Kali Linux 渗透测试实战 2.1 DNS 信息收集

从本节开始，我们从头开始，系统的学习基于 Kali Linux 的 web 应用渗透测试。 本章主要目标是从各个角度搜集测试目标的基本信息，包括搜集信息的途径、各种工具的使用方法，以及简单的示例。 按照循序渐进的原则，第一节讲解如何搜集 DNS 信息。对于工具的使用，我这里不打算把使用说明再搬到这里，意义不大。读者希望 google 就可以了。 如果您对 DNS 的工作原理不是很了解，我建议您先在网上或者书籍上查阅相关资料。本节也对相关概念做了简单诠释，作为学习的辅助。

目录

2.1 DNS 信息收集

从本节开始，我们从头开始，系统的学习基于 Kali Linux 的 web 应用渗透测试。

本章主要目标是从各个角度搜集测试目标的基本信息，包括搜集信息的途径、各种工具的使用方法，以及简单的示例。

按照循序渐进的原则，第一节讲解如何搜集 DNS 信息。对于工具的使用，我这里不打算把使用说明再搬到这里，意义不大。读者希望 google 就可以了。

如果您对 DNS 的工作原理不是很了解，我建议您先在网上或者书籍上查阅相关资料。本节也对相关概念做了简单诠释，作为学习的辅助。

关于 DNS（参考：http://zh.wikipedia.org/zh-cn/%E5%9F%9F%E5%90%8D%E7%B3%BB%E7%BB%9F ;http://man.ddvip.com/linux/debian/bin9/bind9-conf-2.html）：

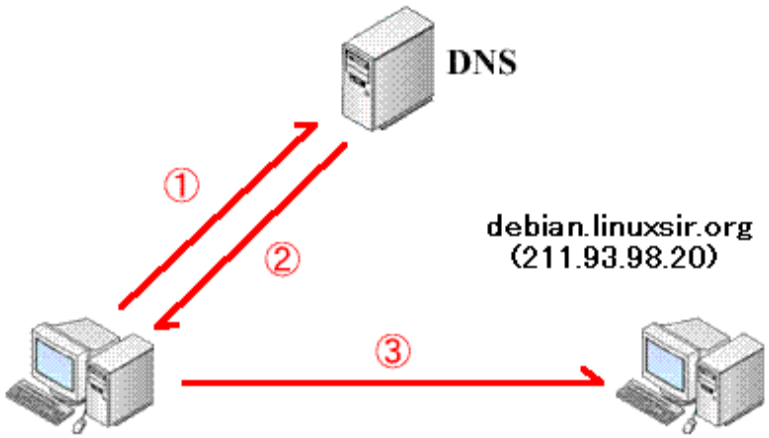
域名系统（英文：Domain Name System，DNS）是因特网的一项服务，它作为将域名和 IP 地址相互映射的一个分布式数据库，能够使人更方便的访问互联网。DNS 使用 TCP 和 UDP 端口 53。当前，对于每一级域名长度的限制是 63 个字符，域名总长度则不能超过 253 个字符。

DNS 命名用于 Internet 等 TCP/IP 网络中，通过用户友好的名称查找计算机和服务。当用户在应用程序中输入 DNS 名称时，DNS 服务可以将此名称解析为与之相关的其他信息，如 IP 地址。

例如，多数用户喜欢使用友好的名称（如 debian.linuxsir.org）来查找计算机，如网络上的邮件服务器或 Web 服务器。友好名称更容易了解和记住。

但是，计算机使用数字地址在网络上进行通讯。为更容易地使用网络资源，DNS 等命名系统提供了一种方法，将计算机或服务的用户友好名称映射为数字地址。

下图显示了 DNS 的基本用途，即根据计算机名称查找其 IP 地址。



本例中，客户端计算机查询 DNS 服务器，要求获得某台计算机（Debian.linuxsir.org）的 IP 地址。由于 DNS 服务器能够根据其本地数据库应答此查询，因此，它将以包含所请求信息的应答来回复客户端，即一条主机（A）资源记录，其中含有 Debian.linuxsir.org 的 IP 地址信息(211.93.98.20)。

