

Port Forwarding SSH from Virtual Box

Note: This post is reference [for a class](#). It is assumed that you have used VirtualBox for [running desktop versions of Linux](#), and have a [basic grasp of Linux commands](#).

In this post I will discuss setting up port forwarding to manage a local linux virtual machine through ssh, with a focus on mirroring the workflow you might follow with a dedicated Linux server.

This guide will be using Ubuntu 16.04.1 and [MobaXterm](#) as the SSH client. You are welcome to use another SSH client such as [bash for windows in Windows 10](#) if you so choose.

Downloads

- Download the latest version of Virtual box: [Download link](#)
- Download the latest free version of MobaXterm: [Download link](#)
- Download the Latest version of Ubuntu Server, which at the time of this writing is 16.04.1: [Download link](#)

Installation

Install VirtualBox and MobaXterm.

Go through the general VirtualBox install for ubuntu. Keep in mind:

- Choose a Dynamically sized Virtual Harddrive of a reasonable size, probably limited to 20GB or so (can always be changed later)
- When starting for the first time, select your downloaded ISO when prompted.
- Most settings should be left default.
- When prompted to choose `Software to install`, make sure to install OpenSSH Server (move the cursor down and press space to select it, **NOT** enter.) <https://i.imgur.com/9mdAgY5.png>
 - If you accidentally skip installing the OpenSSH Server you can always install it later with
`sudo apt-get install openssh-server`

Initial Setup

Once the install has finished, you will be prompted with the standard Linux login.

```
Ubuntu 16.04.1 LTS ubuntu tty1
```

```
ubuntu login: _
```

While you can log in here and use this console to interact with your server, it is very limited:

- You cannot resize the window.
- You cannot copy/paste text from your host OS.
- You cannot scroll to look at more history than is currently on the screen.

Luckily we will only log into it this way once before we can use a proper SSH client.

Log into the account you created during installation. Once logged in run the following command to see if SSH is running: `sudo service ssh status`. If you see something like this:

```
jo@ubuntu:~$ sudo service ssh status
• ssh.service - OpenBSD Secure Shell server
  Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
  Active: active (running) since Wed 2017-01-25 15:45:20 PST; 1min 45s ago
  Main PID: 3202 (sshd)
  CGroup: /system.slice/ssh.service
          └─3202 /usr/sbin/sshd -D
```

Then you have ssh properly installed.

If you see an error then you may need to install ssh with `sudo apt-get install openssh-server`

Once you see that SSH is running we will need the local IP address that VirtualBox is using internally for this Virtual Machine. To do so we can run `ifconfig` and filter it for lines that contain the string "inet addr". Run

```
ifconfig | grep "inet addr"
jo@ubuntu:~$ ifconfig | grep "inet addr"
    inet addr:10.0.2.15 Bcast:10.0.2.255 Mask:255.255.255.0
    inet addr:127.0.0.1 Mask:255.0.0.0
```

