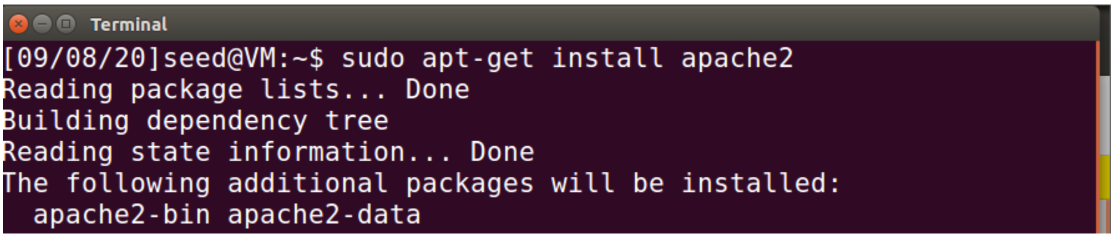
Lab3-report

姓名：王星沣
学号：57118131

实验一 HTTP 基础

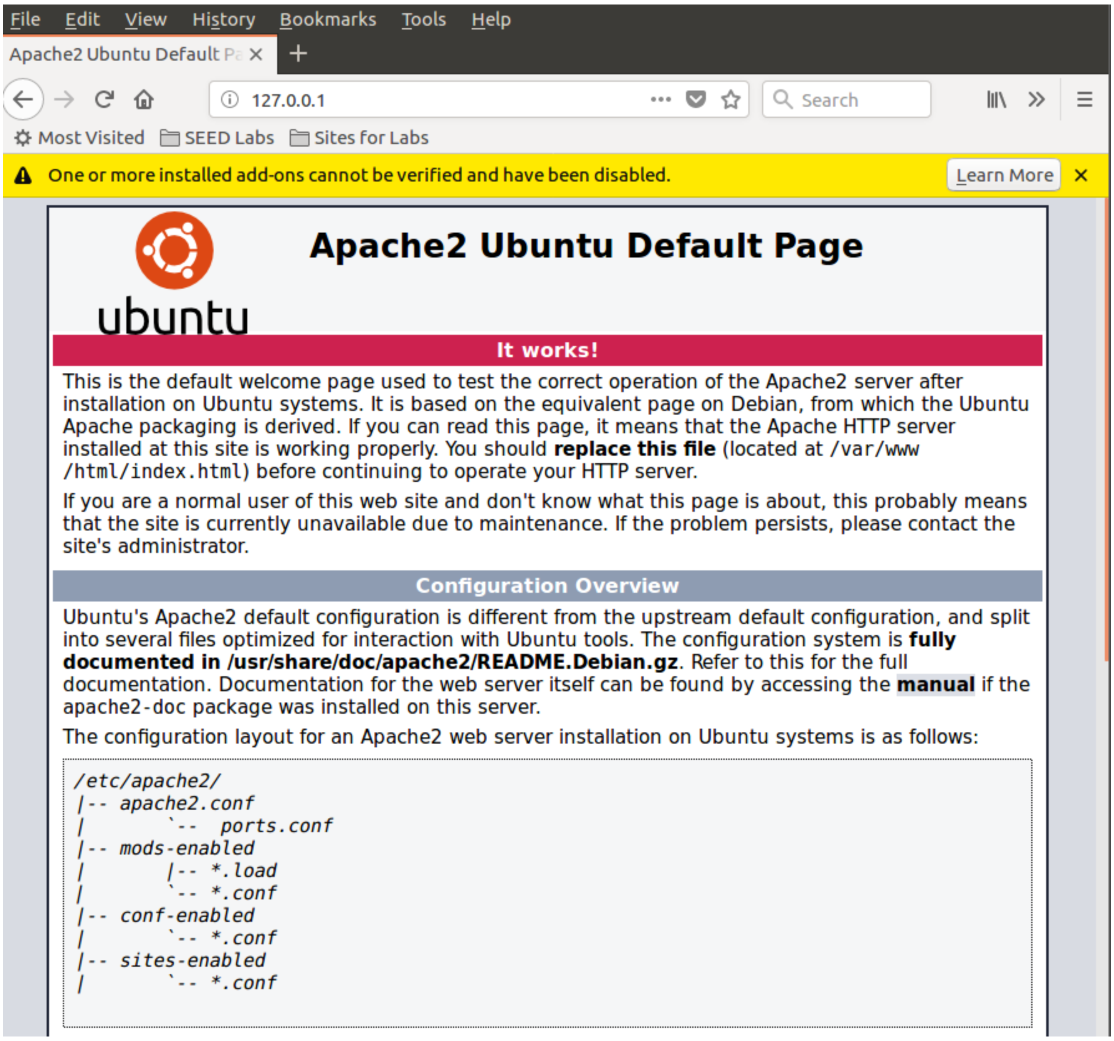
任务一：安装 apache 服务器 并用简单页面验证

步骤 1：在虚拟机中打开 terminal 终端窗口，输入 sudo apt-get install apache2



```
[09/08/20]seed@VM:~$ sudo apt-get install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data
```

步骤 2：Apache 安装完成后，默认的网站根目录是”var/www/html”，在网站根目录路径下有一个 index.html 文件，虚拟机浏览器中输入”127.0.0.1”打开该页面。


File Edit View History Bookmarks Tools Help
Apache2 Ubuntu Default Pa
127.0.0.1
Search
Most Visited
SEED Labs
Sites for Labs
One or more installed add-ons cannot be verified and have been disabled.
Learn More
ubuntu

Apache2 Ubuntu Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at /var/www/html/index.html) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the apache2-doc package was installed on this server.

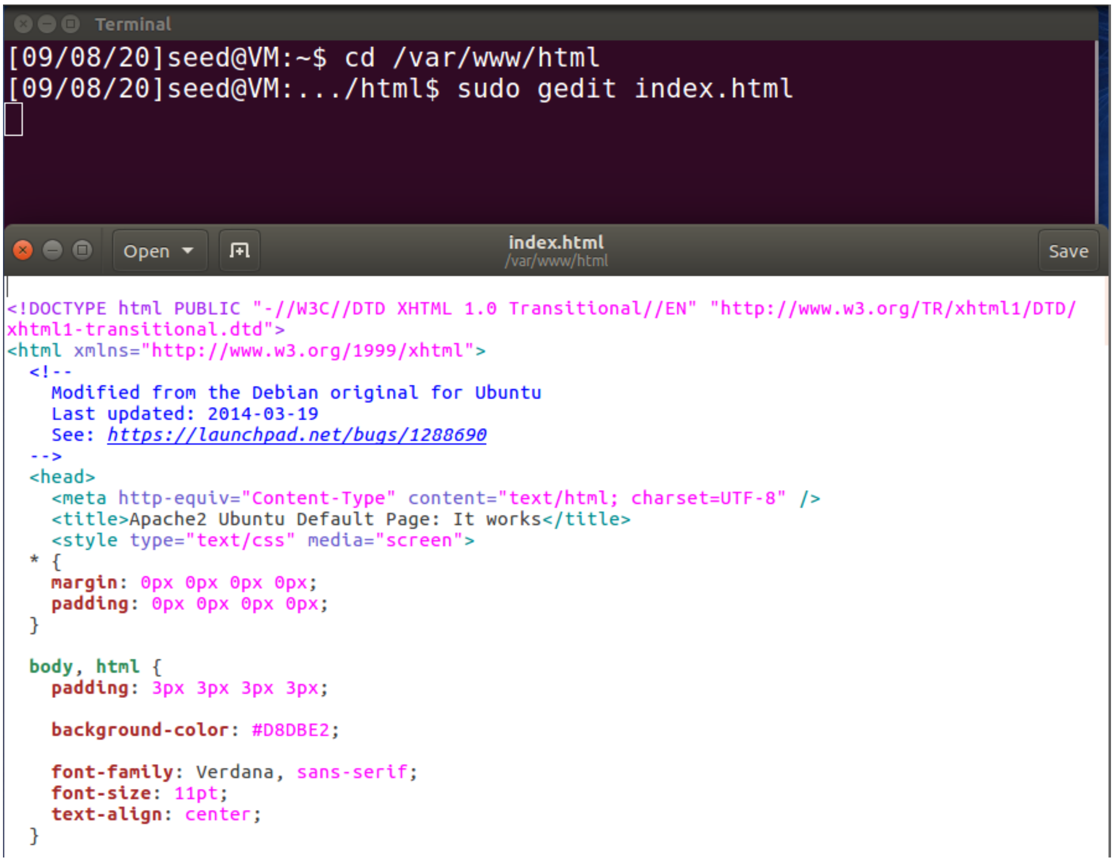
The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|       `--  ports.conf
|-- mods-enabled
|       |-- *.load
|       `-- *.conf
|-- conf-enabled
|       `-- *.conf
|-- sites-enabled
|       `-- *.conf
```

步骤 3:

1. cd /var/www/html

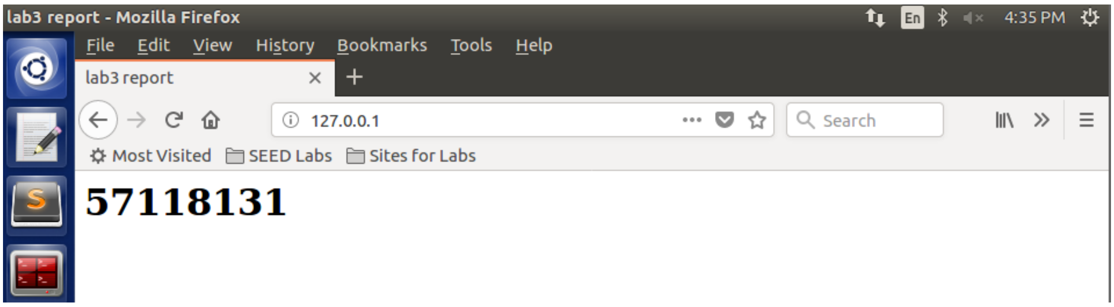
2. 使用 sudo gedit index.html 指令打开 index.html 并进行编写



