家庭网络中的「NAT」到底是什么?

游戏主机上的「NAT 类型」到底是如何判定的?

Matrix 首页推荐

Matrix 是少数派的写作社区，我们主张分享真实的产品体验，有实用价值的经验与思考。我们会不定期挑选 Matrix 最优质的文章，展示来自用户的最真实的体验和观点。

文章代表作者个人观点，少数派仅对标题和排版略作修改。

在使用网络的过程中，或许你会遇到这样的问题：

- Nintendo Switch 等游戏主机上的 NAT Type 是 D， 导致无法联机游戏 ；
- 使用 *eMule* 下载文件时，发现获取到了 Low ID， 导致下载速度变慢 ；
- 家中的 NAS 只能在内网使用，很难将 NAS 上的文件分享给其他人。出门在外时，想要下载 NAS 上的文件，也发现无法连接。

或许你还会对这些问题感到好奇：

- 每天有那么多人使用微信进行视频通话，腾讯的服务器是否能承受这么大的流量？是否需要支付高额的流量费用？
- 为什么在十多年前，我们就看到过 IP 地址资源不足的新闻 。而现在，互联网用户越来越多，IP 地址资源不足却似乎并没有给我们带来什么影响。

其实，这些问题都与 NAT 有关。少数派上已经有两篇文章，也提到了 NAT：

- 入网指南 04 | IP 地址大揭秘 - 少数派
- 局域网游戏串流：让我们都做一回「云」玩家 - 少数派

那么，NAT 到底是什么？NAT 会为我们的「网上冲浪」带来哪些不便之处？如何解决 NAT 为我们带来的不便？…… 本文将尝试详细地解答这些问题。

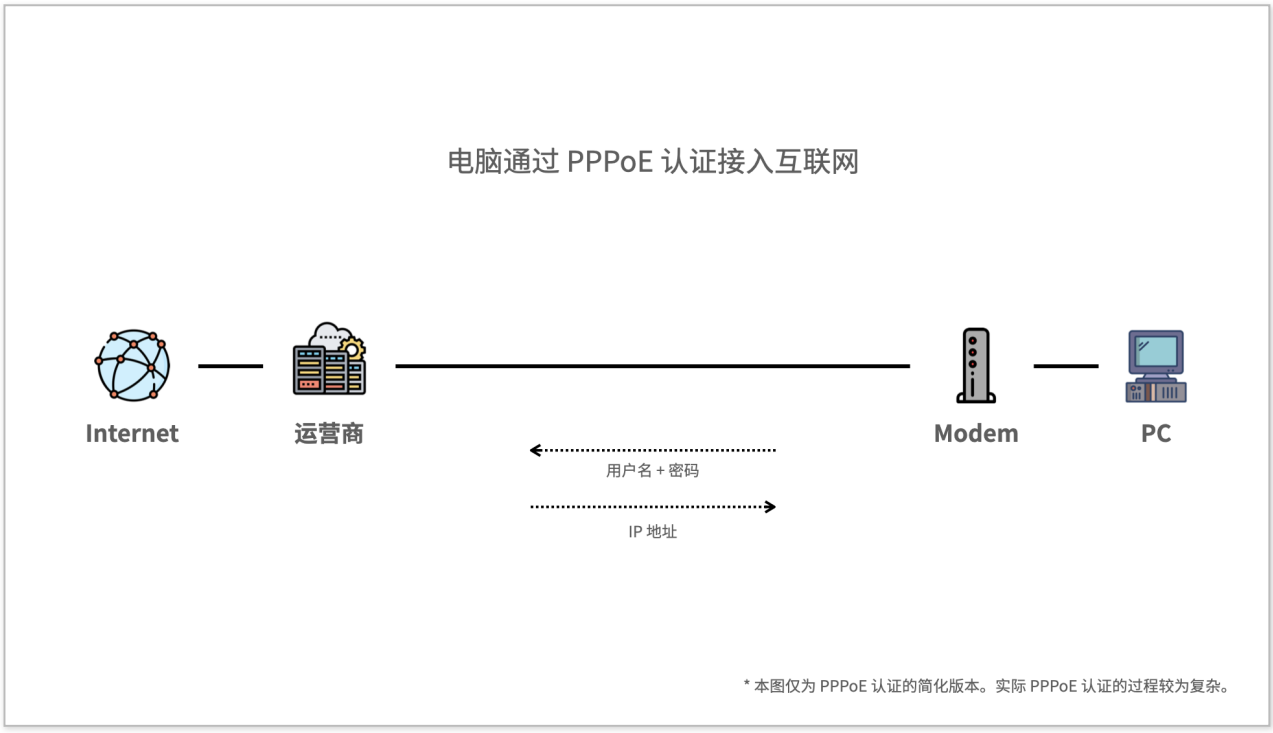
为什么需要 NAT

在 Internet 上，每台设备都有一个 IP 地址。IP 地址和我们日常生活中的家庭住址类似，一台设备想和另一台设备通信，必须知道另一台设备的 IP 地址，才能将数据发送到对方。

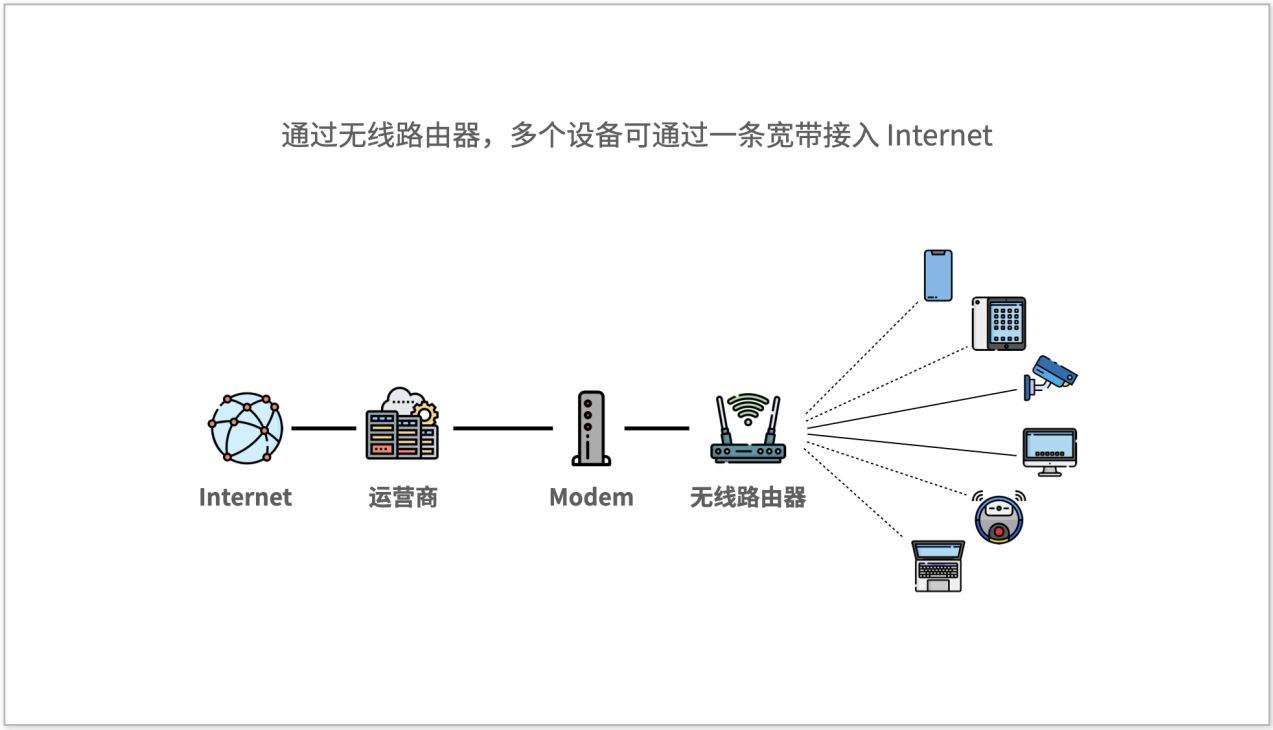
对于 IP 地址的基础知识，少数派已经有一些文章进行了介绍。读者可参考这些文章进行了解：

- 入网指南 04 | IP 地址大揭秘 - 少数派
- 小白也能看懂的网络基础 04 | IP 地址是如何工作的 - 少数派
- 小白也能看懂的网络基础 05 | IP 地址深度学习 - 少数派

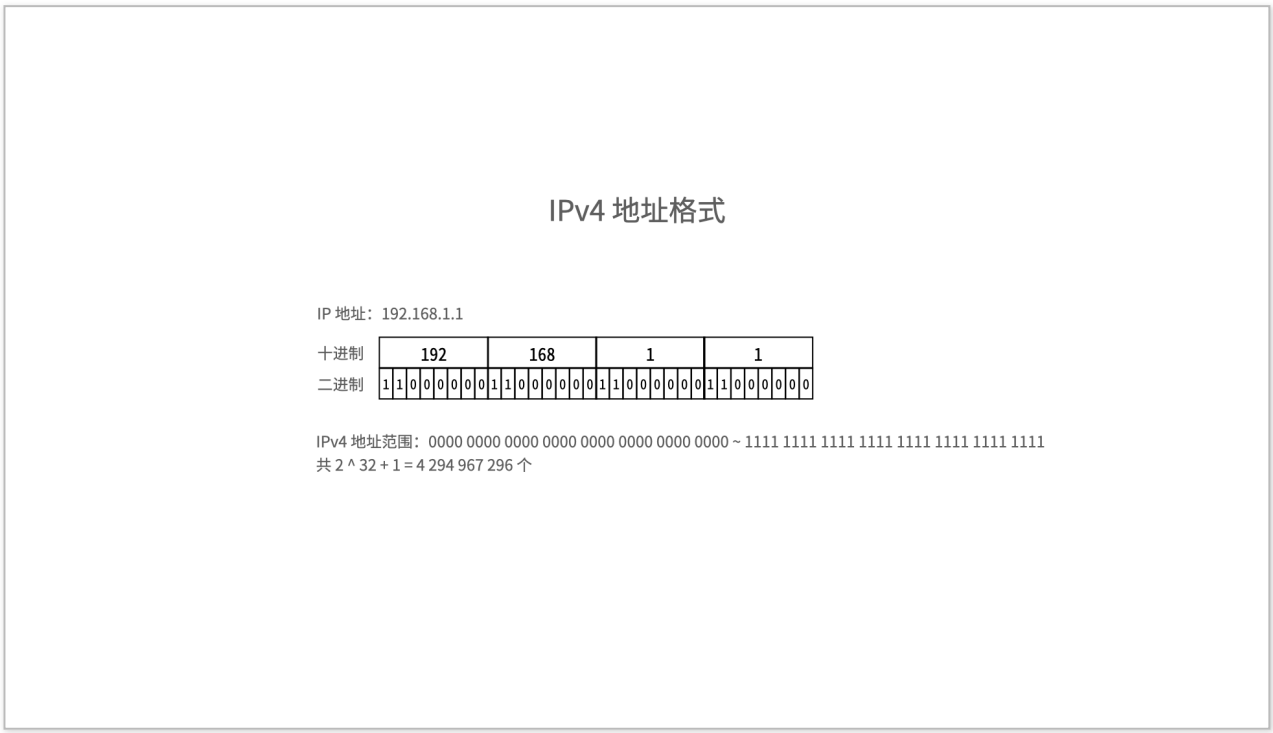
对于个人、家庭的网络设备，IP 地址一般由运营商分配。在过去，一户家庭一般只有一台电脑，这台电脑通过调制解调器（moden）直接接入 Internet，获取运营商分配的 IP 地址。



