Fedora 13

Linux con Seguridad Mejorada

Guía del Usuario

Murray McAllister

Scott Radvan

Daniel Walsh

Dominick Grift

Eric Paris

James Morris

Fedora 13 Linux con Seguridad Mejorada
Guía del Usuario
Edición 1.4

Autor	Murray McAllister	*mmcallis@redhat.com*
Autor	Scott Radvan	*sradvan@redhat.com*
Autor	Daniel Walsh	*dwalsh@redhat.com*
Autor	Dominick Grift	*domg472@gmail.com*
Autor	Eric Paris	*eparis@parisplace.org*
Autor	James Morris	*jmorris@redhat.com*

The SELinux User Guide assists users and administrators in managing and using Security-Enhanced Linux®.

Prefacio **v**
1. Convenciones del Documento v
1.1. Convenciones Tipográficas v
1.2. Convenciones del documento vii
1.3. Notas y Advertencias vii
2. ¡Necesitamos sus comentarios! viii

1. Información de Marca Comercial **1**
1.1. Source Code 1

2. Introducción **3**
2.1. Beneficios de usar SELinux 4
2.2. Ejemplos 5
2.3. Arquitectura de SELinux 6
2.4. SELinux en otros Sistemas Operativos 6

3. Contextos de SELinux **7**
3.1. Transiciones de Dominios 8
3.2. Contextos de SELinux para los Procesos 9
3.3. Contextos de SELinux para los Usuarios 10

4. Política Destinado **11**
4.1. Procesos Confinados 11
4.2. Procesos no Confinados 13
4.3. Usuarios Confinados y no Confinados 17

5. Trabajando con SELinux **19**
5.1. Paquetes de SELinux 19
5.2. Qué Archivo Log se usa 20
5.3. Archivo de Configuración Principal 21
5.4. Habilitando y Deshabilitando SELinux 22
5.4.1. Habilitando SELinux 22
5.4.2. Deshabilitando SELinux 25
5.5. Modos de SELinux 25
5.6. Booleanos 26
5.6.1. Listando los Booleanos 26
5.6.2. Configurando los Booleanos 27
5.6.3. Booleanos para NFS y CIFS 27
5.7. Contextos de SELinux - Etiquetado de Archivos 28
5.7.1. Cambios Temporales: chcon 29
5.7.2. Cambios Persistentes: semanage fcontext 31
5.8. Los tipos file_t y default_t 35
5.9. Montaje de Sistemas de Archivos 36
5.9.1. Montajes de Contexto 36
5.9.2. Cambio del Contexto Predeterminado 37
5.9.3. Montando un Sistema de Archivos NFS 37
5.9.4. Montajes NFS Múltiples 38
5.9.5. Haciendo Persistente los Contextos de Montajes 39
5.10. Mantención de las Etiquetas de SELinux 39
5.10.1. Copia de Directorios y Archivos 39
5.10.2. Movimiento de Archivos y Directorios 41
5.10.3. Chequeando el Contexto SELinux Predeterminado 42
5.10.4. Archivando archivos con tar 43
5.10.5. Archivando archivos con tar 44

6. Confinando a los Usuarios **47**
6.1. Linux y los Mapeos de Usuarios de SELinux 47
6.2. Confinando Usuarios Nuevos de Linux: useradd 47
6.3. Confinando Usuarios Linux Existentes: semanage login 48
6.4. Cambiando el Mapeo Predeterminado 50
6.5. xguest: Modo Kiosk 51
6.6. Booleanos para que los Usuarios Ejecuten Aplicaciones 51

7. Solución a Problemas **53**
7.1. Qué pasa cuando el Acceso es Denegado 53
7.2. Tres Principales Causas de Problemas 54
7.2.1. Problemas de Etiquetados 54
7.2.2. ¿Cómo se Ejecutan los Servicios Confinados? 55
7.2.3. Evolucionando las Reglas y las Aplicaciones Rotas 57
7.3. Corrección de Problemas 57
7.3.1. Permisos de Linux 57
7.3.2. Posibles Causas de las Negaciones Silenciosas 58
7.3.3. Páginas de Manual para Servicios 58
7.3.4. Dominios Permisivos 59
7.3.5. Búsqueda y Revisión de Negaciones 61
7.3.6. Raw Audit Messages 63
7.3.7. Mensajes sealert 64
7.3.8. Permitiendo el Acceso: audit2allow 66

8. Información Adicional **69**
8.1. Contributors 69
8.2. Other Resources 69

A. Revision History **71**

Prefacio

The Fedora 13 SELinux User Guide is for people with minimal or no experience with SELinux. Although system administration experience is not necessary, content in this guide is written for system administration tasks. This guide provides an introduction to fundamental concepts and practical applications of SELinux. After reading this guide you should have an intermediate understanding of SELinux.

Gracias a todos los que nos alentaron, ofrecieron ayuda y lo probaron - la ayuda es muy apreciada. Agradecimientos muy especiales a:

- Dominick Grift, Stephen Smalley y Russell Coker por sus contribuciones, ayuda y paciencia.
- Karsten Wade for his help, adding a component for this guide to *Red Hat Bugzilla*[1], and sorting out web hosting on *http://docs.fedoraproject.org/*.
- Al *Equipo de Infraestructura de Fedora*[2] por proveer el alojamiento.
- Jens-Ulrik Petersen por asegurar que la oficina de Brisbane de Red Hat tenga espejos de Fedora actualizados.

1. Convenciones del Documento

Este manual utiliza varias convenciones para resaltar algunas palabras y frases y llamar la atención sobre ciertas partes específicas de información.

