张向宇 18281059 安全1801

- 一、安装和配置Docker
  - 1）安装Docker
  - 2）配置阿里云加速
  - 3）配置Docker图形化界面 portainer
  - 4）常用配置和工具命令
  - 5）配置Ubuntu Docker
- 二、安装和配置Telnet
- 三、安装和配置VNC
- 四、安装和配置内网穿透服务
  - 1）购买并配置具有公网的VPS
  - 2）在VPS部署FRP服务
- 五、编写并测试WWW服务器
  - 1）编写代码
  - 2）本地测试
  - 3）使用内网穿透变成公网可访问服务器
- 六、遇到的问题和解决方案
  - 1）Docker daemon socket权限不足
  - 2）Docker中Ubuntu更换清华源镜像信任出错

一、安装和配置Docker

脚本安装是最推荐的方式，只需要输入下面的命令，等待自动安装好即可

1）安装Docker

先安装`curl`工具

```
sudo apt install curl
```

然后获取脚本并通过管道工具直接执行

```
sudo curl -sSL https://get.docker.com | sh
```

然后测试Docker

```
sudo docker run hello-world
```

获取到如下信息则说明Docker安装成功

```
zxy@ubuntu:~/Desktop/QTL/ZXYCalc/ZXY_Caculator$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
0e03bdcc26d7: Pull complete
Digest: sha256:d58e752213a51785838f9eed2b7a498ffa1cb3aa7f946dda11af39286c3db9a9
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.
```

```
To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/
```

2）配置阿里云加速

- 进入阿里云镜像加速页面：https://cr.console.aliyun.com/#/accelerator
- 修改/etc/docker/daemon.json文件配置，没有则新建：

  -
    ```
    {
      "registry-mirrors": ["加速器地址"]
    }
    ```
- 重启daemon：`systemctl daemon-reload`
- 重启docker服务：`systemctl restart docker`

3）配置Docker图形化界面 portainer

- 下载portainer：`sudo  docker pull portainer/portainer`
- 创建portainer容器：`sudo docker volume create portainer_data`
- 运行portainer：`sudo docker run -d -p 9000:9000 --name portainer --restart always -v /var/run/docker.sock:/var/run/docker.sock -v portainer_data:/data portainer/portainer`
- 之后查询本机IP地址，通过URL = IP：9000 进入图形化管理界面

4）常用配置和工具命令

```
#查看 Docker 版本
docker -v
sudo docker pull 仓库/镜像:版本（留空的话默认为 latest）
sudo docker run 加参数，用来创建容器
#查看运行容器
sudo docker ps
#查看所有下载的镜像
sudo docker images
#进入容器终端
sudo docker exec -i -t ha /bin/bash
#实时查看10行的 ha 日志
sudo docker logs -f -t --tail 10 ha
#重启 systemctl 守护进程
sudo systemctl daemon-reload
```

```
#设置 Docker 开机启动
sudo systemctl enable docker
#开启 Docker 服务
sudo systemctl start docker
#搜索镜像
docker search [option] keyword
```

5）配置Ubuntu Docker

- 搜索镜像

  -
    ```
    zxy@ubuntu:/etc/docker$ sudo docker search ubuntu
    [sudo] password for zxy:
    NAME                                                      DESCRIPTION
                                      STARS               OFFICIAL
       AUTOMATED
    ubuntu                                                    Ubuntu is a
    Debian-based Linux operating sys…   11089               [OK]
    ```
- 下载镜像

  -
    ```
    zxy@ubuntu:/etc/docker$ docker pull ubuntu
    Using default tag: latest
    latest: Pulling from library/ubuntu
    692c352adcf2: Pull complete
    97058a342707: Pull complete
    2821b8e766f4: Pull complete
    4e643cc37772: Pull complete
    Digest:
    sha256:55cd38b70425947db71112eb5dddfa3aa3e3ce307754a3df2269069d2278ce47
    Status: Downloaded newer image for ubuntu:latest
    docker.io/library/ubuntu:latest
    ```
- 运行容器

  -
    ```
    zxy@ubuntu:/etc/docker$ docker images
    REPOSITORY            TAG                 IMAGE ID            CREATED
              SIZE
    ubuntu                latest              adafef2e596e        27 hours
    ago          73.9MB
    portainer/portainer   latest              cd645f5a4769        5 weeks
    ago           79.1MB
    hello-world           latest              bf756fb1ae65        6 months
    ago          13.3kB
    zxy@ubuntu:/etc/docker$ docker run -it ubuntu /bin/bash
    root@d880c4a6adf1:/#
    ```

二、安装和配置Telnet

- 安装客户端：`sudo apt install telnet`
- 安装服务程序：`sudo apt-get install xinetd`
- 测试命令：`netstat -a | grep telnet`
  - 若有输出则安装完成：`tcp6       0      0 [::]:telnet             [::]:*                  LISTEN`

- 重启机器或重启网络服务：`sudo /etc/init.d/xinetd restart`
- 查看本虚拟机IP地址

  -
    ```
    zxy@ubuntu:/etc/docker$ ifconfig
    docker0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
            inet 172.17.0.1  netmask 255.255.0.0  broadcast 172.17.255.255
            inet6 fe80::42:ccff:fe77:39fc  prefixlen 64  scopeid 0x20<link>
            ether 02:42:cc:77:39:fc  txqueuelen 0  (Ethernet)
            RX packets 10955  bytes 5423270 (5.4 MB)
            RX errors 0  dropped 0  overruns 0  frame 0
            TX packets 15122  bytes 45547780 (45.5 MB)
            TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

    ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
            inet 192.168.159.132  netmask 255.255.255.0  broadcast
    192.168.159.255
            inet6 fe80::1d49:cffb:d73b:cd58  prefixlen 64  scopeid
    0x20<link>
            ether 00:0c:29:c9:48:cd  txqueuelen 1000  (Ethernet)
            RX packets 1113264  bytes 1535931210 (1.5 GB)
            RX errors 0  dropped 0  overruns 0  frame 0
            TX packets 310101  bytes 26061308 (26.0 MB)
            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 6956  bytes 779881 (779.8 KB)
            RX errors 0  dropped 0  overruns 0  frame 0
            TX packets 6956  bytes 779881 (779.8 KB)
            TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

    veth7f64c72: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
            inet6 fe80::9421:8eff:fe47:ad45  prefixlen 64  scopeid
    0x20<link>
            ether 96:21:8e:47:ad:45  txqueuelen 0  (Ethernet)
            RX packets 1398  bytes 5024914 (5.0 MB)
            RX errors 0  dropped 0  overruns 0  frame 0
            TX packets 2747  bytes 691877 (691.8 KB)
            TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
    ```
- Windows下测试Telnet成功：
  - `telnet 192.168.159.132`