此例显示了单个客户端与 DNS 服务器之间的简单 DNS 查询。实际上，DNS 查询要复杂得多，包含此处未显示的许多其他步骤。

当 DNS 客户端需要查询程序中使用的名称时，它会查询 DNS 服务器来解析该名称。客户端发送的每条查询消息都包括三条信息，指定服务器回答的问题：

* 指定的 DNS 域名，规定为完全合格的域名（FQDN）

* 指定的查询类型，可根据类型指定资源记录，或者指定查询操作的专用类型。

* DNS 域名的指定类别。

例如，指定的名称可为计算机的 FQDN，如 Debian.linuxsir.org ，并且指定的查询类型用于通过该名称搜索地址 (A) 资源记录。将 DNS 查询看作客户端向服务器询问由两部分组成的问题，如“您是否拥有名为‘Debian.linuxsir.org’ 的计算机的 A 资源记录？” 当客户端收到来自服务器的应答时，它将读取并解释应答的 A 资源记录，获取根据名称询问的计算机的 IP 地址。

DNS 查询以各种不同的方式进行解析。有时，客户端也可使用从先前的查询获得的缓存信息在本地应答查询。DNS 服务器可使用其自身的资源记录信息缓存来应答查询。DNS 服务器也可代表请求客户端查询或联系其他 DNS 服务器，以便完全解析该名称，并随后将应答返回至客户端。这个过程称为递归。

另外，客户端自己也可尝试联系其他的 DNS 服务器来解析名称。当客户端执行此操作时，它会根据来自服务器的参考答案，使用其他的独立查询。这个过程称为迭代。

总之，DNS 查询进程分两部分进行：

* 名称查询从客户端计算机开始 ,并传输至解析程序即 DNS 客户端服务程序进行解析。

* 不能在本地解析查询时，可根据需要查询 DNS 服务器来解析名称。

记录类型

主条目：域名服务器记录类型列表

DNS 系统中，常见的资源记录类型有：

主机记录(A 记录)：RFC 1035 定义，A 记录是用于名称解析的重要记录，它将特定的主机名映射到对应主机的 IP 地址上。

别名记录(CNAME 记录): RFC 1035 定义，CNAME 记录用于将某个别名指向到某个 A 记录上，这样就不需要再为某个新名字另外创建一条新的 A 记录。

IPv6 主机语录(AAAA 记录): RFC 3596 定义，与 A 记录对应，用于将特定的主机名映射到一个主机的 IPv6 地址。

服务位置记录(SRV 记录): RFC 2782 定义 ,用于定义提供特定服务的服务器的位置，如主机(hostname)，端口(port number)等。

NAPTR 记录: RFC 3403 定义，它提供了正则表达式方式去映射一个域名。NAPTR 记录非常著名的一个应用是用于 ENUM 查询。

完整的记录类型列表参考：dns 记录类型

2.1.1 whois 查询

WHOIS（域名数据库查询）

一个域名的所有者可以通过查询 WHOIS 数据库而被找到 ;对于大多数根域名服务器， 基本的 WHOIS 由 ICANN 维护，而 WHOIS 的细节则由控制那个域的域注册机构维护。

对于 240 多个国家代码顶级域名(ccTLDs)，通常由该域名权威注册机构负责维护 WHOIS。例如中国互联网络信息中心(China Internet Network Information Center)负责 .CN 域名的 WHOIS 维护，香港互联网注册管理有限公司(Hong Kong Internet Registration Corporation Limited) 负责 .HK 域名的 WHOIS 维护，台湾网络信息中心 (Taiwan Network Information Center) 负责 .TW 域名的 WHOIS 维护。

提供 whois 查询的站点很多 google“whois”，你可以得到这些站点。

另外所有的域名提供商都提供 whois 信息查询。比如在万网查询“iprezi.cn”，会得到如下信息：

域名 DomainName	iprezi.cn 访问该网站
域名状态 (这是什么?) Domain Status	clientTransferProhibited
注册商 Sponsoring Registrar	北京万网志成科技有限公司
注册人 Company	汪斌
邮箱 Email	philewong1985@126.com
DNS 服务器 Name Server	dns21.hichina.com, dns22.hichina.com
注册日期 Registration Date(UTC+08:00)	2013-01-21 10:23:57
到期日期 Expiration Date(UTC+08:00)	2015-01-21 10:23:57

