



Instal·lació Openconnect per connectar amb el VPN GLobalProtect - Ubuntu

En aquest document trobareu:

[Introducció](#)

[Instal·lació Openconnect des de repositori Git](#)

[Connectar amb openconnect a VPN UB GlobalProtect](#)

Introducció

Els repositoris oficials d'Ubuntu versió 18.04 LTS inclouen la versió d'openconnect 7.08. Aquesta versió encara no inclou el suport a les vpn's de GlobalProtect. El suport de GlobalProtect està inclòs a partir de la versió 8.xx.

Per tant, primer cal mirar quina versió de openconnect tenim disponible en el repositori:

> `sudo apt update` (per actualitzar les fonts repositoris)

> `apt show openconnect`

```
usuari@ubuntutest:~$ apt show openconnect
Package: openconnect
Version: 7.08-3ubuntu0.18.04.1
Priority: optional
Section: universe/net
Origin: Ubuntu
Maintainer: Ubuntu Developers <ubuntu-devel-discuss@lists.ubuntu.com>
Original-Maintainer: Mike Miller <mtmiller@debian.org>
Bugs: https://bugs.launchpad.net/ubuntu/+filebug
Installed-Size: 2.338 kB
Depends: vpnc-scripts, libc6 (>= 2.14), libgnutls30 (>= 3.5.0), libopenconnect5 (>= 7.08), libproxy1v5 (>= 0.4.14), libxml2 (>= 2.7.4)
Homepage: http://www.infradead.org/openconnect/
Download-Size: 392 kB
APT-Sources: http://es.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages
Description: open client for Cisco AnyConnect VPN
 OpenConnect is a client for Cisco's AnyConnect SSL VPN, which is supported by the ASA5500 Series, by IOS 12.4(9)T or later on Cisco SR500, 870, 880, 1800, 2800, 3800, 7200 Series and Cisco 7301 Routers, and probably others. OpenConnect also supports the Pulse Connect Secure SSL VPN, formerly known as the Juniper SSL VPN or the Junos Pulse VPN.

N: Hay 1 registro adicional. Utilice la opción «-a» para verlo.
```

Si tinguéssim disponible una versió igual o superior a OpenConnect v8.xx ja tindrà el suport de GlobalProtect i podem instal·lar-la fent:

- o `sudo apt install openconnect`

Si tenim una inferior, haurem de instal·lar una versió més nova tal com descrivim al següent pas

Instal·lació Openconnect última 'release'

- Descarreguem les fonts des de la "The latest release" de la [web del desenvolupador](#).



Download

Released versions of OpenConnect are available from the FTP site:

- <ftp://ftp.infradead.org/pub/openconnect/>

Release tarballs (since 3.13) are signed with the PGP key with fingerprint [BE07 D9FD 5480 9AB2 C4B0 FF5F 6376 2CDA 67E2 F359](#).

The latest release is [OpenConnect v8.05 \(PGP signature\)](#) released on 2019-09-12 with the following changelog:

- Fix GlobalProtect ESP stall ([#55](#)).
- Fix HTTP chunked encoding buffer overflow (CVE-2019-16239).

For older releases and change logs, see the [changelog page](#).

- Instal·lem les dependències:
 - `> sudo apt-get install build-essential gettext autoconf automake libproxy-dev libxml2-dev libtool vpnc-scripts pkg-config libssl-dev`
- Descomprimim les fonts descarregades.
 - `> tar -zxvf openconnect-8.xx.tar.gz`
- Entrem dins del directori descomprimit i iniciem la compilació:
 - `> cd openconnect-8.xx`
 - `> ./configure`

```
Archivo Editar Ver Buscar Terminal Ayuda
checking if groff can create UTF-8 XHTML... no. Not building HTML pages
checking for CWRAP... no
checking for nuttcp... no
checking that generated files are newer than configure... done
configure: creating ./config.status
config.status: creating Makefile
config.status: creating openconnect.pc
config.status: creating po/Makefile
config.status: creating www/Makefile
config.status: creating libopenconnect.map
config.status: creating openconnect.8
config.status: creating www/styles/Makefile
config.status: creating www/inc/Makefile
config.status: creating www/images/Makefile
config.status: creating tests/Makefile
config.status: creating tests/softsm2.conf
config.status: creating tests/configs/test-user-cert.config
config.status: creating tests/configs/test-user-pass.config
config.status: creating config.h
config.status: executing depfiles commands
config.status: executing libtool commands
BUILD OPTIONS:
  SSL library:          OpenSSL
  PKCS#11 support:     no
  DTLS support:        yes
  ESP support:         yes
  libproxy support:    yes
  RSA SecurID support: no
  PSKC OATH file support: no
  GSSAPI support:      no
  Yubikey support:     no
  LZ4 compression:    no
  Java bindings:       no
  Build docs:          no
  Unit tests:          no
  Net namespace tests: no
configure: WARNING:
***
*** Be sure to run "make check" to verify OpenSSL DTLS support
***
```