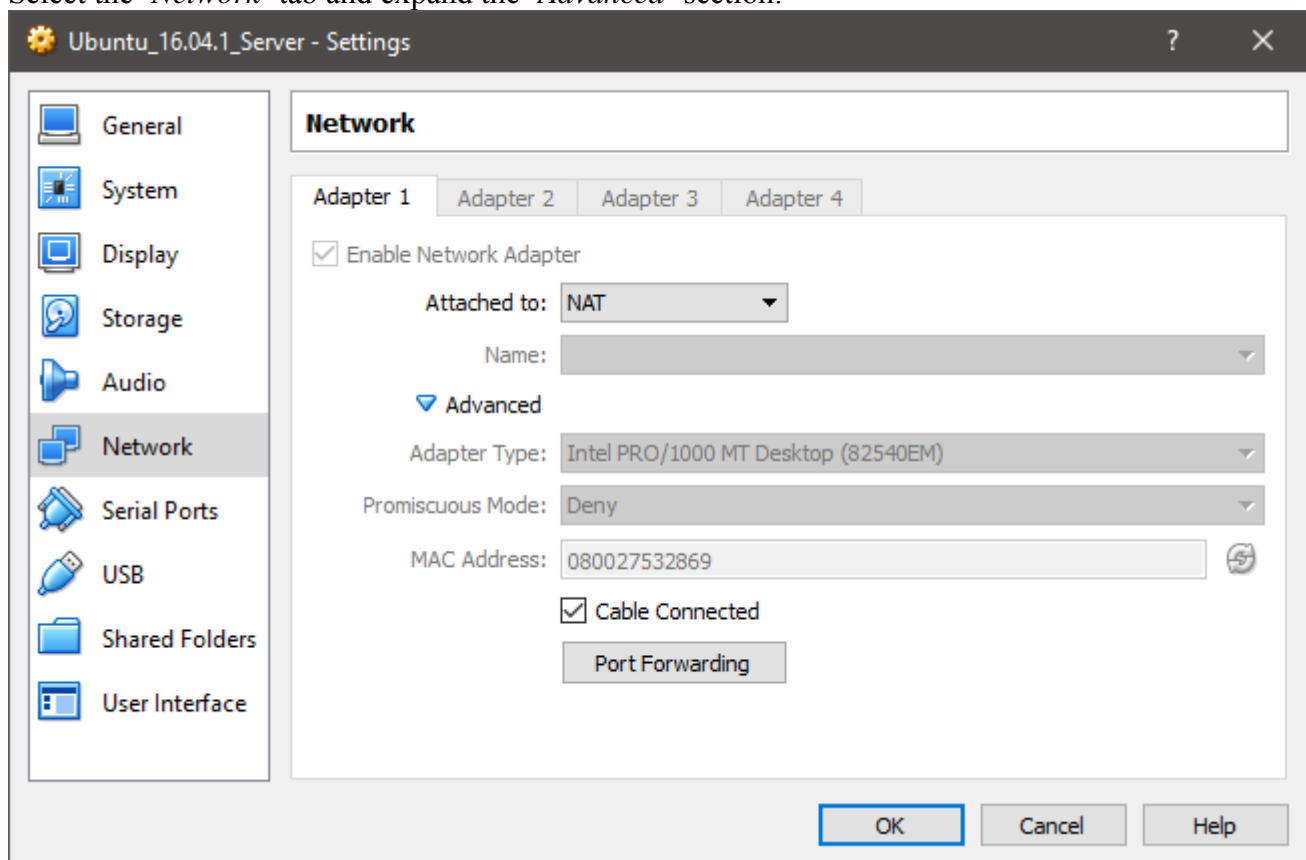
127.0.0.1 is your loopback address so In this case the IP address you want is **10.0.2.15** but that may be different in your case. Make a note of this IP address, so we can use it to port forward SSH connections from your host OS into the virtual machine.

You can now type `exit` to log out of your user session. Ideally we will never have to log in through this console again, and will instead use SSH.

