EXERCICIS configuració de xarxa amb *systemd-nspawn*

1.-a) Torna a entrar en el contenidor "fedora.raw" fent servir el seu propi Systemd. Dins d'aquest contenidor observa (amb la comanda *ip*, que haurà d'estar instal.lada) la direcció IP de les diferents tarjes de xarxa i la porta d'enllaç que té el contenidor. ¿Què veus? Prova (amb la comanda *ping*, que haurà d'estar instal.lada) si el contenidor té, efectivament, connexió de xarxa.

> NOTA: Que comparteixin les mateixes IPs màquina amfitriona i contenidor és una conseqüència de que, per defecte, el contenidor se "n'aprofita" de la configuració de xarxa de la màquina amfitriona (o dit d'una manera més tècnica, que comparteixen el "namespace" de xarxa. Això, òbviament, podria canviar-se, tal com veurem en propers apartats

b) Surt d'aquest contenidor i torna-hi a entrar però ara afegint a la comanda *systemd-nspawn -bi fedora.raw* un nou paràmetre anomenat *--private-network* . Observa (amb *ip -c a*) què li passa a la configuració de xarxa del contenidor (pots consultar també la pàgina del manual de *systemd-nspawn*).

c) Surt d'aquest contenidor i torna-hi a entrar però ara afegint a la comanda *systemd-nspawn --private-network -bi fedora.raw* un nou paràmetre anomenat *--network-interface=enp0s3*. Un cop dins del contenidor, observa (amb *ip -c a*) què li passa a la configuració de xarxa del contenidor (pots consultar també la pàgina del manual de *systemd-nspawn*) i també observa què li passa ara a la configuració de xarxa de la màquina amfitriona.

cII) Configura la tarja *enp0s3* del contenidor per tal de poder tenir Internet (llegeix la "nota" inferior per saber com fer-ho). Comprova-ho que ho has aconseguit fent per exemple *ping www.hola.com* o *dnf upgrade*. Seguidament, instal.la-hi el paquet "nmap-ncat" (ho necessitaràs per més endavant).

> NOTA: Els passos per configurar ràpidament una tarja de xarxa (com *enp0s3*) recordem que són els següents:
> *Activar la tarja: *ip link set up dev enp0s3*
> *Assignar-li una IP: *ip address add 192.168.18.X/24 dev enp0s3*
> *Assignar-li una porta d'enllaç: *ip route add default via 192.168.18.10 dev enp0s3*
> *Assignar un servidor DNS al sistema: *resolvectl dns enp0s3 8.8.8.8*
>
> NOTA: En comptes d'assignar un servidor DNS temporalment a la tarja *enp0s3* amb la comanda *resolvectl dns* com mostra la nota anterior, una alternativa hagués sigut executar la següent comanda: *sed -i "s|#DNS=|DNS=8.8.8.8|" /etc/systemd/resolved.conf* per tal de fixar la configuració DNS global de forma permanent. Ambdues maneres funcionaran, però, només si tenim el servei "systemd-resolved" funcionant dins el contenidor (cosa que serà així simplement havent instal·lat el paquet "systemd-resolved" -a Ubuntu ja és així de sèrie-, ja que el servei automàticament s'activarà per defecte dins del contenidor)
>
> NOTA: Si volem aprofitar l'arxiu "resolv.conf" que ja tinguem funcionant al sistema amfitrió per a què sigui l'utilitzat dins d'un contenidor, *systemd-nspawn* ofereix el paràmetre *--resolv-conf=*... Concretament, si escrivim *--resolv-conf=copy-host* , el fitxer "resolv.conf" de l'amfitrió es copiarà dins del contenidor i si escrivim *--resolv-conf=bind-host* , el fitxer "resolv.conf" del contenidor serà un enllaç al de l'amfitrió (tècnicament, serà un punt de muntatge –bind). És recomanable usar "copy-host" si es preveu que el contenidor pugui realitzar canvis en la seva pròpia configuració DNS (desmarcant-se de la configuració de l'amfitrió); si no és aquest el cas, llavors "bind-host" és preferible ja que aquest mode no permet que el contenidor pugui canviar el seu propi arxiu "/etc/resolv.conf" (en estar muntat de forma read-only)

Existeixen altres opcions similars a *--network-interface* però que, en comptes de fer "desaparèixer" la tarja de xarxa de la màquina amfitriona per "posar-la" al contenidor, són "menys agressives" ja que permeten crear dins del contenidor una tarja de xarxa virtual i connectar-la, com si estiguessin en un switch (o router), a una tarja de xarxa determinada la màquina anfitriona. D'aquesta manera, la màquina amfitriona no perd cap tarja i el contenidor pot disfrutar d'una connexió de xarxa pròpia. Aquestes opcions són, entre d'altres, *--network-macvlan=nomTarjaReal* o *--network-ipvlan=nomTarjaReal*. Cadascuna d'elles internament funcionen de forma diferent (bàsicament, les tarjes "macvlan" actuen a nivell 2 de la capa OSI com els switchos i les tarjes "ipvlan" actuen a nivell 3 com els routers) però de forma superficial a la pràctica són similars. A continuació provarem aquestes dues i també els enllaços "veth".