```
[09/08/20]seed@VM:~$ cd /var/www/html
[09/08/20]seed@VM:.../html$ sudo gedit index.html
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/
xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <!--
    Modified from the Debian original for Ubuntu
    Last updated: 2014-03-19
    See: https://launchpad.net/bugs/1288690
  -->
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>Apache2 Ubuntu Default Page: It works</title>
    <style type="text/css" media="screen">
  * {
    margin: 0px 0px 0px 0px;
    padding: 0px 0px 0px 0px;
  }

  body, html {
    padding: 3px 3px 3px 3px;

    background-color: #D8DBE2;

    font-family: Verdana, sans-serif;
    font-size: 11pt;
    text-align: center;
  }
```

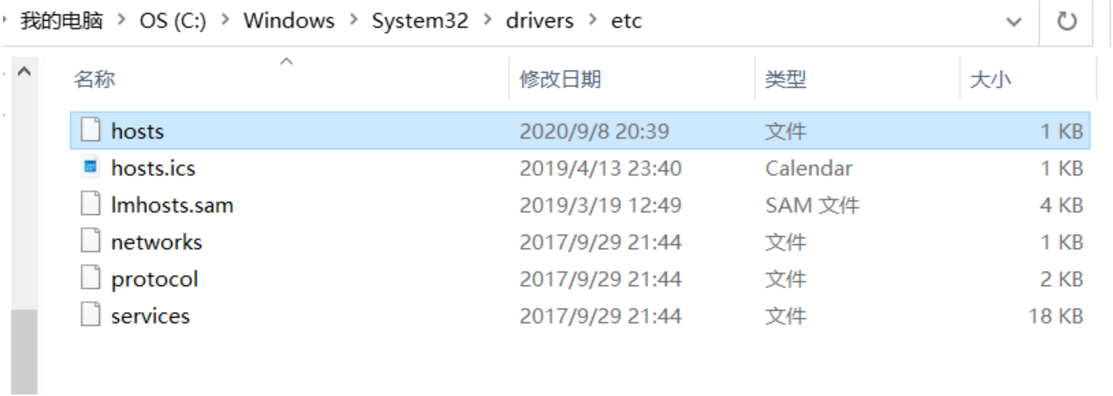
步骤 4：修改后使用浏览器登录 127.0.0.1，页面更改为新主页。



57118131

任务二：通过 host 文件解析名称

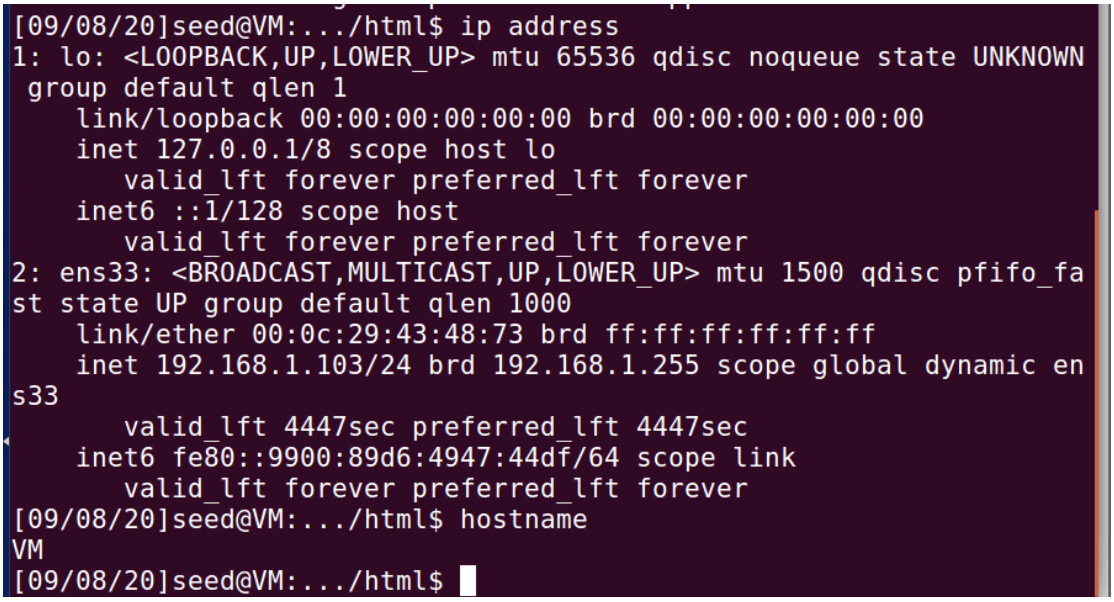
步骤 1： 在 windows 主机中找到 hosts 文件记事本打开，**修改 hosts 文件加入虚拟机 ip 地址与主机名并保存**（C:\Windows\System32\drivers\etc）



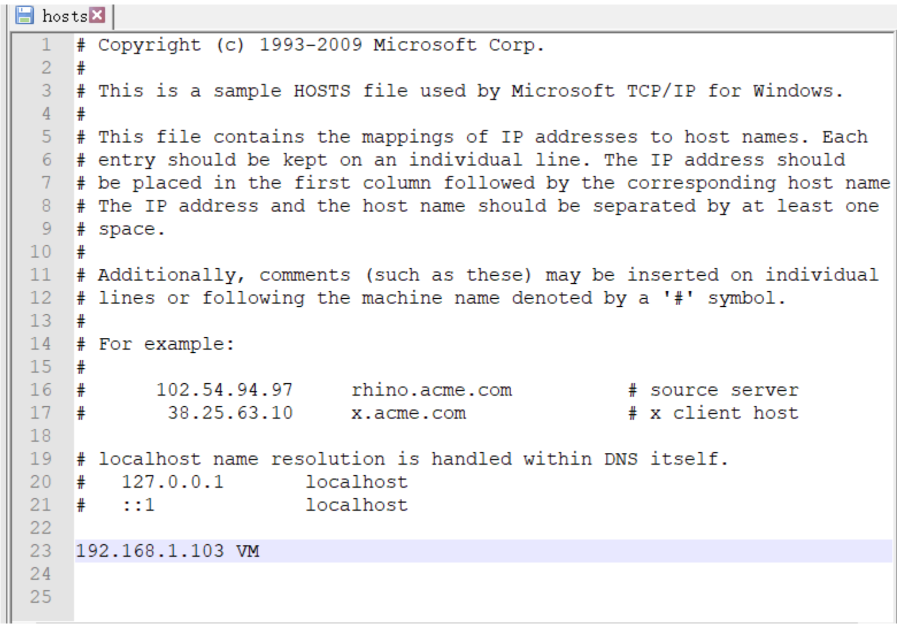
我的电脑 › OS (C:) › Windows › System32 › drivers › etc

名称	修改日期	类型	大小
hosts	2020/9/8 20:39	文件	1 KB
hosts.ics	2019/4/13 23:40	Calendar	1 KB
lmhosts.sam	2019/3/19 12:49	SAM 文件	4 KB
networks	2017/9/29 21:44	文件	1 KB
protocol	2017/9/29 21:44	文件	2 KB
services	2017/9/29 21:44	文件	18 KB

在虚拟机中使用 ip address 命令查看网卡 ip，hostname 查看主机名



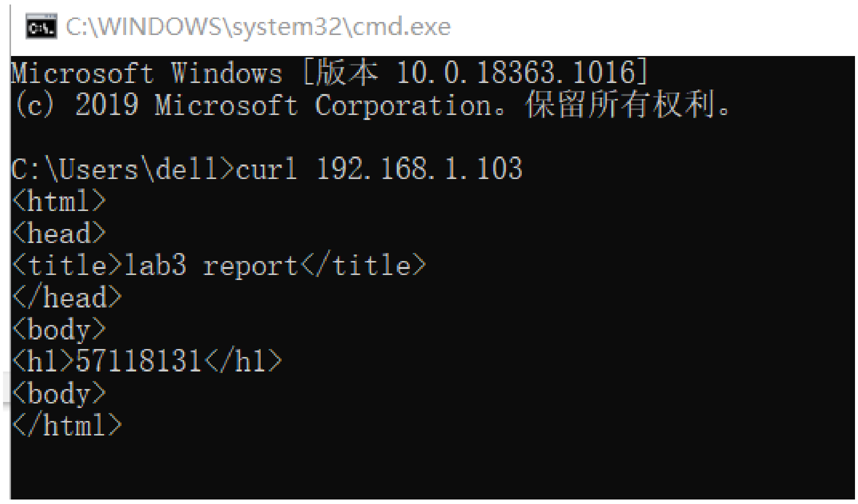
```
[09/08/20]seed@VM:.../html$ ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
 group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fa
st state UP group default qlen 1000
    link/ether 00:0c:29:43:48:73 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.103/24 brd 192.168.1.255 scope global dynamic en
s33
       valid_lft 4447sec preferred_lft 4447sec
    inet6 fe80::9900:89d6:4947:44df/64 scope link
       valid_lft forever preferred_lft forever
[09/08/20]seed@VM:.../html$ hostname
VM
[09/08/20]seed@VM:.../html$ 
```



```
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#      102.54.94.97     rhino.acme.com          # source server
#       38.25.63.10     x.acme.com              # x client host

# localhost name resolution is handled within DNS itself.
#   127.0.0.1       localhost
#   ::1             localhost

192.168.1.103 VM


```

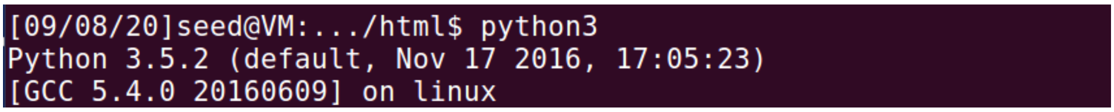
任务三：编写HTTP客户端，使用http库检索站点的主页

步骤1：windows主机中输入curl+虚拟机ip地址可查看编写的index文件内容



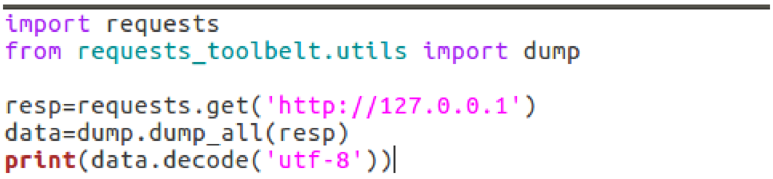
```
Microsoft Windows [版本 10.0.18363.1016]
(c) 2019 Microsoft Corporation。保留所有权利。

