

0. 环境配置

0.1 虚拟机与Docker配置

- Ubuntu-Seed 16.04
- 攻击机 IP：172.18.0.1（主机，对应网卡为 br-e4a075733e38）
- 用户机 IP：172.18.0.2（docker 容器 DNS_User，注意创建时不带 --privileged 选项）
- 服务机 IP：172.18.0.3（docker 容器 DNS_Server，注意创建时不带 --privileged 选项）
- 容器的创建和使用：

```
1 # 查看容器
2 sudo docker ps -a
3 # 创建子网（避免容器的 IP 因启动顺序不同而变化）
4 sudo docker network create --subnet=172.18.0.0/16 net_xba
5 # 创建并运行容器
6 sudo docker run -it --name=DNS_User --hostname=DNS_User --net net_xba --ip 172.18.0.2
7 "seedubuntu" /bin/bash
8 sudo docker run -it --name=DNS_Server --hostname=DNS_Server --net net_xba --ip 172.18.0.3
9 "seedubuntu" /bin/bash
10 # 运行容器
11 sudo docker start DNS_User
12 sudo docker exec -it DNS_User /bin/bash
13 sudo docker start DNS_Server
14 sudo docker exec -it DNS_Server /bin/bash
15 # 关闭容器
16 sudo docker stop DNS_User
17 sudo docker stop DNS_Server
18 # 删除容器
19 sudo docker rm DNS_User
20 sudo docker rm DNS_Server
21 # 主机和容器之间拷贝数据
22 docker cp 容器名称:路径 主机路径
23 docker cp 主机路径 容器名称:路径
```

0.2 配置用户计算机

- 修改文件 /etc/resolv.conf，将服务器 172.17.0.5 添加为文件中的第一个 nameserver，即将此服务器作为主 DNS 服务器。
- 修改后的 /etc/resolv.conf 如下所示：

```
1 root@DNS_User:/# cat /etc/resolv.conf
2 search localdomain
3 nameserver 172.18.0.3
4 # nameserver 127.0.0.11
5 options ndots:0
```

0.3 配置本地DNS服务器

0.3.1 BIND 及其配置文件

- 使用 BIND 作为 DNS 服务软件，其主要配置文件为 /etc/bind/named.conf，实际配置文件存储在该配置文件中的 include 条目对应的文件中，以下修改的配置文件为 /etc/bind/named.conf.options。
- /etc/bind/named.conf 文件内容如下：

```

1 root@DNS_Server:/# cat /etc/bind/named.conf
2 // This is the primary configuration file for the BIND DNS server named.
3 //
4 // Please read /usr/share/doc/bind9/README.Debian.gz for information on the
5 // structure of BIND configuration files in Debian, *BEFORE* you customize
6 // this configuration file.
7 //
8 // If you are just adding zones, please do that in /etc/bind/named.conf.local
9
10 include "/etc/bind/named.conf.options";
11 include "/etc/bind/named.conf.local";
12 include "/etc/bind/named.conf.default-zones";

```

0.3.2 设置DNS缓存

- 向 `/etc/bind/named.conf.options` 中选项块 `options` 中添加 `dump-file` 来设置 DNS 缓存。若指定该选项，则 BIND 则会将其缓存转存到指定的位置；若未指定该选项，则默认转存到 `/var/cache/bind/named_dump.db` 中。
- DNS 缓存相关命令：

```

1 sudo rndc dumpdb -cache // Dump the cache to the sepcified file
2 sudo rndc flush          // Flush the DNS cache