En ediciones PDF y de papel, este manual utiliza tipos de letra procedentes de *Liberation Fonts*[3]. Liberation Fonts también se utilizan en ediciones de HTML si están instalados en su sistema. Si no, se muestran tipografías alternativas pero equivalentes. Nota: Red Hat Enterprise Linux 5 y siguientes incluyen Liberation Fonts predeterminadas.

1.1. Convenciones Tipográficas

Se utilizan cuatro convenciones tipográficas para llamar la atención sobre palabras o frases específicas. Dichas convenciones y las circunstancias en que se aplican son las siguientes:

`Negrita monoespaciado`

Utilizada para resaltar la entrada del sistema, incluyendo comandos de shell, nombres de archivo y rutas. También se utiliza para resaltar teclas claves y combinaciones de teclas. Por ejemplo:

> Para ver el contenido del archivo **`my_next_bestselling_novel`** en su directorio actual de trabajo, escriba el comando **`cat my_next_bestselling_novel`** en el intérprete de comandos de shell y pulse **`Enter`** para ejecutar el comando.

El ejemplo anterior incluye un nombre de archivo, un comando de shell y una tecla clave. Todo se presenta en negrita-monoespaciado y distinguible gracias al contexto.

Las combinaciones de teclas se pueden distinguir de las teclas claves mediante el guión que conecta cada parte de una combinación de tecla. Por ejemplo:

> Pulse **`Enter`** para ejecutar el comando.

[3] https://fedorahosted.org/liberation-fonts/

> Pulse **Control**+**Alt**+**F1** para cambiar a la primera terminal virtual. Pulse **Control**+**Alt**+**F7** para volver a su sesión de Ventanas-X.

La primera oración resalta la tecla clave determinada que se debe pulsar. La segunda resalta dos conjuntos de tres teclas claves que deben ser presionadas simultáneamente.

Si se discute el código fuente, los nombres de las clase, los métodos, las funciones, los nombres de variables y valores de retorno mencionados dentro de un párrafo serán presentados en **Negrita-monoespaciado**. Por ejemplo:

> Las clases de archivo relacionadas incluyen **filename** para sistema de archivos, **file** para archivos y **dir** para directorios. Cada clase tiene su propio conjunto asociado de permisos.

Negrita proporcional

Esta denota palabras o frases encontradas en un sistema, incluyendo nombres de aplicación, texto de cuadro de diálogo, botones etiquetados, etiquetas de cajilla de verificación y botón de radio; títulos de menú y títulos del sub-menú. Por ejemplo:

> Seleccionar **Sistema** → **Preferencias** → **Ratón** desde la barra del menú principal para lanzar **Preferencias de Ratón**. En la pestaña de **Botones**, haga clic en la cajilla **ratón de mano izquierda** y luego haga clic en **Cerrar** para cambiar el botón principal del ratón de la izquierda a la derecha (adecuando el ratón para la mano izquierda).
>
> Para insertar un caracter especial en un archivo de **gedit**, seleccione desde la barra del menú principal **Aplicaciones** → **Accesorios** → **Mapa de caracteres**. Luego, desde la barra del menú **mapa de caracteres** elija **Búsqueda** → **Hallar…**, teclee el nombre del caracter en el campo **Búsqueda** y haga clic en **Siguiente**. El caracter buscado se resaltará en la **Tabla de caracteres**. Haga doble clic en este caracter resaltado para colocarlo en el campo de **Texto para copiar** y luego haga clic en el botón de **Copiar**. Ahora regrese a su documento y elija **Editar** → **Pegar** desde la barra de menú de **gedit**.

El texto anterior incluye nombres de aplicación; nombres y elementos del menú de todo el sistema; nombres de menú de aplicaciones específicas y botones y texto hallados dentro de una interfaz gráfica de usuario, todos presentados en negrita proporcional y distinguibles por contexto.

Itálicas-negrita monoespaciado o ***Itálicas-negrita proporcional***

Ya sea negrita monoespaciado o negrita proporcional, la adición de itálicas indica texto reemplazable o variable. Las itálicas denotan texto que usted no escribe literalmente o texto mostrado que cambia dependiendo de la circunstancia. Por ejemplo:

> Para conectar a una máquina remota utilizando ssh, teclee **ssh *nombredeusuario*@*dominio.nombre*** en un intérprete de comandos de shell. Si la máquina remota es **example.com** y su nombre de usuario en esa máquina es john, teclee **ssh john@example.com**.
>
> El comando **mount -o remount *file-system*** remonta el sistema de archivo llamado. Por ejemplo, para volver a montar el sistema de archivo **/home**, el comando es **mount -o remount /home**.
>
> Para ver la versión de un paquete actualmente instalado, utilice el comando **rpm -q *paquete***. Éste entregará el resultado siguiente: ***paquete-versión-lanzamiento***.

Observe las palabras en itálicas- negrita sobre — nombre de usuario, domain.name, sistema de archivo, paquete, versión y lanzamiento. Cada palabra es un marcador de posición, tanto para el texto que usted escriba al ejecutar un comando como para el texto mostrado por el sistema.

Aparte del uso estándar para presentar el título de un trabajo, las itálicas denotan el primer uso de un término nuevo e importante. Por ejemplo:

> Publican es un sistema de publicación de *DocBook*.

1.2. Convenciones del documento

Los mensajes de salida de la terminal o fragmentos de código fuente se distinguen visualmente del texto circundante.