在 whois 查询中，注册人姓名和邮箱信息，通常对于测试个人站点非常有用，因为我们可以通过搜索引擎，社交网络，挖掘出很多域名所有人的信息。而对于小站点而言，域名所有人往往就是管理员。

对于大型站点，我们更关心 DNS 服务器，很多公司都会有自己的域名服务器，这些服务器可以成为渗透测试过程中的一个突破点。

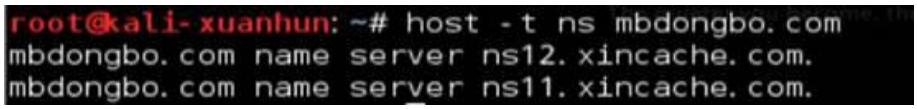
2.1.2 域名基本信息查询

Dns 服务器查询

除了 whois 查询之外，我们还可以通过 host 命令来查询 dns 服务器，命令格式为：

host -t ns domainName

如下图：



通过“host –t ns mbdongbo.com”得到该域名的两个服务器为 ns12.xincache.com，ns11.xincache.com。

a 记录查询

A (Address) 记录是用来指定主机名（或域名）对应的 IP 地址记录。用户可以将该域名下的网站服务器指向到自己的 web server 上。同时也可以设置您域名的子域名。通俗来说 A 记录就是服务器的 IP,域名绑定 A 记录就是告诉 DNS,当你输入域名的时候给你引导向设置在 DNS 的 A 记录所对应的服务器。

通过

host -t a domainName

可以查询 a 记录

mx 记录查询

MX 记录也叫做邮件路由记录，用户可以将该域名下的邮件服务器指向到自己的 mail server 上，然后即可自行操控所有的邮箱设置。您只需在线填写您服

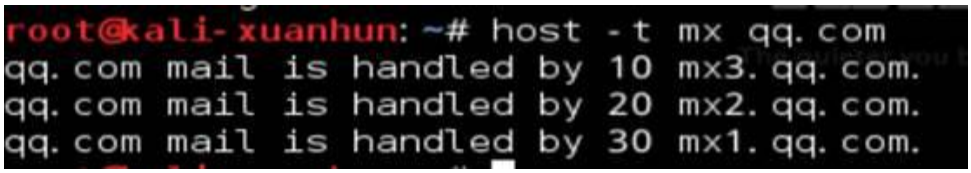
务器的 IP 地址，即可将您域名下的邮件全部转到您自己设定相应的邮件服务器上。

简单的说，通过操作 MX 记录，您才可以得到以您域名结尾的邮局。

通过

host -t mx domainName

可以查询该域名下的 mx 记录，从而可以得到邮件服务器信息。



2.1.3 域名枚举

在得到主域名信息之后，如果能通过主域名得到所有子域名信息，在通过子域名查询其对应的主机 IP，这样我们能得到一个较为完整的信息。

fierse

使用 fierse 工具，可以进行域名列表查询：

fierce -dns domainName

如上图，通过 fierse，成功枚举出某域名下的子域名列表。

关于 fierse 的工作原理，可以查看：http://ha.ckers.org/fierce/。

除 fierse 之外，dnsdict6、dnsenum、dnsmap 都可以进行域名枚举，需要说明的是，每个工具返回的结果并不相同，而且有的工具还有错误，读者进行 dns 信息搜集的时候，要尽量使用不同的工具，尽可能得到完整的信息。dnsdict6、dnsenum、dnsmap 进行枚举的时候都是使用字典，进行扫描，这里以 dnsdict6 为例。

dnsdict6

dnsdict6 使用你提供的一个字典或者内置的列表来枚举，基于 dnsmap。

使用语法：

dnsdict6 [-d46] [-s|-m|-l|-x] [-t 线程] [-D] 域名 [字典路径]

参数说明:

-4 显示 ipv4

-t 指定要使用的线程 默认：8 最大:32

-D = = = = = = = = = = = = = = = = = [只显示字典不扫描] = = = =

-d 显示在 DNS 服务器上的 NS（一种服务记录类型）MX（邮件服务器）ipv6 的域名信息

- [smlx] 选择字典大小 [内置的] -s 小型是 50 条 - m 中等是 796 条[默认] -l 大型 1416 条 - x 最大 3211 条

示例：

2.1.4 反向地址解析

（参考：http://blog.csdn.net/jackxinxu2100/article/details/8145318）

我们经常使用到得 DNS 服务器里面有两个区域，即“正向查找区域”和“反向查找区域”，正向查找区域就是我们通常所说的域名解析，反向查找区域即是这里所说的 IP 反向解析，它的作用就是通过查询 IP 地址的 PTR 记录来得到该 IP 地址指向的域名，当然，要成功得到域名就必需要有该 IP 地址的 PTR 记录。PTR 记录是邮件交换记录的一种，邮件交换记录中有 A 记录和 PTR 记录，A 记录解析名字到地址，而 PTR 记录解析地址到名字。地址是指一个客户端的 IP 地址，名字是指一个客户的完全合格域名。通过对 PTR 记录的查询，达到反查的目的。

反向域名解析系统(Reverse DNS)的功能确保适当的邮件交换记录是生效的。反向域名解析与通常的正向域名解析相反，提供 IP 地址到域名的对应。IP 反向解析主要应用到邮件服务器中来阻拦垃圾邮件，特别是在国外。多数垃圾邮件发送者使用动态分配或者没有注册域名的 IP 地址来发送垃圾邮件，以逃避追踪，使用了域名反向解析后，就可以大大降低垃圾邮件的数量。

比如你用 xxx@name.com 这个邮箱给我的邮箱 123@163.com 发了一封信。163 邮件服务器接到这封信会查看这封信的信头文件，这封信的信头文件会显示这封信是由哪个 IP 地址发出来的。然后根据这个 IP 地址进行反向解析，

如果反向解析到这个 IP 所对应的域名是 name.com 那么就接受这封邮件，如果反向解析发现这个 IP 没有对应到 name.com，那么就拒绝这封邮件。

由于在域名系统中，一个 IP 地址可以对应多个域名，因此从 IP 出发去找域名，理论上应该遍历整个域名树，但这在 Internet 上是不现实的。为了完成逆向域名解析，系统提供一个特别域，该特别域称为逆向解析域 in-addr.arpa。这样欲解析的 IP 地址就会被表达成一种像域名一样的可显示串形式，后缀以逆向解析域域

名"in-addr.arpa"结尾。

例如一个 IP 地址：222.211.233.244，其逆向域名表达方式为：244.233.221.222.in-addr.arpa

两种表达方式中 IP 地址部分顺序恰好相反，因为域名结构是自底向上(从子域到域)，而 IP 地址结构是自顶向下(从网络到主机)的。实质上逆向域名解析是将 IP 地址表达成一个域名,以地址做为索引的域名空间,这样逆向解析的很大部分可以纳入正向解析中。

linux 中常用的反向解析工具为 nslookup 和 dig。

使用 dig 进行反向解析的命令格式为：

dig -x ip @dnsserver #用 dig 查看反向解析

其中 dnsserver 可以不用指定 默认会使用本机配置的域名服务器进行反向查询。指定 dsn 服务器示例如下图：

```
root@kali-xuanhun:~# dig -x 121.101.223.214 @ns12.xincache.com

; <<>> DiG 9.8.4-rpz2+rl005.12-P1 <<>> -x 121.101.223.214 @ns12.xincache.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: SERVFAIL, id: 63115
;; flags: qr rd; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;214.223.101.121.in-addr.arpa.  IN  PTR

;; Query time: 52 msec
;; SERVER: 121.14.250.37#53(121.14.250.37)
;; WHEN: Mon Dec 23 22:33:53 2013
;; MSG SIZE  rcvd: 46
```

不指定 dns 服务：

```
root@kali-xuanhun:~# dig -x 121.101.223.214

; <<>> DiG 9.8.4-rpz2+rl005.12-P1 <<>> -x 121.101.223.214
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 50825
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADD

;; QUESTION SECTION:
```

但是实际情况并不是尽如人意，查找的服务器不同，得到的结果的完整度也不同，比如上图的两个测试，都没有得到想要的结果。很多时候，我们到提供反向查询的网站进行查找，可能效果会更好一点。