```

0.3.3 关闭DNSSEC

- 引入 DNSSEC 是为了防止对 DNS 服务器的 `spoofing` 攻击，通过修改配置文件 `/etc/bind/named.conf.options` 来关闭 DNSSEC：注释 `dnssec-validation auto` 条目，并添加 `dnssec-enable no;` 条目。
- 设置 DNS 缓存和关闭 DNSSEC 后的配置文件如下（即 Ubuntu-Seed 的默认配置）：

```

1 root@DNS_Server:/# cat /etc/bind/named.conf.options
2 options {
3     directory "/var/cache/bind";
4
5     // If there is a firewall between you and nameservers you want
6     // to talk to, you may need to fix the firewall to allow multiple
7     // ports to talk.  See http://www.kb.cert.org/vuls/id/800113
8
9     // If your ISP provided one or more IP addresses for stable
10    // nameservers, you probably want to use them as forwarders.
11    // Uncomment the following block, and insert the addresses replacing
12    // the all-0's placeholder.
13
14    // forwarders {
15    //     0.0.0.0;
16    // };
17
18    //=====
19    // If BIND logs error messages about the root key being expired,
20    // you will need to update your keys.  See https://www.isc.org/bind-keys
21    //=====
22    // dnssec-validation auto;
23    dnssec-enable no;
24    dump-file "/var/cache/bind/dump.db";
25    auth-nxdomain no;    # conform to RFC1035
26
27    query-source port    33333;
28    listen-on-v6 { any; };
29 };

```

0.3.4 启动DNS服务器

- 每次对 DNS 配置进行修改时，都需要重新启动 DNS 服务器。

```
1 | sudo service bind9 restart      # 启动或重新启动 BIND 9 DNS 服务器
2 | netstat -nau                    # 查看网络状态
3 | sudo named -d 3 -f -g           # 查看 BIND 错误信息
```

- 启动DNS服务器并查看其运行状态如下：

```
1 | root@DNS_Server:/# sudo service bind9 start
2 | * Starting domain name service... bind9 [ OK ]
3 | root@DNS_Server:/# netstat -nau
4 | Active Internet connections (servers and established)
5 | Proto Recv-Q Send-Q Local Address           Foreign Address         State
6 | udp      0      0 127.0.0.11:48044        0.0.0.0:*
7 | udp      0      0 0.0.0.0:33333          0.0.0.0:*
8 | udp      0      0 172.18.0.3:53          0.0.0.0:*
9 | udp      0      0 127.0.0.1:53           0.0.0.0:*
10| udp6     0      0 :::53                  :::*
```

0.3.5 测试DNS服务器

- 在DNS_User执行ping www.hust.edu.cn -c 2，前四次执行该指令时所需时间较长，甚至可能无法在超时时间内获得该域名对应的IP地址，在多次尝试并能成功ping通后，ping命令的执行速度大幅提高：

```
root@DNS_User: 172.18.0.2
root@DNS_User:/# ping www.hust.edu.cn -c 2
ping: unknown host www.hust.edu.cn
root@DNS_User:/# ping www.hust.edu.cn -c 2
ping: unknown host www.hust.edu.cn
root@DNS_User:/# ping www.hust.edu.cn -c 2
ping: unknown host www.hust.edu.cn
root@DNS_User:/# ping www.hust.edu.cn -c 2
PING www.hust.edu.cn (202.114.0.245) 56(84) bytes of data.
64 bytes from 245.0.114.202.hust.edu.cn.0.114.202.in-addr.arpa (202.114.0.245): icmp_seq=1
ttl=127 time=4.29 ms
64 bytes from 245.0.114.202.hust.edu.cn.0.114.202.in-addr.arpa (202.114.0.245): icmp_seq=2
ttl=127 time=1.49 ms

--- www.hust.edu.cn ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 6381ms
rtt min/avg/max/mdev = 1.493/2.892/4.292/1.400 ms
root@DNS_User:/# ping www.hust.edu.cn -c 1
PING www.hust.edu.cn (202.114.0.245) 56(84) bytes of data.
64 bytes from 245.0.114.202.hust.edu.cn.0.114.202.in-addr.arpa (202.114.0.245): icmp_seq=1
ttl=127 time=3.18 ms

--- www.hust.edu.cn ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 3.180/3.180/3.180/0.000 ms
```

