

网络安全工程实践：实验二

李家昊 2017011620

李文博 2017011447

Task 0: 注册域名

在浏览器中访问: <http://ns.course.secrank.cn>, 注册域名 `liplus.c.secrank.cn` 和 `lwb.c.secrank.cn`, 将域名的 NS 记录指向虚拟机 IP 10.0.1.29。

Task 1: DNS权威服务

首先安装 bind9

```
1 sudo apt install bind9
```

进入目录 `/etc/bind`，修改配置文件 `named.conf.local` 内容如下：

```
1 zone "lwb.c.secrank.cn" {
2     type master;
3     file "/etc/bind/db.lwb.c.secrank.cn";
4 };
5 zone "liplus.c.secrank.cn" {
6     type master;
7     file "/etc/bind/liplus.c.secrank.cn";
8 };
```

创建并修改 `/etc/bind/liplus.c.secrank.cn` 配置文件，配置 A 记录，指向 10.0.1.29。

```

1 $TTL      604800
2 @         IN      SOA      liplus.c.secrank.cn. root.liplus.c.secrank.cn. (
3                               2              ; Serial
4                               604800         ; Refresh
5                               86400          ; Retry
6                               2419200        ; Expire
7                               604800 )       ; Negative Cache TTL
8 @         IN      NS       liplus.c.secrank.cn.
9 @         IN      AAAA      ::1
10 @        IN      A         10.0.1.29

```

创建并修改 `/etc/bind/db.lwb.c.secrank.cn` 配置文件，配置 A 记录，指向 10.0.1.29。

```

1 $TTL      604800
2 @      IN  SOA lwb.c.secrank.cn. root.lwb.c.secrank.cn. (
3          2          ; Serial
4          604800     ; Refresh
5          86400      ; Retry

```

```

6          2419200      ; Expire
7          604800 )    ; Negative Cache TTL
8 @      IN  NS  lwb.c.secrank.cn.
9 @      IN  A   10.0.1.29

```

利用 dig 工具解析注册域名，输出如下：

```

➔ bind dig @localhost liplus.c.secrank.cn

; <<>> DiG 9.11.3-1ubuntu1.13-Ubuntu <<>> @localhost liplus.c.secrank.cn
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 58442
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 2

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: e9c9a07f60c30e2a0a2847485f8bd65c38c90e5e9fc7429f (good)
;; QUESTION SECTION:
;liplus.c.secrank.cn.          IN      A

;; ANSWER SECTION:
liplus.c.secrank.cn.        604800  IN      A      10.0.1.29

;; AUTHORITY SECTION:
liplus.c.secrank.cn.        604800  IN      NS      liplus.c.secrank.cn.

;; ADDITIONAL SECTION:
liplus.c.secrank.cn.        604800  IN      AAAA    ::1

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sun Oct 18 13:45:00 CST 2020
;; MSG SIZE rcvd: 134

➔ bind dig @localhost lwb.c.secrank.cn

; <<>> DiG 9.11.3-1ubuntu1.13-Ubuntu <<>> @localhost lwb.c.secrank.cn
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 28748
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 8844897bc917b41b993ca32d5f8bd663508fee480c6c178f (good)
;; QUESTION SECTION:
;lwb.c.secrank.cn.           IN      A

;; ANSWER SECTION:
lwb.c.secrank.cn.           604800  IN      A      10.0.1.29

;; AUTHORITY SECTION:
lwb.c.secrank.cn.           604800  IN      NS      lwb.c.secrank.cn.

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sun Oct 18 13:45:07 CST 2020
;; MSG SIZE rcvd: 103

```

成功将域名解析为虚拟机所在 IP。

Task 2: HTTP服务

在 /etc/bind/liplus.c.secrank.cn 和 /etc/bind/db.lwb.c.secrank.cn 配置文件中增加一行 A 记录, 指向虚拟机所在 IP 地址:

```
1 web IN A 10.0.1.29
```