Los mensajes de salida enviados a una terminal se muestran en **`romano monoespaciado`** y se presentan así:

```
books        Desktop   documentation  drafts  mss    photos   stuff  svn
books_tests  Desktop1  downloads      images  notes  scripts  svgs
```

Los listados de código fuente también se muestran en **`romano monoespaciado`**, pero se presentan y resaltan de la siguiente manera:

```
package org.jboss.book.jca.ex1;

import javax.naming.InitialContext;

public class ExClient
{
   public static void main(String args[])
       throws Exception
   {
      InitialContext iniCtx = new InitialContext();
      Object         ref    = iniCtx.lookup("EchoBean");
      EchoHome       home   = (EchoHome) ref;
      Echo           echo   = home.create();

      System.out.println("Created Echo");

      System.out.println("Echo.echo('Hello') = " + echo.echo("Hello"));
   }
}
```

1.3. Notas y Advertencias

Finalmente, utilizamos tres estilos visuales para llamar la atención sobre la información que de otro modo se podría pasar por alto.

Nota

Una nota es una sugerencia, atajo o enfoque alternativo para una tarea determinada. Ignorar una nota no debería tener consecuencias negativas, pero podría perderse de algunos trucos que pueden facilitarle las cosas.

Importante

Important boxes detail things that are easily missed: configuration changes that only apply to the current session, or services that need restarting before an update will apply. Ignoring a box labeled 'Important' won't cause data loss but may cause irritation and frustration.

Advertencia

Las advertencias no deben ignorarse. Ignorarlas muy probablemente ocasionará pérdida de datos.

2. ¡Necesitamos sus comentarios!

Si encuentra un error tipográfico en este manual o si sabe de alguna manera de mejorarlo, nos gustaría escuchar sus sugerencias. Por favor complete un reporte en Bugzilla: *http://bugzilla.redhat.com/bugzilla/* usando el producto **Fedora Documentation**.

Cuando envíe un reporte de error no olvide mencionar el identificador del manual: *selinux-user-guide*

Si tiene una sugerencia para mejorar la documentación, intente ser tan específico como sea posible cuando describa su sugerencia. Si ha encontrado un error, por favor incluya el número de sección y parte del texto que rodea el error para que podamos encontrarlo más fácilmente.

Información de Marca Comercial

Linux® es una marca comercial registrada de Linus Torvalds en los EEUU y en otros países.

UNIX es una marca comercial registrada de El Grupo Abierto.

Type Enforcement (Obligación de Tipos) es una marca comercial de Secure Computing, LLC, una subsidiaria de McAfee, Inc., registrada en los EEUU y en otros países. Ni McAfee ni Secure Computing, LLC, ha consentido el uso o referencia de esta marca comercial para el autor fuera de esta guía.

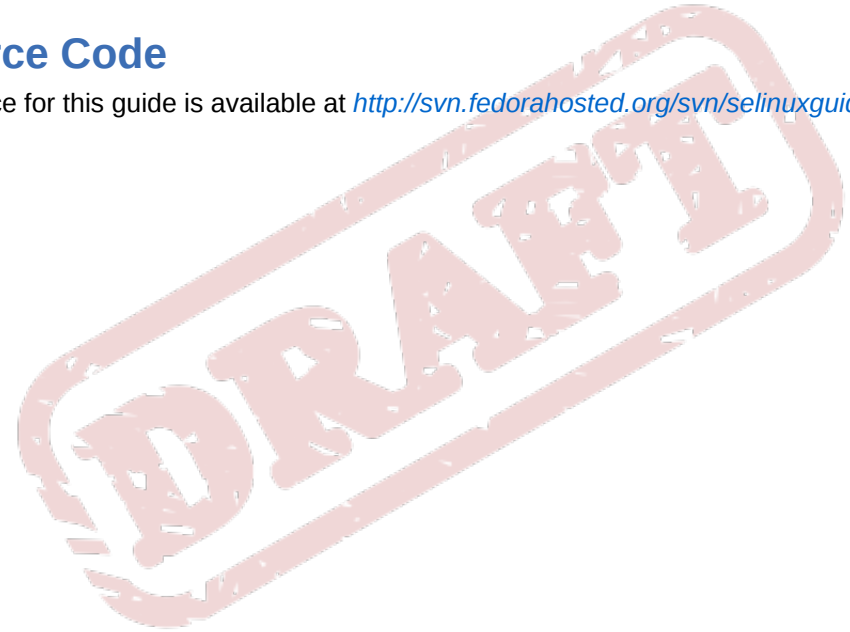
Apache es una marca comercial de La Fundación de Software Apache.

MySQL es una marca comercial o marca comercial registrada de MySQL AB en los EEUU y en otros países.

Other products mentioned may be trademarks of their respective corporations.

1.1. Source Code

The XML source for this guide is available at *http://svn.fedorahosted.org/svn/selinuxguide/*




DRAFT

Introducción

Security-Enhanced Linux (SELinux) is an implementation of a *mandatory access control* mechanism in the Linux kernel, checking for allowed operations after standard *discretionary access controls* are checked. It was created by the National Security Agency and can enforce rules on files and processes in a Linux system, and on their actions, based on defined policy.

When using SELinux, files, including directories and devices, are referred to as objects. Processes, such as a user running a command or the Mozilla® Firefox® application, are referred to as subjects. Most operating systems use a Discretionary Access Control (DAC) system that controls how subjects interact with objects, and how subjects interact with each other. On operating systems using DAC, users control the permissions of files (objects) that they own. For example, on Linux® operating systems, users could make their home directories world-readable, giving users and processes (subjects) access to potentially sensitive information, with no further protection over this unwanted action.