- 在可以ping通后，在一开始执行ping命令时DNS服务器使用了正向DNS缓存，以获得域名对应的IP；在接收到第一个ICMP Reply报文后，服务器使用了反向DNS缓存，以获得IP对应的域名：

No.	Time	Source	Destination	Protocol	Length	Info
545	2022-04-16 12:34:25.443174651	172.18.0.2	172.18.0.3	DNS	75	Standard query 0xe160 A www.hust.edu.cn
546	2022-04-16 12:34:25.443509451	172.18.0.3	172.18.0.2	DNS	189	Standard query response 0xe160 A www.hust.edu.cn A 202...
547	2022-04-16 12:34:25.444163651	172.18.0.2	202.114.0.245	ICMP	98	Echo (ping) request id=0x002d, seq=1/256, ttl=64 (repl...
548	2022-04-16 12:34:25.447310151	202.114.0.245	172.18.0.2	ICMP	98	Echo (ping) reply id=0x002d, seq=1/256, ttl=127 (req...
549	2022-04-16 12:34:25.447599252	172.18.0.2	172.18.0.3	DNS	86	Standard query 0xeebc PTR 245.0.114.202.in-addr.arpa
550	2022-04-16 12:34:25.447891251	172.18.0.3	172.18.0.2	DNS	256	Standard query response 0xeebc PTR 245.0.114.202.in-add...

- 使用Wireshark得到文件0.3.1-测试DNS服务器.pcapng，可以看到在执行ping命令时，客户机先向服务器发送DNS询问请求，前四次执行ping命令对应第1~468个报文，第五次执行ping命令对应第545~548个报文。

No.	Time	Source	Destination	Protocol	Length	Info
1	2022-04-16 12:30:09.718190196	172.18.0.2	172.18.0.3	DNS	75	Standard query 0xdad5 A www.hust.edu.cn
2	2022-04-16 12:30:09.719571697	172.18.0.3	198.41.0.4	DNS	86	Standard query 0x829b A www.hust.edu.cn OPT
3	2022-04-16 12:30:09.719610597	172.18.0.3	198.41.0.4	DNS	70	Standard query 0xcfaa NS <Root> OPT
4	2022-04-16 12:30:09.746236801	198.41.0.4	172.18.0.3	DNS	70	Standard query response 0xcfaa NS <Root> OPT
5	2022-04-16 12:30:09.746273001	198.41.0.4	172.18.0.3	DNS	86	Standard query response 0x829b A www.hust.edu.cn OPT
8	2022-04-16 12:30:10.584439041	172.18.0.3	199.7.91.13	DNS	70	Standard query 0xe5a6 NS <Root> OPT
9	2022-04-16 12:30:10.584513340	172.18.0.3	199.7.91.13	DNS	86	Standard query 0xbb5d A www.hust.edu.cn OPT

0.4 在本地DNS服务器中建一个区域

假设我们拥有一个域名，我们将负责提供有关该域名的响应。我们将使用本地 **DNS** 服务器作为域的权威名称服务器。在本实验中，我们将为 **example.com** 域设置为权威服务器。此域名保留用于文档，并且不由任何人拥有，因此使用它是安全的。

0.4.1 创建区域

- 我们需要在 **DNS** 服务器中创建两个区域条目，方法是将以下内容添加到 **/etc/bind/named.conf.default-zones** 或 **/etc/bind/named.conf** 中。第一个区域用于正向查找（从主机名到 **IP**），第二个区域用于反向查找（从 **IP** 到主机名）。

```
1 zone "xubiang.com" {
2     type master;
3     file "/etc/bind/xubiang.com.db";
4 };
5
6 zone "0.168.192.in-addr.arpa" {
7     type master;
8     file "/etc/bind/192.168.0.db";
9 };
```

