> `DNS(Domain Name System)`,网域名称系统，是互联网上基础性的服务，DNS 系统早在1983年由保罗·莫卡派乔斯发明。DNS将域名和网络服务器的IP地址相互映射，并将这些数据保存至 DNS 服务器。两台主机的通信事实上是依赖于 IP 地址，而不是域名，IP地址非常不方便于人类记忆，而域名的产生，极大的方便了访问互联网，但随着 IP 地址和主机的增加，由少量的服务器负责解析域名显然是不堪重负的，所以 DNS 名称系统使用分层级的思想，当有人发出请求解析时，自上而下的一级一级进行解析。
>
> 例如，有一台服务器的IP为 115.159.22.21，当我们访问这台服务器的资源，输入 www.xuejinwei.com 这个主机名后，DNS服务器自动将域名"翻译"成 IP 地址，然后访问网络的服务器。DNS 是一个域名和 IP 地址相互映射关系的分布式数据库。事实上在浏览器中输入 www.xuejinwei.com 域名，操作系统会先检查自己本地的 hosts 文件是否有这个网址映射关系，如果有，就先调用这个IP地址映射，完成域名解析。DNS 是一个应用层的一个协议，基于 53/UDP 端口工作。

相关概念

域名的组织结构

> `ICANN(The Internet Corporation for Assigned Names and Numbers)` 互联网名称与数字地址分配机构，负责在全球范围内对互联网通用顶级域名 (gTLD) 以及国家和地区顶级域名 (ccTLD) 系统的管理、以及根服务器系统的管理。域名采用层次化结构进行组织，如上图。每个点代表一个层次，例如 `www.xuejinwei.com.` 最后面的 "." 代表根域，常常省略。 `.com` 是顶级域。 `xuejinwei.com` 是二级域。
>
> `子域`是一个相对概念，是更大网域的一部分。例如 `xuejinwei.com` 是 `.com` 的子域， `www.xuejinwei.com`是 `xuejinwei.com` 是子域，子域也是一个网域。

DNS查询方式

> **递归查询**：客户端向本地 DNS 服务器发出请求后，一直处于等待状态，直到本地名称服务器返回查询结果。以访问 www.xuejinwei.com 为例叙述递归查询过程。当客户端向本地 DNS 服务器发出请求后，本地 DNS 服务器查询本机缓存，如果有记录，则直接返回；如果没有，则本地 DNS 服务器以客户端的身份将查询请求发给根名称服务器，根名称服务器通过查询返回给本地 DNS 服务器 .com 顶级名称服务器的IP地址；本地DNS服务器收到 .com 顶级名称服务器的IP地址后继续向`.com`顶级名称服务器发出请求，顶级名称服务器收到请求后查询缓存，如果有记录则直接返回本地DNS服务器，如果没有，则返回`xuejinwe.com`二级名称服务器的IP地址；本地名称服务器继续发出请求，二级名称服务器同样查找缓存，返回`www.xuejinwei.com`的IP地址。
>
> **迭代查询**：客户端和本地DNS服务器的查询方式就属于迭代查询，客户端发出查询请求后，处于等待状态，直到本地DNS服务器返回确定回复或否定答复。

DNS服务器类型

> 1. **主名称服务器(primary name server)**。从域管理员构造的本地磁盘文件中加载域信息，该文件（区域文件）包含着该服务器具有管理权的一部分域结构的最精确信息。主服务器是一种权威性服务器，因为它以绝对的权威去回答对其管辖域的任何查询。
> 2. **从名称服务器(secondary name server)**。它可从主服务器中复制一整套域信息。区域文件是从主服务器中复制出来的，并作为本地磁盘文件存储在辅助服务器中。这种复制称为"区域文件复制"。在辅助域名服务器中有一个所有域信息的完整拷贝，可以有权威地回答对该域的查询。因此，辅助域名服务器也称作权威性服务器。配置辅助域名服务器不需要生成本地区文件，因为可以从主服务器中下载该区文件。

