A VPN-szerverek beállítása könnyen NetworkManager-ben

PCLinuxOS Magazine – 2025. május

Írta: Paul Arnote (parnote)

Miért VPN?

Kétségtelenül növekvőben van azok száma, akik VPN-t (Virtual Private Network = virtuális magánhálózat) használnak. A növekvő félelem a kormányzati felügyelettől és túlkapásoktól, kiegészülve a kormány és a vállalatok részéről a folyamatosan növekvő fenyegetés a személyhez köthető adatok gátlástalan gyűjtése miatt a VPN-ek használatát vonzó alternatívává tették. A VPN nagyon vonzó lehetőség azoknak, akik nagyon ügyelnek a magánszférájukra és azoknak, akik az online működésükben valamivel több névtelenséget szeretnének élvezni.

Természetesen, amikor bejelentkezel egy oldalra, minden ténykedésedet rögzítik, vagyis a VPN nem véd meg az adott oldalon a ténykedésed adatainak gyűjtésétől. Továbbá, (nagyon valószínű) nem véd meg a „Facebook Pixel”-től sem. A monstrum a weblapok zömén (a magazinén NEM) jelen van és gyűjti az adatokat minden látogatóról, függetlenül attól, hogy Facebook-felhasználó-e, vagy sem. De a VPN segít megvédeni téged attól, hogy rögzítsék a tevékenységedet, ahogy weblapról, weblapra lépegetsz.

Az egyik, ami követ téged, az Internetszolgáltatód (ISP). Az ISP figyeli az internetfiókod használatát, hogy tudja, nem csinálsz-evalami „törvénytelent” (idézőjelbe rakom, mert azt, hogy mi illegális, a környezettől függ), és hogy a feltételezett érdeklődési körödnek megfelelő célzott hirdetéseket szolgáltasson. Persze az „érdeklődést” a felkeresett oldalak és azokon a ténykedésedadja meg.Igen, ezt hívják „metaadatnak” és ezekből egy, vagy kettő önagában nem árul el túl sokat rólad. De a metaadatok hatalmas mennyiségét összerakva sokatelárul az érdeklődési körödről, az életkorodról, a jövedelmedről és arról, hogy egyedülálló vagy-e, vagy házas, vannak-e gyerekeid, mit szeretsz enni, mik a hobbijaid és egy jó csomó egyébről, amihez egyébként senkinek sincs semmi köze. Gondolj úgy a metaadatokra, mint egy 10000 darabos puzzle elemeire. Egy, vagy kettő önmagában nem sokat fed fel. Ám a metaadatok szó szerinti tonnáit összerakva (mint egy puzzle-t) sokkal több derül ki rólad, mint amennyit valószínűleg gondolsz, vagy észreveszel.

Szerencsére, VPN-t használva a forgalom a számítógéped és a VPN-t biztosító szerver között titkosított, így az ISP-d nem látja, hogy te mit csinálsz. Mindössze a forgalmat látja. Csak abban kell biztosnak lenned, hogy nincs DNS-szivárgás, ami az ISP-dnek titokban elárulja, hogy mit csinálsz. Megnézheted, hogy van-e DNS-szivárgás, ha meglátogatod e két honlap valamelyikét. Az első az IPLeak.net. Itt megláthatod a szerver címét, amihez csatlakozol, és a DNS-szerver címét, amihez kapcsolódsz. A második oldal a DNSLeakTest.com, Az utóbbi CSAK a DNS-szerver címét ellenőrzi, amihez kapcsolódsz.

Ha bármelyik oldalon látod, hogy az ISP-d IP címe megjelenik az oldal DNS-részénél, adatokat szivárogtatsz a DNS-szervered beállításain keresztül. A DNS-szivárgás az összes szükséges információt megadja az ISP-dnek, hogy követhessen a weben, még VPN mellett is. Szerencsére ezen könnyen változtathatsz. Amikor beállítod a hálózatkezelőt, meghatározhatsz alternatív DNS-szolgáltatót. Én inkább a CloudFare ingyenes DNS-szervert használom (1.1.1.1 és 1.0.0.1), mintsem a Google-ét (8.8.8.8 és 8.8.4.4). A Gogle már túl sokat tud rólam, nem akarom „tálcán kínálni” a DNS-információimat is.