- 修改后的 **/etc/bind/named.conf** 如下：

```
1 root@DNS_Server:/# cat /etc/bind/named.conf
2 // This is the primary configuration file for the BIND DNS server named.
3 //
4 // Please read /usr/share/doc/bind9/README.Debian.gz for information on the
5 // structure of BIND configuration files in Debian, *BEFORE* you customize
6 // this configuration file.
7 //
8 // If you are just adding zones, please do that in /etc/bind/named.conf.local
9
10 include "/etc/bind/named.conf.options";
11 include "/etc/bind/named.conf.local";
12 include "/etc/bind/named.conf.default-zones";
13
14 zone "xubiang.com" {
15     type master;
16     file "/etc/bind/xubiang.com.db";
17 };
18
19 zone "0.168.192.in-addr.arpa" {
20     type master;
21     file "/etc/bind/192.168.0.db";
22 };
```

```
root@DNS_Server: 172.18.0.3
root@DNS_Server:/# cat /etc/bind/named.conf
// This is the primary configuration file for the BIND DNS server named.
//
// Please read /usr/share/doc/bind9/README.Debian.gz for information on the
// structure of BIND configuration files in Debian, *BEFORE* you customize
// this configuration file.
//
// If you are just adding zones, please do that in /etc/bind/named.conf.local

include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
include "/etc/bind/named.conf.default-zones";

zone "xubiang.com" {
    type master;
    file "/etc/bind/xubiang.com.db";
};

zone "0.168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/192.168.0.db";
};
```


0.4.2 设置正向查找区域文件

- 上述区域定义中 **file** 关键字之后的文件名为区域文件，这是存储实际 **DNS** 解析的位置。 创建以下 **xubiang.com.db** 区域文件：

```
1 $TTL 3D ; Default expiration time
2
3 @ IN SOA ns.xubiang.com. admin.xubiang.com. (
4     2001032701 ; Serial
5     8H ; Refresh
6     2H ; Retry
7     4W ; Expire
8     1D) ; Minimum
9
10 @ IN NS ns.xubiang.com.
11 @ IN MX 10 mail.xubiang.com.
12
13 www IN A 192.168.0.101
14 mail IN A 192.168.0.102
15 ns IN A 192.168.0.10
16 *.xubiang.com. IN A 192.168.0.100
```

- 使用 **docker** 的文件复制指令将其放置到 **DNS_Server** 的 **/etc/bind/** 目录中：

```
1 | sudo docker cp xubiang.com.db DNS_Server:/etc/bind/xubiang.com.db
```

```
root@DNS Server: 172.18.0.3
root@DNS_Server:/# cat /etc/bind/xubiang.com.db
$TTL 3D ; Default expiration time

@ IN SOA ns.xubiang.com. admin.xubiang.com. (
    2001032701 ; Serial
    8H ; Refresh
    2H ; Retry
    4W ; Expire
    1D) ; Minimum

@ IN NS ns.xubiang.com.
@ IN MX 10 mail.xubiang.com.

www IN A 192.168.0.101
mail IN A 192.168.0.102
ns IN A 192.168.0.10
*.xubiang.com. IN A 192.168.0.100
```

0.4.3 设置反向查找区域文件

- 为了支持 **DNS** 反向查找，即从 **IP** 地址到主机名，我们还需要设置 **DNS** 反向查找文件。为 **xubiang.com** 域创建以下反向 **DNS** 查找文件 **192.168.0.db**：

```
1 $TTL 3D
2 @ IN SOA ns.xubiang.com. admin.xubiang.com. (
3     2001032701
4     8H
5     2H
6     4W
7     1D)
8 @ IN NS ns.xubiang.com.
9
10 101 IN PTR www.xubiang.com.
11 102 IN PTR mail.xubiang.com.
12 10 IN PTR ns.xubiang.com.
```