> NOTA: Més tècnicament, les tarjes virtuals "macvlan" són creades sobre una determinada tarja real identificant-se cadascuna amb una MAC diferent: cada paquet que arribi a la tarja real serà demultiplexat a la tarja "macvlan" adient segons la direcció MAC de destí que porti. En canvi, les tarjes virtuals "ipvlan" són creades sobre una determinada tarja real amb la mateixa MAC totes, distingint-se entre sí per la seva IP: cada paquet que arribi a la tarja real serà demultiplexat a la tarja "ipvlan" adient segons la direcció IP de destí que porti. Això fa que, no obstant, la màquina amfitriona no es pugui comunicar amb un contenidor amb la seva tarja en mode "ipvlan" tot i tenir IPs de la mateixa xarxa: el fet de compartir MAC fa que els paquets ARP no es transmetin bé entre elles i això fa que la taula ARP dels dos extrems no s'ompli correctament (i no es puguin veure, doncs).

d) Surt del contenidor que tinguis en marxa i executa ara la comanda *systemd-nspawn –private-network --network-ipvlan=enp0s3 -bi fedora.raw* (el paràmetre *-b* aquí és important per poder utilitzar el servei "systemd-resolved" del contenidor!). Un cop dins del contenidor, observa (amb *ip -c a*) què li passa a la configuració de xarxa del contenidor i configura la tarja "ipvlan" que hi haurà aparegut per tal de poder tenir Internet (llegeix la "nota" de l'apartat anterior per recordar com fer-ho). Comprova-ho que ho has aconseguit fent per exemple *ping www.hola.com* o *dnf upgrade*. No apaguis el contenidor encara.

> NOTA: Només tindràs Internet al contenidor si la màquina virtual amfitriona té la seva tarja en mode "adaptador pont". Això és perquè la tarja IPVLAN ha de pertànyer a la mateixa xarxa que la de l'aula sense intermediaris: si la tarja de la màquina virtual fos NAT, per exemple, tindria una IP de la xarxa 10.0.2.0/8 que faria de barrera entre la tarja de la màquina real (que és qui en definitiva al final accedeix a l'exterior) i la tarja IPVLAN (estariem separant dos costats d'una xarxa posant en mig una xarxa diferent)

dII) Comprova quina és la direcció MAC de la tarja "ipvlan" del contenidor. Seguidament, obre un terminal independent al sistema amfitrió i observa quina és la direcció MAC de la tarja real. ¿Coincideixen? ¿Per què? (pots llegir els paràgrafs de teoria previs a l'apartat d) per esbrinar-ho) ¿Què té veure això amb que no es puguin fer "ping" entre la màquina amfitriona i el contenidor tot i tenir IPs de la mateixa xarxa? (pots llegir la nota al final dels paràgrafs de teoria). Finalment, apaga el contenidor.

e) Torna a repetir l'apartat anterior però ara evitant haver de tornar a fer tots els passos manuals per configurar la tarja de xarxa "ipvlan" del contenidor. És a dir, entra en el contenidor ràpidament amb *systemd-nspawn -bi fedora.raw* i crea dins d'ell el fitxer "/etc/systemd/network/iv-enp0s3.network" amb el següent contingut...:

```
[Match]
Name=iv-enp0s3
[Network]
Address=192.168.12.222/24
Gateway=192.168.12.10
DNS=8.8.8.8
```

...i surt del contenidor i ara entra-hi fent servir la tarja "ipvlan" tal com vas fer a l'apartat anterior, així: *systemd-nspawn –private-network --network-ipvlan=enp0s3 -bi fedora.raw* Hauries de veure com la tarja "iv-enp0s3" ja es troba automàticament configurada i amb connexió a Internet. Comprova-ho. Finalment, surt del contenidor.

> NOTA: Recorda que és especialment important arrencar el contenidor amb el paràmetre -b per a què s'executi Systemd (i DBus) dins del contenidor. Si no, ni *systemd-networkd* ni cap altre servei relacionat amb Systemd funcionaran allà dins.

