Gestió de DNS (en el costat client) i mDNS/LLMNR

L'arxiu "/etc/nsswitch.conf" (i "/etc/hosts")

L'arxiu "**/etc/nsswitch.conf**" determina, entre altres coses, l'ordre dels diferents tipus de "fonts d'informació" a les que els programes del nostre ordinador aniran preguntant per saber (gràcies a la primera resposta que se n'obtingui) quina és la IP associada a un determinat nom (o viceversa) cada cop que ho necessitin saber.

> NOTA: Tècnicament, aquest fitxer és llegit per qualsevol programa que faci servir la funció *gethostbyname()* de la llibreria estàndard GLib C. És molt interessant, en aquest sentit, la pàgina del manual d'aquesta funció

Concretament, els tipus de fonts d'informació que poden donar una resposta sobre associacions possibles entre IPs i noms de màquina es troben ordenats dins d'aquest fitxer "/etc/nsswitch.conf" a la línia que comença per la paraula *hosts:* A continuació mostrem el contingut per defecte exacte d'aquesta línia a un sistema Fedora 35 i a un sistema Ubuntu 22.04, respectivament, i tot seguit l'expliquem (per més informació, consulteu la pàgina del manual d'aquest fitxer):

Fedora Works: *hosts: files myhostname mdns4_minimal [NOTFOUND=return] resolve [!UNAVAIL=return] dns*
Fedora Server: *hosts: files myhostname resolve [!UNAVAIL=return] dns*
Ubuntu Desktop: *hosts: files mdns4_minimal [NOTFOUND=return] dns*
Ubuntu Server: *hosts: files dns*

*Directiva ***files*** : Aquest valor apareix sempre perquè es té el paquet "glibc" instal·lat. Indica que les associacions IP<->nom s'han d'anar a buscar a un arxiu de text local anomenat "**/etc/hosts**", les línies del qual ha de tenir el format *dir.ec.cio.IP nomAssociat... nomAssociat2 ...*, i on cadascuna serveix per vincular la IP indicada amb el/s nom/s escrit/s al seu costat.

La idea de col·locar aquesta directiva la primera és per a què el sistema pugui saber, en consultar aquest arxiu primer, la reciprocitat IP<-> nom/s sense haver de consultar a cap servidor DNS (procés que és molt més costós en temps i ús de recursos -i també susceptible a errors si el funcionament o configuració de la xarxa no és correcte-). Per tant, si la primera directiva de la línia *hosts:* és *files*, el sistema primer llegeix el contingut de l'arxiu "/etc/hosts" i, només si no hi trobés la parella IP<->nom buscada, llavors passarà a la següent opció, que sol ser (ara ho veurem) preguntar a algun servidor DNS que hi hagi configurat.

> NOTA: Lògicament la directiva *files* podria aparèixer escrita en una altra posició diferent de la primera; en aquest cas, primer el sistema preguntaria les altres fonts i si aquestes no responguessin, llavors consultaria l'arxiu "/etc/hosts" quan "li toqués el torn".

Hom podria pensar que disposar del fitxer "/etc/hosts" ens podria estalviar l'ús del protocol DNS però aquesta idea té varis problemes. Per exemple, el manteniment actualitzat d'aquest arxiu en múltiples ordinadors de la nostra LAN seria una tasca massa farragosa i propensa a errors/oblits quan s'incorporessin/s'eliminessin màquines (de fet, per aquesta raó es va inventar el protocol DNS, que és un sistema centralitzat de noms). D'altra banda, més enllà de la nostra LAN, si volguéssim tenir un fitxer "/etc/hosts" amb tots els noms d'ordinadors d'Internet (de fet, al principi era així), està clar que això seria molt poc eficient degut a la llargària que acabaria agafant el fitxer (a la vegada que la llista IP<->nom/s sempre seria incompleta). És per això que en organitzacions amb un parc d'ordinadors elevat, la posada en marxa d'un servidor DNS local és la millor manera de simplificar molt l'administració de la xarxa (en qualsevol cas, sempre és possible combinar les dues solucions: afegir una llista petita de noms concrets a l'arxiu "/etc/hosts" i, després, per la resta de noms que allà no hi siguin, preguntar als servidors DNS configurats).

> NOTA: No confondre l'arxiu "/etc/hosts" amb un altre arxiu anomenat "**/etc/hostname**". Aquest darrer arxiu serveix per autoassignar a la màquina el seu nom local, que és el que apareix al prompt. Aquest nom local pot estar repetit als diferents ordinadors de la LAN perquè no es publica de cap manera. Per gestionar aquest tipus de noms es pot fer servir la comanda *hostnamectl* (sense paràmetres mostra aquest nom i altres valors relacionats; si escrivim *hostnamectl set-hostname nouNom* estableix un nou nom).

*Directiva ***myhostname*** : Aquest valor apareix perquè es té el paquet "systemd-libs" (a Fedora) o "libnss-myhostname" (a Ubuntu) ja instal.lat; si no aparegués, només caldria instal·lar-lo i llavors afegir la directiva en qüestió allà on desitjem en la línia *hosts:*. Aquesta directiva serveix per a què els programes associïn automàticament tres coses: el nom de la màquina indicat a "/etc/hostname" (és a dir, l'establert via comanda *hostnamectl*) a les IPs actives de la màquina (o, si no n'hi ha cap, a la IP 127.0.0.2), els noms "localhost" i "localhost.localdomain" a la IP 127.0.0.1 i el nom "_gateway" a la porta d'enllaç per defecte que tingui la màquina. La idea és que, si l'arxiu "/etc/hosts" no ha sigut editat manualment, el contingut d'aquest arxiu "/etc/hosts" (és a dir, el seu contingut per defecte) serà redundant i, per tant, no caldrà fer servir per res aquest arxiu.

---

*Directiva ***mdns4_minimal*** : Aquest valor apareix perquè es té el paquet "nss-mdns" (a Fedora) o "libnss-mdns" (a Ubuntu) ja instal.lat; si no aparegués, només caldria instal·lar-lo i llavors afegir la directiva en qüestió allà on desitjem en la línia *hosts:*. Estudiarem per a què serveix aquest paquet més endavant, quan veiem el protocol mDNS

NOTA: Es pot veure com a continuació d'aquesta directiva apareix l'expressió *[NOTFOUND=return]* (la qual, de fet, com la resta de possibles expressions que hi poden haver -consulteu *man nsswitch.conf* per coneixer-les-, pot afegir-se al darrera de qualsevol altra directiva). Aquestes expressions afecten al valor just anterior on apareixen i, en concret l'expressió *[NOTFOUND=return]* indica que s'aturi la recerca (i, per tant, no es continuï buscant a la resta de valors que segueixin en la línia *hosts:* , cosa que per defecte seria el que passaria) si el valor en qüestió ha trobat la font d'informació adient però aquesta ha contestat que no té resposta (això està pensat per retornar ràpid en fer la resolució dels dominis ".local", exclusius del protocol mDNS i no existents en el protocol DNS). Una altra expressió també habitual és *[UNAVAIL=return]* la qual indica que s'aturi la recerca si el valor en qüestió no ha trobat cap font d'informació, associada a la directiva a la què afecta, disponible. Si es precedeix la paraula en majúscules del símbol "!", s'inverteix el significat

---

*Directiva ***resolve*** : Aquest valor apareix perquè es té el paquet "systemd-libs" (a Fedora) o "libnss-resolve" (a Ubuntu) ja instal.lat; si no aparegués, només caldria instal·lar-lo i llavors afegir la directiva en qüestió allà on desitjem en la línia *hosts:*. Indica que el fitxer "/etc/resolv.conf" és totalment ignorat ja que la gestió dels servidors DNS va a càrrec del dimoni *systemd-resolved* Per tant, tots els programes que facin servir la funció *gethostbyname()* de GLib C preguntaran directament a aquest dimoni (la configuració del qual estudiarem en propers apartats) a quin servidor DNS han de realitzar les peticions i serà aquest dimoni qui els ofereixi la resposta adient.

NOTA: Fixar-se que, a la configuració estàndard de Fedora, després de la directiva *resolve* apareix l'expressió *[!UNAVAIL=return]* . Això vol dir que si *systemd-resolved* està funcionant no se seguirà preguntant, tot i que no s'obtingui cap resultat, ja que la directiva que ve després (*dns*) només hi és per si *systemd-resolved* no està funcionant, com a "mal menor". Fixeu-vos, doncs, que en el cas de no tenir *systemd-resolved* funcionant, sí s'arribarà a llegir l'arxiu "/etc/resolv.conf" degut a la directiva *dns* (que explicarem tot seguit)

NOTA: Tant en Ubuntu com en Fedora encara existeix el fitxer "/etc/resolv.conf" per mantenir la retrocompatibilitat amb els programes que llegeixen directament aquest fitxer en comptes de fer servir "/etc/nsswitch.conf" (és a dir, *gethostbyname()*) per aquesta tasca. No obstant, en aquest cas "/etc/resolv.conf" no és res més que un enllaç a l'arxiu "**/run/systemd/resolve/stub-resolv.conf**" (generat automàticament cada cop que *systemd-resolved* arrenca), el qual conté simplement la línia *nameserver 127.0.0.53*, el que a la pràctica les peticions DNS dels programes vagin dirigides a un "servidor DNS" local, que resultarà ser, oh sorpresa, *systemd-resolved* perquè aquest dimoni escolta "peticions" en aquesta IP loopback.