- 使用 **docker** 的文件复制指令将其放置到 **DNS_Server** 的 **/etc/bind/** 目录中：

```
1 | sudo docker cp 192.168.0.db DNS_Server:/etc/bind/192.168.0.db
```

```
root@DNS_Server: 172.18.0.3
root@DNS_Server:/# cat /etc/bind/192.168.0.db
$TTL 3D
@      IN      SOA      ns.xubiang.com. admin.xubiang.com. (
        2001032701
        8H
        2H
        4W
        1D)
@      IN      NS       ns.xubiang.com.

101    IN      PTR      www.xubiang.com.
102    IN      PTR      mail.xubiang.com.
10     IN      PTR      ns.xubiang.com.
```

0.4.4 检查区域文件的属性设置

- 在实验过程中，发现设置好 `bind9` 配置文件、正向查找区域文件、反向查找区域文件后，启用 `bind9` 服务后使用客户机执行 `dig www.xubiang.com`，得到的结果中没有应答信息：

```
root@DNS_Server: 172.18.0.3
root@DNS_Server:/# service bind9 start
* Starting domain name service... bind9 [ OK ]
root@DNS_Server:/# service bind9 status
* bind9 is running
root@DNS_Server:/#
```

```
root@DNS_User: 172.18.0.2
root@DNS_User:/# dig www.xubiang.com

; <<>> DiG 9.10.3-P4-Ubuntu <<>> www.xubiang.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: SERVFAIL, id: 55750
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;www.xubiang.com.          IN      A

;; Query time: 1 msec
;; SERVER: 172.18.0.3#53(172.18.0.3)
;; WHEN: Fri Apr 22 21:22:04 CST 2022
;; MSG SIZE  rcvd: 44
```

- 而使用 `named` 的 `-f` 或 `-g` 指令使 `named` 在前台运行时再使用客户机执行 `dig www.xubiang.com`，可以得到预期的 DNS 解析信息：

```
root@DNS_Server: 172.18.0.3
root@DNS_Server:/# named -f
```

```

root@DNS User: 172.18.0.2
root@DNS_User:/# dig www.xubiang.com

; <<>> DiG 9.10.3-P4-Ubuntu <<>> www.xubiang.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<- opcode: QUERY, status: NOERROR, id: 55085
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL:
2

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;www.xubiang.com.                IN      A

;; ANSWER SECTION:
www.xubiang.com.                259200  IN      A      192.168.0.101

;; AUTHORITY SECTION:
xubiang.com.                    259200  IN      NS      ns.xubiang.com.

;; ADDITIONAL SECTION:
ns.xubiang.com.                 259200  IN      A      192.168.0.10

;; Query time: 0 msec
;; SERVER: 172.18.0.3#53(172.18.0.3)
;; WHEN: Fri Apr 22 21:22:32 CST 2022
;; MSG SIZE rcvd: 93

```

- 经过问题排查，在 `named -g` 的输出信息中发现如下内容：

```

1 root@DNS_Server:/# named -g
2 ...
3 22-Apr-2022 21:25:55.045 configuring command channel from '/etc/bind/rndc.key'
4 22-Apr-2022 21:25:55.045 open: /etc/bind/rndc.key: permission denied
5 22-Apr-2022 21:25:55.045 couldn't add command channel 127.0.0.1#953: permission denied
6 22-Apr-2022 21:25:55.045 configuring command channel from '/etc/bind/rndc.key'
7 22-Apr-2022 21:25:55.045 open: /etc/bind/rndc.key: permission denied
8 22-Apr-2022 21:25:55.045 couldn't add command channel ::1#953: permission denied
9 ...

```

```

root@DNS_Server: 172.18.0.3
22-Apr-2022 21:25:55.045 configuring command channel from '/etc/bind/rndc.key'
22-Apr-2022 21:25:55.045 open: /etc/bind/rndc.key: permission denied
22-Apr-2022 21:25:55.045 couldn't add command channel 127.0.0.1#953: permission denied
22-Apr-2022 21:25:55.045 configuring command channel from '/etc/bind/rndc.key'
22-Apr-2022 21:25:55.045 open: /etc/bind/rndc.key: permission denied
22-Apr-2022 21:25:55.045 couldn't add command channel ::1#953: permission denied
22-Apr-2022 21:25:55.045 not using config file logging statement for logging due to -g op
tion