而现在，很多家庭会同时拥有手机、电脑、智能家居等多种联网设备，一个 IP 地址完全不够这么多设备使用。



对于常用的 IPv4 地址，其格式是这样的：



可以看出，IPv4 地址一共有 4 294 967 296 个。由于全球互联网用户越来越多，IPv4 地址 已经严重不足 。

综上，如下两个原因，让我们面临了 IP 地址不够用的问题：

1. 家庭中有多个设备需要联网，但运营商只会分配一个 IP 地址
2. 全球联网设备越来越多，但 IPv4 地址资源有限

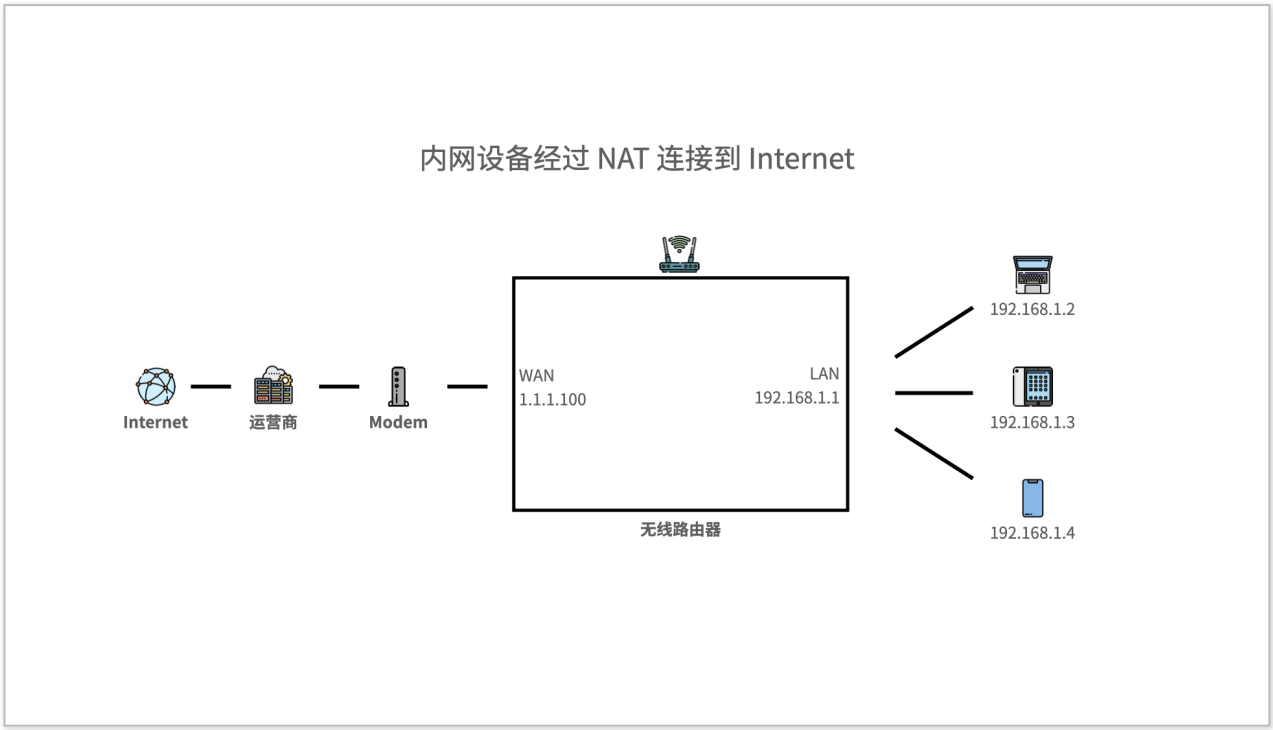
所以，我们需要一种技术，让多个设备「共用」同一个 IP 地址，来缓解 IPv4 地址不够用的问题。

什么是 NAT

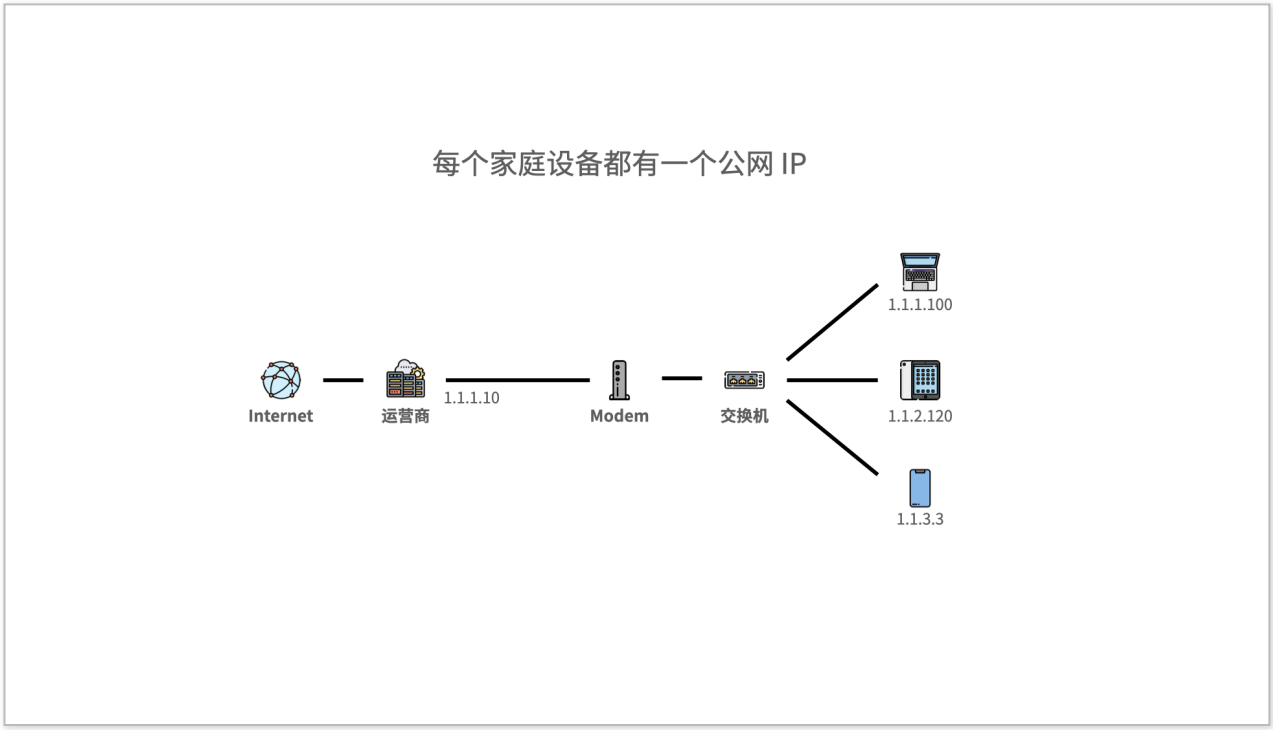
NAT 的全称是「网络地址转换」（Network Address Translation），指的是路由器等网络设备，在传输数据的过程中，改变数据中的 IP 地址的一种技术。

NAT 可用于内网 IP 地址（以下简称为「内网 IP」）和公网 IP 地址（以下简称为「公网 IP」）之间的转换。例如家庭中的多个联网设备，都拥有各自的内网 IP，无线路由器运行 NAT 功能；家中的设备向外发送数据时，数据中的内网 IP，在无线路由器上会被转换为公网 IP；外部数据发送到家庭设备时，数据中的公网 IP，会被转换为内网 IP。

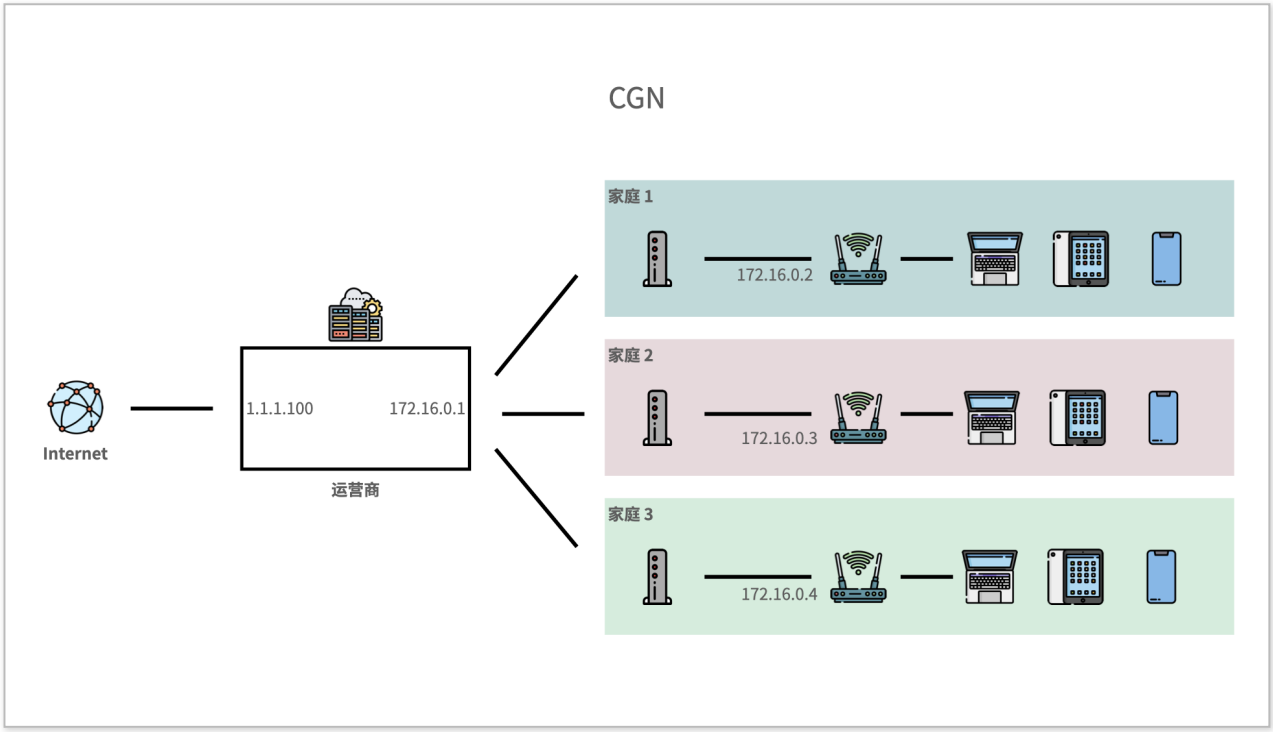
通过这种方式，家庭设备能够「共享」同一个 IP 地址。即使运营商只为用户分配了一个 IP 地址，用户家中的多台设备，也能同时访问 Internet。