NOTA: També pot donar-se el cas (estrany) de què "/etc/resolv.conf" fos un link apuntant a un altre fitxer diferent anomenat "**/run/systemd/resolve/resolv.conf**" , el qual conté la llista d'IPs de tots els servidors DNS recopilada durant l'últim arranc de "systemd-resolved" (cadascun indicat en una línia començada per la paraula *nameserver*); en aquest cas, però, els programes ja no utilitzaran a systemd-resolved per preguntar-li a ell sinò que consultaran aquest arxiu directament, el que és una solució més pobre perquè la llista allà existent és global i no distingeix si els servidors DNS només s'han d'usar per una determinada tarja, etc. De fet, en aquest cas no cal tenir ni tan sols *systemd-resolved* funcionant un cop "/run/systemd/resolve/resolv.conf" ja estigui construït.

NOTA: En cas de tenir el dimoni NetworkManager funcionant, cal saber que podríem tenir diverses situacions:

*Si "/etc/resolv.conf" és un enllaç a "/run/systemd/resolve/stub-resolv.conf" (situació més habitual en les distribucions importants, tal com hem dit), NetworkManager delega la gestió dels servidors DNS en "Systemd-resolved" (els detalls de la qual explicarem al següent apartat)

NOTA:El mateix passarà si "/etc/resolv.conf" és un arxiu però NetworkManager està funcionant amb la línia *dns=systemd-resolved* a la seva configuració (ja que en aquest cas el contingut de "/etc/resolv.conf" serà el mateix, una línia *nameserver 127.0.0.53*, així que a la pràctica no hi veurem gaire diferència).

*Si "/etc/resolv.conf" és un fitxer i NetworkManager no té cap línia *dns=* explícita a la seva configuració (o en té la línia *dns=default*, que ve a ser el mateix) "/etc/resolv.conf" serà gestionat directament pel NetworkManager i a la pràctica això vol dir que en iniciar-se *systemd-resolved* aquest es limitarà a usar com a simple consumidor els servidors DNS indicats a "/etc/resolv.conf" (els quals haurà obtingut NetworkManager a la seva manera; de fet, NetworkManager el que sol fer és convertir "/etc/resolv.conf" en un link a "/var/run/NetworkManager/resolv.conf" i escriure-hi allà els servidors DNS que ell mateix recopila de la seva pròpia configuració o via DHCP.); això no ens interessa pas! D'altra banda, també pot haver la línia *dns=none*, que fa que NetworkManager no interaccioni per res amb "/etc/resolv.conf"

*Directiva ***dns*** : Aquest valor apareix sempre perquè es té el paquet "glibc" instal·lat. Indica que es llegirà l'arxiu "/etc/resolv.conf" i, per tant, que es tindran en compte els servidors DNS allà indicats. En el cas de que el dimoni *systemd-resolved* estigui funcionant, allà hi apareixerà el "servidor" 127.0.0.53, el que fa que, a la pràctica, els programes facin servir igualment aquest dimoni tal com passava amb la directiva *resolve* però amb una gran diferència: ara sí es llegeix l'arxiu "/etc/resolv.conf", amb la qual cosa qualsevol modificació manual que es faci sobre aquest fitxer (cosa gens recomanable!) també es tindrà en compte.

NOTA: Una altra diferència entre les directives *resolve* i *dns* és que la primera fa que els programes es comuniquin amb el dimoni *systemd-resolved* via D-Bus (antigament) o Varlink (actualment) mentre que la segona fa que s'hi comuniquin via un socket IP. En tot cas, un programa podria comunicar-se directament via Varlink (o D-Bus) amb *systemd-resolved* sense necessitat de consultar "/etc/nsswitch.conf" per trobar la paraula *resolve* si és desenvolupat mitjançant l'API de *systemd-resolved* directament en comptes de amb la de Glib C.

*Directiva ***mymachines*** : Aquest valor apareix perquè es té el paquet "systemd-container" (a Fedora) o "libnss-mymachines" (a Ubuntu) ja instal.lat; si no aparegués, només caldria instal·lar-lo i afegir la directiva en qüestió allà on desitjem en la línia *hosts:*. Aquest valor serveix per a què els programes associïn automàticament els noms dels contenidors locals gestionats per *systemd-machined* a les seves corresponents IPs. D'aquesta manera, a la màquina amfitriona es podran fer servir els noms dels contenidors en comptes de les seves IPs per connectar-hi. Aquest valor també s'usa a les línies *passwd:* ... i *group:* ... de l'arxiu "/etc/nsswitch.conf" per tal de reconèixer també els usuaris i grups de dins del contenidor.

NOTA: S'ha de tenir en compte que determinats programes, com per exemple els navegadors, si tenen configurat un proxy, no intenten realitzar cap resolució de noms i deleguen directament aquesta feina al proxy (això té com a conseqüència, per exemple, que l'arxiu "/etc/hosts" local no s'utilitzi o que els servidors DNS configurats a la màquina local tampoc, atenció!).

El servei "resolver" DNS *systemd-resolved*

Als sistemes Linux moderns funciona de forma permanent un dimoni anomenat "systemd-resolved" escoltant de forma local al port 53 (tant TCP com UDP), el qual és el responsable d'atendre de forma centralitzada totes les peticions DNS fetes pels diferents programes del sistema i de reenviar-les al servidor DNS que toqui segons la seva pròpia configuració.

Una avantatge de fer servir *systemd-resolved* és que proporciona a tots els programes del sistema una memòria cau global i comuna contenint les respostes DNS obtingudes recentment. Si no es fes servir, cada programa hauria de mantenir una memòria cau pròpia individual (o bé no mantenir-ne cap i preguntar llavors als servidors DNS constantment les mateixes peticions un cop i un altre).

NOTA: Per confirmar si tots els programes del sistema (fins i tot els que no utilitzen la crida *gethostbyname()*, tal com s'ha explicat a l'apartat anterior) fan servir el dimoni *systemd-resolved*, només cal confirmar si apareix la línia *nameserver 127.0.0.53* dins de l'arxiu "/etc/resolv.conf"; aquesta línia indica als programes que encara llegeixen directament aquest arxiu que el servidor DNS que han de fer servir és un "servidor DNS" local que està escoltant en aquella IP loopback (i en el port 53, tant TCP com UDP), el qual no és res més que el propi dimoni *systemd-resolved*

La llista de servidors DNS als quals el dimoni "systemd-resolved" farà les consultes es regenera en RAM cada cop que aquest dimoni s'inicia, a partir de la recollida d'informació que fa de diferents llocs:

a) Dins de l'arxiu "**/etc/systemd/resolved.conf**", el qual representa la configuració general del servei, es poden indicar les IPs dels servidors DNS a utilitzar mitjançant la línia "*DNS=...*" (una darrera l'altra separades per espais). Aquests servidors són globals per totes les tarjes de xarxa de la màquina, de forma similar a com funcionava l'antic arxiu "/etc/resolv.conf"

NOTA: Una altra forma de personalitzar la configuració de *systemd-resolved* és crear un nom de fitxer "elquesigui.conf" dins de la carpeta "/etc/systemd/resolved.conf.d" amb el contingut següent (per exemple):

```
[Resolve]
DNS=1.1.1.1 1.0.0.1
...
```

En fer-ho així, no es tocarà el fitxer de configuració principal (el qual té la precedència més baixa; això vol dir que les entrades col·locades en un fitxer dins del directori "resolved.conf.d" anul·len les entrades homònimes col·locades al fitxer de configuració principal). D'altra banda, si hi haguessin diversos fitxers dins de la carpeta "resolv.conf.d", s'ordenaran pel seu nom de fitxer en ordre lexicogràfic, de manera que si diversos fitxers especifiquen la mateixa opció, l'entrada al darrer fitxer té prioritat. Per tant, per simplificar l'ordenació dels fitxers, es recomana posar tots els noms de fitxer en aquests subdirectoris amb un número de dos dígits i un guió.

b) Dins de l'arxiu ".network" corresponent a cada tarja de xarxa. Concretament, cada IP dels servidors DNS s'ha d'indicar en una línia "*DNS=...*" diferent. Aquests servidors tenen preferència, per la tarja en qüestió, sobre els servidors indicats al punt anterior.

NOTA: Recordem com era el contingut d'un exemple de fitxer "/etc/systemd/network/elquesigui.network"

```
[Match]
    Name=enp1s0
[Network]
    DHCP=no
    Address=10.1.10.9/24
    Gateway=10.1.10.1
    DNS=10.1.10.2   #Cada servidor DNS ha d'indicar-se en una línia separada. És recomanable
    DNS=10.1.10.3   #indicar dos com a mínim per si el primer falla
```

c) La provinent d'algun servidor DHCP de la xarxa que hagi estat rebuda per alguna tarja de xarxa.

NOTA: Recordem com era el contingut d'un exemple de fitxer "/etc/systemd/network/elquesigui.network"

```
[Match]
   Name=enp1s0
[Network]
   DHCP=yes
```

Recordem també que existeix la possibilitat d'obtenir del servidor DHCP totes les dades necessàries (direcció IP, porta d'enllaç, etc) però rebutjant els servidors DNS oferts (per tal de configurar estàticament uns altres mitjançant la línia "*DNS=...*" comentada al paràgraf anterior) ; això és possible si s'indica la següent secció dins de l'arxiu .network corresponent:

```
[DHCPv4]
UseDNS=no
```

La idea és que tots els programes del sistema (ja sigui directament via la directiva *resolve* del fitxer "/etc/nsswitch.conf" o bé a través de la IP 127.0.0.53 indicada a "/etc/resolv.conf" via la directiva *dns* del mateix fitxer "/etc/nsswitch.conf") preguntaran al dimoni *systemd-resolved* i ell sabrà accedir al servidor DNS adient en cada cas. Aquest dimoni fa les consultes al servidor DNS més específic per cada tarja; si aquest no respongués després de passat un determinat "timeout" -no configurable, encara-, aniria a fer la consulta al següent servidor DNS de la llista (equivalent a l'anterior), i així, fallada rera fallada, fins arribar al darrer servidor més general, indicat a la configuració global. Si s'arribés al darrer servidor DNS de la llista i aquest també fallés, es tornaria a començar pel primer. En qualsevol cas, *systemd-resolved* té memòria i cada petició la comença fent contra el darrer servidor DNS conegut que va contestar bé, no pas des del principi de la llista cada cop. Per més informació, consulteu https://github.com/systemd/systemd/issues/5755

Tant a l'arxiu general "/etc/systemd/resolved.conf" com a l'arxiu ".network" particular d'una tarja de xarxa (o també via DHCP) es pot establir opcionalment la directiva "*Domains=...*" Aquesta directiva serveix per definir els dominis DNS (poden ser varis si es separen entre sí per un espai en blanc) que per defecte s'afegiran a qualsevol nom de màquina simple que s'hi escrigui, de forma que es completi un FQDN sense haver d'escriure'l "a mà".