C:\Users\dell>curl 192.168.1.103
<html>
<head>
<title>lab3 report</title>
</head>
<body>
<h1>57118131</h1>
<body>
</html>
```

步骤2：虚拟机中输入python3 --version查看虚拟机是否有python3.5

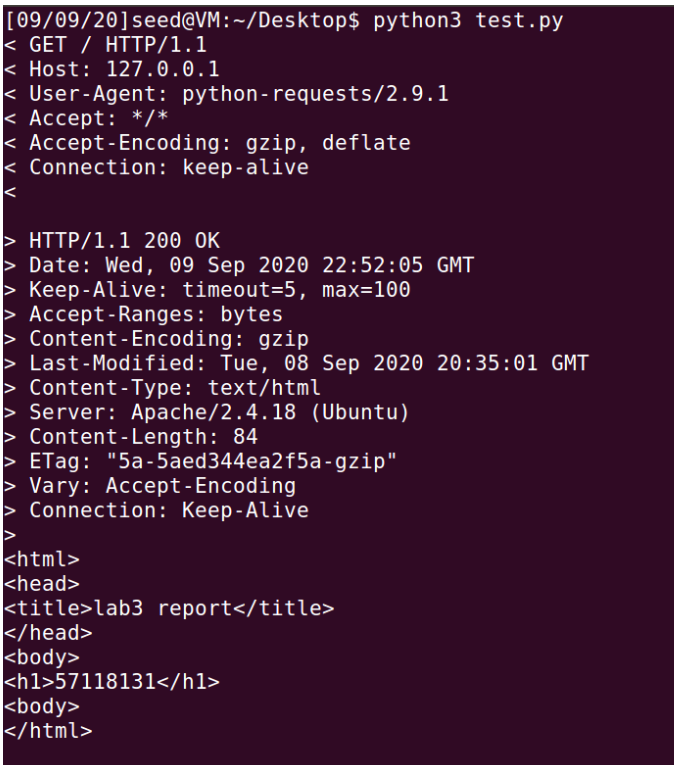


```
[09/08/20]seed@VM:.../html$ python3
Python 3.5.2 (default, Nov 17 2016, 17:05:23)
[GCC 5.4.0 20160609] on linux
```

步骤3：创建test.py的python执行文件并使用python3命令执行



```python
import requests
from requests_toolbelt.utils import dump

resp=requests.get('http://127.0.0.1')
data=dump.dump_all(resp)
print(data.decode('utf-8'))
```



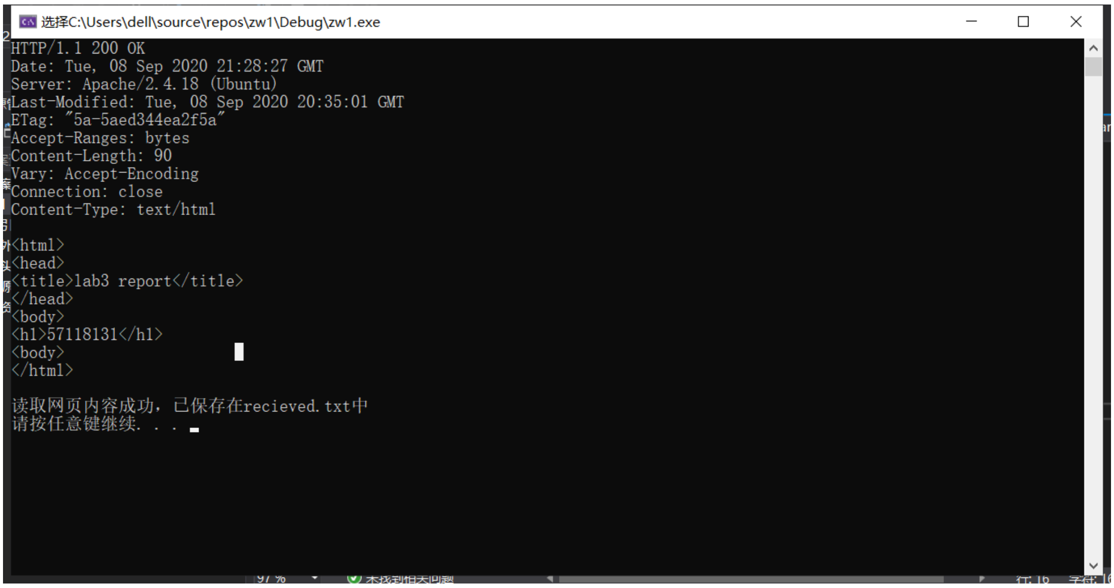
```
[09/09/20]seed@VM:~/Desktop$ python3 test.py
< GET / HTTP/1.1
< Host: 127.0.0.1
< User-Agent: python-requests/2.9.1
< Accept: */*
< Accept-Encoding: gzip, deflate
< Connection: keep-alive
<

> HTTP/1.1 200 OK
> Date: Wed, 09 Sep 2020 22:52:05 GMT
> Keep-Alive: timeout=5, max=100
> Accept-Ranges: bytes
> Content-Encoding: gzip
> Last-Modified: Tue, 08 Sep 2020 20:35:01 GMT
> Content-Type: text/html
> Server: Apache/2.4.18 (Ubuntu)
> Content-Length: 84
> ETag: "5a-5aed344ea2f5a-gzip"
> Vary: Accept-Encoding
> Connection: Keep-Alive
>
<html>
<head>
<title>lab3 report</title>
</head>
<body>
<h1>57118131</h1>
<body>
</html>
```

任务四：编写 HTTP 客户端以使用套接字检索站点的主页，代码如下：

步骤 1：在主机创建 c 语言程序，写入如下代码

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <iostream>
#include <winsock2.h>
#include<time.h>
#pragma comment(lib,"ws2_32.lib")
#pragma warning(disable:4996)          //这里是不启用错误代码为4996的检测

void ReadPage(const char* host)
{
    WSADATA data;
    //winsock版本2.2
    int err = WSAStartup(MAKEWORD(2, 2), &data);
    if (err)
        return;

    //用域名获取对方主机名
    struct hostent* h = gethostbyname(host);
    if (h == NULL)
        return;

    //IPV4
    if (h->h_addrtype != AF_INET)
        return;
    struct in_addr ina;
    //解析IP
    memmove(&ina, h->h_addr, 4);
    LPSTR ipstr = inet_ntoa(ina);

    //Socket封装
    struct sockaddr_in si;
    si.sin_family = AF_INET;
    si.sin_port = htons(80);
    si.sin_addr.S_un.S_addr = inet_addr(ipstr);
    int sock = socket(AF_INET, SOCK_STREAM, IPPROTO_TCP);
    connect(sock, (SOCKADDR*)&si, sizeof(si));
    if (sock == -1 || sock == -2)
        return;

    //发送请求
    char request[1024] = "GET /?st=1 HTTP/1.1\r\nHost:";
    strcat(request, host);
    strcat(request, "\r\nConnection:Close\r\n\r\n");
    int ret = send(sock, request, strlen(request), 0);
    //获取网页内容
    FILE* f = fopen("recieved.txt", "w");
    int isstart = 0;
    while (ret > 0)
    {
        const int bufsize = 1024;
        char* buf = (char*)calloc(bufsize, 1);
        ret = recv(sock, buf, bufsize - 1, 0);
        printf(buf);
        fprintf(f, "%s", buf);
        free(buf);
    }
    fclose(f);
    closesocket(sock);
    WSACleanup();
    printf("读取网页内容成功，已保存在recieved.txt中\n");
    return;
}
int main()
{
    const char* str = "VM";
    ReadPage(str);
    system("pause");
    return 0;
}
```

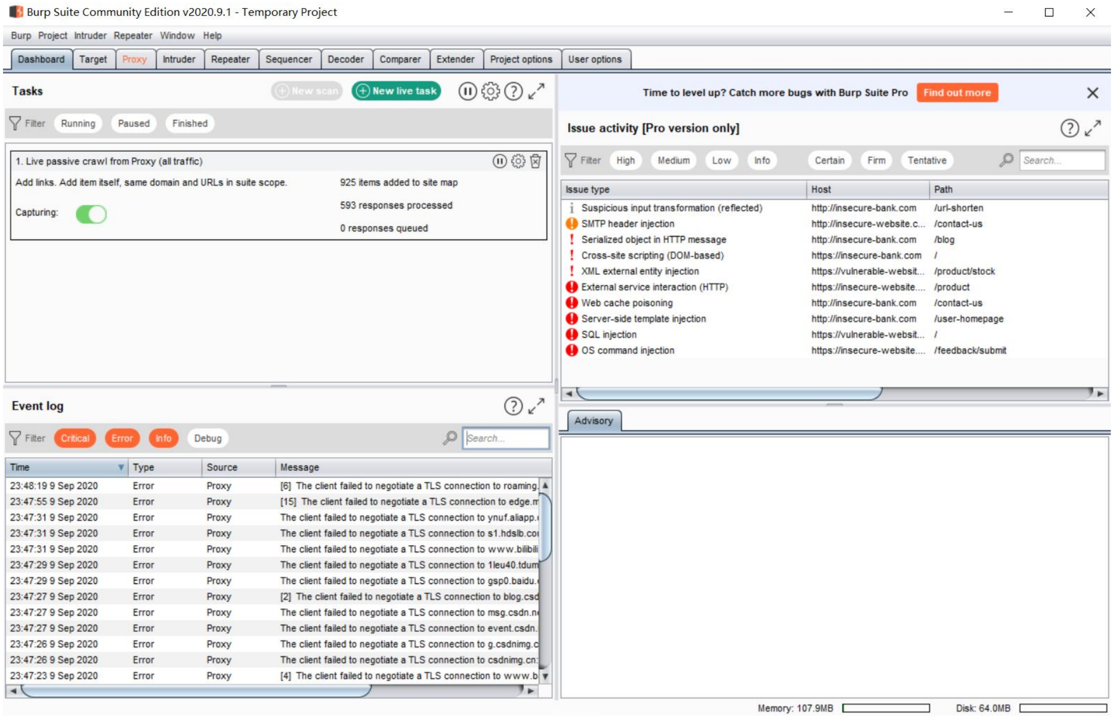
步骤 2：执行该文件，查看网页定向是否正确



```
HTTP/1.1 200 OK
Date: Tue, 08 Sep 2020 21:28:27 GMT
Server: Apache/2.4.18 (Ubuntu)
Last-Modified: Tue, 08 Sep 2020 20:35:01 GMT
ETag: "5a-5aed344ea2f5a"
Accept-Ranges: bytes
Content-Length: 90
Vary: Accept-Encoding
Connection: close
Content-Type: text/html

<html>
<head>
<title>lab3 report</title>
</head>
<body>
<h1>57118131</h1>
<body>
</html>

读取网页内容成功，已保存在recieved.txt中
请按任意键继续. . .
```

任务五：下载软件 Burp Suite 并访问网站查看请求与响应的信息

步骤 1：从 https://portswigger.net/burp 网站中下载 Comuunity 版本（安装过程省略），以下为软件截图


Burp Suite Community Edition v2020.9.1 - Temporary Project
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options
Tasks
New scan
New live task
Filter Running Paused Finished
Time to level up? Catch more bugs with Burp Suite Pro
Find out more
Issue activity [Pro version only]
Filter High Medium Low Info Certain Firm Tentative
Search...
Event log
Filter Critical Error Info Debug
Search...
Advisory
Memory: 107.9MB
Disk: 64.0MB

1. Live passive crawl from Proxy (all traffic)

Add links. Add item itself, same domain and URLs in suite scope.