Port forwarding SSH

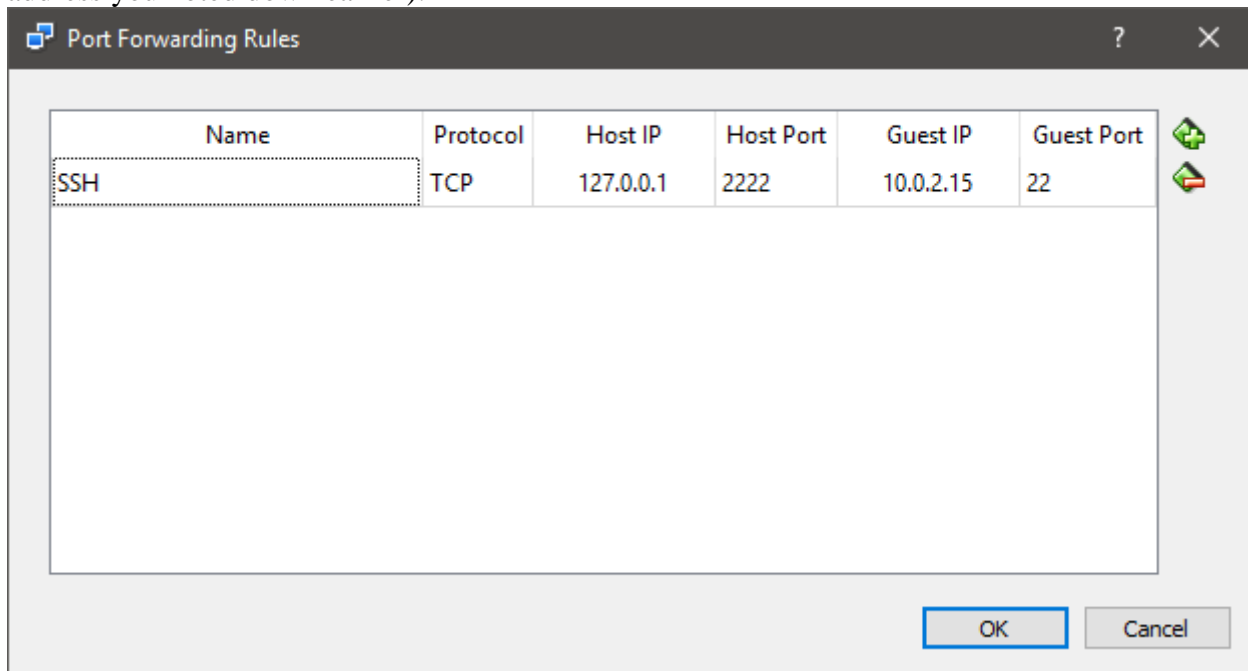
Minimize your server's window (make sure not to close it), select it in the VirtualBox Siderbar and choose 'Settings'.

Select the 'Network' tab and expand the 'Advanced' section:



We will now forward an unused port from our native host OS to the SSH port (port 22) on the virtual machine. To do so we will have to pick a port that is not already in use. In this case we will use port 2222.

Click '*Port Forwarding*', and create a new Rule as follows (make sure to change 10.0.2.15 to match the IP address you noted down earlier):

