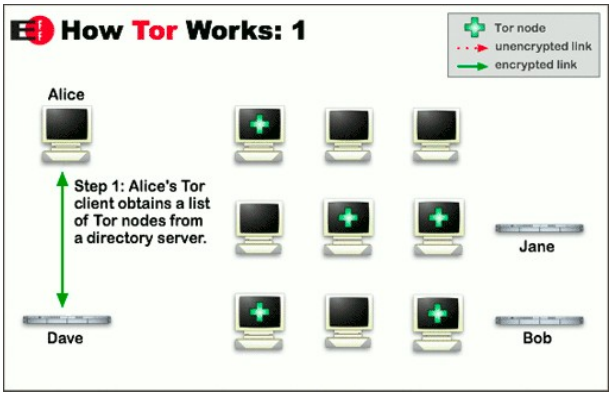
Tor

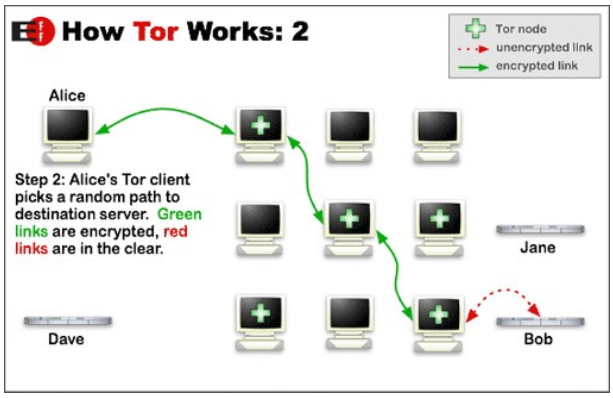
Introducció

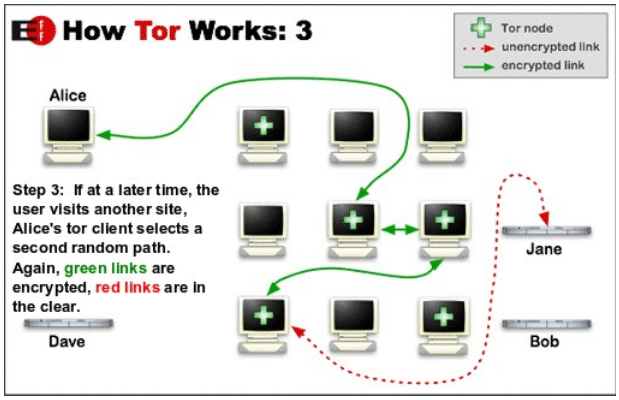
Tor (https://www.torproject.org) és una "intraxarxa" dins d'Internet que implementa una tècnica d'enrutament, diferent de les tècniques estàndards, anomenada Onion Routing, gràcies a la qual es garanteix l'**anonimat** dels extrems i la **privacitat** de les dades transferides. Tor és una xarxa operada per voluntaris que emmascara qui ets i des d'on estàs connectat. També et protegeix dins la mateixa xarxa Tor, ja que pots estar segur que romandràs anònim enfront d'altres usuaris de Tor també. Per tant, és una xarxa molt utilitzada per activistes i periodistes en països on no es respecten els drets humans (a més de la gent en general conscienciada d´aquests temes relacionats amb la cibervigilància).

L'encaminament tradicional que fem servir per connectar-nos a servidors a Internet és directe: del nostre ordinador al nostre router, d'aquí als encaminadors del nostre ISP (proveïdor d'Internet) i després directes al servidor de destinació triat. No obstant, si algú intercepta els paquets de dades en algun punt intermedi sabrà perfectament d'on vénen i on van. Fins i tot encara que es xifrin les dades de cada paquet (per exemple, visitant una pàgina HTTPS) les capçaleres d'aquest no es xifren, i els camps de remitent i destinatari (entre d'altres) segueixen sent visibles. Aquí és on entra l'"Onion Routing". Expliquem-ho:

1) L'ordinador A, que vol enviar el missatge a B, calcula una ruta més o menys aleatòria a la destinació passant per diversos nodes intermedis (també anomenats "relays"), tal com mostren les il·lustracions següents:







NOTA: L'última connexió es xifraria si Alice visités un lloc web HTTPS: tingueu en compte que Tor no pot xifrar el trànsit després de sortir de la xarxa Tor.

2) Tal com mostren les il·lustracions anteriors, la comunicació entre l'ordinador A i l'últim node de la ruta (el "node de sortida de la xarxa Tor") és completament xifrada. Això s'aconsegueix, un cop decidida la ruta dels nodes a recórrer, obtenint d'un ordinador especial (el "directori de nodes") les claus públiques de tots aquests nodes per tal de, usant xifrat asimètric, xifrar per capes (com una ceba) el missatge a enviar: primer es xifrarà el missatge amb la clau pública del node de sortida, per a què només ell el pugui desxifrar (a més del missatge, també inclourà, igualment xifrades, les

instruccions per arribar a la destinació, l'ordinador B). Tot aquest paquet, juntament amb les instruccions per arribar al node de sortida, es xifrarà de nou amb la clau pública del node intermig de la ruta per a què només ell el pugui desxifrar. I aquest procés es torna a repetir de nou (és a dir, el xifrat de tot l'anterior més les instruccions per arribar al node intermig, ara amb la clau pública del primer node de la ruta, el "node d'entrada") per a què només ell el pugui dexifrar.

3) Amb el pas anterior ja tindrem el paquet de dades llest, així que tocarà enviar-lo: l'ordinador A connectarà amb el node d'entrada de la ruta, i li enviarà el paquet; aquest node ho desxifrarà i seguirà les instruccions que ha desxifrat per enviar la resta del paquet al node intermig de la ruta; aquest el desxifrarà de nou i tornarà a seguir les instruccions recentment dexifrades per enviar la resta del paquet al node final, el qual el desxifrarà per tal de saber on enviar finalment el missatge. Es pot veure aquest procés el següent diagrama:

El més important que cal tenir en compte respecte el funcionament anterior és que només el node d'entrada coneix l'adreça IP de la màquina remitent i només el node de sortida coneix l'adreça IP (o nom DNS depén, en parlarem més endavant) de la máquina destí final (i viceversa, que també és important!: la màquina destí final només coneix la IP -o nom DNS- del node de sortida, res més). Cap dels nodes sap ni tan sols quina posició ocupen en la ruta, i molt menys coneixen el contingut del missatge. D'aquesta manera, tot i que s'interceptin les comunicacions entre dos nodes, és impossible saber quines dades transmet, a on van o d'on vénen. Fins i tot encara hi hagués un node infiltrat, un talp a la xarxa, no tindria res a fer amb els missatges que rep. D'altra banda, però, també cal tenir en compte que el node final de sortida pot llegir el missatge original, així que també cal xifrar el missatge original (tal com s'indica a la "nota" anterior).