下面是我在 http://dns.aizhan.com/的查询结果：

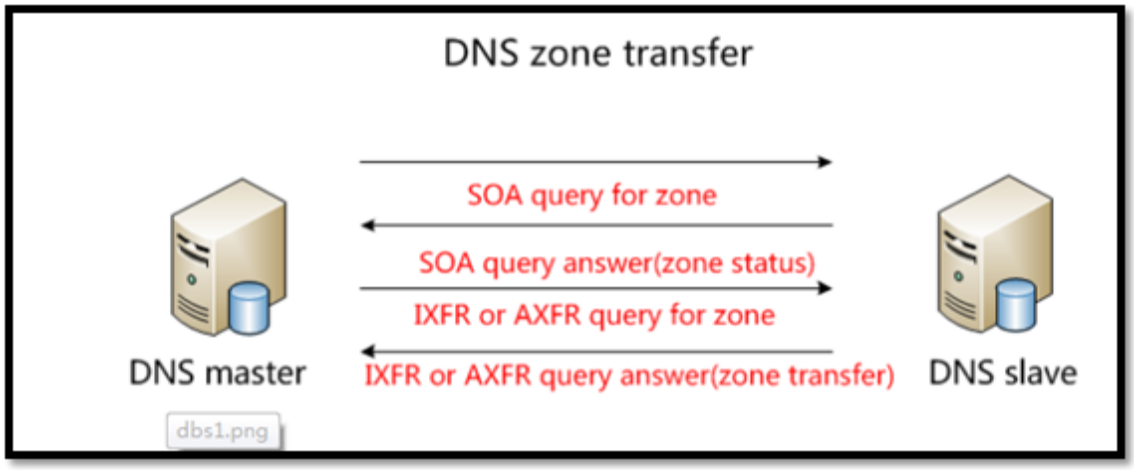
而在 www.lbase.net 的查询结果为：

所以想要获得完整的信息，可以多尝试不同的工具，整合结果。很多工具无法做反向查询的原因，在于域名所有者没有添加反向解析记录。

2.1.5 关于 DNS 区域传送漏洞

很多 dns 探测工具，都会首先尝试 dns 区域传送，然后才是暴力枚举，那么什么是 DNS 区域传送漏洞呢？

区域传送操作指的是一台后备服务器使用来自主服务器的数据刷新自己的 zone 数据库。这为运行中的 DNS 服务提供了一定的冗余度，其目的是为了防止主域名服务器因意外故障变得不可用时影响到全局。一般来说，DNS 区域传送操作只在网络里真的有后备域名 DNS 服务器时才有必要执行，但许多 DNS 服务器却被错误地配置成只要有人发出请求，就会向对方提供一个 zone 数据库的拷贝。如果所提供的信息只是与连到因特网上且具备有效主机名的系统相关，那么这种错误配置不一定是坏事，尽管这使得攻击者发现潜在目标要容易得多。真正的问题发生在一个单位没有使用公用/私用 DNS 机制来分割外部公用 DNS 信息和内部私用DNS信息的时候 此时内部主机名和IP地址都暴露给了攻击者。把内部 IP 地址信息提供给因特网上不受信任的用户，就像是把一个单位的内部网络完整蓝图或导航图奉送给了别人。



使用 dig 工具可以检测 dns 区域传送漏洞，语法如下：

dig axfr @域名服务器 被检测域名

示例：

root@kali-xuanhun:~# dig @wormhole.movie.edu movie.edu axfr

; <<>> DiG 9.8.4-rpz2+rl005.12-P1 <<>> @wormhole.movie.edu movie.edu axfr

; (1 server found)

;; global options: +cmd

;; connection timed out; no servers could be reached

root@kali-xuanhun:~# dig axfr @ns12.zoneedit.com zonetransfer.me

; <<>> DiG 9.8.4-rpz2+rl005.12-P1 <<>> axfr @ns12.zoneedit.com zonetransfer.me

; (1 server found)

;; global options: +cmd

zonetransfer.me. 7200 IN SOA ns16.zoneedit.com. soacontact.zoneedit.com. 2013064418 2400 360 1209600 300

zonetransfer.me. 7200 IN NS ns16.zoneedit.com.

zonetransfer.me. 7200 IN NS ns12.zoneedit.com.

zonetransfer.me. 7200 IN A 217.147.180.162

zonetransfer.me. 7200 IN MX 0 ASPMX.L.GOOGLE.COM.