Capturing:

925 items added to site map

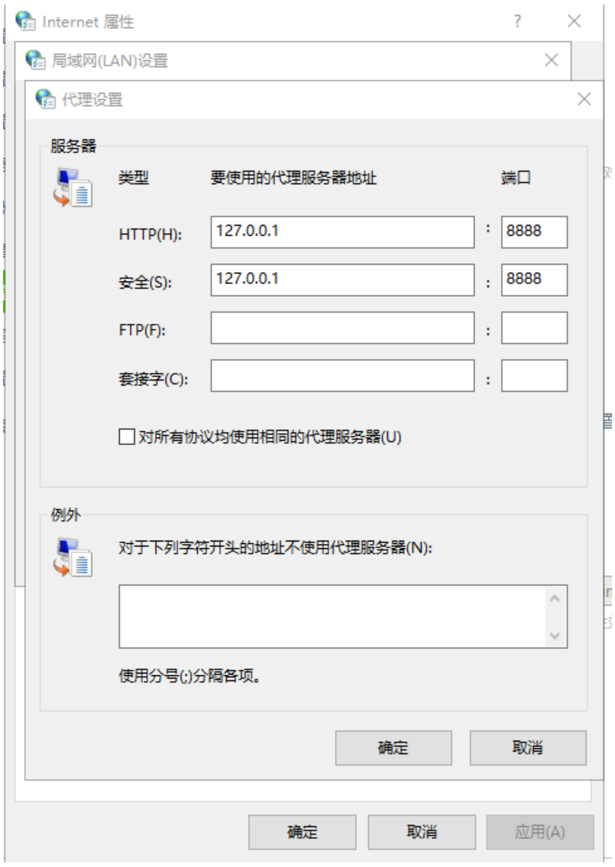
593 responses processed

0 responses queued

Issue type	Host	Path
Suspicious input transformation (reflected)	http://insecure-bank.com	/url-shorten
SMTP header injection	http://insecure-website.c...	/contact-us
Serialized object in HTTP message	http://insecure-bank.com	/blog
Cross-site scripting (DOM-based)	https://insecure-bank.com	/
XML external entity injection	https://vulnerable-websit...	/product/stock
External service interaction (HTTP)	https://insecure-website....	/product
Web cache poisoning	http://insecure-bank.com	/contact-us
Server-side template injection	http://insecure-bank.com	/user-homepage
SQL injection	https://vulnerable-websit...	/
OS command injection	https://insecure-website....	/feedback/submit

Time	Type	Source	Message
23:48:19 9 Sep 2020	Error	Proxy	[6] The client failed to negotiate a TLS connection to roaming.
23:47:55 9 Sep 2020	Error	Proxy	[15] The client failed to negotiate a TLS connection to edge.m
23:47:31 9 Sep 2020	Error	Proxy	The client failed to negotiate a TLS connection to ynuf.aliapp.
23:47:31 9 Sep 2020	Error	Proxy	The client failed to negotiate a TLS connection to s1.hdslb.co
23:47:31 9 Sep 2020	Error	Proxy	The client failed to negotiate a TLS connection to www.bilibili
23:47:29 9 Sep 2020	Error	Proxy	The client failed to negotiate a TLS connection to 1leu40.tdum
23:47:29 9 Sep 2020	Error	Proxy	The client failed to negotiate a TLS connection to gsp0.baidu.
23:47:27 9 Sep 2020	Error	Proxy	[2] The client failed to negotiate a TLS connection to blog.csd
23:47:27 9 Sep 2020	Error	Proxy	The client failed to negotiate a TLS connection to msg.csdn.n
23:47:27 9 Sep 2020	Error	Proxy	The client failed to negotiate a TLS connection to event.csdn.
23:47:26 9 Sep 2020	Error	Proxy	The client failed to negotiate a TLS connection to g.csdnimg.c
23:47:26 9 Sep 2020	Error	Proxy	The client failed to negotiate a TLS connection to csdnimg.cn:
23:47:23 9 Sep 2020	Error	Proxy	[4] The client failed to negotiate a TLS connection to www.b

步骤 2：对测试浏览器 Chrome 进行代理设置，地址设为 127.0.0.1，端口修改为 8888



服务器

类型	要使用的代理服务器地址	端口
HTTP(H):	127.0.0.1	8888
安全(S):	127.0.0.1	8888
FTP(F):		
套接字(C):		

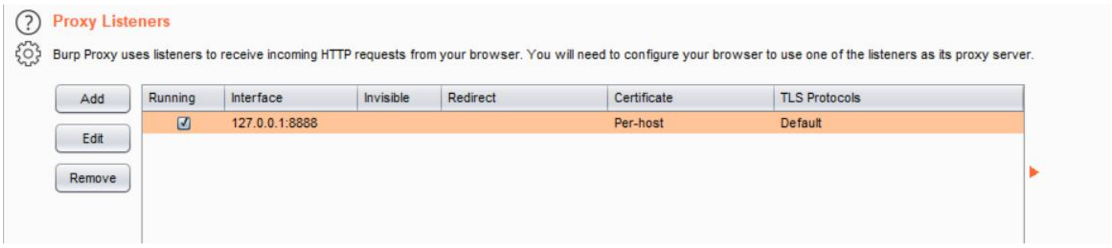
☐ 对所有协议均使用相同的代理服务器(U)

例外

对于下列字符开头的地址不使用代理服务器(N):

使用分号(;)分隔各项。

步骤 3：打开 Burp Suite 界面，设置 Proxy 代理，端口改为 8888

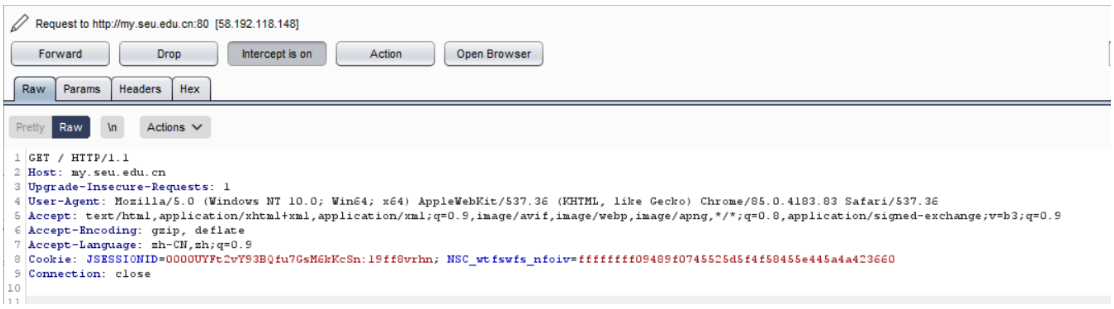


Proxy Listeners

Burp Proxy uses listeners to receive incoming HTTP requests from your browser. You will need to configure your browser to use one of the listeners as its proxy server.

Running	Interface	Invisible	Redirect	Certificate	TLS Protocols
☑	127.0.0.1:8888			Per-host	Default

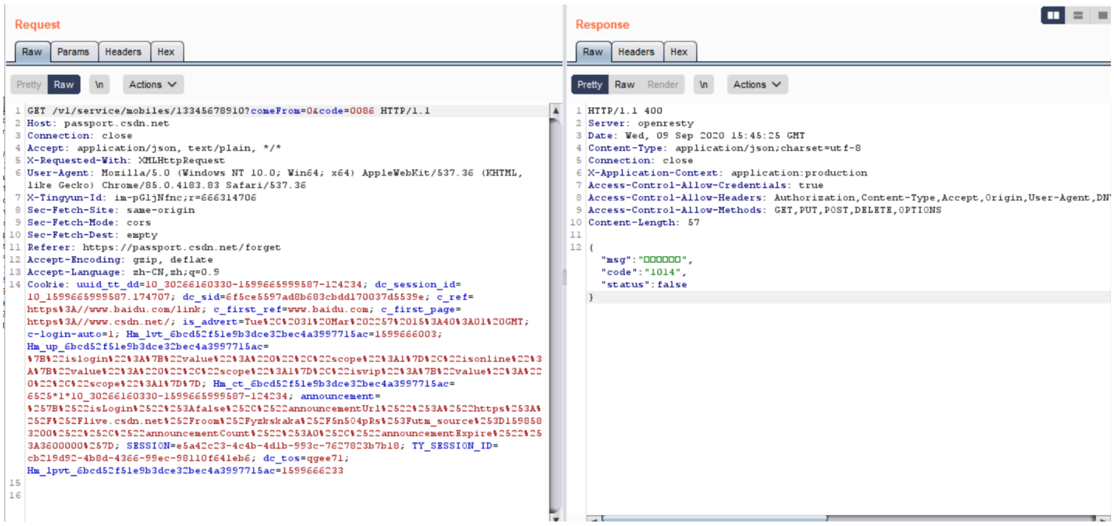
步骤 4：使用浏览器打开 my.seu.edu.cn 查看拦截情况



Request to http://my.seu.edu.cn:80 [58.192.118.148]

```
GET / HTTP/1.1
Host: my.seu.edu.cn
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/85.0.4183.83 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: JSESSIONID=0000UYFt2vY93BQfu7GsM6kKcSn:19ff8vrhn; NSC_wtfswfs_nfoiv=ffffffff09489f0745525d5f4f58455e445a4a423660
Connection: close
```

步骤 5：测试 CSDN 通过发送验证码找回密码功能，查看 Request 和 Response 功能(网站进行访问时需要点击 forward 按钮才能不断发送请求与接收响应，在测试 CSDN 之前需要对网页进行多次访问，因此可以先关闭拦截，点击 Intercept is on 按钮进行关闭，在需要拦截时再打开)



```
GET /v1/service/mobiles/13345678910?comeFrom=0&code=0086 HTTP/1.1
Host: passport.csdn.net
Connection: close
Accept: application/json, text/plain, */*
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/85.0.4183.83 Safari/537.36
X-Tingyun-Id: im-pGljNfnc;r=666314706
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://passport.csdn.net/forget
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: uuid_tt_dd=10_30266160330-1599665999587-124234; dc_session_id=
10_1599665999587.174707; dc_sid=6f5ce5597ad8b683cbdd170037d5539e; c_ref=
https%3A//www.baidu.com/link; c_first_ref=www.baidu.com; c_first_page=
https%3A//www.csdn.net/; is_advert=Tue%2C%2031%20Mar%202257%2015%3A40%3A01%20GMT;
c-login-auto=1; Hm_lvt_6bcd52f51e9b3dce32bec4a3997715ac=1599666003;
Hm_up_6bcd52f51e9b3dce32bec4a3997715ac=
%7B%22islogin%22%3A%7B%22value%22%3A%220%22%2C%22scope%22%3A1%7D%2C%22isonline%22%3
A%7B%22value%22%3A%220%22%2C%22scope%22%3A1%7D%2C%22isvip%22%3A%7B%22value%22%3A%22
0%22%2C%22scope%22%3A1%7D%7D; Hm_ct_6bcd52f51e9b3dce32bec4a3997715ac=
6525*1*10_30266160330-1599665999587-124234; announcement=
%257B%2522isLogin%2522%253Afalse%252C%2522announcementUrl%2522%253A%2522https%253A%
252F%252Flive.csdn.net%252Froom%252Fyzkskaka%252F5n504pRs%253Futm_source%253D159858
3200%2522%252C%2522announcementCount%2522%253A0%252C%2522announcementExpire%2522%25
3A3600000%257D; SESSION=e5a42c23-4c4b-4d1b-993c-7627823b7b18; TY_SESSION_ID=
cb219d92-4b8d-4366-99ec-98110f641eb6; dc_tos=qgee71;
Hm_lpvt_6bcd52f51e9b3dce32bec4a3997715ac=1599666233
```