Per al lector interessat, en https://securityinabox.org/en/guide/anonymity-and-circumvention es desenvolupen diversos conceptes relacionats amb l´**elusió de la censura** a Internet, un aspecte on la xarxa Tor també pot ajudar ja que s´hi accedeix a ella a través del node de sortida, el qual sol estar en un altre lloc geogràfic (i sota l´accés d´un altre ISP) que la nostra màquina "Alice". En aquest sentit, cal dir que la xarxa Tor és molt difícil de tombar: en estar els nodes distribuïts, caldria tombar tots i cada un d'ells per poder aturar les comunicacions.

NOTA: Cal tenir en compte, però, que hi ha un compromís entre l'anonimat i la velocitat: fer saltar el trànsit a través de diversos servidors de diverses parts del món quasi sempre serà més lent que una connexió directa a Internet.

El protocol Tor està detallat aquí: https://gitlab.torproject.org/tpo/core/torspec . Altres llocs web molt complets que expliquen els detalls interns de la xarxa Tor són https://github.com/tfukui95/tor-experiment i https://witestlab.poly.edu/blog/anonymous-routing-of-network-traffic-using-tor

NOTA: Per poder establir els circuits virtuals a la xarxa Tor, cal disposar d'una llista que detalli l'estat (direcció, ample de banda, etc.) dels diferents nodes disponibles. Aquest document és anomenat "Consens". A cada client Tor es troba embegut la informació de 10 nodes fiables que són la base de la xarxa Tor, amb IPs conegudes. La missió d'aquests nodes és d'actualitzar i mantenir el Consens, per la qual cosa són anomenats autoritats de directori (Directory authorities). El procés es resumeix com:

Cada autoritat genera una llista de nodes.
Cada autoritat calcula els estats dels nodes, l'ample de banda que disposen i diferents flags.
Un cop calculats aquests paràmetres, l'autoritat puja aquesta informació (vot) per a la resta d'autoritats
Cada autoritat descarrega els vots de la resta, combina la informació, recalcula i signa el resultat.
Es puja novament aquesta informació signada a la resta d'autoritats.
Si hi ha majoria, es valida el consens i és publicat per cada autoritat.
El procés anterior i l'actualització del consens es duen a terme cada una hora.
Una lectura interessant sobre això és https://blog.torproject.org/lifecycle-new-relay

Tor Browser

Per als que necessiten ocasionalment d'anonimat i privacitat quan naveguen per llocs web, el "Tor Browser" ofereix una manera ràpida i fàcil d'utilitzar la xarxa Tor. Aquest navegador (https://www.torproject.org/projects/torbrowser.html.en) no és més que una versió modificada de el navegador Firefox (per tant, lliure) que és capaç de:

* Amagar la IP dels usuaris
* Eliminar qualsevol tipus de sistema de seguiment en línia
* Habilitar l'accés a pàgines web censurades (o només existents dins de la pròpia xarxa Tor)

NOTA: Com ja sabem, la xarxa Tor està formada per milers de servidors gestionats per voluntaris de tot el món. Cada vegada que el navegador Tor fa una nova connexió, selecciona tres d'aquests "relays" Tor i es connecta a Internet a través d'ells. Quan utilitzeu el navegador Tor, el vostre trànsit d'Internet semblarà provenir d'una adreça IP diferent (sovint en un país diferent); aquesta IP diferent pertany a l'últim node Tor "fronterer", responsable de "sortir" de Tor a Internet. Com a resultat, el navegador Tor amaga la vostra adreça IP dels llocs web als quals accediu, mentre que també oculta els llocs web als quals accediu de tercers que podrien intentar controlar el vostre trànsit. També garanteix que cap relay Tor no pugui esbrinar tant la vostra ubicació a Internet com els llocs web que visiteu (tot i que alguns d'ells coneixeran l'un o l'altre).

El "Tor Browser" no requereix instal·lació: només cal descomprimir el paquet "tar.xz" obtingut de la descàrrega i fer doble clic sobre l´executable anomenat *start-tor-browser*, ja està. Per tant, és una solució portable.

La primera vegada que inicieu el navegador Tor, podrem optar per connectar-nos directament a la xarxa Tor o bé establir una configuració particular de xarxa pel navegador Tor. Seleccionarem la primera opció si el nostre accés a Internet no és restringit i l'ús de Tor no està bloquejat, prohibit ni controlat on estem ubicats; seleccionarem la segona opció, en canvi, si el nostre accés a Internet està restringit o si l'ús de Tor està bloquejat, prohibit o controlat on estem ubicats (els ISPs poden, tot i no saber-ne el contingut, reconéixer que un tràfic és de tipus Tor i, per tant, censurar-lo).

Concretament, en el cas d'usar el navegador Tor des d'una ubicació on la xarxa Tor està bloquejada, en anar a la finestra de configuració (*about:preferences#connection*), caldrà indicar un node de tipus pont. Els **ponts** són relays Tor d'entrada que no apareixen al directori públic, així que són més difícils de bloquejar.

Per passar desapercebuts, els ponts Tor poden utilitzar opcionalment uns protocols de tunelatge anomenats "**transports connectables**" que intenten dissimular el trànsit entre el client Tor i el pont triat, fent-lo passar aparentment per altre tipus de trànsit de nivell 7 innocent per tal d'evitar així que els tallafocs i IDS d'Internet els identifiquin i bloquegin. El "transport connectable" per defecte és l'anomenat "*obfs4*", el qual camufla el tràfic Tor fent-lo sembla tràfic aleatori, però n'hi d'altres que es poden triar, com els anomenats "*meek*" (que fan semblar que el tràfic es correspon a tràfic de navegació estàndard a llocs Azure legítims) o "*snowflake*" (que, mitjançant nodes intermediaris operats per voluntaris, fan sembla que el tràfic es correspon a una video-trucada).

