

这篇文章要说的，是通过路由器设置 IPv6 地址来访问 IPv4 所无法访问的网站（主要是记录一下本次折腾过程以后可能有用，其实这个方法应该很早以前就可以使用了，不过无所谓啦）。目前 GFW 是没有针对 IPv6 有很好的限制手段（以后的话以后再说吧），只是针对 IPv6 的 DNS 解析污染是很严重的（IPv4 的 DNS 污染更严重，不过就算没污染也无法访问就不去考虑啦）。

两个前提条件：

- 一个公网的 IPv4 地址（无所谓动态还是静态）
- 一台支持刷第三方固件的路由器，我用的是 OpenWRT，所以别的也就不清楚啦（如果就单单一台电脑需要上网，那就更简单了，就不多说明了，只在下面提一下）

准备好路由器

我的路由器是网件的 R6250 和华硕的 AC55U（掩面，都是渣渣，不过能用）。R6250 是用于做外网网关的，AC55U 是相当于做 AP 的（因为 R6250 刷了 OpenWRT 后无线信号渣成翔啊有木有）。关于如何给 R6250 刷 OpenWRT 就不说了，资料一查一大堆（其他也一样，主要是讲原理）。

去 [tunnelbroker.net](https://tunnelbroker.net) 申请一个 **IPv6** 地址

虽然部分地区已经有运营商分 IPv6 了，但国内大部分还是就电信有 IPv6，而且渣的像 X 一样，完全没法用，分到的应该也就一个 /64 的地址，就只能用于一台设备，如果需要用于给多台设备分 IPv6，还需要更复杂的设置。（其他运营商的 IPv6 还没有用到过，当然教育网的 IPv6 非常好）

关于 IPv6 和 IPv4 的区别可以看[这篇文章](#)，从实际体验上来看，用了 IPv6 后你会发现的是基本不存在局域网的概念了

HE 公司的 tunnelbroker.net 上一个账户可以申请最多 5 个 IPv6 地址，使用原理是把 IPv6 封包再封成为 IPv4 封包，通过 IPv4 网络发送到 HE 提供的服务端解包后去访问世界上的 IPv6 网络。

除了这个还有国内的比如 6plat、6box 这样子的服务，前者申请麻烦还要提供身份证，且需要先设置 DDNS，后者有点贵，所以我不考虑了。

还有一种是可以使用清大 ISATAP 隧道，但是这个基于一个公网 IP 只能分一个 /64 地址，且会根据公网 IPv4 地址不同而变化，可以看如下几篇文章：

- [关于如何在 nat 后面使用 isatap](#)
- [github.com/tuna/ipv6.tsinghua.edu.cn/blob/master/isatap.md](https://github.com/tuna/ipv6.tsinghua.edu.cn/blob/master/isatap.md)
- [OpenWRT 配置 IPv6 的 NAT（一般结合 isatap 使用）](#)

我使用 Windows 尝试了一下上述第二篇文章的内容，不过失败了，没去深究。有兴趣的可以看看，听说清华的 ISATAP 隧道速度带宽都很足。

我根据自己的实际地理位置（江苏电信）选择的是一个 Los Angeles 的端口，创建好后在 `IPv6 Tunnel` 这个标签下会有几个信息：

- Server IPv4 Address
- Server IPv6 Address
- Client IPv4 Address
- Client IPv6 Address

顾名思义，不多解释，需要解释的是下面 2 个：

- Routed /64 这个是默认就有的，用来给本地网络其他设备分配公网 IPv6 的。HE 已经自动把这个地址段路由到 Client IPv6 Address 了。
- Routed /48 这个默认不生成，但是可以点击后面的 Assign/48 去请求分配一个，同样已经自动把这个地址段路由到 Client IPv6 Address 了，用途是这样子的：如果本地不止一个路由器来分配 IPv6，那么可以给主路由分配这个前缀上的一个地址，然后给其他路由分配主路由上这个前缀对应的 /64 地址作为前缀再分配给其他路由连接的设备，解释可能有点苍白，后面设置的时候就知道了。

tunnelbroker.net 对应页面有更详细的解释，可以自己去看。

## 给路由器设置好 **IPv6** 地址和路由

我这里已经被分配到电信的 IPv6 地址了，但正如我上面所说，渣成 X，所以可以在 OpenWRT 的设置界面上取消获取 IPv6（不多解释如何操作了，不取消也没关系）。