利用 dig 工具解析注册域名, 输出如下:

```
→ ~ dig web.liplus.c.secrank.cn

; <<>> DiG 9.11.3-lubuntu1.13-Ubuntu <<>> web.liplus.c.secrank.cn
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 19302
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 3

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
web.liplus.c.secrank.cn.      IN      A

;; ANSWER SECTION:
web.liplus.c.secrank.cn. 601887 IN      A      10.0.1.29

;; AUTHORITY SECTION:
liplus.c.secrank.cn.      604800 IN      NS      liplus.c.secrank.cn.

;; ADDITIONAL SECTION:
liplus.c.secrank.cn.      601489 IN      A      10.0.1.29
liplus.c.secrank.cn.      601489 IN      AAAA    ::1

;; Query time: 8 msec
;; SERVER: 10.1.0.2#53(10.1.0.2)
;; WHEN: Sun Oct 18 14:15:57 CST 2020
;; MSG SIZE rcvd: 126

→ ~ dig web.lwb.c.secrank.cn

; <<>> DiG 9.11.3-lubuntu1.13-Ubuntu <<>> web.lwb.c.secrank.cn
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 12764
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 2

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
web.lwb.c.secrank.cn.      IN      A

;; ANSWER SECTION:
web.lwb.c.secrank.cn. 552383 IN      A      10.0.1.29

;; AUTHORITY SECTION:
lwb.c.secrank.cn.      593252 IN      NS      lwb.c.secrank.cn.

;; ADDITIONAL SECTION:
lwb.c.secrank.cn.      549544 IN      A      10.0.1.29

;; Query time: 0 msec
;; SERVER: 10.1.0.2#53(10.1.0.2)
;; WHEN: Sun Oct 18 14:16:20 CST 2020
;; MSG SIZE rcvd: 95
```

下面搭建 HTTP 文件服务器, 首先安装 nginx

```
1 sudo apt install nginx
```

创建 nginx 配置文件 /etc/nginx/sites-available/ljh, 监听 80 端口, 将根路径映射到 /var/www/html/ljh, 并启用文件目录索引

```
1 server {
2     listen 80 default_server;
3     listen [::]:80 default_server;
4     server_name liplus.c.secrank.cn;
5     location / {
6         root /var/www/html/ljh;
7         autoindex on;
8     }
9 }
```

然后在 /var/www/html/ljh 目录下创建学号文件 2017011620.txt, 并写入学号

```
1 sudo bash -c "echo 2017011620 > 2017011620.txt"
```

同样创建 nginx 配置文件 /etc/nginx/sites-available/lwb, 监听 80 端口, 将根路径映射到 /var/www/html/lwb, 并启用文件目录索引, 注意 default_server 只能有一个。

```
1 server {
2     listen 80;
3     listen [::]:80;
4     server_name web.lwb.c.secrank.cn;
5     location / {
6         root /var/www/html/lwb;
7         autoindex on;
8     }
9 }
```

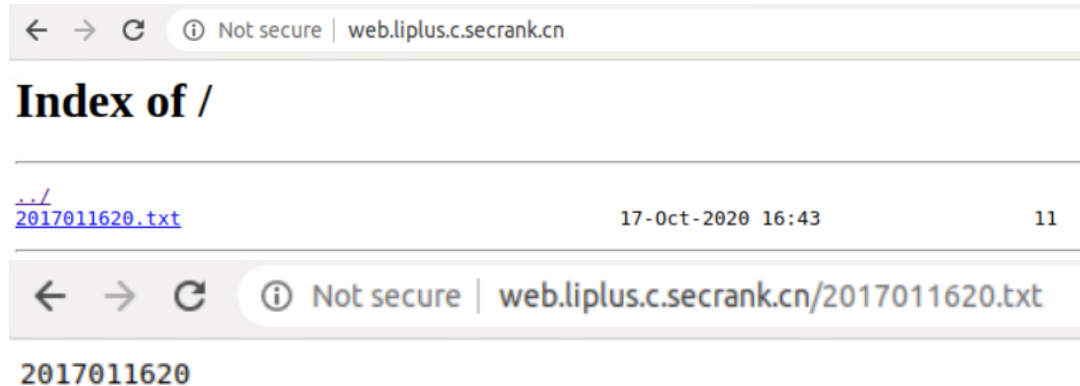
然后在 /var/www/html/lwb 目录下创建学号文件

```
1 sudo bash -c "echo 2017011447> 2017011447.txt"
```