> NOTA: Aquests dominis que automàticament s'afegeixen com una "cua" a vegades s'anomenen "dominis search" perquè si els FQDNs complets no es poden resoldre, s'intenta de nou traient el seus subdominis de més a l'esquerra, i així de forma recursiva fins provar només afegint els TLD (és a dir, es va "buscant" a veure quina "cua" és la "bona"). Per exemple, si tenim la directiva *Domains= cocacola.com pepsicola.com*, en fer *ping pepe* en realitat s'executarà primer *ping pepe.cocacola.com* i, si no es resol el nom, llavors s'intentarà *ping pepe.pepsicola.com* i si tampoc, finalment s'intentarà *ping pepe.com* (en aquest cas el TLD coincideix als dos dominis).

Si algun dels dominis indicats com a valor de la directiva *Domains=* (ja sigui en un arxiu ".network" o bé en l'arxiu general "resolved.conf") ve precedit del símbol "~" (és a dir, per exemple així: *Domains=~hola.com*), aquest domini és anomenat "domini d'enrutament". En aquest cas, aquest domini no s'afegeix com a cua a cap nom escrit sinó que serveix per indicar que per tal de resoldre noms DNS amb aquest domini caldrà usar els servidors DNS concrets configurats a la tarja de xarxa en qüestió (via directiva *DNS=* del mateix arxiu ".network" o bé rebuts via DHCP) o bé usant els servidors DNS globals (si la directiva *Domains=* està especificada a l'arxiu "resolved.conf").

> NOTA: L'explicat al paràgraf anterior és important de tenir en compte, per exemple, a l'hora de configurar VPNs de manera que les peticions DNS no "s'escapin" a servidors fora d'elles.

Cal tenir en compte, però, que els "dominis search" són per defecte també "dominis d'enrutament", així que sí se n'especifica algun a la directiva *Domains=* d'un fitxer ".network" o del fitxer global "resolved.conf", primer es completarà el FQDN del nom a resoldre (si calgués perquè només s'hagi indicat el nom curt, sense cap punt) i, en tot cas, tot seguit es farà la resolució del nom FQDN, tal com s'explica al paràgraf anterior.

> NOTA: Existeix el domini d'enrutament especial "~.", que serveix per indicar que totes les resolucions DNS es faran a través dels servidors DNS configurats per la interfície de xarxa en qüestió (a no ser que hi hagi algun domini d'enrutament més específic associat a una altra interfície de xarxa diferent). Alternativament, també es pot activar/desactivar per cada tarja de xarxa un "flag" anomenat "default-route" (de seguida veurem com) que, si val *true* (valor per defecte), permet que qualsevol petició DNS per la qual no hi hagi associat en cap altra tarja cap domini d'enrutament pugui enviar-se a través de la tarja en qüestió; si val *false* llavors només les peticions DNS dirigides als dominis d'enrutament explícitament indicats per la tarja en qüestió podran ser enviades a través d'ella (si no n´hi ha cap, llavors cap petició DNS podrà ser enviada per allí).

Així doncs, "systemd-resolved" ha de decidir, cada cop que vol fer una petició DNS, per quina de les tarjes de xarxa del sistema ha d'enviar-la, i aquesta decisió la pren en base al valor de la directiva *Domains=* que estigui assignada a cada tarja en qüestió, emprant llavors el servidor indicat a la directiva *DNS=* corresponent (o bé l'obtingut via DHCP)

> NOTA: Cal saber que existeix la possibilitat d'obtenir del servidor DHCP totes les dades necessàries (direcció IP, porta d'enllaç, etc) però rebutjant els "dominis search" oferts (per tal així de configurar estàticament uns altres mitjançant la línia "*Domains=...*" comentada al paràgraf anterior) ; això és possible si s'indica la següent secció dins de l'arxiu .network corresponent (de fet, és el seu valor per defecte):
>
> *[DHCPv4]*
> *UseDomains=no*

El client de "systemd-resolved" *resolvectl*

La comanda *resolvectl* està dissenyada específicament per preguntar al servei *systemd-resolved* de la màquina local. D'aquesta manera podrem comprovar interactivament si aquest ens contesta i què ens contesta. Algunes de les seves opcions són:

resolvectl [status [nomTarja]]	Obté la llista de servidors DNS que *systemd-resolved* té recopilats de diferents orígens, ja siguin globals o per cada tarja de xarxa (o només per una en concret si s'indica). També informa dels dominis "search" i/o d'"enrutament" configurats (globalment o per cada tarja) i de si hi està activat DNSSEC, DNSoverTLS o altres protocols alternatius a DNS com LLMNR o mDNS.
resolvectl dns [nomTarja]	Mostra (només) quins servidors DNS s'utilitzen globalment i/o per cada tarja de xarxa de forma efectiva NOTA: Hi ha una comanda similar per cada protocol que resolvectl coneix: *resolvectl llmnr, resolvectl mdns, resolvectl dnssec, resolvectl dnsovertls* ... En realitat, tots aquests no són més que filtres de *resolvectl status*
resolvectl dns nomTarja ip.serv.DNS	Estableix de forma efímera un servidor DNS particular per la tarja indicada
resolvectl domain [nomTarja]	Mostra els dominis "search" i/o d'"enrutament" configurats, ja siguin globals o per cada tarja (o només per una en concret si s'indica)
resolvectl domain nomTarja dom.ini	Estableix de forma efímera un domini "search" particular (o d'"enrutament", si es precedeix amb el símbol "~") per la tarja indicada
resolvectl default-route [nomTarja]	Mostra si les tarjes tenen el "flag" "default-route" activat o no
resolvectl default-route nomTarja { true | false }	Estableix el valor del "flag" "default-route" per la tarja de xarxa indicada
resolvectl query nomMaquina	Demana al servei *systemd-resolved* que pregunti als servidors DNS que té recopilats (o que busqui a la seva cau local) per obtenir la IP de la màquina indicada. NOTA:Es pot fer també una consulta mDNS així: *resolvectl -p mdns query nom.local*
resolvectl query direccioIP	Demana al servei *systemd-resolved* que pregunti als servidors DNS que té recopilats (o que busqui a la seva cau) pel obtenir nom de la màquina associat a la IP indicada.
resolvectl -t registre query {nomMaquina |dom.ini}	Demana al servei *systemd-resolved* que pregunti als servidors DNS que té recopilats (o que busqui a la seva cau) pel obtenir el valor dels registres indicats (A, AAAA, CNAME, MX, NS...) pertanyents a la nom de màquina (o domini) indicat.
resolvectl {statistics | reset-statistics }	Informa sobre/Reinicialitza les estadístiques de peticions i respostes
resolvectl flush-caches	Esborra la cau local amb les respostes DNS obtingudes fins ara. Actualment no existeix (encara) cap opció per veure les entrades actualment existents a la memòria cau (veure https://github.com/systemd/systemd/issues/14796)

NOTA: A les comandes *resolvectl query* ... es pot afegir el paràmetre *--legend=no* per a què no es mostrin les metadades de la resposta i només es vegi a pantalla les dades estrictament demanades.

El client DNS *dig*:

De clients DNS n'hi ha per tot arreu: quasi qualsevol aplicació de xarxa porta integrada a dins un client DNS: des d'un navegador fins la comanda *ping* han de primer preguntar a un servidor DNS la IP corresponent al nom introduït per l'usuari per poder començar a realitzar la seva tasca. No obstant, existeixen clients DNS "per se" genèrics (és a dir, no especialitzats en preguntar a *systemd-resolved* com és *resolvectl*) que ens permeten enviar peticions DNS més personalitzades a servidors DNS triats específicament i estudiar així amb més profunditat les respostes DNS rebudes. Exemples de clients DNS són la comanda *host*, la comanda *nslookup* o la comanda *dig* (més completa i la que provarem ara).

La comanda *dig*, pertanyent al paquet "dnsutils" (a Ubuntu) o "bind-utils" (a Fedora) és un client DNS pur; permet obtenir informació emmagatzemada als servidors DNS que preguntem (sobre el domini que preguntem) d´una forma directa. Funciona així:

dig [@ip.serv.DNS] nomDNS [tipusRegistre] [opcions]

on

"*@ip.serv.Dns*" és la IP del servidorDns al que es preguntarà; es pot indicar qualsevol servidor DNS públic disponible, com ara els del nostre ISP, els d'alguna altra empresa (per exemple els oferts per Google -8.8.8.8 o 8.8.4.4-, per Cloudflare -1.1.1.1 o 1.0.0.1-, etc) o qualsevol dels llistats a http://public-dns.info: són tots equivalents. Si no s´especifica cap, s´utilitzarà el primer configurat al sistema.

