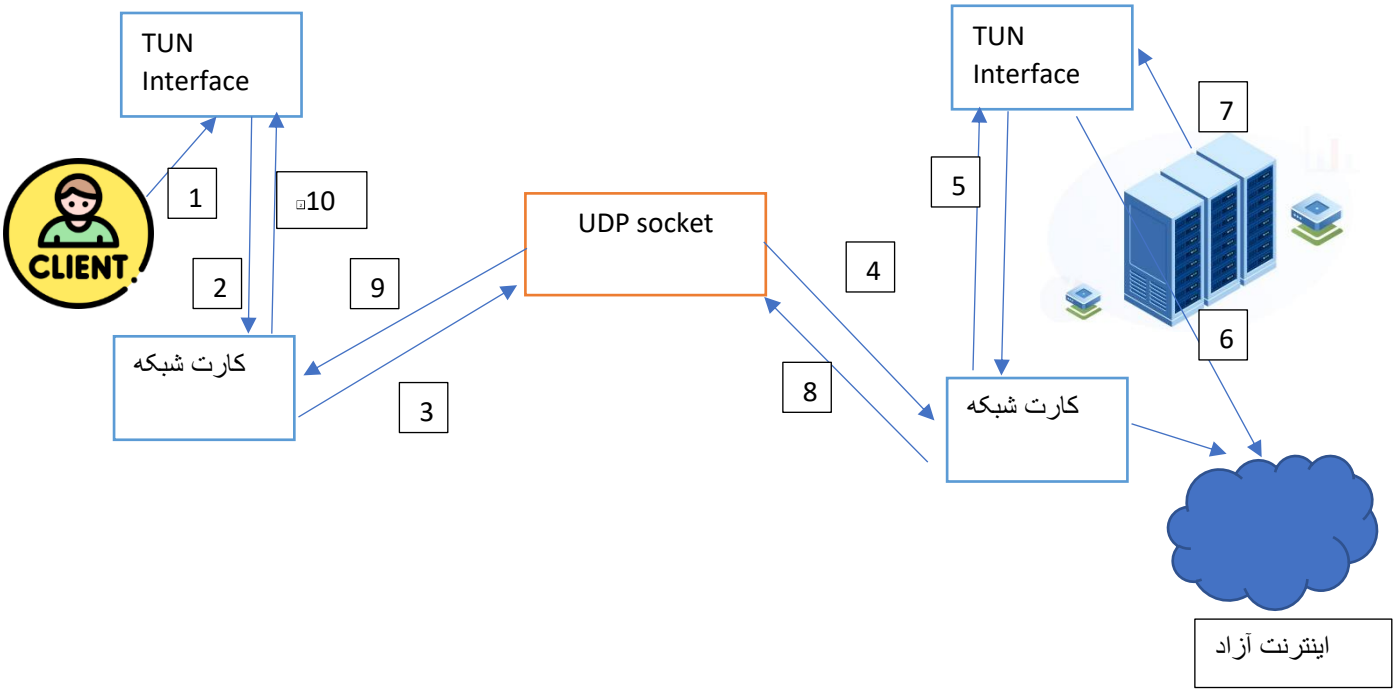
مقدمات

هدف این پروژه زدن یک سرویس کلاینت و سروری برای یک VPN می باشد. به صورت کلی تمام کاری که لازم است برای این پروژه انجام شود این است که پکت های TCP که بر روی سیستم شما به سمت اینترنت ارسال می شوند را به نحوی (که جلوتر بحث میکنیم) دریافت و سپس بجای فرستادن آن ها به سمت اینترنت آزاد (که ممکن است فیلتر باشد) به یک سرور که در حال listen کردن بر روی پورت خاصی می باشد بفرستیم. این سرور درواقع همان سرور VPN ما می باشد و وظیفه دارد تا پکت هایی که دریافت می کند را به سمت اینترنت آزاد ارسال و جواب آن ها را برای ما روی همان کانکشن باز شده برگرداند. به صورت خلاصه هدف ما در زدن این پروژه، کامل کردن این کلاینت و سرور می باشد.

