dns

**DNS协议运行在UDP协议之上，使用端口号53。**

**DNS是： (将用户提供的主机名解析为IP地址)**

1、一个由分层的DNS服务器实现的分布式数据库

2、一个使得主机能够查询分布式数据库的应用层协议

**目前的DNS服务器大致分为3种类型的DNS服务器：**

根DNS服务器、顶级域名DNS服务器、权威DNS服务器

除此之外，还有一种很重要的DNS，成为本地DNS服务器

缓存: 有DNS的地方，就有缓存。浏览器、操作系统、本地 DNS服务器、根DNS服务器，它们都会对DNS结果做一定程度的缓存。

DNS查询链：本地DNS服务器-->根DNS服务器-->顶级域DNS服务器-->权威DNS服务器

1、发起基于域名的请求后，首先检查**本地缓存**（浏览器缓存-->操作系统的hosts文件）

2、如果本地缓存中有，直接返回目标IP地址，否则将域名解析请求发送给**本地DNS服务器**

3、如果本地DNS服务器中有，直接返回目标IP地址，到这一步基本能解析80%的域名。如果没有，本地DNS服务器将解析请求发送给**根DNS服务器**

4、根DNS服务器会返回给本地DNS服务器一个所查询的顶级域名DNS服务器地址

5、本地DNS服务器再向上一步返回的**顶级域名DNS服务器**发送请求，顶级域名DNS服务器查询并返回域名对应的权威DNS服务器的地址

6、本地DNS服务器再向上一步返回的**权威DNS服务器**发送请求，权威域名服务器会查询存储的域名和IP的映射关系表，将IP连同一个TTL（过期时间）值返回给本地DNS服务器

7、**本地DNS服务器**会将IP和主机名的映射**保存**起来，保存时间由TTL来控制

8、本地DNS服务器把解析的结果返回给用户，用户根据TTL值缓存在**本地系统缓存**中，域名解析过程结束

巧记:

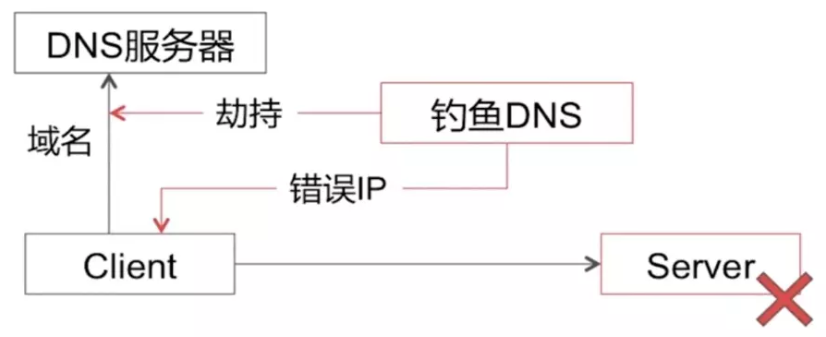
本地缓存 → 本地DNS服务器 → 根DNS服务器 → 顶级域名DNS服务器 → 权威DNS服务器 → 本地DNS服务器保存 →用户操作系统的host文件缓存

 从请求主机到本地DNS服务器的查询是递归，其余查询是迭代的这种方式。

调用系统缓存需要跨进程，消耗大，所以就有浏览器DNS缓存。

DNS解析安全问题

1. DNS劫持



一种可能的域名劫持方式即黑客侵入了宽带路由器并对终端用户的本地DNS服务器进行篡改，指向黑客自己伪造的本地DNS服务器，进而通过控制本地DNS服务器的逻辑返回错误的IP信息进行域名劫持。

**(返回错误信息使用户不能访问想访问的服务器)**

另一方面，由于DNS解析主要是基于UDP协议，除了上述攻击行为外，攻击者还可以监听终端用户的域名解析请求，并在本地DNS服务器返回正确结果之前将伪造的DNS解析响应传递给终端用户，进而控制终端用户的域名访问行为。

**(返回指定的信息,让用户访问黑客希望的服务器)**

==>如何解决DNS劫持？

DNS解析发生在HTTP协议之前，DNS解析和DNS劫持和HTTP没有关系

要想解决DNS劫持：

①可以使用HttpDNS的方案：使用 HTTP协议向DNS服务器的80端口进行请求,来规避DNS劫持,比如：http://119.29.29.29/d?dn=domain&ip=clientIp

②在终端上，可以更换DNS服务器，不管手机还是电脑，都能手动配置DNS

2、缓存污染（DNS污染）。

我们知道在接收到域名解析请求时，本地DNS服务器首先会查找缓存，如果缓存命中就会直接返回缓存结果，不再进行递归DNS查询。这时候如果本地DNS服务器针对部分域名的缓存进行更改，比如将缓存结果指向第三方的广告页，就会导致用户的访问请求被引导到这些广告页地址上。