NOTA: Hi ha vàries maneres d'usar els ponts; qualsevol d'elles es pot triar des de l'apartat "Ponts" de la finestra de configuració *about:preferences#connection* del navegador Tor. Concretament:

*Es poden usar els ponts proporcionats per defecte pel propi navegador Tor, podent-ne escollir el "transport connectable" desitjat
*Es poden sol·licitar a través de l'assistent que proporciona el propi navegador Tor.
*Es poden indicar manualment després d'haver-ne obtingut les seves adreces a https://bridges.torproject.org/options (aquesta web fa el mateix procediment que el punt anterior)

Si, no obstant, no es pot accedir al lloc web del Projecte Tor, es poden sol·licitar adreces de pont personalitzades enviant un correu electrònic a *bridges@torproject.org* mitjançant un compte de Riseup, Gmail o Yahoo i incloent la frase, "Get bridges" al cos del missatge. O també contactant al bot de Telegram #GetBridgesBot i escrivint "/bridges". Llegiu, en tot cas, la documentació present a https://bridges.torproject.org

Quan ja tingueu el navegador Tor en funcionament, podeu verificar que utilitzeu la xarxa Tor al lloc https://check.torproject.org. A més, podeu anar a qualsevol lloc informant sobre la vostra IP pública (per exemple, https://www.iplocation.net o https://www.ip2location.com entre molts altres) per veure si està falsificada

Si vols, pots canviar el node intermig i el de sortida del teu circuit (el d'entrada no) pulsant CTRL+SHIFT+L (o anant a l'opció corresponent del menú estàndard del Firefox); d'aquesta manera veuràs que sembla que provens d'una nova adreça IP. D'altra banda, també pots anar més enllà i crear-te una "nova identitat" pulsant CTRL+SHIFT+U (o mitjançant l'opció corresponent del menú); això significa que a més de triar de nou (a l'atzar) un nou conjunt de nodes Tor (aquest cop, tots tres canviaran), el navegador Tor esborrarà llavors el teu historial de navegació i les galetes i després reiniciarà; un cop reiniciat, pots confirmar igualment que sembla que provens d'una nova adreça IP.

NOTA: Teniu una llista dels dubtes més comuns sobre la xarxa Tor i el Tor Browser en particular, amb la seva resposta, a https://www.infosecmatter.com/the-onion-router-and-privacy

Software complementari

El projecte Tor desenvolupa altres programes complementaris del Tor Browser, els quals es llisten en https://www.torproject.org/projects/projects.html.en . D´entre ells, podem destacar els següents:

https://guardianproject.info/apps/orbot : Proxy Tor per les aplicacions executades en un sistema Android. Un projecte relacionat és Orlib; una llibreria per ser usada en el desenvolupament de qualsevol aplicació Android per tal d´encaminar el seu tràfic d´Internet a través de Orbot/Tor. En qualsevol cas, també hi ha disponible el propi "Tor Browser" per plataformes Android.

https://nyx.torproject.org : Nyx és un programa que monitoritza l´estat del nostre proxy Tor en el terminal (útil per si el nostre proxy és remot i ens volem connectar des de SSH). Proporciona informació en temps real sobre l´ús de recursos de la màquina, connexions, etc

https://metrics.torproject.org : Portal on es mostren dades analítiques de la xarxa Tor, incloent gràfics sobre l'amplada de banda disponible, la base d'usuaris estimada, el funcionament actual dels bridges i relays, etc. Aquest és un recurs fantàstic per a investigadors interessats en estadístiques detallades sobre Tor.

https://stem.torproject.org : Llibreria Python per ser usada en aplicacions i scripts propis que volem que interaccionin amb Tor.

https://tpo.pages.torproject.net/core/arti : (Re)Implementació del proxy Tor en llenguatge Rust

https://ooni.org : Xarxa d'observació global que té com a objectiu recopilar dades mitjançant metodologies obertes per tal de compartir observacions sobre els diversos tipus, mètodes i quantitats de mecanismes de censura existents a Internet.

http://shadow.github.io : Simulador de xarxa que executa el programari Tor real com a "plug-in". Permet una experimentació Tor precisa, eficient, controlada i repetible. Una altra eina similar és Chutney (https://gitweb.torproject.org/chutney.git)

Configuració de l'accés a la xarxa Tor des de qualsevol programa

Si volem que més programes instal·lats en el nostre sistema puguin accedir a la xarxa Tor, més allà del "Tor Browser", haurem d'utilitzar el programari Tor pròpiament dit (https://dist.torproject.org). Després d'instal·lar aquest programari (ja sigui directament a partir del paquet homònim del dipòsit de la distribució que usem, o bé del paquet disponible en el dipòsit propi de Tor o bé a partir de la compilació del codi font descarregat de l'enllaç anterior) podrem posar en marxa un servidor proxy local mitjançant les comandes habituals: *systemctl start tor, systemctl enable tor*, etc. A partir de llavors, qualsevol programa que suporti proxies de tipus SOCKS podrà accedir a la xarxa anònima.

NOTA: El protocol SOCKS és una tecnologia genèrica de proxificació TCP de la qual el protocol TOR és un cas concret
Per tenir més informació sobre aquest protocol, podeu llegir l'article https://en.wikipedia.org/wiki/SOCKS

NOTA: Pot ser que el programa que usem inclogui la nostra IP en els dades que envia, i llavors tot l´encaminament i els xifratges que fem no ens serviran per a res. Per això en general es recomana l'ús de "Tor Browser", ja que aquests aspectes ja els té resolts, en lloc de configurar manualment el proxy Tor.
NOTA: A sistemes Ubuntu/Debian, un cop instal·lat el paquet "tor" apareixeran dos serveis, un anomenat "tor" i un altre "tor@default". En principi, aquest darrer no l'haurem de tenir en compte (tot i que és realment qui gestiona "de veritat" el servei "tor")

Concretament, el programa en qüestió s´haurà de configurar per usar un proxy de tipus SOCKS. Tor només escolta per defecte en la IP 127.0.0.1 i al port **9050** (encara que si fem servir Tor Browser aquest també pot actuar com a proxy d'altres aplicacions, escoltant en aquest cas al port 9150). Així doncs, si volem que un navegador Firefox "estàndard" pugui accedir a la xarxa Tor, simplement haurem d'anar a "Paràmetres → General → Paràmetres de xarxa -> Configuració manual del servidor intermediari" i allà seleccionar "Ordinador central SOCKS (SOCKSv5)" indicant la IP de la màquina i port d'escolta on s'està executant el software Tor.