  -
    ```
    Ubuntu 20.04 LTS
    ubuntu login: zxy

    Password:

    Welcome to Ubuntu 20.04 LTS (GNU/Linux 5.4.0-40-generic x86_64)
    ```

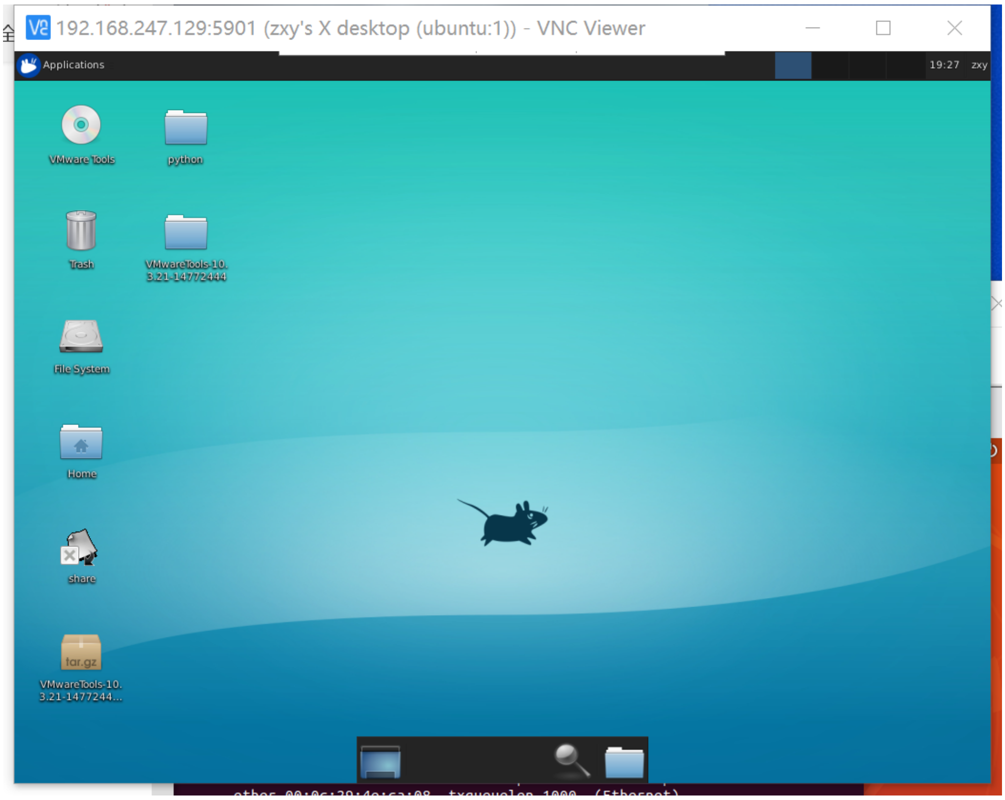
三、安装和配置VNC

- 因为貌似Ubuntu的桌面环境Gnome与VNC存在兼容性问题，安装桌面环境Xfce

    - `sudo apt-get install xfce4 xfce4-goodies`
- 安装VNC服务端
    - `sudo apt-get install tightvncserver`
- 设置VNC连接密码设置以及生成配置文件
    - 首先执行`vncserver`命令来设置VNC连接密码以及生成VNC配置文件

    -
      ```
      You will require a password to access your desktops.
      Password:
      Verify:
      ```
    - 配置文件在下面目录里面：`/home/your_username/.vnc/`
    - 第一次运行`vncserver`命令会自动启动VNC实例，分配到`:1`上，对应端口为5901 (端口5901=5900+1，如果是`:2`，则端口为5902，以此类推)。由于要配置VNC，所以先要关闭VNC实例
        - `vncserver -kill :1`
- 配置VNC
    - 要配置的文件为`xstartup`，该文件在`$HOME/.vnc`里面
    - 首先备份原始配置文件：`mv ~/.vnc/xstartup ~/.vnc/xstartup.bak`
    - 然后创建新的配置文件：`touch ~/.vnc/xstartup`
    - 编辑该文件，添加以下内容：

        -
          ```
          #!/bin/sh

          # Uncomment the following two lines for normal desktop:
          unset SESSION_MANAGER
          unset DBUS_SESSION_BUS_ADDRESS
          # exec /etc/X11/xinit/xinitrc

          [ -x /etc/vnc/xstartup ] && exec /etc/vnc/xstartup
          [ -r $HOME/.Xresources ] && xrdb $HOME/.Xresources
          xsetroot -solid grey
          vncconfig -iconic &
          x-terminal-emulator -geometry 80x24+10+10 -ls -title "$VNCDESKTOP Desktop" &
          #x-window-manager &
          #gnome-session &
          x-session-manager & xfdesktop & xfce4-panel &
          xfce4-menu-plugin &
          xfsettingsd &
          xfconfd &
          xfwm4 &
          ```
    - 为了保证VNC配置文件能够生效，赋予该文件执行权限：`chmod +x ~/.vnc/xstartup`
- VNC连接
    - 启动VNC实例：`vncserver`
    - 查看端口开启情况：`ss -ltn`
    - 使用VNC Viewer连接成功

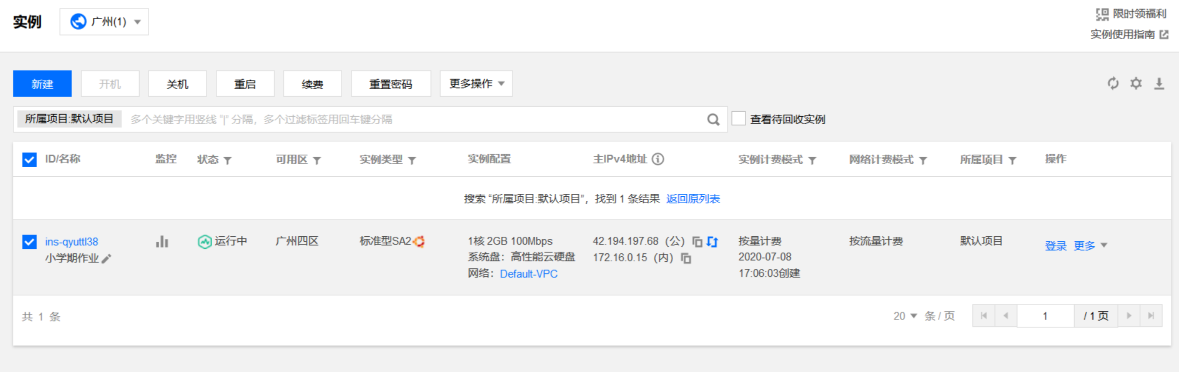

192.168.247.129:5901 (zxy's X desktop (ubuntu:1)) - VNC Viewer
Applications
19:27
zxy
VMware Tools
python
Trash
VMwareTools-10.3.21-14772444
File System
Home
share
tar.gz
VMwareTools-10.3.21-1477244...
ether 00:0c:29:4e:ca:08 txqueuelen 1000 (Ethernet)