3. **缓存名称服务器(caching-only server)**。可运行域名服务器软件，但是没有域名数据库软件。它从某个远程服务器取得域名服务器查询的结果，一旦取得一个，就将它放在高速缓存中，以后查询相同的信息时就用它予以回答。高速缓存服务器不是权威性服务器，因为它提供的所有信息都是间接信息。当 BIND 被配置为缓存服务器的时候，它只会回应已缓存的请求，并将所有其他的请求转发到上游的 DNS 服务器。缓存名称服务器只需要`.`这个zone file文件即可。
4. **转发名称服务器(forwarding DNS)**。转发域名服务器需要指定上层DNS服务器作为转发目标，转发名称服务器甚至不需要`.`这个zone file文件，全部查询都交给上层DNS服务器进行查询。

解析类型及结果

> 正向解析：通过域名查找 IP 的过程
>
> 反向解析：通过 IP 查找域名的过程
>
> 肯定答案：正确返回了发出请求的内容
>
> 否定答案：请求的条目不存在等原因无法返回的结果，叫否定答案
>
> 权威答案：由负责此域的解析的 DNS 服务器返回的解析结果
>
> 非权威答案：从非权威服务器上获得的的解析结果

FQDN

`FQDN(Fully Qualified Domain Name)` 完全合格域名/全称域名，唯一地标识在 DNS 分层树中的主机的位置，通过指定的路径中点分隔从根引用的主机的名称列表。

DNS资源记录类型

> DNS服务器数据库中每一个条目都是一个资源记录（Resource Record，RR）。资源记录格式中的域名可以写成 FQDN格式，也可以写成主机名，写成主机名格式时，会自动补上区域名称，@表示区域名称。

SOA

> SOA(Start of Authority)，SOA 名叫起始授权机构记录，SOA 记录说明了在众多NS记录里那一台才是主名称服务器。更通俗的讲就是：当前解析库被哪个域所用，由那个 DNS 服务器负责。

```
# 格式
ZONE_NAME    TTL IN  SOA NS_FQDN admin_mailbox   (
             sereial number
             refresh time
             retry time
             ecpire time
             negative answer TTL )
# 示例
@   600 IN  SOA dns.xuejinwei.com.
jinweiayy.gmail.com (
             2017092201
             1H
             2M
             1W
             1D )
```

> `sereial number`：";"号后面为注释。serial number，序列号，x从服务器同步依据，主服务器记录发生变化，序列号加1，序列号不能超过10位，例：2017022001
>
> `refresh`：refresh time 刷新时间，每隔多久到主服务器检查一次资源记录是否改变，时间单位默认是秒
>
> `retry`：retry time 重试时间，小于refresh time。从服务器从主服务器请求同步解析库失败时，再次尝试的时间间隔

> `exprire`：ecpire time 过期时间。在此时间后，如果从服务器一直无法连接主服务器，则从服务器的区域数据过期，主动停止自己的解析职责
>
> `negative answer TTL`：negative answer TTL 否定答案的缓存 TTL 时间。否定答案缓存失效时间
>
> **注意：**ZONE_NAME为区域名称，例如：xuejinwei.com.，通常可以简写为@符号；时间单位：默认是秒。M是分钟、H是小时、D是天；SOA必须是区域数据库文件的第一条记录；邮箱中出现`@`符号换成`.`符号；可以使用宏定义例如：`$TTL 600`表示默认的 TTL 值。

NS

> NS(Name Server)，域名服务器记录，用来指定该域名由哪个DNS服务器来进行解析。NS记录必须要有一条A记录，指明DNS服务器的IP地址。NS记录可以写多条。

```
# 示例
@   600 IN  NS  dns.xuejinwei.com.
# 或者
xuejinwei.com.  600     IN      NS
ns1.xuejinwei.com.
```

A

> `A`记录也称为主机记录。 FQDN 指向一个IPv4的IP地址，一个域名可以创建多个不同的IPv4地址FQDN可以使用相对名称，例如将 www.xuejinwei.com. 写成 www 。

```
# 示例
www IN  A   xxx.xxx.xxx.xxx
```

> **注意：**同一个主机名可以有多条不同的A记录，DNS 会以轮询的方式解析主机名；多个主机名可以指向同一个值；*.xuejinwei.com. 表示泛域名解析。