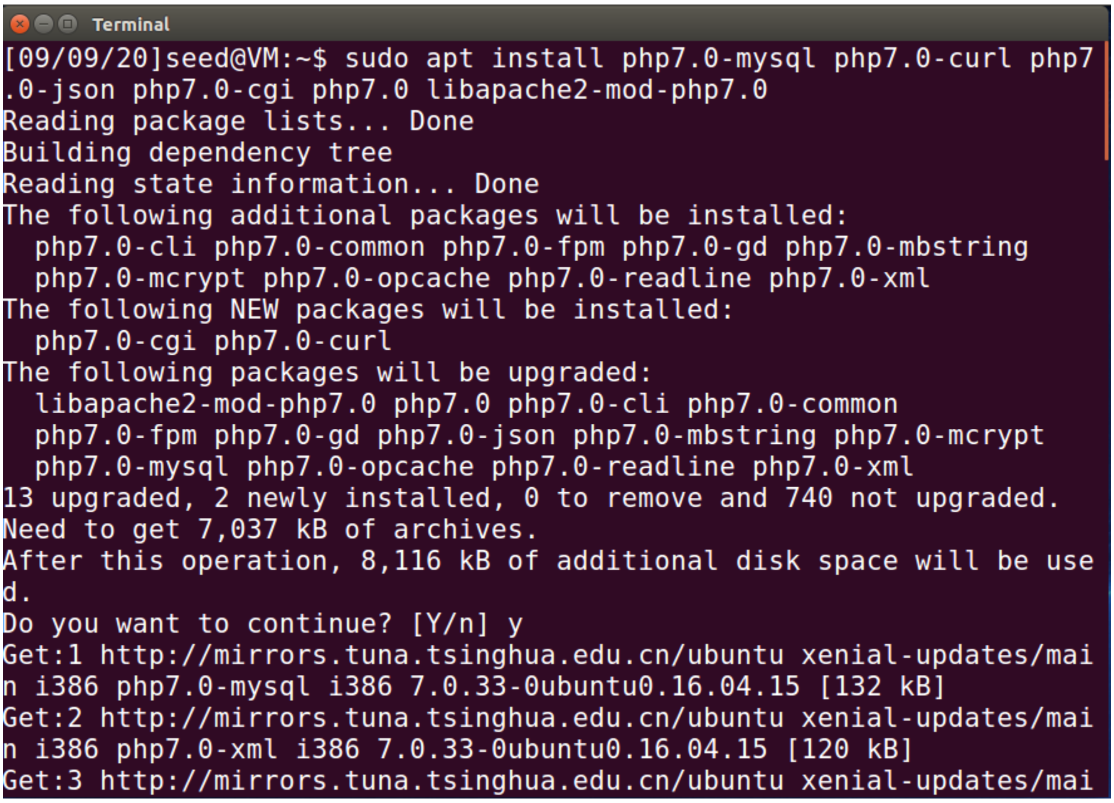
```
HTTP/1.1 400
Server: openresty
Date: Wed, 09 Sep 2020 15:45:25 GMT
Content-Type: application/json;charset=utf-8
Connection: close
X-Application-Context: application:production
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Authorization,Content-Type,Accept,Origin,User-Agent,DN
Access-Control-Allow-Methods: GET,PUT,POST,DELETE,OPTIONS
Content-Length: 57

{
  "msg":"□□□□□□",
  "code":"1014",
  "status":false
}
```

实验二　使用 PHP 和 Mysql 搭建一个简单的站点

任务一：在虚拟机中安装 PHP（使用以前的 Apache 安装），编写一个脚本以回显 URL 中的参数。

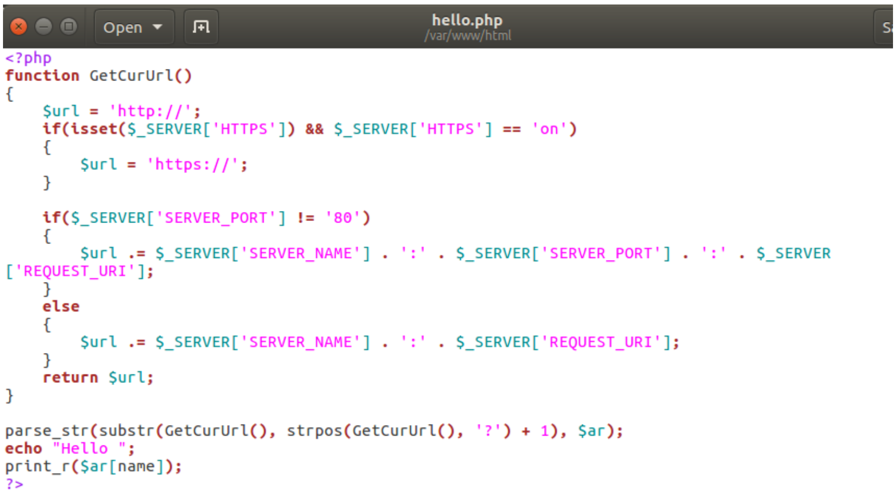
步骤 1：查询本机 php 版本（本机 7.0），在终端中安装对应版本的依赖库，执行如下的命令来安装 PHP 7.0 依赖库：

sudo apt install php7.0-mysql php7.0-curl php7.0-cgi php7.0 libapache2-mod-php7.0



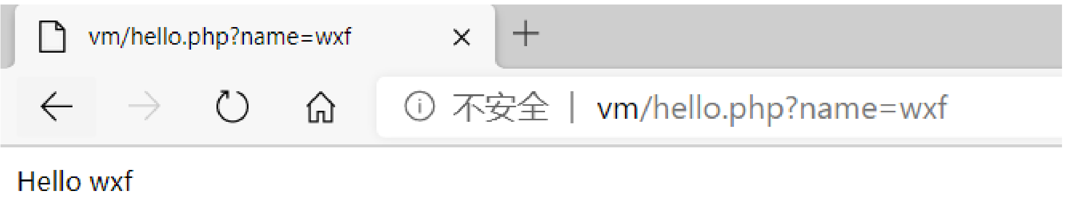
```
[09/09/20]seed@VM:~$ sudo apt install php7.0-mysql php7.0-curl php7
.0-json php7.0-cgi php7.0 libapache2-mod-php7.0
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  php7.0-cli php7.0-common php7.0-fpm php7.0-gd php7.0-mbstring
  php7.0-mcrypt php7.0-opcache php7.0-readline php7.0-xml
The following NEW packages will be installed:
  php7.0-cgi php7.0-curl
The following packages will be upgraded:
  libapache2-mod-php7.0 php7.0 php7.0-cli php7.0-common
  php7.0-fpm php7.0-gd php7.0-json php7.0-mbstring php7.0-mcrypt
  php7.0-mysql php7.0-opcache php7.0-readline php7.0-xml
13 upgraded, 2 newly installed, 0 to remove and 740 not upgraded.
Need to get 7,037 kB of archives.
After this operation, 8,116 kB of additional disk space will be use
d.
Do you want to continue? [Y/n] y
Get:1 http://mirrors.tuna.tsinghua.edu.cn/ubuntu xenial-updates/mai
n i386 php7.0-mysql i386 7.0.33-0ubuntu0.16.04.15 [132 kB]
Get:2 http://mirrors.tuna.tsinghua.edu.cn/ubuntu xenial-updates/mai
n i386 php7.0-xml i386 7.0.33-0ubuntu0.16.04.15 [120 kB]
Get:3 http://mirrors.tuna.tsinghua.edu.cn/ubuntu xenial-updates/mai
```

步骤 2:编写 hello.php，使用命令 sudo nautilus 以管理员方式打开文件管理器，将该文件放入 var/www/html，删除原来编写的 index.html 文件



```php
<?php
function GetCurUrl()
{
    $url = 'http://';
    if(isset($_SERVER['HTTPS']) && $_SERVER['HTTPS'] == 'on')
    {
        $url = 'https://';
    }

    if($_SERVER['SERVER_PORT'] != '80')
    {
        $url .= $_SERVER['SERVER_NAME'] . ':' . $_SERVER['SERVER_PORT'] . ':' . $_SERVER
['REQUEST_URI'];
    }
    else
    {
        $url .= $_SERVER['SERVER_NAME'] . ':' . $_SERVER['REQUEST_URI'];
    }
    return $url;
}

parse_str(substr(GetCurUrl(), strpos(GetCurUrl(), '?') + 1), $ar);
echo "Hello ";
print_r($ar[name]);
?>
```

