Web Lab 1

1. Task 1:DNS

1.1. DNS服务器查询与IP地址访问

向DNS服务器查询指定域名 `cubicy.icu`的地址。

```
PS C:\Users\Direwolf> nslookup
默认服务器:  dns1.zju.edu.cn
Address:  10.10.0.21

> server itzel.ns.cloudflare.com
默认服务器:  itzel.ns.cloudflare.com
Addresses:  2606:4700:50::a29f:262a
          2a06:98c1:50::ac40:222a
          2803:f800:50::6ca2:c22a
          172.64.34.42
          162.159.38.42
          108.162.194.42

> cubicy.icu
服务器:  itzel.ns.cloudflare.com
Addresses:  2606:4700:50::a29f:262a
          2a06:98c1:50::ac40:222a
          2803:f800:50::6ca2:c22a
          172.64.34.42
          162.159.38.42
          108.162.194.42

名称:    cubicy.icu
Addresses:  2606:4700:3033::ac43:b3f4
          2606:4700:3033::6815:1fce
          172.67.179.244
          104.21.31.206
```

1. 先使用 `nslookup cubicy.icu`来进行查询，得到的回复是来自 `dns1.zju.edu.cn` 缓存的地址，属于非权威应答。
2. 为了得到储存 `cubicy.icu`地址的权威域名服务器，使用 `nslookup -qt=ns cubicy.icu`进行查询，得到两个nameserver：`itzel.ns.cloudflare.com`和 `sam.ns.cloudflare.com`。
3. 在命令行使用 `nslookup`，使用 `server itzel.ns.cloudflare.com`修改默认服务器，然后直接输入 `cubicy.icu`进行查询，得到地址：

```
名称:    cubicy.icu
Addresses:  2606:4700:3033::ac43:b3f4
```

```
2606:4700:3033::6815:1fce
172.67.179.244
104.21.31.206
```

1.2. DNS A记录查询

直接使用 `nslookup -qt=a cubicy.icu`进行多次DNS A记录查询，得到以下两种结果：

```
PS C:\Users\Direwolf> nslookup -qt=a cubicy.icu
服务器:  dns1.zju.edu.cn
Address:  10.10.0.21

非权威应答:
名称:    cubicy.icu
Addresses:  104.21.31.206
          172.67.179.244

PS C:\Users\Direwolf> nslookup -qt=a cubicy.icu
服务器:  dns1.zju.edu.cn
Address:  10.10.0.21

非权威应答:
名称:    cubicy.icu
Addresses:  172.67.179.244
          104.21.31.206
```

相邻两次的查询记录是不一样的，出现了 `172.67.179.244`和 `104.21.31.206`两种地址，连续查询时两者的先后顺序会不断交换。

这种现象是DNS轮询，可以实现负载均衡，将不同的用户请求分配到不同的服务器中，能够有效避免单一服务器的过载，有助于提高网站的稳定性、可用性和响应速度。

1.3. DNS记录中的隐藏文本

通过 `nslookup -qt=TXT cubicy.icu`对DNS记录中的文本进行查询，得到一段文本：

```
C:\Users\Direwolf>nslookup -qt=TXT cubicy.icu
服务器:  dns1.zju.edu.cn
Address:  10.10.0.21

非权威应答:
cubicy.icu      text =

        "v=spf1 -all"
cubicy.icu      text =

        "google-site-verification=1fhjV2lfeA6mIocyby2UVcZ8bC8o8NpJreyw1OLPDUY"
cubicy.icu      text =

        "5aSx5oGL44K944Oz44Kw5rKi5bGx6IG044GE44GmCuazo+OBhOOBpuOBsOOBi+OCiuOBruengeOBr+OCguOBhgrmjajjgabjgZ/jgYTjgYvjgok
K5b+Y44KM44Gf44GE44GL44KJCuOCguOBhiDlkJvjga7jgZPjgarjgpPjgaYK5b+Y44KM44Gh44KD44GG44KD44GG44KJ44GtICAKU1VLSVNVS0lTVUtJU1V
LSVNVS0k="

cubicy.icu      nameserver = sam.ns.cloudflare.com
cubicy.icu      nameserver = itzel.ns.cloudflare.com
```

