Red Hat Enterprise Linux 8

Uso de SELinux

Configuración básica y avanzada de Security-Enhanced Linux (SELinux)

Last Updated: 2021-02-28

Red Hat Enterprise Linux 8 Uso de SELinux

Configuración básica y avanzada de Security-Enhanced Linux (SELinux)

Enter your first name here. Enter your surname here.
Enter your organisation's name here. Enter your organisational division here.
Enter your email address here.

Legal Notice

Resumen

Este título ayuda a los usuarios y administradores a aprender los fundamentos y principios sobre los que funciona SELinux y describe tareas prácticas para instalar y configurar varios servicios.

Table of Contents

HACER QUE EL CÓDIGO ABIERTO SEA MÁS INCLUSIVO

Red Hat se compromete a sustituir el lenguaje problemático en nuestro código, documentación y propiedades web. Estamos empezando con estos cuatro términos: maestro, esclavo, lista negra y lista blanca. Debido a la enormidad de este esfuerzo, estos cambios se implementarán gradualmente a lo largo de varias versiones próximas. Para más detalles, consulte el mensaje de nuestro CTO Chris Wright .

PROPORCIONAR COMENTARIOS SOBRE LA DOCUMENTACIÓN DE RED HAT

Agradecemos su opinión sobre nuestra documentación. Por favor, díganos cómo podemos mejorarla. Para ello:

- Para comentarios sencillos sobre pasajes concretos:
  1. Asegúrese de que está viendo la documentación en el formato *Multi-page HTML*. Además, asegúrese de ver el botón **Feedback** en la esquina superior derecha del documento.
  2. Utilice el cursor del ratón para resaltar la parte del texto que desea comentar.
  3. Haga clic en la ventana emergente **Add Feedback** que aparece debajo del texto resaltado.
  4. Siga las instrucciones mostradas.
- Para enviar comentarios más complejos, cree un ticket de Bugzilla:
  1. Vaya al sitio web de Bugzilla.
  2. Como componente, utilice **Documentation**.
  3. Rellene el campo **Description** con su sugerencia de mejora. Incluya un enlace a la(s) parte(s) pertinente(s) de la documentación.
  4. Haga clic en **Submit Bug**.

CAPÍTULO 1. INTRODUCCIÓN A SELINUX

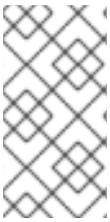
1.1. INTRODUCCIÓN A SELINUX

Security Enhanced Linux (SELinux) proporciona una capa adicional de seguridad del sistema. SELinux responde fundamentalmente a la pregunta: *May <subject> do <action> to <object>?*, por ejemplo *May a web server access files in users' home directories?*

La política de acceso estándar basada en el usuario, el grupo y otros permisos, conocida como Control de Acceso Discrecional (DAC), no permite a los administradores del sistema crear políticas de seguridad completas y de grano fino, como restringir aplicaciones específicas para que sólo vean los archivos de registro, mientras que se permite a otras aplicaciones añadir nuevos datos a los archivos de registro.

SELinux implementa el Control de Acceso Obligatorio (MAC). Cada proceso y recurso del sistema tiene una etiqueta de seguridad especial llamada *SELinux context*. Un contexto de SELinux, a veces denominado *SELinux label*, es un identificador que abstrae los detalles a nivel de sistema y se centra en las propiedades de seguridad de la entidad. Esto no sólo proporciona una forma consistente de referenciar objetos en la política de SELinux, sino que también elimina cualquier ambigüedad que pueda encontrarse en otros métodos de identificación. Por ejemplo, un archivo puede tener múltiples nombres de ruta válidos en un sistema que hace uso de montajes bind.

La política de SELinux utiliza estos contextos en una serie de reglas que definen cómo los procesos pueden interactuar entre sí y con los distintos recursos del sistema. Por defecto, la política no permite ninguna interacción a menos que una regla conceda explícitamente el acceso.



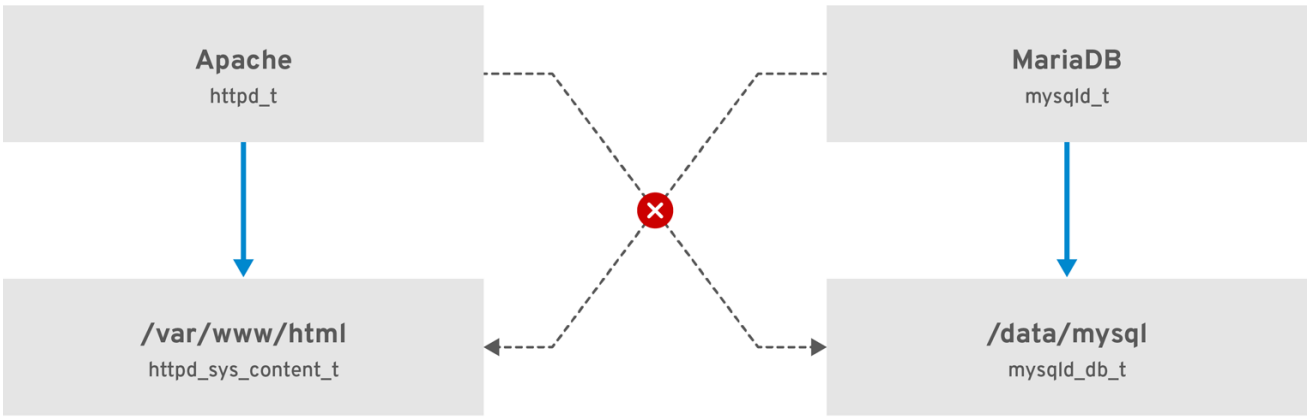
NOTA

Recuerda que las reglas de política de SELinux se comprueban después de las reglas DAC. Las reglas de política de SELinux no se utilizan si las reglas DAC deniegan el acceso primero, lo que significa que no se registra ninguna denegación de SELinux si las reglas DAC tradicionales impiden el acceso.

Los contextos de SELinux tienen varios campos: usuario, rol, tipo y nivel de seguridad. La información del tipo de SELinux es quizás la más importante cuando se trata de la política de SELinux, ya que la regla de política más común que define las interacciones permitidas entre los procesos y los recursos del sistema utiliza los tipos de SELinux y no el contexto completo de SELinux. Los tipos de SELinux terminan con **_t**. Por ejemplo, el nombre del tipo para el servidor web es **httpd_t**. El contexto de tipo para los archivos y directorios que normalmente se encuentran en **/var/www/html/** es **httpd_sys_content_t**. El contexto de tipo para los archivos y directorios que normalmente se encuentran en **/tmp** y **/var/tmp/** es **tmp_t**. El contexto de tipo para los puertos del servidor web es **http_port_t**.

Hay una regla de política que permite a Apache (el proceso del servidor web que se ejecuta como **httpd_t**) acceder a los archivos y directorios con un contexto que normalmente se encuentra en **/var/www/html/** y otros directorios del servidor web (**httpd_sys_content_t**). No hay ninguna regla de permiso en la política para los archivos que normalmente se encuentran en **/tmp** y **/var/tmp/**, por lo que el acceso no está permitido. Con SELinux, incluso si Apache está comprometido, y un script malicioso obtiene acceso, todavía no es capaz de acceder al directorio **/tmp**.

Figura 1.1. Un ejemplo de cómo SELinux puede ayudar a ejecutar Apache y MariaDB de forma segura.



Como muestra el esquema anterior, SELinux permite que el proceso de Apache que se ejecuta como **httpd_t** acceda al directorio **/var/www/html/** y le niega al mismo proceso el acceso al directorio **/data/mysql/** porque no existe una regla de permiso para los contextos de tipo **httpd_t** y **mysqld_db_t**. Por otro lado, el proceso de MariaDB que se ejecuta como **mysqld_t** puede acceder al directorio **/data/mysql/** y SELinux también deniega correctamente al proceso con el tipo **mysqld_t** el acceso al directorio **/var/www/html/** etiquetado como **httpd_sys_content_t**.

Recursos adicionales

Para más información, consulte la siguiente documentación:

- La página de manual **selinux(8)** y las páginas de manual listadas por el comando **apropos selinux**.
- Páginas de manual listadas por el comando **man -k _selinux** cuando el paquete **selinux-policy-doc** está instalado.
- El libro para colorear de SELinux le ayuda a entender mejor los conceptos básicos de SELinux.
- Preguntas frecuentes de la Wiki de SELinux

1.2. VENTAJAS DE EJECUTAR SELINUX

SELinux proporciona los siguientes beneficios:

- Todos los procesos y archivos están etiquetados. Las reglas de política de SELinux definen cómo los procesos interactúan con los archivos, así como cómo los procesos interactúan entre sí. Sólo se permite el acceso si existe una regla de política de SELinux que lo permita específicamente.
- Control de acceso detallado. Más allá de los permisos tradicionales de UNIX, que se controlan a discreción del usuario y se basan en los ID de usuario y grupo de Linux, las decisiones de acceso de SELinux se basan en toda la información disponible, como un usuario, un rol, un tipo y, opcionalmente, un nivel de seguridad de SELinux.
- La política de SELinux se define administrativamente y se aplica en todo el sistema.
- Mejora de la mitigación de los ataques de escalada de privilegios. Los procesos se ejecutan en dominios y, por tanto, están separados unos de otros. Las reglas de política de SELinux definen

cómo los procesos acceden a los archivos y a otros procesos. Si un proceso se ve comprometido, el atacante sólo tiene acceso a las funciones normales de ese proceso, y a los archivos a los que el proceso ha sido configurado para tener acceso. Por ejemplo, si el Servidor HTTP Apache está comprometido, un atacante no puede usar ese proceso para leer archivos en los directorios personales de los usuarios, a menos que una regla de política SELinux específica haya sido agregada o configurada para permitir ese acceso.

- SELinux puede utilizarse para reforzar la confidencialidad e integridad de los datos, así como para proteger los procesos de las entradas no fiables.

Sin embargo, SELinux no lo es:

- software antivirus,
- reemplazo de contraseñas, cortafuegos y otros sistemas de seguridad,
- solución de seguridad todo en uno.

SELinux está diseñado para mejorar las soluciones de seguridad existentes, no para sustituirlas. Incluso cuando se ejecuta SELinux, es importante seguir las buenas prácticas de seguridad, como mantener el software actualizado, utilizar contraseñas difíciles de adivinar y cortafuegos.

1.3. EJEMPLOS DE SELINUX

Los siguientes ejemplos demuestran cómo SELinux aumenta la seguridad:

- La acción por defecto es denegar. Si no existe una regla de política de SELinux que permita el acceso, como por ejemplo para un proceso que abre un archivo, el acceso se deniega.
- SELinux puede confinar a los usuarios de Linux. Existen varios usuarios confinados de SELinux en la política de SELinux. Los usuarios de Linux pueden ser asignados a usuarios confinados de SELinux para aprovechar las reglas y mecanismos de seguridad aplicados a ellos. Por ejemplo, al mapear un usuario de Linux al usuario de SELinux **user_u**, se obtiene un usuario de Linux que no puede ejecutar, a menos que se configure de otra manera, aplicaciones con ID de usuario (setuid), como **sudo** y **su**, además de evitar que ejecuten archivos y aplicaciones potencialmente maliciosas en su directorio raíz.
- Mayor separación de procesos y datos. El concepto de SELinux *domains* permite definir qué procesos pueden acceder a determinados archivos y directorios. Por ejemplo, cuando se ejecuta SELinux, a menos que se configure de otra manera, un atacante no puede comprometer un servidor Samba, y luego utilizar ese servidor Samba como un vector de ataque para leer y escribir en archivos utilizados por otros procesos, como las bases de datos MariaDB.
- SELinux ayuda a mitigar los daños causados por los errores de configuración. Los servidores del Sistema de Nombres de Dominio (DNS) suelen replicar la información entre ellos en lo que se conoce como transferencia de zona. Los atacantes pueden utilizar las transferencias de zona para actualizar los servidores DNS con información falsa. Cuando se ejecuta el Berkeley Internet Name Domain (BIND) como un servidor DNS en Red Hat Enterprise Linux, incluso si un administrador se olvida de limitar qué servidores pueden realizar una transferencia de zona, la política SELinux por defecto evita que los archivos de zona [1] sean actualizados mediante transferencias de zona, por el propio demonio BIND **named** y por otros procesos.

1.4. ARQUITECTURA Y PAQUETES SELINUX

SELinux es un módulo de seguridad de Linux (LSM) que está integrado en el núcleo de Linux. El

subsistema SELinux en el kernel está dirigido por una política de seguridad que es controlada por el administrador y cargada en el arranque. Todas las operaciones de acceso a nivel de kernel relevantes para la seguridad en el sistema son interceptadas por SELinux y examinadas en el contexto de la política de seguridad cargada. Si la política cargada permite la operación, ésta continúa. En caso contrario, la operación se bloquea y el proceso recibe un error.