Algunes directives importants del fitxer de configuració del proxy Tor ("**/etc/tor/torrc**") són:

**SOCKSPort {nº | ip:nº }* : Port (o combinació IP:port) per on escoltarà el proxy. Es pot indicar la IP 0.0.0.0 per referir-se a "qualsevol que tingui el sistema". Si val 0, aquesta funcionalitat estarà desactivada

**SOCKSPolicy {accept | reject } {ip | ipXarxa/mask | * }* : Regles que accepten o rebutjen connexions de clients amb una determinada IP o que provinguin d'una determinada xarxa o qualsevol. Es poden afegir les línies que es vulguin i la primera que quadri s'aplica i no es segueix llegint (a l'estil de Nftables). Per defecte s'accepten totes les connexions

**EntryNodes {fingerprintNodeoCodiPais, fingerprintNodeoCodiPais, ... }* : Llista els codis identificatius ("fingerprints") o bé codis de països sencers (que inclouran tots els nodes allà ubicats) que es volen utilitzar com a nodes d'entrada (en lloc de triar-los a l'atzar).

NOTA: Els codis de països són de tipus ISO3166 s'han d'escriure entre claus (així, *{es}*)

**ExitNodes {fingerprintNodeoCodiPais, fingerprintNodeoCodiPais, ... }* : Llista els codis identificatius ("fingerprints") o bé codis de països sencers (que inclouran tots els nodes allà ubicats) que es volen utilitzar com a nodes de sortida (en lloc de triar-los a l'atzar).

NOTA: Els codis de països són de tipus ISO3166 s'han d'escriure entre claus (així, *{es}*)

**ExcludeNodes {fingerprintNodeoCodiPais, fingerprintNodeoCodiPais, ... }* : Llista els codis identificatius ("fingerprints") o bé codis de països sencers (que inclouran tots els nodes allà ubicats) que NO es volen utilitzar per realitzar un circuit Tor (ja sigui d'entrada, intermig o de sortida). Cal notar que aquesta directiva només indica una preferència; si realment es volen evitar, caldrà afegir a més la directiva *StrictNodes 1*

**ExcludeExitNodes {fingerprintNodeoCodiPais, fingerprintNodeoCodiPais, ... }* : Llista els codis identificatius ("fingerprints") o bé codis de països sencers (que inclouran tots els nodes allà ubicats) que NO es volen utilitzar explícitament com a nodes de sortida.

**DNSPort {nº | ip:nº | auto}* : Port (el qual mitjançant la paraula "auto" pot ser triat pel propi Tor) o combinació IP:port per on es resoldran anònimament les peticions DNS rebudes. Si val 0, aquesta funcionalitat estarà desactivada (per defecte és així)

**Log nivell /ruta/arxiu* : Indica el nivell de gravetat ("info", "warn", "error",…) a partir del qual es gravaran els missatges a l'arxiu de registre la ruta del qual s'hagi indicat

NOTA: Altres directives importants (més a nivell de funcionament intern, i que en principi no tocarem) són *RunAsDaemon {0|1}* o *DataDirectory /var/lib/tor* , entre altres. Es poden consultar totes elles a *man tor*

Si volem usar una aplicació que no suporti proxificació SOCKS nativa, es poden utilizar diferents aplicacions que afegeixen aquesta funcionalitat "on-the-fly" a qualsevol aplicació, tal com *socat* (http://www.dest-unreach.org/socat), *proxychains* (https://github.com/haad/proxychains), *proxychains-ng* (https://github.com/rofl0r/proxychains-ng), *shadowsocks* (https://www.shadowsocks.org) o, més en concret per la xarxa Tor, *torsocks (*https://gitlab.torproject.org/tpo/core/torsocks) . Una llista encara més completa de proxies es troba a https://en.wikipedia.org/wiki/Comparison_of_proxifiers Per més informació podeu consultar https://gitlab.torproject.org/legacy/trac/-/wikis/doc/TorifyHOWTO

> NOTA: A vegades s'usa amb les eines anteriors un altre programa anomenat *Privoxy* (http://www.privoxy.org). Aquest és proxy HTTP, no pas TCP (és a dir, és més similar a Squid). Està especialitzat en augmentar la privacitat de la comunicació web gràcies a les capacitats de filtratge que incorpora (modificant les capçaleres HTTP, controlant l'accés, treient anuncis/ad-ware, no catxejant cap connexió, etc) però només serveix per comunicacions HTTP. Si s'enllaça Tor amb Privoxy podrem fem que llavors totes les connexions estiguin "torificades" i que les de tipus web, a més, passin per Privoxy. D´altra banda, Privoxy pot servir igualment com a intermediari amb Tor per aplicacions que no admeten ser configurades mitjançant proxies SOCKS però sí HTTP

La manera d'utilitzar el client Torsocks és simplement, des d'un terminal, escriure: *torsocks nomPrograma* Algunes directives importants del seu fitxer de configuració ("**/etc/tor/torsocks.conf**") són *TorAddress ip (*IP del proxy Tor on es connectarà -per defecte 127.0.0.1-) i *TorPort nº* (Port del proxy Tor on es connectarà -per defecte 9050-), però en té moltes més; podeu consultar la seva documentació a la pàgina del manual o bé a https://gitlab.torproject.org/tpo/core/torsocks/

En qualsevol cas, cal tenir en compte que el programa "torsificat" (ja sigui de forma nativa o bé via *torsocks*) no realitza cap petició DNS per resoldre el nom del destí sinó que la petició desitjada (ja sigui HTTP o de qualsevol altre tipus però on ha d´aparèixer el "hostname" on es vol anar) es va transmetent per tots els "relays" Tor sense cap tipus de processament fins arribar al relay de sortida, que serà qui realment faci la petició DNS per tal de saber ell (que és, en definitiva, qui fa la connexió final) on ha d´acabar arribant la petició desitjada.