```

- 因此检查 `/etc/bind/rndc.key` 文件的权限，发现其他用户对该文件没有任何权限：

```

1 root@DNS_Server:/# ls -l /etc/bind
2 total 60
3 -rw----- 1 root root 225 Apr 22 21:15 192.168.0.db
4 -rw-r--r-- 1 root root 2389 Jun 29 2017 bind.keys
5 -rw-r--r-- 1 root root 237 Jun 29 2017 db.0
6 -rw-r--r-- 1 root root 271 Jun 29 2017 db.127
7 -rw-r--r-- 1 root root 237 Jun 29 2017 db.255
8 -rw-r--r-- 1 root root 353 Jun 29 2017 db.empty
9 -rw-r--r-- 1 root root 270 Jun 29 2017 db.local
10 -rw-r--r-- 1 root root 3171 Jun 29 2017 db.root
11 -rw-r--r-- 1 root bind 624 Apr 22 21:07 named.conf

```



```

12 -rw-r--r-- 1 root bind 665 Apr 22 21:07 named.conf.default-zones
13 -rw-r--r-- 1 root bind 165 Jun 29 2017 named.conf.local
14 -rw-r--r-- 1 bind bind 978 Jul 26 2017 named.conf.options
15 -rw-r----- 1 bind bind 77 Jul 26 2017 rndc.key
16 -rw----- 1 root root 516 Apr 22 21:15 xubiang.com.db
17 -rw-r--r-- 1 root root 1317 Jun 29 2017 zones.rfc1918

```

root@DNS_Server: 172.18.0.3

root@DNS_Server:/# ls -l /etc/bind

total 60

```

-rw----- 1 root root 225 Apr 22 21:15 192.168.0.db
-rw-r--r-- 1 root root 2389 Jun 29 2017 bind.keys
-rw-r--r-- 1 root root 237 Jun 29 2017 db.0
-rw-r--r-- 1 root root 271 Jun 29 2017 db.127
-rw-r--r-- 1 root root 237 Jun 29 2017 db.255
-rw-r--r-- 1 root root 353 Jun 29 2017 db.empty
-rw-r--r-- 1 root root 270 Jun 29 2017 db.local
-rw-r--r-- 1 root root 3171 Jun 29 2017 db.root
-rw-r--r-- 1 root bind 624 Apr 22 21:07 named.conf
-rw-r--r-- 1 root bind 665 Apr 22 21:07 named.conf.default-zones
-rw-r--r-- 1 root bind 165 Jun 29 2017 named.conf.local
-rw-r--r-- 1 bind bind 978 Jul 26 2017 named.conf.options
-rw-r----- 1 bind bind 77 Jul 26 2017 rndc.key
-rw----- 1 root root 516 Apr 22 21:15 xubiang.com.db
-rw-r--r-- 1 root root 1317 Jun 29 2017 zones.rfc1918

```

- 给其添加读权限，并再次使用 `named -g` 查看输出信息，此时错误消失，且在前台运行时依然能够获得正确的解析结果：

```

1 root@DNS_Server:/# chmod +r /etc/bind/rndc.key
2 root@DNS_Server:/# named -g
3 ...
4 22-Apr-2022 21:30:09.493 configuring command channel from '/etc/bind/rndc.key'
5 22-Apr-2022 21:30:09.493 command channel listening on 127.0.0.1#953
6 22-Apr-2022 21:30:09.493 configuring command channel from '/etc/bind/rndc.key'
7 22-Apr-2022 21:30:09.493 command channel listening on ::1#953
8 ...