"*nomDNS*" generalment és el nom de la màquina complet (és a dir, el nom seguit del domini -el que s'anomena FQDN-) del qual es vol saber la informació del tipus indicat a "tipusRegistre". En algunes consultes concretes (com ara les fetes per esbrinar el valor dels registres de tipus NS o MX), no obstant, aquest nom es correspon al domini en sí.

"*tipusRegistre*" és un valor que pot ser qualsevol dels tipus de registre DNS estandaritzats: "A", "AAAA", "CNAME", "TXT", "NS", "MX"... Si no es posa cap, s´interpreta un tipus "A".

NOTA: També es pot especificar com a tipus de registre “AXFR”, obtenint així una transferència de zona.

"*opcions*": són un conjunt d'opcions de *dig* que serveixen per modificar la recerca a fer. Per exemple:

+tcp : s´utilitza TCP (i no UDP) per fer la petició
+short : es vol obtenir un resum de la informació rebuda (similar a +nostats +nocomments)
+multiline: el contrari de +short
+trace : mostra tots els passos recursius que el servidor emprat farà per donar-nos la resposta
+norecurse: es vol fer una petició no recursiva
+nocomments : no es vol obtenir comentaris extres a la informació rebuda
+nocmd : no es vol obtenir el comentari inicial que indica la versió de dig
+noquestion : no es vol obtenir la secció de Question a la informació rebuda
+nostats : no es vol obtenir estadístiques sobre la informació rebuda
+noall +answer : només s´obté la secció d´Answer a la informació rebuda, i res més.
+time=3 : s´esperarà tres segons a rebre la resposta abans de tornar-ho a intentar (o desistir)
+tries=3 : es farà un màxim de tres intents de consulta

Un exemple de petició DNS seria aquesta: *dig @8.8.8.8 www.hola.com CNAME +short +tcp +time=3* , el significat de la qual es deixa al lector com a exercici.

D'altra banda, en el cas de voler realitzar consultes inverses (és a dir, preguntar quina és la IP associada a un FQDN conegut -o el que és el mateix, el valor del registre PTR), la sintaxis és: *dig -x [@ip.serv.DNS] ip.ip.ip.ip [opcions]* (fixeu-vos en la presència del *paràmetre -x*).

Intentem desxifrar la resposta que ens dóna la comanda. Per exemple, si preguntem pel host "www.ignside.net", obtenim:

```
; <<>> DiG 9.4.0-(Hawk)-8.02 <<>> www.ignside.net
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 1580
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;www.ignside.net.               IN      A
;; ANSWER SECTION:
www.ignside.net.        2886    IN      CNAME   irvnet.nexenservices.com.
irvnet.nexenservices.com. 6486  IN      CNAME   sauterne.nexen.net.
sauterne.nexen.net.     486     IN      A       217.174.203.10
;; Query time: 15 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Thu Jul 28 22:33:36 2005
;; MSG SIZE  rcvd: 116
```

Les respostes que comencen per ";" són comentaris introduïts per *dig*, no provenen del servidor DNS. En les dues primeres línies, *dig* es limita a informar de la versió del programa en execució i del domini objecte de consulta. La línia "*;; global options: printcmd*" es refereix a les opcions generals usades en la consulta. Pots evitar aquestes dues línies usat l'opció *+nocmd*

La següent secció "Got Answer" ens ofereix detalls de la consulta rebuda, entre ells, el nombre de respostes rebudes, i si aquesta ens l'ha donat o no una servidor amb autoritat en el domini sol·licitat. Les banderes (flags) ens donen detalls de la consulta i resposta: "**QR**" (Query/Response) serveix per diferenciar la consulta de la resposta. "**RD**" (Recursion Desired), és una modalitat de la consulta, que és replicada o no en la resposta amb la bandera "**RA**" (Recursion Allowed), i significa que demanem al servidor que si no pot resoldre la resposta per si mateix, consulti recursivament a un altre servidor. L'acceptació de la petició pel servidor és opcional; "**AA**" significaria que la resposta és d'un servidor autoritatiu. Altres flags són: "**TC**" (Truncated Response), que vol dir que la resposta s'ha fraccionat per ser de major grandària del permès, "**AD**" (Authentic Data) i "**CD**" (Checking Disabled) tenen a veure amb l'extensió DNSSEC, etc

La tercera secció ens dóna detalls de la consulta; a més de mostrar el FQDN consultat, ens informa que estem preguntant sobre el valor dels registres A (és a dir, estem preguntant quina és la IPv4 associada a aquest FQDN; si no s'indica cap registre concret, serà aquest el que se sol·licitarà). Com ja sabem, si indiqués AAAA en el seu lloc voldria dir que estem preguntant sobre el valor dels registres AAAA (és a dir, sobre la IPv6 associada a l'FQDN consultat), i així.

En l'exemple que hem posat més amunt, la resposta té diverses línies: la línia "www.ignside.net. 2886 IN CNAME irvnet.nexenservices.com" ens diu que el nom pel qual preguntem és un àlies (CNAME) de "irvnet.nexenservices.com"; la següent línia ens indica que "irvnet.nexenservices.com" és un àlies de "sauterne.nexen.net", i la tercera línia ens indica, gràcies a observar el valor del registre A, que l'IPv4 de "sauterne.nexen.net " és 217.174.203.10

El que cap de les respostes que hem obtingut sigui AUTHORITY no diu res sobre la fiabilitat de la resposta, simplement diu que el servidor que ens l'ha donat no és responsable del domini. Finalment podem trobar-nos amb una secció de respostes addicionals, que típicament ens informaria de les IPs dels servidors DNS retornats a la secció AUTHORITY, si n'hi hagués.

La darrera secció ens explica el temps que ha trigat a resoldre la consulta, el nombre de bytes de la resposta, la data i el servidor DNS consultat.

NOTA: Per més informació sobre usos de la comanda *dig* i exemples, consulteu https://www.madboa.com/geek/dig
NOTA:També existeix el client *drill*, alternativa a *dig*. Altres clients DNS no tan versàtils com *dig* però que poden ser d'utilitat, tal com ja hem dit, són *host* o *nslookup*. A continuació es mostra un parell d'exemples del seu ús bàsic (on opcionalment es pot indicar el servidor DNS a preguntar -si no s'indica cap, s'usarà el configurat al sistema-); la seva sortida és la IP (o IPs equivalents) associades al nom indicat i també els "àlies" que té aquest nom:

nslookup nomDNS [ip.serv.DNS]
host nomDNS [ip.serv.DNS]

Systemd-resolved i el protocols mDNS+DNS-SD/LLMNR+SSDP

El dimoni *Systemd-resolved*, a més de servir com a "resolver-cache" centralitzat de peticions DNS, també implementa altres protocols alternatius de resolució de noms, com és la parella de protocols mDNS + DNS-SD o també la parella de protocols similars però alternatius LLMNR + SSDP (aquests darrers són més habituals a sistemes Windows). Tant una com l'altra parella de protocols possibiliten dues coses, en ordre:

1.-Que, mitjançant el protocol mDNS, aquesta màquina es pugui autoassignar un nom (el qual serà igual al seu nom local -és a dir, el valor indicat dins del fitxer "**/etc/hostname**" però sense el seu eventual domini perquè per contra se li afegirà automàticament el domini "**.local**") i seguidament informar d'aquesta autoassignació a la resta de màquines que facin servir aquest protocol per a què aquestes dades es reconeguin com a vàlides per tothom que pertanyi a la xarxa local **169.254.0.0/16** (en relació a aquesta xarxa, llegiu els següents paràgrafs). D'aquesta manera, s'aconsegueix tenir una xarxa local capaç d'emprar noms automàticament, sense haver de mantenir cap servidor DHCP/DNS central (ni fer servir cap arxiu "/etc/hosts").

NOTA: La "m" de mDNS prové de "Multicast". Aquest tipus de tràfic és similar al de tipus "broadcast" en el sentit de què un sol paquet arriba a diversos destinataris però, a diferència del tràfic "broadcast", al "multicast" una màquina s'ha de subscriure explícitament a un "grup de multidifusió" abans d'obtenir dades. Les màquines pertanyents a aquest grup podran veure les preguntes i les respostes dels altres (a través del port 5353/udp) i aprendre d'ells, treballant en aquest cas com a servidor DNS distribuït sense autoritat central.
NOTA: Les IPs de la xarxa 169.254.0.0/16 no són enrutables; això vol dir que estan limitades a fer-se servir en una LAN perquè els routers no traspassen paquets que les continguin (com a origen i/o destí)

2.-Que, mitjançant el protocol DNS-SD (de "DNS-Service Discovery"), cada màquina informi a la resta de màquines pertanyents a la xarxa 169.254.0.0/16 dels serveis que ofereix. A sistemes Systemd, aquests serveis (SSH, HTTPD, etc) es publiquen gràcies a l'existència fitxers de text anomenats "*.dnssd" (ha d'haver un per cada servidor que volguem publicar), els quals han d'estar ubicats a les carpetes "**/usr/lib/systemd/dnssd**" o "**/etc/systemd/dnssd**" (més informació tot seguit).