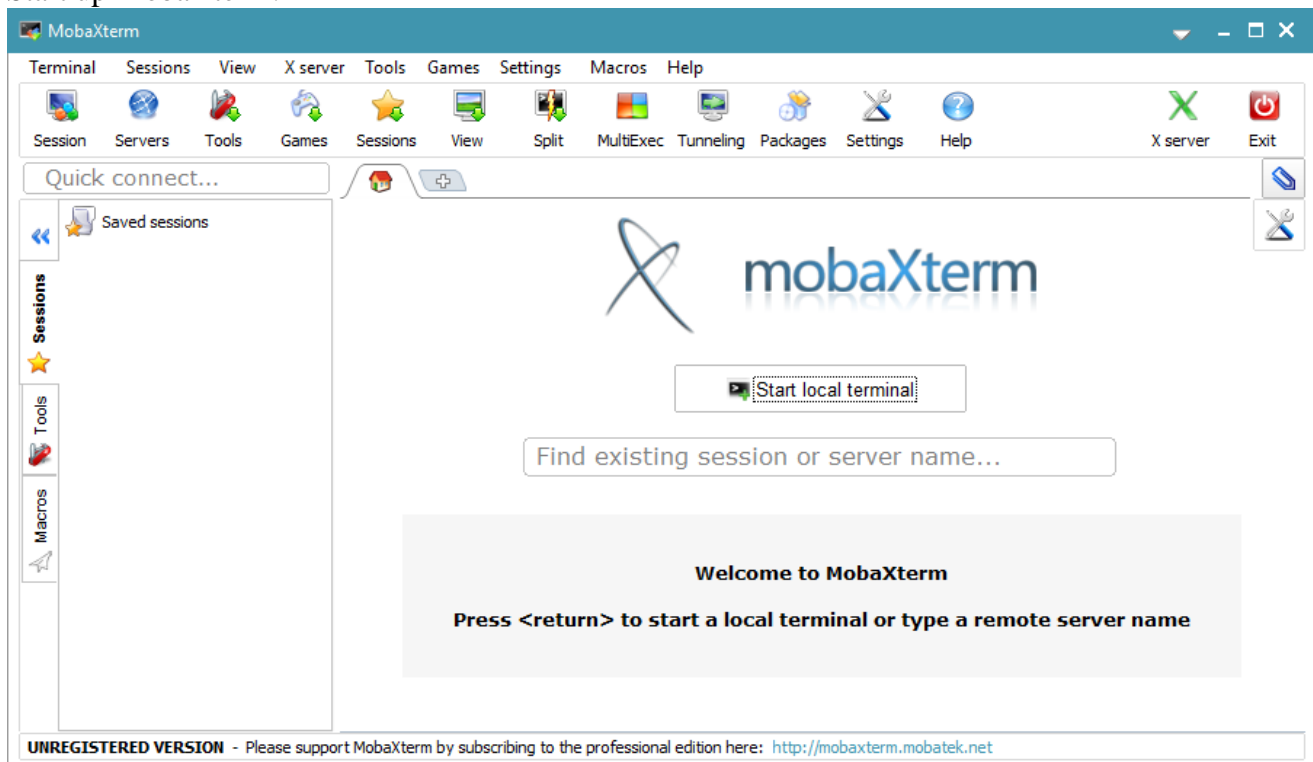


Basically this rule allows any requests made on port 2222 of your local machine to be redirected to port 22 of the virtual machine.

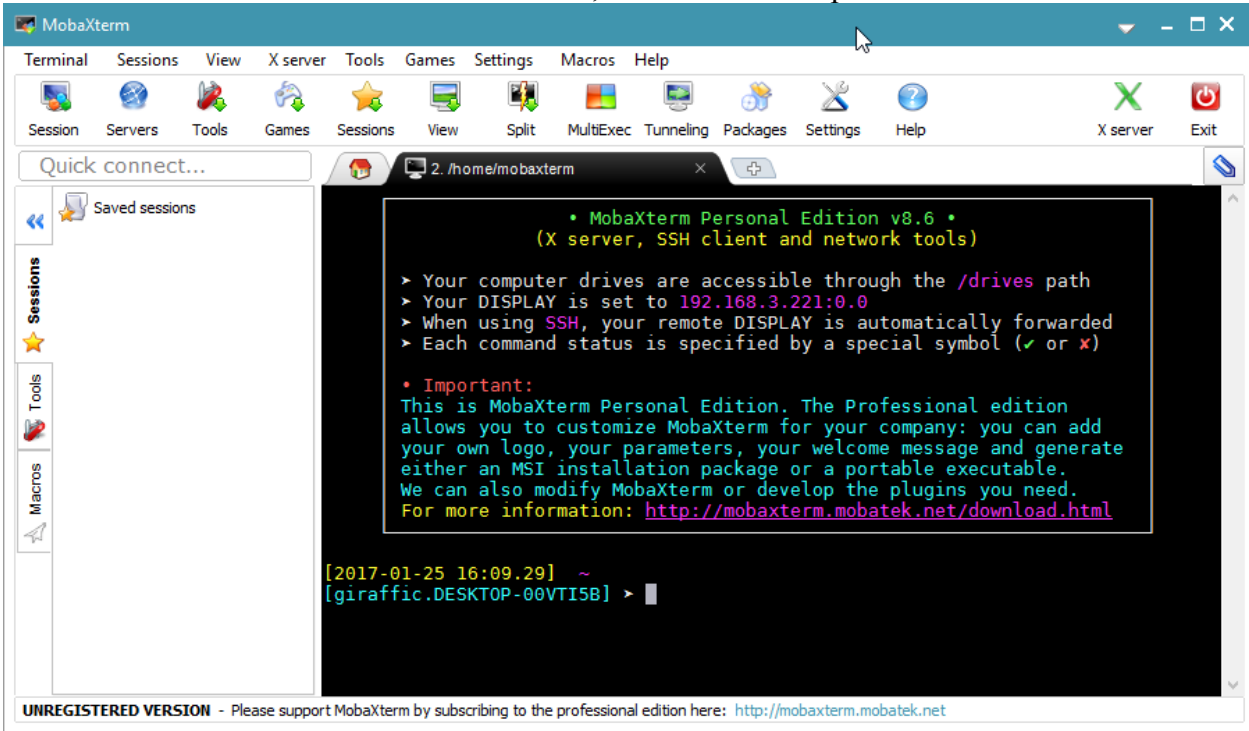
Click 'OK' to close the settings window and open your SSH client of choice, such as MobaXterm.