做为对比，如果没有 NAT，家中每一个设备，都需要获取一个独立的公网 IP 地址，对于本来就少的 IPv4 地址资源，就显得有点「奢侈」了。



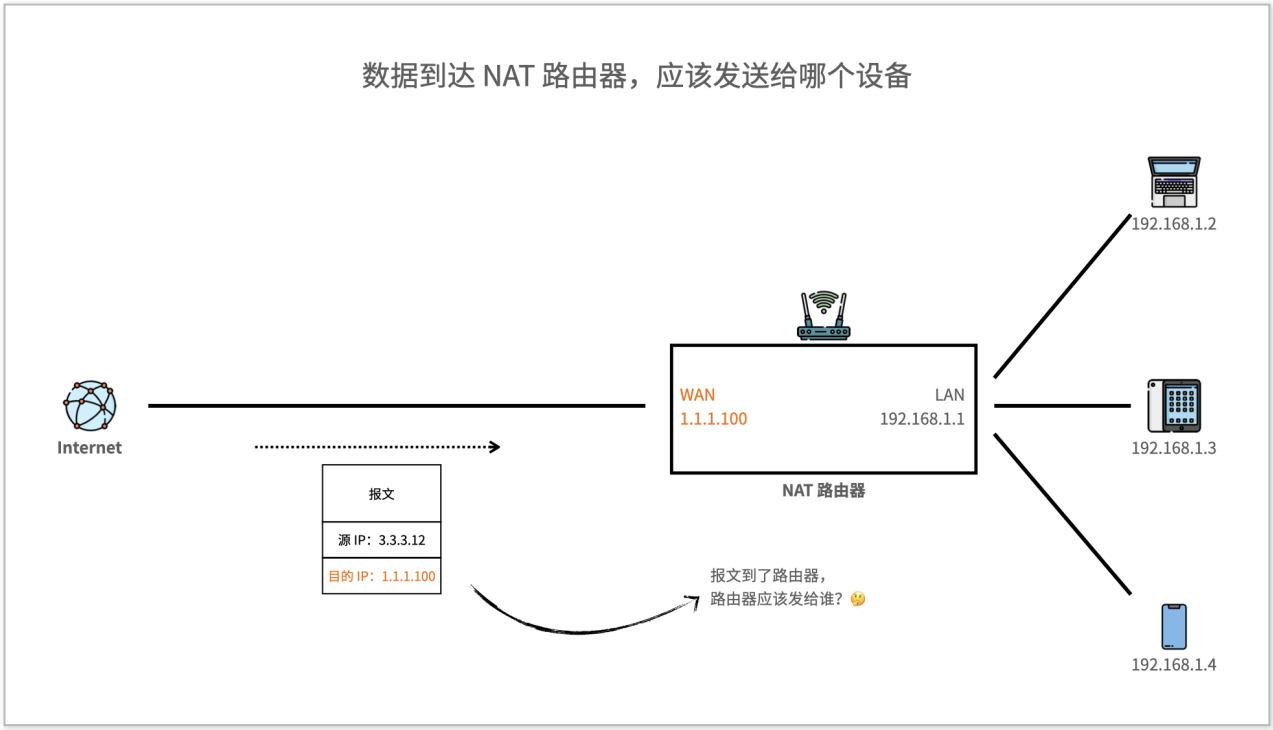
另外，在 IPv4 地址资源越来越紧张的今天，很多电信运营商，已经不再为用户分配公网 IP；而是直接在运营商自己的路由器上运营 NAT，为用户分配内网 IP。这样，只需要少量的 IP 地址，就可以支撑大量用户的上网需求。这样的 NAT 又叫做 CGN（Carrier-grade NAT，电信级 NAT） 。



NAT 是如何工作的

NAT 改变了报文中的 IP 地址。但是，为什么我们平时上网时，并没感觉到 NAT 的存在？

在上文中的例子里，多台内网设备共用同公网 IP。外部数据到达路由器后，路由器应该将数据发送给哪个内网设备?



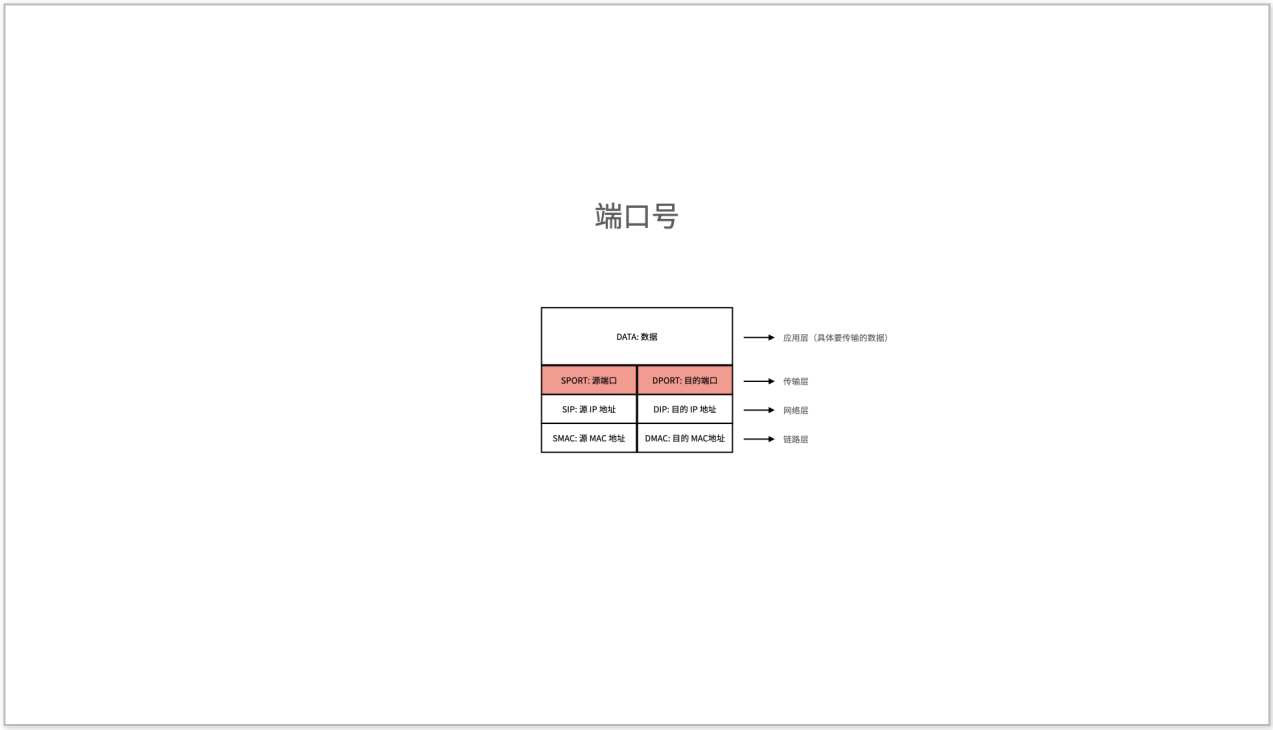
在介绍 NAT 的工作原理之前，让我们先了解一下另外一个概念：端口号。

- 小白也能看懂的网络基础 07 | TCP 和 UDP 是如何工作的？ - 少数派

简单说，在网络中传输的数据，会被拆分一个个较小的片段。每个片段被称为一个报文。在报文中，除了 IP 地址，一般还包含了端口号。

IP 地址每台电脑唯一，用来找到网络中的电脑。端口号每个应用程序唯一，报文到达电脑后，可根据端口号匹配到应用程序。

在报文中，包含了两个 IP 地址和两个端口号，分别是来源 IP、目的 IP、来源端口、目的端口：



我们使用的大部分软件，例如网页浏览器，都是客户端软件。客户端需要主动向服务器发起连接。

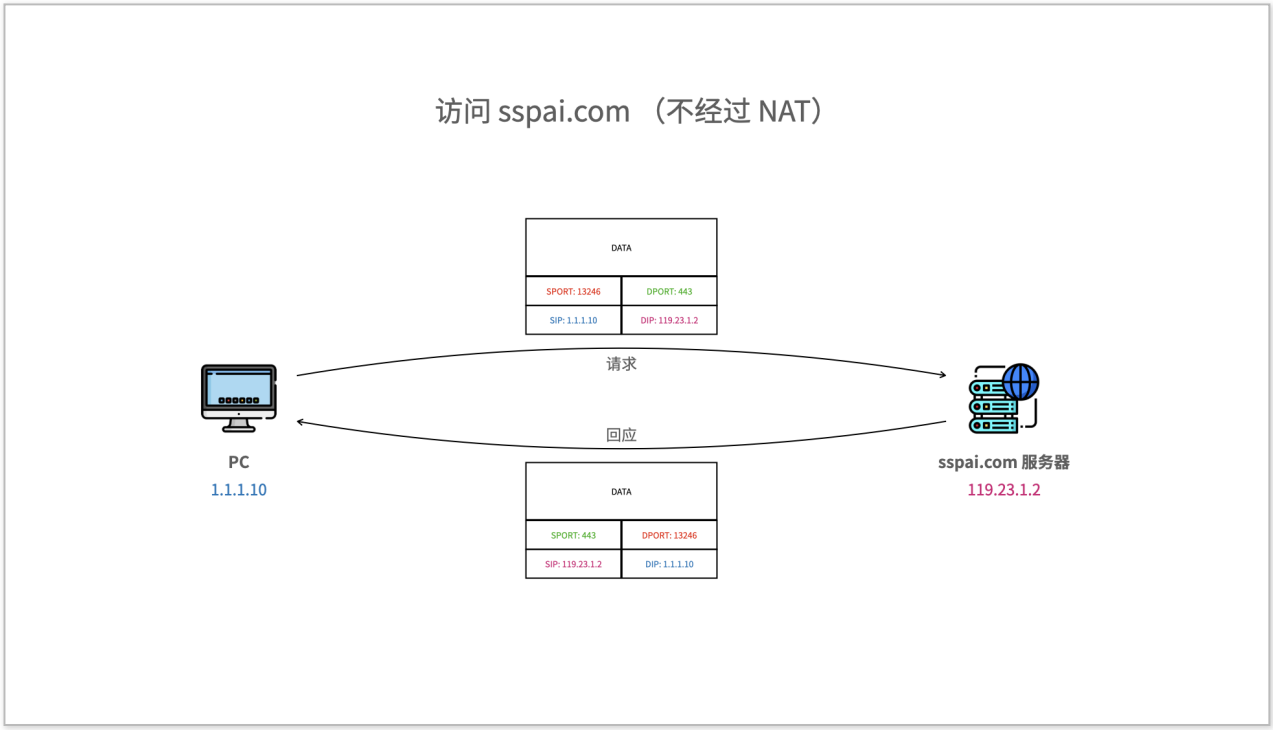
例如，当我们访问少数派网站 `sspai.com` 后，浏览器能够解析到网站的 IP 地址（下文以 `119.23.1.2` 为例，不代表网站真实 IP），主动向该地址发起报文，建立连接。由于浏览网页使用的 HTTPS 协议，端口号为 443，所以报文中的目的端口号填充为 443，来源端口号是一个随机分配的值。

当少数派网站收到请求后，则会向用户的浏览器发送网页数据

实际上，TCP 连接需要经过三次握手。此处简化了描述。

。发送的数据中，来源端口号为 443，目的端口号则为用户请求报文中的源端口号。

从下图中也可以看出，回应报文的目的 IP，就是请求报文的源 IP，也就是用户电脑的 IP 地址；回应报文的目的端口号，就是请求报文中的源端口号，也就是用户浏览器的端口号。这样，少数派服务器的回应报文，就能根据 IP 地址发送回用户电脑，并根据端口号最终到达浏览器：