Relying on DAC mechanisms alone is fundamentally inadequate for strong system security. DAC access decisions are only based on user identity and ownership, ignoring other security-relevant information such as the role of the user, the function and trustworthiness of the program, and the sensitivity and integrity of the data. Each user has complete discretion over their files, making it impossible to enforce a system-wide security policy. Furthermore, every program run by a user inherits all of the permissions granted to the user and is free to change access to the user's files, so no protection is provided against malicious software. Many system services and privileged programs must run with coarse-grained privileges that far exceed their requirements, so that a flaw in any one of these programs could be exploited to obtain further system access.[1]

The following is an example of permissions used on Linux operating systems that do not run Security-Enhanced Linux (SELinux). The permissions and output in these examples may differ from your system. Use the **`ls -l`** command to view file permissions:

```
$ ls -l file1
-rwxrw-r-- 1 user1 group1 0 2009-08-30 11:03 file1
```

Los primeros tres bits de permisos, **`rwx`**, controlan el acceso que el usuario Linux **`usuario1`** (en este caso, el dueño) tiene para el **`archivo1`**. Los siguientes tres bits de permisos, **`rw-`**, controlan el acceso que el grupo Linux **`grupo1`** tiene para el **`archivo1`**. Los últimos tres bits de permisos, **`r--`**, controlan el acceso que todo el mundo tiene para el **`archivo1`**, que incluyen a todos los usuarios y procesos.

Security-Enhanced Linux (SELinux) adds Mandatory Access Control (MAC) to the Linux kernel, and is enabled by default in Fedora. A general purpose MAC architecture needs the ability to enforce an administratively-set security policy over all processes and files in the system, basing decisions on labels containing a variety of security-relevant information. When properly implemented, it enables a system to adequately defend itself and offers critical support for application security by protecting against the tampering with, and bypassing of, secured applications. MAC provides strong separation of applications that permits the safe execution of untrustworthy applications. Its ability to limit the privileges associated with executing processes limits the scope of potential damage that can result from the exploitation of vulnerabilities in applications and system services. MAC enables information

"Integrating Flexible Support for Security Policies into the Linux Operating System", by Peter Loscocco and Stephen Smalley. This paper was originally prepared for the National Security Agency and is, consequently, in the public domain. Refer to the *original paper* [http://www.nsa.gov/research/_files/selinux/papers/freenix01/index.shtml] for details and the document as it was first released. Any edits and changes were done by Murray McAllister.

to be protected from legitimate users with limited authorization as well as from authorized users who have unwittingly executed malicious applications.[2]

The following is an example of the labels containing security-relevant information that are used on processes, Linux users, and files, on Linux operating systems that run SELinux. This information is called the SELinux *context*, and is viewed using the **`ls -Z`** command:

```
$ ls -Z file1
-rwxrw-r--  user1 group1 unconfined_u:object_r:user_home_t:s0      file1
```

In this example, SELinux provides a user (**`unconfined_u`**), a role (**`object_r`**), a type (**`user_home_t`**), and a level (**`s0`**). This information is used to make access control decisions. With DAC, access is controlled based only on Linux user and group IDs. It is important to remember that SELinux policy rules are checked *after* DAC rules. SELinux policy rules are not used if DAC rules deny access first.

Linux y los Usuarios SELinux

On Linux operating systems that run SELinux, there are Linux users as well as SELinux users. SELinux users are part of SELinux policy. Linux users are mapped to SELinux users. To avoid confusion, this guide uses "Linux user" and "SELinux user" to differentiate between the two.

2.1. Beneficios de usar SELinux

- Todos los procesos y archivos se etiquetan con un tipo. Un tipo define un dominio para los procesos y un tipo para los archivos. Los procesos se separan entre sí corriéndolos en sus propios dominios, y las reglas de políticas de SELinux define cómo interactúan los procesos con los archivos, así como la forma en que interactúan entre sí. El acceso sólo se permite si existe una regla de política de SELinux que específicamente lo permita.
- Control de acceso más fino. Yendo un paso más alla de los permisos tradicionales de UNIX® que se controlan a discreción del usuario y se basa en los IDs de usuario y de grupos de Linux, las decisiones de accesos de SELinux se basan en toda la información disponible, tales como un usuario SELinux, el rol, el tipo y, opcionalmente, un nivel.
- La política de SELinux se define administrativamente, obligando a todo el sistema, y no se pone a discreción del usuario.
- Reduced vulnerability to privilege escalation attacks. One example: since processes run in domains, and are therefore separated from each other, and because SELinux policy rules define how processes access files and other processes, if a process is compromised, the attacker only has access to the normal functions of that process, and to files the process has been configured to have access to. For example, if the Apache HTTP Server is compromised, an attacker can not use that process to read files in user home directories, unless a specific SELinux policy rule was added or configured to allow such access.
- Se linux se puede usar para asegurar la confidencialidad e integridad de los datos, así como proteger los procesos de entradas no confiables.

"Meeting Critical Security Objectives with Security-Enhanced Linux", by Peter Loscocco and Stephen Smalley. This paper was originally prepared for the National Security Agency and is, consequently, in the public domain. Refer to the *original paper* [http://www.nsa.gov/research/_files/selinux/papers/ottawa01/index.shtml] for details and the document as it was first released. Any edits and changes were done by Murray McAllister.

SELinux no es:

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

SELinux está diseñado para mejorar las soluciones de seguridad existentes, no reemplazarlas. Aún cuando corra SELinux, siga las buenas prácticas de seguridad, tales como mantener el software actualizado, usar contraseñas difíciles de adivinar, cortafuegos y demás.