重启 nginx 服务


```
1 sudo systemctl restart nginx
```

这样我们就把两个域名映射到同一端口, 并且能够通过不同域名访问不同的资源, 首先访问 web.liplus.c.secrank.cn 及其下面的文件



然后访问 web.lwb.c.secrank.cn 及其下面的文件

 Not secure web.lwb.c.secrank.cn		
Index of /		
../ 2017011447.txt	18-Oct-2020 04:55	11

 Not secure web.lwb.c.secrank.cn/2017011447.txt		
2017011447		

可以看出，服务器分别返回了两位同学的学号。

Task 3: PKI/HTTPS服务

首先安装 certbot

```
1 sudo apt install certbot
```

通过 certbot 申请证书

```
1 sudo certbot certonly --server
https://ca.course.secrank.cn/acme/acme/directory
```

选择 standalone 方式，在交互式程序中输入域名 web.lwb.c.secrank.cn，验证后即可获得证书，包括证书链文件和私钥文件，文件保存在 /etc/letsencrypt/live/web.lwb.c.secrank.cn 文件夹下。

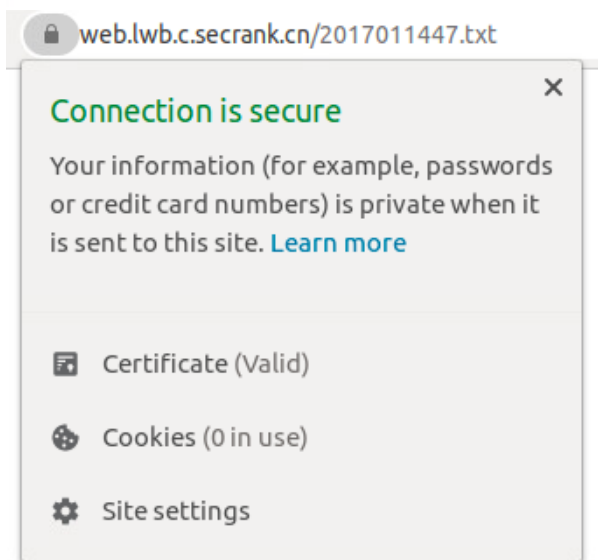
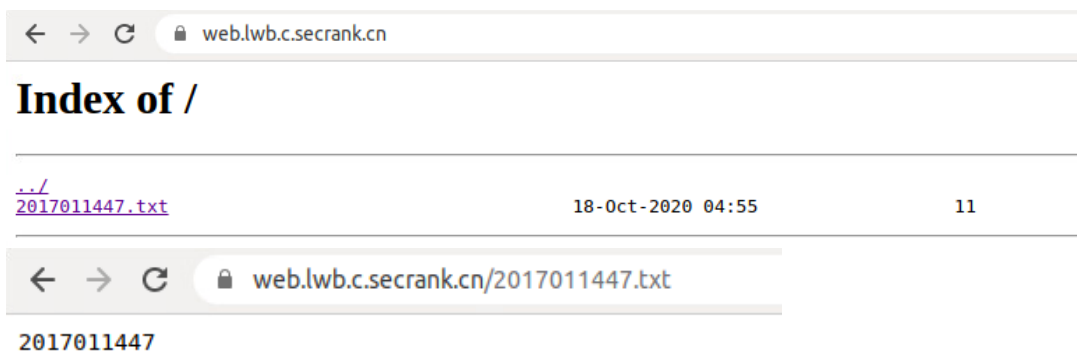
修改 nginx 配置文件 /etc/nginx/sites-available/lwb，增加 443 端口的监听，以及相关证书的配置

```
1 server {
2     listen 443 ssl;
3     listen [::]:443 ssl;
4
5     ssl_certificate
/etc/letsencrypt/live/web.lwb.c.secrank.cn/fullchain.pem;
6     ssl_certificate_key
/etc/letsencrypt/live/web.lwb.c.secrank.cn/privkey.pem;
7     keepalive_timeout 70;
8     ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
9
10    ssl_ciphers HIGH:!aNULL:!MD5;
11    server_name web.lwb.c.secrank.cn;
12    location / {
13        root /var/www/html/lwb;
14        autoindex on;
15    }
16 }
```