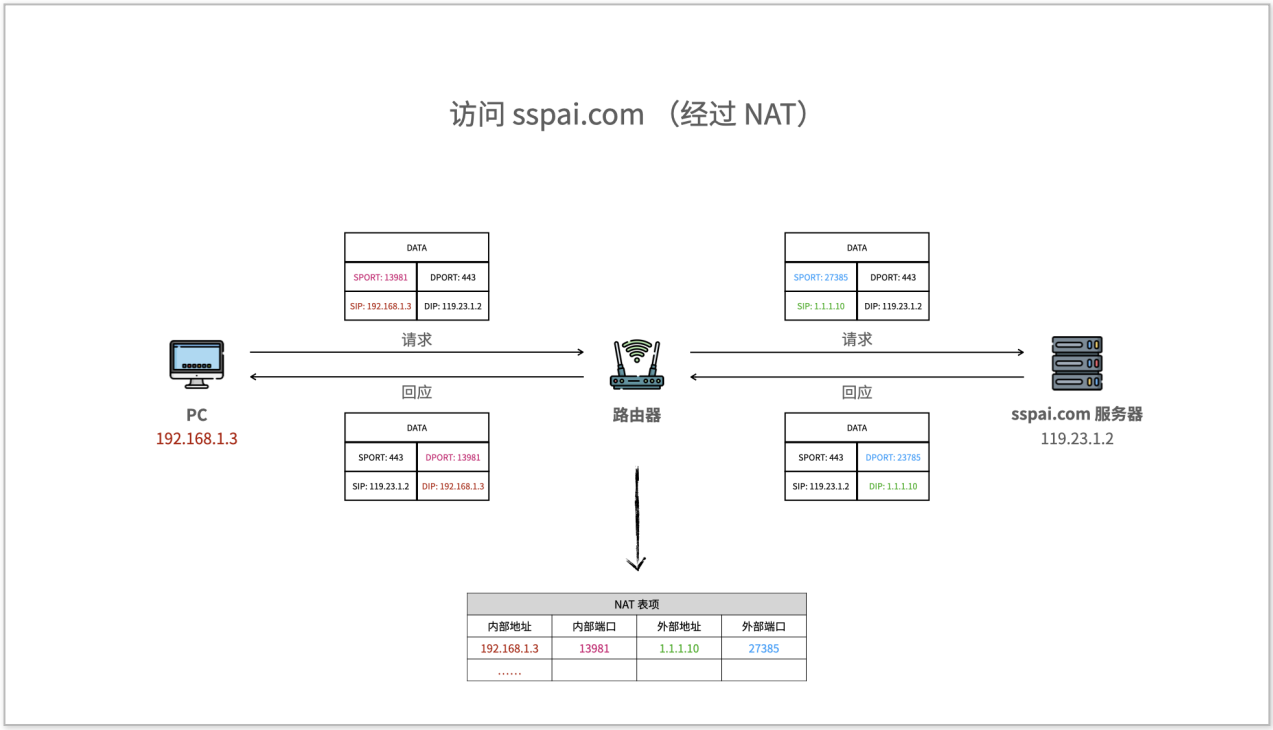
那么，如果用户的电脑是内网设备，经过了 NAT，使用浏览器访问少数派网站时，又会是什么样的过程呢？

首先，浏览器同样会发送请求报文，报文的源 IP 为电脑的内网 IP （此处以 192.168.1.126 为例）。

当报文到达路由器后，路由器将报文源 IP 修改为公网 IP（1.1.1.10），并分配一个新的源端口号。**在这个过程中，路由器会记录下源 IP 和源端口号，在转换前后的对应关系，形成 NAT 表项。**

路由器将源 IP 和源端口号转换后的报文发送到服务器，服务器回应的报文，目的 IP 和目的端口端口号，就是请求报文中的源 IP 和源端口号。这样，报文就能根据目的 IP，到达用户的路由器上。

路由器收到来自服务器的报文，根据 NAT 表项，将目的 IP 和目的端口号，从外部 IP、外部端口，转换为内部 IP、内部端口。这样，报文就能顺利到达用户电脑的浏览器上。



从上面的例子中可以看出，NAT 通过记录端口号

我们常用的 NAT 需要记录端口号信息，所以也叫 NAPT。也有一些不需要端口号的 NAT，不在本文的介绍范围内。

、IP 地址的对应关系，将出方向报文的源 IP、源端口号从内部地址转换为外部地址，将入方向报文的目的 IP、目的端口号从外部地址转换为内部地址，让内网设备也能正常访问 Internet。但如下两种情况，是 NAT 难以做到的：

1. 内网设备做为服务器，外部设备主动向内网设备发起连接

2. 使用 TCP、UDP 之外的、没有端口号的协议进行通信

由于我们日常上网，使用的基本上都是 TCP 和 UDP 协议。而且自己的设备一般是做为客户端，主动连接第三方服务器的。所以，在日常上网的情况下，我们一般不会感受到 NAT 的存在。

NAT 为我们带来了哪些不便之处

NAT 缓解了 IP 地址资源不足的问题，同时能使家庭中的多个设备共享同一条宽带，同时上网。另外，启用 NAT 后，外部设备无法主动发起对内网设备的连接，相当于起到了防火墙的作用，保护了内网设备，一定程度上提高了安全性。

NAT 通过「巧妙」的方式，在内部地址和外部地址之间进行转换。大部分情况下，我们感受不到 NAT 的存在。但仍有部分应用，需要内网设备做为服务器，被外部连接，例如：

- 远程访问家中的 NAS、监控摄像头
- eMule、BitTorrent 等 P2P 文件分享应用，使自己的设备可供外部连接，从而能够连接到更多分享者，获取更快的下载、上传速度
- 部分语音通话、视频会议应用，通信双方直接连接，获取更好的通话质量
- 部分联机游戏，不会经过第三方服务器，需要玩家之间直接建立连接

对于这些应用，如果设备位于 NAT 之内，没有公网 IP，就难以实现了。

如何在 NAT 场景下，使内网设备可被连接

那么，在 NAT 环境下，应该如何让内网设备做为服务器，使内网设备被外部连接？下文将介绍几种常见的方式。

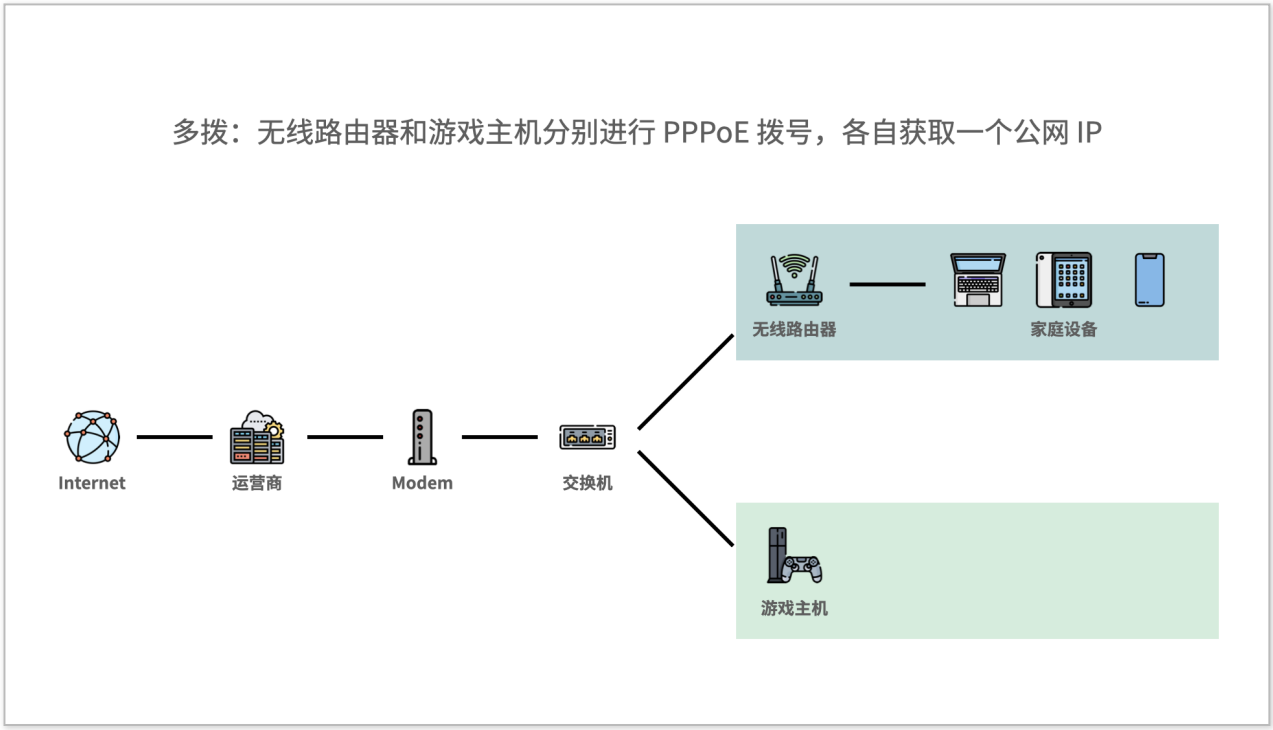
多拨

部分运营商，支持在多个设备上，通过 PPPoE 登录同一个宽带账号。每个设备都能获取到一个独立的公网 IP

具体可以阅读宽带用户协议，或实际尝试，来确定运营商是否支持。

。

如果想让游戏主机等设备获取独立的公网 IP，供外部连接，可以在光猫之后连接交换机。游戏主机连接交换机，直接进行 PPPoE 拨号。无线路由器也连接交换机，家中的其他网络设备经过无线路由器访问 Internet。



但是多拨的局限也很大：

- 仅部分运营商支持多拨
- 一些运营商已不再为用户分配公网 IP，即使通过多拨，也获取不到公网 IP
- 越来越多的设备不再支持 PPPoE。例如 Xbox 360 支持 PPPoE， 但 Xbox One 之后的版本已不再支持
- 设备直接获取公网 IP，暴露在公网上，安全性较差。可能需要单独设置防火墙
- 需要额外购买交换机，连接在光猫和路由器之间。会改变家庭网络拓扑，操作比较复杂

所以，这种方式不太常用。

端口转发、DMZ

上文中介绍的 NAT，路由器会根据内网设备发出的报文，自动形成 NAT 表项。实际上，用户还可以在路由器上手动配置端口映射关系，让内网设备可被外部访问。

其中，DMZ 功能，可以指定一台内网设备为 DMZ 主机。到达路由器上的报文，如果没有匹配 NAT 表项，就会转发到 DMZ 主机。从而使 DMZ 主机可被外部访问。

- DMZ 的介绍与设定 | 官方支持 | ASUS 中国

DMZ 功能能让一台内网设备上的所有端口，都能被公网访问。但这样做也影响了内网设备的安全性，如果没有特殊需要，不建议打开这一功能。

而 **端口转发** 功能，可以手动设置端口映射关系，让指定内网设备的指定端口，能够被公网访问：

- 如何在 ASUS 路由器设置端口转发规则？ | 官方支持 | ASUS 中国

这种方式能够精确控制哪些设备的哪些端口可被公网访问。但需要用户具有一定的网络知识，知道需要被公网访问的应用的端口号，才能正确设置。

另外批评一下任天堂，在官网的帮助文档中，直接让用户打开了所有 UDP 端口的端口转发。这样做降低了安全性，而且可能与用户的其他端口转发规则冲突。不过从另一个方面，也说明了根据实际应用，手动设置端口转发规则，对部分用户来说，确实是一件门槛较高的事情：

- jo𝔥nnʏ Xмäϕ on Twitter: "Uhhh why is @NintendoAmerica telling *every* Switch owner to *effectively expose their consoles directly to the internet? Real excited for how fat-fingering the IP and forgetting / not being able to set UDP only can also expose anything else on folks' home networks. https://t.co/R5tSkbatfx" / Twitter

UPnP IGD、NAT-PMP

上文中的**端口转发**功能，需要手动配置端口转发规则，操作起来比较麻烦。而 UPnP IGD 和 NAT-PMP 协议，则能实现自动配置端口转发规则。

UPnP IGD（互联网网关设备协议）和 NAT-PMP（NAT 端口映射协议）分别由微软和 Apple 提出，功能类似，都可以让应用程序告诉路由器需要打开的端口，让路由器自动设置端口转发规则。

- 互联网网关设备协议 - 维基百科，自由的百科全书
- NAT 端口映射协议 - 维基百科，自由的百科全书

UPnP IGD 和 NAT-PMP 的工作，需要应用程序和路由器的配合。首先需要在路由器上打开 UPnP 或 NAT-PMP 功能：

- 如何设置无线路由器的 UPNP 功能？ - TP-LINK 服务支持
- Universal Plug'n'Play and NAT-PMP on OpenWrt

还需要使用支持的应用程序。目前 *e*Mule、BitTorrent 等常见的 P2P 文件共享工具，以及 Synology DiskStation 等 NAS 设备，以及 Xbox 等游戏主机，都已经支持相关协议：

- eMule UPnP 设置
- PortForwardingGuide – Transmission
- 路由器配置 | DSM - Synology 知识中心

