EXERCICIS *machinectl*

1.-a) Executa ara la comanda *systemd-nspawn -bi fedora.raw* (el paràmetre *-b* aquí és important). Mantenint en marxa aquest contenidor, obre un terminal independent a la màquina amfitriona i executa-hi allà la comanda *machinectl list*. ¿Què veus?

> NOTA: La comanda *machinectl* és capaç de comunicar-se només amb contenidors que tinguin el servei especial D-Bus (a més del propi Systemd) funcionant al seu interior. Això s'aconsegueix, com ja sabem, indicant el paràmetre -b a *systemd-nspawn.* D'altra banda, tant D-Bus com Systemd són automàticament instal.lats a totes les imatges generades per *mkosi* independentment de com s'arrenquin posteriorment amb *systemd-nspawn* (amb -b o sense). En el cas d'utilitzar una altra eina per generar imatges (com ara directament *debootstrap, dnf –installroot*, etc), caldrà instal.lar D-Bus (i Systemd) a mà.

b) ¿Què veus si executes, sempre en el terminal de la màquina amfitriona, *machinectl status fedora* ?

c) Consulta el manual de *systemctl* per esbrinar per a què serveix el seu paràmetre -M. ¿Què passa, per tant, si executes la comanda *systemctl -M fedora status systemd-journald* (has de ser "root")? ¿I *systemctl -M fedora list-units --type=service* ?

> NOTA: Molts altres executables de Systemd tenen el paràmetre -M, com ara *journalctl, hostnamectl, systemd-analyze*,etc

d) ¿Què passa si executes *sudo machinectl poweroff fedora* a la màquina amfitriona?

> NOTA: Un sinònim d'aquesta comanda és *machinectl stop nomContenidor*. D'altra banda, també existeix la comanda *machinectl terminate nomContenidor,* però aquesta atura el contenidor indicat de forma abrupta. D'altra banda, també existeix la comanda *machinectl reboot nomContenidor*

2.-a) Amb el/s contenidor/s apagat/s, en un terminal de la màquina amfitriona executa la comanda *machinectl list-images* ¿Per què aquesta comanda no mostra res? (la resposta a aquesta pregunta està relacionada amb el significat de la carpeta "/var/lib/machines"; consulta a la pàgina del manual de *machinectl* la secció "Files and directories" per esbrinar-ho i també l'apartat corresponent a la subcomanda *list-images*)

aII) Executa la comanda *sudo machinectl import-raw ...* per a què tant "ubuntu.raw" com "fedora.raw" apareguin, ara sí, a la sortida de *machinectl list-images*

> NOTA: Si el sistema amfitrió fos un Fedora, la comanda anterior potser no funciona. Per a què funcioni (i també la resta de comandes d'aquest exercici) prèviament has d'executar (com administrador) la comanda *setenforce permissive* a la màquina amfitriona, la qual relaxa un mecanisme de seguretat anomenat SELinux responsable de bloquejar la gestió de contenidors per part de *machinectl* i que està activat per defecte a sistemes Fedora
> NOTA: També podries haver optat per copiar simplement els fitxers "raw" a dins de la carpeta "/var/lib/machines" i ja està (o fins i tot, creant-hi un enllaç simbòlic ja n'hi hauria prou)
> NOTA: A més de la subcomanda *import-raw* també existeix *import-tar* (si les imatges estan en aquest format) i també *import-fs* (si les imatges estan en format "directory")

b) Executa ara *sudo machinectl start ubuntu* ¿Quina diferència visible observes ara respecte si haguessis iniciat el contenidor amb *systemd-nspawn* (pots llegir la nota següent com a pista)?

> NOTA: Una de les diferències clau entre *systemd-nspawn* i *machinectl start* és que la primera inicia un contenidor directament en primer pla i la segona ho fa en segon pla (havent d'utilitzar *machinectl login* per accedir al seu shell com veurem al proper apartat). Això fa que la segona sigui molt més adient per posar en marxa serveis dins de contenidors, tal com veurem en els exercicis següents. D'altra banda, *machinectl start* només funciona amb contenidors ubicats dins de "/var/lib/machines"
> NOTA: En realitat, la comanda *machinectl start ubuntu* és equivalent a *systemctl start systemd-nspawn@ubuntu.service.* Igualment, la comanda *machinectl stop ubuntu* és equivalent a *systemctl stop systemd-nspawn@ubuntu.service.*
> Per tant, si volguéssim canviar algun aspecte de la manera d'iniciar els contenidors mitjançant *machinectl start,* caldria editar l'arxiu de configuració del service genèric, que és */usr/lib/systemd/system/systemd-nspawn@.service*
> (allà es poden canviar, per exemple, els paràmetres concrets passats per defecte a la comanda *systemd-nspawn* invocada per *machinectl start* via paràmetre *ExecStart=*, com són els parametres -b, -n, -U,... entre d'altres aspectes).