对文本进行Base64解码，得到一段文本：

失恋ソング沢山聴いて 泣いてばかりの私はもう 捨てたいから 忘れたいから もう 君のことなんて 忘れちゃうからね SUKISUKISUKISUKISUKI

据查，这是番剧《夏日重现》ed的一段歌词。

1.4. 服务提供商提供的域名特点

通过对 `cubicy.icu`、`www.cubicy.icu`和 `blog.cubicy.icu`的DNS A查询，得到其IPv4地址：

```
PS C:\Users\Direwolf>  nslookup -qt=a cubicy.icu
服务器:  dns1.zju.edu.cn
Address:  10.10.0.21

非权威应答:
名称:    cubicy.icu
Addresses:  172.67.179.244
          104.21.31.206

PS C:\Users\Direwolf>  nslookup -qt=a www.cubicy.icu
服务器:  dns1.zju.edu.cn
Address:  10.10.0.21

非权威应答:
名称:    www.cubicy.icu
Addresses:  172.67.179.244
          104.21.31.206

PS C:\Users\Direwolf>  nslookup -qt=a blog.cubicy.icu
服务器:  dns1.zju.edu.cn
Address:  10.10.0.21

非权威应答:
名称:    blog.cubicy.icu
Addresses:  104.21.31.206
          172.67.179.244
```

忽略顺序，三个域名的返回结果一致，这是因为网站可以拥有多个别名，可以定位到相同的IP地址。

但直接访问这些域名会弹出：Direct IP access not allowed 的提示，是无法直接访问的，说明这些地址并不是服务器的真实地址。

结合提示内容，说明服务器的内容挂载在了服务提供商的服务器上。在这个例子中，是采用了反向代理（内网穿透）的技术将内网的端口暴露到外网，此时Cloud Flare承担了请求转发的工作。

2. Task 2:HTTP

2.1. 使用BurpSuite抓包登录过程

```
GET /user/index HTTP/1.1
Host: courses.zju.edu.cn
Cookie: _csrf=S8mwplVi9KWoF2WQ0TlCeFZeV2xFy%2BdjI3FlGlxzuw0%3D;
_pv0=IwZUWEx3YE2nmeh5MIlsY3ajgl9Fn7n7k%2Fdb1ShJCiL%2F3Jy6THg2clQPwtw8dGHEmsC4xyDrdR
m%2Ba0uTmGsFT133m1811njidHuQRLlJBu%2Bdk9HD3jGE4lXaIiJptVE6fNp%2BHWx8ElwgsgzHVjZ4BdY
kYAiIjx8hnmVivGQgt7Rtp9ENzOH73BVy6QDBZA33wiUKl6dbLYkdvcbaQ%2FK%2BWYD%2FHEQqVSFV0HJ9
sKW55WWdqUDQKY0bLcowtZON7PKoGHuEcrXueJv14DvpzivPfKW8H446%2FO0g6le7fT0LDtGdrWu8IYY5%
2Bi9DmpNoZvjWjV%2FdDI4X6BBebOmRD8xRu490gUbqeEB7fHMGAdF9jAi%2Bh9jj7gg2tweDSx2NMkh3xb
skOIkZdVIZkneNU7lGBpfslSH22%2FSjRqa0WX1NU8s%3D;
_pf0=AZfKUK4GWWzuTUWyZ5EhwfBR2ioV%2ByaWKOjOnl2QV8g%3D;
_pc0=reyDX%2Fn08sh6fTv3QaGeN1KrRQ8tGxxh3LO3sbs1a2BoL%2B259%2BAYIbJeJva8iww1;
iPlanetDirectoryPro=%2FwZku8xUpLm%2Fs8sbGZ8cS9iY9IF9RTV5y8pLm1H0U9alvDHmGnp0k3LCCIx
bZPulK9PyNao2t5KopNuuLCbui%2FaB3%2F6ZdxCuDQ0K3l1mvz066PkO4KFavWswfdWv7RZBC0JEb5pkf6
RfOtrZ7HN9UkymnjEgCT36pNaVSDliEw0B%2B9AeSPVmpLrOf%2FifY6ZmBhVALA6hAIN4kHLpTzWbH7fOt
6cOwclruM3tXJR6SN5lRbBdL1s28hyLKSzNknAjkJ7OltoscHjFczc0gURhsobmitA0mpDPZKxvE5TO0JWg
Ibh0uGgwb1lkWdQpz2E4oDj0O3zXO0moQ9CrJLUm8TWdRrQTAkBOSh%2FuftCPsLM%3D;
_ga=GA1.3.2001545788.1720141747; _gat=1; session=V2-1-1076ad44-38c5-4277-86f8-
ed6be81bf4e3.MjQyNjU5.1720228148518.Njnw3ixrhey_4RAMlVWrJ7utxfE
Cache-Control: max-age=0
Sec-Ch-Ua: "Not/A)Brand";v="8", "Chromium";v="126"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Accept-Language: zh-CN
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/126.0.6478.127 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/a
png,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: https://zjuam.zju.edu.cn/
Accept-Encoding: gzip, deflate, br
Priority: u=0, i
Connection: keep-alive
```