光猫改为桥接模式，使用路由器拨号

安装宽带时，运营商附送的光猫，一般会默认打开路由功能。这时光猫同时能作为路由器使用。

但光猫的功能和性能有限，一部分型号的光猫不支持 UPnP IGD 等协议，或者不能手动配置端口转发规则。

所以，可以考虑将光猫修改为桥接模式，通过自己的无线路由器拨号，充分利用路由器上端口转发、UPnP IGD、DMZ 等功能。

正常情况下，光猫改桥接，最简单的方式是拨打运营商的电脑，让运营商远程下发配置。也可以登录光猫的管理页面，自行进行修改，具体需要上网搜索运营商名和光猫型号来查找教程。

向运营商申请获取公网 IP

由于 IPv4 地址资源不足，不少运营商已经不再分配公网 IP。

在部分地区，可以尝试拨打运营商客服电话，申请分配公网 IP。

另外一部分地区的运营商，在进行 PPPoE 拨号时，用户名中加入 `pub`，即可获取公网 IP：

- 广东电信真方便，用户名加个 pub. 就能拿到公网 IP 了。 - V2EX

部分地区需要付费购买，例如北京移动宽带，之前提供有公网 IP 叠加包。

但是，由于 IPv4 地址资源本身已经不足，不一定能够成功申请到公网 IP。另外，拥有公网 IP，家中的路由器能直接被公网访问，如果没有配置好，可能会带来更多安全问题。所以，需要根据自己的实际需要，来决定是否申请公网 IP。

PCP

对于运行 NAT 的家庭路由器，通过 UPnP IGD 或 NAT-PMP 协议，可以方便地将端口映射到公网。

但是，由于 IPv4 地址的不足，电信运营商也开始使用 NAT，不再为用户分配公网 IP。那么是否有一种类似 UPnP IGD 或者 NAT-PMP 的协议，运行在运营商的路由器上，能直接在运营商路由器上创建端口转发规则？

PCP 就是这样一种协议：

- Port Control Protocol (PCP) - RFC6887

该协议由 NAT-PMP 发展而来，运行在运营商的路由器上。用户的应用程序可通过 PCP 协议，申请在运营商路由器上打开端口。

PCP 需要运营商的配合，选用支持的网络设备，并打开 PCP 功能，才能正常工作。根据 V2EX 网友的测试，国内已有运营商支持该协议，能通过 PCP 使 *eMule* 获得 High ID：

- 介绍一个可能有助于 CGN NAT 端口映射的工具 - V2EX

服务器中转

上文中介绍了一系列使内网设备可被外部访问的方式。但这些方式或者需要用户手动配置，或者路由器的支持，或者需要运营商的支持…… 如果上述方式都不可用，就要通过第三方服务器中转的方式，让内网设备供外部访问。

这种方式虽然需要第三方服务器的参与，浪费资源，但成功率最高，所以应用范围也很普遍。例如常见的游戏加速器，就可以通过第三方服务器中转的方式，为游戏主机提供更高的 NAT 类型：

- 网易 UU 加速盒

也有不少开源的反向代理工具，可以搭建在自己的服务器上，使内网服务可在公网访问：

- fatedier/frp
- ehang-io/nps

服务器中转需要额外的服务器，且需要消耗服务器上的流量。所以这种方式往往需要用户额外付费，例如购买游戏加速器会员，或者自行购买虚拟服务器，并在服务器上搭建反向代理应用。

而对于微信语音、视频通话等应用，默认也会使用其他 NAT 穿透技术，来节省微信服务器的流量费用，降低成本。当其他 NAT 穿透方式不可用时，则采用服务器中转的方式，保证能够正常通话。

NAT 打洞

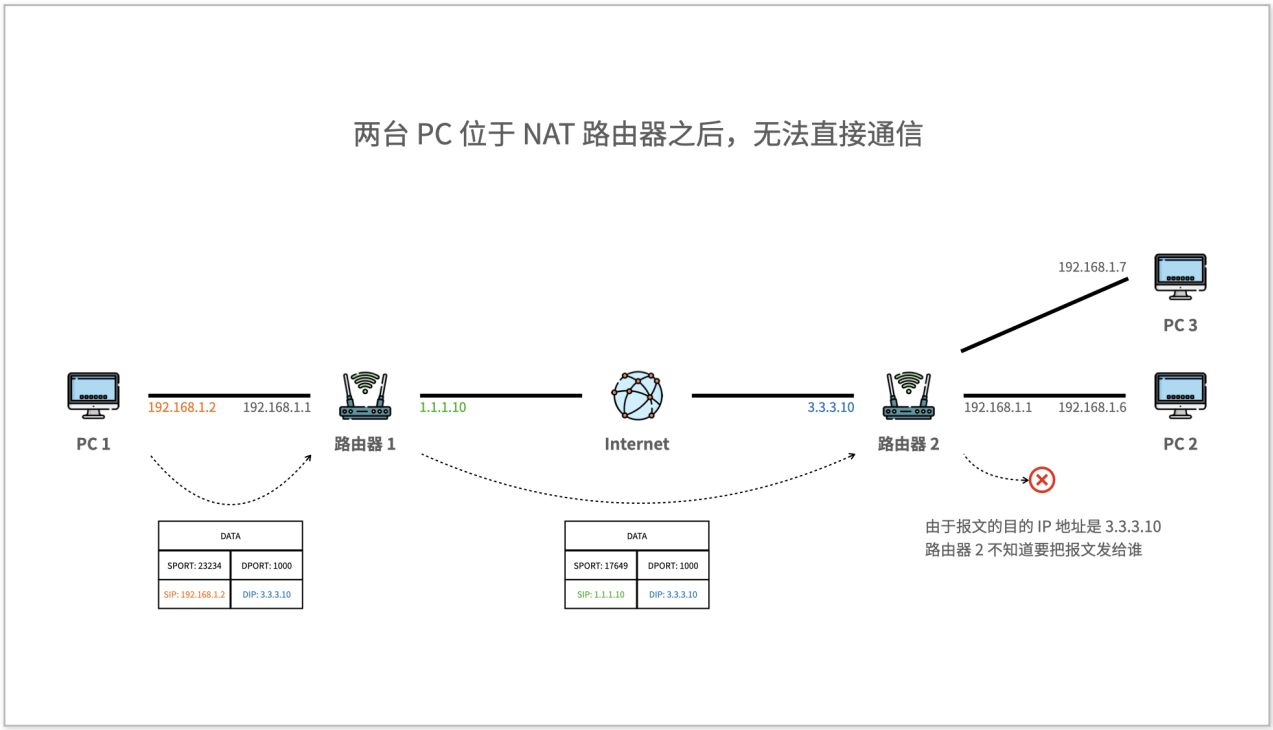
NAT 打洞的工作过程

如果两台设备都位于 NAT 路由器之后，没有公网 IP。在没有第三方服务器的中转下，是不是就没有办法直接进行通信了？

答案并不是这样的。NAT 打洞，就可以使两台内网设备能够直接通信，不需要第三方服务器的中转、不需要对路由器进行特殊设置、也不需要运营商的配置。**微信语音、腾讯会议、Skype 通信等消耗流量较大的应用，都会利用 NAT 打洞实现内网设备间的直接通信。**