2.2. Ejemplos

Los siguientes ejemplos demuestran cómo SELinux aumenta la seguridad:

- The default action is deny. If an SELinux policy rule does not exist to allow access, such as for a process opening a file, access is denied.
- SELinux can confine Linux users. A number of confined SELinux users exist in SELinux policy. Linux users can be mapped to confined SELinux users to take advantage of the security rules and mechanisms applied to them. For example, mapping a Linux user to the SELinux user_u user, results in a Linux user that is not able to run (unless configured otherwise) set user ID (setuid) applications, such as **sudo** and **su**, as well as preventing them from executing files and applications in their home directory - if configured, this prevents users from executing malicious files from their home directories.
- Process separation is used. Processes run in their own domains, preventing processes from accessing files used by other processes, as well as preventing processes from accessing other processes. For example, when running SELinux, unless otherwise configured, an attacker can not compromise a Samba server, and then use that Samba server as an attack vector to read and write to files used by other processes, such as databases used by MySQL®.
- SELinux helps limit the damage made by configuration mistakes. *Domain Name System (DNS)*[3] servers often replicate information between each other in what is known as a zone transfer. Attackers can use zone transfers to update DNS servers with false information. When running the *Berkeley Internet Name Domain (BIND)*[4] as a DNS server in Fedora, even if an administrator forgets to limit which servers can perform a zone transfer, the default SELinux policy prevents zone files [5] from being updated via zone transfers, by the BIND `named` daemon itself, and by other processes.
- Refer to the *Red Hat® Magazine*[6] article, *Risk report: Three years of Red Hat Enterprise Linux 4*[78], for exploits that were restricted due to the default SELinux targeted policy in Red Hat® Enterprise Linux® 4.
- Refer to the *LinuxWorld.com*[9] article, *A seatbelt for server software: SELinux blocks real-world exploits*[1011], for background information about SELinux, and information about various exploits that SELinux has prevented.
- Refer to James Morris's *SELinux mitigates remote root vulnerability in OpenPegasus*[12] blog post for information about an exploit in *OpenPegasus*[13] that was mitigated by SELinux as shipped with Red Hat Enterprise Linux 4 and 5.

El sitio web de *Tresys Technology*[14] tiene una sección de *Noticias de Migración a SELinux*[15] (en la parte derecha), que lista los ataques recientes que fueron mitigados o prevenidos por SELinux.

2.3. Arquitectura de SELinux

SELinux es un módulo de seguridad de Linux que se construye dentro del kernel de Linux. SELinux se maneja por reglas de políticas cargables. Cuando un acceso de seguridad relevante se lleva a cabo, tal como un proceso que trata de abrir un archivo, la operación es interceptada por SELinux en el kernel. Si una regla de política de SELinux permite la operación, continúa, sino, la operación se bloquea y el proceso recibe un error.

SELinux decisions, such as allowing or disallowing access, are cached. This cache is known as the Access Vector Cache (AVC). Caching decisions decreases how often SELinux policy rules need to be checked, which increases performance. Remember that SELinux policy rules have no effect if DAC rules deny access first.

2.4. SELinux en otros Sistemas Operativos

Vaya a la siguiente información sobre cómo correr SELinux en sistemas operativos:

- Hardened Gentoo: *http://www.gentoo.org/proj/en/hardened/selinux/selinux-handbook.xml*.
- Debian: *http://wiki.debian.org/SELinux*.
- Ubuntu: *https://wiki.ubuntu.com/SELinux* and *https://help.ubuntu.com/community/SELinux*.
- Linux para Empresas de Red Hat: *Guía de Despliegue del Linux para Empresas de Red Hat*[16] y la *Guía de SELinux para el Linux para Empresas de Red Hat*[17].
- Fedora: *http://fedoraproject.org/wiki/SELinux* and the *Fedora Core 5 SELinux FAQ*[18].

[14] http://www.tresys.com/
[15] http://www.tresys.com/innovation.php

Contextos de SELinux

Processes and files are labeled with an SELinux context that contains additional information, such as an SELinux user, role, type, and, optionally, a level. When running SELinux, all of this information is used to make access control decisions. In Fedora, SELinux provides a combination of Role-Based Access Control (RBAC), Type Enforcement® (TE), and, optionally, Multi-Level Security (MLS).

The following is an example showing SELinux context. SELinux contexts are used on processes, Linux users, and files, on Linux operating systems that run SELinux. Use the **`ls -Z`** command to view the SELinux context of files and directories:

```
$ ls -Z file1
-rwxrw-r--  user1 group1 unconfined_u:object_r:user_home_t:s0      file1
```

Los contextos de SELinux siguen la *SELinux nombre-de-usuario:role:type:level* sintaxis:

SELinux user

The SELinux user identity is an identity known to the policy that is authorized for a specific set of roles, and for a specific MLS range. Each Linux user is mapped to an SELinux user via SELinux policy. This allows Linux users to inherit the restrictions placed on SELinux users. The mapped SELinux user identity is used in the SELinux context for processes in that session, in order to define what roles and levels they can enter. Run the **`semanage login -l`** command as the Linux root user to view a list of mappings between SELinux and Linux user accounts:

```
# /usr/sbin/semanage login -l

Login Name                SELinux User              MLS/MCS Range

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

Output may differ slightly from system to system. The **`Login Name`** column lists Linux users, and the **`SELinux User`** column lists which SELinux user the Linux user is mapped to. For processes, the SELinux user limits which roles and levels are accessible. The last column, **`MLS/MCS Range`**, is the level used by Multi-Level Security (MLS) and Multi-Category Security (MCS). Levels are briefly discussed later.

role

Parte de SELinux es el modelo de seguridad de Control de Acceso Basado en Roles (RBAC). El rol es un atributo de RBAC. Los usuarios de SELinux son autorizados para ciertos roles y los roles son autorizados para ciertos dominios. Los roles sirven como un intermediario entre dominios y usarios SELinux. Los roles en los que se puede ingresar determinan los dominios a los que se ingresan - al final, esto controla los tipos de objetos que se pueden acceder. Esto ayuda a reducir la vulnerabilidad de ataques de escalada de privilegios.

type

The type is an attribute of Type Enforcement. The type defines a domain for processes, and a type for files. SELinux policy rules define how types can access each other, whether it be a domain accessing a type, or a domain accessing another domain. Access is only allowed if a specific SELinux policy rule exists that allows it.

level

El nivel es un atributo de MLS y la Seguridad Multi Categoría (MCS). Un rango MLS es un par de niveles, escrito como *bajonivel-altonivel* si los niveles sin distintos, o *bajonivel* si los niveles son idénticos (**s0-s0** es lo mismo que **s0**). Cada nivel es un par sensible a categorías, donde las categorías son opcionales. Si no hay categorías, el nivel se escribe como *sensibilidad:conjunto-de-categoría*. Si no hay categorías, se escribe como *sensibilidad*.

If the category set is a contiguous series, it can be abbreviated. For example, **c0.c3** is the same as **c0,c1,c2,c3**. The **/etc/selinux/targeted/setrans.conf** file maps levels (**s0:c0**) to human-readable form (ie. **CompanyConfidential**). Do not edit **setrans.conf** with a text editor: use **semanage** to make changes. Refer to the semanage(8) manual page for further information. In Fedora, targeted policy enforces MCS, and in MCS, there is just one sensitivity, **s0**. MCS in Fedora supports 1024 different categories: **c0** through to **c1023**. **s0-s0:c0.c1023** is sensitivity **s0** and authorized for all categories.

MLS enforces the *Bell-La Padula Mandatory Access Model*[1], and is used in Labeled Security Protection Profile (LSPP) environments. To use MLS restrictions, install the *selinux-policy-mls* package, and configure MLS to be the default SELinux policy. The MLS policy shipped with Fedora omits many program domains that were not part of the evaluated configuration, and therefore, MLS on a desktop workstation is unusable (no support for the X Window System); however, an MLS policy from the *upstream SELinux Reference Policy*[2] can be built that includes all program domains.

3.1. Transiciones de Dominios

Un proceso transiciona de un dominio a otro ejecutando una aplicación que tiene el tipo **entrypoint** en el nuevo dominio. Los permisos **entrypoint** se usan en las políticas de SELinux, y controlan qué aplicaciones pueden usarse para ingresar a un dominio. El siguiente ejemplo muestra una transición de dominio.

1. A user wants to change their password. To do this, they run the **passwd** application. The **/usr/bin/passwd** executable is labeled with the **passwd_exec_t** type:

```
$ ls -Z /usr/bin/passwd
-rwsr-xr-x  root root system_u:object_r:passwd_exec_t:s0 /usr/bin/passwd
```

   La aplicación **passwd** accede **/etc/shadow**, que está etiquetado con el tipo **shadow_t**:

```
$ ls -Z /etc/shadow
-r--------. root root system_u:object_r:shadow_t:s0    /etc/shadow
```

2. Una regla de política de SELinux dice que los procesos que se ejecutan en el dominio **passwd_t** no pueden leer y escribir archivos etiquetados con el tipo **shadow_t**. El tipo **shadow_t** sólo se aplica a archivos que necesitan un cambio de contraseñas. Esto incluye a **/etc/gshadow**, **/etc/shadow**, y sus archivos de respaldo.

3. Una regla de política de SELinux fija que el dominio **passwd_t** tiene permiso de **entrypoint** al tipo **passwd_exec_t**.

[1] http://en.wikipedia.org/wiki/Bell-LaPadula_model
[2] http://oss.tresys.com/projects/refpolicy

4. When a user runs the **/usr/bin/passwd** application, the user's shell process transitions to the **passwd_t** domain. With SELinux, since the default action is to deny, and a rule exists that allows (among other things) applications running in the **passwd_t** domain to access files labeled with the **shadow_t** type, the **passwd** application is allowed to access **/etc/shadow**, and update the user's password.

Este ejemplo no es exhaustivo, y se usa como un ejemplo básico para explicar la transición de dominio. Aunque hay una regla actual que permite a sujetos corriendo en el dominio **passwd_t** accedan objetos etiquetados con el tipo de archivo **shadow_t**, otras reglas de política de SELinux se deben cumplir para que el sujeto pueda transicionar a un nuevo dominio. En este ejemplo, la Obligación de Tipo asegura:

- el dominio **passwd_t** sólo se puede ingresar ejecutando una aplicación con la etiqueta del tipo **passwd_exec_t**; sólo pueden ejecutar desde bibliotecas compartidas autorizadas, tales como las del tipo **lib_t**; y no pueden ejecutar ninguna otra aplicación.
- sólo los dominios autorizados, tales como **passwd_t**, pueden escribir en archivos con la etiqueta del tipo **shadow_t**. Aún si otros procesos corren con privilegios de superusuario, esos procesos no podrán escribir archivos etiquetados con el tipo **shadow_t**, porque no están corriendo en el dominio **passwd_t**.
- sólo los dominios autorizados pueden transicionar al dominio **passwd_t**. Por ejemplo, el proceso `sendmail` corriendo en el dominio **sendmail_t** no tiene una razón legítima para ejecutar **passwd**; por lo tanto, no puede transicionar nunca al dominio **passwd_t**.
- los procesos que se ejecutan en el dominio **passwd_t** sólo pueden leer y escribir a tipos autorizados, tales como archivos etiquetados con los tipos **etc_t** o **shadow_t**. Esto impide a la aplicación **passwd** de ser modificada para leer o escribir en archivos arbitrarios.

3.2. Contextos de SELinux para los Procesos

Use el comando **ps -eZ** para ver los contextos de SELinux para los procesos. Por ejemplo:

1. Abra una terminal, como la de **Aplicaciones** → **Herramientas del Sistema** → **Terminal**.
2. Ejecute el comando **/usr/bin/passwd**. No ingrese una nueva contraseña.
3. Abra una nueva pestaña, u otra terminal, y ejecute el comando **ps -eZ | grep passwd**. La salida es similar a la siguiente:

