

SEGURIDAD EN SISTEMAS OPERATIVOS

4º Grado en Informática - Complementos de Ing. del Software

Curso 2018-19

Práctica 1. Administración de la seguridad en Linux

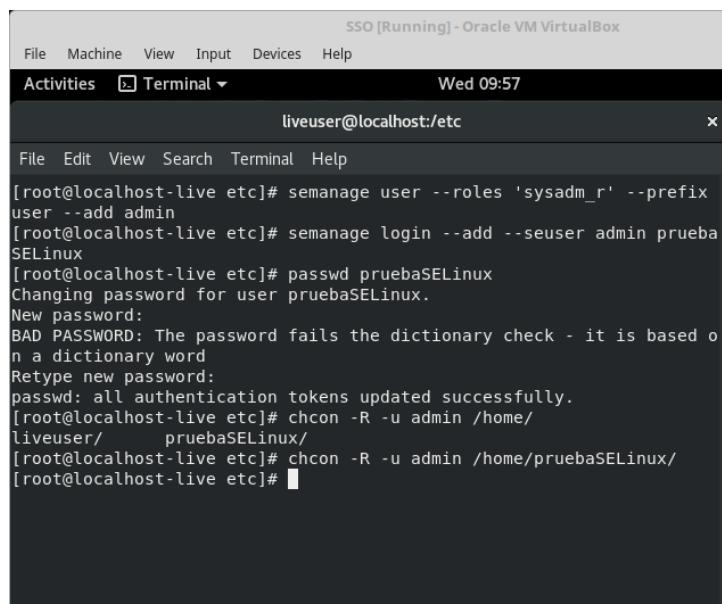
Sesión 4. SELinux (Security Enhanced Linux)

Autor¹: Víctor García Carrera

Ejercicio 1.

En esta práctica trabajamos con SELinux, viendo tanto la terminología como su estructura y funcionamiento. Comenzamos creando un nuevo usuario SELinux, diferente a un usuario típico de sistema (varios usuarios del sistema pueden tener un mismo usuario SELinux) que es utilizado para definir los roles que puede tomar un usuario.

Creamos un nuevo usuario del sistema llamado *pruebaSELinux* con el comando *useradd -m pruebaSELinux*, al cual le asociaremos el nuevo usuario SELinux que vamos a crear con el comando *semanage* llamado *admin*. De *admin* definiremos como rol *sysadm_r* en su contexto de seguridad. Este rol está diseñado para tareas de administración del sistema, por lo que es muy privilegiado. A continuación una captura de este proceso de creación y configuración de un usuario SELinux:



The terminal window shows the following session:

```
SSO [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal ▾ Wed 09:57
liveuser@localhost:etc
File Edit View Search Terminal Help
[root@localhost-live etc]# semanage user --roles 'sysadm_r' --prefix user --add admin
[root@localhost-live etc]# semanage login --add --seuser admin pruebaSELinux
[root@localhost-live etc]# passwd pruebaSELinux
Changing password for user pruebaSELinux.
New password:
BAD PASSWORD: The password fails the dictionary check - it is based on a dictionary word
Retype new password:
passwd: all authentication tokens updated successfully.
[root@localhost-live etc]# chcon -R -u admin /home/
liveuser/ pruebaSELinux/
[root@localhost-live etc]# chcon -R -u admin /home/pruebaSELinux/
[root@localhost-live etc]#
```

Primero creamos un usuario llamado *pruebaSELinux*. A continuación creamos el **usuario SELinux** llamado **admin** con el rol de **sysadm_r** que asociamos al nuevo *pruebaSELinux*. A este último le asignamos una contraseña y finalmente cambiamos el contexto de seguridad para su directorio home con *admin*. Podemos comprobar que efectivamente hemos creado y añadido al sistema el nuevo usuario SELinux *admin* asociado al usuario *pruebaSELinux* visualizando el contenido del archivo */etc/selinux/targeted*:

¹ Como autor declaro que los contenidos del presente documento son originales y elaborados por mi. De no cumplir con este compromiso, soy consciente de que, de acuerdo con la “Normativa de evaluación y de calificaciones de los estudiantes de la Universidad de Granada” esto “conllevará la calificación numérica de cero ... independientemente del resto de calificaciones que el estudiante hubiera obtenido ...”

```

SSO [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal ▾ Wed 09:59
liveuser@localhost:/etc/selinux/targeted
File Edit View Search Terminal Help
GNU nano 2.9.8 seusers
# This file is auto-generated by libsemanage
# Do not edit directly.

root:unconfined_u:s0-s0:c0.c1023
default :unconfined u:s0-s0:c0.c1023
pruebaSELinux:admin:s0

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^N Replace ^U Uncut Text ^T To Spell ^ ^ Go To Line

```

Ejercicio 2.