步骤 3：在主机中打开浏览器，输入链接 http://VM/hello.php?name=wxf，将会显示 hello wxf

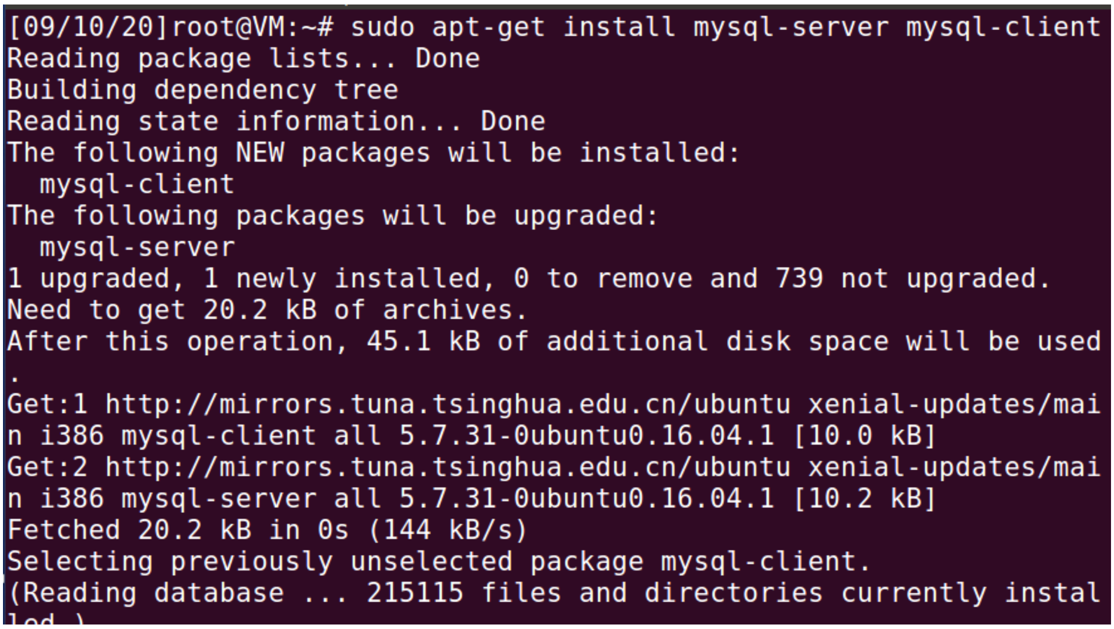


Hello wxf

任务二：安装 mysql 服务

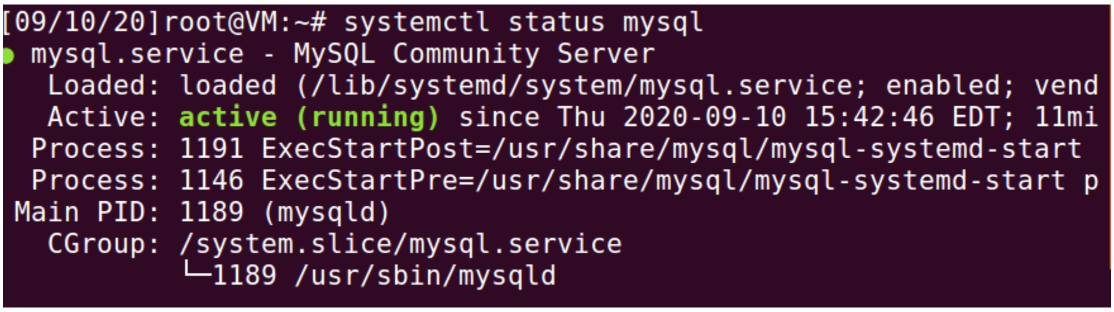
步骤 1：安装 Mysql

在终端输入 sudo apt-get install mysql-server mysql-client 进行安装



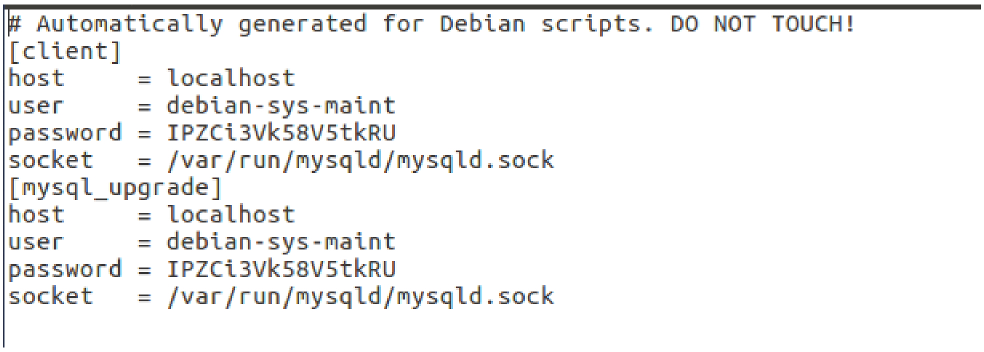
```
[09/10/20]root@VM:~# sudo apt-get install mysql-server mysql-client
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  mysql-client
The following packages will be upgraded:
  mysql-server
1 upgraded, 1 newly installed, 0 to remove and 739 not upgraded.
Need to get 20.2 kB of archives.
After this operation, 45.1 kB of additional disk space will be used
.
Get:1 http://mirrors.tuna.tsinghua.edu.cn/ubuntu xenial-updates/mai
n i386 mysql-client all 5.7.31-0ubuntu0.16.04.1 [10.0 kB]
Get:2 http://mirrors.tuna.tsinghua.edu.cn/ubuntu xenial-updates/mai
n i386 mysql-server all 5.7.31-0ubuntu0.16.04.1 [10.2 kB]
Fetched 20.2 kB in 0s (144 kB/s)
Selecting previously unselected package mysql-client.
(Reading database ... 215115 files and directories currently instal
lled.)
```

步骤 2：输入 systemctl status mysql 查看 mysql 状态是否启动



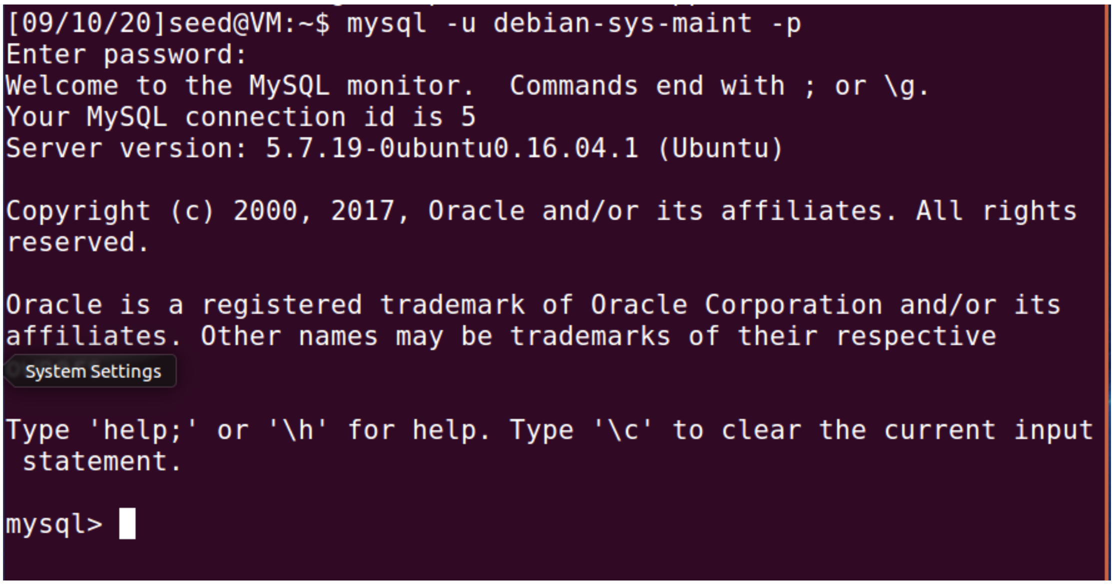
```
[09/10/20]root@VM:~# systemctl status mysql
● mysql.service - MySQL Community Server
   Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vend
   Active: active (running) since Thu 2020-09-10 15:42:46 EDT; 11mi
  Process: 1191 ExecStartPost=/usr/share/mysql/mysql-systemd-start
  Process: 1146 ExecStartPre=/usr/share/mysql/mysql-systemd-start p
 Main PID: 1189 (mysqld)
   CGroup: /system.slice/mysql.service
           └─1189 /usr/sbin/mysqld
```

步骤 3：gedit /etc/mysql/debian.cnf 打开该文件，查看 mysql 为我们创建的的一个用户，找到用户名和密码



```
# Automatically generated for Debian scripts. DO NOT TOUCH!
[client]
host     = localhost
user     = debian-sys-maint
password = IPZCi3Vk58V5tkRU
socket   = /var/run/mysqld/mysqld.sock
[mysql_upgrade]
host     = localhost
user     = debian-sys-maint
password = IPZCi3Vk58V5tkRU
socket   = /var/run/mysqld/mysqld.sock
```

步骤4：然后在终端输入mysql -u debian-sys-maint -p然后回车输入文件里显示的密码（IPZCi3Vk58V5tkRU）



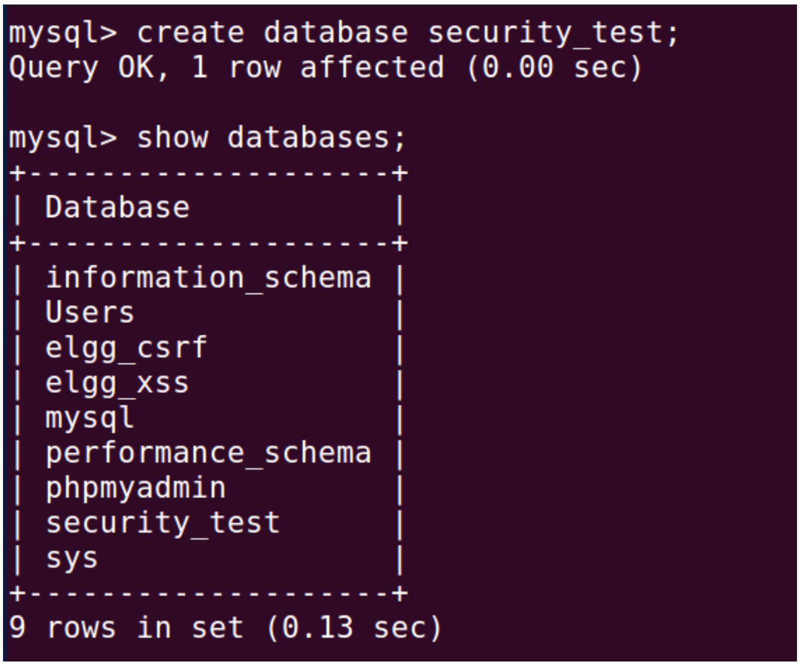
```
[09/10/20]seed@VM:~$ mysql -u debian-sys-maint -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.7.19-0ubuntu0.16.04.1 (Ubuntu)

Copyright (c) 2000, 2017, Oracle and/or its affiliates. All rights
reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective


Type 'help;' or '\h' for help. Type '\c' to clear the current input
 statement.

mysql> █
```

步骤5：进入mysql操作界面后，创建接下来需要使用的数据库以及相关的表

1. 创建数据库和表

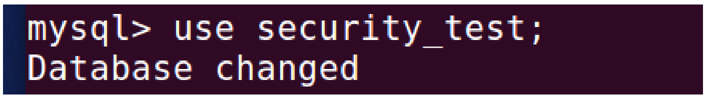
1）mysql>create database security_test;