```
unconfined_u:unconfined_r:passwd_t:s0-s0:c0.c1023 13212 pts/1 00:00:00 passwd
```

4. In the first tab/terminal, press **Ctrl+C** to cancel the **passwd** application.

In this example, when the **/usr/bin/passwd** application (labeled with the **passwd_exec_t** type) is executed, the user's shell process transitions to the **passwd_t** domain. Remember: the type defines a domain for processes, and a type for files.

Use el comando **ps -eZ** para ver los contextos SELinux de los procesos en ejecución. El siguiente es un ejemplo limitado de la salida, y puede cambiar en su sistema:

```
system_u:system_r:dhcpc_t:s0     1869 ?        00:00:00 dhclient
system_u:system_r:sshd_t:s0-s0:c0.c1023 1882 ? 00:00:00 sshd
system_u:system_r:gpm_t:s0       1964 ?        00:00:00 gpm
system_u:system_r:crond_t:s0-s0:c0.c1023 1973 ? 00:00:00 crond
system_u:system_r:kerneloops_t:s0 1983 ?       00:00:05 kerneloops
system_u:system_r:crond_t:s0-s0:c0.c1023 1991 ? 00:00:00 atd
```

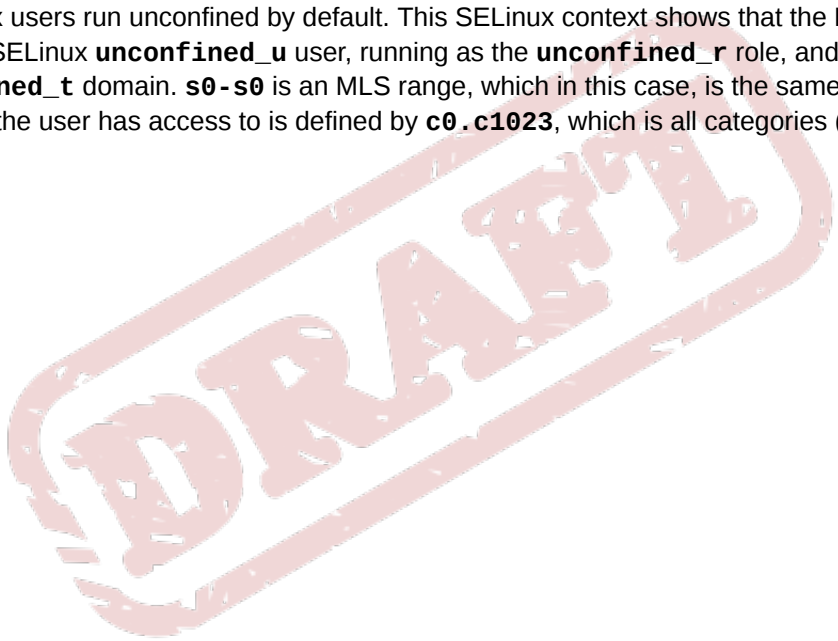
El rol **`system_r`** se usa para procesos de sistema, como los demonios. El tipo obligatorio los separa luego en dominios.

3.3. Contextos de SELinux para los Usuarios

Use el comando **`id -Z`** para ver el contexto SELinux asociado con su usuario Linux:

```
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

In Fedora, Linux users run unconfined by default. This SELinux context shows that the Linux user is mapped to the SELinux **`unconfined_u`** user, running as the **`unconfined_r`** role, and is running in the **`unconfined_t`** domain. **`s0-s0`** is an MLS range, which in this case, is the same as just **`s0`**. The categories the user has access to is defined by **`c0.c1023`**, which is all categories (**`c0`** through to **`c1023`**).



Política Destinado

Targeted policy is the default SELinux policy used in Fedora. When using targeted policy, processes that are targeted run in a confined domain, and processes that are not targeted run in an unconfined domain. For example, by default, logged in users run in the **unconfined_t** domain, and system processes started by init run in the **initrc_t** domain - both of these domains are unconfined.

Unconfined domains (as well as confined domains) are subject to executable and writeable memory checks. By default, subjects running in an unconfined domain can not allocate writeable memory and execute it. This reduces vulnerability to *buffer overflow attacks*[1]. These memory checks are disabled by setting Booleans, which allow the SELinux policy to be modified at runtime. Boolean configuration is discussed later.

4.1. Procesos Confinados

Almost every service that listens on a network is confined in Fedora. Also, most processes that run as the Linux root user and perform tasks for users, such as the **passwd** application, are confined. When a process is confined, it runs in its own domain, such as the `httpd` process running in the **httpd_t** domain. If a confined process is compromised by an attacker, depending on SELinux policy configuration, an attacker's access to resources and the possible damage they can do is limited.