四、安装和配置内网穿透服务

由于我家使用的互联网运营商是中国电信，没有直接给我分配公网IP，我需要进行内网穿透才能让配合的同学访问我的主机，我在这里选用的内网穿透软件是**FRP**

1）购买并配置具有公网的VPS

这里我选择腾讯云的广州区EPYC云服务器



ID/名称	监控	状态	可用区	实例类型	实例配置	主IPv4地址	实例计费模式	网络计费模式	所属项目	操作
搜索 "所属项目:默认项目"，找到 1 条结果 返回原列表										
☑ ins-qyuttl38 小学期作业		运行中	广州四区	标准型SA2	1核 2GB 100Mbps 系统盘：高性能云硬盘 网络：Default-VPC	42.194.197.68（公） 172.16.0.15（内）	按量计费 2020-07-08 17:06:03创建	按流量计费	默认项目	登录 更多

共 1 条　20 条 / 页　1 / 1 页

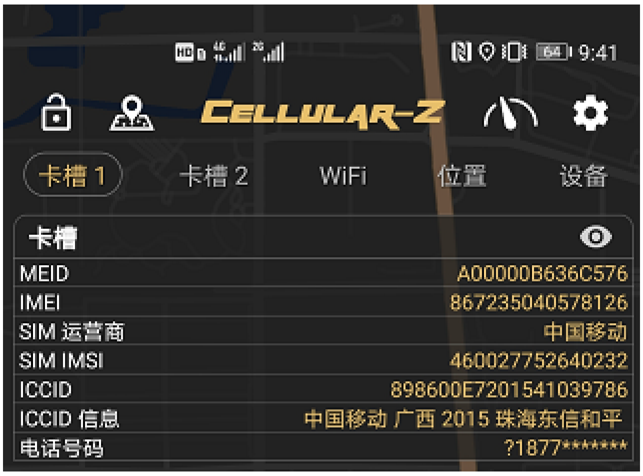
2）在VPS部署FRP服务

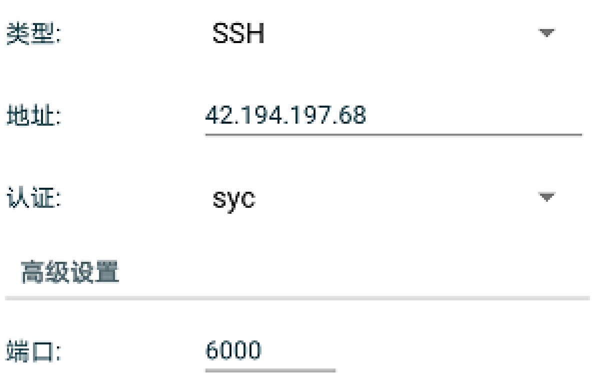
- 下载地址：https://github.com/fatedier/frp/releases
- 配置服务端（公网服务器），首先删掉frpc、frpc.ini两个文件，然后再进行配置，vi ./frps.ini

    -
      ```
      Congratulations, frps install completed!
      ================================================
      You Server IP      : 42.194.197.68
      Bind port          : 5443
      KCP support        : true
      vhost http port    : 80
      vhost https port   : 443
      Dashboard port     : 6443
      token              : nA9rqFk76s8lusqH
      subdomain_host     : 42.194.197.68
      ```

```
tcp_mux             : true
Max Pool count      : 50
Log level           : info
Log max days        : 3
Log file            : enable
================================================
frps Dashboard      : http://42.194.197.68:6443/
Dashboard user      : admin
Dashboard password  : sigh987yu
================================================
```