Implementació de serveis ocults ("onion")

A la xarxa Tor també poden haver servidors web, SSH, de missatgeria, etc. No obstant això, aquests servidors només estan disponibles dins de la pròpia xarxa Tor, no des de fora. Per això se'ls sol cridar serveis "ocults". Com en la xarxa Tor no hi cap sistema centralitzat de noms tipus DNS, per aconseguir que un servei ocult estigui disponible per als altres nodes el que fa és crear diversos "punts d'introducció" en certs nodes de la xarxa, i notificar anònimament a una base de dades replicada (els "directory servers") quins nodes són (en forma de llista, anomenada "onion service descriptor"). Quan un client vulgui connectar-se, enviarà a un d'aquests nodes la direcció d'un determinat punt de trobada (al que està connectat) i una clau única. El punt d'introducció connectarà amb el servei ocult, que es connectarà al punt de trobada, establint així una comunicació entre el client i el servei. La forma en què estan plantejats aquests serveis ocults permet connectar-nos a servidors de correu o de xat sense saber ni tan sols la seva adreça exacta, usant intermediaris i circuits Tor anònims (de fet, és pràcticament impossible rastrejar l'enviament d'un correu: ni tan sols el propi servidor de correu sap amb quin ordinador sestava comunicant).

En tot cas, per oferir un determinat servei a la xarxa Tor només cal afegir a l'arxiu "/etc/tor/torrc" les següents línies (i reiniciar el servei) ...:

HiddenServiceDir /var/lib/tor/unserveiocult
HiddenServicePort 80 127.0.0.1:80

... on en aquest exemple concret estem redireccionant el port 80 de la nostra màquina (que és on els clients de la xarxa Tor aniran a connectar-se a aquest servidor) a la IP loopback i port 80 de la mateixa (que és on se suposa que està escoltant realment el nostre servidor). La primera línia indica la carpeta (que es crearà automàticament en reiniciar el servidor intermediari Tor, amb permisos de lectura i escriptura per a l'usuari que executi aquest servei) on s'emmagatzemaran dos fitxers (creats també automàticament en iniciar el proxy Tor): una clau privada (que identificarà el servei en qüestió) i un fitxer anomenat "**hostname**" el valor del contingut del qual (consistent en un conjunt de caràcters inintel·ligibles -en realitat, un resum de la clau

pública associada al servei- finalitzats pel sufix ".onion") serà el nom "de domini" que haurem d'utilitzar en el client adequat (navegador, client SSH, missatgeria, etc.) per a accedir, dins de la xarxa Tor, a aquest servidor. La clau privada permet que aquest nom sigui sempre igual; de fet, simplement copiant el contingut de la carpeta "/var/lib/tor/hidden_service" a un altre ordinador (i tenint el fitxer ".torrc" amb la mateixa configuració també) es pot continuar oferint el mateix servei amb el mateix nom com si no hagués passat res. En tot cas, per a cada servei que vulguem posar en marxa a la xarxa Tor haurem d'indicar les dues línies anteriors en aquest ordre: primer *HiddenServiceDir* i després *HiddenServicePort*

> NOTA: El procés tècnic per generar adreces "onion" és molt interessant. En primer lloc, es genera un parell de claus RSA de 1024 bits. A continuació, es cerca el hash SHA-1 de la clau pública ASN.1 codificada en format DER. S´agafa la primera meitat d'aquest hash (20 de 40 caràcters) i es converteix en Base32. S´afegeix el sufix ".onion" i ja es té l'adreça. Això es pot fer manualment a l'ordinador mitjançant OpenSSL per generar la clau i amb l´ajuda d´un convertidor de Base32 (com la comanda *base32* del paquet "coreutils"). Si es fa correctament, s´hauria d'acabar amb una adreça de 16 caràcters formada només per les lletres minúscules a-z i els números 2-7.
> NOTA: Es pot fer que el servei només estigui disponible per clients autoritzats. Per més informació veure https://community.torproject.org/onion-services/advanced/client-auth i també l'article https://kushaldas.in/posts/setting-up-authorized-v3-onion-services.html
> NOTA: En qualsevol cas, recomano llegir https://support.torproject.org per saber més sobre els detalls tècnics sobre com funcionen i es localitzen els serveis "onion".
> NOTA: Una de les raons de l´existència del serveis "onion" és el fet que els nodes de sortida (els que connecten la xarxa Tor a l´Internet obert) són una debilitat important del sistema ja que poden ser controlats per entitats malintencionades per diversos motius, i són el focus de gairebé qualsevol atac a la xarxa Tor. Amb els serveis "onion", però, es poden tenir llocs web i serveis exclusivament a la xarxa Tor, de manera que els usuaris no hagin d'accedir a Internet visible.

Un detall interessant dels "hidden services" és que ofereixen un nom sense haver de registrar-lo en cap autoritzat centralitzada (com passa amb els noms DNS) i, sobre tot, no és necessari tenir cap adreça IP pública ni contractar cap VPS o servidor de hosting online, ni tan sols "obrir cap port" del nostre enrutador casolà per tractar amb la configuració NAT: l'accés al "hidden service", un cop implementat, és directe si s'és dins de la xarxa Tor.

Implementació de "relays"

Per defecte Tor actua només com a client d'entrada a la xarxa. Si es vol ajudar a enfortir i engrandir la xarxa Tor, es pot convertir el nostre node Tor a un node intern (un "relay") per permetre que trànsit d'altres clients travessin el nostre node com un punt més del seu itinerari. Això s'aconsegueix tenint a l'arxiu "/etc/tor/torrc" d'un sistema operatiu amb IP pública (que pot ser un contractat, per exemple, en Linode, DigitalOcean o AWS) la següent línia:

> * *ORPort {nº|auto}*: Informa als clients de la xarxa Tor de la seva presència per a què l´utilitzin. Es pot indicar el nombre d´un port concret per rebre les peticions dels clients (normalment es tria el 9001) o bé la paraula "auto" per a què es trïi un port a l'atzar automàticament.
>
> > NOTA: Com a mesura suplementària de seguretat, se sol establir a més la línia *SocksPort* a 0 per evitar que el nostre node sigui un node d'entrada.
> > NOTA: També és recomanable establir un quantitat màxima de bytes entrants/sortints del node Tor en un periode determinat de temps. Això s'aconsegueix amb les directives *AccountingMax nº {bytes|KBytes|MBytes|GBytes}* i *AccountingStart {day|week|month} nºdia hh:mm* Per més informació podeu consultar man tor o bé aquest article: https://support.torproject.org/relay-operators/limit-total-bandwidth També són d'interès, en aquest sentit les directives *BandwidthRate/BandwidthBurst* i *RelayBandwidthRate/RelayBandwidthBurst* , documentades a https://support.torproject.org/relay-operators/bandwidth-shaping
> > NOTA: Altres línies, opcionals, són *Nickname nom* o *Contact-Info direccio@corr.eu*

També es pot indicar que desitgem que el nostre node Tor es converteixi, a més d'un node intern, en un node de sortida (un "output relay") per així permetre que trànsit d'altres clients utilitzin el nostre node també per sortir a Internet. Cal anar amb compte amb això perquè en aquest cas les activitats que realitzen aquests clients apareixeran com que les està realitzant el nostre node de sortida. Això s'aconsegueix afegint a l'arxiu "/etc/tor/torrc", a més de la directiva *ORPort*, les següents línies:

> * *ExitRelay {0|1}* : El valor 1 activa la funcionalitat d'"exit relay" i 0 -valor per defecte-, no.