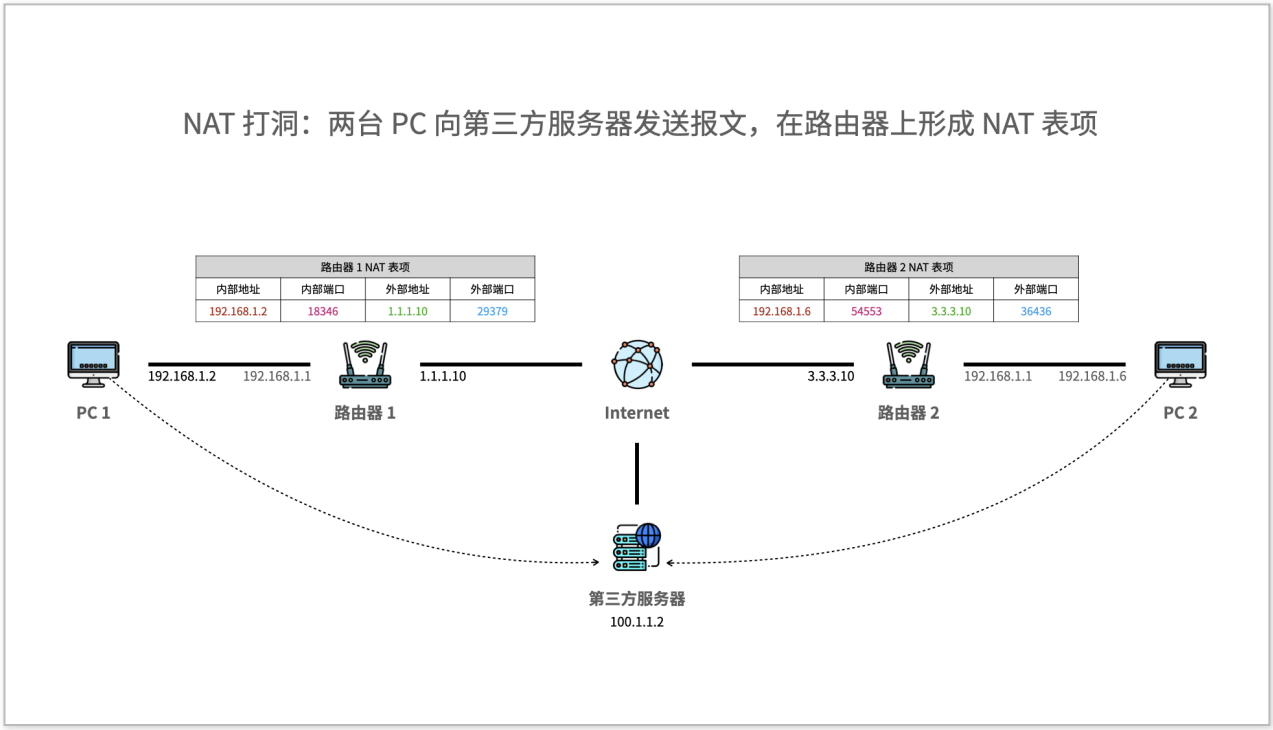
这项技术听起来很神奇，但是原理并不复杂：

我们以 PC 1、PC 2 两台主机的通信为例。两台主机均位于 NAT 路由器之后，各自的 IP 地址都是内网地址，无法互相通信：

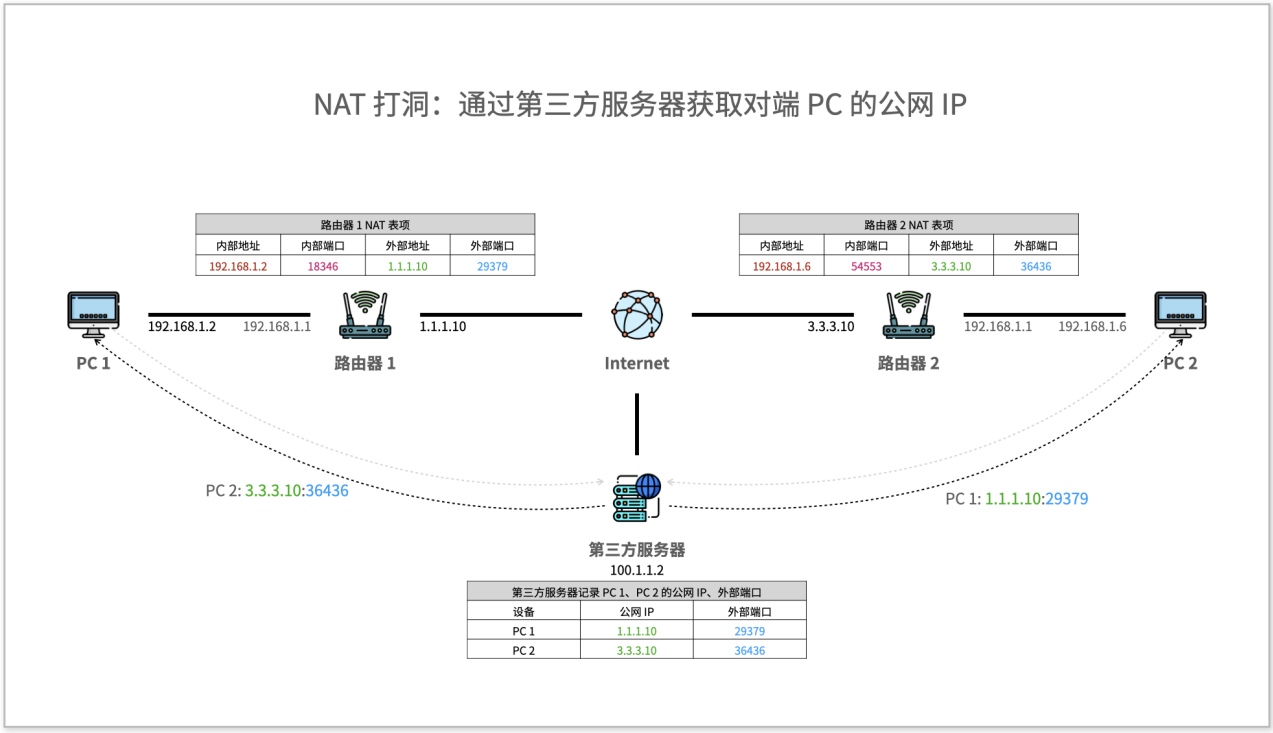


在两台主机能够直接通信之前，需要一台第三方服务器：

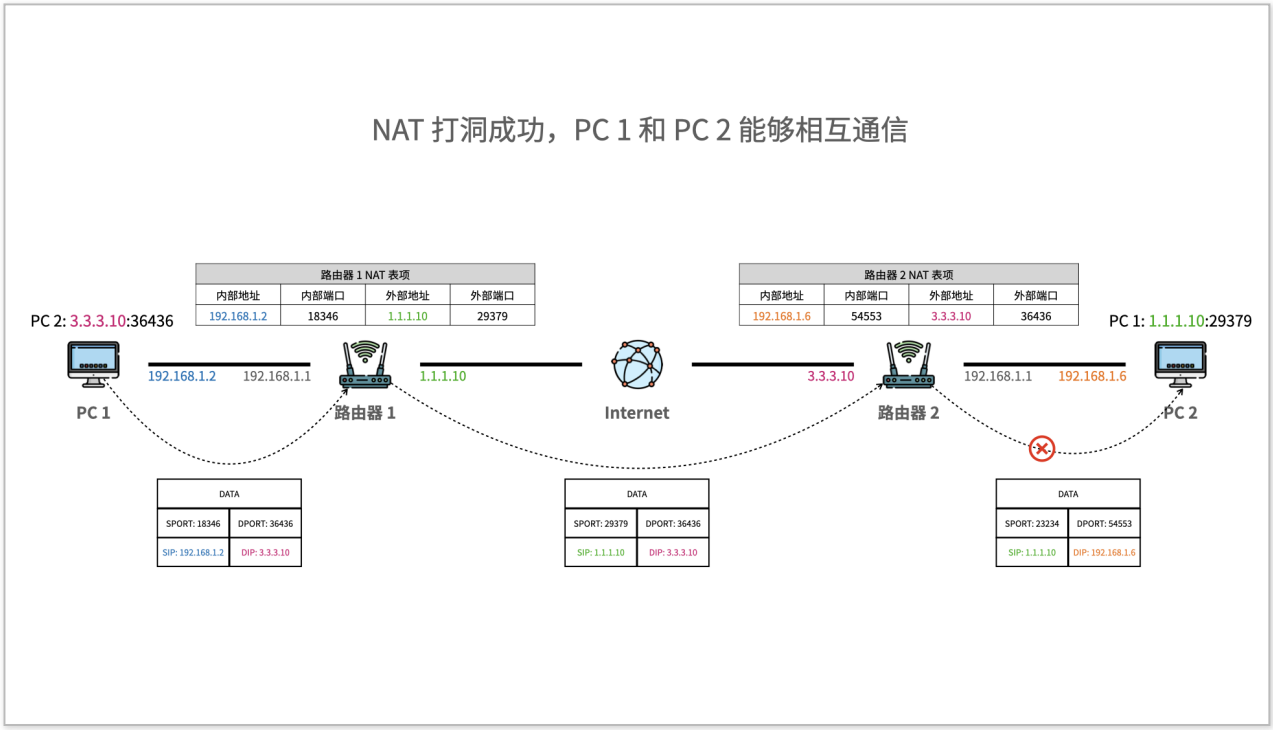
PC 1、PC 2 首先需要给服务器发送一个报文。经过 NAT 路由器后，报文的源 IP 和源端口号被转换，同时在路由器上形成 NAT 表项：



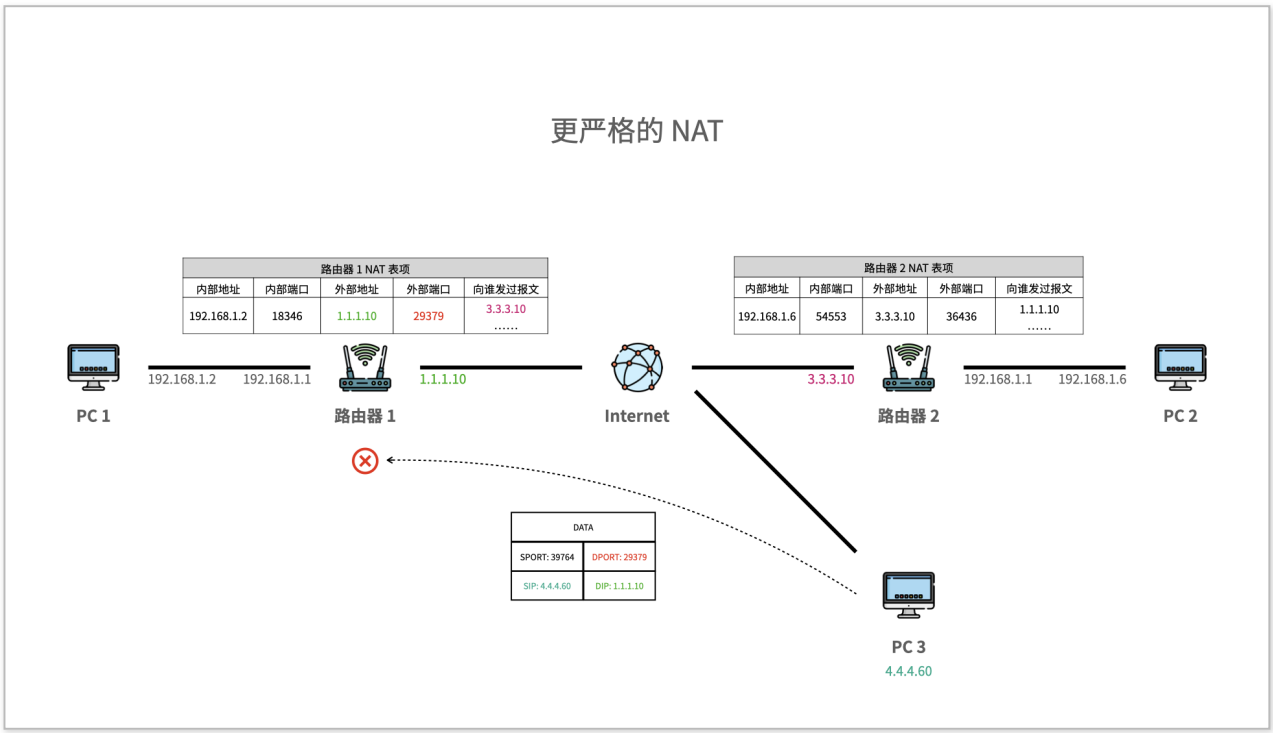
报文到达服务器后，服务器记录下 PC 1、PC 2 两侧报文的源 IP 和源端口号，也就是 PC 1、PC 2 两侧的公网 IP 和外部端口号。然后，服务器将两台设备的公网 IP、外部端口号发送给对方。这样，PC 1、PC 2 都能相互知道对方的公网 IP 和外部端口号：



其中一部分路由器的 NAT 检查比较宽松。一旦 NAT 表项建立，只要路由器上收到的报文，目的 IP 和目的端口号能够匹配到 NAT 表项，都会转发到表项对应的内网设备。对于这样的路由器，PC 1、PC 2 互相用对方的公网 IP 与外部端口号，就能直接通信了，不再需要第三方服务器：



另一部分路由器的 NAT 检查比较严格，只有内网设备向指定的目的 IP、目的端口号发送过数据，来自这个 IP 和端口号的报文，才能转发到内网设备：



对于这样的路由器，PC 1、PC 2 两台主机需要同时向对方的公网 IP 和外部端口号发送一个报文。这样，PC 1 侧的路由器认为 PC 1 向 PC 2 发送过数据；PC 2 侧的路由器认为 PC 2 向 PC 1 发送过数据，PC 1 和 PC 2 就能相互通信了。

经历了上述步骤，NAT 打洞成功，两台设备就可以不依赖第三方路由器，直接进行通信。当然，上述过程只是一个简化的描述，不完全描述。如果想要进一步详细了解 NAT 打洞的过程，建议参考文末的 RFC 文

档链接。

可以看出，NAT 打洞可以在无需路由器特殊配置、无需运营商配合的情况下，实现两个内网设备的相互通信。另外，对于多层 NAT 的网络环境（例如运营商和家庭路由器各进行一级 NAT），NAT 打洞也能正常处理。

NAT 类型与打洞成功率

在一些路由器的设置页面或文档中，我们会看到，NAT 能设置成不同的类型，例如 Full cone、Restricted cone、Port-Restricted cone、Symmetric：

- [无线路由器] 如何在华硕路由器中更改 NAT 类型？ | 官方支持 | ASUS 中国

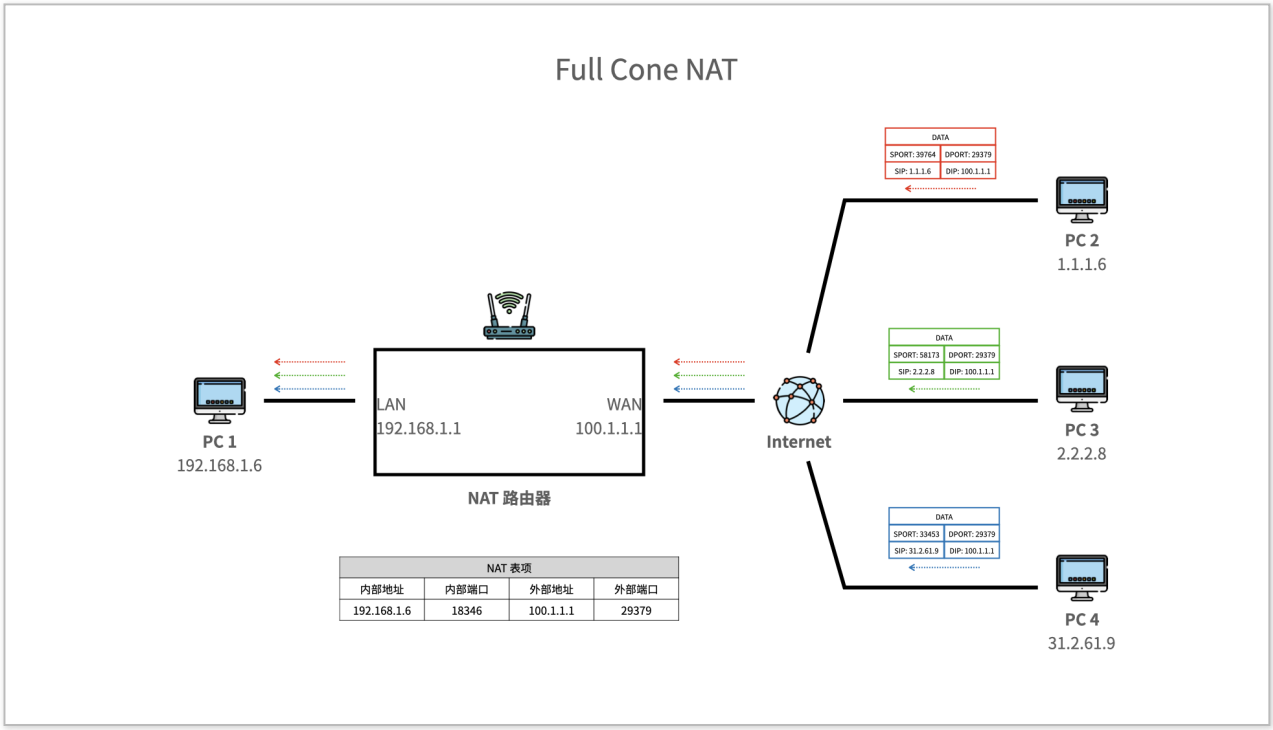
在 Xbox、PlayStation、Nintendo Switch 等游戏主机上，我们也能看到不同的 NAT 类型，例如 open、moderate、strict 等：