- 保存然后启动服务./frps -c ./frps.ini，这是前台启动，后台启动命令为nohup ./frps -c ./frps.ini &
- 配置客户端（内网服务器），首先删掉frps、frps.ini两个文件,然后再进行配置，vi ./frpc.ini

  -
    ```
    [common]
    server_addr = 42.194.197.68
    server_port = 5443
    token = nA9rqFk76s8lusqH

    [ssh]
    type = tcp
    local_ip = 127.0.0.1
    local_port = 22
    remote_port = 6000
    ```
- 测试SSH服务
  - 测试环境：手机目前使用的是移动网络，与虚拟机不在同一网段内，属于外网环境
    - 

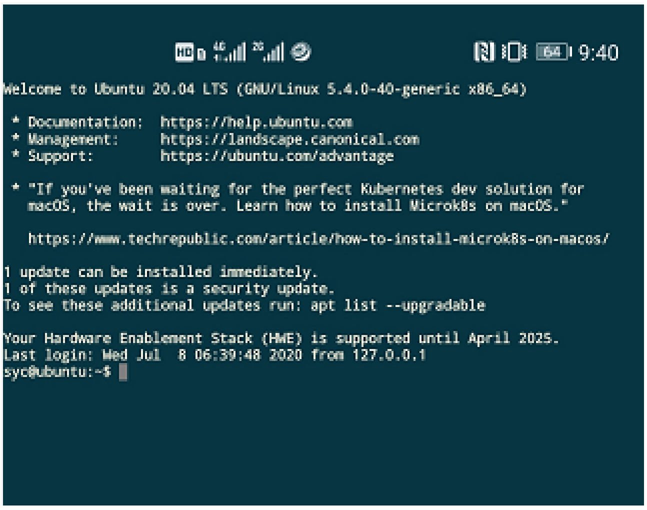
MEID	A00000B636C576
IMEI	867235040578126
SIM 运营商	中国移动
SIM IMSI	460027752640232
ICCID	898600E7201541039786
ICCID 信息	中国移动 广西 2015 珠海东信和平
电话号码	?1877*******
  - 配置SSH参数，使用的IP是中间代理服务器的IP，端口是之前设置的映射端口
    - 
      类型: SSH

      地址: 42.194.197.68

      认证: syc

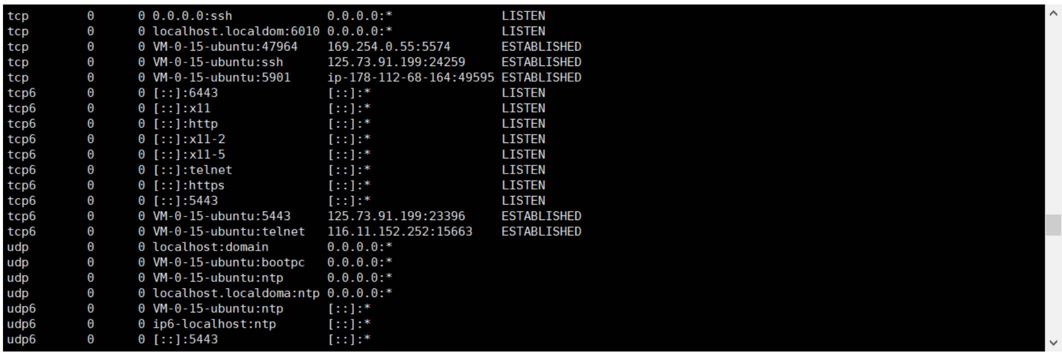
      高级设置

      端口: 6000
  - 连接成功，说明内网穿透实验成功

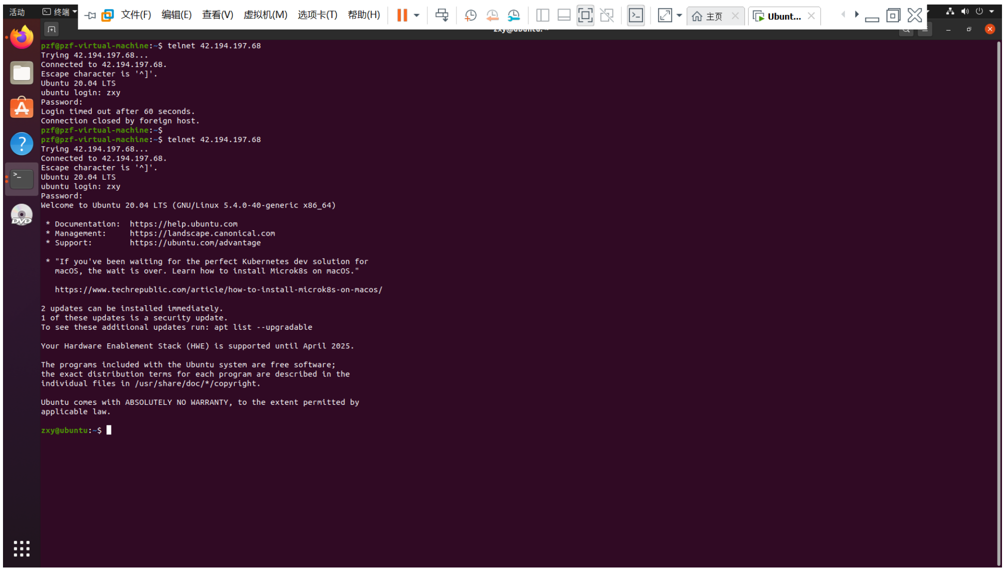
- 

```
Welcome to Ubuntu 20.04 LTS (GNU/Linux 5.4.0-40-generic x86_64)

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

 * "If you've been waiting for the perfect Kubernetes dev solution for
   macOS, the wait is over. Learn how to install Microk8s on macOS."

   https://www.techrepublic.com/article/how-to-install-microk8s-on-macos/

1 update can be installed immediately.
1 of these updates is a security update.
To see these additional updates run: apt list --upgradable

Your Hardware Enablement Stack (HWE) is supported until April 2025.
Last login: Wed Jul  8 06:39:48 2020 from 127.0.0.1
syc@ubuntu:~$ 
```

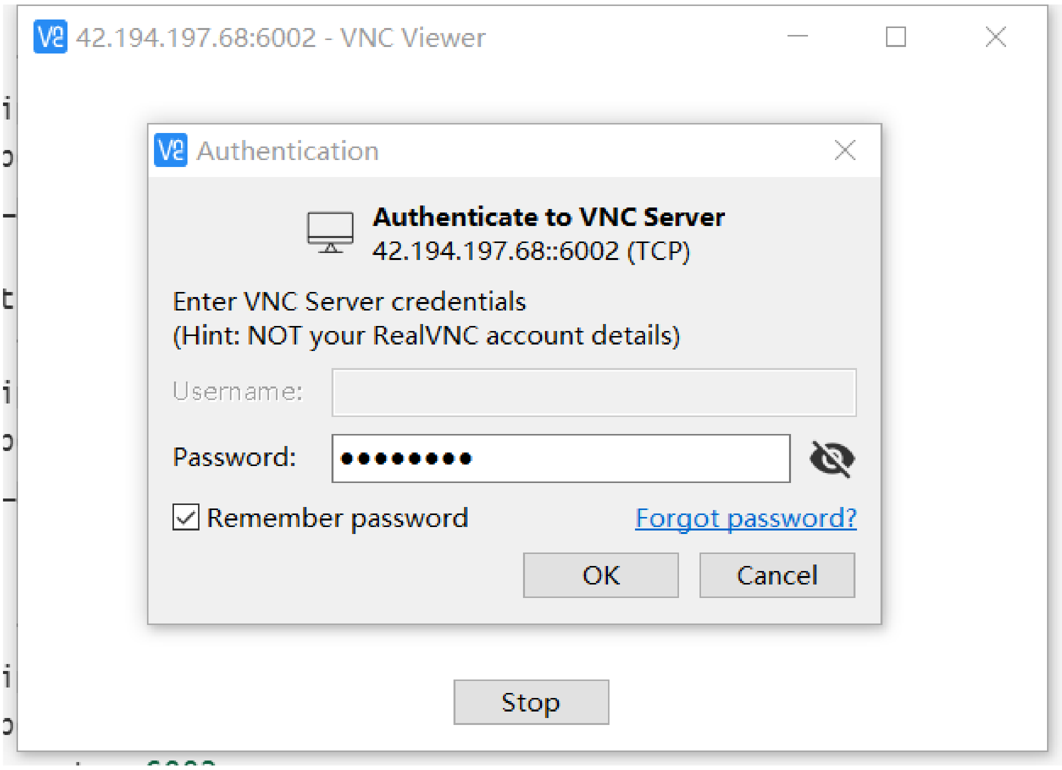
- 配置Telnet和VNC端口映射服务
  -
    ```
    [common]
    server_addr = 42.194.197.68
    server_port = 5443
    token = nA9rqFk76s8lusqH