En este ejercicio vamos a tratar con los logs del sistema para entender los mensajes que SELinux escribe en los mismos. Estos mensajes muestran la actuación de SELinux sobre diversas acciones del sistema junto con información acerca del contexto de la misma (qué proceso realizaba la acción y sobre qué recurso...). Vamos a visualizar el log del demonio *auditd* localizado en */var/log/audit/audit.log*.

Primero utilizamos el comando *ausearch* para acceder al mismo y filtrar las entradas relacionadas con el AVC (Access Vector Cache), donde podemos además especificar que muestre las más recientes. El comando propuesto es */sbin/ausearch -m avc -ts recent*. Este comando no tiene una salida esperada pues nunca encuentra ninguna coincidencia. En esta situación recurrimos directamente a visualizar el contenido del log *audit.log* mediante el comando *sudo cat /var/log/audit/audit.log*.

```

SSO [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal ▾ Wed 13:17
liveuser@localhost:/etc
File Edit View Search Terminal Help
[liveuser@localhost-live etc]$ sudo /sbin/ausearch -m avc
<no matches>
[liveuser@localhost-live etc]$ sudo /sbin/ausearch -m avc -ts recent
<no matches>
[liveuser@localhost-live etc]$ sudo cat /var/log/audit/audit.log
type=DAEMON_START msg=audit(1541613026.038:8036): op=start ver=2.8.3 format=raw kernel=4.16.3-301.fc28.x86_64 auid=4294967295 pid=9
10 uid=0 ses=4294967295 subj=system_u:system_r:unconfined_u:system_r:unconfined_u:t:s0 op=a
dd_rule key=(null) list=1 res=1
type=CONFIG_CHANGE msg=audit(1541613026.233:84): auid=4294967295 ses=4294967295 subj=system_u:system_r:unconfined_u:system_r:unconfined_u:t:s0 op=a
dd_rule key=(null) list=1 res=1
type=SERVICE_START msg=audit(1541613026.233:85): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init_t:s0 msg='u
nit=auditd comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'
type=SYSTEM_BOOT msg=audit(1541613026.233:86): pid=931 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init_t:s0 msg='
comm="systemd-update-utmp" exe="/usr/lib/systemd/systemd-update-utmp" hostname=? addr=? terminal=? res=success'
type=SERVICE_START msg=audit(1541613026.243:87): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init_t:s0 msg='u
nit=systemd-update-utmp comm="systemd" exe="/usr/lib/systemd/systemd-update-utmp" hostname=? addr=? terminal=? res=success'
type=SERVICE_START msg=audit(1541613026.343:88): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init_t:s0 msg='u
nit=mcelog comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'
type=SERVICE_START msg=audit(1541613026.343:89): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init_t:s0 msg='u
nit=vboxservice comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'
type=SERVICE_START msg=audit(1541613026.363:90): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init_t:s0 msg='u
nit=dbus comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'
type=SERVICE_START msg=audit(1541613026.633:91): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init_t:s0 msg='u
nit=alsa-state comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'
type=SERVICE_START msg=audit(1541613026.643:92): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init_t:s0 msg='u
nit=rngd comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'
type=SERVICE_START msg=audit(1541613026.643:93): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init_t:s0 msg='u
nit=switcheroo-control comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'
type=SERVICE_STOP msg=audit(1541613026.643:94): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init_t:s0 msg='u
nit=switcheroo-control comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'