- 解决 NAT 错误和多人游戏问题 | Xbox Support
- 测试互联网连接 | PlayStation®4 用户指南
- 无法与其他在线玩家连接 - Nintendo Switch 常见问题

那么，这些 NAT 类型到底意味着什么？如何提升 NAT 类型，来获取更好的游戏体验？让我们先来了解一下基础的 NAT 类型：

完全圆锥形 NAT（Full cone NAT）：

对于完全圆锥形 NAT，内网 IP 和内网端口号，被映射为外部 IP 和外部端口号。当路由器收到来自外部的报文时，只要报文的目的 IP 和目的端口号，匹配到 NAT 表项的外部 IP 和外部端口号，都会转换为对应的内网 IP 和内网端口号，转发到内网设备。



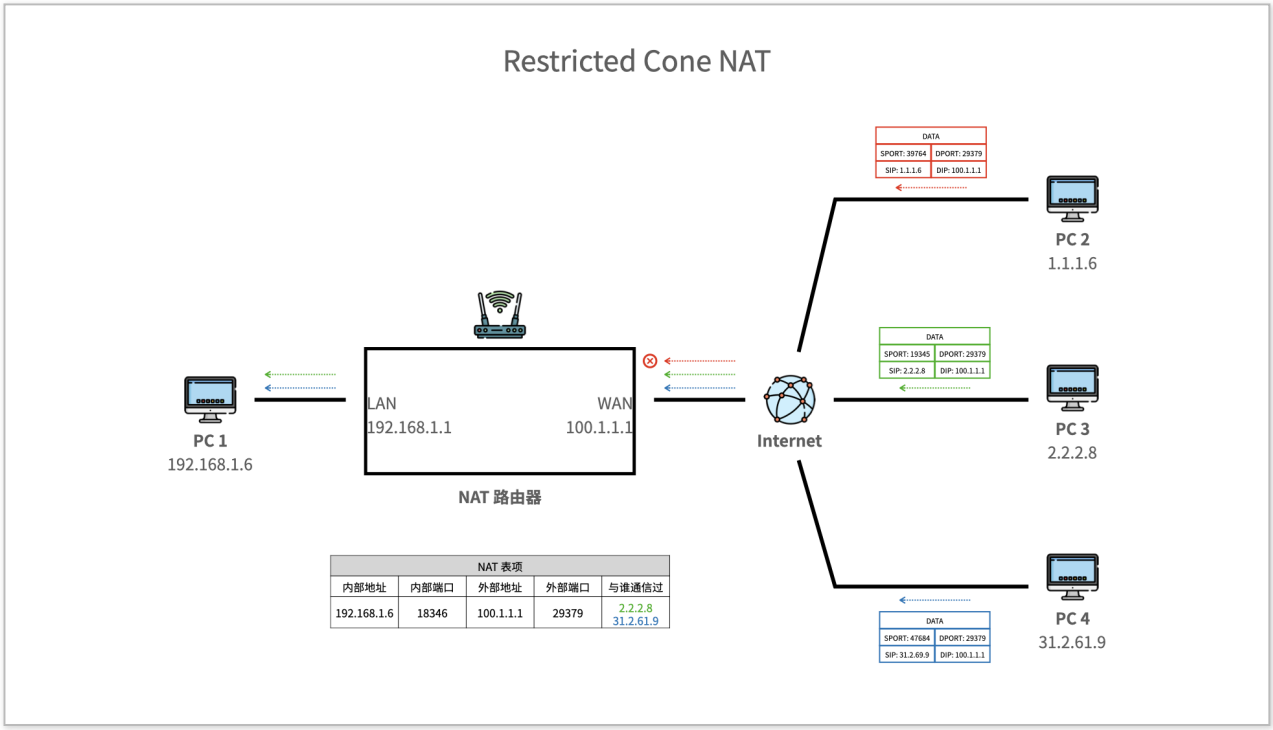
对于外部报文，路由器并不关心报文的源 IP 和源端口号（即报文来自谁），只要收到匹配 NAT 表项的报文，都能发送到内网设备。所以，完全圆锥形 NAT 是最宽松的 NAT，打洞最方便。

受限圆锥形 NAT（Restricted cone NAT）：

与完全圆锥形 NAT 相比，受限圆锥形 NAT，在内网设备向外发送报文时，路由器除了生成 NAT 表项，还会根据报文的目的 IP，记录下内网设备正在与哪些外部设备通信。

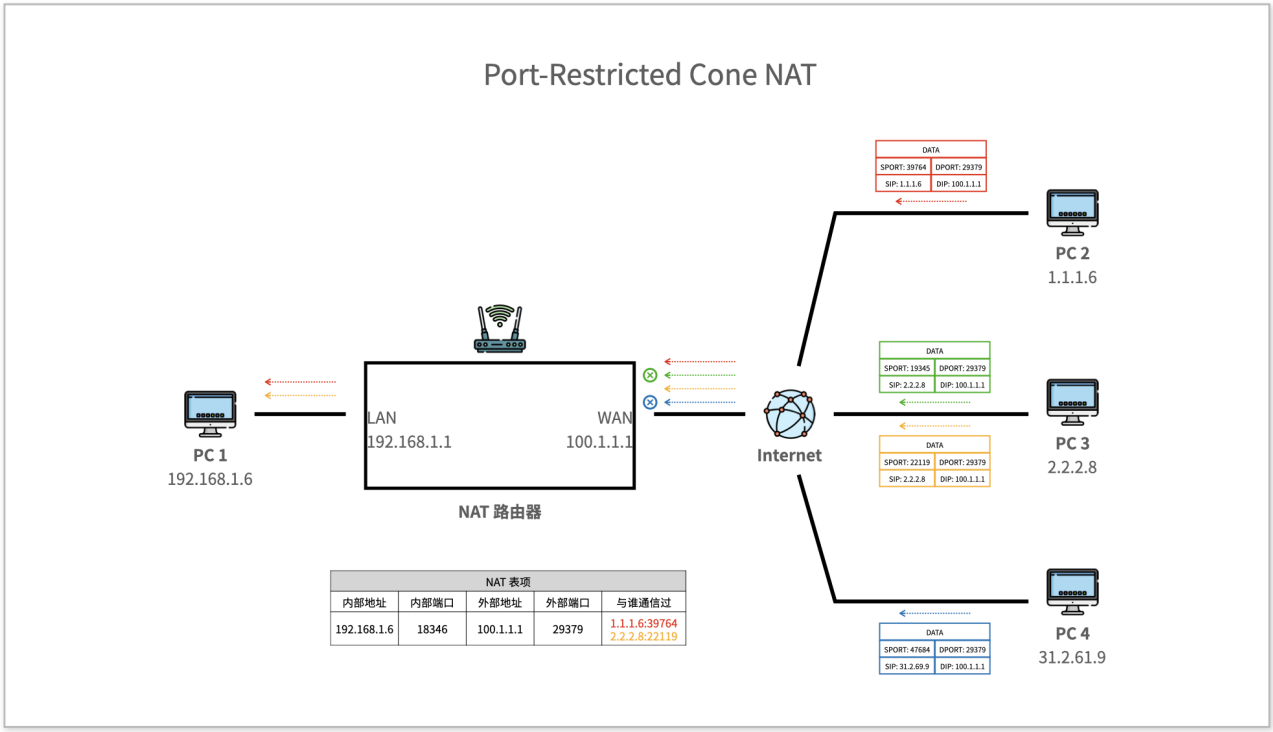
这样，只有内网设备先发送报文给外部设备，外部设备回应的报文，才会被转发到内网设备。而其他外部设备发送过来的报文，即使匹配 NAT 表项，也无法发送到内网设备。

这样的 NAT 安全性有一定的提高，但是也提高了打洞难度。两台内网设备需要互相给对方发送一个报文，才能打洞成功。



端口受限圆锥形 NAT（Port-Restricted cone NAT）：

端口受限圆锥形 NAT 和受限圆锥形 NAT 类似，但增加了检查的严格程度：**受限圆锥形 NAT**，只会外部设备的 IP 地址，来检查内网设备与哪些外部设备通信过。而**端口受限圆锥形 NAT**，会同时根据 IP 地址和端口号来进行检查。



对称 NAT（Symmetric NAT）：

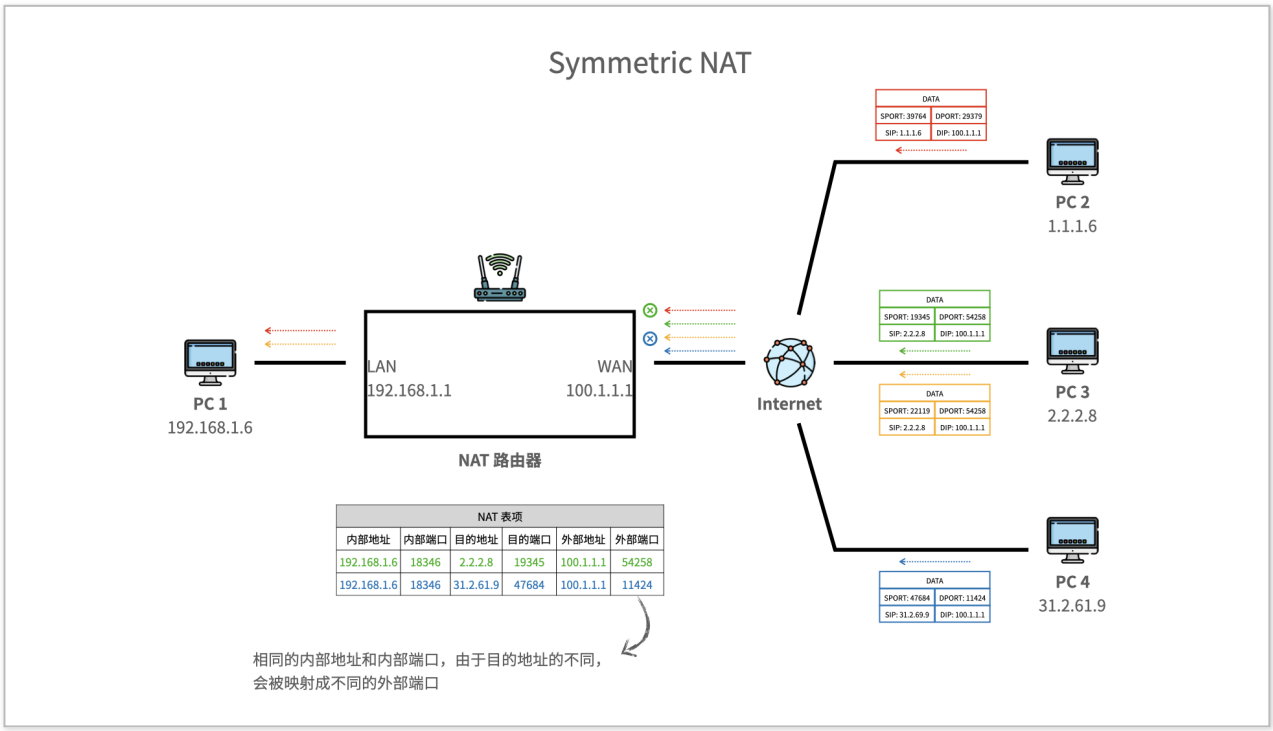
前面的三种**圆锥形 NAT**，会根据内网设备发出去的报文的源 IP、源端口号两个信息建立 NAT 表项，将内网 IP 和内网端口号映射到外部 IP 和外部端口号。内网设备发出去的报文，无论目的 IP 和目的端口号如