2）mysql>show databases; 查看是否创建成功



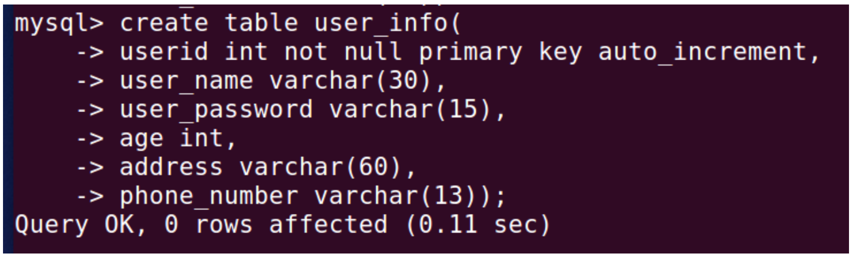
```
mysql> create database security_test;
Query OK, 1 row affected (0.00 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| Users              |
| elgg_csrf          |
| elgg_xss           |
| mysql              |
| performance_schema |
| phpmyadmin         |
| security_test      |
| sys                |
+--------------------+
9 rows in set (0.13 sec)
```

3）mysql>use security_test;进入创建好的数据库



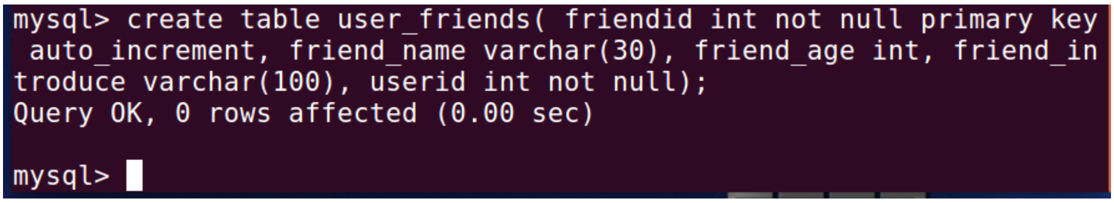
```
mysql> use security_test;
Database changed
```

4）创建用户信息表



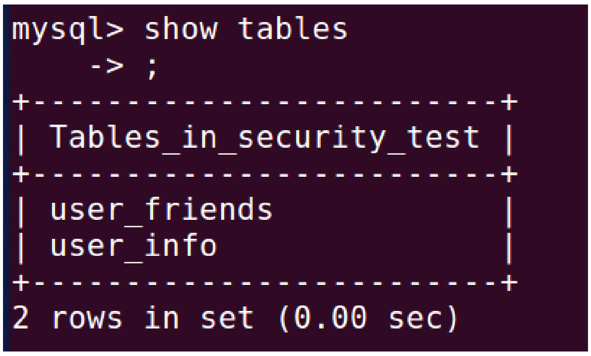
```
mysql> create table user_info(
    -> userid int not null primary key auto_increment,
    -> user_name varchar(30),
    -> user_password varchar(15),
    -> age int,
    -> address varchar(60),
    -> phone_number varchar(13));
Query OK, 0 rows affected (0.11 sec)
```

5）创建用户好友列表



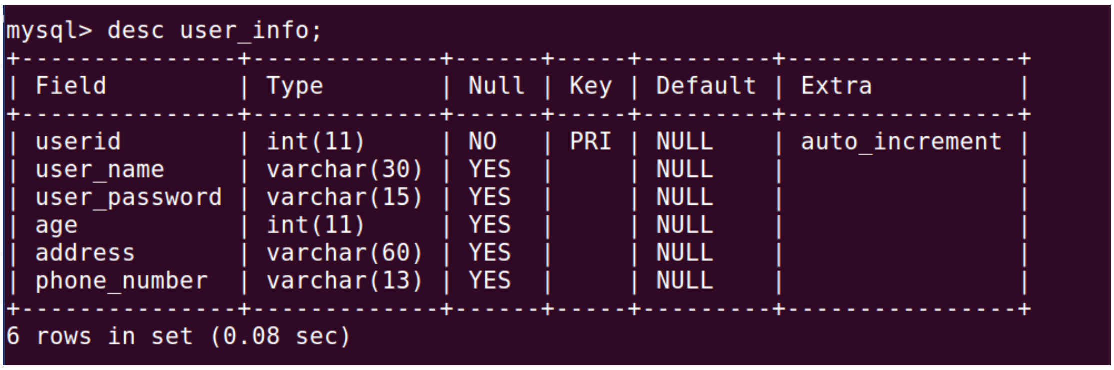
```
mysql> create table user_friends( friendid int not null primary key
 auto_increment, friend_name varchar(30), friend_age int, friend_in
troduce varchar(100), userid int not null);
Query OK, 0 rows affected (0.00 sec)

mysql> 
```

6）使用 show tables;查看表信息



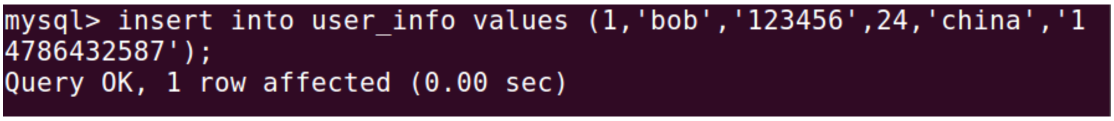
```
mysql> show tables
    -> ;
+-------------------------+
| Tables_in_security_test |
+-------------------------+
| user_friends            |
| user_info               |
+-------------------------+
2 rows in set (0.00 sec)
```

使用 desc user_info;查看表字段信息

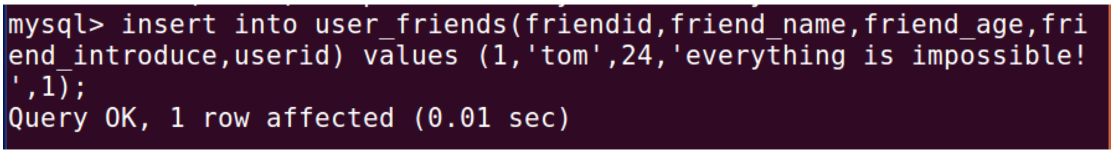


```
mysql> desc user_info;
+---------------+-------------+------+-----+---------+----------------+
| Field         | Type        | Null | Key | Default | Extra          |
+---------------+-------------+------+-----+---------+----------------+
| userid        | int(11)     | NO   | PRI | NULL    | auto_increment |
| user_name     | varchar(30) | YES  |     | NULL    |                |
| user_password | varchar(15) | YES  |     | NULL    |                |
| age           | int(11)     | YES  |     | NULL    |                |
| address       | varchar(60) | YES  |     | NULL    |                |
| phone_number  | varchar(13) | YES  |     | NULL    |                |
+---------------+-------------+------+-----+---------+----------------+
6 rows in set (0.08 sec)
```

7）为两个表插入数据，插入格式如下，可自行多插入几条数据



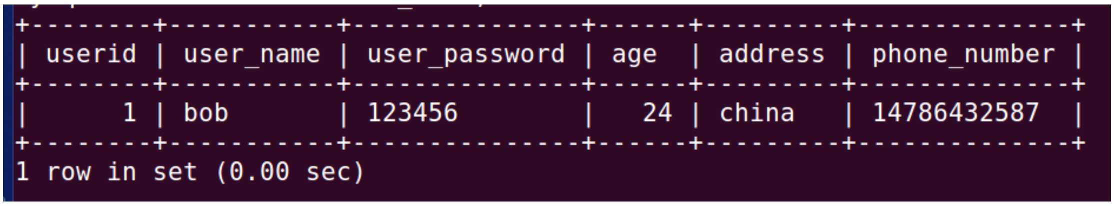
```
mysql> insert into user_info values (1,'bob','123456',24,'china','1
4786432587');
Query OK, 1 row affected (0.00 sec)
```



```
mysql> insert into user_friends(friendid,friend_name,friend_age,fri
end_introduce,userid) values (1,'tom',24,'everything is impossible!
',1);
Query OK, 1 row affected (0.01 sec)
```

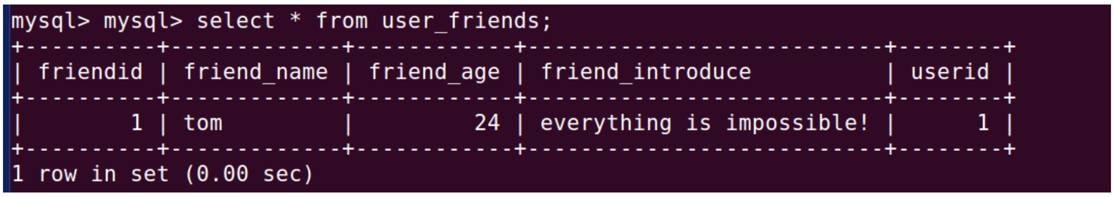
8）查看表格信息

mysql> select * from user_info;



```
+--------+-----------+---------------+------+---------+--------------+
| userid | user_name | user_password | age  | address | phone_number |
+--------+-----------+---------------+------+---------+--------------+
|      1 | bob       | 123456        |   24 | china   | 14786432587  |
+--------+-----------+---------------+------+---------+--------------+
1 row in set (0.00 sec)
```

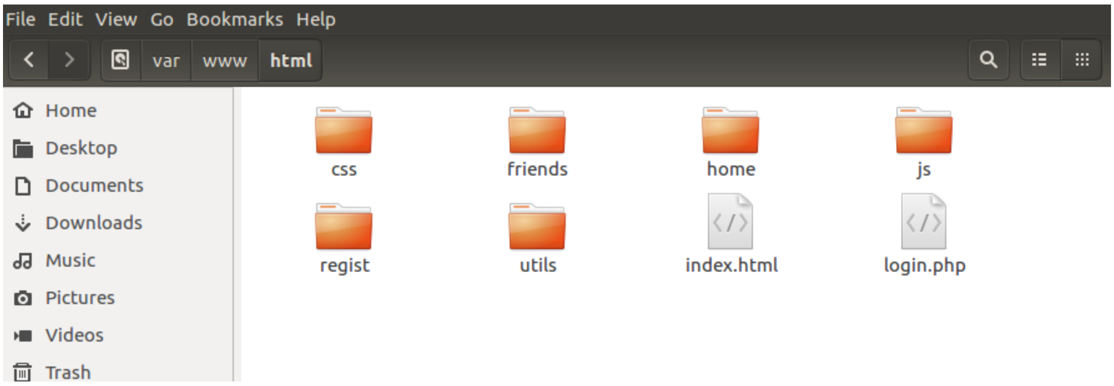
mysql> select * from user_friends;