```

Vamos a generar algunos mensajes de denegación en este log para analizarlos. Este es el aspecto de *audit.log* antes de ello:

```
liveuser@localhost:~  
File Edit View Search Terminal Help  
ntors=pam_localuser,pam_unix acct="root" exe="/usr/bin/sudo" hostname=? addr=? terminal=/dev/pts/0 res=success'  
type=USER_AUTH msg=audit(1541612100.119:655): pid=14378 uid=1000 auid=1000 ses=2  
subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:authentication grantors=pam_succeed_if,pam_localuser,pam_unix acct="liveuser" exe="/usr/bin/sudo" hostname=? addr=? terminal=/dev/pts/0 res=success'  
type=USER_ACCT msg=audit(1541612100.119:656): pid=14378 uid=1000 auid=1000 ses=2  
subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:accounting grantors=pam_unix,pam_localuser acct="liveuser" exe="/usr/bin/sudo" hostname=? addr=? terminal=/dev/pts/0 res=success'  
type=USER_CMD msg=audit(1541612100.119:657): pid=14378 uid=1000 auid=1000 ses=2  
subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='cwd="/home/liveuser" cmd=636174202F7661722F6C6F672F61756469742F61756469742E6C6F67 terminal=pts/0 res=success'  
type=CRED_REFR msg=audit(1541612100.119:658): pid=14378 uid=0 auid=1000 ses=2 subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:setcred grantors=pam_localuser,pam_unix acct="root" exe="/usr/bin/sudo" hostname=? addr=? terminal=/dev/pts/0 res=success'  
type=USER_START msg=audit(1541612100.119:659): pid=14378 uid=0 auid=1000 ses=2 subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:session_open grantors=pam_keyinit,pam_limits,pam_keyinit,pam_limits,pam_systemd,pam_unix acct="root" exe="/usr/bin/sudo" hostname=? addr=? terminal=/dev/pts/0 res=success'  
[liveuser@localhost-live ~]$
```

Primero intentamos acceder al sistema con el recien creado usuario *pruebaSELinux* mediante el comando *su pruebaSELinux*. Introducimos una contraseña erronea y visualizamos *audit.log*, donde destaca el siguiente mensaje:

```
liveuser@localhost:~  
File Edit View Search Terminal Help  
ntors=pam_env,pam_localuser,pam_unix acct="root" exe="/usr/bin/sudo" hostname=? addr=? terminal=/dev/pts/0 res=success'  
type=USER_START msg=audit(1541612348.739:679): pid=14460 uid=0 auid=1000 ses=2 subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:session_open grantors=pam_keyinit,pam_limits,pam_keyinit,pam_limits,pam_systemd,pam_unix acct="root" exe="/usr/bin/sudo" hostname=? addr=? terminal=/dev/pts/0 res=success'  
type=USER_END msg=audit(1541612348.760:680): pid=14460 uid=0 auid=1000 ses=2 subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:session_close grantors=pam_keyinit,pam_limits,pam_keyinit,pam_limits,pam_systemd,pam_unix acct="root" exe="/usr/bin/sudo" hostname=? addr=? terminal=/dev/pts/0 res=success'  
type=CRED_DISP msg=audit(1541612348.760:681): pid=14460 uid=0 auid=1000 ses=2 subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:setcred grantors=pam_env,pam_localuser,pam_unix acct="root" exe="/usr/bin/sudo" hostname=? addr=? terminal=/dev/pts/0 res=success'  
type=USER_AUTH msg=audit(1541612370.868:682): pid=14470 uid=1000 auid=1000 ses=2 subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:authentication grantors=? acct="pruebaSELinux" exe="/usr/bin/su" hostname=localhost-live addr=? terminal=pts/0 res=failed'  
type=USER_CMD msg=audit(1541612378.465:683): pid=14481 uid=1000 auid=1000 ses=2 subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='cwd="/home/liveuser" cmd=636174202F7661722F6C6F672F61756469742F61756469742E6C6F67 terminal=pts/0 res=success'  
type=CRED_REFR msg=audit(1541612378.465:684): pid=14481 uid=0 auid=1000 ses=2 su
```

Vemos que se trata de un mensaje de autenticación de usuario con respuesta fallida (denegada) que proviene del módulo PAM donde el usuario en cuestión es *pruebaSELinux* y el comando ejecutado */usr/bin/su*. Además contiene los diversos ID del proceso implicado entre otros detalles.