在 Tunnelbroker 创建好了 IPv6 后 `Example Configurations` 标签页有很多系统下的配置例子，非常方便，OpenWRT 选择 `OpenWRT Barrier Breaker` 这个的配置比较合适，我贴在下面来说明：

```
root# opkg update                                # 如果重启了系统后没有更新过这个软件库的话一定要先更新一下，不然会搜索不到
root# opkg install 6in4                          # 这个是启用 Tunnel IPv6inIPv4 必备的，其他依赖会自动安装

root# uci set network.henet=interface            # 从这里开始一组是用来新建一个端口的，看你上面的配置，别看我的
root# uci set network.henet.proto=6in4
root# uci set network.henet.peeraddr=216.x.x.x
root# uci set network.henet.ip6addr='2001:470:x8:xxxx::2/64'
root# uci set network.henet.ip6prefix='2001:470:fxxx::/48' # 这个就是我上面有解释的 Routed /64，注意：就算请求生成了 Routed /48 也不会在这边显示，但是你替换一下就好了
root# uci set network.henet.tunnelid=xxxxxx
```

```
root# uci set network.henet.username=xxxxxx
root# uci set network.henet.password='UPDATE_KEY_OR_PASSWORD' # 这边需要输入你的更新密码, 在 'Advanced' 标签下有个 'Update Key' 。
如果你的公网 IPv4 地址是动态的, OpenWRT 已经有内置了更新的脚本, 也是通过这个密码来更新的
root# uci set network.henet.defaultroute='0' # 这个设置在 tunnelbroker 上是没有写的, 但是我建议设置一下, 不然之后
的路由会有问题(至少我这里是这样的)
root# uci commit network # 提交修改

root# uci set firewall.@zone[1].network='wan henet' # 下面的是通用的设置和重启服务
root# uci commit firewall

root# /etc/init.d/network restart
root# /etc/init.d/firewall reload
```

设置路由, 这块是重点

当设置好了上面的部分后, 路由器上目前的地址和路由情况应该是类似这样子的:

```
root# ip -6 addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qlen 1000
    inet6 fe80::526a:3ff:feeb:1b0b/64 scope link
        valid_lft forever preferred_lft forever
7: br-lan: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500
    inet6 2001:470:fxxx::1/48 scope global noprefixroute
        valid_lft forever preferred_lft forever
    inet6 fe80::xxxx:3ff:feeb:1b0b/64 scope link
        valid_lft forever preferred_lft forever
11: 6in4-henet@NONE: <POINTOPOINT,NOARP,UP,LOWER_UP> mtu 1280
    inet6 2001:470:x8:xxxx::2/64 scope global
        valid_lft forever preferred_lft forever
    inet6 fe80::xxxx:964d/64 scope link
        valid_lft forever preferred_lft forever

root# ip -6 route
```

```
2001:470:x8:xxxx::/64 dev 6in4-henet proto kernel metric 256
2001:470:fxxx::/64 dev br-lan metric 1024
unreachable 2001:470:fxxx::/48 dev lo metric 1024 error -101
fe80::/64 dev eth0 proto kernel metric 256
fe80::/64 dev br-lan proto kernel metric 256
fe80::/64 dev 6in4-henet proto kernel metric 256
```

这是无法正常使用的，因为我是 2 个路由器，所以我是按照 /48 的配置的，/64 的只会更简单。

首选判断出上面错误的路由有 2 个，2001:470:fxxx::/64 dev br-lan metric 1024 和 unreachable 2001:470:fxxx::/48 dev lo metric 1024 error -101。删掉它们：

```
root# ip -6 route del 2001:470:fxxx::/64 dev br-lan
root# ip -6 route del unreachable 2001:470:fxxx::/48 dev lo
```

然后要建立起有用的路由表，设定好默认路由，先让主路由可以通过 IPv6 连接上互连网络，需要这样子添加：

```
root# ip -6 route add default via 2001:470:x8:xxxx::1 dev 6in4-henet # 注意这里需要指定一下连接互联网的网关地址，就是 HE 提供的 'Server IPv6 Address'
```