    [ssh]
    type = tcp
    local_ip = 127.0.0.1
    local_port = 22
    remote_port = 6000

    [telnet]
    type = tcp
    local_ip = 127.0.0.1
    local_port = 23
    remote_port = 6003

    [vnc]
    type = tcp
    local_ip = 127.0.0.1
    local_port = 5901
    remote_port = 6002
    ```
- 与同学一同测试Telnet服务
  - `PS C:\Users\syc> telnet 42.194.197.68:6003`
  -
    ```
    Ubuntu 20.04 LTS
    ubuntu login: zxy

    Password:

    Welcome to Ubuntu 20.04 LTS (GNU/Linux 5.4.0-40-generic x86_64)
    ```
- 使用`netstat -a`查看连接情况
  -



```
tcp        0      0 0.0.0.0:ssh             0.0.0.0:*               LISTEN
tcp        0      0 localhost.localdom:6010 0.0.0.0:*               LISTEN
tcp        0      0 VM-0-15-ubuntu:47964    169.254.0.55:5574       ESTABLISHED
tcp        0      0 VM-0-15-ubuntu:ssh      125.73.91.199:24259     ESTABLISHED
tcp        0      0 VM-0-15-ubuntu:5901     ip-178-112-68-164:49595 ESTABLISHED
tcp6       0      0 [::]:6443               [::]:*                  LISTEN
tcp6       0      0 [::]:x11                [::]:*                  LISTEN
tcp6       0      0 [::]:http               [::]:*                  LISTEN
tcp6       0      0 [::]:x11-2              [::]:*                  LISTEN
tcp6       0      0 [::]:x11-5              [::]:*                  LISTEN
tcp6       0      0 [::]:telnet             [::]:*                  LISTEN
tcp6       0      0 [::]:https              [::]:*                  LISTEN
tcp6       0      0 [::]:5443               [::]:*                  LISTEN
tcp6       0      0 VM-0-15-ubuntu:5443     125.73.91.199:23396     ESTABLISHED
tcp6       0      0 VM-0-15-ubuntu:telnet   116.11.152.252:15663    ESTABLISHED
udp        0      0 localhost:domain        0.0.0.0:*
udp        0      0 VM-0-15-ubuntu:bootpc   0.0.0.0:*
udp        0      0 VM-0-15-ubuntu:ntp      0.0.0.0:*
udp        0      0 localhost.localdoma:ntp 0.0.0.0:*
udp6       0      0 VM-0-15-ubuntu:ntp      [::]:*
udp6       0      0 ip6-localhost:ntp       [::]:*
udp6       0      0 [::]:5443               [::]:*
```

- 获得同学的IP：116.11.152.252，经过验证同样是同学的IP
- 同学终端切换成功：
  - 

```
pzf@pzf-virtual-machine:~$ telnet 42.194.197.68
Trying 42.194.197.68...
Connected to 42.194.197.68.
Escape character is '^]'.
Ubuntu 20.04 LTS
ubuntu login: zxy
Password:
Login timed out after 60 seconds.
Connection closed by foreign host.
pzf@pzf-virtual-machine:~$
pzf@pzf-virtual-machine:~$ telnet 42.194.197.68
Trying 42.194.197.68...
Connected to 42.194.197.68.
Escape character is '^]'.
Ubuntu 20.04 LTS
ubuntu login: zxy
Password:
Welcome to Ubuntu 20.04 LTS (GNU/Linux 5.4.0-40-generic x86_64)

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

 * "If you've been waiting for the perfect Kubernetes dev solution for
   macOS, the wait is over. Learn how to install Microk8s on macOS."

   https://www.techrepublic.com/article/how-to-install-microk8s-on-macos/

2 updates can be installed immediately.
1 of these updates is a security update.
To see these additional updates run: apt list --upgradable

Your Hardware Enablement Stack (HWE) is supported until April 2025.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