Connecting via SSH

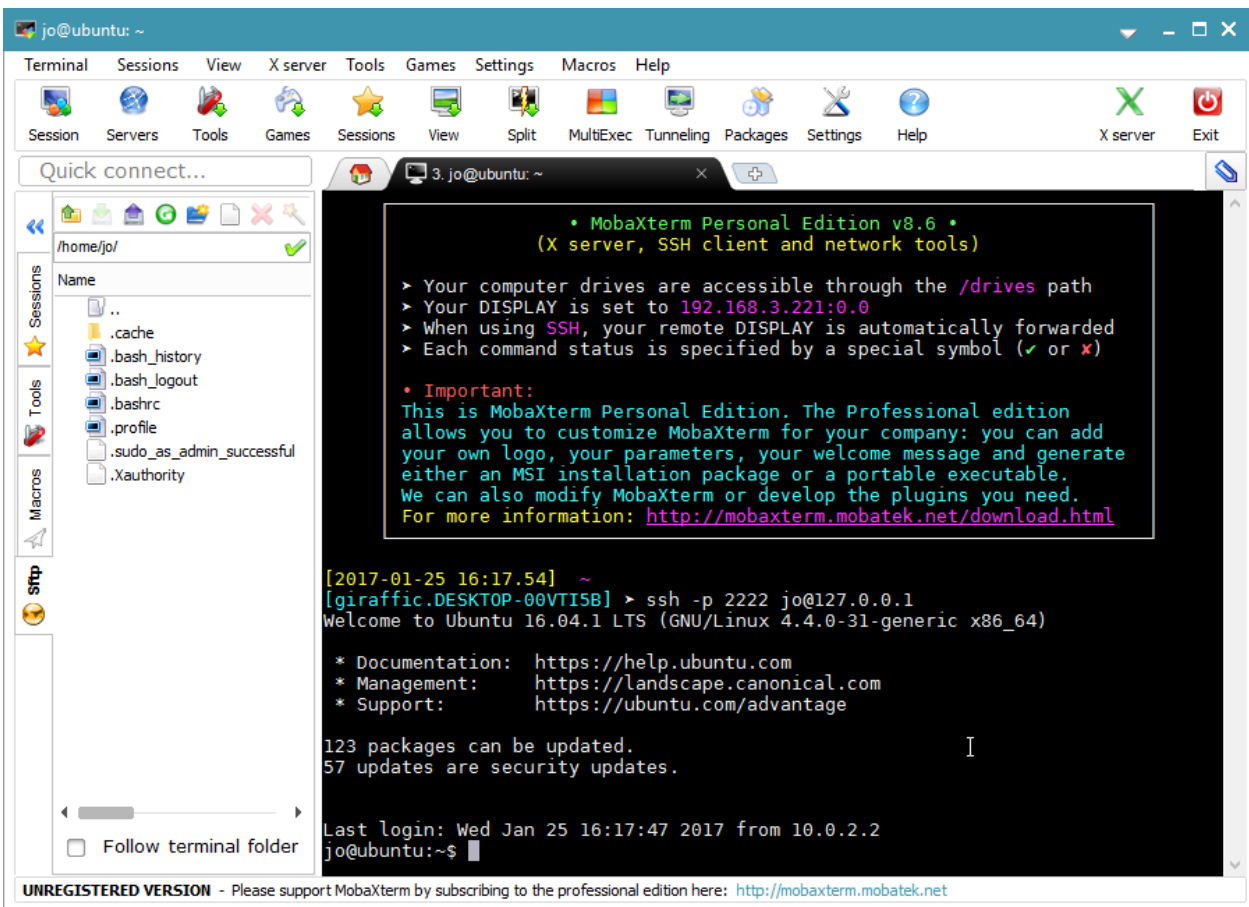
Start up MobaXterm:



In the main section click 'Start local terminal', which will start up a standard bash shell:

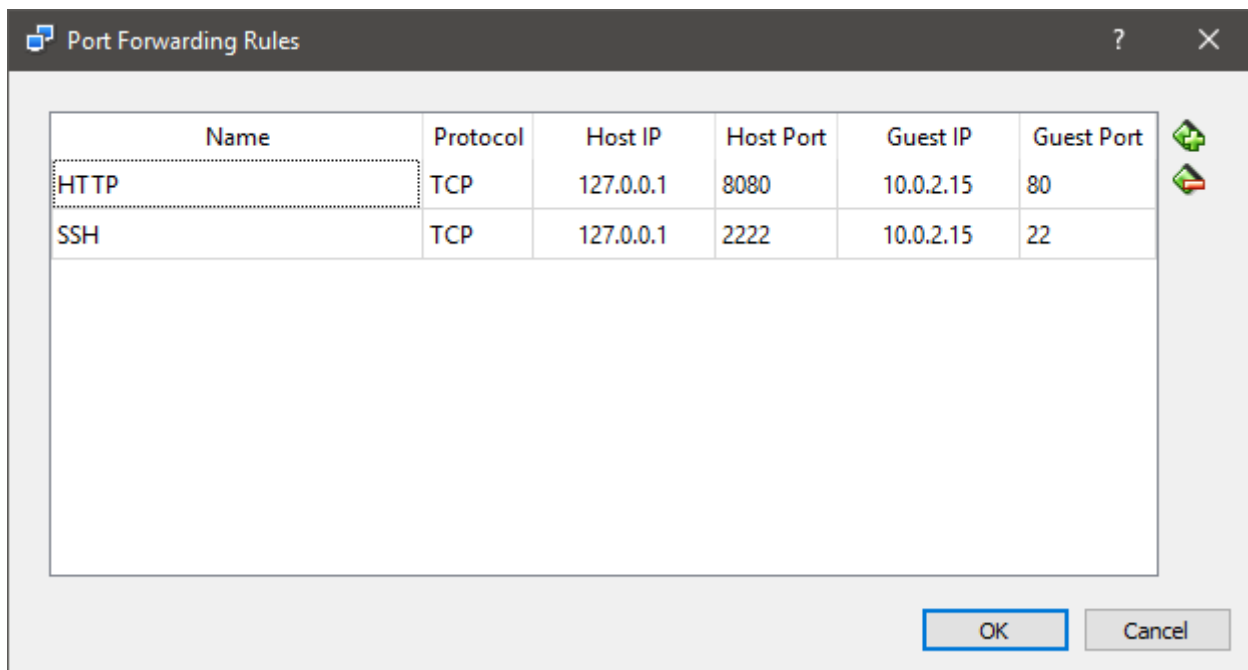


You should now be able to connect to your Virtual Machine via this command: `ssh -p 2222 USER@127.0.0.1` where `user` is the name of the user you created:

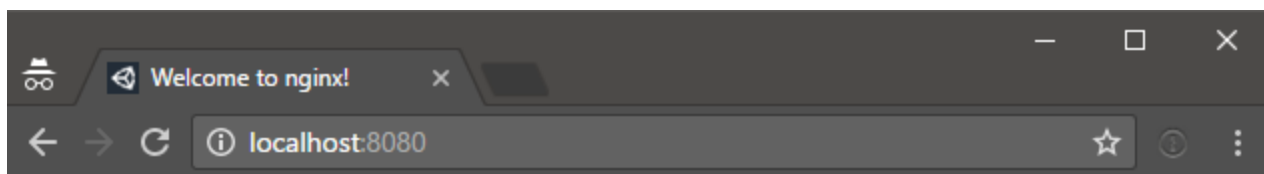


More port forwarding

You can forward any port from inside your VM this way. For example, if you running nginx as a webhost on port 80, you can forward it to your host OS to view the page it is hosting. Since port 80 would already be used by your main OS, you would use another unused port, such as port 8080:



Which you can then view in your host OS' browser as follows:



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