NOTA: Com sempre que parlem de Systemd, tots els fitxers de configuració inclosos sota "/usr/lib" i sota "/etc" es mesclen indistintament ordenant-se pel seu nom i la darrera directiva homònima s'aplica. En el cas de tenir dos fitxer homònims, l'ubicat a "/etc" sobreescriu a l'ubicat a "/usr/lib".
NOTA: El protocol SSDP (alternativa al DNS-SD) és un dels utilitzats també dins de l'estàndar UPnP -juntament amb d'altres per realitzar altres tasques, com el HTTPU-, el qual a la seva variant més famosa, el "profile" DLNA, permet comunicar dispositius multimèdia entre sí sense cap intervenció de l'usuari

No obstant, per a què tant mDNS com LLMNR puguin funcionar, primer cal haver-se produït un procés automàtic d'autoassignació d'IPs. És a dir, cal que la màquina s'hagi autoassignat de forma autònoma una direcció IP pseudoaleatòria (que serà sempre de la xarxa 169.254.0.0/16 -si és IPv4- o fe80::/10 -si és IPv6- i sempre després d'haver comprovat primer que ningú de la LAN no tingui ja per casualitat aquesta mateixa direcció). Aquesta capacitat d'autoassignació d'IP és el que se sol anomenar informalment "**IPv4LL**" (a Unix) o APIPA (a Windows) i permet no haver de tenir a la LAN cap servidor DHCP funcionant ni de configurar IPs estàtiques. Per tal d'activar el "IPv4LL" en una tarja de xarxa en sistemes Systemd, només cal afegir la línia *LinkLocalAddressing="yes" (o "ipv4" o "ipv6"* si ho volem restringir només a un tipus d'IP ; per defecte val "ipv6") dins de la secció *[Network]* del fitxer "*.network" corresponent a la tarja de xarxa en qüestió i reiniciar el servei "systemd-networkd".

NOTA: Una tarja pot tenir una IP aconseguida via "IPv4ALL" i a la vegada tenir-ne altres IPs diferents obtingudes d'una altra forma (fixes, dinàmiques, etc), no hi ha cap problema.

Per tal d'activar mDNS en sistemes Systemd cal fer dues coses: assegurar-se de tenir la línia *MulticastDNS=yes* a l'arxiu general de configuració de *systemd-resolved* ("/etc/systemd/resolved.conf") i, a més, assegurar-se de tenir la mateixa línia dins de la secció *[Network]* del fitxer "*.network" corresponent a la tarja de xarxa en qüestió. Alternativament, també hi podria haver la línia *LLMNR=yes* (als dos llocs també) per aconseguir el mateix objectiu fent servir el protocol alternatiu.

Per tal de publicar a la xarxa mitjançant DNS-SD la presència d'algun servei de manera que la resta de màquines de la "subxarxa mDNS" se n'assabentin de la seva existència, en sistemes Systemd cal crear, tal com hem dit, un arxiu "*.dnssd" dins de (preferiblement) la carpeta "/etc/systemd/dnssd" que tingui un contingut similar al següent:

[Service]
Name=nomServei
Type=_http._tcp
Port=nºport
TxtText=clau1=valor1 clau2=valor2 clau3=valor3

La línia "Name" pot tenir un valor qualsevol però admet valors concrets especials, com ara *%H* (equivalent al hostname de la màquina), *%v* (idèntic a la sortida de la comanda *uname -r*), etc (per més informació veure *man systemd.dnssd*). D'altra banda, els valors possibles per la línia "Type" venen especificats a l'estàndar RFC 6763 i alguns exemples podrien ser: *_ssh._tcp* , *_nfs._tcp* , *_ftp._tcp* , *_ipp._tcp*, etc. Finalment, les parelles clau<->valor escrites a la línia "TxtText" representen informació addicional relativa al servei en qüestió; per exemple, una parella com *path=/portal/index.html* en un servidor HTTPD serviria per informar de quina és la pàgina web publicada; una parella com *path=/data/shared* en un servidor NFS serviria per informar de quina és la carpeta compartida, un parella com *t=temperature_sensor* serviria per informar de la naturalesa de la informació oferida, etc En el cas de què el valor sigui binari, s'haurà de codificar prèviament mitjançant Base64 i s'haurà de fer servir la línia "TxtData" en comptes de "TxtText", així: *TxtData= dades=YW55IGJpbmFyeSBkYXRhCg==*

Un cop publicat un servei DNS-SD, amb la comanda *resolvectl* podrem comprobar si el podem trobar, executant per exemple: *resolvectl service nomMaquina._http._tcp.local*

El servei "Avahi" és una altra implementació (anterior a "Systemd-resolved") dels protocols mDNS i DNS-SD per sistemes Linux (equivalent a la implementació "Bonjour" a OS X o, respectivament, als protocols LLMNR i SSDP a Windows), a més del protocol "IPv4ALL". En el cas de Avahi, totes les màquines de la LAN que tinguin funcionant el servei (*systemctl status avahi-daemon*) pertanyeran automàticament a la mateixa "subxarxa" 169.254.0.0/16 i compartiran el domini de difusió multicast pel tràfic mDNS. D'altra banda, els serveis DNS-SD es publiquen gràcies a l'existència fitxers de text anomenats "*.services" (ha d'haver un per cada servidor que volguem publicar) i que han d'estar ubicats dins de la carpeta "/etc/avahi/services". La creació manual d'aquests fitxers no és gaire complicada però, afortunadament, Avahi ofereix uns quants fitxers llestos per utilitzar dins de la carpeta "/usr/share/doc/avahi". Per tant, per tal d'informar a la xarxa que a una màquina concreta hi ha (per exemple) un servidor SSH en marxa, el més fàcil és copiar el fitxer "/usr/share/doc/avahi/ssh.service" a la seva carpeta "/etc/avahi/services" i reiniciar el servidor *avahi-daemon*.

EXERCICIS:

1.- Sabem que els noms de les màquines remotes es poden obtenir de diferents maneres (fitxer "/etc/hosts", servidors DNS, servidors mDNS,...). ¿Què signifiquen les paraules *files*, *mdns4*, *resolve* i *dns* que apareixen a la línia *hosts:* ... del fitxer "/etc/nsswitch.conf" i què implica l'ordre en què allà hi apareixen?

2.-a) Ja sabem quina informació conté l'arxiu "/etc/hosts" però...¿i l'arxiu "/etc/networks"? (pots consultar l'arxiu que hi ha a la teva màquina real i la seva pàgina del manual, si cal)

b) Utilitza el paràmetre *-S* de la comanda *dpkg* (a Ubuntu) o *-qf* de la comanda *rpm* (a Fedora) per esbrinar en quin paquet instal.lat vénen els arxius "/etc/hosts", "/etc/hostname", "/etc/nsswitch.conf" i "/etc/networks". Seguidament, utilitza el paràmetre *-L* de la comanda *dpkg* (a Ubuntu) o *-ql* de la comanda *rpm* (a Fedora) per esbrinar quins altres arxius contenen aquests paquets-

3.-a) Arrenca una màquina virtual (tant se val, Ubuntu o Fedora) que tingui almenys una tarja de xarxa en mode "adaptador pont". Utilitza la comanda *hostnamectl set-hostname* per assignar a aquesta màquina el nom local "maquina**VIRT***ElTeuNom*.asix2.com". Comprova que el nom ha canviat mirant el contingut de l'arxiu "/etc/hostname" (o executant la comanda *hostnamectl status*).

b) Associa en l'arxiu "/etc/hosts" de la màquina virtual el nom "maquina**REAL***ElTeuNom*.asix2.com" a la IP de la teva màquina real.

bII) Associa també en l'arxiu "/etc/hosts" de la màquina virtual el nom "maquina**VIRT***ElTeuNom*.asix2.com" a la IP 127.0.0.1 (afegint-lo als altres noms que hi puguin haver ja associats, com ara "localhost", etc).

> NOTA: Si no fas aquest apartat hi haurà programes que donaran errors estranys. D'altra banda, el fet d'escriure un FQDN en comptes d'un nom curt és degut al que s'explica a la "NOTA IMPORTANT" de l'exercici 9

c) Executa a la màquina virtual la comanda *ping maquina**REAL**ElTeuNom.asix2.com* ¿Què passa i per què? ¿I si executes *ping maquina**REAL**ElTeuNom*, què passa i per què? ¿I si executes *ping maquina**VIRT**ElTeuNom.asix2.com* a la màquina real, què passa i per què?

Un mètode per navegar més segur consisteix en usar llistes negres que inclouen llocs maliciosos que realitzen "**tracking**" -també anomenat "**profiling**"- (és a dir, un seguiment dels nostres costums de navegació per vendre aquesta informació a agències de publicitat) o fins i tot que serveixen "**malware**" (és a dir, programes que infecten el nostre sistema, els quals, segons el tipus, poden deixar inutilitzat aquest o bé fer-lo servir com a "zombi" per atacs remots o bé poden robar dades sensibles, o espiar les accions de l'usuari, etc) o fan "**phishing**" (és a dir, suplantar un web oficial per un altre impostor i així poder, per exemple, robar dades personals com ara nº de tarja de crèdit, introduïdes allà per usuaris que no descobreixen el parany).

Si no es vol fer servir software de tercers (software antimalware, plugins "adblockers" al navegador, etc) es pot implementar un sistema de "blacklisting" artesà simplement fent servir l'arxiu "/etc/hosts" del nostre sistema simplement assignant a cada nom de lloc maliciós una IP inabastable (com 127.0.0.1 o 0.0.0.0). L'inconvenient d'aquest sistema és que no té granularitat (és a dir, només es poden bloquejar domini complets) i que és necessari actualitzar "a mà" regularment el fitxer (o bé utilitzar un script que automatitzi el procés agafant les actualitzacions des de la font o fonts escollides). D'altra banda, es té l'avantatge de no dependre d'un servei extern i es té el control absolut de les entrades a la llista; tampoc hi ha un retard perceptible a l'hora d'accedir a llocs mitjançant navegador encara que el fitxer hosts tingui milions d'entrades.