- `> make`

```
Archivo Editar Ver Buscar Terminal Ayuda
rm -f && /usr/bin/msgfmt -o bs.mo bs.po
rm -f && /usr/bin/msgfmt -o ca.mo ca.po
rm -f && /usr/bin/msgfmt -o cs.mo cs.po
rm -f && /usr/bin/msgfmt -o da.mo da.po
rm -f && /usr/bin/msgfmt -o de.mo de.po
rm -f && /usr/bin/msgfmt -o el.mo el.po
rm -f && /usr/bin/msgfmt -o en_GB.mo en_GB.po
rm -f && /usr/bin/msgfmt -o en_US.mo en_US.po
rm -f && /usr/bin/msgfmt -o es.mo es.po
rm -f && /usr/bin/msgfmt -o eu.mo eu.po
rm -f && /usr/bin/msgfmt -o fi.mo fi.po
rm -f && /usr/bin/msgfmt -o fr.mo fr.po
rm -f && /usr/bin/msgfmt -o gl.mo gl.po
rm -f && /usr/bin/msgfmt -o hu.mo hu.po
rm -f && /usr/bin/msgfmt -o id.mo id.po
rm -f && /usr/bin/msgfmt -o it.mo it.po
rm -f && /usr/bin/msgfmt -o lt.mo lt.po
rm -f && /usr/bin/msgfmt -o nl.mo nl.po
rm -f && /usr/bin/msgfmt -o pa.mo pa.po
rm -f && /usr/bin/msgfmt -o pl.mo pl.po
rm -f && /usr/bin/msgfmt -o pt_BR.mo pt_BR.po
rm -f && /usr/bin/msgfmt -o pt.mo pt.po
rm -f && /usr/bin/msgfmt -o sk.mo sk.po
rm -f && /usr/bin/msgfmt -o sl.mo sl.po
rm -f && /usr/bin/msgfmt -o sr@latin.mo sr@latin.po
rm -f && /usr/bin/msgfmt -o sr.mo sr.po
rm -f && /usr/bin/msgfmt -o sv.mo sv.po
rm -f && /usr/bin/msgfmt -o tg.mo tg.po
rm -f && /usr/bin/msgfmt -o tr.mo tr.po
rm -f && /usr/bin/msgfmt -o ug.mo ug.po
rm -f && /usr/bin/msgfmt -o uk.mo uk.po
rm -f && /usr/bin/msgfmt -o zh_CN.mo zh_CN.po
rm -f && /usr/bin/msgfmt -o zh_TW.mo zh_TW.po
make[2]: se sale del directorio '/home/usuari/openconnect/po'
make[2]: se entra en el directorio '/home/usuari/openconnect'
CC      openconnect-xml.o
CC      openconnect-main.o
CCLD    openconnect
make[2]: se sale del directorio '/home/usuari/openconnect'
make[1]: se sale del directorio '/home/usuari/openconnect'
usuari@buntutest:~/openconnect$
```

- o > *sudo make install && sudo ldconfig*

```
Archivo Editar Ver Buscar Terminal Ayuda
openconnect.so; }; })
libtool: install: /usr/bin/install -c .libs/libopenconnect.lai /usr/local/lib/li
bopenconnect.la
libtool: finish: PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/
bin:/snap/bin:/sbin" ldconfig -n /usr/local/lib
-----
Libraries have been installed in:
  /usr/local/lib

If you ever happen to want to link against installed libraries
in a given directory, LIBDIR, you must either use libtool, and
specify the full pathname of the library, or use the '-LLIBDIR'
flag during linking and do at least one of the following:
- add LIBDIR to the 'LD_LIBRARY_PATH' environment variable
during execution
- add LIBDIR to the 'LD_RUN_PATH' environment variable
during linking
- use the '-Wl,-rpath -Wl,LIBDIR' linker flag
- have your system administrator add LIBDIR to '/etc/ld.so.conf'

See any operating system documentation about shared libraries for
more information, such as the ld(1) and ld.so(8) manual pages.
-----
/bin/mkdir -p '/usr/local/libexec/openconnect'
/usr/bin/install -c trojans/csd-post.sh trojans/csd-wrapper.sh trojans/tncc-wra
pper.py trojans/hipreport.sh trojans/hipreport-android.sh '/usr/local/libexec/op
enconnect'
/bin/mkdir -p '/usr/local/sbin'
/bin/bash ./libtool --mode=install /usr/bin/install -c openconnect '/usr/loc
al/sbin'
libtool: install: /usr/bin/install -c .libs/openconnect /usr/local/sbin/openconn
ect
/bin/mkdir -p '/usr/local/include'
/usr/bin/install -c -m 644 openconnect.h '/usr/local/include'
/bin/mkdir -p '/usr/local/share/man/man8'
/usr/bin/install -c -m 644 openconnect.8 '/usr/local/share/man/man8'
/bin/mkdir -p '/usr/local/lib/pkgconfig'
/usr/bin/install -c -m 644 openconnect.pc '/usr/local/lib/pkgconfig'
make[2]: se sale del directorio '/home/usuari/openconnect'
make[1]: se sale del directorio '/home/usuari/openconnect'
usuari@buntutest:~/openconnect$
```

