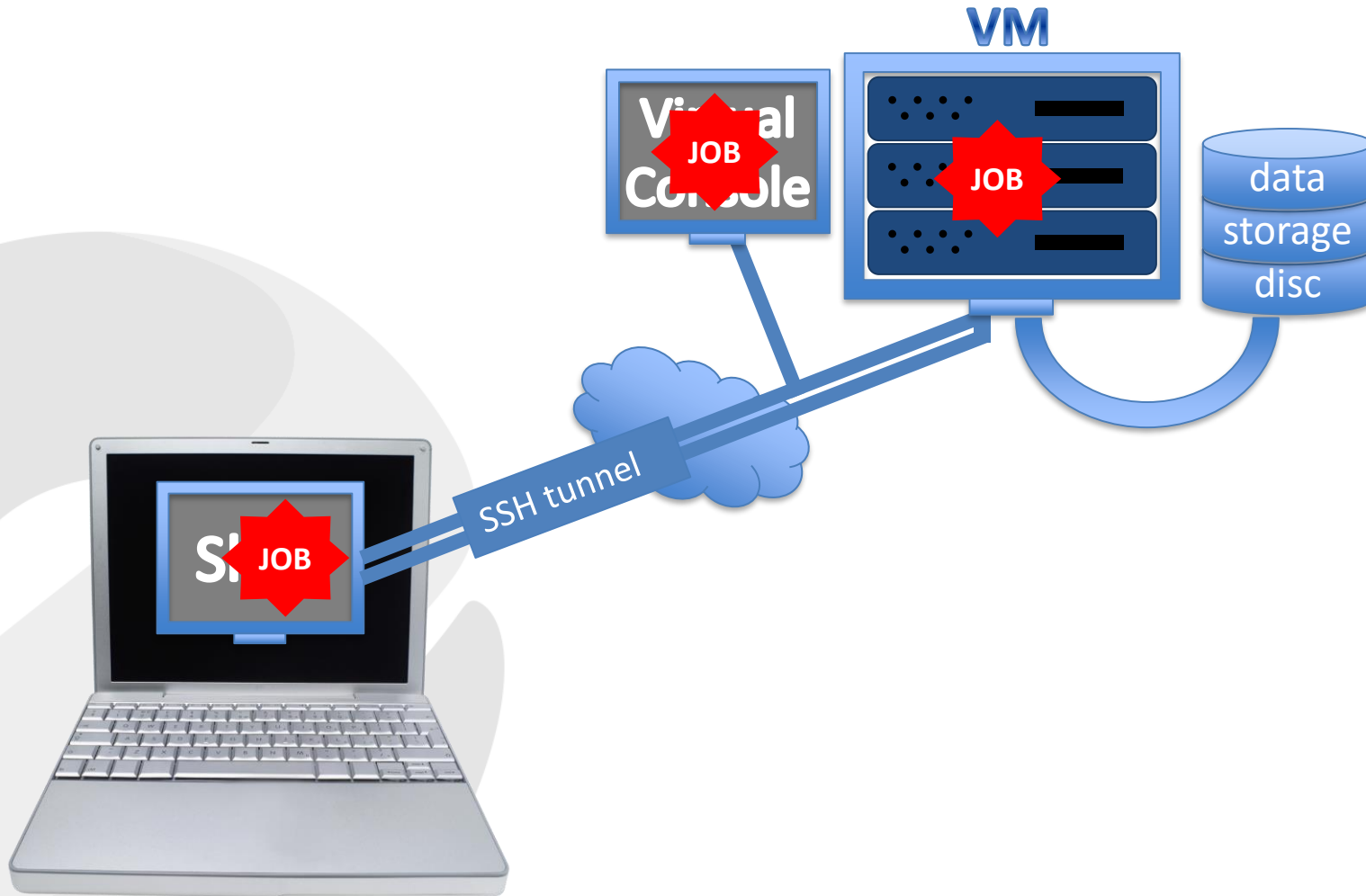


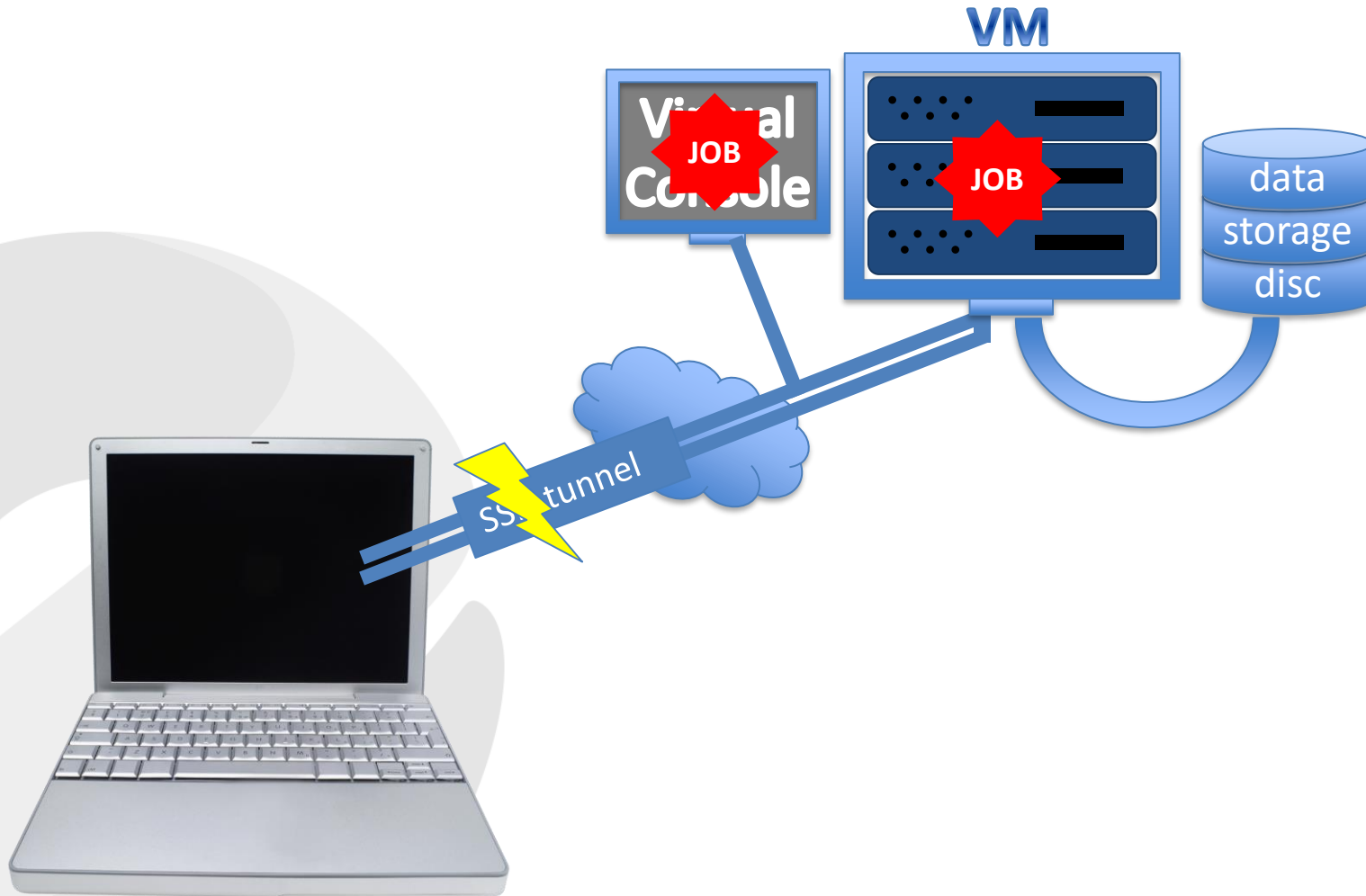
**GNU Screen**  
and **Tmux** –  
