Implement if:then function for split tunnel (for lack of a better term) Under Review

- S Shane
- **Forum name:** #Suggestion

First let me say I'm super grateful for the split tunnel UI that we already have. I don't want to imply that it's bad or inadequate, it's great and already saves a lot of time. I just think it could be improved even further, and that improvement would be massive and would cost very little. I don't know if there's a better phrase for what I'm trying to describe, so I'll just call it an if:then function. What I mean is, it would be good to have an option in the split tunnel UI to define a rule like:

if (program A is running and is connected to the VPN) {

program B bypasses the VPN

} else if (program A is not running and connected to the VPN) {

program B must connect to the VPN
}

That way if you're trying to hide program A, you can use program B to get around VPN blocking without having to turn off the VPN. But if you're not trying to hide program A, and are instead trying to hide program B, you don't have to wait for the VPN to connect. Basically give us a way to define one program as always-VPN, and another program as a non-VPN backup that still uses the VPN if the first program isn't running.

The application I have in mind which led me to wishing this feature exists is using a browser while downloading torrents or doing some other private network activity that you're trying to protect or isolate. For example I use stremio and I connect to the VPN so I will not be exposed to any unscrupulous peers. But while I'm watching a film or whatever I often get the impulse to google something. Like maybe I'm watching a documentary about Julius Caesar and suddenly I just feel the need to search up the religious customs of the Gallic tribes. Only when I do that, I get instantly brick walled by not only google but now, even wikipedia.

I have been a loyal customer for several years and have never even tried another VPN since I started using PIA. But I don't hear customers of the other big VPNs complaining that even wikipedia blocks them. In the past it was just google's image hosting, and I could stomach that I guess. But then it expanded to include practically google's entire service and apparently some of cloudflare's CDN hosts? Then youtube stopped working, and just this last month (for me) apparently EVEN wikipedia has blocked us, or some link in that chain. That's shocking because I thought wikipedia only blocked editing for known VPN IP ranges. Now I can't even load a page on wikipedia most of the time, and if I can, none of the images load. I use the "auto" option so I'm generally connecting to california, west, or las vegas, for what that's worth.

Anyway I don't complain about getting blocked by these sites because I know the low turnover of servers and the use of comparatively "tainted" IP ranges is what keeps the prices down. I only mention this stuff now to explain why I think this kind of feature is important. If you're going to keep using IPs that are banned by half the most trafficked websites on the internet, then we should have at least some basic tools that require very little development time to implement, and zero ongoing costs. Just a simple if:then rule would let me set Stremio to always-VPN, and set my browser to non-VPN if Stremio is running/connected-to-VPN, but always-VPN otherwise.

And I don't think that would require much, if any, changes to your backend and is really just simple logic you could layer on top of the existing split tunnel system. And for that tiny investment, I would no longer have to disconnect from the VPN, (and completely stop the torrent and quit stremio lest my IP leak) do my browser stuff, then reconnect to the VPN and

the torrent. It would completely eliminate this super-irritating process of quitting programs and waiting for the VPN to disconnect and reconnect, and so on.

Also, I'm aware one could implement this kind of behavior in a script, I just think this is the type of usage that is extremely common among your customers. For better or worse, most people are using this kind of VPN to protect certain traffic and would like to be able to simultaneously avoid VPN blocks without needing to give an entire program a universal bypass. So if the UI natively supported this kind of rule, as long as you did an adequate job of designing the UI and explaining the feature in the changelog, (just like you already did with the existing split tunnel system) I think it would see a lot of use and it would be a very clear advantage for PIA over other VPNs. Even if it's technically easy, the average user simply isn't going to bother looking into scripting to even discover that they could write this kind of behavior themselves. So baking it into the basic UI is just smart business.

Thanks so much!