Ahogy azt a múlt hónapi Firefox TOU-cikkben jeleztem, nem vagyok egy „összeesküvő” típus. Ugyanakkor némilegelvárom, hogy magamban utazgathassak a neten. Úgy vélem,

ahhoz senkinek sincs semmi köze, hogy mit csinálok, merre járok, kivel beszélek, vagy egyebek. Kicsit olyan mint a postai levél. Amikor postán keresztül küldök levelet valakinek, joggal várom el, hogy a levél tartalma privátban maradjon köztem és aközött, akinek címezni szándékoztam. Az online tevékenységem sem lehet másképpen.
Nem vagyok olyan sem, aki „Én nem tettem semmi rosszat, így nem kell rejtőzködnöm.” Egyszerűen senkinek sincs ahhoz köze, hogy mit csinálok a neten. Ezt hívják privátnak. Hogy a online magánéletre és a postai levélre miért tekintenek másképpen, teljesen érthetetlen számomra. Az előbbi csak az utóbbi modern változata, mégis az utóbbi kap nagyobb védelmet. Ettől „megzavarodom”.

Azt is tartsd észben, hogy vannak oldalak, amik nem szeretik, ha VPN-en kapcsolódsz. Könyörtelenül akadályozzák, hogy VPN-t használj az oldalaikon különböző okokból. Ez esetben három lehetőséged van. Első, teljesen elkerülöd azt az oldalt, ami nem mindig célszerű, vagy lehetséges. Második, hogy másik VPN-szervert választasz. Lehet olyan szerver, amit „blokkolnak”, miközben a VPN-szolgáltatód másik VPN-szerve működik. Harmadik, hogy időlegesen kikapcsolod a VPN-kapcsolódást az adott oldalhoz, majd újra bekapcsolod a VPN-t amikor végeztél az adott oldalon a tevékenységeddel.

A VPN és a NetworkManager (hálózatkezelő)

A NetworkManager megjelenéséig a PCLinuxOS-felhasználók az online-kapcsolódás kezelésére a net-applet és a wicd közül választhattak. És amikor megjelent a VPN-kapcsolat szerintem a földön csak két ember volt sikerrel képes a PCC-eszközével beállítani VPN-kapcsolódást. Komolyan! Apropó, nem én voltam az egyik. Ezért kitaláltam egy alternatív módot arra, hogy a net_applet-et használva képes legyek csatlakozni a VPN-emhez. Köszönhetően a NetworkManager megjelenésének, a „régi” szkriptemre már nem volt szükség, többé nem releváns.

A NetworkManager nem csak az online-kapcsolat beállítását tette sokkal DE SOKKAL könnyebbé, de a VPN-ed beállítását is sokkal, de sokkal egyszerűbbé tette. És ez az amiről ez a cikk szól.

2023 szeptemberében valami nagyon elromlott a NetworkManager-nél a frissítés után. Valamilyen okból, amit már elfelejtettem, a felhasználók nem tudták szerkeszteni a kapcsolatot a NetworkManager-ben, A NetworkManager-t beállító párbeszéd összeomlott. Szerencsére nem tartott sokáig ez az állapot, mivel a következő frissítés „befoltozta a léket a hajón”. Ám az esetet megvitatva a fórumon, néhány nagyobb koponya előállt a VPN-szolgáltató ÖSSZES szerverének importálásának módjával.

Hogy teljesen őszinte legyek akkoriban terveztem, hogy a szkriptről írok a magazinban, de elfeledkeztem róla … egészen addig, amíg el nem keveredtem az „úti” laptopom telepítésében (erről a 2025. márciusi számban írtam) és mindent újra kellett telepítenem a laptop SSD-jének cseréjekor. Természetesen, el akartam érni a VPN-emet a laptopról is, ezért újra átnéztem a szripte(ke)t, amik a fórumos egyeztetés során kerültek elő.

És egészen az adott VPN-nél (az én VPN-szolgáltatóm a Private Internet Access, alias PIA) elérhető összes VPN-szervert importáló szkript előkerüléséig nálam a PIA által a világon elérhető 99 szerverből sosem volt több mint hat beállítva használatra. Mindegyiket kézzel állítottam be. Ám a szkript első ízben elérhetővé tette mind a 99 VPN-szervert.

Ugyanakkor, mielőtt tovább mennél, a VPN-szolgáltatódtól le kell töltened az OpenVPN-fájlokat. Én a „strong” encryption (erős titkosítás) jelzésű OpenVPN-fájlok használatára hajlok. Senkinek sem akarom könnyebbé tenni az online tevékenységem utáni kémkedést. Csak csomagold ki az OpenVPN-fájlokat tartalmazó tömörített fájlt egy tetszőleges könyvtárba. Ehhez a szkripthez kicsomagolhatod akár egy külön könyvtárba is a Letöltések könyvtáron belül. Ugyanakkor, azért hogy bárki, aki a gépemet használja, hozzáférjen a VPN–hez (még ha én vagyok az egyetlen „élő” felhasználó bármelyik gépemen), azokat a fájlokat az /etc/openvpn könyvtárba szoktam másolni. Te is megteheted, ha akarod. De a szkript futtatása szempontjából teljesen megfelel, ha a Letöltések könyvtáradban egy külön könyvtárba csomagolod ki.