2.-a) Sabent que un dispositiu "veth" és un conjunt de dues tarjes ethernet virtuals connectades entre si de forma que el que surt per una arriba a l'altra, torna a entrar al contenidor executant ara la comanda *systemd-nspawn –private-network --network-veth -bi fedora.raw* Seguidament executa *ip -c a* tant a la màquina amfitriona com a dins del contenidor. ¿Quines tarjes de xarxa noves veus? (llegeix la següent "nota" per entendre el perquè dels seus noms respectius)

> NOTA: Si el nom del contenidor és "foo", el nom de la interfície Ethernet virtual de l'amfitrió serà "ve-foo" i el nom de la interfície Ethernet virtual del contenidor serà "host0". En examinar les interfícies amb *ip link show*, els noms de les interfícies es mostraran amb un sufix, com ara "ve-foo@if2" i "host0@if9". El "@ifN" no forma part del nom de la interfície però *ip link* afegeix aquesta informació per indicar a quina “ranura” es connecta el cable Ethernet virtual a l’altre extrem. Per exemple, una interfície Ethernet virtual d'amfitrió que es mostra com "ve-foo@if2" es connectarà al contenidor foo i dins del contenidor a la segona interfície de xarxa, la que es mostra amb l'índex 2 quan s'executa *ip link* dins del contenidor. Similarment, al contenidor, la interfície anomenada "host0@if9" es connectarà a la 9a ranura de l'amfitrió.
> NOTA: En comptes dels paràmetres *--private-network --network-veth* es pot escriure simplement el paràmetre equivalent *-n*

aII) Assigna (amb la comanda *ip address add...*) una direcció IP qualsevol (però de la mateixa xarxa) a cadascuna de les dues tarjes noves que han aparegut en fer l'apartat anterior (tant la de la màquina amfitriona com la del contenidor, respectivament), i activa-les (amb la comanda *ip link set up...*) . Fes *ping* entre elles. ¿Què passa? D'altra banda, ¿el contenidor té connexió a Internet? ¿Per què? Surt del contenidor.

NOTA: Respecte aquest apartat és molt interessant llegir la secció de la pàgina del manual de *systemd-nspawn* que parla sobre l'existència dels fitxers "/usr/lib/systemd/network/80-container-ve.network" i "/usr/lib/systemd/network/80-container-host0.network" , els quals configuren automàticament les dades de xarxa dels contenidors mitjançant DHCP i, per tant, no cal assignar IP a mà (no obstant, aquests fitxer només s'apliquen si tant al contenidor com a la màquina amfitriona funciona *systemd-networkd*)

b) Executa ara la comanda *systemd-nspawn -p tcp:2000:3000 -nbi fedora.raw* (fixa't en el paràmetre -n!) A partir d'aquí:

NOTA: Si no s'especifica el protocol al paràmetre *-p*, s'assumeix "tcp". Si no s'indica el port del contenidor, s'assumeix que és el mateix que l'indicat per la màquina amfitriona

1.-Executa les següents comandes al contenidor:

ip link set up dev host0
ip address add 192.168.55.2/24 dev host0
ncat -l -p 3000

2.-Obre un terminal a la màquina amfitriona i executar-hi allà les següents comandes:

sudo ip link set up dev ve-fedora
sudo ip address add 192.168.55.1/24 dev ve-fedora
nc 192.168.55.1 2000

¿Què passa? Per què?

NOTA: Aquest apartat es podria haver fet amb qualsevol altre servei (Apache2, etc) en comptes del Netcat

c) Executa la comanda *systemd-nspawn --network-zone=lalala -nbi fedora.raw* i, en un altre terminal de la màquina amfitriona, la comanda *systemd-nspawn --network-zone=lalala -nbi ubuntu.raw* Amb això estaràs ficant els dos contenidors en una mateixa zona, que és el mateix que dir que els has comunicat mitjançant un "switch" virtual per fer una "xarxa interna". Observa com a la màquina amfitriona han aparegut vàries interfícies de xarxa: una anomenada "vz-lalala", que representa el "switch" que uneix els diferents contenidors (i que desapareixerà quan l'últim contenidor pertanyent a la zona s'aturi) i tantes tarjes com contenidors hi hagi dins de la zona (anomenades "vb-nomContenidorRespectiu", on "vb" vol dir "veth bridge"); als contenidors continuaran apareixen les mateixes interfícies veth "host0" ja conegudes. Assigna (amb la comanda *ip address add...*) una direcció IP qualsevol (però de la mateixa xarxa) a cada tarjes "host0" dels contenidors respectius i activa-les (amb la comanda *ip link set up...*), a més d'activar la pròpia interfície "vz-lalala". Fes *ping* d'un contenidor a l'altre i des de la màquina amfitriona a qualsevol dels contenidors. ¿Què passa? Finalment, surt de tots els contenidors.

NOTA: Respecte aquest apartat és molt interessant llegir la secció de la pàgina del manual de *systemd-nspawn* que parla sobre l'existència dels fitxers "/usr/lib/systemd/network/80-container-vz.network" i "/usr/lib/systemd/network/80-container-host0.network", els quals configuren automàticament les dades de xarxa dels contenidors mitjançant DHCP i, per tant, no cal assignar IP ni porta d'enllaç a mà (no obstant, aquests fitxer només s'apliquen si tant al contenidor com a la màquina amfitriona funciona *systemd-networkd*)