```

root@DNS_Server: 172.18.0.3

```

22-Apr-2022 21:30:09.493 configuring command channel from '/etc/bind/rndc.key'
22-Apr-2022 21:30:09.493 command channel listening on 127.0.0.1#953
22-Apr-2022 21:30:09.493 configuring command channel from '/etc/bind/rndc.key'
22-Apr-2022 21:30:09.493 command channel listening on ::1#953
22-Apr-2022 21:30:09.493 not using config file logging statement for logging due to -g option

```

- 此时使用 `bind9` 进行测试，仍然无法得到正确的解析结果
- 再经过一些尝试后，类比 `rndc.key` 文件的错误，观察到组用户和其他用户对物理机编辑得到的 `xubiang.com.db` 及 `192.168.0.db` 文件没有任何权限，从而使用 `docker` 的复制语句复制到容器中后也缺少对应权限，从而导致 `bind9` 作为守护进程时读取文件失败：

```

1 [04/22/22]seed@VM:~/.../2022.04.15.DNS$ ls -l
2 总用量 236
3 -rw----- 1 seed seed 112396 4月 16 12:34 0.1.测试DNS服务器.pcapng
4 -rw----- 1 seed seed 225 4月 22 21:15 192.168.0.db
5 -rw-r--r-- 1 root root 105239 4月 21 23:30 name_d3_log.txt
6 -rw-r--r-- 1 root root 11366 4月 21 23:26 named_log.txt
7 -rw----- 1 seed seed 516 4月 22 21:15 xubiang.com.db

```

```

1 root@DNS_Server:/# ls -l /etc/bind
2 total 60
3 -rw----- 1 root root 225 Apr 22 21:15 192.168.0.db
4 -rw-r--r-- 1 root root 2389 Jun 29 2017 bind.keys
5 -rw-r--r-- 1 root root 237 Jun 29 2017 db.0
6 -rw-r--r-- 1 root root 271 Jun 29 2017 db.127
7 -rw-r--r-- 1 root root 237 Jun 29 2017 db.255
8 -rw-r--r-- 1 root root 353 Jun 29 2017 db.empty

```



```

 9 -rw-r--r-- 1 root root 270 Jun 29 2017 db.local
10 -rw-r--r-- 1 root root 3171 Jun 29 2017 db.root
11 -rw-r--r-- 1 root bind 624 Apr 22 21:07 named.conf
12 -rw-r--r-- 1 root bind 665 Apr 22 21:07 named.conf.default-zones
13 -rw-r--r-- 1 root bind 165 Jun 29 2017 named.conf.local
14 -rw-r--r-- 1 bind bind 978 Jul 26 2017 named.conf.options
15 -rw-r--r-- 1 bind bind 77 Jul 26 2017 rndc.key
16 -rw----- 1 root root 516 Apr 22 21:15 xubiang.com.db
17 -rw-r--r-- 1 root root 1317 Jun 29 2017 zones.rfc1918

```

```
seed@Attacker: 172.18.0.1
```

```
[04/22/22]seed@VM:~/.../2022.04.15.DNS$ ls -l
```

```
总用量 236
```

```

-rw----- 1 seed seed 112396 4月 16 12:34 0.1.测试DNS服务器.pcapng
-rw----- 1 seed seed 225 4月 22 21:15 192.168.0.db
-rw-r--r-- 1 root root 105239 4月 21 23:30 name_d3_log.txt
-rw-r--r-- 1 root root 11366 4月 21 23:26 named_log.txt
-rw----- 1 seed seed 516 4月 22 21:15 xubiang.com.db