applications  
that provide a  
“virtual  
console” which  
is persistent



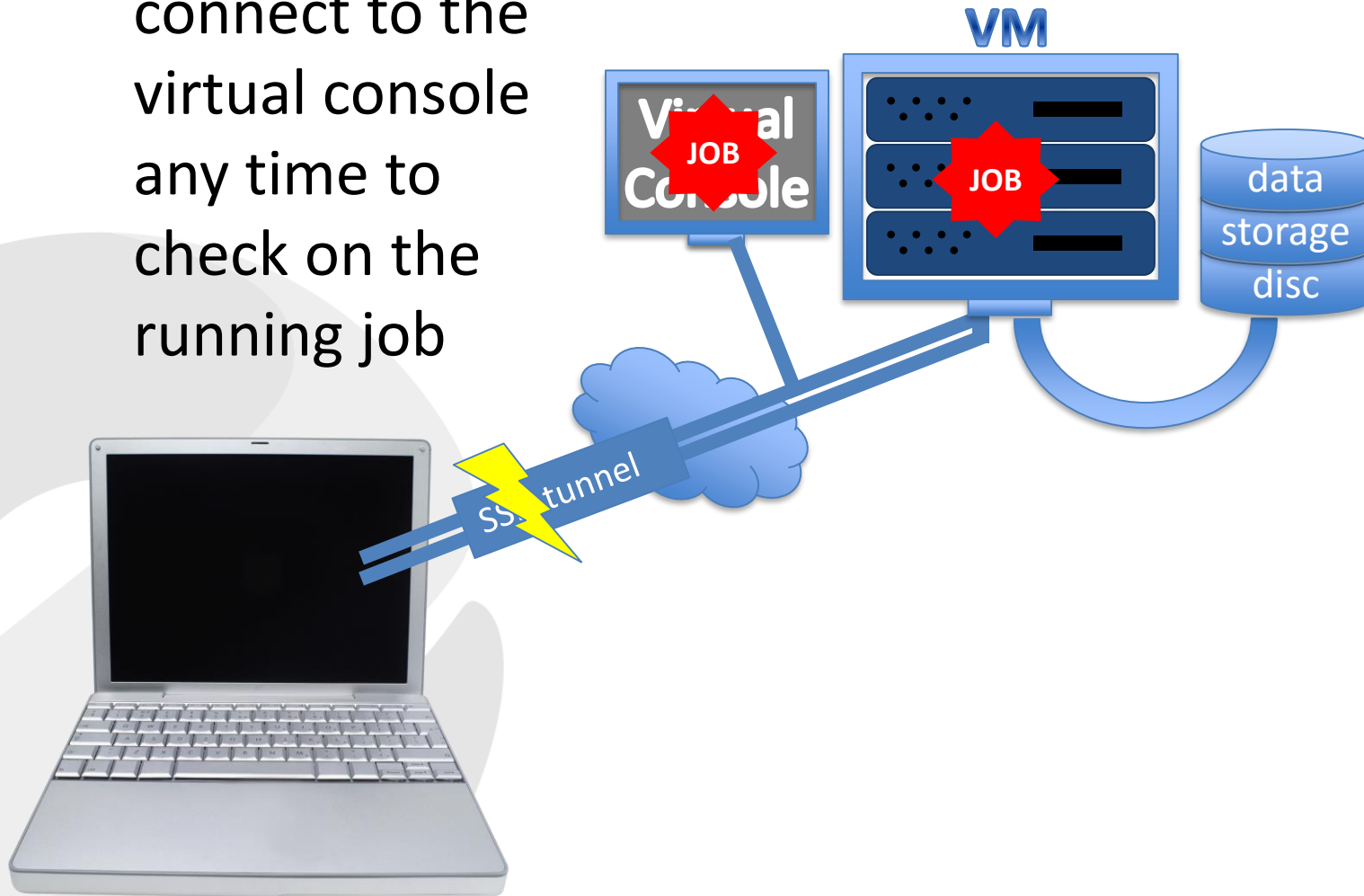