* *ExitPolicy {accept | reject } {ip | ipXarxa/mask | * }* : Regles que accepten o rebutjen connexions destinades a una determinada IP o a una determinada xarxa o a qualsevol destí. Es poden afegir les línies que es vulguin i la primera que quadri s'aplica i no es segueix llegint (a l'estil de Nftables). Per defecte s'accepten totes les connexions sortints excepte les dirigides a ports concrets, com el 25, 119, 135-139, 445 i algun més (veure *man tor* per més informació). Noteu que només es tindrà en compte aquesta directiva si *ExitRelay* val 1

Una altra opció que tenim és implementar un "bridge". Això es pot aconseguir indicant, a més de la directiva *ORPort* , aquesta altra directiva:

* *BridgeRelay {0|1}* : El valor 1 activa la funcionalitat de "bridge" i 0 -valor per defecte-, no. Això a la pràctica el que fa és que es publiqui el descriptor del servidor en la base de dades de "bridges" en lloc de en les autoritats del directori públic.

NOTA: Com a mesura suplementària de seguretat, se sol establir a més la línia *SocksPort* a 0 per evitar que el nostre node sigui un node d'entrada. Altres línies, opcionals, són *Nickname nom* o *Contact-Info direccio@corr.eu*
NOTA: Tingueu en compte, però, que aquest "bridge" serà molt fàcilment censurable ja que no incorporarà cap "pluggable transport"; si es vol afegir aquesta funcionalitat extra, caldrà instal·lar el binari apropiat ("obfs4proxy" en el cas de voler usar "obfs4", tal com s'explica aquí; https://community.torproject.org/relay/setup/bridge o "snowflake" en el cas de voler usar el PT homònim, tal com s'explica aquí: https://community.torproject.org/relay/setup/snowflake/standalone

Igualment, també podríem configurar el nostre node Tor com un "directory server" mitjançant la directiva *DirPort {nº|auto}*

NOTA: Per a més informació, https://support.torproject.org I, en general, a https://community.torproject.org/relay i https://blog.torproject.org/tips-running-exit-node
NOTA: Cal tenir en compte, en qualsevol cas, que perquè el nostre node sigui reconegut efectivament com un "relay" vàlid (ja sigui intern, de sortida o de directori) primer les autoritats hauran d'aconseguir el Consens per a això.

Altres xarxes

Tor no és l'única xarxa "oculta" que existeix a Internet. A continuació llistem altres xarxes ocultes que són importants. Els protocols interns de cadascuna són diferents però totes elles funcionen en aparença de manera similar: hem d'executar en el nostre ordinador un determinat programari que ens dóna accés a la xarxa oculta, dins de la qual podem disposar de diversos serveis, entre els quals és "sortir a la superfície" d'Internet de nou de forma anònima.

I2P (https://geti2p.net/en)
GNUNet (https://gnunet.org)
Nym Mixnet (https://nymtech.net/about/mixnet)
Lantern (https://getlantern.org) -App concreta per esquivar la censura-
FreeNet (https://freenetproject.org)
ZeroNet (https://zeronet.io)

D'altra banda també convé destacar IPFS (https://ipfs.io), el qual és una xarxa de nodes que no actuen com a elements independents sinó com a sistema P2P global de fitxers en xarxa. Això permet allotjar contingut en aquesta xarxa d'una manera distribuïda i, per tant, molt difícil de censurar i eliminar. D'altra banda, la llibreria Libp2p (https://docs.libp2p.io) permet desenvolupar aplicacions P2P.

EXERCICIS:

1.-a) Descarrega el "Tor Browser" a la màquina real i obre'l fent doble click sobre l'executable "start-tor-browser". Digues, a la pantalla inicial que apareix, que vols una connexió directa a Tor (ja que aquesta xarxa no està prohibida -encara- a Espanya). Vés-hi llavors a https://check.torproject.org per comprovar que tens una IP pública diferent de la real (la qual pots conèixer amb un navegador "estàndard" anant a la mateixa pàgina web)

b) Selecciona l'opció "New Tor circuit for this site" del menú (o pulsa CTRL+SHIFT+L) i digues quina informació ha canviat a la pàgina web anterior. ¿Per què?

c) Vés a https://geoip.com i comprova de quin país és la teva IP pública vista a l´apartat anterior (o millor dit, la de l'"exit relay" que Tor t'ha assignat)

d) Pulsa sobre la icona del cadenat que apareix a l´esquerra del tot de la barra d´adreces del navegador. ¿Què mostra la secció "Tor circuit"? D´altra banda, ¿per a què serveix la icona de l'escombra que apareix a la dreta de la barra d´adreces?

e) Clica en el botó amb la icona de l´escut que hi ha al costat de l'escombra i selecciona l'opció "Advanced security settings" (o escriu a la barra d´adreces *about:preferences#privacy*. ¿Quina diferència hi ha entre els valors "Standard", "Safer" i "Safest"?

f) ¿Per a què serveixen les seccions "Bridges" i "Advanced" de la pàgina d´opcions mostrada en *about:preferences#connection* ?

2.-a) Juga amb les quatre possibilitats que t´ofereix aquesta pàgina https://tb-manual.torproject.org/secure-connections (Tor:ON, HTTPS:ON ; Tor:ON, HTTPS:OFF ; Tor:OFF, HTTPS:ON ; Tor:OFF, HTTPS:OFF) i raona els resultats mostrats als diagrames interactius.

b) En aquest sentit, ¿què significa aquest paràgraf?

> The foreign powers whose information was compromised made a basic mistake; they misunderstood how Tor works and what it is for. The assumption is that Tor is an end-to-end encryption tool. It isn't. Tor will anonymize the origin of your browsing and message, but not the content.
>
> If you are using Tor to browse the regular internet, an exit node can snoop on your browsing session. That provides a powerful incentive for unscrupulous people to set up exit nodes solely for espionage, theft, or blackmail.