A következőkben láthatod a szkriptet, amivel **sixte** állt elő a fórumon. A sorszámozást én adtam hozzá az olvashatóság javítása érdekében. Azokat nem kell begépelned.

Mindazonáltal, ***gondoskodj*** arról, hogy begépelés és mentés után a szkript (és a többi) végrehajthatóvá váljon a futtatás előtt,.

```
1. #!/bin/bash
2. USERNAME="xxxxxxxxxxxxxxxx"
3. PASS="yyyyyyyyyyyyyyyyyy"
4.
5.
6. for f in *.ovpn
7. do
8.    name=`basename -s .ovpn $f`;
9.    nmcli connection import type openvpn file $f
10.    nmcli connection modify "${name}" +vpn.data connection-type=password-tls
11.    nmcli connection modify "${name}" +vpn.data username="${USERNAME}"
12.    nmcli connection modify "${name}" +vpn.data password-flags="0"
13.    nmcli connection modify "${name}" +vpn.secrets password="${PASS}"
14. done
```

Nyiss terminált és addig váltogass könyvtárat, amíg be nem léptél abba a könyvtárba, ahova a VPN-fájlokat mentetted. Az „**xxxxxxxxxxxxxxxx**” helyére a VPN-szolgáltatód nevét írd be és az „**yyyyyyyyyyyyyyyyyy**”-t pedig cseréld le a VPN-fiókodban használt jelszóra.

Kevesebb, mint egy perc alatt a VPN-ed összes OpenVPN-szerverét importálja és használatra beállítja a NetworkManager-edben. A gépen a szkriptnek én a **vpn-import.sh** nevet adtam.

Nos igen, a szkript ezen változatában a felhasználóneved és a jelszavad „jól olvasható”. Ha te vagy a számítógéped egyetlen felhasználója, ez nem jelenthet túl nagy biztonsági kockázatot.

Tehát, **tbs** előállt a szkript egy másik változatával (l. a következő hasábban).


```
1.  #!/bin/bash
2.  #
3.  read -p "User name: " vname
4.  read -p "Password : " vpass
5.  vpndir=<path-to-.ovpn-files>
6.  cd $vpndir
7.
8.  for vpnfile in *.ovpn; do
9.  nmcli connection import type openvpn file $vpnfile
10.  vpnname=$(basename -s .ovpn $vpnfile)
11.  nmcli connection modify "${vpnname}" +vpn.data connection-type=password-tls
12.  nmcli connection modify "${vpnname}" +vpn.data password-flags="0"
13.  nmcli connection modify "${vpnname}" +vpn.data username="${vname}"
14.  nmcli connection modify "${vpnname}" +vpn.secrets password="${vpass}"
15.  done
```

A szkript ezen verziója kéri a felhasználótól a felhasználónév és a jelszó begépelését. Valószínűleg ez a legbiztonságosabb módja a név és a jelszó megadásának. Nem készül feljegyzés ezekről a létfontosságú adatokról. Minden a felhasználó fejében van. Biztosíthatlak, hogy csak egyszer kell megadnod ezt az információt, amikor először importálod az OpenVPN-fájlokat a NetworkManager-be. Nekem nagyon nehéz fejben tartanom a felhasználónevemet. Betűk és számok titkosító keveréke, amit nem használok olyan gyakran, hogy megjegyezzem.

Tehát eszembe jutott valami, amit a régi-többé-nem-releváns VPN-szkriptemben használtam a net_applet-tel a VPN-emhez kapcsolódásra. Annál, az összes belépési adatomat egy egyszerű szövegfájlban tároltam, amit a /etc/openvpn-be raktam, abba a könyvtárba, ahol a többi OpenVPN fájlt is tároltam a gépen. A bejelentkezési adatok formátuma nem lehet ennél egyszerűbb. Az első sorban a felhasználóneved, a második sorban a jelszavad van. Ennyi az egész. Két sor. Az egész valahogy így néz ki: (Következő oldalon balra fent)

felhasználó_neved
jelszavad

Most ezt a „login” fájlt nem rakhatom be a /etc/openvpn könyvtárba, a hozzáférés miatt. Normál felhasználóként akarom és kell futtatni,de a normál felhasználó nem fér hozzá rendszergazda fájljaihoz. Tehát a következő legjobb dolgot tettem, eldugtam valahol máshol a fájlt a /home-omban. Hogy elrejtsem a kíváncsi szemek elől, rejtett fájl csináltam belőle a név elé egy pont illesztésével. Nincs alkönyvtárban. A /home alatt van. Mivel rejtett, senki sem látja. Természetesen bárki, aki belenéz a szkriptbe, pontosan tudni fogja, hol keresse a VPN-adataimat tartalmazó fájlt, de egy kicsit meg kell dolgoznia az eléréséhez. Más szóval, ne hidd, hogy a fedett, vagy rejtett fájlok „biztosak”