NOTA: És interessant saber que existeix una manera d'iniciar un contenidor (via *machinectl start...*) sense haver de ser root. Resulta que la comanda *machinectl* no és més que un client D-Bus d'un servei anomenat "systemd-machined", el qual és qui realment realitza la feina d'iniciar, parar, etc els contenidors (per tant, la comanda *machinectl* no funcionarà mai si aquest servei està apagat). Doncs bé, resulta que aquest servei utilitza el sistema de seguretat Polkit per autentificar els usuaris que en voler fer ús. Més en concret, al fitxer "/usr/share/polkit-1/actions/org.freedesktop.machine1.policy" trobem les accions concretes (moltes amb una relació directa amb un verb de *machinectl*) a les que podem establir restriccions, com ara "org.freedesktop.machine1.login", "org.freedesktop.machine1.shell", "org.freedesktop.machine1.manage-machines" (útil per *start* i *stop*) o "org.freedesktop.machine1.manage-images", entre d'altres. Així doncs, l'únic que caldrà fer per a què poguem realitzar l'acció escollida sense haver de ser administrador és canviar a "yes" (o "auth_self") el valor de la línia <allow_active> associada a aquesta acció (valor que segurament per defecte serà "auth_admin").

c) ¿Què passa si executes (en un altre terminal) *sudo machinectl login ubuntu*?

NOTA: La comanda anterior només funciona si el contenidor s'ha iniciat prèviament amb *machinectl start* i no amb *systemd-nspawn*. Això és perquè *machinectl login* reclama per a sí un terminal exclusiu, i com que *systemd-nspawn* en genera un d'interactiu, entra en competència. D'altra banda, *machinectl login* obliga, per funcionar correctament, a què dins del contenidor s'estigui executant Systemd, però això no és problema perquè *machinectl start* compleix aquest requisit perquè utilitza internament el paràmetre *-b* de *systemd-nspawn*, tal com s'explica a la NOTA de l'apartat anterior).
NOTA: Si el contenidor és un Debian/Ubuntu, la comanda anterior potser no funciona. L'explicació tècnica es troba aquí: https://github.com/systemd/systemd/issues/852#issuecomment-127652768 però bàsicament, el que cal fer és afegir dins de l'arxiu "/etc/securetty" del contenidor una línia que posi *pts/0* (i per si de cas, més línies que posin *pts/1*, *pts/2*, etc)

d) Surt del contenidor amb "CTRL+]" i seguidament executa *machinectl status ubuntu* ¿Continua funcionant el contenidor?

e) ¿Què passa si executes la comanda *sudo machinectl shell root@ubuntu /bin/ls /etc* ?

NOTA: Si el contenidor es va arrencar fent servir directament *systemd-nspawn* en comptes de *machinectl start* , la comanda *machinectl shell* donarà un error similar a "Failed to get shell PTY" degut a què *systemd-nspawn* ja ofereix un shell interactiu per treballar-hi (impedint la connexió de qualsevol altre shell) mentre que amb *machinectl start* no hi ha aquest problema perquè inicia el contenidor en segon pla
NOTA: La gran diferència entre *machinectl login* i *machinectl shell* és que el primer prepara l'entorn getty per oferir l'opció d'iniciar sessió interactiva dins del contenidor amb un usuari a escollir en aquest moment, mentre que el segon ja entra directament al shell (interactiu) predefinit per l'usuari indicat a la comanda

f) ¿Què veus si executes *machinectl image-status ubuntu* ? ¿I si executes *machinectl show-image ubuntu* ?

g) ¿Què passa si executes *sudo machinectl copy-to ubuntu /etc/fstab /opt/fstab*? (pots consultar la pàgina del manual de *machinectl* o bé fer *machinectl shell root@ubuntu /bin/ls /opt* per comprovar-ho)

NOTA: La comanda anterior també pot copiar carpetes senceres, així *machinectl copy-to ubuntu /etc /opt/etc* , per exemple. D'altra banda, també existeix la comanda *machinectl copy-from ubuntu /opt/fstab /etc/fstab*, que fa la còpia de contenidor a host. En qualsevol cas, si el contenidor i host comparteixen l'espai d'UIDs (és a dir, *--private-users=false*) la propietat dels fitxers/carpetes copiades és preservada; si no, els fitxers/carpetes copiats pertanyeran a l'usuari i grup root (UID/GID 0)

h) ¿Què passa si executes *sudo machinectl clone ubuntu ubuntu2*? Prova-ho. ¿I si executes tot seguit *sudo machinectl rename ubuntu2 ubuntuNou* ? Prova-ho també.