zxy@ubuntu:~$
```

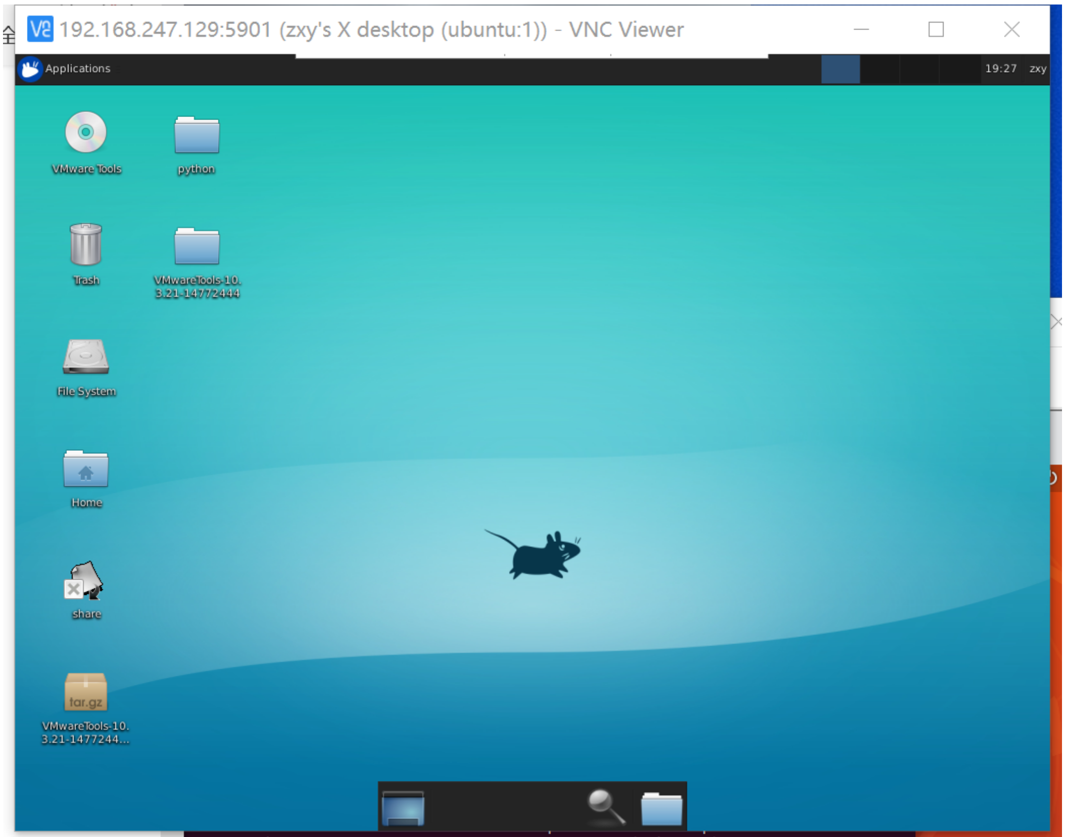
- 与同学一同测试VNC服务
  - 

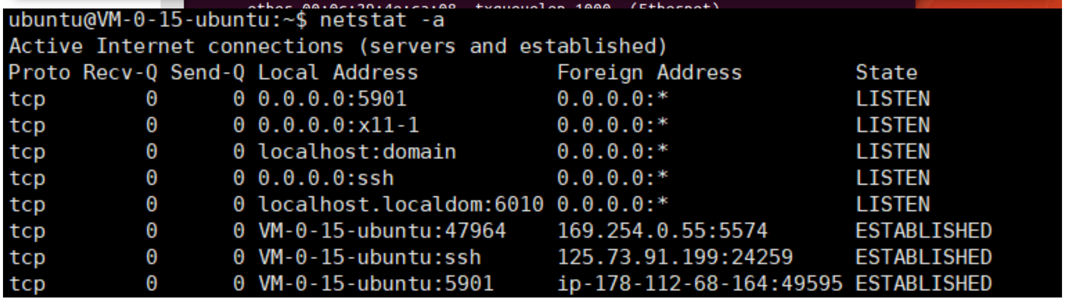
    Authenticate to VNC Server
    42.194.197.68::6002 (TCP)

    Enter VNC Server credentials
    (Hint: NOT your RealVNC account details)

    Username:
    Password: ●●●●●●●●

    ☑ Remember password
    Forgot password?

- 
192.168.247.129:5901 (zxy's X desktop (ubuntu:1)) - VNC Viewer
Applications
19:27
zxy
VMware Tools
python
Trash
VMwareTools-10.3.21-14772444
File System
Home
share
tar.gz
VMwareTools-10.3.21-1477244...

- 

```
ubuntu@VM-0-15-ubuntu:~$ netstat -a
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:5901            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:x11-1           0.0.0.0:*               LISTEN
tcp        0      0 localhost:domain        0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:ssh             0.0.0.0:*               LISTEN
tcp        0      0 localhost.localdom:6010 0.0.0.0:*               LISTEN
tcp        0      0 VM-0-15-ubuntu:47964    169.254.0.55:5574       ESTABLISHED
tcp        0      0 VM-0-15-ubuntu:ssh      125.73.91.199:24259     ESTABLISHED
tcp        0      0 VM-0-15-ubuntu:5901     ip-178-112-68-164:49595 ESTABLISHED
```

五、编写并测试WWW服务器

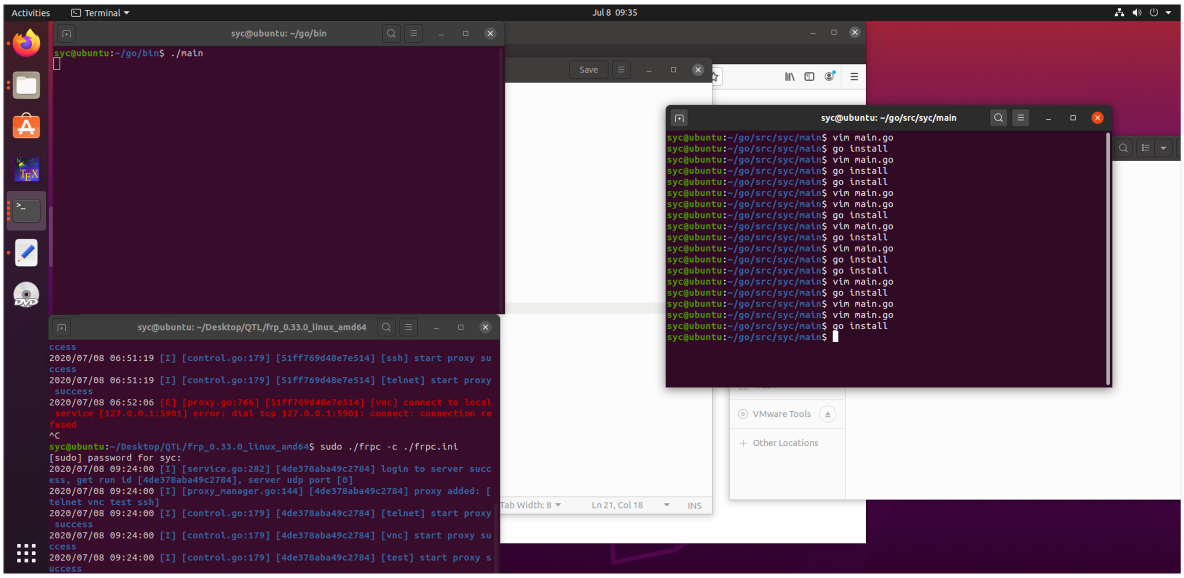
1）编写代码

我使用的是用Go语言编写的轻量级HTTP服务程序，源代码：

```
package main

import (
    "fmt"
    "net/http"
    "path/filepath"
)