Además, vamos a ejecutar el comando *ls /proc/1* que sin tener privilegios de root no podemos visualizar cierto contenido. Cuando volvemos a comprobar el estado de *audit.log* vemos el siguiente mensaje de denegación:

```

liveuser@localhost:~ liveuser@localhost:~ x
File Edit View Search Terminal Help
ccess'
type=USER_AUTH msg=audit(1541612459.074:692): pid=14541 uid=1001 auid=1000 ses=2
subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:authentication
grantors=pam_succeed_if,pam_localuser,pam_unix acct="pruebaSELinux" exe="/usr/bin/sudo"
hostname=? addr=? terminal=/dev/pts/0 res=success'
type=USER_ACCT msg=audit(1541612459.074:693): pid=14541 uid=1001 auid=1000 ses=2
subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:accounting
grantors=pam_unix,pam_localuser acct="pruebaSELinux" exe="/usr/bin/sudo" hostname=? addr=? terminal=/dev/pts/0 res=success'
type=USER_CMD msg=audit(1541612459.074:694): pid=14541 uid=1001 auid=1000 ses=2
subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='cwd="/home/liveuser" cmd=636174202F7661722F6C6F672F61756469742F61756469742E6C6F67 terminal=pts/0
res=failed'
type=USER_END msg=audit(1541612467.681:695): pid=14498 uid=1000 auid=1000 ses=2
subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:session_close
grantors=pam_keyinit,pam_limits,pam_systemd,pam_unix,pam_umask acct="pruebaSELinux" exe="/usr/bin/su" hostname=localhost-live addr=? terminal=pts/0 res=succes
type=CRED_DISP msg=audit(1541612467.681:696): pid=14498 uid=1000 auid=1000 ses=2
subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:setcred
grantors=pam_localuser,pam_unix acct="pruebaSELinux" exe="/usr/bin/su" hostname=
localhost-live addr=? terminal=pts/0 res=success'
type=USER_CMD msg=audit(1541612501.821:697): pid=14565 uid=1000 auid=1000 ses=2
subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='cwd="/home/liveu

```

Se trata de un mensaje acerca de un comando de usuario ejecutado desde el directorio de trabajo `/home/liveuser` donde se deniegan los permisos al comando.

Ejercicio 3.

Existen diversos modos en los que se puede ejecutar SELinux. El modo *enforcing* obliga a que se aplique la política definida y los accesos se basan en esta. El modo *permissive* no obliga a que se aplique la política pero notifica en el log de la misma manera que en el modo *enforcing*. El modo *disabled* deshabilita SELinux. Para cambiar entre los modos *permissive* y *enforcing* utilizamos el comando `setenforce`. En la siguiente imagen se ejemplifica el cambio de *permissive* a *enforcing* mediante el comando `sudo setenforce Enforcing`:

```

liveuser@localhost:~ liveuser@localhost:~ x
File Edit View Search Terminal Help
[liveuser@localhost-live ~]$ sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:              targeted
Current mode:                   permissive
Mode from config file:          enforcing
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:    actual (secure)
Max kernel policy version:      31
[liveuser@localhost-live ~]$ sudo setenforce Enforcing
[liveuser@localhost-live ~]$ sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:              targeted
Current mode:                   enforcing
Mode from config file:          enforcing
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:    actual (secure)
Max kernel policy version:      31
[liveuser@localhost-live ~]$ 

```

También podemos realizar lo mismo mediante el comando análogo `sudo setenforce 0`:

```
liveuser@localhost:~
```

```
File Edit View Search Terminal Help
[liveuser@localhost-live ~]$ sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:              targeted
Current mode:                   permissive
Mode from config file:          enforcing
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:    actual (secure)
Max kernel policy version:     31
[liveuser@localhost-live ~]$ sudo setenforce 0
[liveuser@localhost-live ~]$ sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:              targeted
Current mode:                   permissive
Mode from config file:          enforcing
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:    actual (secure)
Max kernel policy version:     31
[liveuser@localhost-live ~]$
```

Ejercicio 4.

Finalmente vamos a completar la siguiente información acerca de la distribución de Linux en la que estamos trabajando:

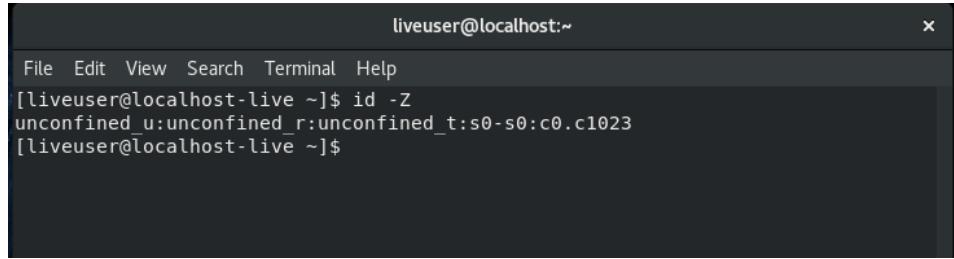
Distribución	Policy store name	MLS?	deny_unknown	Unconfined domains?	UBAC?
--------------	----------------------	------	--------------	------------------------	-------

