



链滴

# NAT, NAPT, ALG 和你的内网 ip 和外网 ip 有啥关系

作者: [kakj-go](#)

原文链接: <https://ld246.com/article/1571906823981>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<h3 id="问题1--内网和外网ip的区别">问题 1: 内网和外网 ip 的区别</h3>

<p>内网 ip 其实也是一种外网 ip, 只是这种 ip 被因特网设定为私有 ip 地址, 不会被中央路由器和边界路由器进行路由转发, 其中有以下几个地址段被分为私有 ip 地址</p>

<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">A 类: 10.0.0.0 ~ 10.255.255.255

</span></span><span class="highlight-line"><span class="highlight-cl">B 类: 172.16.0.0 172.31.255.255

</span></span><span class="highlight-line"><span class="highlight-cl">C 类: 192.168.0. ~ 192.168.255.255

</span></span></code></pre>

<h3 id="问题2--啥是中央路由器-啥是边界路由器">问题 2: 啥是中央路由器, 啥是边界路由器</h3>

<p>这就要扯到很远, 首先最开始网络 ip 是分类的, 类别由网络号区分, 然后不同分类中由主机区分, 最后发现这样进行分太难管理了, 所以出现边界路由器, 目的就是分担网络, 我中央路由器只对应网络号的流量进行转发给相应的边界路由器, 然后对应要到那个主机再由边界路由器进行定位</p>

<h3 id="问题3--啥是网络号和主机号">问题 3: 啥是网络号和主机号</h3>

<p>自行百度</p>

<h3 id="问题4---我的ip是192-168-0-1-是私有ip咋么也可以连接因特网">问题 4: 我的 ip 是 192.168.0.1, 是私有 ip 咋么也可以连接因特网</h3>

<p>这个问题, 首先要牵涉到很久以前, 以前没有人使用私有 ip 一说, 大家都用公有 ip 地址, 然后可以联通, 但是 ip 地址有限, 要是这样下去肯定有些人需要共享 ip 地址了, 但是共享 ip 地址怎么区分流量是来自那台电脑, 这时候就利用上了私有 ip 和使用 NAT 网络地址转换技术来解决</p>

<h3 id="问题5--啥事NAT">问题 5: 啥事 NAT</h3>

<p>NAT 意思就是网络地址转换, 就是将一组私有 ip 转化为公有 ip 进行对因特网的访问</p>

<h3 id="问题6---一个私有ip对应一个共有ip那么也没省ip地址啊">问题 6: 一个私有 ip 对应一个共有 ip 那么也没省 ip 地址啊</h3>

<p>是的, 这样确实没有减轻 ip 地址的使用, 所以出现了 NAT</p>

<h3 id="问题7--NAPT是啥">问题 7: NAPT 是啥</h3>

<p>NAPT 很简单, 就是在网络地址转换的时候我给每个私有 ip 的请求转换成公有 ip 加端口, 这样可以实现一个 ip 多用了, 这样的话路由器就需要读取 tcp/udp 中的端口信息了, 路由器需要建立一个网络地址转换的表, 表意思是 <code>私有ip:端口->公有ip:端口, 公有ip:端口->外网ip:端口. 后: 私有ip:端口->外网ip:端口 </code>, 具体连接流程如下</p>

<ol>

<li>客户端发起连接去连接外网 ip</li>

<li>NAT 路由将客户端 ip+NAT 路由随机生成的端口来代理客户端去访问外网</li>

<li>然后外网的数据就会被 NAT 路由器转发到内网 ip</li>

</ol>

<p>应对这种连接有一种问题就是</p>

<ol>

<li>只能内网连外网, 不能外网连接内网</li>

<li>有些应用层协议就会出问题, 比如 ftp, ftp 的数据传输和控制是分开连接的, 控制连接保持验证命令控制, 当你想要获取或上传某些数据的时候, 就需要建立一个数据连接, (主动模式)这时候为了全就用控制连接发送 ip 和端口让 ftp 服务器知道, 但是这里就有一个问题, 你发送的 ip 和端口是在用层协议的包中, 这样 ftp 记录的是内网的 ip 和地址, 到时候 ftp 发送数据连接握手就会发现连接上, 为了解决这个问题就出现了 ALG 网关, 来处理应用层协议的问题, 比如解析你发送给 ftp 的 ip 端口进行转化成 nat 的 ip 和端口, 对应上到时候数据连接的时候你连接 nat 转换的端口, 具体的连图如下</li>

</ol>

<p></p>

<h3 id="ftp数据连接的2种模式">ftp 数据连接的 2 种模式</h3>

<ol>

<li>主动模式, 由客户端发送 ip 和端口, 然后服务器主动建立连接</li>

<li>被动魔兽, 由服务端发送端口, 客户端主动发起连接, 被动模式为了防止防火墙拦截</li>

</ol>  
<h3 id="ALG协议">ALG 协议</h3>  
<p>ALG 的作用就是解析特定的应用层协议的内容进行解析转换，用来配合 nat 地址转换来用的，是因为内网所有的地址都会被转换，所以在应用层协议的包中发送的内网 ip 和端口需要转化为 nat 的 p 和端口，然后让你们连接的时候用转化的 ip 和端口去玩</p>  
<h3 id="引用">引用</h3>  
<ol>  
<li><a href="https://ld246.com/forward?goto=https%3A%2F%2Fbaike.baidu.com%2Fitem%2Fnat%2F320024%3Ffr%3Daladdin" target="\_blank" rel="nofollow ugc">https://baike.baidu.com/item/nat/320024?fr=aladdin</a></li>  
<li><a href="https://ld246.com/forward?goto=https%3A%2F%2Fblog.csdn.net%2Fxiaoshengqdlg%2Farticle%2Fdetails%2F22417211" target="\_blank" rel="nofollow ugc">https://blog.csdn.net/xiaoshengqdlg/article/details/22417211</a></li>  
<li><a href="https://ld246.com/forward?goto=https%3A%2F%2Fblog.csdn.net%2Fgui95175%2Farticle%2Fdetails%2F79593307" target="\_blank" rel="nofollow ugc">https://blog.csdn.net/gui951753/article/details/79593307</a></li>  
<li><a href="https://ld246.com/forward?goto=https%3A%2F%2Fbaike.baidu.com%2Fitem%2FALG%25E7%25BD%2591%25E5%2585%25B3%2F144199%3Ffr%3Daladdin" target="\_blank" rel="nofollow ugc">https://baike.baidu.com/item/ALG%E7%BD%91%E5%85%B3/144199?fr=aladdin</a></li>  
<li>图解 tcp/ip 第五版-5.6 NAT</li>  
<li>Wireshark 网络分析就这么简单-ftp<a href="https://b3logfile.com/file/2019/11/20120601169543image001747033976650-21d6dac1.jpg">201206011369543image001747033976650.jpg</a></li>  
</ol>