zonetransfer.me. 7200 IN MX 10 ALT1.ASPMX.L.GOOGLE.COM.

zonetransfer.me. 7200 IN MX 10 ALT2.ASPMX.L.GOOGLE.COM.

zonetransfer.me. 7200 IN MX 20 ASPMX2.GOOGLEMAIL.COM.

zonetransfer.me. 7200 IN MX 20 ASPMX3.GOOGLEMAIL.COM.

zonetransfer.me. 7200 IN MX 20 ASPMX4.GOOGLEMAIL.COM.

zonetransfer.me. 7200 IN MX 20 ASPMX5.GOOGLEMAIL.COM.

zonetransfer.me. 301 IN TXT "Remember to call or email Pippa on +44 123 4567890 or pippa@zonetransfer.me when making DNS changes"

zonetransfer.me. 301 IN TXT "google-site-verification=tyP28J7JAUHA9fw2sHXMgcCC0I6XBmmoVi04VlMewxA"

testing.zonetransfer.me. 301 IN CNAME www.zonetransfer.me.

164.180.147.217.in-addr.arpa.zonetransfer.me. 7200 IN PTR www.zonetransfer.me.

ipv6actnow.org.zonetransfer.me. 7200 IN AAAA 2001:67c:2e8:11::c100:1332

asfdbauthdns.zonetransfer.me. 7900 IN AFSDB 1 asfdbbox.zonetransfer.me.

office.zonetransfer.me. 7200 IN A 4.23.39.254

owa.zonetransfer.me. 7200 IN A 207.46.197.32

info.zonetransfer.me. 7200 IN TXT "ZoneTransfer.me service provided by Robin Wood - robin@digininja.org. See www.digininja.org/projects/zonetransferme.php for more information."

asfdbbox.zonetransfer.me. 7200 IN A 127.0.0.1

canberra_office.zonetransfer.me. 7200 IN A 202.14.81.230

asfdbvolume.zonetransfer.me. 7800 IN AFSDB 1 asfdbbox.zonetransfer.me.

email.zonetransfer.me. 2222 IN NAPTR 1 1 "" "E2U+email" "" email.zoneedit.com.zonetransfer.me.

dzc.zonetransfer.me. 7200 IN TXT "AbCdEfG"

dr.zonetransfer.me. 300 IN LOC 53 20 56.558 N 1 38 33.526 W 0.00m 1m 10000m 10m

rp.zonetransfer.me. 321 IN RP robin.zonetransfer.me.zonetransfer.me. robinwood.zonetransfer.me.

sip.zonetransfer.me. 3333 IN NAPTR 2 3 "au" "E2U+sip" "!^.*$!sip:customer-service@zonetransfer.me!" .

alltcpportsopen.firewall.test.zonetransfer.me. 301 IN A 127.0.0.1

www.zonetransfer.me. 7200 IN A 217.147.180.162

staging.zonetransfer.me. 7200 IN CNAME www.sydneyoperahouse.com.

deadbeef.zonetransfer.me. 7201 IN AAAA dead:beaf::

robinwood.zonetransfer.me. 302 IN TXT "Robin Wood"

vpn.zonetransfer.me. 4000 IN A 174.36.59.154

_sip._tcp.zonetransfer.me. 14000 IN SRV 0 0 5060 www.zonetransfer.me.

dc_office.zonetransfer.me. 7200 IN A 143.228.181.132

zonetransfer.me. 7200 IN SOA ns16.zoneedit.com. soacontact.zoneedit.com. 2013064418 2400 360 1209600 300

;; Query time: 425 msec

;; SERVER: 209.62.64.46#53(209.62.64.46)

;; WHEN: Tue Dec 24 14:12:21 2013

;; XFR size: 37 records (messages 37, bytes 2673)

小结

运用 DNS 信息探测，结合社会工程方法，我们可以得到关于网站拥有者、服务器基本组织结构等方面的信息。

我故意淡化了各种工具的详细使用方法，因为如果把每种工具都详细的罗列出来篇幅过长，同时也没这个必要，读者可以很方便的在网络上找到每种工具的使用手册。

DNS 记录类型有几十种，我这里只是列出我认为重要的信息，希望读者能查看我给出的链接。