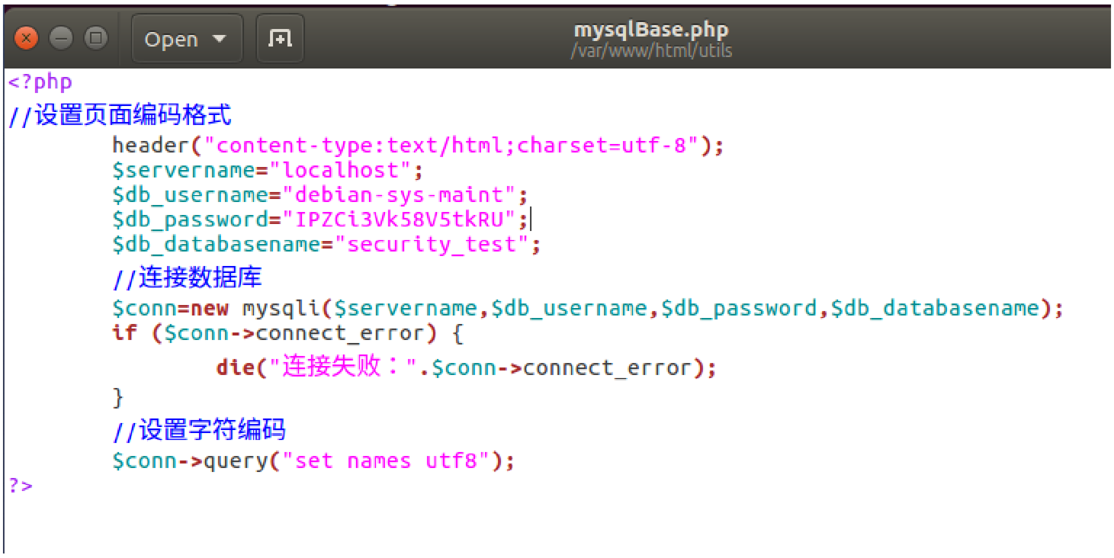
```
mysql> mysql> select * from user_friends;
+----------+-------------+------------+----------------------------+--------+
| friendid | friend_name | friend_age | friend_introduce           | userid |
+----------+-------------+------------+----------------------------+--------+
|        1 | tom         |         24 | everything is impossible! |      1 |
+----------+-------------+------------+----------------------------+--------+
1 row in set (0.00 sec)
```

任务三：测试运行一个简易的项目模板

步骤 1：将项目解压到 apache 的启动目录（/var/www/html），移除该文件夹原来编写的测试文件



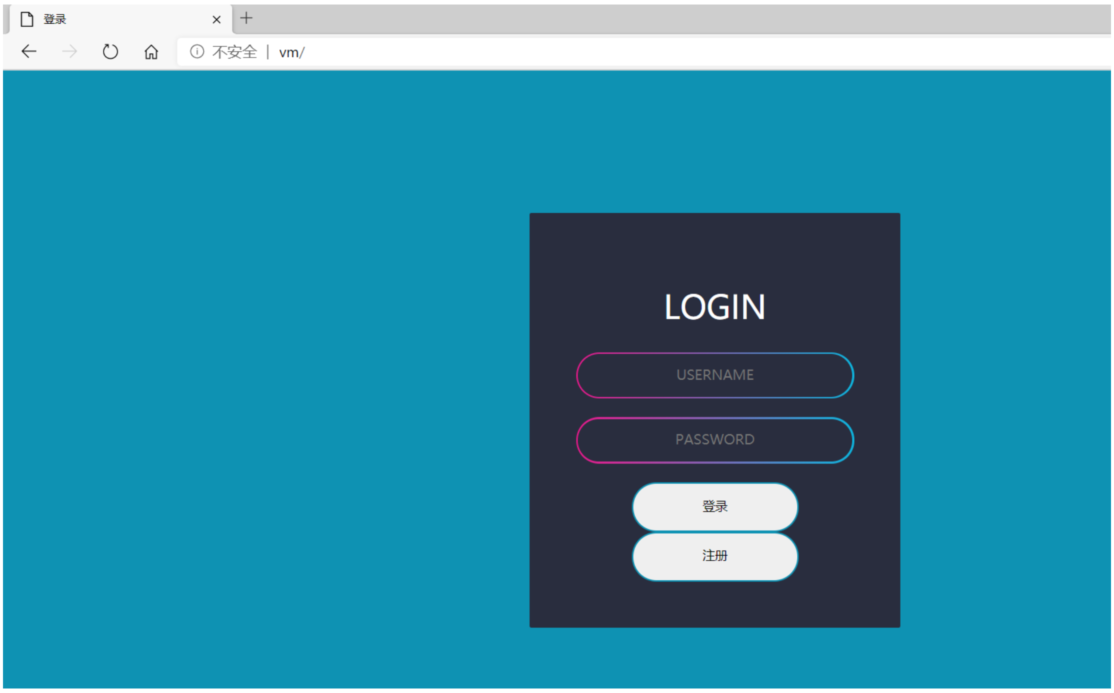
步骤 2：打开项目的 utils 文件夹下的 mysqlBase.php，核对虚拟机上的 mysql 用户名与密码还有数据库名称



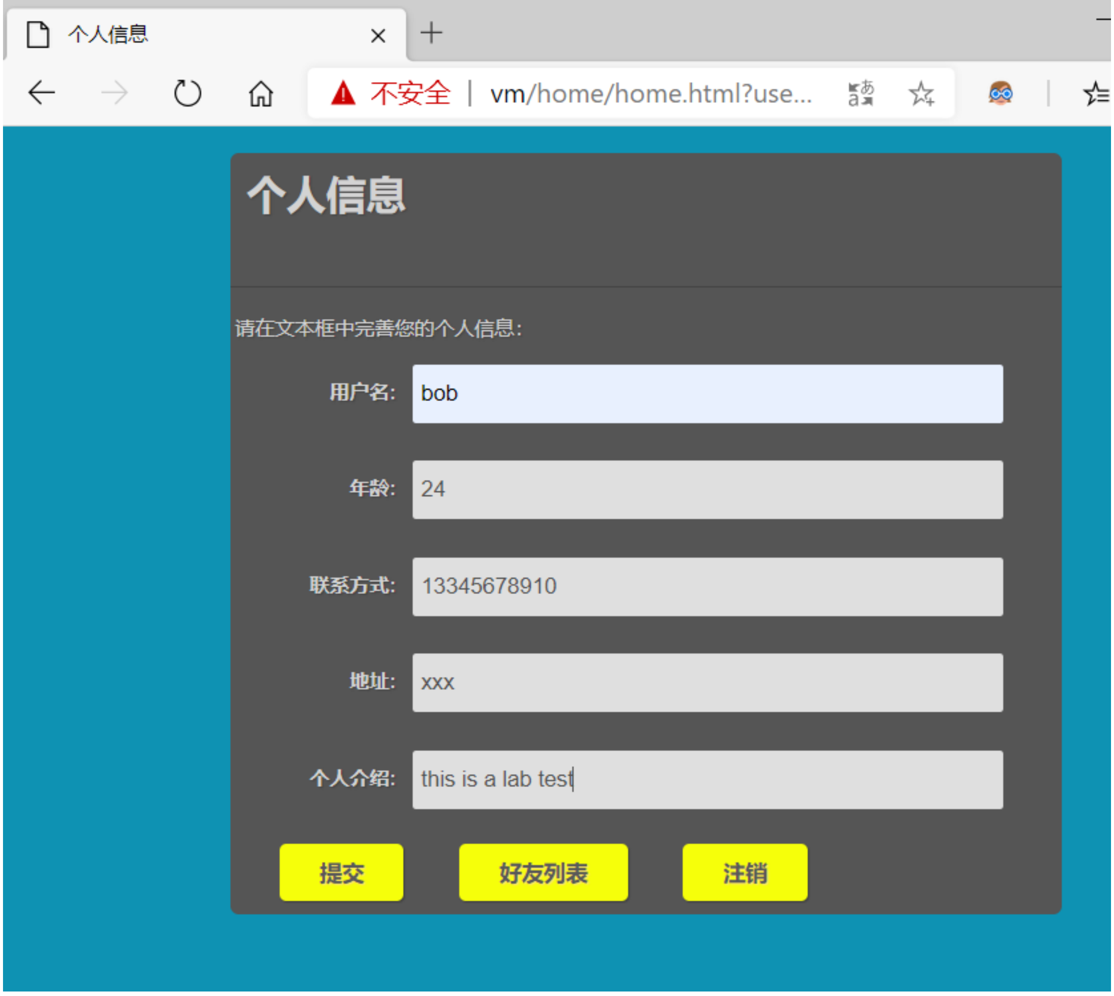
```
<?php
//设置页面编码格式
        header("content-type:text/html;charset=utf-8");
        $servername="localhost";
        $db_username="debian-sys-maint";
        $db_password="IPZCi3Vk58V5tkRU";
        $db_databasename="security_test";
        //连接数据库
        $conn=new mysqli($servername,$db_username,$db_password,$db_databasename);
        if ($conn->connect_error) {
                die("连接失败：".$conn->connect_error);
        }
        //设置字符编码
        $conn->query("set names utf8");
?>
```

步骤 3：使用主机浏览器访问地址：http://VM ，即可测试登录注册，修改个人信息，添加删除好友列表等等

登录界面：



可以输入之前在 mysql 中插入的用户名和密码，实现登录等操作，也可以进行账号注册

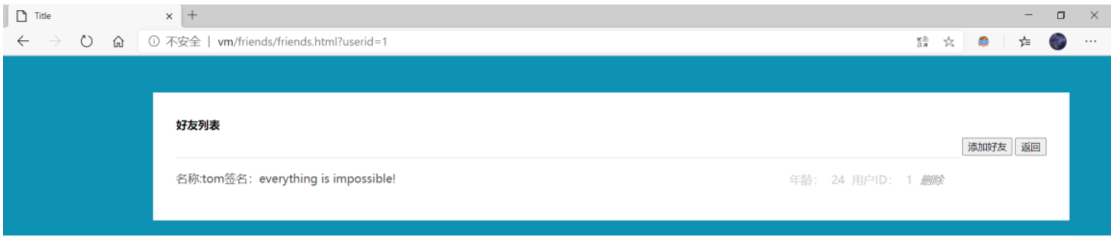


个人信息

请在文本框中完善您的个人信息：

用户名:	bob
年龄:	24
联系方式:	13345678910
地址:	xxx
个人介绍:	this is a lab test

点击好友列表后，进入好友列表界面，可以实现添加好友。



好友列表

名称:tom签名：everything is impossible! 年龄： 24 用户ID： 1 *删除*