Las decisiones de SELinux, como permitir o no el acceso, se almacenan en la caché. Esta caché se conoce como Access Vector Cache (AVC). Cuando se utilizan estas decisiones en caché, las reglas de política de SELinux necesitan ser comprobadas menos, lo que aumenta el rendimiento. Recuerda que las reglas de política de SELinux no tienen efecto si las reglas DAC deniegan el acceso primero. Los mensajes de auditoría sin procesar se registran en **/var/log/audit/audit.log** y comienzan con la cadena **type=AVC**.

En Red Hat Enterprise Linux 8, los servicios del sistema son controlados por el demonio **systemd**; **systemd** inicia y detiene todos los servicios, y los usuarios y procesos se comunican con **systemd** usando la utilidad **systemctl**. El demonio **systemd** puede consultar la política de SELinux y comprobar la etiqueta del proceso que llama y la etiqueta del archivo de la unidad que la persona que llama intenta gestionar, y luego preguntar a SELinux si la persona que llama tiene o no permiso de acceso. Este enfoque refuerza el control de acceso a las capacidades críticas del sistema, que incluyen el inicio y la detención de los servicios del sistema.

El demonio **systemd** también funciona como un gestor de acceso de SELinux. Recupera la etiqueta del proceso que ejecuta **systemctl** o el proceso que envió un mensaje **D-Bus** a **systemd**. El demonio busca entonces la etiqueta del archivo de unidad que el proceso quería configurar. Finalmente, **systemd** puede recuperar información del kernel si la política SELinux permite el acceso específico entre la etiqueta del proceso y la etiqueta del archivo de unidad. Esto significa que una aplicación comprometida que necesita interactuar con **systemd** para un servicio específico puede ahora ser confinada por SELinux. Los escritores de políticas también pueden utilizar estos controles de grano fino para confinar a los administradores.

IMPORTANTE

Para evitar un etiquetado incorrecto de SELinux y los problemas subsiguientes, asegúrese de iniciar los servicios utilizando un comando **systemctl start**.

Red Hat Enterprise Linux 8 proporciona los siguientes paquetes para trabajar con SELinux:

- políticas: **selinux-policy-targeted**, **selinux-policy-mls**
- herramientas: **policycoreutils**, **policycoreutils-gui**, **libselinux-utils**, **policycoreutils-python-utils**, **setools-console**, **checkpolicy**

1.5. ESTADOS Y MODOS DE SELINUX

SELinux puede ejecutarse en uno de los tres modos: reforzado, permisivo o deshabilitado.

- El modo de aplicación es el modo de operación por defecto, y el recomendado; en el modo de aplicación SELinux opera normalmente, aplicando la política de seguridad cargada en todo el sistema.
- En el modo permisivo, el sistema actúa como si SELinux estuviera aplicando la política de seguridad cargada, incluyendo el etiquetado de objetos y la emisión de entradas de denegación de acceso en los registros, pero en realidad no deniega ninguna operación. Aunque no se recomienda para sistemas de producción, el modo permisivo puede ser útil para el desarrollo y depuración de políticas de SELinux.

- Se desaconseja encarecidamente el modo deshabilitado; el sistema no sólo evita aplicar la política de SELinux, sino que también evita etiquetar cualquier objeto persistente, como los archivos, lo que dificulta la habilitación de SELinux en el futuro.

Utilice la utilidad **setenforce** para cambiar entre el modo de aplicación y el modo permisivo. Los cambios realizados con **setenforce** no persisten a través de los reinicios. Para cambiar al modo obligatorio, introduzca el comando **setenforce 1** como usuario root de Linux. Para cambiar al modo permisivo, introduzca el comando **setenforce 0**. Utilice la utilidad **getenforce** para ver el modo SELinux actual:

```
# getenforce
Enforcing
```

```
# setenforce 0
# getenforce
Permissive
```

```
# setenforce 1
# getenforce
Enforcing
```

En Red Hat Enterprise Linux, puede poner dominios individuales en modo permisivo mientras el sistema se ejecuta en modo forzoso. Por ejemplo, para hacer que el dominio *httpd_t* sea permisivo:

```
# semanage permissive -a httpd_t
```

Tenga en cuenta que los dominios permisivos son una herramienta poderosa que puede comprometer la seguridad de su sistema. Red Hat recomienda utilizar los dominios permisivos con precaución, por ejemplo, al depurar un escenario específico.

---

[1] Archivos de texto que incluyen información, como mapeos de nombres de host a direcciones IP, que son utilizados por los servidores DNS.

CAPÍTULO 2. CAMBIO DE ESTADOS Y MODOS DE SELINUX

Cuando está habilitado, SELinux puede funcionar en uno de los dos modos: enforcing o permissive. Las siguientes secciones muestran cómo cambiar permanentemente a estos modos.

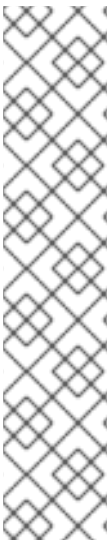
2.1. CAMBIOS PERMANENTES EN LOS ESTADOS Y MODOS DE SELINUX

Como se discute en Estados y modos de SELinux, SELinux puede estar habilitado o deshabilitado. Cuando está habilitado, SELinux tiene dos modos: enforcing y permissive.

Utilice los comandos **getenforce** o **sestatus** para comprobar en qué modo se está ejecutando SELinux. El comando **getenforce** devuelve **Enforcing**, **Permissive**, o **Disabled**.

El comando **sestatus** devuelve el estado de SELinux y la política de SELinux que se está utilizando:

```
$ 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
```



NOTA

Cuando los sistemas ejecutan SELinux en modo permisivo, los usuarios y los procesos pueden etiquetar varios objetos del sistema de archivos de forma incorrecta. Los objetos del sistema de archivos creados mientras SELinux está deshabilitado no son etiquetados en absoluto. Este comportamiento causa problemas cuando se cambia al modo de aplicación porque SELinux se basa en el etiquetado correcto de los objetos del sistema de archivos.

Para evitar que los archivos incorrectamente etiquetados y sin etiquetar causen problemas, los sistemas de archivos se vuelven a etiquetar automáticamente al cambiar del estado deshabilitado al modo permisivo o de aplicación. En el modo permisivo, utilice el comando **fixfiles -F onboot** como root para crear el archivo **/.autorelabel** que contiene la opción **-F** para garantizar que los archivos se vuelvan a etiquetar en el siguiente reinicio.

2.2. CAMBIO AL MODO PERMISIVO

Utilice el siguiente procedimiento para cambiar permanentemente el modo de SELinux a permisivo. Cuando SELinux se ejecuta en modo permisivo, la política de SELinux no se aplica. El sistema permanece operativo y SELinux no deniega ninguna operación, sino que sólo registra los mensajes de CVA, que pueden utilizarse para la resolución de problemas, la depuración y la mejora de la política de SELinux. Cada CVA se registra sólo una vez en este caso.

Requisitos previos

- Los paquetes **selinux-policy-targeted**, **libselinux-utils**, y **policycoreutils** están instalados en su sistema.
- Los parámetros del núcleo **selinux=0** o **enforcing=0** no se utilizan.

Procedimiento

1. Abra el archivo **/etc/selinux/config** en un editor de texto de su elección, por ejemplo:

```
# vi /etc/selinux/config
```

2. Configure la opción **SELINUX=permissive**:

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#       enforcing - SELinux security policy is enforced.
#       permissive - SELinux prints warnings instead of enforcing.
#       disabled - No SELinux policy is loaded.
SELINUX=permissive
# SELINUXTYPE= can take one of these two values:
#       targeted - Targeted processes are protected,
#       mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

3. Reinicia el sistema:

```
# reboot
```

Pasos de verificación

1. Después de reiniciar el sistema, confirme que el comando **getenforce** devuelve **Permissive**:

```
$ getenforce
Permissive
```

2.3. CAMBIO AL MODO DE APLICACIÓN

Utilice el siguiente procedimiento para cambiar SELinux al modo de aplicación. Cuando SELinux se ejecuta en modo de aplicación, aplica la política de SELinux y deniega el acceso basándose en las reglas de la política de SELinux. En RHEL, el modo de aplicación está activado por defecto cuando el sistema se instaló inicialmente con SELinux.

Requisitos previos

- Los paquetes **selinux-policy-targeted**, **libselinux-utils**, y **policycoreutils** están instalados en su sistema.
- Los parámetros del núcleo **selinux=0** o **enforcing=0** no se utilizan.

Procedimiento

1. Abra el archivo **/etc/selinux/config** en un editor de texto de su elección, por ejemplo:

```
# vi /etc/selinux/config
```

2. Configure la opción **SELINUX=enforcing**:

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#       enforcing - SELinux security policy is enforced.
#       permissive - SELinux prints warnings instead of enforcing.
#       disabled - No SELinux policy is loaded.
SELINUX=enforcing
# SELINUXTYPE= can take one of these two values:
#       targeted - Targeted processes are protected,
#       mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

3. Guarde el cambio y reinicie el sistema:

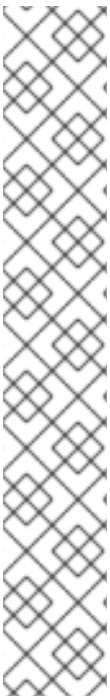
```
# reboot
```

En el siguiente arranque, SELinux reetiqueta todos los archivos y directorios dentro del sistema y añade el contexto de SELinux para los archivos y directorios que fueron creados cuando SELinux estaba deshabilitado.

Pasos de verificación

1. Después de reiniciar el sistema, confirme que el comando **getenforce** devuelve **Enforcing**:

```
$ getenforce
Enforcing
```



NOTA

Después de cambiar al modo de aplicación, SELinux puede denegar algunas acciones debido a reglas de política de SELinux incorrectas o ausentes. Para ver qué acciones deniega SELinux, introduzca el siguiente comando como root:

```
# ausearch -m AVC,USER_AVC,SELINUX_ERR,USER_SELINUX_ERR -ts today
```

Alternativamente, con el paquete **setroubleshoot-server** instalado, introduzca:

```
# grep "SELinux is preventing" /var/log/messages
```

Si SELinux está activo y el demonio de auditoría (**auditd**) no se está ejecutando en su sistema, entonces busque ciertos mensajes de SELinux en la salida del comando **dmesg**:

```
# dmesg | grep -i -e type=1300 -e type=1400
```

Para más información, consulte Solución de problemas relacionados con SELinux .

2.4. HABILITACIÓN DE SELINUX EN SISTEMAS QUE ANTERIORMENTE LO TENÍAN DESHABILITADO

Cuando habilite SELinux en sistemas que lo tenían previamente deshabilitado, para evitar problemas, como que los sistemas no puedan arrancar o que se produzcan fallos en los procesos, siga este procedimiento:

Procedimiento

1. Habilitar SELinux en modo permisivo. Para más información, consulte Cambiar al modo permisivo.
2. Reinicie su sistema:

```
# reboot
```

3. Compruebe si hay mensajes de denegación de SELinux. Para más información, consulte Identificación de denegaciones de SELinux.
4. Si no hay denegaciones, cambie al modo de refuerzo. Para más información, consulte Cambio de los modos de SELinux en el arranque.

Pasos de verificación

1. Después de reiniciar el sistema, confirme que el comando **getenforce** devuelve **Enforcing**:

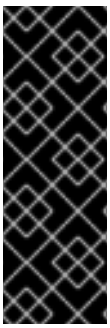
```
$ getenforce
Enforcing
```

Recursos adicionales

- Para ejecutar aplicaciones personalizadas con SELinux en modo de refuerzo, elija uno de los siguientes escenarios:
  - Ejecute su aplicación en el dominio **unconfined_service_t**.
  - Escriba una nueva política para su aplicación. Consulte el artículo de la base de conocimientos Escribir una política personalizada de SELinux para obtener más información.
- Los cambios temporales en los modos se tratan en Estados y modos de SELinux.

2.5. DESACTIVACIÓN DE SELINUX

Utilice el siguiente procedimiento para desactivar SELinux de forma permanente.



IMPORTANTE

Cuando SELinux está deshabilitado, la política de SELinux no se carga en absoluto; no se aplica y los mensajes de AVC no se registran. Por lo tanto, se pierden todos los beneficios de ejecutar SEL inux.

Red Hat recomienda encarecidamente utilizar el modo permisivo en lugar de desactivar permanentemente SELinux. Vea Cambiar al modo permisivo para más información sobre el modo permisivo.