> NOTA: Una altra opció seria fer consultes a servidors DNS que incloguin aquestes característiques de protecció (els quals redireccionen a una advertència explicativa quan bloquegen l'accés a un lloc i/o permeten aplicar filtres de continguts personalitzats); d'aquesta manera no haurem de mantenir la "llista negra" (però es té menys control). Alguns exemples són:
>
> *Comodo Secure DNS: 8.26.56.26 i 8.20.247.20
> *Dyn Internet Guide: 216.146.35.35 i 216.146.36.36
> *FoolDNS: 87.118.111.215 i 213.187.11.62
> *GreenTeam Internet: 81.218.119.11 i 209.88.198.133
> *OpenDNS: 208.67.222.222 i 208.67.220.220. Per més: 208.67.222.123 i 208.67.220.123
> *Norton ConnectSafe: 199.85.126.10 i 199.85.127.10.
> Per pornografia: 199.85.126.20 199.85.127.20. Per més: 199.85.126.30 i 199.85.127.30

4.-a) A una màquina virtual qualsevol (Ubuntu o Fedora, tant se val) que tingui almenys una tarja de xarxa en mode "adaptador pont", assegura't de que la seva configuració (IP, porta d'enllaç, servidor DNS) sigui l'estàndar de qualsevol màquina de l'aula (és a dir, que no tingui cap possible configuració feta per nosaltres dels exercicis anteriors). Descarrega't allà un dels següents arxius hosts prefabricats amb molts llocs "sospitosos"...:

> https://someonewhocares.org/hosts/zero/hosts
> https://winhelp2002.mvps.org/hosts.txt
> http://pgl.yoyo.org/as/serverlist.php?showintro=0;hostformat=hosts (obtingut de https://pgl.yoyo.org/adservers)
> https://github.com/mitchellkrogza/ultimate.hosts.blacklist
> https://github.com/StevenBlack/hosts

NOTA: A https://filterlists.com pots trobar una "metallista" de moltes llistes de dominis potencialment perillosos (aquestes llistes poden venir en diversos formats, entre els quals el format de l'arxiu "hosts", per aprofitar-les en diversos programes d'"ad-block, tallafocs, etc). Més "metallistes" són https://blocklist-tools.developerdan.com/blocklists o https://firebog.net

...i escull un domini de la llista per escriure'l al navegador (que no tingui configurat cap proxy); observa si pots accedir-hi. ¿Què passa llavors si substitueixes l'arxiu "/etc/hosts" del sistema pel fitxer descarregat: pots continuar accedint? ¿Què hi té a veure l'arxiu "/etc/nsswitch.conf" en tot això?

b) Escriu a l'arxiu "/etc/hosts" la IP de "www.redhat.com" i associa-la al nom "www.apple.com". ¿Què passa llavors si escrius aquest nom a la barra del navegador (que no tingui configurat cap proxy) i s'hi vol accedir?

5.-Consulta el contingut de l'arxiu "/etc/systemd/resolved.conf" a una màquina qualsevol (real o virtual) i, amb l'ajuda dels apunts de teoria i la que t'ofereix *man resolved.conf* digues per a què serveixen les següents línies de configuració del servei *systemd-resolved* que allà apareixen: *DNS= , FallbackDNS= , Domains=* i *Cache= i ReadEtcHosts=*

NOTA: Les opcions *LLMNR=* i *MulticastDNS=* , també presents a l'arxiu "/etc/systemd/resolved.conf", tenen a veure amb els protocols LLMNR i mDNS estudiats a la teoria descrita en paràgrafs anteriors d'aquest document. Més endavant hi ha un exercici on es demanarà profundir-hi. D'altra banda, les opcions *DNSOverTLS=* i *DNSSEC=*, també presents a l'arxiu "/etc/systemd/resolved.conf", les estudiarem més endavant, en un exercici específic.

6.-Arrenca una màquina virtual qualsevol (Ubuntu o Fedora, tant se val) però ara amb dues tarjes: la primera (enp0s3) en mode "adaptador pont" i la segona (enp0s8) en mode "xarxa interna". En el cas d'Ubuntu, esborra tots els arxius ubicats sota la carpeta "/etc/netplan" i tot seguit executa la comanda *netplan apply*. Finalment, crea un arxiu ".network" que assigni a la tarja enp0s8 la IP 192.168.123.123/24, la porta d'enllaç 192.168.123.1, els servidors DNS 1.1.1.1 i 1.0.0.1 i el domini d'enrutament "redhat.com"; finalment reinicia el servei "systemd-networkd". A partir d'aquí:

a) Executa *resolvectl dns* ¿D'on provenen els valors mostrats a la secció "Global" (si n'hi aparegués algun)? ¿I els mostrats a la secció corresponent a la tarja en mode "xarxa interna"? ¿I els mostrats a la secció corresponent a la tarja en mode "adaptador pont"? En tot cas, ¿quins valors tenen preferència: els particulars d'una tarja concreta o els globals?

b) Executa *resolvectl domain* i dedueix, a partir de la sortida que obtens, per quina tarja s'enviaran (en primera instància) les peticions de resolució DNS pel domini "redhat.com". ¿I per qualsevol altre domini?

aib) Dedueix, a partir de les respostes dels apartats anteriors, quin serà el servidor DNS que farà servir llavors el teu sistema en primera instància per saber la IP de "www.redhat.com" i de "www.marca.com"

aII) ¿Què fa la comanda *sudo resolvectl dns enp0s3 1.1.1.1* ? Comprova-ho executant *resolvectl dns*

bII) ¿Què fa la comanda *sudo resolvectl domain enp0s3 "hola.com"* ? Comprova-ho executant la comanda *ping www* i també *resolvectl domain*

c) Executa *resolvectl query www.hola.com* ¿Quina informació et mostra? ¿I si executes *resolvectl -t CNAME query www.hola.com* ? Compara la sortida anterior amb la sortida de la comanda *host www.hola.com*, ¿quines diferències veus? ¿Quina informació se't mostra si executes ara *resolvectl query 104.83.70.70*? ¿I si executes *resolvectl -t NS query hola.com* ? ¿I si executes *resolvectl -t MX query hola.com* ?

d) Executa *resolvectl statistics* i fixa't en el tamany de la catxé DNS i la quantitat d'encerts ("hits") i fallades ("misses"). Ara esborra la catxé DNS amb *resolvectl flush-caches* i seguidament torna a observar el seu tamany i la quantitat de "hits" i "misses". ¿Què ha passat?

dII) Després de llegir això: *"The time-to-live (TTL) is the number of seconds that client and intermediary DNS resolvers can cache and reuse the same DNS responses without having to make another lookup. Longer cache times means fewer repeated (and most of the time redundant) lookups and fewer lookups means better performance."* ¿què creus que explica aquest article: https://www.ctrl.blog/entry/dns-client-ttl.html ?

e) ¿Quina informació et dona la comanda *resolvectl default-route*?

7.-Segueix utilitzant la mateixa màquina virtual de l'exercici anterior. En el cas de què sigui de tipus Server, fes l'apartat *a)*; en el cas de que sigui de tipus Desktop/Workstation fes l'apartat *aII)* -ambdós són equivalents, només que en el primer cas s'usa Systemd-Networkd per omplir de contingut la configuració DNS a gestionar per Systemd-resolved i en el segon cas s'usa NetworkManager per fer el mateix-:

a) Crea l'arxiu "/etc/systemd/network/lerele.network" amb el següent contingut (i reinicia el servei "systemd-networkd"):

[Match]
Name=enp0s3
[Network]
DHCP=yes
DNS=208.67.220.123

aII) Vés a l'apartat de "Xarxes" del panell de control de Gnome i allí clica sobre el botó amb la icona de la roda dentada corresponent a la tarja enp0s3; en el quadre que apareix, clica sobre la pestanya "Ipv4" i afegeix la IP 208.67.220.123 a la caixa de text titulada "DNS". Un cop fet això, desactiva i torna a activar la tarja de xarxa en qüestió per tal de què aquest canvi tingui efecte (això ho pots fer simplement clicant sobre les opcions del menú "Desactiva" i "Connecta" respectivament).

La IP indicada als apartats anteriors (208.67.220.123) es correspon a un dels servidor d'OpenDNS, una empresa que ofereix serveis de resolució de noms que incorporen "llistes negres" de dominis, molt útils per fer control parental, per exemple. Aquest servidor DNS en concret té incorporada una llista negra de dominis bàsicament pornogràfics. Això vol dir que una màquina que faci servir aquest servidor no podrà accedir a aquest tipus de webs perquè el servidor DNS no li resoldrà (i, per contra, anirà a parar a un servidor propi advertint-lo del fet).

b) Una manera de comprovar si estem fent servir el servidor d'OpenDNS (és a dir, si tenim efectiva la prohibició de resoldre llocs pornogràfics) és anant a https://welcome.opendns.com i veure si obtenim el missatge d'"OK". Fes-ho. Hauries de veure que no l'estàs fent servir. Dedueix per què observant la sortida de la comanda *resolvectl dns*

Ara farem els canvis necessaris a la configuració del sistema per a què el servidor d'OpenDNS sí es tingui en compte finalment. Per això, en el cas de què sigui el sistema on estiguis treballant sigui de tipus Server, fes l'apartat *c)*; en el cas de que sigui de tipus Desktop/Workstation fes l'apartat *cII)* -ambdós són equivalents, només que en el primer cas s'usa Systemd-Networkd per omplir de contingut la configuració DNS a gestionar per Systemd-resolved i en el segon cas s'usa NetworkManager per fer el mateix-:

c) Afegeix al final de l'arxiu "/etc/systemd/network/lerele.network" les següents línies noves, sense modificar les anteriors (i reinicia el servei "systemd-networkd"):

[DHCPv4]
UseDNS=no

cII) Vés a l'apartat de "Xarxes" del panell de control de Gnome i allí clica sobre el botó amb la icona de la roda dentada corresponent a la tarja enp0s3; en el quadre que apareix, clica sobre la pestanya "Ipv4" i desactiva el botó etiquetat com "Automàtic" associat a la caixa de text titulada "DNS". Un cop fet això, desactiva i torna a activar la tarja de xarxa en qüestió per tal de què aquest canvi tingui efecte (això ho pots fer simplement clicant sobre les opcions del menú "Desactiva" i "Connecta" respectivament).

d) Torna a anar a https://welcome.opendns.com i observa si ara obtens el missatge d'"OK". Hauries de veure que ara sí l'estàs fent servir. Dedueix per què observant la sortida de la comanda *resolvectl dns* ¿Què passa si intentes accedir ara al lloc pornogràfic "www.redtube.com" des d'un navegador? ¿I si hi fas *ping*, què veus?