3.-a) Arrenca una màquina virtual (Ubuntu o Fedora, Server o Desktop, és igual) amb la tarja de xarxa en mode "adaptador pont" i instal.la-hi el paquet "tor". Configura, a l'arxiu "/etc/tor/torrc", la seva directiva *SOCKSPort* per a què en lloc d´escoltar a la IP "loopback" escolti a la IP de la tarja de xarxa i tot seguit posa en marxa el dimoni. Comprova que aquest funcioni (amb *systemctl status tor*) i que efectivament estigui escoltant al port 9050 amb *ss -tnl* ¿Què ensenyen les línies que comencen per la paraula "Bootstrapped" de la sortida de *journalctl -u tor*?

aII) Observa la comanda indicada a la línia *ExecStart=* que apareix a la sortida de la comanda *systemctl cat tor* i esbrina (consultant si cal la secció "COMMAND-LINE OPTIONS" de *man tor*) quin és el fitxer que conté la configuració per defecte del servei *tor* (i que pot ser modificable, si així s'escau, dins de "/etc/tor/torrc"). Observa també les directives que hi apareixen en aquest fitxer i comprova el seu significat a *man tor*

b) Configura el navegador Firefox "estàndar" de la màquina real per fer-lo servir (activant a més, atenció, l'opció "Proxy dns when using socks v5", per una raó que veurem a l´apartat e) d'aquest exercici) i tot seguit obre'l. Comprova que hagis entrat efectivament a Tor anant un altre cop a https://check.torproject.org

> NOTA: Recorda que per a què el Firefox pugui accedir a la xarxa Tor li has de configurar un servidor proxy "SOCKSv5"
> NOTA: L'opció "Proxy dns when using socks v5" equival a la clau d configuració *network.proxy.socks_remote_dns* del Firefox
> NOTA: En realitat, el que fa el "Tor Browser" és redireccionar totes les peticions a un "proxy" Tor local embegut, disponible 127.0.0.1:9150, el qual farà el xifratge i reenviament dels paquets al node d´entrada de la xarxa Tor (amb la qual cosa, tot el trànsit que surti de la màquina local ja serà anònim, cosa que si fem servir un proxy Tor remot -que podria ser local també- en combinació amb un navegador estàndard, com fet en aquest apartat, no passarà fins que no arribem a ell).

c) Arrenca una segona màquina virtual amb la tarja de xarxa en mode "adaptador pont" i instal.la el paquet "torsocks" (si no ha sigut instal·lat com a dependència ja) i configura'l per a què es connecti al servidor Tor remot que està funcionant a l´altra màquina virtual. Executa la comanda *curl ipinfo.io/ip* i tot seguit *torsocks curl ipinfo.io/ip* ¿Quina diferència hi ha entre el resultat de la primera i la segona?

> NOTA: També pot fer servir la comanda curl https://check.torproject.org/api/ip

cII) Si tornes a executar de nou les mateixes comandes *torsocks* anteriors, després d'haver reiniciat el servei *tor*, ¿la IP obtinguda a la segona execució és la mateixa que la que vas obtenir el primer cop amb torsocks i sense torsocks? Per què?

cIII) Torna a repetir la comanda *torsocks curl ipinfo.io/ip* però ara amb un Wireshark/Tshark encés a la màquina "client" (instal·la'l abans si calgués). ¿Quin tipus de tràfic veus?

> NOTA: Has de tenir en compte que el tràfic Tor, vist des del punt de vista del Wireshark/Tshark, és simple tràfic TCP amb TLS. Per tant, no disposarem de cap dissector integrat que ens indiqui que els paquets observats són de tipus Tor. Però sí que podem observar els ports implicats (9050, 9001, 9030 -aquest últim per contactar amb els servidors de directori per obtenir la llista de nodes-, etc) per intentar reconéixer aquest tipus de tràfic. Per més informació, podeu llegir el següent article: https://samsclass.info/122/proj/how-socks5-works.html

d) Torna a repetir de nou la comanda *torsocks curl ipinfo.io/ip* però ara indicant al Wireshark/Tshark el display filter "dns" ¿Per què no apareixen les peticions DNS? En aquest sentit, ¿a quina comanda creus que seria equivalent la comanda *torsocks curl www.hola.com*: a *curl -x socks5://ip.serv.Tor:9050 www.hola.com* o a *curl –x socks5**h**://ip.serv.Tor:9050 www.hola.com* (consulta la pàgina del manual de *curl*)?

e) Configura ara el servidor *tor* de la màquina virtual "servidora" per a què només utilitzi nodes d'entrada de França i nodes de sortida d'Itàlia. Després de reiniciar-lo, obre el navegador Firefox de la màquina real (ja configurat per fer-lo servir) i vés a https://check.torproject.org de nou. ¿De quin país és la IP que uses ara? (o pots comprovar consultant-la a webs com https://geoip.com)

f) ¿Quines directives hauries d'indicar al fitxer de configuració de Tor (no cal que ho facis, només explica-les) per convertir el teu servidor *tor* en un "exit relay"? ¿I en un "bridge" (sense "pluggable transport")?

4.-a) Instal.la el paquet "nyx" a la màquina virtual de treball. Edita la configuració del proxy Tor per a descomentar la línia *ControlPort* (la qual permet habilitar un port -per defecte 9051- en exclusiva per poder ser administrat i monitoritzat), escriu la línia *CookieAuthentication 0* (la qual deshabilita qualsevol mecanisme d'autenticació per controlar del proxy...això ho fem per simplificar) i reinicia'l. Executa en un terminal la comanda *nyx* i tot seguit comença a navegar, amb el navegador Firefox "estàndar" fent servir el servei Tor del sistema ¿Què veus?