Se trata de la versión Fedora 28. Con el comando `sestatus`, que muestra información básica acerca de SELinux en nuestro sistema, podemos ver el nombre de la política:`targeted` (línea 5), si hay MLS (MultiLevel Security):`enabled` y la política `deny_unknown:allowed`

```
liveuser@localhost:~
```

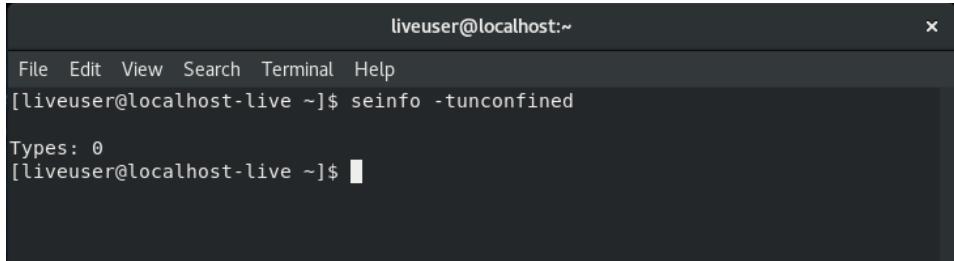
```
File Edit View Search Terminal Help
[liveuser@localhost-live ~]$ sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:              targeted
Current mode:                   permissive
Mode from config file:          enforcing
Policy MLS status:              enabled
Policy deny unknown status:     allowed
Memory protection checking:    actual (secure)
Max kernel policy version:     31
[liveuser@localhost-live ~]$
```

Que MLS esté habilitado concuerda con haber observado que existe el cuarto campo de rango/categoría (o sensibilidad) en el contexto SELinux. Con el comando `id -Z` que imprime el contexto de seguridad del usuario actual podemos ver cómo existe este campo que, en este caso, vale `s0-s0:c0.c1023`



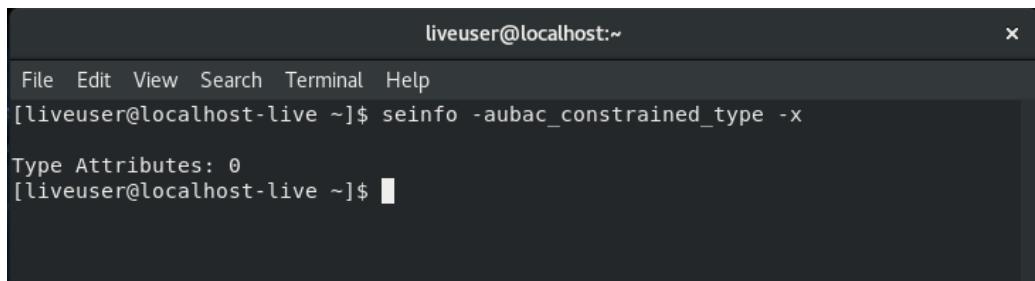
```
liveuser@localhost:~  
File Edit View Search Terminal Help  
[liveuser@localhost-live ~]$ id -Z  
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023  
[liveuser@localhost-live ~]$
```

Para comprobar si nuestro sistema tiene habilitados o no los dominios sin confinar (unconfined domains) utilizamos el comando `seinfo -tunconfined` que si devuelve error significa que no están habilitados. Este no es nuestro caso, como se aprecia en la siguiente imagen, por lo que los dominios confinados están habilitados:



```
liveuser@localhost:~  
File Edit View Search Terminal Help  
[liveuser@localhost-live ~]$ seinfo -tunconfined  
Types: 0  
[liveuser@localhost-live ~]$
```

Finalmente, para comprobar si en nuestra distribución existe UBAC o control de acceso basado en usuario, utilizamos el comando `seinfo -aubac_constrained_type -x` que nos devuelve la siguiente salida, confirmando que no existe UBAC:



```
liveuser@localhost:~  
File Edit View Search Terminal Help  
[liveuser@localhost-live ~]$ seinfo -aubac_constrained_type -x  
Type Attributes: 0  
[liveuser@localhost-live ~]$
```

Por lo tanto, la tabla quedaría de la siguiente manera:

Distribución	Policy store name	MLS?	deny_unknown	Unconfined domains?	UBAC?
Fedora 28	targeted	Yes (enabled)	allowed	Yes	No