```

```
root@DNS_Server: 172.18.0.3
```

```
root@DNS_Server:/# ls -l /etc/bind
```

```
total 60
```

```

-rw----- 1 root root 225 Apr 22 21:15 192.168.0.db
-rw-r--r-- 1 root root 2389 Jun 29 2017 bind.keys
-rw-r--r-- 1 root root 237 Jun 29 2017 db.0
-rw-r--r-- 1 root root 271 Jun 29 2017 db.127
-rw-r--r-- 1 root root 237 Jun 29 2017 db.255
-rw-r--r-- 1 root root 353 Jun 29 2017 db.empty
-rw-r--r-- 1 root root 270 Jun 29 2017 db.local
-rw-r--r-- 1 root root 3171 Jun 29 2017 db.root
-rw-r--r-- 1 root bind 624 Apr 22 21:07 named.conf
-rw-r--r-- 1 root bind 665 Apr 22 21:07 named.conf.default-zones
-rw-r--r-- 1 root bind 165 Jun 29 2017 named.conf.local
-rw-r--r-- 1 bind bind 978 Jul 26 2017 named.conf.options
-rw-r--r-- 1 bind bind 77 Jul 26 2017 rndc.key
-rw----- 1 root root 516 Apr 22 21:15 xubiang.com.db
-rw-r--r-- 1 root root 1317 Jun 29 2017 zones.rfc1918

```

- 因此赋予其对应权限，再重启 bind9 服务，此时使用客户机执行 `dig www.xubiang.com`，可以得到预期的 DNS 解析信息：

```

1 root@DNS_Server:/# chmod +r /etc/bind/xubiang.com.db
2 root@DNS_Server:/# chmod +r /etc/bind/192.168.0.db
3 root@DNS_Server:/# service bind9 restart
4 * Stopping domain name service... bind9
5 waiting for pid 497 to die
6
6 [ OK ]
7 * Starting domain name service... bind9
7 [ OK ]

```

```

1 root@DNS_User:/# dig www.xubiang.com
2
3 ; <<>> DiG 9.10.3-P4-Ubuntu <<>> www.xubiang.com
4 ;; global options: +cmd
5 ;; Got answer:
6 ;; ->HEADER<-- opcode: QUERY, status: NOERROR, id: 17559
7 ;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 2
8
9 ;; OPT PSEUDOSECTION:
10 ; EDNS: version: 0, flags:; udp: 4096
11 ;; QUESTION SECTION:
12 ;www.xubiang.com. IN A
13
14 ;; ANSWER SECTION:
15 www.xubiang.com. 259200 IN A 192.168.0.101

```

```
16
17 ;; AUTHORITY SECTION:
18 xubiang.com.          259200 IN NS ns.xubiang.com.
19
20 ;; ADDITIONAL SECTION:
21 ns.xubiang.com.       259200 IN A 192.168.0.10
22
23 ;; Query time: 14 msec
24 ;; SERVER: 172.18.0.3#53(172.18.0.3)
25 ;; WHEN: Fri Apr 22 22:41:07 CST 2022
26 ;; MSG SIZE rcvd: 93
```

```
root@DNS_Server: 172.18.0.3
```

```
root@DNS_Server:/# chmod +r /etc/bind/xubiang.com.db
root@DNS_Server:/# chmod +r /etc/bind/192.168.0.db
root@DNS_Server:/# service bind9 restart
* Stopping domain name service... bind9
waiting for pid 497 to die
```

```
[ OK ]
```

```
* Starting domain name service... bind9
```

```
[ OK ]
```

```
root@DNS_User: 172.18.0.2
```

```
root@DNS_User:/# dig www.xubiang.com
```

```
; <<>> DiG 9.10.3-P4-Ubuntu <<>> www.xubiang.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 17559
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL:
2

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;www.xubiang.com.          IN      A

;; ANSWER SECTION:
www.xubiang.com.          259200 IN      A      192.168.0.101

;; AUTHORITY SECTION:
xubiang.com.              259200 IN      NS      ns.xubiang.com.

;; ADDITIONAL SECTION:
ns.xubiang.com.           259200 IN      A      192.168.0.10

;; Query time: 14 msec
;; SERVER: 172.18.0.3#53(172.18.0.3)
;; WHEN: Fri Apr 22 22:41:07 CST 2022
;; MSG SIZE rcvd: 93
```

0.4.5 重新启动BIND服务器并进行测试

- 重启BIND服务后，使用 `dig www.xubiang.com` 命令向本地DNS服务器询问 `www.xubiang.com` 的IP地址，结果见上图，能够正确解析。