func main() {
    p, _ := filepath.Abs(filepath.Dir("./Resource"))//是获取当前可执行程序所在的绝对
路径
    http.Handle("/", http.FileServer(http.Dir(p)))//是开启一个文件服务器，使用当前可执
行文件所在的路径
    err := http.ListenAndServe(":8080", nil)//是监听8080端口并开启文件服务器
    if err != nil {
        fmt.Println(err)
    }
}
```

2）本地测试

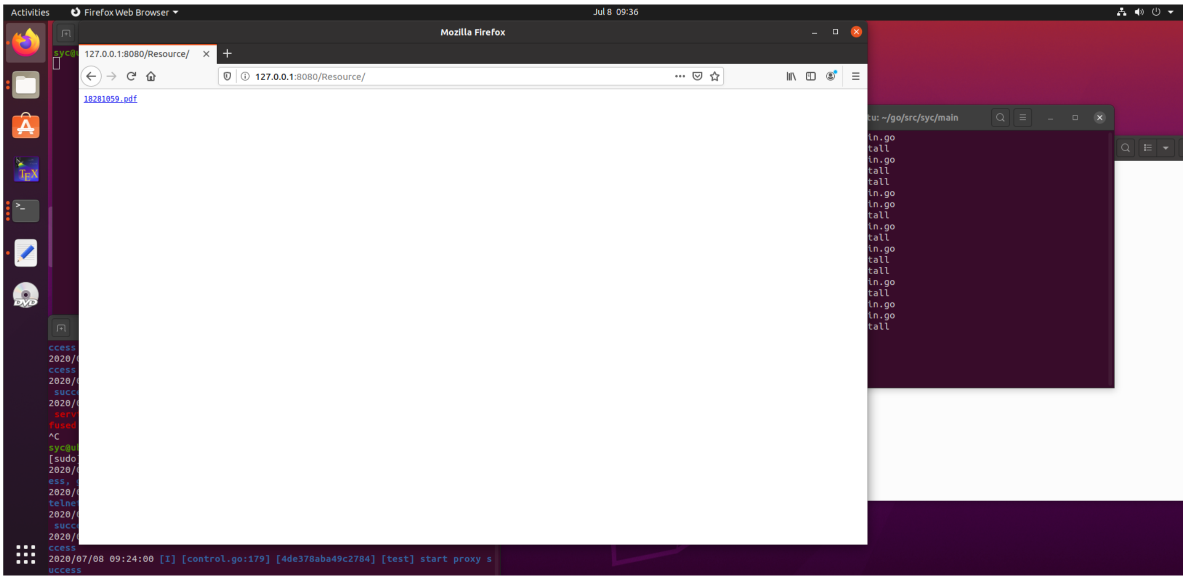
在Linux虚拟机运行本地测试：



```
syc@ubuntu:~/go/bin$ ./main
```

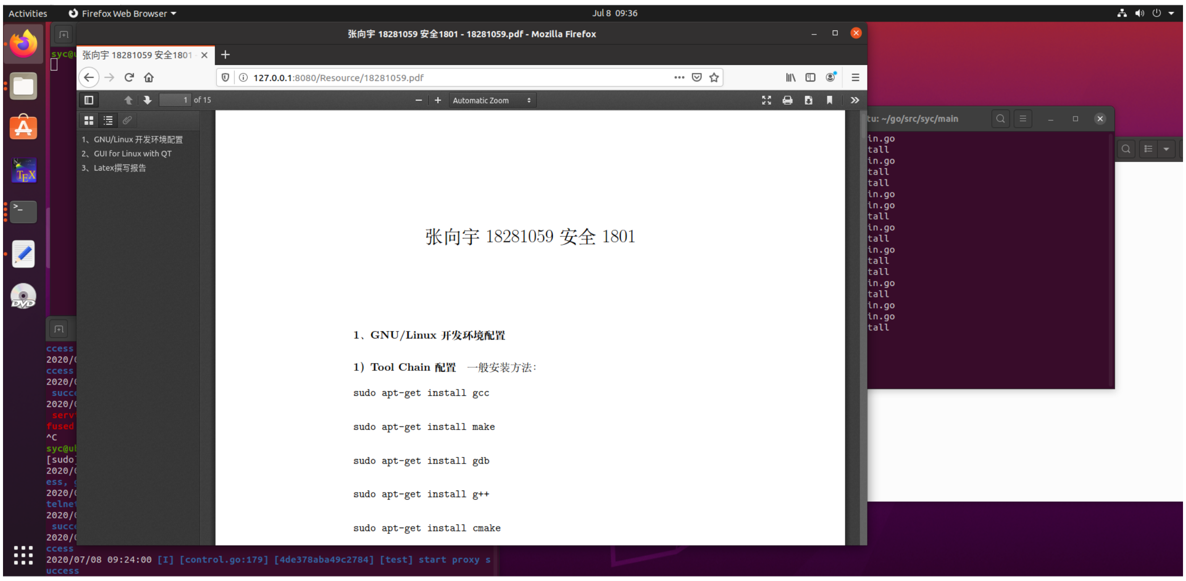
```
ccess
2020/07/08 06:51:19 [I] [control.go:179] [51ff769d48e7e514] [ssh] start proxy su
ccess
2020/07/08 06:51:19 [I] [control.go:179] [51ff769d48e7e514] [telnet] start proxy
 success
2020/07/08 06:52:06 [E] [proxy.go:766] [51ff769d48e7e514] [vnc] connect to local
 service [127.0.0.1:5901] error: dial tcp 127.0.0.1:5901: connect: connection re
fused
^C
syc@ubuntu:~/Desktop/QTL/frp_0.33.0_linux_amd64$ sudo ./frpc -c ./frpc.ini
[sudo] password for syc:
2020/07/08 09:24:00 [I] [service.go:282] [4de378aba49c2784] login to server succ
ess, get run id [4de378aba49c2784], server udp port [0]
2020/07/08 09:24:00 [I] [proxy_manager.go:144] [4de378aba49c2784] proxy added: [
telnet vnc test ssh]
2020/07/08 09:24:00 [I] [control.go:179] [4de378aba49c2784] [telnet] start proxy
 success
2020/07/08 09:24:00 [I] [control.go:179] [4de378aba49c2784] [vnc] start proxy su
ccess
2020/07/08 09:24:00 [I] [control.go:179] [4de378aba49c2784] [test] start proxy s
uccess
```