重启 nginx 服务

```
1 sudo systemctl restart nginx
```

访问 https://web.lwb.c.secrank.cn



成功访问学号文件，且浏览器显示 https 证书有效。

此外，我们对 http 服务开启了重定向，当用户访问 `http://web.lwb.c.secrank.cn` 时，重定向到 https 服务 `https://web.lwb.c.secrank.cn`。配置方法如下：修改 nginx 配置文件 `/etc/nginx/sites-available/lwb`，增加如下配置

```
1 server {
2     listen 80;
3     listen [::]:80;
4     return 301 https://$host$request_uri;
5     server_name web.lwb.c.secrank.cn;
6 }
```

在浏览器中访问 `http://web.lwb.c.secrank.cn`，成功重定向到

`https://web.lwb.c.secrank.cn`，在浏览器中看到 301 重定向数据包

Name	Status	Type	Initiator	Size	Time
web.lwb.c.secrank.cn	301	document...	Other	213 B	3 ms
web.lwb.c.secrank.cn	200	document	web.lwb.c.secrank...	373 B	3 ms

至此完成了 `web.lwb.c.secrank.cn` 域名的 https 服务配置，`web.liplus.c.secrank.cn` 域名的 https 服务配置过程与之类似，不再赘述。

思考题

Q1: 我们的 CA 是如何验证申请者具有对应服务器以及域名的控制权限的？

A1: 当申请者拥有域名以及服务器，在向 CA 申请证书时，CA 首先通过 DNS 解析出服务器的 IP，然后通过 HTTPS 协议将一串字符串返回给服务器，服务器收到字符串后，启动一个

HTTP 服务，当 CA 通过 HTTP GET 访问服务器时，服务器返回这个字符串，这时 CA 就能验证申请者具有域名控制权以及对对应服务器的控制权限了。

申请证书时在服务器抓包如下图，发现 CA 对服务器发起 HTTP GET /.well-known/acme-challenge/zvKq72yfkNlAf95fBNjUfEOYUopsM3V，服务器返回的数据中包含 zvKq72yfkNlAf95fBNjUfEOYUopsM3V.IWDXiQC4T B3hrQT GMT ZT v3j4wr91 scp0Pqbg 8EIGQg8，这应该就是服务器发来的完整字符串。