چرا به TUN/TAP نیاز داریم و دقیقا چه کار میکنند؟

در توضیحات بالا اگر دقت کنید گفتیم که نیاز است پکت های TCP که بر روی سیستم قرار است ارسال شوند را به نحوی کپچر کنیم و آن ها را تغییر بدهیم. برای این کار نیاز به ساختن نوعی interface مجازی داریم. اگر آز شبکه پاس کرده باشید interface محل اتصال بین کامپیوتر و اینترنت می باشد. درواقع با ساختن یک interface و هدایت کردن پکت ها با مبدا خاص به آن interface می توانیم تمامی پکت هایی که آدرس خاصی دارند را ابتدا دریافت و تغییراتی روی آن ها اعمال کنیم. همچنین پکت هایی که به آدرس خاصی ارسال می شوند که با آن range همخوانی دارد را ابتدا دریافت و سپس میتوانیم تغییراتی روی آن ها اعمال کنیم. بیاید دقیقا سناریویی که چه بر سر یک پکت می آید را باهم بررسی کنیم تا توضیحات گنگ بالا :) واضح تر شود.



گام ها را به ترتیب میگوییم (bear with me until then end, every thing will be clear)

در این گاید جزئیات دستور هارا فعلا نمیگوییم تا شهود کلی کار به دست بیاید.

از encapsulate کردن درون پکت های DNS فعلا برای سادگی خودداری میکنیم.

شما به عنوان کلاینت میخواهید به سایت neverss.com یک curl بزنید و از قضا این سایت فیلتر است.

1- شما در لپتاپ خودتون curl neverssl.com را میزنید.
2- برای تست پکت هایی که به ip nerverssl هستند را در ip route تغییر دهید تا بر روی interface tun ای که ساخته اید هدایت شوند. (پس یک اینترفیستun باید ساخته باشید و در شبکه اتان ip route ان را عوض کرده باشید)
در این مرحله شما اگر دستور ip -c a بزنید باید یک interface جدید مثلا بنام tun0 داشته باشید و پکت curl ما چون به این interface هدایت شده است بر روی کارت شبکه قرار نخواهد گرفت و سمت interface دریافت می شود.
3- حال که پکت را در interface و درون کد دریافت کرده اید دقت کنید ساختن اینترفیس درون کد است و شما یک while true طور دارید و با خواندن مکرر از tun هنگامی که پکت به ان هدایت شد آن را کاملا دارید، لازم است تا کار های مورد نظر را بکنیم. حال لازم است تا پکت را بجای فرستادن به سمت neverssl، به سمت سروری که داریم بفرستیم. پکت را بر روی یک پورت UDP یا TCP به سمت سرور میفرستیم.
4- سرور پکت ما را دریافت میکند و آن را بر روی interface tun خود مینویسد. از آنجایی که در سرور NAT اینگ مناسب لازم است انجام شود پکت هایی که از source این interface tun باشند NAT خواهند شد و به سمت اینترنت آزاد هدایت می شوند.
5- حال دقت کنید که جواب این پکت نیز بر روی TUN خوانده خواهد شد و پاسخ را (پکت پاسخ) به سمت کلاینت بر روی پورت سوکت UDP مینویسیم.
6- در این مرحله کلاینت پکت را دریافت کرده و بر روی interface TUN خود آن را مینویسد و کار تمام است.

برخی نکات و مشکلات

1- در طی پروژه برخی باگ ها به وجود آمد که تا امروز دلایل آن ها برایم آشکار نیست.

```
Sudo ip addr add subnet dev TUN_NAME
sudo  ip link set up dev TUN_NAME
```

دستورات بالا برای ساختن TUN استفاده میشود لازم است آن را درون یک کد پایتون اجرا کنید. می توانید برای این کار از subprocess.run استفاده کنید. به صورت عجیبی اگر از os.system استفاده میکردم کار نمیکرد . ممکن است باگ از جای دیگری باشد(احتمالا ولی گفتن این نکته خالی از لطف نبود)

2- اگر از نظر مالی برایتان مشکل ندارد 2 سرور بخرید (از پارس پک سرور ایران روی 10 هزارتومن تقریبا بود فک کنم) . من توانستم با ترکیب vmbox و wsl پروژه را ران کنم اما یکی از دوستانم با vmware مشکلات عجیبی میخورد

3- اگر با EDNS و کتابخانه scapy کار میکنید به ساختار درختی آن توجه ویژه بکنید. کافی است یک پکت رندم TCP بسازید و درون EDNS قرار بدهید و show. بزنید تا ساختار درختی آن مشخص شود. اگر ساختار درختی را به درستی رعایت نکنید پکت ها ، به پکت DNS تبدیل نمی شوند. (به علت اینکه اگر جزئیات بیشتر کدی بگویم احتمالا تقلب یا .. حساب شود به صورت کلی این موضوع رو گفتم) . ساختار درختی آن با یک فیلد ar شروع و ... در نهایت به اتمام میرسد. دقیقا مانند درختی که show میشود جلو بروید و فیلد هارا ست کنید.