在这里需要定好几点：

0. 我的主路由用什么网段来分配给连接该路由器的其他设备？A: 2001:470:fxxx::/64
1. 我的另外一台路由器用什么地址做网关？A: 我定为 2001:470:fxxx::1/64
2. 我的另外一台路由器用什么地址做 IPv6 的 WAN 口地址？A: 我定为 2001:470:fxxx::2/64
3. 我的另外一台路由器用什么地址段来分配给连接该路由器的其他设备？A: 我定为 2001:470:fxxx:10e1::/64

因为都是通过 br-lan 这个网桥连接的，所以要这样子添加：

```
root# ip -6 route add 2001:470:fxxx::/48 dev br-lan # 这边因为我可以完全控制 2001:470:fxxx::/48 这个 IP 段，所以直接 /48 不用 /64
root# ip -6 route add 2001:470:fxxx:10e1::/64 via 2001:470:fxxx::2 dev br-lan
```

这样子主路由器的路由就设置好了，可以把上述删添路由的命令加入到 OpenWRT 的开机启动脚本下。

测试发现，当重启路由器后可能不会自动去更新 tunnelbroker 服务器上的本地 IPv4 记录（非重启重新拨号是可以自动更新的，OpenWRT 下），那么可以在开机脚本添加如下命令（关于这个链接可以在 tunnelbroker.net 的 Tunnel Details 页面上的 Advanced 标签下看到）：

如果不需要路由器，直接电脑设置的话，tunnelbroker.net 的 Tunnel Details 页面上的 Example Configurations 标签下有很多线程的系统的配置，Linux 和 Windows 的都有。

```
curl https://xxxxx:xxxxxxxxxxxxx@ipv4.tunnelbroker.net/nic/update?hostname=xxxxxx
```

然后下面就是设置第二个路由器的 IPv6 了，找到设置 IPv6 的界面，选择静态 IPv6 地址，WAN 口 IP 设置为 2001:470:fxxx::2，前缀设置为 64 位，网关设置为 2001:470:fxxx::1，LAN 口的 IP 设置为 2001:470:fxxx:10e1::1，前缀设置为 64 位。每个路由器设置位置有差异，具体如何设置就不说了。（注意这边设置 DNS 服务器的时候，一定要设置为主路由器的网络地址，也就是 IPv6 的话是 2001:470:fxxx::1，IPv4 的话根据你的路由来看，为后文做铺垫，用于解决 DNS 污染问题的）

这样子设置后如果不出意外，连接两个路由器的设备就都应该可以被自动分配到 IPv6 地址了。（目前应该都是默认安装好了 DHCPv6 了）

## 解决 DNS 污染的问题

一般来说有 2 种方法来解决，一种是设置默认使用 TCP 方式连接 DNS 服务器查询，一种是将常用的被污染的 DNS 设置成为 hosts 文件查询。从长远的便捷性来考虑的话，使用第一种比较好，不过我差了一下一个不错的实现方案 (pdnsd) 已经不再维护，还有一种 (unbound) 评价说速度太慢，加上我也懒得再装软件配置，就选择了第二种方案，使用 hosts 文件 + 路由器上本身自带的 dnsmasq 服务解决 IPv6 DNS 污染问题。

很简单，过程如下：

首先我是找到了一个关于 Google 的 ipv6 hosts 的项目，地址是：<https://github.com/lenny1xx/ipv6-hosts>，因为我端口出口在洛杉矶，所以它默认的解析地址本不好用，于是我临时用 DigitalOcean 开了一台 San Francisco 的 VPS 使用项目的脚本重新生成了一遍 hosts 文件在[这里](#)，然后把它与路由器原本的 /etc/hosts 文件合并，并打开路由器的设置界面，设置好 dnsmasq 的上游 DNS 服务器其他保持原样，最后设置另一台路由器默认的 DNS 服务器以及主路由器默认 DNS 服务器均为主路由器即可。

完成！

有一个问题：我的 Windows 10 通过网线连接主路由器的时候，怎么也无法分配到正确的 IPv6 地址，但是我的 Gentoo 可以非常轻松的获取到。尝试了很多次无果，我默认他们八字相冲吧，等 OpenWRT 版本更新后看看会不会有好转。Windows 10 通过 WiFi 去连接另外一台路由器就非常正常。总之现在已经可以正常使用了，我的 iPad 也可以正常通过 IPv6 访问网络了。