361	20.205455664	10.1.0.4	10.0.1.29	TLsv1.2	1103	Application Data
368	20.219455305	10.0.1.29	10.1.0.4	TLsv1.2	450	Application Data
369	20.219511910	10.0.1.29	10.1.0.4	TLsv1.2	892	Application Data
370	20.220041948	10.1.0.4	10.0.1.29	TCP	66 443 → 40260 [ACK] Seq=3840 Ack=4744 Win=64128 Len=0 TSval=330793163 TSecr=1028479370	
373	20.234236914	10.1.0.4	10.0.1.29	TCP	74 57548 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=330793177 TSecr=0 WS=128	
374	20.234252861	10.0.1.29	10.1.0.4	TCP	74 80 → 57548 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM=1 TSval=1028479385 TSecr=330793177 W.	
375	20.234493697	10.1.0.4	10.0.1.29	TCP	66 57548 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=330793178 TSecr=1028479385	
376	20.234837750	10.1.0.4	10.0.1.29	HTTP	229 GET /.well-known/acme-challenge/zvKq72yfkNlAf95fBNjUfEOYUopsM3V HTTP/1.1	
377	20.234843572	10.0.1.29	10.1.0.4	TCP	66 80 → 57548 [ACK] Seq=1 Ack=164 Win=65024 Len=0 TSval=1028479386 TSecr=330793178	
378	20.235770516	10.0.1.29	10.1.0.4	TCP	157 80 → 57548 [PSH, ACK] Seq=1 Ack=164 Win=65024 Len=91 TSval=1028479387 TSecr=330793178 [TCP segment of a r...	
379	20.235863264	10.0.1.29	10.1.0.4	HTTP	142 HTTP/1.0 200 OK	
380	20.236123669	10.1.0.4	10.0.1.29	TCP	66 57548 → 80 [ACK] Seq=164 Ack=92 Win=64256 Len=0 TSval=330793179 TSecr=1028479387	
381	20.236270081	10.1.0.4	10.0.1.29	TCP	66 57548 → 80 [FIN, ACK] Seq=164 Ack=169 Win=64256 Len=0 TSval=330793179 TSecr=1028479387	
382	20.236276894	10.0.1.29	10.1.0.4	TCP	66 80 → 57548 [ACK] Seq=169 Ack=165 Win=65024 Len=0 TSval=1028479387 TSecr=330793179	
383	20.240145161	10.1.0.4	10.0.1.29	TLsv1.2	743	Application Data
390	20.28325166	10.0.1.29	10.1.0.4	TCP	66 40260 → 443 [ACK] Seq=4744 Ack=4517 Win=64128 Len=0 TSval=1028479434 TSecr=330793183	
450	23.252250859	10.0.1.29	10.1.0.4	TLsv1.2	446	Application Data
451	23.252336000	10.0.1.29	10.1.0.4	TLsv1.2	820	Application Data
▼ Hypertext Transfer Protocol						
GET /.well-known/acme-challenge/zvKq72yfkNlAf95fBNjUfEOYUopsM3V HTTP/1.1\r\n						
Host: web.liplus.c.secrank.cn\r\n						
User-Agent: Go-http-client/1.1\r\n						
Accept-Encoding: gzip\r\n						
\r\n						
[Full request URI: http://web.liplus.c.secrank.cn/.well-known/acme-challenge/zvKq72yfkNlAf95fBNjUfEOYUopsM3V]						
[HTTP request 1/1]						
[Response in frame: 379]						
0040	59	99	47	45	54	20 2f 2e 77 65 6c 6c 2d 6b 6e 6f Y-GET /. well-kno
0050	77	6e	2f	61	63	6d 65 2d 63 68 61 6c 6c 65 6e 67 wn/acme challeng
0060	65	2f	7a	76	4b	71 37 32 79 66 6b 4e 4c 41 66 39 e/zvKq72 yfkNlAf9
0070	35	66	42	4e	4a	6a 55 46 45 4f 59 55 6f 70 73 4d 5fBNjUf EOYUopsM
0080	33	56	20	48	54	50 2f 31 2e 31 0d 0a 48 6f 73 3V HTTP/ 1.1·Hos
0090	74	3a	20	77	65	62 2e 6c 69 70 6c 75 2e 63 2e t: web.1 iplus.c.
00a0	73	65	63	72	61	6e 6b 2e 63 6e 0d 0a 55 73 65 72 secrank. cn~User
370	20.220041948	10.1.0.4	10.0.1.29	TCP	66 443 → 40260 [ACK] Seq=3840 Ack=4744 Win=64128 Len=0 TSval=330793163 TSecr=1028479370	
