如何让VMWare虚拟机走物理机的代理VPN流量

不同于传统的隧道型VPN，只有**应用层支持代理协议的程序**才能走代理VPN，例如浏览器。不支持代理的协议（比如ICMP）依旧会走物理机直连，不会被代理。例如在物理机terminal中运行curl ifconfig.me，显示本机IP而非代理IP。本文后续提到的VPN，均指代理型VPN。

经测试发现，VPN开启后，NAT模式下的虚拟机通过VMWare**的虚拟NAT网关**出去，再使用宿主机的物理网卡访问外部网络。虚拟机的流量对外表现为物理机IP。因此无法访问外网。

点击VPN程序的终端代理，会返回：

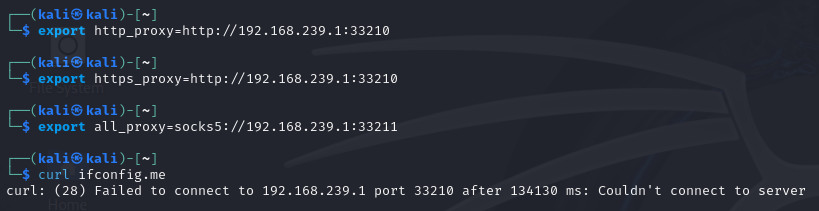
export https\_proxy=http://127.0.0.1:33210

http\_proxy=http://127.0.0.1:33210

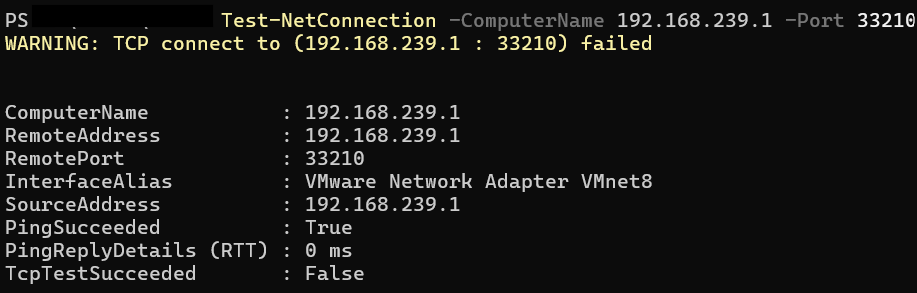
all\_proxy=socks5://127.0.0.1:33211

由此确认该VPN是**本地代理模式**，只在监听物理机的回环地址代理 (127.0.0.1)，不监听来自NAT网关的流量。现在尝试让虚拟机**能够访问物理机上的代理**。

首先尝试在虚拟机内把终端代理指向物理机的NAT IP＋指定端口：



发现curl请求超时，但是ping成功。Ping用的是**ICMP 协议**，它直接走网络层，不会经过 HTTP/SOCKS代理。Curl默认走**HTTP/HTTPS 协议**，虚拟机发送到物理机代理的被阻止。猜测是防火墙问题。在物理机PowerShell进行TCP测试，确认端口访问被阻止：



此时选择在物理机建立端口转发，把 127.0.0.1:33210映射到物理机的NAT IP，这样虚拟机访问 NAT IP时就能通过VPN出口。

**Windows端口转发:**

netsh interface portproxy add v4tov4 listenaddress=192.168.239.1 listenport=33210 connectaddress=127.0.0.1 connectport=33210

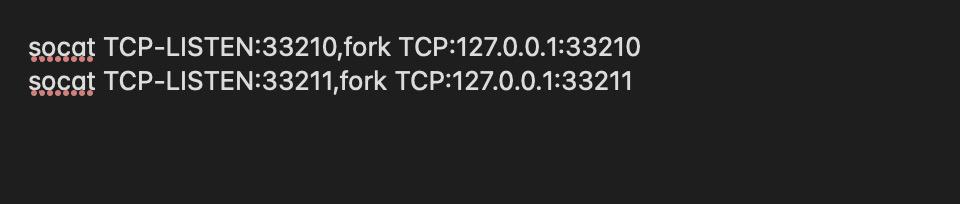
netsh interface portproxy add v4tov4 listenaddress=192.168.239.1 listenport=33211 connectaddress=127.0.0.1 connectport=33211

**Windows 防火墙端口放行：**

New-NetFirewallRule -DisplayName "VM NAT VPN Proxy 33210" -Direction Inbound -LocalPort 33210 -Protocol TCP -Action Allow

New-NetFirewallRule -DisplayName "VM NAT VPN Proxy 33211" -Direction Inbound -LocalPort 33211 -Protocol TCP -Action Allow

MAC:



完成后，虚拟terminal中执行curl ifconfig.me，确认IP已变为VPN IP。但是在浏览器中测试，IP没变。此时可以为虚拟机设置全局代理：

sudo vim /etc/environment

并添加：

http\_proxy="http://192.168.239.1:33210"

https\_proxy="http://192.168.239.1:33210"

all\_proxy="socks5://192.168.239.1:33211"

至此虚拟机设置完成，后续可以清楚端口转发以及防火墙放行：

netsh interface portproxy delete v4tov4 listenaddress=192.168.239.1 listenport=33210

netsh interface portproxy delete v4tov4 listenaddress=192.168.239.1 listenport=33211

Remove-NetFirewallRule -DisplayName "VM NAT VPN Proxy 33210"

Remove-NetFirewallRule -DisplayName "VM NAT VPN Proxy 33211"