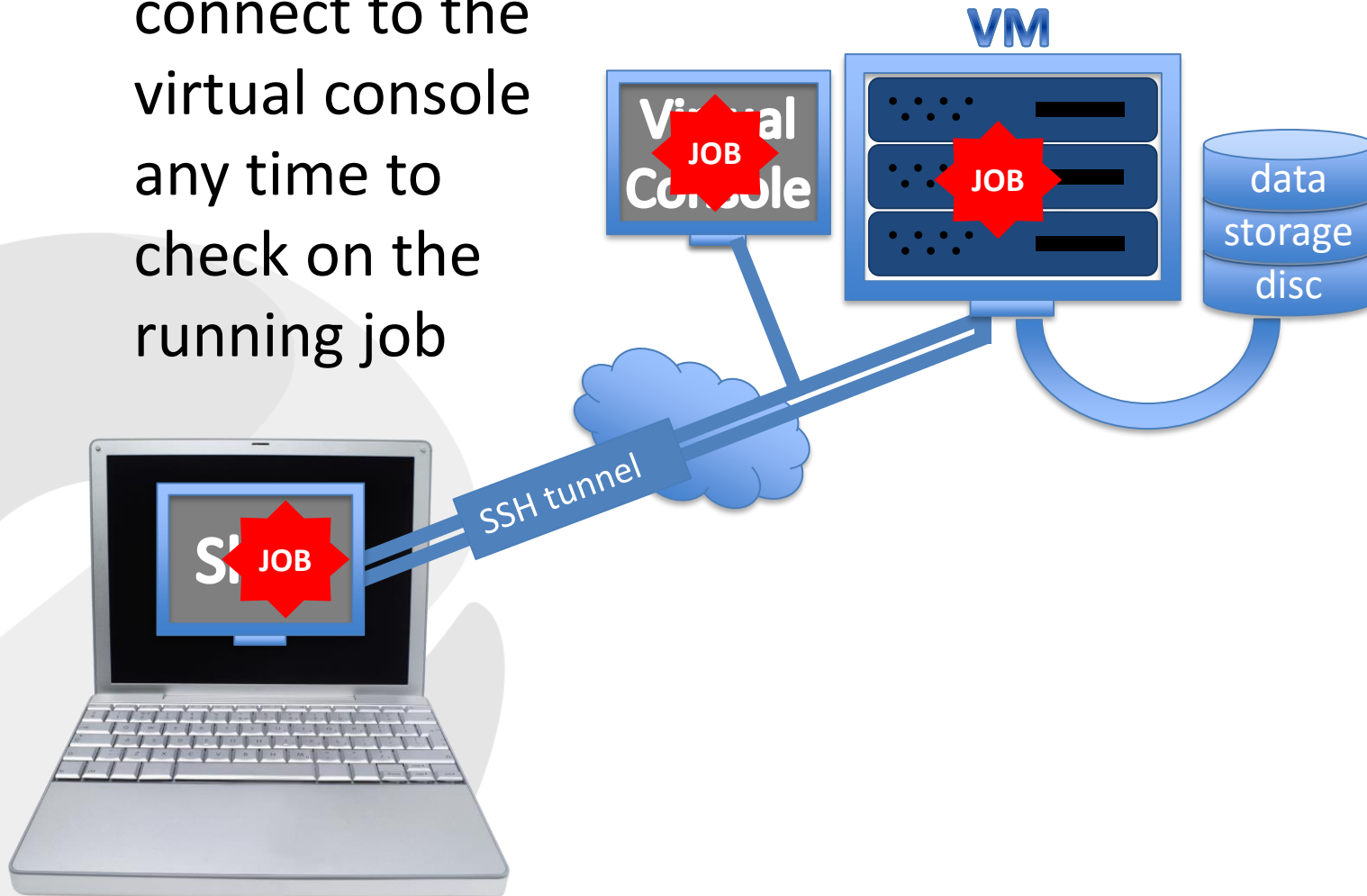




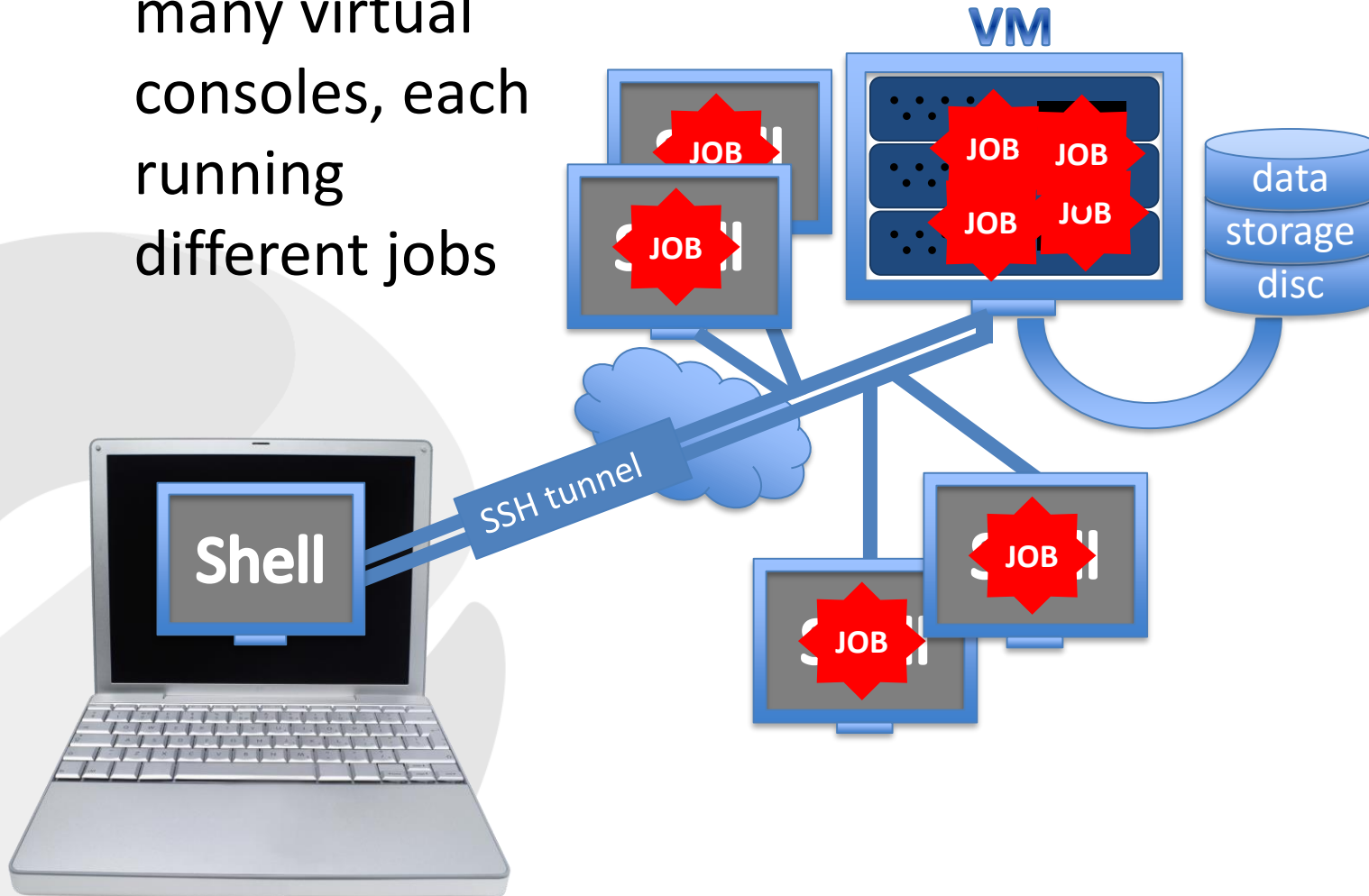
You can  
connect to the  
virtual console  
any time to  
check on the  
running job



You can  
connect to the  
virtual console  
any time to  
check on the  
running job



You can have  
many virtual  
consoles, each  
running  
different jobs



Combine  
“nohup” and a  
virtual console