第一行为请求行，使用GET方法请求/user/index下的数据，并且指明了使用的HTTP版本为HTTP/1.1/。

从第二行到结束都是请求头，主要包含了以下部分：一段Cookie数据，客户端的缓存机制，浏览器的相关信息（UA界面，语言，类型，版本。操作系统），以及一些控制部分：

- `Upgrade-Insecure-Requests`: 告诉服务器客户端支持升级到更安全的连接，值 `1` 表示启用。
- `Accept`: 告诉服务器客户端能够处理的内容类型，如 `text/html`、`application/xhtml+xml`等。

- `Sec-Fetch-Site`, `Sec-Fetch-Mode`, `Sec-Fetch-User`, `Sec-Fetch-Dest`: 一组请求头部，用于提供有关请求的上下文信息。
- `Referer`: 表示请求发起的源地址。在这里是 `https://zjuam.zju.edu.cn/`。
- `Accept-Encoding`: 浏览器告诉服务器它支持的内容编码，这里是 `gzip`、`deflate`、`br`。
- `Priority`: 表示请求的优先级，`u=0, i`表示优先级的值。
- `Connection`: 控制不同HTTP请求/响应之间的连接策略，`keep-alive`表示持久连接。

网页保存用户数据的方法是Cookies，在用户登录过学在浙大后就会从服务器获得一段cookie，之后每次访问都会向服务器发送这段cookie，从而让服务器识别到用户，完成自动登录之类的操作。

2.2. HTTP区分不同包的原理

HTTP虽然工作在无边界的TCP协议上，但是通过对**数据格式**和**传输规则**加以约束后，能够区分不同的请求和响应，如上面提到的请求行、请求头以及其组成部分都对请求的不同部分做了约束，就像是课堂上提到的摩斯电码的规范一样。