AVISO

Desactivar SELinux usando la opción **SELINUX=disabled** en el **/etc/selinux/config** resulta en un proceso en el que el kernel arranca con SELinux activado y cambia al modo desactivado más tarde en el proceso de arranque. Debido a que pueden ocurrir fugas de memoria y condiciones de carrera que causen pánicos en el kernel, prefiera deshabilitar SELinux agregando el parámetro **selinux=0** a la línea de comandos del kernel como se describe en Cambiar los modos de SELinux en el arranque si su escenario realmente requiere deshabilitar completamente SELinux.

Procedimiento

1. Abra el archivo **/etc/selinux/config** en un editor de texto de su elección, por ejemplo:

```
# vi /etc/selinux/config
```

2. Configure la opción **SELINUX=disabled**:

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#       enforcing - SELinux security policy is enforced.
#       permissive - SELinux prints warnings instead of enforcing.
#       disabled - No SELinux policy is loaded.
SELINUX=disabled
# SELINUXTYPE= can take one of these two values:
#       targeted - Targeted processes are protected,
#       mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

3. Guarde el cambio y reinicie su sistema:

```
# reboot
```

Pasos de verificación

1. Después de reiniciar, confirme que el comando **getenforce** devuelve **Disabled**:

```
$ getenforce
Disabled
```

2.6. CAMBIO DE LOS MODOS DE SELINUX EN EL ARRANQUE

En el arranque, puedes establecer varios parámetros del kernel para cambiar la forma en que SELinux se ejecuta:

aplicación=0

La configuración de este parámetro hace que el sistema se inicie en modo permisivo, lo que resulta útil para solucionar problemas. El uso del modo permisivo puede ser la única opción para detectar un problema si su sistema de archivos está demasiado dañado. Además, en modo permisivo, el sistema

sigue creando las etiquetas correctamente. Los mensajes AVC que se crean en este modo pueden ser diferentes que en el modo de aplicación.
En el modo permisivo, sólo se informa de la primera denegación de una serie de denegaciones iguales. Sin embargo, en el modo de aplicación, puede obtener una denegación relacionada con la lectura de un directorio, y una aplicación se detiene. En el modo permisivo, se obtiene el mismo mensaje de AVC, pero la aplicación continúa leyendo archivos en el directorio y se obtiene un AVC por cada denegación adicional.

selinux=0

Este parámetro hace que el kernel no cargue ninguna parte de la infraestructura SELinux. Los scripts de init notan que el sistema arrancó con el parámetro **selinux=0** y tocan el archivo **/.autorelabel**. Esto hace que el sistema se vuelva a etiquetar automáticamente la próxima vez que arranque con SELinux activado.

IMPORTANTE

Red Hat no recomienda utilizar el parámetro **selinux=0**. Para depurar su sistema, prefiera utilizar el modo permisivo.

autorelabel=1

Este parámetro obliga al sistema a reetiquetar de forma similar a los siguientes comandos:

```
# touch /.autorelabel
# reboot
```

Si un sistema de archivos contiene una gran cantidad de objetos mal etiquetados, inicie el sistema en modo permisivo para que el proceso de autoetiquetado tenga éxito.

Recursos adicionales

- Para conocer otros parámetros de arranque del kernel relacionados con SELinux, como **checkreqprot**, consulte el archivo **/usr/share/doc/kernel-doc-*<KERNEL_VER>*/Documentation/admin-guide/kernel-parameters.txt** instalado con el paquete **kernel-doc**. Sustituya la cadena *<KERNEL_VER>* por el número de versión del kernel instalado, por ejemplo:

```
# yum install kernel-doc
$ less /usr/share/doc/kernel-doc-4.18.0/Documentation/admin-guide/kernel-parameters.txt
```

CAPÍTULO 3. GESTIÓN DE USUARIOS CONFINADOS Y NO CONFINADOS

Las siguientes secciones explican el mapeo de usuarios de Linux a usuarios de SELinux, describen los dominios básicos de usuarios confinados, y demuestran el mapeo de un nuevo usuario a un usuario de SELinux.

3.1. USUARIOS CONFINADOS Y NO CONFINADOS

Cada usuario de Linux se asigna a un usuario de SELinux utilizando la política de SELinux. Esto permite que los usuarios de Linux hereden las restricciones de los usuarios de SELinux.

Para ver la asignación de usuarios de SELinux en su sistema, utilice el comando **semanage login -l** como root:

```
# semanage login -l
Login Name           SELinux User         MLS/MCS Range        Service

__default__          unconfined_u         s0-s0:c0.c1023       *
root                 unconfined_u         s0-s0:c0.c1023       *
```

En Red Hat Enterprise Linux, los usuarios de Linux son asignados al login SELinux ***default*** por defecto, el cual está mapeado al usuario de SELinux **unconfined_u**. La siguiente línea define el mapeo por defecto:

```
__default__          unconfined_u         s0-s0:c0.c1023       *
```

Los usuarios confinados y no confinados de Linux están sujetos a comprobaciones de memoria ejecutable y escribible, y también están restringidos por MCS o MLS.

Para listar los usuarios de SELinux disponibles, introduzca el siguiente comando:

```
$ seinfo -u
Users: 8
   guest_u
   root
   staff_u
   sysadm_u
   system_u
   unconfined_u
   user_u
   xguest_u
```

Tenga en cuenta que el comando **seinfo** lo proporciona el paquete **setools-console**, que no está instalado por defecto.

Si un usuario de Linux no confinado ejecuta una aplicación que la política de SELinux define como una que puede pasar del dominio **unconfined_t** a su propio dominio confinado, el usuario de Linux no confinado sigue estando sujeto a las restricciones de ese dominio confinado. El beneficio de seguridad de esto es que, aunque un usuario de Linux esté ejecutando sin confinar, la aplicación permanece confinada. Por lo tanto, la explotación de una falla en la aplicación puede ser limitada por la política.

Del mismo modo, podemos aplicar estas comprobaciones a los usuarios confinados. Cada usuario confinado está restringido por un dominio de usuario confinado. La política de SELinux también puede definir una transición de un dominio de usuario confinado a su propio dominio confinado de destino. En

tal caso, los usuarios confinados están sujetos a las restricciones de ese dominio confinado de destino. El punto principal es que se asocian privilegios especiales a los usuarios confinados según su rol.

3.2. CAPACIDADES DE LOS USUARIOS DE SELINUX

La siguiente tabla proporciona ejemplos de dominios confinados básicos para usuarios de Linux en Red Hat Enterprise Linux:

Tabla 3.1. Capacidades de los usuarios de SELinux

Usuario	Papel	Dominio	Sistema X Window	su o sudo	Ejecutar en el directorio principal y en /tmp (por defecto)	Red
sysadm_u	sysadm_r	sysadm_t	sí	su y sudo	sí	sí
staff_u	staff_r	staff_t	sí	sólo sudo	sí	sí
usuario_u	usuario_r	usuario_t	sí	no	sí	sí
guest_u	guest_r	guest_t	no	no	sí	no
xguest_u	xguest_r	xguest_t	sí	no	sí	Sólo para Firefox

- Los usuarios de Linux en los dominios **user_t**, **guest_t**, y **xguest_t** sólo pueden ejecutar aplicaciones setuid si la política SELinux lo permite (por ejemplo, **passwd**). Estos usuarios no pueden ejecutar las aplicaciones setuid de **su** y **sudo**, y por lo tanto no pueden usar estas aplicaciones para convertirse en root.

- Los usuarios de Linux en los dominios **sysadm_t**, **staff_t**, **user_t**, y **xguest_t** pueden conectarse utilizando el sistema X Window y un terminal.

- Por defecto, los usuarios de Linux en los dominios **staff_t**, **user_t**, **guest_t**, y **xguest_t** pueden ejecutar aplicaciones en sus directorios personales y **/tmp**.
  Para evitar que ejecuten aplicaciones, que heredan los permisos de los usuarios, en directorios a los que tienen acceso de escritura, establezca los booleanos **guest_exec_content** y **xguest_exec_content** en **off**. Esto ayuda a evitar que aplicaciones defectuosas o maliciosas modifiquen los archivos de los usuarios.

- El único acceso a la red que tienen los usuarios de Linux en el dominio **xguest_t** es Firefox para conectarse a las páginas web.

- El usuario **sysadm_u** no puede iniciar sesión directamente utilizando SSH. Para habilitar los inicios de sesión SSH para **sysadm_u**, establezca el booleano **ssh_sysadm_login** en **on**:

```
# setsebool -P ssh_sysadm_login on
```

Tenga en cuenta que **system_u** es una identidad de usuario especial para los procesos y objetos del

sistema. Nunca debe asociarse a un usuario de Linux. Además, **unconfined_u** y **root** son usuarios no confinados. Por estas razones, no están incluidos en la tabla anterior de capacidades de usuario de SELinux.

Junto con los usuarios de SELinux ya mencionados, hay roles especiales, que pueden ser asignados a esos usuarios usando el comando **semanage user**. Estos roles determinan lo que SELinux permite hacer al usuario:

- **webadm_r** sólo puede administrar los tipos de SELinux relacionados con el servidor HTTP Apache.
- **dbadm_r** sólo puede administrar tipos de SELinux relacionados con la base de datos MariaDB y el sistema de gestión de bases de datos PostgreSQL.
- **logadm_r** sólo puede administrar los tipos de SELinux relacionados con los procesos **syslog** y **auditlog**.
- **secadm_r** sólo puede administrar SELinux.
- **auditadm_r** sólo puede administrar los procesos relacionados con el subsistema de Auditoría.

Para listar todos los roles disponibles, introduzca el comando **seinfo -r**:

```
$ seinfo -r
Roles: 14
   auditadm_r
   dbadm_r
   guest_r
   logadm_r
   nx_server_r
   object_r
   secadm_r
   staff_r
   sysadm_r
   system_r
   unconfined_r
   user_r
   webadm_r
   xguest_r
```

Tenga en cuenta que el comando **seinfo** lo proporciona el paquete **setools-console**, que no está instalado por defecto.

Recursos adicionales

- Para más información, consulte las páginas de manual **seinfo(1)**, **semanage-login(8)**, y **xguest_selinux(8)**.

3.3. AÑADIR UN NUEVO USUARIO MAPEADO AUTOMÁTICAMENTE AL USUARIO SELINUX UNCONFINED_U

El siguiente procedimiento demuestra cómo añadir un nuevo usuario Linux al sistema. El usuario se asigna automáticamente al usuario de SELinux **unconfined_u**.

Requisitos previos

- El usuario **root** se ejecuta sin restricciones, como lo hace por defecto en Red Hat Enterprise Linux.

Procedimiento

1. Introduzca el siguiente comando para crear un nuevo usuario de Linux llamado *example.user*:

```
# useradd example.user
```

2. Para asignar una contraseña al usuario de Linux *example.user*:

```
# passwd example.user
Changing password for user example.user.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

3. Salga de su sesión actual.

4. Inicia sesión como el usuario de Linux *example.user*. Al iniciar la sesión, el módulo PAM de **pam_selinux** asigna automáticamente el usuario de Linux a un usuario de SELinux (en este caso, **unconfined_u**), y configura el contexto de SELinux resultante. El shell del usuario de Linux se lanza entonces con este contexto.

Pasos de verificación

1. Al iniciar la sesión como usuario de *example.user*, compruebe el contexto de un usuario de Linux:

```
$ id -Z
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

Recursos adicionales

- Para más información, consulte la página de manual **pam_selinux(8)**.

3.4. AÑADIR UN NUEVO USUARIO COMO USUARIO CONFINADO EN SELINUX

Utiliza los siguientes pasos para añadir un nuevo usuario confinado en SELinux al sistema. Este procedimiento de ejemplo asigna el usuario al derecho de usuario de SELinux **staff_u** con el comando para crear la cuenta de usuario.

Requisitos previos

- El usuario **root** se ejecuta sin restricciones, como lo hace por defecto en Red Hat Enterprise Linux.