何变化，不管发给哪台外部设备，都会被映射为相同的外部 IP 和外部端口号。

而**对称 NAT**，会同时根据内网设备出方向报文的源 IP、源端口号、目的 IP、目的端口号四个信息来建立 NAT 表项。如果报文的目的 IP、目的端口号发生了变化，映射到的外部端口号也会发生改变。

对于对称 NAT，我们再来回顾一下前文中 NAT 打洞的过程。内网设备首先和第三方服务器通信，内网 IP 和内网端口号会被映射为一个外部 IP 和外部端口号。接下来，内网设备和另一台设备通信，相同的内网 IP 和内网端口号，又会被映射为另外一个外部端口号。这样，NAT 打洞就无法成功。

所以，在对称 NAT 下，很难进行 NAT 打洞。



而 Xbox、PlayStation、Nintendo Switch 上的 NAT 类型，基本上也跟上述四种 NAT 类型对应。例如 Xbox 上的三种 NAT 类型：

- open: 开放。当前主机可被所有外部主机连接。如果主机拥有公网 IP，或通过端口转发等方式，使主机可被外部访问，就能获取到这种类型。
- moderate: 当前主机无法被外部主机直接连接。但是可以通过 NAT 打洞等方式，实现主机间的通信。一般对应上文中的**三种圆锥形 NAT**。
- strict: 但是 NAT 比较严格，无法打洞成功。一般对应**对称 NAT**。

PlayStation 的 NAT 类型也与之类似，可参考如下链接：

- 使 PS4 获得最佳 NAT 类型 更新华硕路由解决方案 - PS4/PS5 综合讨论区 - A9VG 电玩部落论坛

所以，对于能设置 NAT 类型的路由器（例如华硕的部分型号），将 NAT 类型设置为 Full cone NAT，能够更容易地实现 NAT 打洞，使游戏主机更容易被外部连接。

Q&A

在这一章节，我会解答关于 NAT 的一些常见问题。如果还有其他疑问，欢迎在评论区提出，我会将相关问题与解答，更新到这一章节。

对于「对称 NAT」等更严格的 NAT，是否还可以打洞？

按照上文中的描述，「对称 NAT」是无法打洞成功的。但是，事实真的是这样吗？

实际上，运用一点点概率知识，对称 NAT 也是可以打动成功的。让我们先来了解一下「生日问题」：

> 生日问题是指，如果在一个房间要多少人，则两个人的生日相同的概率要大于 50%？答案是 23 人。 这就意味着在一个典型的标准小学班级（30 人）中，存在两人生日相同的可能性更高。对于 60 或者更多的人，这种概率要大于 99%。 — Wikipedia

假设内网设备 A 位于「圆锥形 NAT」之后，内网设备 B 位于「对称 NAT」之后。这时候 A 向 B 发送报文，B 并不知道 A 的报文经过 NAT 之后，公网端口号会被转换为什么，所以无法向 A 发送数据。

当然，在 B 不知道 A 的外部端口的情况下，也可以用不同的端口号，依次尝试向 A 发送报文，直到发送成功为止。但端口号一共有 65535 个，扫描所有端口并不现实……

那么，如果我们运用上「生日问题」，A 随机向 B 以不同的源端口号，发送 256 个报文，形成 256 个 NAT 表项；B 同时也随机以不同的目的端口号，向 A 发送 256 个报文进行探测，打洞成功率就能达到 64%。如果随机探测 1024 个端口，打洞成功率则能达到 98%。

如果想要更详细地了解，可以参考这篇文章中 *The benefits of birthdays* 部分：

- How NAT traversal works • Tailscale

果壳 DATAMUSE 团队制作了一个有趣的交互页面，来解释生日问题。如果感兴趣，不妨打开如下链接尝试一下：

- 生日悖论 - DATAMUSE

利用 NAT 打洞，能实现哪些有趣的应用

前文中已经提到，**语音通话、视频会议应用**，以及**在线游戏**，都用到了 NAT 打洞。那么，利用 NAT 打洞，还能实现哪些有趣的应用？

其实，最常见的应用，就是通过 NAT 打洞，将多个设备组建一个虚拟局域网。例如在家中有 NAS，且没有公网 IP 的情况下，通过这些利用 NAT 打洞的工具，仍然可以在离开家的时候，用手机直接访问 NAS 上的文件。而且由于是直接通信，不会因为第三方服务器中转而降低传输速度。

- ZeroTier
- Tailscale

另外， UniFi Dream Machine 等路由器，也通过 NAT 打洞等技术，使用户安全、方便地远程访问路由器。

TCP 也是一种比较常用的协议，为什么很少见到基于 TCP 的 NAT 打洞？

前面提到的 NAT 打洞，一般**都是基于 UDP 协议**的。但是，在 Internet 上，TCP 协议也十分常见。例如我们浏览网页、下载文件常用的 HTTP 协议，就是基于 TCP 的。

那么，NAT 打洞时，为什么不常用 TCP 协议？

这主要是因为，TCP 相对 UDP 复杂得多。UDP 协议是基于报文，一个一个报文收发的。而 TCP 需要先建立连接，然后才能传输数据。

所以，应用程序可以方便地发送 UDP 报文，进行打洞。而对于 TCP，操作系统底层处理了连接建立、断开等过程。应用程序无法方便地控制单个报文的发送。

另外，部分路由器的 NAT 实现，也会对 TCP 报文的状态进行额外检查。如果发送报文不属于某个连接，就会丢弃报文。

所以，TCP 打洞理论上能实现，但实现复杂，且成功率不高。如果需要进行 TCP 打洞，建议改用 QUIC 协议。QUIC 协议虽然基于 UDP，但引入了与 TCP 类似的可靠传输、拥塞控制等机制。最新的 HTTP3 协议，就是基于 QUIC 的。

- QUIC, a multiplexed transport over UDP - The Chromium Projects

NAT 打洞必定会成功吗

在网上，我们会常常看到关于「某个工具的 NAT 打洞成功率更高，而另一些工具经常打洞失败」的讨论。那么，如何理解这里说的「成功率」？

NAT 打洞为内网设备之间的直接通信，提供了可能性。但是，不同厂商、不同设备的 NAT 实现，不是完全相同的。另外，两台内网设备之间，可能会经过多个路由器和防火墙，更增加了复杂性。

所以，NAT 打洞无法保证完全成功。例如前文中介绍的对称 NAT，就无法保证 100% 成功。

所以，一个比较好的 NAT 穿透实现，会进行多种尝试：例如先尝试 UPnP IGD 和 NAT- PMP，然后尝试不同的 NAT 打洞方案。在最终无法打洞成功时，选择服务器中转等备用方案。

NAT 打洞是否完美解决了 NAT 引入的问题？是否存在缺点？

NAT 打洞虽然能让内网设备直接通信，那么，是不是大部分设备，都不需要公网 IP，直接位于 NAT 之后就行了。

其实不是这样，比起使用公网 IP 直接通信，NAT 打洞仍存在不少缺点，例如：

- NAT 打洞仍需第三方服务器的参与
- NAT 打洞无法 100% 成功，尤其是对称 NAT，更难打洞成功
- NAT 打洞的过程，需要开发者对应用程序进行适配
- 为了节省资源，路由器上的 NAT 表项会超时删除。所以，NAT 打洞后，需要定期发送报文，维持路由器上的 NAT 表项。否则需要重新打洞
- NAT 打洞的操作本身，也增加了延迟
- NAT 打洞对 TCP 的支持不佳，一般使用 UDP。但不少运营商会对 UDP 进行限速，导致打洞后虽然设备间能直接通信，但无法以较快的速率传输

NAT 的未来

根据前面的介绍，我们看到，NAT 在缓解 IPv4 地址资源不足的问题上，做出了巨大的贡献；同时，NAT 的出现，也避免了个人设备暴露在公网，「意外地」提升了安全性。

但是，NAT 带给我们更多的，是各种各样的限制…… 目前，IPv6 正在逐步普及，等我们彻底用上了 IPv6，NAT 可能就没有了存在的意义

NAT 也有一些其他用途，例如做为 IPv6 过渡技术，或者通过 IPv6 NAT 使 Docker 中的应用方便被外部访问。本文暂不讨论这些应用场景。

。

但是，为了保证安全性，家用路由器、PC、NAS 等设备上，一般都有防火墙功能。防火墙默认会阻止传入连接，除非用户手动配置防火墙，打开特定端口。所以，即使 NAT 被淘汰，类似 NAT 打洞的技术，在 IPv6 时代仍会得到应用。

那么，当 IPv6 普及，NAT 消失，Internet 是否会诞生新的有趣的应用？家中的每个物联网设备都有了公网 IPv6 地址，是否会有一些新的玩法？是否会带来新的安全性问题？随着国内大幅度推进 IPv6 的建设，这些问题应该很快会有答案。

参考链接、延伸阅读

关于 NAT 的具体实现，可参考如下 RFC：

- rfc1631
- rfc2663
- rfc3022
- rfc4787
- rfc5382
- rfc5508

在 NAT 环境中进行 P2P 通信，会遇到哪些问题？应该如何解决？可参考如下 RFC：

- rfc5128

已经有一系列协议，实现了完整的 NAT 打洞过程，可参考如下 RFC：

- rfc3489：STUN 协议。参考部分已有的 NAT 打洞技术，并将其标准化。
- rfc5389：另一个 STUN 协议（与 RFC 3489 全称不同，但缩写相同）。支持 TCP 打洞。
- rfc5766：TURN 协议。可实现 NAT 打洞失败后，使用服务器中转。
- rfc5245：ICE 协议。更完整的协议，除了如何打洞，还描述了打洞成功后如何通信。

这篇文章介绍了基于 NAT 穿透的虚拟局域网工具 Tailscale 的工作原理，可参考阅读：

- How Tailscale works · Tailscale

在本文的写作过程中，还参考了如下文章。文章中有关于 NAT 的更多知识，如果对 NAT 感兴趣，可结合本文一起阅读：

- How NAT traversal works · Tailscale
- P2P 通信原理与实现 - evilpan
- P2P 通信标准协议 (一) 之 STUN - evilpan
- P2P 通信标准协议 (二) 之 TURN - evilpan
- P2P 通信标准协议 (三) 之 ICE - evilpan
- NAT 穿透 - 维基百科，自由的百科全书
- P2P 通信原理与实现 (C++) - 有价值炮灰 - 博客园
- NAT 科普与类型提升 – 晨鹤小站
- 42 张图详解 NAT：换个马甲就能上网

本文配图素材来自 Flaticon，作者包括 Eucalyp、Freepik、Icongeek26、xnimrodx、Good Ware、srip、phatplus。

题图来源：

- rectangular gray corded device on black rack photo – Free Computer Image on Unsplash

对于家庭网络的更多内容，欢迎关注我在 GitHub 上创建的 Awesome List：

- blanboom/awesome-home-networking-cn

> 下载少数派 客户端 、关注 少数派公众号 ，了解更妙的数字生活

> 想申请成为少数派作者？ 冲!

全文完