```
syc@ubuntu:~/go/src/syc/main$ vim main.go
syc@ubuntu:~/go/src/syc/main$ go install
syc@ubuntu:~/go/src/syc/main$ vim main.go
syc@ubuntu:~/go/src/syc/main$ go install
syc@ubuntu:~/go/src/syc/main$ go install
syc@ubuntu:~/go/src/syc/main$ vim main.go
syc@ubuntu:~/go/src/syc/main$ vim main.go
syc@ubuntu:~/go/src/syc/main$ go install
syc@ubuntu:~/go/src/syc/main$ vim main.go
syc@ubuntu:~/go/src/syc/main$ go install
syc@ubuntu:~/go/src/syc/main$ vim main.go
syc@ubuntu:~/go/src/syc/main$ go install
syc@ubuntu:~/go/src/syc/main$ go install
syc@ubuntu:~/go/src/syc/main$ vim main.go
syc@ubuntu:~/go/src/syc/main$ go install
syc@ubuntu:~/go/src/syc/main$ vim main.go
syc@ubuntu:~/go/src/syc/main$ vim main.go
syc@ubuntu:~/go/src/syc/main$ go install
syc@ubuntu:~/go/src/syc/main$
```



127.0.0.1:8080/Resource/

18281059.pdf



127.0.0.1:8080/Resource/18281059.pdf

张向宇 18281059 安全 1801

1、GNU/Linux 开发环境配置

1）Tool Chain 配置　一般安装方法：

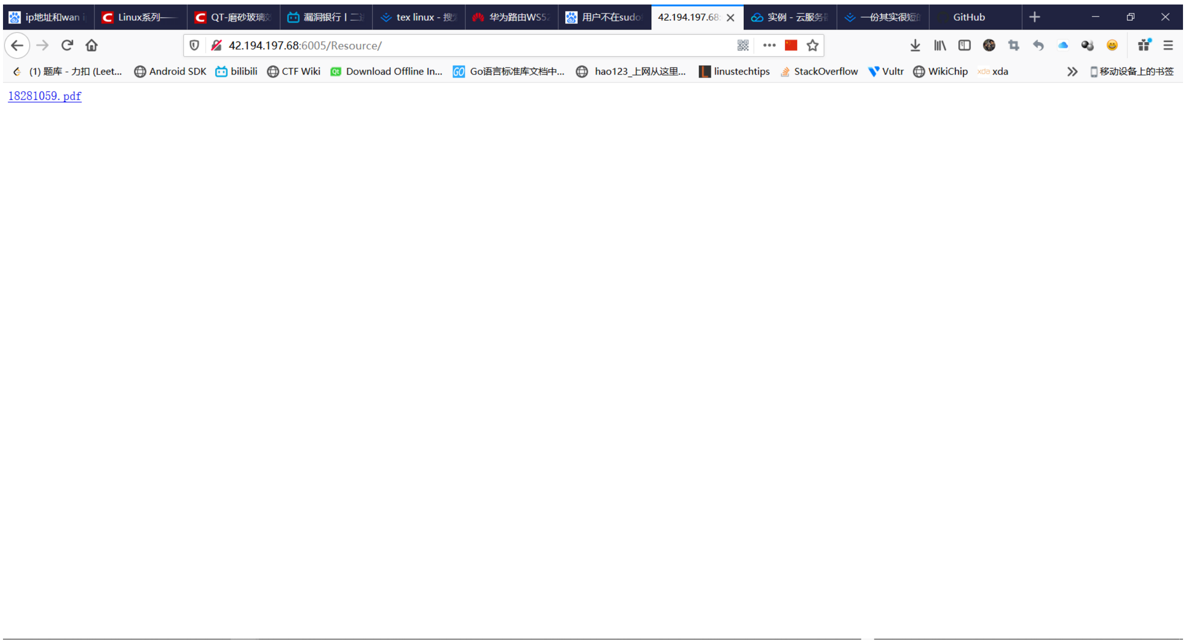
```
sudo apt-get install gcc

sudo apt-get install make

sudo apt-get install gdb

sudo apt-get install g++

sudo apt-get install cmake
```

3）使用内网穿透变成公网可访问服务器

配置FRP文件：

```
[common]
server_addr = 42.194.197.68
server_port = 5443
token = nA9rqFk76s8lusqH

[ssh]
type = tcp
local_ip = 127.0.0.1
local_port = 22
remote_port = 6000

[telnet]
type = tcp
local_ip = 127.0.0.1
local_port = 23
remote_port = 6003

[vnc]
type = tcp
local_ip = 127.0.0.1
local_port = 5901
remote_port = 6002

[MyHTTP]
type = tcp
local_ip = 127.0.0.1
local_port = 8080
remote_port = 6005
```

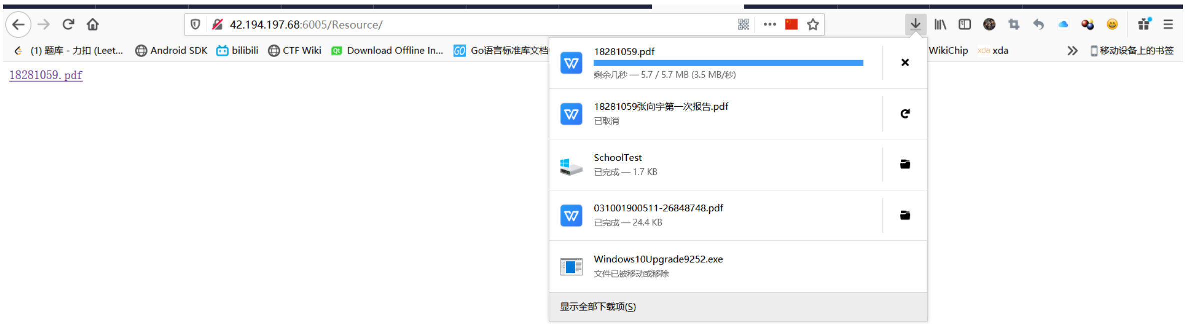
外网成功：



42.194.197.68:6005/Resource/

18281059.pdf

下载成功：



六、遇到的问题和解决方案

1）Docker daemon socket权限不足

运行docker出现以下报错：

```
docker: Got permission denied while trying to connect to the Docker daemon
socket at unix:///var/run/docker.sock: Post
http://%2Fvar%2Frun%2Fdocker.sock/v1.38/containers/create: dial unix
/var/run/docker.sock: connect: permission denied.
```

出现上面问题是因为：

```
Manage Docker as a non-root user
即：管理Docker的不是root用户
```

- 方案一：使用sudo获取管理员权限，运行docker命令
- 方案二：添加docker group组，将用户添加进去

```
zxy@ubuntu:~$ sudo groupadd docker #添加docker用户组
groupadd：“docker”组已存在
zxy@ubuntu:~$ sudo gpasswd -a $USER docker #将登陆用户加入到docker用户组中
正在将用户“syc”加入到“docker”组中
zxy@ubuntu:~$ newgrp docker #更新用户组
zxy@ubuntu:~$ docker ps #测试当前用户是否可以正常使用docker命令
CONTAINER ID        IMAGE               COMMAND             CREATED
STATUS              PORTS               NAMES
```

2）Docker中Ubuntu更换清华源镜像信任出错

There was a problem with the certificates installed in this base image. I managed to fix it by install `ca-certificates`:

```
sudo apt install ca-certificates
```