- Ara comprovarem que tot ha anat bé i que tenim la versió d'openconnect correcta:



- > `openconnect --version` (comprovem que tenim una versió OpenConnect v8.xx)

```
Archivo Editar Ver Buscar Terminal Ayuda
usuari@ubuntutest:~/openconnect$ openconnect --version
OpenConnect versión v8.05-95-gbc3f3891
Usando OpenSSL. Características disponibles: TPM (Motor OpenSSL no disponible),
HOTP software token, TOTP software token, DTLS, ESP
Protocolos admitidos: anyconnect (predeterminado), nc, gp, pulse
```

- Amb això ja haurem acabat la instal·lació

Connectar amb openconnect a VPN UB GlobalProtect

El openconnect te bastants problemes amb els networks manager gui i de moment la connexió més senzilla és per consola.

Per connectar-se a una VPN haurem d'executar en una consola la següent comanda:

- > `sudo openconnect --protocol=gp vpnub.ub.edu`

```
Archivo Editar Ver Buscar Terminal Ayuda
usuari@ubuntutest:~$ sudo openconnect --protocol=gp vpnub.ub.edu
[sudo] contraseña para usuari:
POST https://vpnub.ub.edu/global-protect/prelogin.esp?tmp=tmp&clientVer=4100&cli
entos=Linux
Conectado a 161.116.221.98:443
Negociación SSL con «vpnub.ub.edu»
Conectó a HTTPS en vpnub.ub.edu
Enter login credentials
Username usuari
Password:
POST https://vpnub.ub.edu/global-protect/getconfig.esp
1 servidores de puerta de enlace disponibles:
  vpnub.ub.edu (vpnub.ub.edu)
Seleccione una puerta de enlace GlobalProtect.
PUERTA DE ENLACE: [vpnub.ub.edu]:vpnub.ub.edu
POST https://vpnub.ub.edu/ssl-vpn/login.esp
El inicio de sesión de GlobalProtect devolvió authentication-source=Authe_LDAP-G
EN
POST https://vpnub.ub.edu/ssl-vpn/getconfig.esp
La sesión expirará después de 43200 minutos.
El tiempo de espera del túnel (intervalo rekey) es de 180 minutos.
El tiempo de espera de inactividad es de 180 minutos.
No se recibió MTU. Se calculó 1422 para ESP tunnel
POST https://vpnub.ub.edu/ssl-vpn/hipreportcheck.esp
Conectado como 10.100.247.183, usando SSL, con ESP en progreso
Sesión ESP establecida con el servidor
Túnel ESP conectado; saliendo del bucle principal HTTPS.
```

Al fer un sudo, les primeres credencials són les de l'usuari local de la màquina des d'on ens connectem. Les segones credencials són les de VPN UB:

- Usuari : mail sense @ub.edu
- Password : password del mail UB

Si haguéssim de fer la connexió al **VPN Tècnic** seria el mateix però canviant el portal de connexió:

- `sudo openconnect --protocol=gp vpntecnic.ub.edu`

Hem de deixar la consola oberta per mantenir la connexió. Per tancar la connexió només caldrà prémer "ctrl+c" o tancar la consola.



- *També és pot obrir més d'una consola amb un vpn en cadascuna, per exemple VPN tècnic i VPNUB, en aquest cas el ordre d'obertura serà important per com quedaran definides les rutes.*
- *A diferencia del client VPNC, openconnect si que ens agafa correctament els DNS i les rutes del VPN.*