8.-A la màquina real (o a una màquina virtual qualsevol amb una tarja de xarxa en mode "adaptador pont"), fes servir la comanda *dig* per fer les consultes DNS següents. Concretament:

a) Troba la direcció IP (ó IPs!) de la màquina www.pamapam.org. ¿Quin servidor DNS t'ha respost i per què? ¿Quin significat té la secció "Question Section" mostrada a la sortida de la comanda *dig*?

b) Troba la mateixa direcció IP (ó IPs!) de la màquina www.pamapam.org però preguntant ara directament a un altre servidor d'Internet com per exemple el 1.1.1.1 de CloudFlare o qualsevol dels altres que apareixen a http://public-dns.info Confirma mirant la sortida del *dig* que, efectivament, s'estigui contestant el servidor que li has indicat. ¿La resposta és la mateixa? Per què?

c) Escriu la IP trobada a la consulta anterior a la barra del navegador per intentar accedir a la web corresponent d'aquesta forma, sense fer servir noms DNS. ¿Què veus? ¿Per què?

cII) Troba ara la direcció IP (o IPs!) de la màquina www.rebelion.org i un cop trobada/es escriu-la/es a la barra del navegador per intentar accedir a la web corresponent d'aquesta forma. ¿Què veus ara? ¿Per què?

d) Utilitza la IP de www.pamapam.org per fer ara una consulta inversa. ¿Obtens el mateix nom www.pamapam.org? ¿I si fas una consulta inversa a partir de la IP de www.rebelion.org , què obtens?

e) Intenta trobar les direccions Ipv6 de la màquina "www.pamapam.org" . ¿Quina resposta obtens?
f) Esbrina quins àlies té el nom www.hola.com. Prova si els àlies que hagis trobat són equivalents escrivint-los a la barra de direccions d'un navegador

g) ¿Què et diu la resposta a la consulta *dig hola.com NS*? ¿I la resposta a la consulta *dig com NS* ? ¿I la resposta a la comanda *dig . NS* ? ¿Quina diferència hi ha entre la comanda anterior i *dig . NS +short* ?

h) ¿Què et mostra la comanda *dig @a.root-servers.net www.hola.com +norec*? ¿I la comanda *dig @f.gtld-servers.net www.hola.com +norec* ? ¿I la comanda *dig @a18-64.akam.net www.hola.com +norec* ?

i) Prova les següents opcions del *dig* amb una consulta qualsevol i digues per a què serveixen: *+trace* , *+noall +answer* (les dues juntes), *+noques*, *+nostats*, *+nocmd* (les tres juntes o per separat).

Recordem ara els següents conceptes de teoria:

a) Un servidor DNS de tipus cau és un servidor DNS que no conté cap informació autoritativa ell per sí mateix sinó que es limita a remetre les peticions que li fan a un altre servidor DNS, el qual sí serà qui respongui. La gràcia llavors és que el servidor DNS de tipus cau, com diu el seu nom, es guardarà la resposta obtinguda a la seva memòria de forma que una altra petició posterior ja la podrà respondre ell mateix sense haver de tornar a preguntar. Les respostes es guarden en memòria cau un temps configurable.

b) Un servidor DNS autoritatiu, en canvi, és un servidor DNS que controla un determinat domini perquè conté la informació explícita respecte la relació que hi ha entre les IPs i els noms de les màquines que formen el domini en qüestió. Tothom que vulgui saber alguna part d'aquesta informació (clients, altres servidors DNS, etc) haurà de preguntar més tard o més d'hora a un servidor DNS autoritatiu perquè és l'únic origen canònic.

A partir d'aquí, i en aquest sentit, "Systemd-resolved" pot ser configurat per a què:

*Escolti peticions de clients DNS remots. Això es fa indicant a l'arxiu "resolved.conf" la línia (o línies, perquè n'hi pot haver més d'una): *DNSStubListenerExtra=ip.tarja:nºport* (on "ip.tarja" representa la IP de la tarja de xarxa de la màquina per on es voldran rebre peticions remotes, por ser "0.0.0.0"; i "nºport és el port d'escolta (si no s'indica, per defecte serà el 53 tcp i udp; per indicar només un protocol, cal escriure la cadena "tcp:" o "udp:" abans de la IP indicada). D'aquesta manera, el nostre servei "Systemd-resolved" podrà donar servei cau de noms centralitzat als clients remots de la xarxa.

NOTA: Si es vol desactivar el canal local per on "Systemd-resolved" rep les peticions de la nostra pròpia màquina (no té molt de sentit, però es pot fer), caldrà llavors escriure a més la línia *DNSStubListener=false*

*Per tots els clients admesos (el local i/o els remots), fer de servidor DNS autoritatiu del contingut present a l'arxiu "/etc/hosts" local. Això és possible si es té indicada la directiva *ReadEtcHosts=yes* de l'arxiu "resolved.conf" (per defecte ja és així)

NOTA: Dnsmasq (http://www.thekelleys.org.uk/dnsmasq/doc.html) és un programa més especialitzat en aquestes tasques, ja que és un servidor DHCP(+TFTP/PXE)+DNS de tipus cau (via consulta als servidors DNS remots especificats al seu arxiu "/etc/resolv.conf" local) que també pot funcionar com servidor DNS autoritatiu del contingut present a l'arxiu "/etc/hosts" local però amb moltes més opcions que no pas té "Systemd-resolved"

9.-a) En una màquina virtual (Ubuntu o Fedora, tant se val) que tingui almenys una tarja de xarxa en mode "adaptador pont", afegeix al seu arxiu "/etc/hosts" una línia que associ la IP 1.2.3.4 amb el nom "pepe.asix2.com"

NOTA IMPORTANT: Cal tenir en compte que per a què el *systemd-resolved* dels clients reconegui els noms de les màquines demanades com de tipus DNS (en comptes de noms de tipus LLMNR) i per tant, faci la consulta al servidor DNS pertinent i no a cap altra cosa que no sigui DNS, aquests hauran de ser sempre plenament qualificats (és a dir, hauran de ser FQDN). Això vol dir que *systemd-resolved* no resol via DNS els noms simples "sense domini" (a no ser que hi hagi un domini *search* definit al client). Aquest comportament és important tenir-lo en compte també si volem muntar un servidor DNS autoritatiu fent servir els noms escrits a l'arxiu "/etc/hosts": els noms que han d'aparèixer en aquest arxiu hauran de ser sempre, degut a aquest motiu, plenament qualificats

b) Modifica la configuració del servei "Systemd-resolved" per tal de què escolti peticions provinents de la xarxa i per a què pugui actuar com a servidor autoritatiu de la "zona" definida a l'arxiu "/etc/hosts" local. Pista: als paràgrafs blaus anteriors s'explica com fer-ho.

c) Observa a la màquina virtual la sortida de la comanda *sudo ss -tulnp* i confirma que el servidor "Systemd-resolved" escolti al port 53 a través de la xarxa i que també estigui igualment escoltant al port 53 només en local, com fins ara.

d) Executa a la màquina real la comanda *dig @ip.maq.virtual pepe.asix2.com* ¿Què veus? ¿Per què, en canvi, la comanda *dig pepe.asix2.com* no mostra res (o, el que és el mateix, ¿per què la comanda *ping pepe.asix2.com* no funciona)? ¿Com es podria solucionar aquest problema des de la pròpia configuració del servei "Systemd-networkd"?

e) Executa a la màquina real la comanda *dig @ip.maq.virtual www.cooperativa.cat* ¿Què passa? ¿Per què?

f) ¿Què explica aquest article: https://www.ctrl.blog/entry/homenet-domain-name.html?