AAAA

> `AAAA`记录将DNS域名指向一个IPv6的IP地址。

MX

> `MX`(Mail exchanger)，邮件服务记录。

```
# 示例
xuejinwei.com.  600 IN  MX  10  mail.xuejinwei.com.
mail.xuejinwei.com. 600 IN  MX  xxx.xxx.xxx.xxx
```

> **注意：**一个区域内，MX记录可有多个，每个记录 VALUE 之前应该有一个数字0-99，表示此服务器的优先级，数字越小，优先级越高；任何一个MX资源记录，都应该有一个A记录。

PTR

> (PoinTeR)，将一个 IP 地址解析为 FQDN，(IP --> FQDN)。

```
# 格式
NAME    TTL IN  RRType  VALUE
# 示例
7.100.16.172.in-addr.arpa.  IN  PTR
www.xuejinwei.com.
```

> **注意：**NAME：IP，逆向的主机地址，即主机地址要反过来写例如：172.16.100.7 的 name 为：7.100
> VALUE：FQDN

CNAME

> CNAME（Canonical Name)别名记录，由一个域名指向另一个域名，DNS系统将继续解析别名指向的域名。

```
# 示例
www.xuejinwei.com.  IN  CNAME   www.xuejinwei.me
```

DNS报文

> DNS报文格式，不论是请求报文，还是DNS服务器返回的应答报文，都使用统一的格式。不废话，先上图，通过Wireshark 抓 DNS 请求响应包，再对照下面对DNS报文的说明，就一目了然了。
>
> **DNS请求报文**
>
> **DNS响应报文**

DNS报文格式

> Header　　报文头12字节
> Question　　查询的问题
> Answer　　应答(响应报文)
> Authority　　授权应答(响应报文)
> Additional　　附加信息(响应报文)

DNS报文格式头

```
ID　　2个字节(16bit)，标识字段，客户端会解析服务器返回的DNS应答报文，获取ID值与请求报文设置的ID值做比较,如果相同，则认为是同一个DNS会话。
FLAGS　　2个字节(16bit)的标志字段。包含以下属性:
QR　　0表示查询报文，1表示响应报文;
opcode　通常值为 0(标准查询)，其他值为 1(反向查询) 2(服务器状态请求) [3,15]保留值;
AA　　表示授权回答(authoritative answer)- 这个比特位在应答的时候才有意义，指出给出应答的服务器是查询域名的授权解析服务器;
TC　　表示可截断的(truncated)-用来指出报文比允许的长度还要长，导致被截断;
RD　　表示期望递归(Recursion Desired) - 这个比特位被请求设置，应答的时候使用的相同的值返回。如果设置了RD，就建议域名服务器进行递归解析，递归查询的支持是可选的;
RA　　表示支持递归(Recursion Available) - 这个比特位在应答中设置或取消，用来代表服务器是否支持递归查询;
Z　　保留值，暂未使用;
```

```
RCODE    应答码(Response code) - 这4个比特位在应答报文中设置，代表的含义如下：
                0 ：没有错误。
                1 ：报文格式错误(Format error) - 服务器不能理解请求的报文；
                2 ：服务器失败(Server failure) - 因为服务器的原因导致没办法处理这个请求；
                3 ：名字错误(Name Error) - 只有对授权域名解析服务器有意义，指出解析的域名不存在；
                4 ：没有实现(Not Implemented) - 域名服务器不支持查询类型；
                5 ：拒绝(Refused) - 服务器由于设置的策略拒绝给出应答.比如，服务器不希望对某些请求者给出应答，或者服务器不希望进行某些操作（比如区域传送zone transfer）；
                [6,15] ：保留值，暂未使用。
QDCOUNT      无符号16bit整数表示报文请求段中的[问题记录数]
ANCOUNT      无符号16bit整数表示报文回答段中的[回答记录数]
NSCOUNT      无符号16bit整数表示报文授权段中的[授权记录数]
ARCOUNT      无符号16bit整数表示报文附加段中的[附加记录数]
```

参考

维基百科-域名：https://zh.wikipedia.org/wiki/%E5%9F%9F%E5%90%8D

DNS 的 SOA 记录简介:http://www.sigma.me/2011/01/01/about_dns_soa.html