A szkript legfrissebb verziója ez:

```
#!/bin/bash

# Your login information is stored in a plain text file
# with your username on the first line, and your
# password on the second line

IFS=$'\n' read -d '' -r -a data < ~/.login
USERNAME="${data[0]}"
PASS="${data[1]}"

 for f in *.ovpn
 do
    name=`basename -s .ovpn $f`;
    nmcli connection import type openvpn file $f
    nmcli connection modify "${name}" +vpn.data connection-type=password-tls
    nmcli connection modify "${name}" +vpn.data username="${USERNAME}"
    nmcli connection modify "${name}" +vpn.data password-flags="0"
    nmcli connection modify "${name}" +vpn.secrets password="${PASS}"
done
```

A szkript ezen verziója a bejelentkezési adataimat abból az egyetlen fájlból olvassa ki a /home-ban, amire az első IFS kezdődő sor mutat. A fájl a **~/.login**. Vedd észre a „pontot” a „login” név előtt. A fájl ettől lesz rejtett. Most, te bármilyen nevet adhatsz neki és akárhol tárolhatod (a /home könyvtáradban), amíg a parancs a megfelelő fájlra mutat (hívhatod akár bevásárló listának is, ha akarod). A read parancs készít egy tömböt a login fájl soraiból és azokat a USERNAME és PASS szövegláncokban helyezi el, amiket a nmcli parancs használ fel.

A három szkript közötti EGYETLEN különbség a belépési adatoknál van. sixte szkriptje nyíltan tartalmazza azokat. tbs szkriptje kéri a felhasználót, hogy gépelje be azokat, egyenként. Az enyémnél a belépési adatok (felhasználónév és jelszó) egy külön fájlban vannak és onnan olvassa a szkript. Ha aggódsz a biztonság miatt, akkor az első valószínűleg a legkevésbé az, a második a legbiztonságosabb és a harmadik valahol a kettő között van. Természetesen, hogy az információk nyilvánosságra kerüljenek, felhasználónak kell a gépnél ülnie (hacsak véletlenül nem osztod meg a bejelentkezési adatokat tartalmazó szkriptet valakivel), és akkor valószínűleg sokkal nagyobb dolgok miatt aggódhatsz.

Hátrányok

Ha vagy olyan szerencsétlen, hogy a szkriptet kétszer, vagy többször kell lefuttatnod a gépen, akkor valószínűleg a VPN-szolgáltatód szerverei között ismétlődő bejegyzéseket kapsz.

Akár be is léphetsz és egyenként törölheted kézzel az ismétlődő bejegyzéseket, ami noha végrehajtható, némi időbe kerülhet. Ha ezt az utat akarod választani, a NetworkManager a VPN-szerverek beállításait tartalmazó fájlokat a /etc/NetworkManager/system-connections alatt tárolja.

Valószínűleg az a legegyszerűbb, ha kiüríted a könyvtárat az összes tartalom törlésével és újra lefuttatod a importáló szkriptet.

Szerencsére tbs előállt egy másik szkripttel is, ami nagyban megkönnyíti a feladatot. Ő **nm-vpn-delete**-nek nevezte el. Íme itt van:

```
#!/bin/bash

# Shut down active VPN connection
vpnactive=$(nmcli con show --active | grep vpn | cut -d' ' -f1)
nmcli con down $vpncon

# Delete existing VPN connections
vpnlist=$(nmcli con show | grep vpn | cut -d' ' -f1)
for vpncon in $vpnlist; do
    nmcli con down $vpncon
    nmcli con delete $vpncon
done
```

Csak annyit kell tenned, hogy futtatod a szkriptet a megadott VPN-kapcsolatok törlésére és újra futtatod az első szkriptet (vagy a három verzió bármelyikét) a VPN-szerverek újbóli beállítására, hogy választhass közülük.

Összegzés

Kétségtelen, a NetworkManager hozzáadása a PCLinuxOS-hez HATALMAS előrelépés volt a hálózatok kezelését tekintve. És ami a VPN-kapcsolatok beállítását illeti, a NetworkManager fényévekkel jár a „régi” PCC-s eljárás előtt. Még kézzel beállítva is végtelenül könnyebb, mint a régi eljárással volt, amiről szinte senki sem tudta, hogyan kell használni. Ezek a szkriptek a VPN-kapcsolatok fölött teljes kontrollt biztosítanak és annyira egyszerűen és nagyon gyorsan teszik.