10.- Arrenca una màquina virtual qualsevol (Ubuntu o Fedora, tant se val) que tingui dues tarjes de xarxa: la primera (enp0s3) en mode "adaptador pont" i la segona (enp0s8) en mode "xarxa interna". Elimina els arxius "*.network" que hagis pogut crear en aquesta mateixa màquina en exercicis anteriors. Reinicia el servei "systemd-networkd" i tot seguit comprova que no tinguis cap direcció IP definida a la tarja en mode "xarxa interna" (això ho pots fer per exemple amb la comanda *networkctl --all -n 0 status*) A partir d'aquí:

a) Activa el suport al protocol MulticastDNS per part de "systemd-resolved". Això ho pots fer establint a *yes* (o *true*) el valor de la línia *MulticastDNS=* dins de l'arxiu "/etc/systemd/resolved.conf" (i descomentant-la!). No t'oblidis de reiniciar el dimoni "systemd-resolved" un cop fet aquest canvi. Confirma que ho hagis fet bé observant el valor de les línies adients a de l'apartat "global" de la sortida de la comanda *resolvectl status*

> NOTA: Per defecte el servei mDNS escolta al port 5355/UDP (això ho pots comprovar executant *ss -tlnp*, per exemple). Confirma, doncs, que el tallafocs del sistema deixi obert aquest port

b) Fes que la màquina virtual autoassigni manualment una IP de la xarxa 169.254.0.0/16 a la seva tarja enp0s8 i que aquesta tingui habilitat explícitament el reconeixement del protocol MulticastDNS. Per fer ambdues coses, has de crear un fitxer anomenat (per exemple) "/etc/systemd/network/lerele.network" amb aquest contingut:

```
[Match]
Name=enp0s8
[Network]
LinkLocalAddressing=ipv4
MulticastDNS=yes
```

Després d'haver reiniciat el servei "systemd-networkd", comprova, passats uns segons, que la màquina tingui de cop una IP d'aquesta xarxa. Confirma que la tarja enp0s8 "comprén" el protocol mDNS executant *resolvectl status* (o, alternativament, *resolvectl mdns*) i observant el valor de les línies adients a l'apartat específic de la tarja en qüestió.

c) Apaga la màquina virtual on estaves treballant i tot seguit crea-li un clon. Arrenca ara les dues màquines alhora i comprova que totes dues tenen a la seva tarja enp0s8 respectiva una IP diferent de la xarxa 169.254.0.0/16 i que, per tant, es poden comunicar entre elles (prova-ho fent "pings" entre elles usant aquestes IPs recentment autoassignades).

d) Comprova (amb la comanda *hostnamectl*) que el nom local de cada màquina (és a dir, el seu "hostname") sigui diferent. Si no és així (que no serà perquè són clons), canvia'ls (amb la comanda *hostnamectl set-hostname* ...). Tot seguit torna a provar de fer "pings" entre elles però ara no fent servir la IP de l'altre extrem sinó el seu nom mDNS (el qual esta format sempre, recordem, pel seu "hostname" treient-li el seu eventual domini i afegint-hi el domini ".local"; és a dir, si el "hostname" d'una màquina fos, per exemple, "pepe.elmeudomini.net", el seu nom mDNS corresponent seria "pepe.local").

dII) ¿Què fa la comanda *resolvectl -p mdns query nomAltraMaquina.local* executada des d'una de les dues màquines (fent referència a l'altra)?

11.-a) Descarrega https://github.com/farrokhi/dnsdiag/releases/download/v2.0.2/dnsdiag-2.0.2.linux-x86_64-bin.tar.gz a la màquina real i descomprimeix-lo. Tot seguit vés a un terminal, situa't dins de la carpeta resultat de la descompressió, executa les següents comandes i digues què fan (consulta la seva documentació per esbrinar el significat dels seus paràmetres):

aII) *./dnsping -c 3 -t A -s 8.8.8.8 www.redhat.com*

aIII) *./dnseval -c 3 -t A -f llistaDNS.txt www.redhat.com* , on "llistaDNS.txt" és un fitxer el contingut del qual ha de ser un llistat d'IPs (una per línia) corresponents als servidors DNS que volem provar, com per exemple: 8.8.8.8, 8.8.4.4, 1.1.1.1, 1.0.0.1, 4.2.2.1, 4.2.2.2, 4.2.2.3, 4.2.2.4, 4.2.2.5, 209.244.0.3, 209.244.0.4, 192.168.12.10, 195.46.39.39, 195.46.39.40)

> NOTA: La columna "TTL" mostrada a la sortida de *dnseval* indica el temps suggerit que la resposta DNS es mantindrà en catxé al client per tal de què aquest no torni a fer cap petició sobre el mateix domini mentre no s'arribi al final d'aquest temps. En aquest sentit, recordeu l'article https://www.ctrl.blog/entry/dns-client-ttl.html mencionat a l'exercici 6 per veure com segons el client usat, aquest valor "TTL" es té en compte (o no!)
> NOTA: La columna "flags" indica les característiques de les respostes DNS rebudes. Per conèixer el significat de cada "flag" consulteu https://www.iana.org/assignments/dns-parameters/dns-parameters.xhtml#dns-parameters-12

NOTA: Una comanda similar a *dnsping* és *dnsmeter* (https://github.com/DNS-OARC/dnsmeter), proporcionada pel projecte DNS-OARC, el qual ofereix més eines de diagnòstic DNS a la seva web (https://www.dns-oarc.net/oarc/tools i, sobre tot, https://www.dns-oarc.net/oarc/software)
NOTA: A Ubuntu es pot instal·lar el paquet "dnsdiag" en comptes d'haver de descarregar-ho manualment. D'altra banda, la seva pàgina web és https://dnsdiag.org

b) ¿Què explica aquest article: https://www.ctrl.blog/entry/dotblog-tld-performance.html?

c) Vés a les següents pàgines i explica quin servei ofereix cadascuna d'elles:

* https://www.dnsperf.com , i més concretament:
  https://www.dnsperf.com/#!dns-resolvers
  https://www.dnsperf.com/#!dns-root-servers
  https://www.dnsperf.com/dns-provider/cloudflare
  https://www.dnsperf.com/dns-speed-benchmark

* https://atlas.ripe.net/results/maps/root-server-performance
* https://atlas.ripe.net/dnsmon
* https://atlas.ripe.net/results/maps/comparative-dns-root-rtt

* http://dig.ping.pe

* https://dnsmap.io (o alternativament, https://dnschecker.org o https://www.whatsmydns.net)

NOTA: Similar als serveis anteriors és l'ofert per https://www.perfopos.net però sota registre (gratuït). En especial les eines "Analytics->CDN/DNS/Cloud Providers" i "Analytics->DNS Resolvers"
NOTA: Unes eines similars a les oferides per "Dnsperf" però enfocades al rendiment dels servidors "Common Delivery Network" (CDN) són les següents (sobre les CDNs en parlarem més endavant):
https://www.cdnperf.com
https://www.cdnperf.com/cdn-compare
https://www.cdnperf.com/cdn-provider/akamai-cdn
https://www.cdnperf.com/tools/cdn-latency-benchmark
https://www.cdnperf.com/tools/cdn-calculator

d) ¿Quina informació mostra https://dailychanges.domaintools.com ?

NOTA: Una pàgina similar però un xic més tècnica és https://www.statdns.com

12.- a) ¿Què mostra l'opció "Reverse IP Lookup" de https://viewdns.info si escrius "marca.es"? ¿I si escrius "ns1.linode.com" a l'opció "Reverse NS Lookup" ? D'altra banda, ¿quin tipus d'informació obtens si escrius "xeill.net" a l'opció "DNS Report"?

NOTA: Altres webs similars (però no exactament iguals...recomano la seva consulta també) són https://www.ipaddress.com, http://www.dnsstuff.com/tools , https://hackertarget.com/ip-tools , https://www.ultratools.com, https://major.io/icanhazip-com-faq , https://www.yougetsignal.com , https://ping.eu o https://mxtoolbox.com/NetworkTools.aspx

b) ¿Les opcions "Reverse IP" i "Reverse NS" de https://dnslytics.com/tools retornen el mateix resultat que les opcions homònimes de l'apartat anterior?

NOTA: Aquesta web té moltes altres eines interessants, com ara "Reverse TinyURL" o "MAC Address Lookup" entre d'altres. Des d'aquí s'anima a provar aquestes i totes les que et cridin l'atenció)

13.-a) Llegeix amb calma aquest article i digues quins diferents objectius persegueix el protocol DNSSEC i el protocol DNS over TLS (DoT): https://blog.apnic.net/2018/08/20/dnssec-and-dns-over-tls

b) Llegeix *man systemd.network* i digues què significa el valor "allow-downgrade" de la directiva *DNSSEC=* i el valor "opportunistic" de la directiva *DNSOverTLS=*, ambdues de l'arxiu "/etc/systemd/resolved.conf" (o de qualsevol arxiu .network, on hi tindria preferència)

Una alternativa a DoT és DoH ("DNS over HTTPS"). Aquest altre protocol fa servir l'encriptació proporcionada ja "de sèrie" per HTTPS per utilitzar-lo com a canal a través del qual enviar les peticions DNS necessàries (i rebre'n les respostes). Està especialment pensat per ser usat en navegadors, que ja fan servir HTTPS per defecte. Cal tenir en compte, però, no tots els servidors DNS admeten aquest tipus de canal per establir comunicació amb els clients; és per això que els navegadors incorporen una llista (petita) de servidors DNS coneguts compatibles amb el protocol DoH, d'entre els quals es pot escollir el que desitgem utilitzar. Cal tenir llavors en compte, però, que el navegador en qüestió no farà servir pas els servidors DNS definits a nivell de sistema sinó els indicats a la seva configuració particular, de forma totalment independent a la resta de programes del nostre sistema!.

c) Desfés tot rastre de configuració que puguis tenir encara al sistema en relació als servidors d'OpenDNS realitzada a l'exercici 7; comprova-ho observant amb *resolvectl dns* que no aparegui la IP 208.67.220.123. A partir d'aquí, vés al menú "Preferències->General" del Firefox i allà pulsa el botó "Paràmetres de xarxa" que apareix al final de tot; al quadre que apareix activa l'opció "Habilita DNS sobre HTTPS" i al desplegable associat titulat "Utilitza el proveïdor", indica el valor "Personalitzat"; finalment, a la caixa de text que apareix titulada "Personalitzat" escriu la següent url: *https://doh.familyshield.opendns.com/dns-query* Amb això estàs indicant que vols que Firefox faci servir com a servidor DNS justament el mateix servidor 208.67.220.123 (però en comptes d'oferir-se com a servidor DNS pla, ho farà com a servidor DoH). Per tant, a partir d'ara, tot i que la comanda *resolvectl dns* continui mostrant la mateixa sortida que al principi, si vas a qualsevol pàgina pornogràfica (com per exemple "www.redtube.com") amb el Firefox, hauràs de tornar a veure la pàgina de "censura" de l'OpenDNS. Prova-ho.