Procedimiento

1. Introduzca el siguiente comando para crear un nuevo usuario de Linux llamado *example.user* y asignarlo al usuario de SELinux **staff_u**:

```
# useradd -Z staff_u example.user
```

2. Para asignar una contraseña al usuario de Linux *example.user*:

```
# passwd example.user
Changing password for user example.user.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

3. Salga de su sesión actual.

4. Inicie la sesión como usuario de Linux *example.user*. El shell del usuario se lanza con el contexto **staff_u**.

Pasos de verificación

1. Al iniciar la sesión como usuario de *example.user*, compruebe el contexto de un usuario de Linux:

```
$ id -Z
uid=1000(example.user) gid=1000(example.user) groups=1000(example.user)
context=staff_u:staff_r:staff_t:s0-s0:c0.c1023
```

Recursos adicionales

- Para más información, consulte la página de manual **pam_selinux(8)**.

3.5. CONFIGURAR EL SISTEMA PARA CONFINAR A LOS USUARIOS DE SELINUX

Por defecto, todos los usuarios de Linux en Red Hat Enterprise Linux, incluyendo los usuarios con privilegios administrativos, son asignados al usuario no confinado de SELinux **unconfined_u**. Puede mejorar la seguridad del sistema asignando usuarios a usuarios confinados de SELinux. Esto es útil para cumplir con la Guía de Implementación Técnica de Seguridad V-71971 . Para más información acerca de los usuarios confinados y no confinados, consulte Gestión de usuarios confinados y no confinados .

3.5.1. Confinamiento de los usuarios habituales

Puede confinar a todos los usuarios normales de su sistema asignándolos al usuario **user_u** SELinux.

Procedimiento

1. Muestra la lista de registros de inicio de sesión de SELinux. La lista muestra las asignaciones de usuarios de Linux a usuarios de SELinux:

```
# semanage login -l

Login Name    SELinux User  MLS/MCS Range   Service

__default__   unconfined_u  s0-s0:c0.c1023      *
root          unconfined_u  s0-s0:c0.c1023      *
```

2. Mapea el usuario __default__, que representa a todos los usuarios sin un mapeo explícito, al usuario **user_u** SELinux:

```
# semanage login -m -s user_u -r s0 __default__
```

Pasos de verificación

1. Comprueba que el usuario __default__ está mapeado al usuario **user_u** SELinux:

```
# semanage login -l

Login Name    SELinux User   MLS/MCS Range    Service

__default__   user_u         s0               *
root          unconfined_u   s0-s0:c0.c1023   *
```

2. Comprueba que los procesos de un nuevo usuario se ejecutan en el contexto SELinux **user_u:user_r:user_t:s0**.

   a. Crear un nuevo usuario:

```
# adduser example.user
```

   b. Defina una contraseña para *example.user*:

```
# passwd example.user
```

   c. Cierre la sesión como **root** e inicie la sesión como el nuevo usuario.

   d. Muestra el contexto de seguridad para el ID del usuario:

```
[example.user@localhost ~]$ id -Z
user_u:user_r:user_t:s0
```

   e. Muestra el contexto de seguridad de los procesos actuales del usuario:

```
[example.user@localhost ~]$ ps axZ
LABEL                             PID TTY      STAT   TIME COMMAND
-                                   1 ?        Ss     0:05 /usr/lib/systemd/systemd --switched-root --
system --deserialize 18
-                                3729 ?        S      0:00 (sd-pam)
user_u:user_r:user_t:s0          3907 ?        Ss     0:00 /usr/lib/systemd/systemd --user
-                                3911 ?        S      0:00 (sd-pam)
user_u:user_r:user_t:s0          3918 ?        S      0:00 sshd: example.user@pts/0
user_u:user_r:user_t:s0          3922 pts/0    Ss     0:00 -bash
user_u:user_r:user_dbusd_t:s0    3969 ?        Ssl    0:00 /usr/bin/dbus-daemon --session --
address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
user_u:user_r:user_t:s0          3971 pts/0    R+     0:00 ps axZ
```

3.5.2. Confinar a los usuarios administradores

Puede utilizar uno de los dos métodos siguientes para confinar a los usuarios administradores.

3.5.2.1. Confirmar un administrador mediante la asignación a sysadm_u

Puedes confinar a un usuario con privilegios administrativos asignando el usuario directamente al usuario **sysadm_u** SELinux. Cuando el usuario se conecta, la sesión se ejecuta en el contexto **sysadm_u:sysadm_r:sysadm_t** SELinux.

Requisitos previos

- El usuario **root** se ejecuta sin restricciones. Este es el valor por defecto de Red Hat Enterprise Linux.

Procedimiento

1. Opcional: Para permitir que los usuarios de **sysadm_u** se conecten al sistema mediante SSH:

```
# setsebool -P ssh_sysadm_login on
```

2. Cree un nuevo usuario, añada el usuario al grupo de usuarios **wheel** y asigne el usuario al usuario de SELinux **sysadm_u**:

```
# adduser -G wheel -Z sysadm_u example.user
```

3. Opcional: Asigne un usuario existente al usuario **sysadm_u** SELinux y añada el usuario al grupo de usuarios **wheel**:

```
# usermod -G wheel -Z sysadm_u example.user
```

Pasos de verificación

1. Comprueba que ***example.user*** esté asignada al usuario de SELinux **sysadm_u**:

```
# semanage login -l | grep example.user
example.user     sysadm_u    s0-s0:c0.c1023   *
```

2. Inicie la sesión como ***example.user***por ejemplo, utilizando SSH, y mostrar el contexto de seguridad del usuario:

```
[example.user@localhost ~]$ id -Z
sysadm_u:sysadm_r:sysadm_t:s0-s0:c0.c1023
```

3. Cambia al usuario **root**:

```
$ sudo -i
[sudo] password for example.user:
```

4. Compruebe que el contexto de seguridad no se ha modificado:

```
# id -Z
sysadm_u:sysadm_r:sysadm_t:s0-s0:c0.c1023
```

5. Intente una tarea administrativa, por ejemplo, reiniciar el servicio **sshd**:

```
# systemctl restart sshd
```

Si no hay salida, el comando ha terminado con éxito.

Si el comando no termina con éxito, imprime el siguiente mensaje:

```
Failed to restart sshd.service: Access denied
See system logs and 'systemctl status sshd.service' for details.
```

3.5.2.2. Confinar a un administrador usando sudo y el rol sysadm_r

Puede asignar un usuario específico con privilegios administrativos al usuario **staff_u** SELinux, y configurar **sudo** para que el usuario pueda obtener el rol de administrador **sysadm_r** SELinux. Este rol le permite al usuario realizar tareas administrativas sin que se le niegue SELinux. Cuando el usuario se conecta, la sesión se ejecuta en el contexto **staff_u:staff_r:staff_t** SELinux, pero cuando el usuario introduce un comando usando **sudo**, la sesión cambia al contexto **staff_u:sysadm_r:sysadm_t**.

Requisitos previos

- El usuario **root** se ejecuta sin restricciones. Este es el valor por defecto de Red Hat Enterprise Linux.

Procedimiento

1. Cree un nuevo usuario, añada el usuario al grupo de usuarios **wheel** y asigne el usuario al usuario de SELinux **staff_u**:

```
# adduser -G wheel -Z staff_u example.user
```

2. Opcional: Asigne un usuario existente al usuario **staff_u** SELinux y añada el usuario al grupo de usuarios **wheel**:

```
# usermod -G wheel -Z staff_u example.user
```

3. Para permitir que *example.user* obtenga el rol de administrador de SELinux, cree un nuevo archivo en el directorio **/etc/sudoers.d/**, por ejemplo:

```
# visudo -f /etc/sudoers.d/example.user
```

4. Añada la siguiente línea al nuevo archivo:

```
example.user ALL=(ALL) TYPE=sysadm_t ROLE=sysadm_r ALL
```

Pasos de verificación

1. Comprueba que ***example.user*** esté asignada al usuario de SELinux **staff_u**:

```
# semanage login -l | grep example.user
example.user     staff_u    s0-s0:c0.c1023   *
```

2. Inicie sesión como *example.user*, por ejemplo, utilizando SSH, y cambie al usuario **root**:

```
[example.user@localhost ~]$ sudo -i
[sudo] password for example.user:
```

3. Muestra el contexto de seguridad de **root**:

```
# id -Z
staff_u:sysadm_r:sysadm_t:s0-s0:c0.c1023
```

4. Intente una tarea administrativa, por ejemplo, reiniciar el servicio **sshd**:

```
# systemctl restart sshd
```

Si no hay salida, el comando ha terminado con éxito.

Si el comando no termina con éxito, imprime el siguiente mensaje:

```
Failed to restart sshd.service: Access denied
See system logs and 'systemctl status sshd.service' for details.
```

3.5.3. Recursos adicionales

- Para conocer otras opciones, consulte el artículo de la base de conocimientos Cómo configurar un sistema con usuarios confinados en SELinux.
- Para más información, consulte las páginas de manual **user_selinux(8)**, **staff_selinux(8)**, y **sysadm_selinux(8)**.

3.6. RECURSOS ADICIONALES

- Para más información, consulte la página de manual **unconfined_selinux(8)**.

CAPÍTULO 4. CONFIGURACIÓN DE SELINUX PARA APLICACIONES Y SERVICIOS CON CONFIGURACIONES NO ESTÁNDAR

Cuando SELinux está en modo de aplicación, la política por defecto es la política objetivo. Las siguientes secciones proporcionan información sobre cómo establecer y configurar la política de SELinux para varios servicios después de cambiar los valores predeterminados de configuración, como los puertos, las ubicaciones de las bases de datos o los permisos del sistema de archivos para los procesos.

En los siguientes procedimientos, aprenderás a cambiar los tipos de SELinux para los puertos no estándar, a identificar y arreglar las etiquetas incorrectas para los cambios de directorios por defecto, y a ajustar la política usando booleanos de SELinux.

4.1. PERSONALIZACIÓN DE LA POLÍTICA SELINUX PARA EL SERVIDOR HTTP APACHE EN UNA CONFIGURACIÓN NO ESTÁNDAR

Puede configurar el servidor HTTP Apache para que escuche en un puerto diferente y proporcione contenido en un directorio no predeterminado. Para evitar las consiguientes denegaciones de SELinux, siga los pasos de este procedimiento para ajustar la política de SELinux de su sistema.

Requisitos previos

- Se instala el paquete **httpd** y se configura el servidor HTTP Apache para que escuche en el puerto TCP 3131 y utilice el directorio **/var/test_www/** en lugar del directorio predeterminado **/var/www/**.
- Los paquetes **policycoreutils-python-utils** y **setroubleshoot-server** están instalados en su sistema.

Procedimiento

1. Inicie el servicio **httpd** y compruebe el estado:

```
# systemctl start httpd
# systemctl status httpd
...
httpd[14523]: (13)Permission denied: AH00072: make_sock: could not bind to address
[::]:3131
...
systemd[1]: Failed to start The Apache HTTP Server.
...
```

2. La política de SELinux asume que **httpd** se ejecuta en el puerto 80:

```
# semanage port -l | grep http
http_cache_port_t              tcp      8080, 8118, 8123, 10001-10010
http_cache_port_t              udp      3130
http_port_t                    tcp      80, 81, 443, 488, 8008, 8009, 8443, 9000
pegasus_http_port_t            tcp      5988
pegasus_https_port_t           tcp      5989
```

3. Cambie el tipo de SELinux del puerto 3131 para que coincida con el puerto 80:

```
# semanage port -a -t http_port_t -p tcp 3131
```

4. Comienza de nuevo **httpd**:

```
# systemctl start httpd
```

5. Sin embargo, el contenido sigue siendo inaccesible:

```
# wget localhost:3131/index.html
...
HTTP request sent, awaiting response... 403 Forbidden
...
```

Encuentre el motivo con la herramienta **sealert**:

```
# sealert -l "*"
...
SELinux is preventing httpd from getattr access on the file /var/test_www/html/index.html.
...
```

6. Compare los tipos de SELinux para la ruta estándar y la nueva usando la herramienta **matchpathcon**:

```
# matchpathcon /var/www/html /var/test_www/html
/var/www/html       system_u:object_r:httpd_sys_content_t:s0
/var/test_www/html  system_u:object_r:var_t:s0
```

7. Cambie el tipo de SELinux del nuevo directorio de contenido **/var/test_www/html/** al tipo del directorio por defecto **/var/www/html**:

```
# semanage fcontext -a -e /var/www /var/test_www
```

8. Reetiquetar el directorio **/var** recursivamente:

```
# restorecon -Rv /var/
...
Relabeled /var/test_www/html from unconfined_u:object_r:var_t:s0 to
unconfined_u:object_r:httpd_sys_content_t:s0
Relabeled /var/test_www/html/index.html from unconfined_u:object_r:var_t:s0 to
unconfined_u:object_r:httpd_sys_content_t:s0
```

Pasos de verificación

1. Compruebe que el servicio **httpd** está funcionando:

```
# systemctl status httpd
...
Active: active (running)
...
systemd[1]: Started The Apache HTTP Server.
httpd[14888]: Server configured, listening on: port 3131
...
```

2. Compruebe que el contenido proporcionado por el servidor HTTP Apache es accesible:

```
# wget localhost:3131/index.html
...
HTTP request sent, awaiting response... 200 OK
Length: 0 [text/html]
Saving to: ‘index.html’
...
```

Recursos adicionales

- Las páginas de manual **semanage(8)**, **matchpathcon(8)**, y **sealert(8)**.

4.2. AJUSTE DE LA POLÍTICA PARA COMPARTIR VOLÚMENES NFS Y CIFS UTILIZANDO BOOLEANOS DE SELINUX

Puedes cambiar partes de la política de SELinux en tiempo de ejecución utilizando booleanos, incluso sin tener conocimiento de la escritura de la política de SELinux. Esto permite realizar cambios, como permitir el acceso de servicios a volúmenes NFS, sin necesidad de recargar o recompilar la política de SELinux. El siguiente procedimiento demuestra el listado de booleanos de SELinux y su configuración para lograr los cambios requeridos en la política.

Los montajes NFS en el lado del cliente se etiquetan con un contexto por defecto definido por una política para volúmenes NFS. En RHEL, este contexto por defecto utiliza el tipo **nfs_t**. Asimismo, los recursos compartidos de Samba montados en el lado del cliente se etiquetan con un contexto predeterminado definido por la política. Este contexto por defecto utiliza el tipo **cifs_t**. Puede habilitar o deshabilitar booleanos para controlar qué servicios pueden acceder a los tipos **nfs_t** y **cifs_t**.

Para permitir que el servicio de servidor HTTP Apache (**httpd**) acceda y comparta volúmenes NFS y CIFS, realice los siguientes pasos:

Requisitos previos

- Opcionalmente, instale el paquete **selinux-policy-devel** para obtener descripciones más claras y detalladas de los booleanos de SELinux en la salida del comando **semanage boolean -l**.

Procedimiento

1. Identificar los booleanos de SELinux relevantes para NFS, CIFS y Apache:

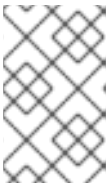
```
# semanage boolean -l | grep 'nfs\|cifs' | grep httpd
httpd_use_cifs                 (off  ,  off)  Allow httpd to access cifs file systems
httpd_use_nfs                  (off  ,  off)  Allow httpd to access nfs file systems
```

2. Lista el estado actual de los booleanos:

```
$ getsebool -a | grep 'nfs\|cifs' | grep httpd
httpd_use_cifs --> off
httpd_use_nfs --> off
```

3. Habilitar los booleanos identificados:

```
# setsebool httpd_use_nfs on
# setsebool httpd_use_cifs on
```



NOTA

Utilice **setsebool** con la opción **-P** para que los cambios persistan en los reinicios. Un comando **setsebool -P** requiere una reconstrucción de toda la política, y puede llevar algún tiempo dependiendo de su configuración.

Pasos de verificación

1. Comprueba que los booleanos son **on**:

```
$ getsebool -a | grep 'nfs\|cifs' | grep httpd
httpd_use_cifs --> on
httpd_use_nfs --> on
```

Recursos adicionales

- Las páginas de manual **semanage-boolean(8)**, **sepolicy-booleans(8)**, **getsebool(8)**, **setsebool(8)**, **booleans(5)**, y **booleans(8)**.

4.3. RECURSOS ADICIONALES

- Consulte Solución de problemas relacionados con SELinux para obtener más detalles sobre la identificación y el análisis de las denegaciones de SELinux.

CAPÍTULO 5. SOLUCIÓN DE PROBLEMAS RELACIONADOS CON SELINUX

Si planeas habilitar SELinux en sistemas en los que ha sido previamente deshabilitado o si ejecutas un servicio en una configuración no estándar, puede que necesites solucionar situaciones potencialmente bloqueadas por SELinux. Tenga en cuenta que, en la mayoría de los casos, las denegaciones de SELinux son signos de una configuración incorrecta.

5.1. IDENTIFICACIÓN DE LAS DENEGACIONES DE SELINUX

Siga sólo los pasos necesarios de este procedimiento; en la mayoría de los casos, sólo tendrá que realizar el paso 1.

Procedimiento

1. Cuando su escenario está bloqueado por SELinux, el archivo **/var/log/audit/audit.log** es el primer lugar en el que hay que buscar más información sobre una denegación. Para consultar los registros de auditoría, utilice la herramienta **ausearch**. Dado que las decisiones de SELinux, como permitir o denegar el acceso, se almacenan en caché y esta caché se conoce como caché de vectores de acceso (AVC), utilice los valores **AVC** y **USER_AVC** para el parámetro de tipo de mensaje, por ejemplo:

   ```
   # ausearch -m AVC,USER_AVC,SELINUX_ERR,USER_SELINUX_ERR -ts recent
   ```

   Si no hay coincidencias, compruebe si el demonio de Auditoría se está ejecutando. Si no lo hace, repita el escenario denegado después de iniciar **auditd** y compruebe de nuevo el registro de Auditoría.

2. En caso de que **auditd** se esté ejecutando, pero no haya coincidencias en la salida de **ausearch**, compruebe los mensajes proporcionados por el Diario **systemd**:

   ```
   # journalctl -t setroubleshoot
   ```

3. Si SELinux está activo y el demonio de auditoría no se está ejecutando en su sistema, busque ciertos mensajes de SELinux en la salida del comando **dmesg**:

   ```
   # dmesg | grep -i -e type=1300 -e type=1400
   ```

4. Incluso después de las tres comprobaciones anteriores, es posible que no haya encontrado nada. En este caso, los rechazos del CVA pueden ser silenciados debido a las reglas de **dontaudit**.
   Para desactivar temporalmente las reglas de **dontaudit**, permitiendo que se registren todos los rechazos:

   ```
   # semodule -DB
   ```

   Después de volver a ejecutar su escenario denegado y encontrar los mensajes de denegación utilizando los pasos anteriores, el siguiente comando habilita de nuevo las reglas de **dontaudit** en la política:

   ```
   # semodule -B
   ```

5. Si aplica los cuatro pasos anteriores y el problema sigue sin identificarse, considere si SELinux realmente bloquea su escenario:

   - Cambia al modo permisivo:

     ```
     # setenforce 0
     $ getenforce
     Permissive
     ```

   - Repite tu escenario.

   Si el problema sigue ocurriendo, algo diferente a SELinux está bloqueando su escenario.

5.2. ANÁLISIS DE LOS MENSAJES DE DENEGACIÓN DE SELINUX

Después de identificar que SELinux está bloqueando su escenario, es posible que tenga que analizar la causa raíz antes de elegir una solución.

Requisitos previos

- Los paquetes **policycoreutils-python-utils** y **setroubleshoot-server** están instalados en su sistema.

Procedimiento

1. Listar más detalles sobre una denegación registrada usando el comando **sealert**, por ejemplo:

   ```
   $ sealert -l "*"
   SELinux is preventing /usr/bin/passwd from write access on the file
   /root/test.

   *****  Plugin leaks (86.2 confidence) suggests ******************************

   If you want to ignore passwd trying to write access the test file,
   because you believe it should not need this access.
   Then you should report this as a bug.
   You can generate a local policy module to dontaudit this access.
   Do
   # ausearch -x /usr/bin/passwd --raw | audit2allow -D -M my-passwd
   # semodule -X 300 -i my-passwd.pp

   *****  Plugin catchall (14.7 confidence) suggests **************************

   ...

   Raw Audit Messages
   type=AVC msg=audit(1553609555.619:127): avc:  denied  { write } for
   pid=4097 comm="passwd" path="/root/test" dev="dm-0" ino=17142697
   scontext=unconfined_u:unconfined_r:passwd_t:s0-s0:c0.c1023
   tcontext=unconfined_u:object_r:admin_home_t:s0 tclass=file permissive=0

   ...

   Hash: passwd,passwd_t,admin_home_t,file,write
   ```

2. Si el resultado obtenido en el paso anterior no contiene sugerencias claras:

   - Habilite la auditoría de ruta completa para ver las rutas completas de los objetos a los que se ha accedido y para hacer visibles los campos adicionales de eventos de Auditoría de Linux:

     ```
     # auditctl -w /etc/shadow -p w -k shadow-write
     ```

   - Borrar la caché de **setroubleshoot**:

     ```
     # rm -f /var/lib/setroubleshoot/setroubleshoot.xml
     ```

   - Reproducir el problema.

   - Repite el paso 1.
     Después de terminar el proceso, desactive la auditoría de ruta completa:

     ```
     # auditctl -W /etc/shadow -p w -k shadow-write
     ```

3. Si **sealert** devuelve sólo sugerencias de **catchall** o sugiere añadir una nueva regla usando la herramienta **audit2allow**, compare su problema con los ejemplos listados y explicados en las denegaciones de SELinux en el registro de Auditoría .

Recursos adicionales

- La página de manual **sealert(8)**.

5.3. CORRECCIÓN DE LAS DENEGACIONES DE SELINUX ANALIZADAS

En la mayoría de los casos, las sugerencias proporcionadas por la herramienta **sealert** le dan la orientación correcta sobre cómo solucionar los problemas relacionados con la política de SELinux. Consulte Análisis de los mensajes de denegación de SELinux para obtener información sobre cómo utilizar **sealert** para analizar las denegaciones de SELinux.

Tenga cuidado cuando la herramienta sugiere usar la herramienta **audit2allow** para los cambios de configuración. No debe utilizar **audit2allow** para generar un módulo de política local como primera opción cuando vea una denegación de SELinux. La resolución de problemas debe comenzar con una comprobación de si hay un problema de etiquetado. El segundo caso más frecuente es que hayas cambiado la configuración de un proceso, y hayas olvidado avisar a SELinux de ello.

Problemas de etiquetado

Una causa común de problemas de etiquetado es cuando se utiliza un directorio no estándar para un servicio. Por ejemplo, en lugar de usar **/var/www/html/** para un sitio web, un administrador podría querer usar **/srv/myweb/**. En Red Hat Enterprise Linux, el directorio **/srv** está etiquetado con el tipo **var_t**. Los archivos y directorios creados en **/srv** heredan este tipo. También, los objetos recién creados en directorios de nivel superior, como **/myserver**, pueden ser etiquetados con el tipo **default_t**. SELinux impide que el servidor HTTP Apache (**httpd**) acceda a estos dos tipos. Para permitir el acceso, SELinux debe saber que los archivos en **/srv/myweb/** deben ser accesibles por **httpd**:

```
# semanage fcontext -a -t httpd_sys_content_t "/srv/myweb(/.*)?"
```

Este comando **semanage** agrega el contexto para el directorio **/srv/myweb/** y todos los archivos y directorios bajo él a la configuración del contexto de archivos de SELinux. La utilidad **semanage** no cambia el contexto. Como root, utilice la utilidad **restorecon** para aplicar los cambios:

```
# restorecon -R -v /srv/myweb
```

Contexto incorrecto

La utilidad **matchpathcon** comprueba el contexto de una ruta de archivo y lo compara con la etiqueta por defecto para esa ruta. El siguiente ejemplo demuestra el uso de **matchpathcon** en un directorio que contiene archivos incorrectamente etiquetados:

```
$ matchpathcon -V /var/www/html/*
/var/www/html/index.html has context unconfined_u:object_r:user_home_t:s0, should be
system_u:object_r:httpd_sys_content_t:s0
/var/www/html/page1.html has context unconfined_u:object_r:user_home_t:s0, should be
system_u:object_r:httpd_sys_content_t:s0
```

En este ejemplo, los archivos **index.html** y **page1.html** están etiquetados con el tipo **user_home_t**. Este tipo se utiliza para los archivos de los directorios personales de los usuarios. Si se utiliza el comando **mv** para mover archivos desde el directorio personal, los archivos pueden ser etiquetados con el tipo **user_home_t**. Este tipo no debería existir fuera de los directorios personales. Utilice la utilidad **restorecon** para restaurar dichos archivos a su tipo correcto:

```
# restorecon -v /var/www/html/index.html
restorecon reset /var/www/html/index.html context unconfined_u:object_r:user_home_t:s0-
>system_u:object_r:httpd_sys_content_t:s0
```

Para restaurar el contexto de todos los archivos de un directorio, utilice la opción **-R**:

```
# restorecon -R -v /var/www/html/
restorecon reset /var/www/html/page1.html context unconfined_u:object_r:samba_share_t:s0-
>system_u:object_r:httpd_sys_content_t:s0
restorecon reset /var/www/html/index.html context unconfined_u:object_r:samba_share_t:s0-
>system_u:object_r:httpd_sys_content_t:s0
```

Aplicaciones confinadas configuradas de forma no estándar

Los servicios pueden ejecutarse de diversas maneras. Para tener en cuenta esto, es necesario especificar cómo se ejecutan los servicios. Esto se puede lograr mediante booleanos de SELinux que permiten cambiar partes de la política de SELinux en tiempo de ejecución. Esto permite cambios, como permitir el acceso de los servicios a los volúmenes NFS, sin recargar o recompilar la política de SELinux. Además, la ejecución de servicios en números de puerto no predeterminados requiere la actualización de la configuración de la política mediante el comando **semanage**.

Por ejemplo, para permitir que el servidor HTTP Apache se comunique con MariaDB, active el booleano **httpd_can_network_connect_db**:

```
# setsebool -P httpd_can_network_connect_db on
```

Tenga en cuenta que la opción **-P** hace que la configuración persista a través de los reinicios del sistema.

Si se deniega el acceso a un servicio concreto, utiliza las utilidades **getsebool** y **grep** para ver si hay algún booleano que permita el acceso. Por ejemplo, utilice el comando **getsebool -a | grep ftp** para buscar booleanos relacionados con FTP:

```
$ getsebool -a | grep ftp
ftpd_anon_write --> off
```

```
ftpd_full_access --> off
ftpd_use_cifs --> off
ftpd_use_nfs --> off

ftpd_connect_db --> off
httpd_enable_ftp_server --> off
tftp_anon_write --> off
```

Para obtener una lista de booleanos y saber si están activados o desactivados, utilice el comando **getsebool -a**. Para obtener una lista de booleanos con su significado y saber si están activados o desactivados, instale el paquete **selinux-policy-devel** y utilice el comando **semanage boolean -l** como root.

Números de puerto

Dependiendo de la configuración de la política, los servicios sólo pueden ejecutarse en determinados números de puerto. Si se intenta cambiar el puerto en el que se ejecuta un servicio sin cambiar la política, es posible que el servicio no se inicie. Por ejemplo, ejecute el comando **semanage port -l | grep http** como root para listar los puertos relacionados con **http**:

```
# semanage port -l | grep http
http_cache_port_t              tcp      3128, 8080, 8118
http_cache_port_t              udp      3130
http_port_t                    tcp      80, 443, 488, 8008, 8009, 8443
pegasus_http_port_t            tcp      5988
pegasus_https_port_t           tcp      5989
```

El tipo de puerto **http_port_t** define los puertos en los que puede escuchar el Servidor HTTP Apache, que en este caso son los puertos TCP 80, 443, 488, 8008, 8009 y 8443. Si un administrador configura **httpd.conf** para que **httpd** escuche en el puerto 9876 (**Listen 9876**), pero la política no se actualiza para reflejar esto, el siguiente comando falla:

```
# systemctl start httpd.service
Job for httpd.service failed. See 'systemctl status httpd.service' and 'journalctl -xn' for details.

# systemctl status httpd.service
httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled)
   Active: failed (Result: exit-code) since Thu 2013-08-15 09:57:05 CEST; 59s ago
  Process: 16874 ExecStop=/usr/sbin/httpd $OPTIONS -k graceful-stop (code=exited,
status=0/SUCCESS)
  Process: 16870 ExecStart=/usr/sbin/httpd $OPTIONS -DFOREGROUND (code=exited,
status=1/FAILURE)
```

Un mensaje de denegación de SELinux similar al siguiente se registra en **/var/log/audit/audit.log**:

```
type=AVC msg=audit(1225948455.061:294): avc:  denied  { name_bind } for  pid=4997
comm="httpd" src=9876 scontext=unconfined_u:system_r:httpd_t:s0
tcontext=system_u:object_r:port_t:s0 tclass=tcp_socket
```

Para permitir que **httpd** escuche en un puerto que no está listado para el tipo de puerto **http_port_t**, utilice el comando **semanage port** para asignar una etiqueta diferente al puerto:

```
# semanage port -a -t http_port_t -p tcp 9876
```

La opción **-a** añade un nuevo registro; la opción **-t** define un tipo; y la opción **-p** define un protocolo. El último argumento es el número de puerto a añadir.

Casos de esquina, aplicaciones en evolución o rotas, y sistemas comprometidos

Las aplicaciones pueden contener errores, haciendo que SELinux deniegue el acceso. Además, las reglas de SELinux están evolucionando

Para estas situaciones, después de que se deniegue el acceso, utilice la utilidad **audit2allow** para crear un módulo de política personalizado que permita el acceso. Puedes informar de las reglas que faltan en la política de SELinux en **Red Hat Bugzilla**. Para Red Hat Enterprise Linux 8, cree errores contra el producto **Red Hat Enterprise Linux 8** y seleccione el componente **selinux-policy**. Incluya la salida de los comandos **audit2allow -w -a** y **audit2allow -a** en dichos informes de errores.

Si una aplicación pide privilegios de seguridad importantes, podría ser una señal de que la aplicación está comprometida. Utilice herramientas de detección de intrusos para inspeccionar ese comportamiento sospechoso.

La página web **Solution Engine** en el Portal del Cliente de Red Hat también puede proporcionar orientación en forma de un artículo que contiene una posible solución para el mismo problema o uno muy similar que usted tiene. Seleccione el producto y la versión pertinentes y utilice palabras clave relacionadas con SELinux, como *selinux* o *avc*, junto con el nombre de su servicio o aplicación bloqueada, por ejemplo: **selinux samba**.

5.4. DENEGACIONES DE SELINUX EN EL REGISTRO DE AUDITORÍA

El sistema de Auditoría de Linux almacena por defecto las entradas de registro en el archivo **/var/log/audit/audit.log**. Para listar sólo los registros relacionados con SELinux, utilice el comando **ausearch** con el parámetro de tipo de mensaje establecido en **AVC** y **AVC_USER** como mínimo, por ejemplo:

```
# ausearch -m AVC,USER_AVC,SELINUX_ERR,USER_SELINUX_ERR
```

Una entrada de denegación de SELinux en el archivo de registro de auditoría puede tener el siguiente aspecto:

```
type=AVC msg=audit(1395177286.929:1638): avc:  denied  { read } for  pid=6591 comm="httpd"
name="webpages" dev="0:37" ino=2112 scontext=system_u:system_r:httpd_t:s0
tcontext=system_u:object_r:nfs_t:s0 tclass=dir
```

Las partes más importantes de esta entrada son:

- **avc: denied** - la acción realizada por SELinux y registrada en Access Vector Cache (AVC)
- **{ read }** - la acción denegada
- **pid=6591** - el identificador del proceso del sujeto que intentó realizar la acción denegada
- **comm="httpd"** - el nombre del comando que se utilizó para invocar el proceso analizado
- **httpd_t** - el tipo de SELinux del proceso
- **nfs_t** - el tipo de SELinux del objeto afectado por la acción del proceso
- **tclass=dir** - la clase de objeto de destino

La entrada de registro anterior puede traducirse en:

*SELinux denied the **httpd** process with PID 6591 and the **httpd_t** type to read from a directory with the **nfs_t** type.*

El siguiente mensaje de denegación de SELinux se produce cuando el servidor HTTP Apache intenta acceder a un directorio etiquetado con un tipo para el conjunto Samba:

```
type=AVC msg=audit(1226874073.147:96): avc:  denied  { getattr } for  pid=2465 comm="httpd"
path="/var/www/html/file1" dev=dm-0 ino=284133 scontext=unconfined_u:system_r:httpd_t:s0
tcontext=unconfined_u:object_r:samba_share_t:s0 tclass=file
```

- **{ getattr }** - la entrada **getattr** indica que el proceso de origen estaba intentando leer la información de estado del archivo de destino. Esto ocurre antes de leer los archivos. SELinux niega esta acción porque el proceso accede al archivo y no tiene una etiqueta apropiada. Los permisos comúnmente vistos incluyen **getattr**, **read**, y **write**.
- **path="/var/www/html/file1"** - la ruta del objeto (objetivo) al que el proceso intentó acceder.
- **scontext="unconfined_u:system_r:httpd_t:s0"** - el contexto SELinux del proceso (fuente) que intentó la acción denegada. En este caso, es el contexto SELinux del servidor HTTP Apache, que se ejecuta con el tipo **httpd_t**.
- **tcontext="unconfined_u:object_r:samba_share_t:s0"** - el contexto SELinux del objeto (objetivo) al que el proceso intentó acceder. En este caso, es el contexto SELinux de **file1**.

Esta negación de SELinux se puede traducir en:

*SELinux denied the **httpd** process with PID 2465 to access the **/var/www/html/file1** file with the **samba_share_t** type, which is not accessible to processes running in the **httpd_t** domain unless configured otherwise.*

Recursos adicionales

- Para más información, consulte las páginas de manual **auditd(8)** y **ausearch(8)**.

5.5. INFORMACIÓN RELACIONADA

- El artículo Solución de problemas básicos de SELinux en CLI en el Portal del Cliente.
- La presentación ¿Qué está tratando de decirme SELinux? Las 4 causas principales de la presentación de errores de SELinux en Fedora People

CAPÍTULO 6. USO DE LA SEGURIDAD MULTINIVEL (MLS)

La política de seguridad multinivel (MLS) utiliza *levels* de autorización, tal y como fue diseñada originalmente por la comunidad de defensa estadounidense. La MLS cumple con un conjunto muy estrecho de requisitos de seguridad basados en la forma en que se gestiona la información en entornos rígidamente controlados como el militar.

El MLS es difícil de trabajar y no se adapta bien a los escenarios de casos generales.

6.1. SEGURIDAD MULTINIVEL (MLS)

La tecnología de seguridad multinivel (MLS) clasifica los datos utilizando los niveles de seguridad de la información:

- [más alto] Alto secreto
- [alto] Secreto
- [baja] Confidencial
- [más bajo] Sin clasificar

Las reglas que se aplican al flujo de datos operan desde los niveles inferiores a los superiores, y nunca a la inversa.

MLS denomina a los procesos como *subjects*, y a los archivos, dispositivos y otros componentes pasivos del sistema como *objects*. Tanto los sujetos como los objetos se etiquetan con un nivel de seguridad, que implica la autorización de un sujeto o la clasificación de un objeto.

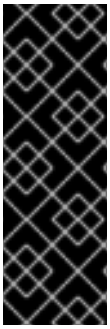
SELinux utiliza el modelo Bell-La Padula (BLP). Este modelo especifica cómo puede fluir la información dentro del sistema basándose en las etiquetas adjuntas a cada sujeto y objeto. En BLP, los procesos pueden leer el mismo nivel de seguridad o uno inferior, pero sólo pueden escribir en el mismo nivel de seguridad o uno superior.

El sistema siempre combina las reglas de acceso MLS con los permisos de acceso convencionales (permisos de archivo). Por ejemplo, si un usuario con un nivel de seguridad "Secreto" utiliza el Control de Acceso Discrecional (DAC) para bloquear el acceso a un archivo por parte de otros usuarios, esto también bloquea el acceso de los usuarios con un nivel de seguridad "Alto Secreto". Un nivel de seguridad superior no permite automáticamente navegar arbitrariamente por un sistema de archivos.

Los usuarios con autorizaciones de alto nivel no adquieren automáticamente derechos administrativos en los sistemas de varios niveles. Aunque pueden tener acceso a toda la información del ordenador, esto es diferente de tener derechos administrativos.

6.2. CAMBIANDO LA POLÍTICA DE SELINUX A MLS

Siga los siguientes pasos para cambiar la política de SELinux de dirigida a la seguridad multinivel (MLS).



IMPORTANTE

Red Hat no recomienda utilizar la política MLS en un sistema que esté ejecutando el sistema X Window. Además, cuando reetiqueta el sistema de archivos con etiquetas MLS, el sistema puede impedir el acceso a dominios confinados, lo que impide que su sistema se inicie correctamente. Por lo tanto, asegúrese de cambiar SELinux a modo permisivo antes de reetiquetar los archivos. En la mayoría de los sistemas, se ven muchas denegaciones de SELinux después de cambiar a MLS, y muchas de ellas no son triviales de arreglar.

Procedimiento

1. Instale el paquete **selinux-policy-mls**:

```
# yum install selinux-policy-mls
```

2. Abra el archivo **/etc/selinux/config** en un editor de texto de su elección, por ejemplo:

```
# vi /etc/selinux/config
```

3. Cambie el modo de SELinux de enforcing a permissive y cambie de la política dirigida a MLS:

```
SELINUX=permissive
SELINUXTYPE=mls
```

Guarde los cambios y salga del editor.

4. Antes de activar la política MLS, debe reetiquetar cada archivo del sistema de archivos con una etiqueta MLS:

```
# fixfiles -F onboot
System will relabel on next boot
```

5. Reinicia el sistema:

```
# reboot
```

6. Comprueba si hay denegaciones de SELinux:

```
# ausearch -m AVC,USER_AVC,SELINUX_ERR,USER_SELINUX_ERR -ts recent -i
```

Debido a que el comando anterior no cubre todos los escenarios, consulte Solución de problemas relacionados con SELinux para obtener orientación sobre la identificación, el análisis y la fijación de las denegaciones de SELinux.

7. Después de asegurarse de que no hay problemas relacionados con SELinux en su sistema, vuelva a poner SELinux en modo de aplicación cambiando la opción correspondiente en **/etc/selinux/config**:

```
SELINUX=aplicación
```

8. Reinicia el sistema:

```
# reboot
```

IMPORTANTE

Si su sistema no arranca o no puede iniciar sesión después de cambiar a MLS, añada el parámetro **enforcing=0** a la línea de comandos del kernel. Consulte Cambiar los modos de SELinux en el momento del arranque para obtener más información.

También tenga en cuenta que en MLS, los inicios de sesión SSH como el usuario **root** asignado al rol de SELinux **sysadm_r** difieren del inicio de sesión como **root** en **staff_r**. Antes de iniciar su sistema en MLS por primera vez, considere permitir los inicios de sesión SSH como **sysadm_r** estableciendo el booleano SELinux **ssh_sysadm_login** a **1**. Para habilitar **ssh_sysadm_login** más tarde, ya en MLS, debe iniciar sesión como **root** en **staff_r**, cambiar a **root** en **sysadm_r** usando el comando **newrole -r sysadm_r**, y luego establecer el booleano a **1**.

Pasos de verificación

1. Verifique que SELinux se ejecuta en modo de refuerzo:

```
# getenforce
Enforcing
```

2. Comprueba que el estado de SELinux devuelve el valor **mls**:

```
# sestatus | grep mls
Loaded policy name:             mls
```

Recursos adicionales

- Las páginas de manual **fixfiles(8)**, **setsebool(8)**, y **ssh_selinux(8)**.

CAPÍTULO 7. ESCRIBIR UNA POLÍTICA SELINUX PERSONALIZADA

Esta sección le guía sobre cómo escribir y utilizar una política personalizada que le permita ejecutar sus aplicaciones confinadas por SELinux.

7.1. POLÍTICAS PERSONALIZADAS DE SELINUX Y HERRAMIENTAS RELACIONADAS

Una política de seguridad SELinux es una colección de reglas SELinux. Una política es un componente central de SELinux y es cargada en el kernel por las herramientas de espacio de usuario de SELinux. El kernel impone el uso de una política SELinux para evaluar las solicitudes de acceso en el sistema. Por defecto, SELinux deniega todas las solicitudes excepto las que corresponden a las reglas especificadas en la política cargada.

Cada regla de política de SELinux describe una interacción entre un proceso y un recurso del sistema:

```
ALLOW apache_process apache_log:FILE READ;
```

Puede leer esta regla de ejemplo como *The **Apache** process can **read** its **logging file***. En esta regla, **apache_process** y **apache_log** son **labels**. Una política de seguridad SELinux asigna etiquetas a los procesos y define las relaciones con los recursos del sistema. De este modo, una política asigna entidades del sistema operativo a la capa de SELinux.

Las etiquetas de SELinux se almacenan como atributos extendidos de los sistemas de archivos, como **ext2**. Puedes listarlas usando la utilidad **getfattr** o un comando **ls -Z**, por ejemplo:

```
$ ls -Z /etc/passwd
system_u:object_r:passwd_file_t:s0 /etc/passwd
```

Donde **system_u** es un usuario SELinux, **object_r** es un ejemplo de rol SELinux, y **passwd_file_t** es un dominio SELinux.

La política de SELinux por defecto provista por los paquetes **selinux-policy** contiene reglas para aplicaciones y demonios que son parte de Red Hat Enterprise Linux 8 y son provistos por paquetes en sus repositorios. Las aplicaciones no descritas en una regla de esta política de distribución no están limitadas por SELinux. Para cambiar esto, tiene que modificar la política usando un módulo de política, el cual contiene definiciones y reglas adicionales.

En Red Hat Enterprise Linux 8, puede consultar la política SELinux instalada y generar nuevos módulos de política utilizando la herramienta **sepolicy**. Los scripts que **sepolicy** genera junto con los módulos de política siempre contienen un comando que utiliza la utilidad **restorecon**. Esta utilidad es una herramienta básica para arreglar problemas de etiquetado en una parte seleccionada de un sistema de archivos.

Recursos adicionales

- Para más información, consulte las páginas de manual **sepolicy(8)** y **getfattr(1)**.

7.2. CREACIÓN Y APLICACIÓN DE UNA POLÍTICA SELINUX PARA UNA APLICACIÓN PERSONALIZADA

Este procedimiento de ejemplo proporciona los pasos para confinar un simple demonio mediante SELinux. Sustituya el demonio por su aplicación personalizada y modifique la regla de ejemplo según los requisitos de esa aplicación y su política de seguridad.

Requisitos previos

- El paquete **policycoreutils-devel** y sus dependencias están instalados en su sistema.

Procedimiento

1. Para este procedimiento de ejemplo, prepare un demonio simple que abra el archivo **/var/log/messages** para escribir:

   a. Cree un nuevo archivo y ábralo en un editor de texto de su elección:

   ```
   $ vi mydaemon.c
   ```

   b. Inserte el siguiente código:

   ```
   #include <unistd.h>
   #include <stdio.h>

   FILE *f;

   int main(void)
   {
   while(1) {
   f = fopen("/var/log/messages","w");
           sleep(5);
           fclose(f);
       }
   }
   ```

   c. Compilar el archivo:

   ```
   $ gcc -o mydaemon mydaemon.c
   ```

   d. Cree un archivo de unidad **systemd** para su demonio:

   ```
   $ vi mydaemon.service
   [Unit]
   Description=Simple testing daemon

   [Service]
   Type=simple
   ExecStart=/usr/local/bin/mydaemon

   [Install]
   WantedBy=multi-user.target
   ```

   e. Instalar e iniciar el demonio:

   ```
   # cp mydaemon /usr/local/bin/
   # cp mydaemon.service /usr/lib/systemd/system
   ```

```
# systemctl start mydaemon
# systemctl status mydaemon
● mydaemon.service - Simple testing daemon
   Loaded: loaded (/usr/lib/systemd/system/mydaemon.service; disabled; vendor preset:
disabled)
   Active: active (running) since Sat 2020-05-23 16:56:01 CEST; 19s ago
 Main PID: 4117 (mydaemon)
    Tasks: 1
   Memory: 148.0K
   CGroup: /system.slice/mydaemon.service
           └─4117 /usr/local/bin/mydaemon

May 23 16:56:01 localhost.localdomain systemd[1]: Started Simple testing daemon.
```

f. Comprueba que el nuevo demonio no está limitado por SELinux:

```
$ ps -efZ | grep mydaemon
system_u:system_r:unconfined_service_t:s0 root 4117    1  0 16:56 ?        00:00:00
/usr/local/bin/mydaemon
```

2. Generar una política personalizada para el demonio:

```
$ sepolicy generate --init /usr/local/bin/mydaemon
Created the following files:
/home/example.user/mysepol/mydaemon.te # Type Enforcement file
/home/example.user/mysepol/mydaemon.if # Interface file
/home/example.user/mysepol/mydaemon.fc # File Contexts file
/home/example.user/mysepol/mydaemon_selinux.spec # Spec file
/home/example.user/mysepol/mydaemon.sh # Setup Script
```

3. Reconstruya la política del sistema con el nuevo módulo de política utilizando el script de configuración creado por el comando anterior:

```
# ./mydaemon.sh
Building and Loading Policy
+ make -f /usr/share/selinux/devel/Makefile mydaemon.pp
Compiling targeted mydaemon module
Creating targeted mydaemon.pp policy package
rm tmp/mydaemon.mod.fc tmp/mydaemon.mod
+ /usr/sbin/semodule -i mydaemon.pp
...
```

Tenga en cuenta que el script de configuración reetiqueta la parte correspondiente del sistema de archivos mediante el comando **restorecon**:

```
restorecon -v /usr/local/bin/mydaemon /usr/lib/systemd/system
```

4. Reinicie el demonio, y compruebe que ahora se ejecuta confinado por SELinux:

```
# systemctl restart mydaemon
$ ps -efZ | grep mydaemon
system_u:system_r:mydaemon_t:s0 root       8150       1  0 17:18 ?        00:00:00
/usr/local/bin/mydaemon
```

5. Dado que el demonio está ahora confinado por SELinux, éste también le impide acceder a **/var/log/messages**. Muestra el mensaje de denegación correspondiente:

```
# ausearch -m AVC -ts recent
...
type=AVC msg=audit(1590247112.719:5935): avc:  denied  { open } for  pid=8150
comm="mydaemon" path="/var/log/messages" dev="dm-0" ino=2430831
scontext=system_u:system_r:mydaemon_t:s0 tcontext=unconfined_u:object_r:var_log_t:s0
tclass=file permissive=1
...
```

6. También puede obtener información adicional utilizando la herramienta **sealert**:

```
$ sealert
SELinux is preventing mydaemon from open access on the file /var/log/messages.

 Plugin catchall (100. confidence) suggests *

If you believe that mydaemon should be allowed open access on the messages file by
default.
Then you should report this as a bug.
You can generate a local policy module to allow this access.
Do
allow this access for now by executing:
# ausearch -c 'mydaemon' --raw | audit2allow -M my-mydaemon
# semodule -X 300 -i my-mydaemon.pp

Additional Information:
Source Context                system_u:system_r:mydaemon_t:s0
Target Context                unconfined_u:object_r:var_log_t:s0
Target Objects                /var/log/messages [ file ]
Source                        mydaemon

...
```

7. **Utilice la herramienta audit2allow para sugerir cambios:**

```
$ ausearch -m AVC -ts recent | audit2allow -R

require {
 type mydaemon_t;
}

#============= mydaemon_t ==============
logging_write_generic_logs(mydaemon_t)
```

8. **Dado que las reglas sugeridas por audit2allow pueden ser incorrectas para ciertos casos, utilice sólo una parte de su salida para encontrar la interfaz de política correspondiente:**

```
$ grep -r "logging_write_generic_logs" /usr/share/selinux/devel/include/ | grep .if
/usr/share/selinux/devel/include/system/logging.if:interface(logging_write_generic_logs',
```

9. **Compruebe la definición de la interfaz:**

```
$ cat /usr/share/selinux/devel/include/system/logging.if
...
interface(logging_write_generic_logs',
        gen_require(`
                type var_log_t;
        ')

        files_search_var($1)
        allow $1 var_log_t:dir list_dir_perms;
        write_files_pattern($1, var_log_t, var_log_t)
')
...
```

10. **En este caso, puede utilizar la interfaz sugerida. Añada la regla correspondiente a su archivo de aplicación de tipos:**

```
$ echo "logging_write_generic_logs(mydaemon_t)" >> mydaemon.te
```

También puede añadir esta regla en lugar de utilizar la interfaz:

```
$ echo "allow mydaemon_t var_log_t:file { open write getattr };" >> mydaemon.te
```

11. **Vuelva a instalar la política:**

```
# ./mydaemon.sh
Building and Loading Policy
+ make -f /usr/share/selinux/devel/Makefile mydaemon.pp
Compiling targeted mydaemon module
Creating targeted mydaemon.pp policy package
rm tmp/mydaemon.mod.fc tmp/mydaemon.mod
+ /usr/sbin/semodule -i mydaemon.pp
...
```

Pasos de verificación

1. **Compruebe que su aplicación se ejecuta confinada por SELinux, por ejemplo:**

```
$ ps -efZ | grep mydaemon
system_u:system_r:mydaemon_t:s0 root        8150       1  0 17:18 ?        00:00:00
/usr/local/bin/mydaemon
```

2. **Compruebe que su aplicación personalizada no causa ninguna denegación de SELinux:**