i) ¿Què passaria si executessis *sudo machinectl read-only ubuntuNou true*? (no cal que ho provis)

j) Executa *sudo machinectl stop ubuntu* i comprova que, efectivament, el contenidor ja no estigui funcionant

k) ¿Per a què serviria la comanda *sudo machinectl enable ubuntu* (equivalent a *sudo systemctl enable systemd-nspawn@ubuntu.service*)? Executa-la. Per saber si ha funcionat hauràs de reiniciar la màquina amfitriona. Un cop comprovat, atura el contenidor amb *sudo machinectl stop ubuntu*

l) Elimina la imatge "ubuntu" de "/var/lib/machines" executant la comanda: *sudo machinectl remove ubuntu*

En el cas de voler executar la comanda *machinectl start ...* (o *machinectl enable ...*) és interessant disposar d'algun mètode per poder passar paràmetres personalitzats al contenidor en qüestió (com ara el mode de les seves tarja de xarxa, per exemple), i poder sobreescriure, per tant, els paràmetres per defecte que apareixen a la plantilla "/usr/lib/systemd/system/systemd-nspawn@.service". Per aconseguir això es pot crear un arxiu dins de la carpeta "**/etc/systemd/nspawn**" (o també dins de la carpeta "/var/lib/machines" però llavors cal afegir a systemd-nspawn el paràmetre *--settings=trusted* per a què el reconegui) anomenat igual que el contenidor i amb extensió ".nspawn" que tingui un contingut similar a aquest (per a més informació, consultar *man systemd.nspawn*...hi ha moltes opcions més, quasi tantes com paràmetres de *systemd-nspawn*):

[Exec]
Boot=yes #Si val "yes" (o "true" o "on") és equivalent a haver escrit el paràmetre ***-b*** de *systemd-nspawn*
Parameters=/bin/ls -l #Equivalent a haver escrit el paràmetre ***-a /bin/ls -l*** al final de *systemd-nspawn*
Environment=PATH=/bin #Estableix el valor d'una variable d'entorn al contenidor. Equivalent al paràmetre ***--setenv***
LimitXXX=xxx #Estableixen diferents límits (*LimitCPU,LimitFSIZE,LimitNOFile...*) pel contenidor. Equival a ***--rlimit***
ResolvConf=copy-host #Estableix com es gestionarà l'arxiu /etc/resolv.conf del contenidor. Equival a ***–resolv-conf***
PrivateUsers=yes #Equivalent a ***--private-users*** (útil quan val "no" per desactivar el *-U* que s'aplica amb *machinectl start*)
[Files]
ReadOnly=yes #Equivalent al paràmetre ***--read-only*** . Una altra opció similar és *Volatile*
Bind=/opt:/var/opt #Munta la carpeta /opt de l'amfitrió a la carpeta /var/opt del contenidor. Equival a ***--bind***
BindReadOnly=/opt:/opt #Similar a *Bind* però el muntatge és de només lectura. Equival a ***--bind-ro***
[Network]
Private=yes #Equivalent a ***--private-network***
Interface=enp0s3 #Equivalent a ***--network-interface***
MACVLAN=enp0s8 #Equivalent a ***--network-macvlan***
IPVLAN=enp0s8 #Equivalent a ***--network-ipvlan***
VirtualEthernet=yes *#Equivalent a* ***-n*** (útil quan val "no" per desactivar el *-n* que s'aplica amb *machinectl start*)
Port=udp:x:y #Equivalent a ***-p*** (normalment només s'empra en combinació amb *-n*)

NOTA: Una forma més "artesanal" de sobreescriure els valors que venen a la plantilla "systemd-nspawn@.service" per un determinat contenidor (l'anomenarem "ubuntu") és, en comptes de crear un fitxer *.nspawn dins de "/etc/systemd/nspawn", crear un arxiu anomenat com "/etc/systemd/system/systemd-nspawn@ubuntu.service.d/override.conf" (ja sigui manualment o bé via la comanda *systemctl edit systemd-nspawn@ubuntu*) amb un contingut similar a aquest:

[Service]
ExecStart=
ExecStart=/usr/bin/systemd-nspawn ....

3.- Crea dins de la carpeta "/etc/systemd/nspawn" un fitxer anomenat "fedora.nspawn" que faci que en iniciar el contenidor "fedora" amb *machinectl start/enable*...:

1.-Automàticament es munti la carpeta /var de la màquina amfitriona a la carpeta /opt del contenidor
2.-Que el mode de la tarja de xarxa del contenidor sigui el per defecte en no indicar cap paràmetre específic a *systemd-nspawn* (és a dir, *Private=no*)

...i, un cop iniciat, comprova que, efectivament, la seva carpeta "/opt" estigui poblada i que disposi d'una tarja de xarxa amb la mateixa configuració que la de la màquina amfitriona (comprova que, efectivament, el contenidor tingui connexió amb l'exterior). Atura el contenidor.