2.3. 通过IP直接访问服务器

尝试直接访问 `cubicy.icu` 的源服务器地址 `101.132.222.48`返回了403错误，无法直接访问。

通过对比直接访问IP和用域名访问所使用的HTTP报文，将前者的报文拦截并修改为后者的报文进行访问，得到以下情形：

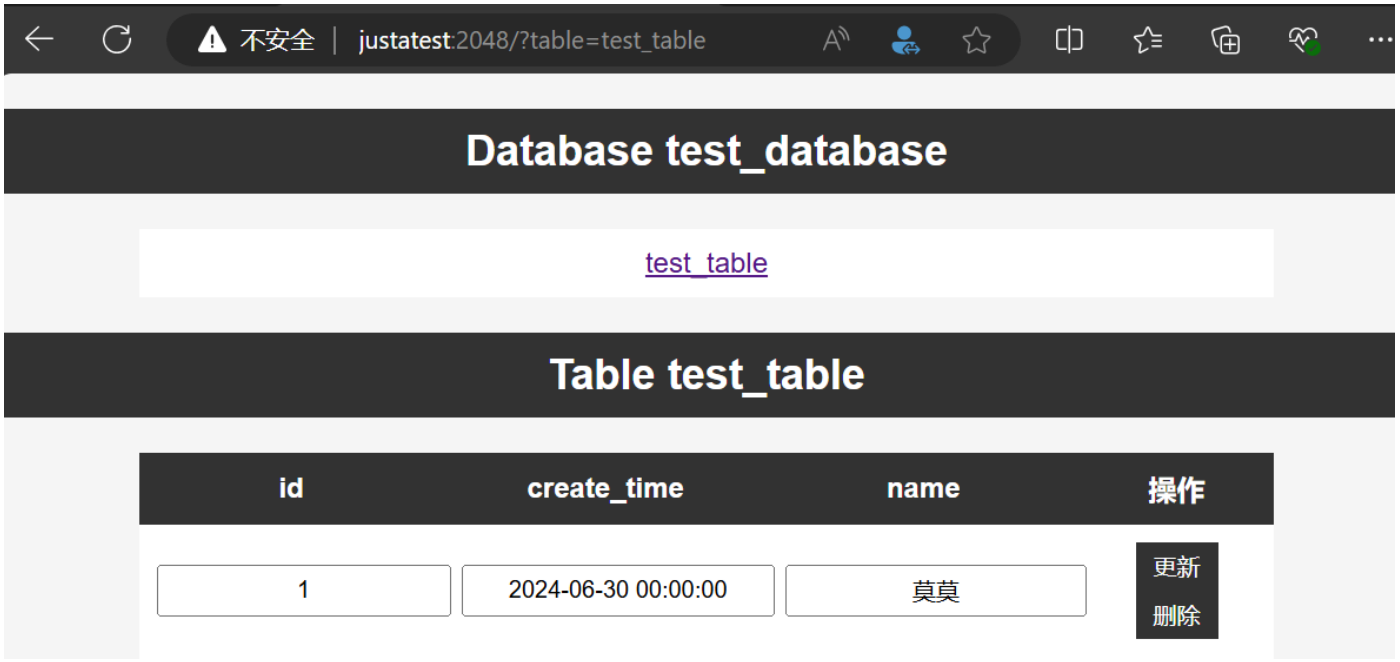
3. Task 3:预习

3.1. 了解基础知识

PHP、SQL：使用PHP Study配置本地运行环境。

Javascript：使用Node.js配置本地运行环境。

在了解相关知识的基础上，使用PHP Study搭建了本地服务器，并在服务器上连接了MySql进行基本的操作：



Javascript相关内容也在与浏览器开发者工具Console中得到了实践。

3.2. 用Python抓取成绩

已知有三个地方可以查到自己成绩：教务网（zdbk），eta，钉钉的成绩查询。在完成作业时发现只有前面两个是可以正常查询的，而且页面的数据似乎都是JavaScript动态加载的，源代码中没有成绩数据，但页面元素中有，仅使用requests库没想到该如何完成。结合自己之前的项目完成方式，决定改用Selenium库来完成。

思路：在代码内储存账号密码，向登录界面发送登录请求；完成登录后进入教务网的成绩查询页面，使用教务网自带的“打印”功能输出PDF文档。

遇到的困难：登录发送后会报错，但通过重新刷新一次能够正常登录；浏览器的安全功能禁用了自动化脚本的下载操作，即使关闭了安全功能仍旧无法下载，因此改用`save_screenshot`方法来获取页面，保存到指定路径。

代码：

```
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.chrome.options import Options
import time

opt = Options()
opt.add_argument('--headless')  # 设置为无头
opt.add_argument('--disable-gpu')  # 设置没有使用gpu
opt.add_argument("download.prompt_for_download=false")
driver = webdriver.Chrome(options=opt)
# 打开登录页面
```

```
driver.get('https://zjuam.zju.edu.cn/cas/login?
service=http%3A%2F%2Fzdbk.zju.edu.cn%2Fjwglxt%2Fxtgl%2Flogin_ssologin.html')
time.sleep(0.1)  # 根据实际情况调整等待时间
# 填写登录信息并提交
driver.find_element(By.ID, 'username').send_keys('3230105060')
driver.find_element(By.ID, 'password').send_keys('*********') #密码已略去
time.sleep(0.1)
driver.find_element(By.ID, 'dl').click()
time.sleep(0.1)
driver.get('https://zjuam.zju.edu.cn/cas/login?
service=http%3A%2F%2Fzdbk.zju.edu.cn%2Fjwglxt%2Fxtgl%2Flogin_ssologin.html') #二次访
问
# 访问成绩页面
driver.get('http://zdbk.zju.edu.cn/jwglxt/cxdy/xscjcx_cxXscjIndex.html?
gnmkdm=N5083&layout=default') #成绩页面
# 等待成绩页面加载
time.sleep(1)  # 根据实际情况调整等待时间
driver.find_element(By.ID, 'btn_dc').click() #点击打印键
time.sleep(2)
windows = driver.window_handles
driver.switch_to.window(windows[-1]) #转到弹出的新窗口
time.sleep(3)
driver.execute_script("document.body.style.zoom='70%'") #选择合适的缩放比例
#driver.find_element(By.ID, 'download').click()
time.sleep(0.5)
driver.save_screenshot('C:/Users/Direwolf/Desktop/grades_page.png') #保存截图，根据
自己电脑的情况修改路径
# 关闭浏览器
driver.quit()
```

得到的截图示例如下：

（做了个人信息的打码处理）

4. Bonus