4- به دنبال رفتن برای دستورات پیچیده NAT نروید. دستورات ساده جواب است و مشکل از جای دیگری است . به شخصه چندین نوع مختلف تست کردم ولی همان ورژنی که در کارگاه میزنند کافی است .

https://github.com/ArshiAAkhavan/ark

(کد کامل اقای اخوان).

البته قابل ذکر است برای اینکه این کد NAT اینگش کار کند ادرس سابنت ها باید دو طرف یکی باشند. (اگر درست متوجه شده باشم)

5-

```
sudo sysctl -w net.ipv4.ip_forward=1
```

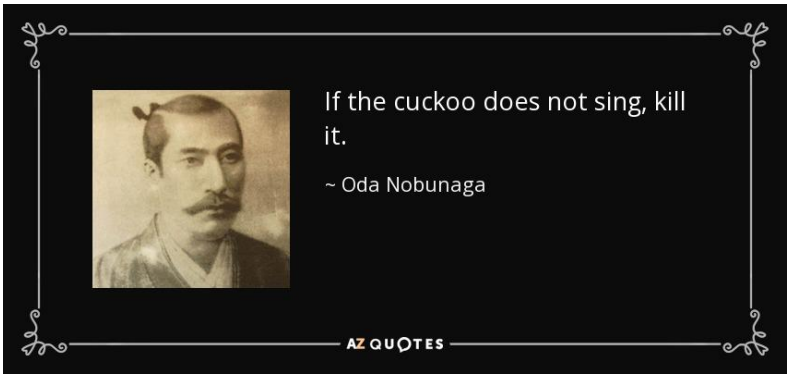
این دستور را برای اجازه NAT شدن فراموش نکنید.

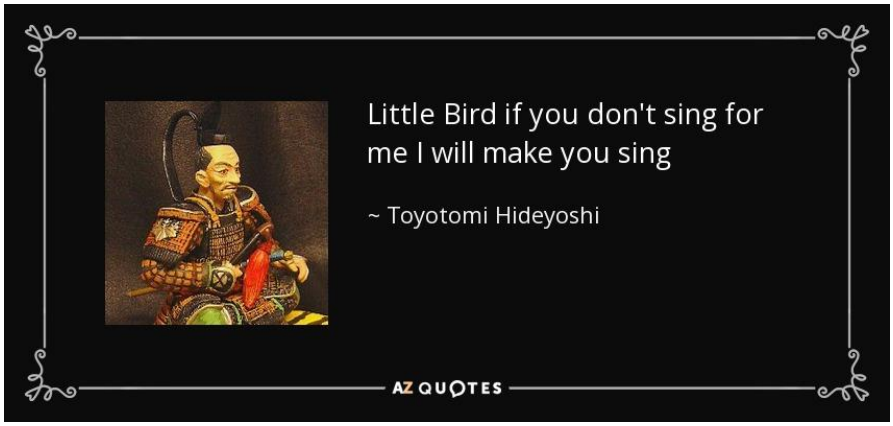
در نهایت بنظرم بعد فهمیدن نحوه کار کردن پروژه و جزئیات TUN TAP EDNS و Scapy (کتابخانه کار های ip طور) بهترین کار شروع تبدیل گام به گام کد اقای اخوان می باشد.
https://github.com/ArshiAAkhavan/ark

6- برخی نکات جانبی مربوط به scapy.

- برای ساختن دوباره internet checksum کافی است ان را دیلیت و دوباره کال کنید. (اگر از این کتابخونه استفاده میکنید)
- هنگام تغییر MSS کافی است روی فیلد ها for بزنید و فیلدی که مربوط به MSS هست رو کم کنید.(opt[0]=MSS)
- در هرمرحله کتابخونه scapy به شما قابلیتshow. میده . روی پکت های TCP/IP/.. برای دیباگینگ ()show. کال کنید و مطمئن شید پکت درستی دارید میفرستید اگه جواب نمیگرفتید

در اخر میخواهم از جای اضافه برای گذاشتن quote 3 جالب از the three great conquerors of Japan بگذارم





شما با کدام یکی از این 3 تا موافق هستید؟ اگر هم نظر هستیم بیاید یک coffee بزنیم و بقیه پروژه را شاید برایتان بزنم :دی