```
# ausearch -m AVC -ts recent
<no matches>
```

Recursos adicionales

- **Para más información, consulte las páginas de manual sepolgen(8), ausearch(8), audit2allow(1), audit2why(1), sealert(8) y restorecon(8).**

7.3. RECURSOS ADICIONALES

- Para obtener más detalles y más ejemplos, consulte el taller de políticas de SELinux

CAPÍTULO 8. CREACIÓN DE POLÍTICAS SELINUX PARA CONTENEDORES

RHEL 8 proporciona una herramienta para generar políticas SELinux para contenedores utilizando el paquete **udica**. Con **udica**, puede crear una política de seguridad a medida para controlar mejor cómo un contenedor accede a los recursos del sistema anfitrión, como el almacenamiento, los dispositivos y la red. Esto le permite endurecer sus despliegues de contenedores contra las violaciones de la seguridad y también simplifica la consecución y el mantenimiento del cumplimiento normativo.

8.1. INTRODUCCIÓN AL GENERADOR DE POLÍTICAS SELINUX DE UDICA

Para simplificar la creación de nuevas políticas de SELinux para contenedores personalizados, RHEL 8 proporciona la utilidad **udica**. Puede utilizar esta herramienta para crear una política basada en una inspección del archivo de notación de objetos JavaScript (JSON) del contenedor, que contiene definiciones de capacidades Linux, puntos de montaje y puertos. La herramienta combina consecuentemente las reglas generadas usando los resultados de la inspección con las reglas heredadas de un bloque de SELinux Common Intermediate Language (CIL) especificado.

El proceso de generación de la política SELinux para un contenedor utilizando **udica** tiene tres partes principales:

1. Análisis del archivo de especificaciones del contenedor en formato JSON
2. Encontrar las reglas adecuadas para permitir el uso de los resultados de la primera parte
3. Generación de la política final de SELinux

Durante la fase de análisis, **udica** busca las capacidades de Linux, los puertos de red y los puntos de montaje.

Basándose en los resultados, **udica** detecta qué capacidades de Linux necesita el contenedor y crea una regla SELinux que permite todas estas capacidades. Si el contenedor se vincula a un puerto específico, **udica** utiliza las bibliotecas de espacio de usuario de SELinux para obtener la etiqueta SELinux correcta de un puerto que es utilizado por el contenedor inspeccionado.

Después, **udica** detecta qué directorios están montados en el espacio de nombres del sistema de archivos del contenedor desde el host.

La función de herencia de bloques del CIL permite a **udica** crear plantillas de SELinux*allow rules* centradas en una acción específica, por ejemplo:

- *allow accessing home directories*
- *allow accessing log files*
- *allow accessing communication with Xserver.*

Estas plantillas se llaman bloques y la política final de SELinux se crea fusionando los bloques.

Recursos adicionales

- Para más detalles sobre el proceso de generación de una política SELinux con **udica**, consulte el artículo Generar políticas SELinux para contenedores con udicaRed Hat Blog.

8.2. CREACIÓN Y USO DE UNA POLÍTICA SELINUX PARA UN CONTENEDOR PERSONALIZADO

Para generar una política de seguridad SELinux para un contenedor personalizado, siga los pasos de este procedimiento.

Requisitos previos

- La herramienta **podman** para la gestión de contenedores está instalada. Si no lo está, utilice el comando **yum install podman**.
- Un contenedor Linux personalizado - *ubi8* en este ejemplo.

Procedimiento

1. Instale el paquete **udica**:

```
# yum install -y udica
```

Como alternativa, instale el módulo **container-tools**, que proporciona un conjunto de paquetes de software para contenedores, incluido **udica**:

```
# yum module install -y container-tools
```

2. Inicia el contenedor *ubi8* que monta el directorio/**home** con permisos de sólo lectura y el directorio /**var**/**spool** con permisos de lectura y escritura. El contenedor expone el puerto*21*.

```
# podman run --env container=podman -v /home:/home:ro -v /var/spool:/var/spool:rw -p
21:21 -it ubi8 bash
```

Tenga en cuenta que ahora el contenedor se ejecuta con el tipo de SELinux **container_t**. Este tipo es un dominio genérico para todos los contenedores en la política de SELinux y podría ser demasiado estricto o demasiado flojo para su escenario.

3. Introduzca el comando **podman ps** para obtener el ID del contenedor:

```
# podman ps
CONTAINER ID  IMAGE                                 COMMAND  CREATED         STATUS
PORTS  NAMES
37a3635afb8f  registry.access.redhat.com/ubi8:latest  bash     15 minutes ago  Up 15
minutes ago          heuristic_lewin
```

4. Cree un archivo JSON contenedor y utilice **udica** para crear un módulo de política basado en la información del archivo JSON:

```
# podman inspect 37a3635afb8f > container.json
# udica -j container.json my_container
Policy my_container with container id 37a3635afb8f created!
[...]
```

Alternativamente:

```
# podman inspect 37a3635afb8f | udica my_container
```

```
Policy my_container with container id 37a3635afb8f created!

Please load these modules using:
# semodule -i my_container.cil
/usr/share/udica/templates/{base_container.cil,net_container.cil,home_container.cil}

Restart the container with: "--security-opt label=type:my_container.process" parameter
```

5. **Como sugiere la salida de udica en el paso anterior, cargue el módulo de política:**

```
# semodule -i my_container.cil
/usr/share/udica/templates/{base_container.cil,net_container.cil,home_container.cil}
```

6. **Detenga el contenedor y vuelva a iniciarlo con la opción --security-opt label=type:my_container.process:**

```
# podman stop 37a3635afb8f
# podman run --security-opt label=type:my_container.process -v /home:/home:ro -v
/var/spool:/var/spool:rw -p 21:21 -it ubi8 bash
```

Pasos de verificación

1. **Compruebe que el contenedor funciona con el tipo my_container.process:**

```
# ps -efZ | grep my_container.process
unconfined_u:system_r:container_runtime_t:s0-s0:c0.c1023 root 2275 434  1 13:49 pts/1
00:00:00 podman run --security-opt label=type:my_container.process -v /home:/home:ro -v
/var/spool:/var/spool:rw -p 21:21 -it ubi8 bash
system_u:system_r:my_container.process:s0:c270,c963 root 2317 2305  0 13:49 pts/0
00:00:00 bash
```

2. **Compruebe que SELinux permite ahora el acceso a los puntos de montaje /home y /var/spool:**

```
[root@37a3635afb8f /]# cd /home
[root@37a3635afb8f home]# ls
username
[root@37a3635afb8f ~]# cd /var/spool/
[root@37a3635afb8f spool]# touch test
[root@37a3635afb8f spool]#
```

3. **Compruebe que SELinux permite el enlace sólo con el puerto 21:**

```
[root@37a3635afb8f /]# yum install nmap-ncat
[root@37a3635afb8f /]# nc -lvp 21
Ncat: Version 7.60 ( https://nmap.org/ncat )
Ncat: Generating a temporary 1024-bit RSA key. Use --ssl-key and --ssl-cert to use a
permanent one.
Ncat: SHA-1 fingerprint: 6EEC 102E 6666 5F96 CC4F E5FA A1BE 4A5E 6C76 B6DC
Ncat: Listening on :::21
Ncat: Listening on 0.0.0.0:21

[root@37a3635afb8f /]# nc -lvp 80
Ncat: Version 7.60 ( https://nmap.org/ncat )
```

```
Ncat: Generating a temporary 1024-bit RSA key. Use --ssl-key and --ssl-cert to use a
permanent one.
Ncat: SHA-1 fingerprint: 6EEC 102E 6666 5F96 CC4F E5FA A1BE 4A5E 6C76 B6DC
Ncat: bind to :::80: Permission denied. QUITTING.
```

Recursos adicionales

- Para más información, consulte las páginas de manual **udica(8)** y **podman(1)**.
- Para obtener orientación sobre cómo empezar con los contenedores en RHEL y cómo trabajar con imágenes de contenedores, consulte el documento Construir, ejecutar y gestionar contenedores.

8.3. RECURSOS ADICIONALES

- Para más detalles sobre la creación de políticas con **udica**, vea la página udica - Generar políticas SELinux para contenedores.

CAPÍTULO 9. IMPLANTACIÓN DE LA MISMA CONFIGURACIÓN DE SELINUX EN VARIOS SISTEMAS

Esta sección proporciona dos formas recomendadas para desplegar su configuración verificada de SELinux en múltiples sistemas:

- Uso de los roles del sistema RHEL y Ansible
- Utilización de los comandos de exportación e importación de **semanage** en sus scripts

9.1. INTRODUCCIÓN AL ROL DEL SISTEMA SELINUX

RHEL System Roles es una colección de roles y módulos de Ansible que proporcionan una interfaz de configuración consistente para gestionar remotamente múltiples sistemas RHEL. El rol de sistema SELinux permite las siguientes acciones:

- Limpieza de las modificaciones de las políticas locales relacionadas con los booleanos de SELinux, los contextos de archivos, los puertos y los inicios de sesión.
- Configuración de booleanos de la política SELinux, contextos de archivos, puertos e inicios de sesión.
- Restauración de contextos de archivo en archivos o directorios especificados.

La siguiente tabla proporciona una visión general de las variables de entrada disponibles en el rol del sistema SELinux.

Tabla 9.1. Variables de rol del sistema SELinux

Variable de rol	Descripción	Alternativa CLI
selinux_policy	Elige una política de protección de procesos específicos o una protección de seguridad multinivel.	**SELINUXTYPE** en **/etc/selinux/config**
selinux_state	Cambia los modos de SELinux. Ver **ansible-doc selinux**	**setenforce** y **SELINUX** en **/etc/selinux/config**.
selinux_booleans	Activa y desactiva los booleanos de SELinux. Véase **ansible-doc seboolean**.	**setsebool**
selinux_fcontextos	Añade o elimina una asignación de contexto de archivo SELinux. Véase **ansible-doc sefcontext**.	**semanage fcontext**
selinux_restore_dirs	Restaura las etiquetas de SELinux en el árbol del sistema de archivos.	**restorecon -R**

Variable de rol	Descripción	Alternativa CLI
selinux_ports	Establece las etiquetas SELinux en los puertos. Véase **ansible-doc seport**.	**semanage port**
selinux_logins	Establece los usuarios en el mapeo de usuarios de SELinux. Véase **ansible-doc selogin**.	**semanage login**

El libro de jugadas de ejemplo **/usr/share/doc/rhel-system-roles/selinux/example-selinux-playbook.yml** instalado por el paquete**rhel-system-roles** demuestra cómo establecer la política dirigida en modo de aplicación. El libro de jugadas también aplica varias modificaciones de la política local y restaura los contextos de los archivos en el directorio **/tmp/test_dir/**.

Recursos adicionales

- Para una referencia detallada sobre las variables de rol de SELinux, instale el paquete **rhel-system-roles**, y vea los archivos**README.md** o **README.html** en el directorio **/usr/share/doc/rhel-system-roles/selinux/**.
- Para obtener más información sobre las funciones del sistema RHEL, consulte Introducción a las funciones del sistema RHEL

9.2. USO DEL ROL DE SISTEMA SELINUX PARA APLICAR LA CONFIGURACIÓN DE SELINUX EN VARIOS SISTEMAS

Siga los pasos para preparar y aplicar un playbook de Ansible con su configuración de SELinux verificada.

Requisitos previos

- Su suscripción a Red Hat Ansible Engine está conectada al sistema. Consulte el artículo Cómo descargar e instalar Red Hat AnsibleEngine para obtener más información.

Procedimiento

1. Habilitar el repositorio RHEL Ansible, por ejemplo:

```
# subscription-manager repos --enable ansible-2-for-rhel-8-x86_64-rpms
```

2. Instale el motor Ansible:

```
# yum install ansible
```

3. Instalar los roles del sistema RHEL:

```
# yum install rhel-system-roles
```

4. Aplique su libro de jugadas con un rol de sistema SELinux.
   El siguiente comando aplica un playbook de ejemplo, que forma parte del paquete **rhel-system-roles**. Puede utilizar este libro de jugadas como plantilla:

```
# ansible-playbook -i host1,host2,host3 /usr/share/doc/rhel-system-roles/selinux/example-selinux-playbook.yml
```

Recursos adicionales

- Para más información, instale el paquete **rhel-system-roles** y consulte los directorios **/usr/share/doc/rhel-system-roles/selinux/** y **/usr/share/ansible/roles/rhel-system-roles.selinux/**.

9.3. TRANSFERENCIA DE LA CONFIGURACIÓN DE SELINUX A OTRO SISTEMA CON SEMANAGE

Utilice los siguientes pasos para transferir su configuración personalizada y verificada de SELinux entre sistemas basados en RHEL 8.

Requisitos previos

- El paquete **policycoreutils-python-utils** está instalado en su sistema.

Procedimiento

1. Exporte su configuración de SELinux verificada:

```
# semanage export -f ./my-selinux-settings.mod
```

2. Copie el archivo con la configuración en el nuevo sistema:

```
# scp ./my-selinux-settings.mod new-system-hostname:
```

3. Inicie sesión en el nuevo sistema:

```
$ ssh root@new-system-hostname
```

4. Importe la configuración en el nuevo sistema:

```
new-system-hostname# semanage import -f ./my-selinux-settings.mod
```

Recursos adicionales

- **semanage-export(8)** y **semanage-import(8)** páginas man