373	20.234236914	10.1.0.4	10.0.1.29	TCP	74 57548 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=330793177 TSecr=0 WS=128	
374	20.234252861	10.0.1.29	10.1.0.4	TCP	74 80 → 57548 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM=1 TSval=1028479385 TSecr=330793177 W.	
375	20.234493697	10.1.0.4	10.0.1.29	TCP	66 57548 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=330793178 TSecr=1028479385	
376	20.234837750	10.1.0.4	10.0.1.29	HTTP	229 GET /.well-known/acme-challenge/zvKq72yfkNlAf95fBNjUfEOYUopsM3V HTTP/1.1	
377	20.234843572	10.0.1.29	10.1.0.4	TCP	66 80 → 57548 [ACK] Seq=1 Ack=164 Win=65024 Len=0 TSval=1028479386 TSecr=330793178	
378	20.235770516	10.0.1.29	10.1.0.4	TCP	157 80 → 57548 [PSH, ACK] Seq=1 Ack=164 Win=65024 Len=91 TSval=1028479387 TSecr=330793178 [TCP segment of a r...	
379	20.235863264	10.0.1.29	10.1.0.4	HTTP	142 HTTP/1.0 200 OK	
380	20.236123669	10.1.0.4	10.0.1.29	TCP	66 57548 → 80 [ACK] Seq=164 Ack=92 Win=64256 Len=0 TSval=330793179 TSecr=1028479387	
381	20.236270081	10.1.0.4	10.0.1.29	TCP	66 57548 → 80 [FIN, ACK] Seq=164 Ack=169 Win=64256 Len=0 TSval=330793179 TSecr=1028479387	
382	20.236276894	10.0.1.29	10.1.0.4	TCP	66 80 → 57548 [ACK] Seq=169 Ack=165 Win=65024 Len=0 TSval=1028479387 TSecr=330793179	
383	20.240145161	10.1.0.4	10.0.1.29	TLsv1.2	743	Application Data
390	20.28325166	10.0.1.29	10.1.0.4	TCP	66 40260 → 443 [ACK] Seq=4744 Ack=4517 Win=64128 Len=0 TSval=1028479434 TSecr=330793183	
450	23.252250859	10.0.1.29	10.1.0.4	TLsv1.2	446	Application Data
▼ Hypertext Transfer Protocol						
HTTP/1.0 200 OK\r\n						
Server: BaseHTTP/0.6 Python/3.6.9\r\n						
Date: Sun, 18 Oct 2020 05:29:38 GMT\r\n						
\r\n						
[HTTP response 1/1]						
[Time since request: 0.001025514 seconds]						
[Request in frame: 376]						
[Request URI: http://web.liplus.c.secrank.cn/.well-known/acme-challenge/zvKq72yfkNlAf95fBNjUfEOYUopsM3V]						
File Data: 76 bytes						
▼ Data (76 bytes)						
Data: 7a764b7137379666b4e4c4166393566424e4a6a5546454f_						
[Length: 76]						
0030	01	fc	15	94	00	00 01 01 08 0a 3d 4d 59 9b 13 b7 00----->MY...
0040	80	da	7a	76	4b	71 37 32 79 66 6b 4e 4c 41 66 39 ..zvKq72 yfkNlAf9
0050	35	66	42	4e	4a	6a 55 46 45 4f 59 55 6f 70 73 4d 5fBNjUf EOYUopsM
0060	33	56	2e	6c	57	44 58 69 51 43 34 54 42 33 68 72 3V.IwDXi QC4T8hr
0070	51	54	47	4d	54	5a 54 76 33 6a 34 77 72 39 31 73 QTGMTZtv 3j4wr91s
0080	63	70	30	50	71	62 67 38 45 49 47 51 67 38 cp0Pqbg8 EIGQg8

假设申请者不具有域名控制权，只有服务器控制权限，那么域名将被指向域名所有者的服务器的 IP，攻击者不能控制该服务器，因此 CA 访问服务器时，服务器将不能返回对应的字符串。

假设申请者不具有服务器控制权限，只有域名控制权，即攻击者将域名解析到不属于自己的服务器，那么当 CA 访问服务器时，服务器同样不能返回对应的字符串。

Q2: 其中可能会存在哪些安全风险？

经过上述分析可以看出，如果攻击者不具有域名控制权，但恰好某个域名解析到了攻击者的服务器 IP 上，那么攻击者同样可以申请该域名的证书，使用该域名提供 HTTPS 服务。

但成熟的云服务提供商早已能避免这种风险，比如阿里云通过域名+主机备案的方式来绑定域名和主机的控制权，域名或主机过期后备案记录失效。如果域名过期，那么解析记录就失效了，用户将不能再通过域名访问对应主机；如果主机过期，这时再通过域名访问对应主机，阿里的云盾将会检测并拦截这些流量。

分工情况

两人独立对各自域名完成实验配置任务，报告共同撰写。