ssh:
-q —> Quit mode
-f —> Request ssh to go to background
-n —> Redirects stdin from /dev/null
-g —> Allows remote hosts to connect to local forwarded ports
-N —> Do not execute a remote command
-T —> Disable pseudo-terminal allocation

-R —> [bind address:]port:host:hostport
-R —> [bind address:]port:local_socket
-R —> remote_socket:host:hostport
-R —> remote_socket:local_socket
-R —> [bind address:]port

-L —> [bind address:]port:host:hostport
-L —> [bind address:]port:remote_socket
-L —> local_socket:host:hostport
-L —> local_socket:remote_socket

-l —> login_name
-B —> bind_interface
-b —> bind_address
-C —> Request compression of all data

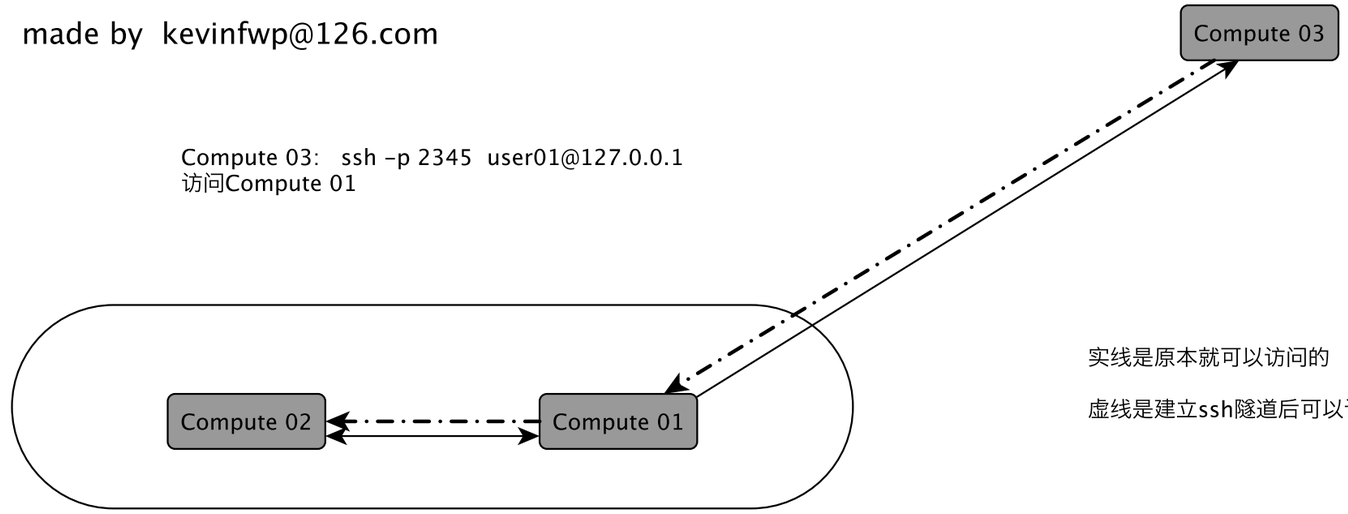
-D —> [bind address:]port
Specifies a local 'dynamic' applicaton-level port forwarding
-i —> identity_file
Select a file from which the identity (private key)
-M —> Places the ssh client into 'master' mode for connection sharing
-p —> port
-t —> Force pseudo-terminal allocation
-V —> Display the version number and exit
-v —> Verbose mode. The maximum is 3
-X —> Enables X11 forwarding
-x —> Disable X11 forwarding
-Y —> Enable trusted X11 forwarding

~/.ssh/authorized_keys

~. - terminate connection (and any multiplexed sessions)
~B - send a BREAK to the remote system
~C - open a command line
~R - request rekey
~V/v - decrease/increase verbosity (LogLevel)
~^Z - suspend ssh
~# - list forwarded connections
~& - background ssh (when waiting for connections to terminate)
~? - this message
~~ - send the escape character by typing it twice

ssh> help
Commands:
-L[bind_address:]port:host:hostport Request local forward
-R[bind_address:]port:host:hostport Request remote forward
-D[bind_address:]port Request dynamic forward
-KL[bind_address:]port Cancel local forward
-KR[bind_address:]port Cancel remote forward
-KD[bind_address:]port Cancel dynamic forward

made by kevinfwp@126.com



Compute 03: ssh -p 2345 user01@127.0.0.1
访问Compute 01

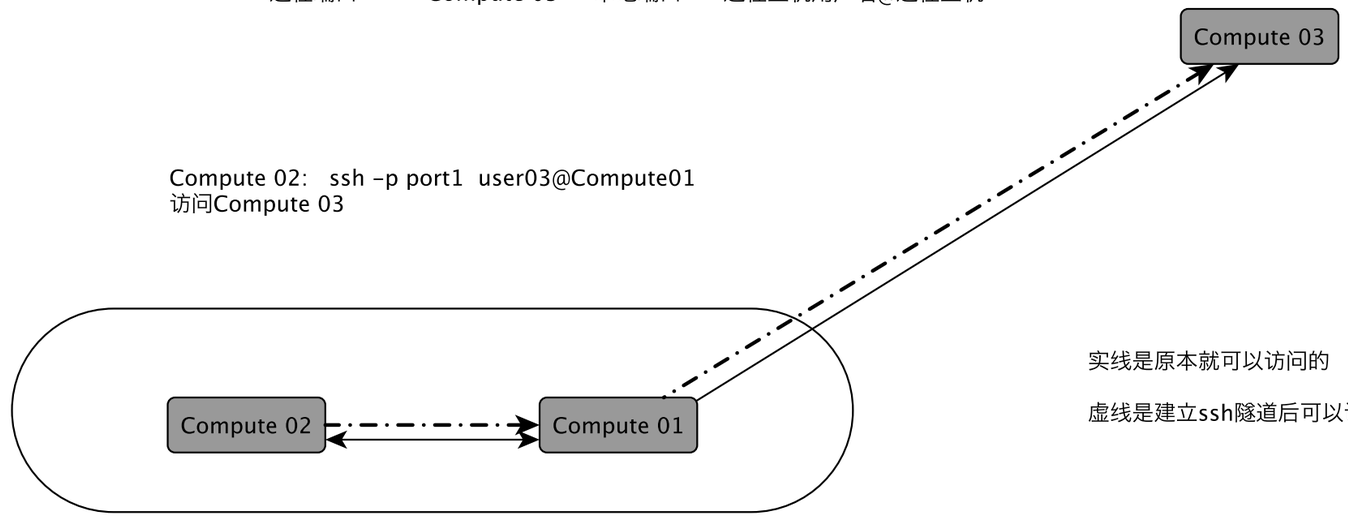
实线是原本就可以访问的

虚线是建立ssh隧道后可以访问

Compute 01可以访问Compute 03，Compute 03不能访问Compute 01和Compute 02，
通过Compute 01建立ssh反向连接，让Compute 03可以访问Compute 01和Compute 02

Compute 01: ssh -p 2222 -qngfNTR 2345:localhost:22 【user】@【Compute 03】

远程端口 Compute 03 本地端口 远程主机用户名@远程主机



Compute 02: ssh -p port1 user03@Compute01
访问Compute 03

实线是原本就可以访问的

虚线是建立ssh隧道后可以访问

Compute 01可以访问Compute 03，Compute 02不能访问Compute 03，但Compute 02可以访问Compute 01
通过Compute 01建立ssh正向连接，让Compute 02可以访问Compute 03

Compute 01: ssh -qfNTL *:【port1】:【Compute 03】:【port2】 localhost

本地被访问地址 本地被访问端口 C03主机 C03端口 本地主机