> NOTA: Si volguessis utilitzar la comanda *nyx* remotament, caldria primer que entressis a la màquina servidora via SSH o similar, ja que el port 9051 només està disponible de forma local

b) ¿Què passa si, a la pantalla del nyx, pulses la tecla "i"? I la tecla "s"? I la tecla "e"? I "f"? I "c"?

c) Si pulses "m" i vas al menú "View->Connection" i la sel.lecciones pulsant "Espai", ¿què veus? ¿Què passa, si en aquesta pantalla, pulses la tecla "s"? I si pulses "Enter" en una connexió determinada? (pots mirar l'ajuda amb la tecla "h")

d) Si pulses "m" i vas al menú "View->Config" i la sel.lecciones pulsant "Espai", ¿què veus? ¿Què passa, si en aquesta pantalla, pulses "Enter" sobre una línia determinada? I si pulses "w"? (pots mirar l'ajuda amb la tecla "h"). Pulsa finalment "q" per sortir

> NOTA: Un exemple d'arxiu de configuració del propi Nyx es troba a https://nyx.torproject.org/nyxrc.sample

5.-a) Instal.la un servidor Apache2 a la màquina "servidora" i fes que només escolti a través de la IP loopback (això ho pots fer indicant la directiva *Listen 127.0.0.1:80* en lloc d'on estigui present la directiva *Listen* en la configuració actual del servidor...a Ubuntu és a l'arxiu "/etc/apache2/ports.conf" i a Fedora a l'arxiu "/etc/httpd/conf/httpd.conf").

aII) Fes ara que aquest servei estigui disponible públicament al port 80 dins de la xarxa Tor com a servei "onion", tal com s'explica a la teoria (bàsicament es tracta d'afegir dues directives de configuració a l'arxiu "/etc/tor/torrc").

aIII) Accedeix des de la màquina "client" (bé amb el "Tor Browser" o bé mitjançant un curl "torificat") al domini ".onion" autogenerat del teu servidor per confirmar que la pàgina per defecte de l'Apache es troba visible. Comprova també que des d'un navegador estàndar sense fer servir la xarxa Tor aquest domini ".onion" no porta enlloc.

aIV) Demana a algun company que es connecti a la xarxa Tor i intenti arribar al teu servidor web implementat a l'apartat anterior. Pot?

> NOTA: Teniu més informació sobre els serveis "onion" a https://community.torproject.org/onion-services/overview I una guia de "bones pràctiques" a https://riseup.net/en/security/network-security/tor/onionservices-best-practices . També teniu un article sobre com usar TLS en serveis "onion" a https://community.torproject.org/onion-services/advanced/https i un altre article sobre com evitar atacs DoS a https://community.torproject.org/onion-services/advanced/dos

b) ¿Per a què serveixen https://www.tor2web.org o https://www.tor2web.nl i com es pot fer ús d´aquests projectes?

c) ¿Per a què serveix aquest programa: https://github.com/cathugger/mkp224o ?

d) ¿Per a què serveis aquest programa: https://github.com/torproject/onionbalance ?

6.-a) Llegeix a la teoria (o a https://community.torproject.org/relay/types-of-relays , com vulguis) la resposta a les preguntes:

* ¿Què és un "bridge"?
* ¿Què és un "pluggable transport" (PT)?
* ¿Quina diferència hi ha entre els PT de tipus "obfs4" i els de tipus "meek"?
*¿Què hauries de fer per sol·licitar l´ús d´un determinat "bridge" ofert per BridgeDB?

aII) ¿Per a què serveix aquesta web https://relay.love ?

b) Obre el Wireshark/Tshark i el navegador Tor. ¿Quina diferència veus en el tràfic registrat si comences a navegar sense tenir cap "pluggable transport (PT)" activat o bé si n'actives alguns dels disponibles per defecte?

c) ¿Quina diferència hi ha entre el que expliquen les guies disponibles a https://community.torproject.org/relay/setup/guard i a https://community.torproject.org/relay/setup/bridge? ¿I a https://community.torproject.org/relay/setup/exit?

> NOTA: En qualsevol cas, també convé llegir-se https://community.torproject.org/relay/setup/post-install

cII) ¿Què explica aquest article: https://community.torproject.org/relay/relays-requirements? ¿I aquest https://community.torproject.org/relay/technical-considerations ?

d) ¿Per què és important llegir-se https://community.torproject.org/relay/community-resources abans d'implementar un "exit relay"?

dII) Llegeix l'article https://www.fwhibbit.es/24-horas-en-la-vida-de-un-nodo-de-salida-de-tor i digues què explica. ¿Per a què s'usa cada aplicació de les allà mencionades: *vnstat*, *dnstop*, *darkstat* i *tcpdump*?

dIII) ¿Què explica https://support.torproject.org/#operators-7 ? ¿I https://support.torproject.org/#exit-relay-expectations? ¿I https://support.torproject.org/#exit-policies?

7.-a) Menciona dues estadístiques diferents que mostri cadascuna dels apartats "Users", "Servers", "Traffic", "Performance", "Onion Services" i "Applications" de la pàgina https://metrics.torproject.org

> NOTA: Pots saber més sobre com funciona la infraestructura interna d´aquesta pàgina a https://metrics.torproject.org/about.html

b) ¿Per a què serveix el servei "Relay search" disponible a https://metrics.torproject.org/services.html?

> NOTA: Una aplicació web que funciona sobre el servei anterior però que és més fàcil és https://irl.github.io/atlas

c) ¿Què mostra la web https://consensus-health.torproject.org ?

d) ¿Què mostra la web https://www.dan.me.uk/tornodes ?

> NOTA: Una web similar és https://torstatus.rueckgr.at (el seu codi font és https://github.com/paulchen/torstatus). No obstant, no està mantinguda. Actualment, l'alternativa oficial és Onionoo (https://www.torproject.org/projects/onionoo.html.en), la qual proporciona informació de l'estat de la xarxa Tor en format JSON; d'altra banda, Atlas (https://atlas.torproject.org) presenta aquestes dades en un format més agradable pels humans. Atlas pot agafar les dades de la instància pública Onionoo o bé d'una instància Onionoo pròpia que volguem instal·lar en els nostres servidors.

e) ¿Per a què serveix el formulari web https://www.ipqualityscore.com/tor-ip-address-check ?

f) ¿Per a què serveixen webs com https://ahmia.fi , https://tor.link o https://onionsearchengine.com ?

g) ¿Què pretén aquest projecte: https://ooni.org/nettest ? En aquest sentit, tot i que no té res a veure amb Tor, ¿què implementa el projecte https://www.opennic.org?

h) ¿Què pretén aquest projecte: https://securedrop.org ? ¿I https://onionshare.org?

i) ¿Què pretén aquest projecte: https://gfwatch.org/overview ?

8.- a) Instal·la i prova l´applicació Orbot al teu dispositiu Android (si és que en tens)

b) ¿Què és Tails (https://tails.net)? ¿I Whonix (https://www.whonix.org)?