The following example demonstrates how SELinux prevents the Apache HTTP Server (`httpd`) from reading files that are not correctly labeled, such as files intended for use by Samba. This is an example, and should not be used in production. It assumes that the *httpd*, *wget*, *setroubleshoot-server*, *dbus* and *audit* packages are installed, that the SELinux targeted policy is used, and that SELinux is running in enforcing mode:

1. Ejecute el comando **sestatus** para confirmar que SELinux está activado, se ejecuta en modo obligatorio y que la política destinada se está usando:

```
$ /usr/sbin/sestatus
SELinux status:                 enabled
SELinuxfs mount:                /selinux
Current mode:                   enforcing
Mode from config file:          enforcing
Policy version:                 24
Policy from config file:        targeted
```

   SELinux status: enabled is returned when SELinux is enabled. **Current mode: enforcing** is returned when SELinux is running in enforcing mode. **Policy from config file: targeted** is returned when the SELinux targeted policy is used.

2. Como usuario root de Linux, ejecute el comando **touch /var/www/html/prueba** para crear un archivo.

3. Ejecute el comando **ls -Z /var/www/html/prueba** para ver el contexto SELinux:

```
-rw-r--r--  root root unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/testfile
```

[1] http://en.wikipedia.org/wiki/Buffer_overflow

By default, Linux users run unconfined in Fedora, which is why the **testfile** file is labeled with the SELinux **unconfined_u** user. RBAC is used for processes, not files. Roles do not have a meaning for files - the **object_r** role is a generic role used for files (on persistent storage and network file systems). Under the **/proc/** directory, files related to processes may use the **system_r** role.[2] The **httpd_sys_content_t** type allows the `httpd` process to access this file.

4. Como usuario root de Linux, ejecute el comando **service httpd start** para iniciar el proceso `httpd`. La salida escomo sigue si `httpd` inicia con éxito:

```
# /sbin/service httpd start
Starting httpd:                                            [  OK  ]
```

5. Cambie al directorio donde su usuario Linux tenga acceso de escritura y ejecute el comando **wget http://localhost/prueba**. A menos que hubieran cambios en la configuración predeterminada, este comando tiene éxito:

```
--2009-11-06 17:43:01--  http://localhost/testfile
Resolving localhost... 127.0.0.1
Connecting to localhost|127.0.0.1|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 0 [text/plain]
Saving to: `testfile'

[ <=>                                ] 0     --.-K/s   in 0s

2009-11-06 17:43:01 (0.00 B/s) - `testfile' saved [0/0]
```

6. El comando **chcon** reetiqueta archivos; sin embargo, tales cambios de etiquetas no sobreviven cuando el sistema se reetiqueta. Para que los cambios sobrevivan un reetiquetado del sistema, use el comando **semanage**, que se discute más adelante. Como usuario root de Linux, corra el siguiente comando para cambiar el tipo a un tipo usado por Samba:

   chcon -t samba_share_t /var/www/html/testfile

   Ejecute el comando **ls -Z /var/www/html/prueba** para ver los cambios:

```
-rw-r--r--  root root unconfined_u:object_r:samba_share_t:s0 /var/www/html/testfile
```

7. Nota: los permisos de DAC actuales permiten al proceso `httpd` acceder al **prueba**. Cambie al directorio donde el usuario Linux tenga permiso de escritura y ejecute el comando **wget http://localhost/prueba**. A menos que hayan cambios en la configuración predeterminada, este comando fallará:

```
--2009-11-06 14:11:23--  http://localhost/testfile
Resolving localhost... 127.0.0.1
Connecting to localhost|127.0.0.1|:80... connected.
HTTP request sent, awaiting response... 403 Forbidden
2009-11-06 14:11:23 ERROR 403: Forbidden.
```

8. Como usuario root de Linux, corra el comando **`rm -i /var/www/html/prueba`** para borrar **`prueba`**.

9. Si no necesita que corra `httpd`, como usuario root de Linux corra el comando **`service httpd stop`** para detener a `httpd`:

```
# /sbin/service httpd stop
Stopping httpd:                                            [  OK  ]
```

Este ejemplo muestra la seguridad adicional agregada por SELinux. Aunque las reglas de DAC permitieron al proceso `httpd` acceder a **`prueba`** en el paso 7, dado que estaba etiquetado con un tipo al que el proceso `httpd` no tenía acceso, SELinux negó el acceso. Después del paso 7, un error similar al siguiente se guarda en **`/var/log/messages`**:

```
May 6 23:00:54 localhost setroubleshoot: SELinux is preventing httpd (httpd_t) "getattr"
to /var/www/html/testfile (samba_share_t). For complete SELinux messages.
run sealert -l c05911d3-e680-4e42-8e36-fe2ab9f8e654
```

Archivos log previos pueden usar el formato **`/var/log/messages.`*`YYYYMMDD`***. Cuando se ejecuta **syslog-ng**, los archivos log previos pueden usar el formato **`/var/log/messages.`*`X`***. Si los procesos `setroubleshootd` y `auditd` están ejecutándose, errores similares a los siguientes se registran en **`/var/log/audit/audit.log`**:

```
type=AVC msg=audit(1220706212.937:70): avc:  denied  { getattr } for  pid=1904 comm="httpd"
 path="/var/www/html/testfile" dev=sda5 ino=247576 scontext=unconfined_u:system_r:httpd_t:s0
 tcontext=unconfined_u:object_r:samba_share_t:s0  tclass=file

type=SYSCALL msg=audit(1220706212.937:70): arch=40000003 syscall=196 success=no exit=-13
 a0=b9e21da0 a1=bf9581dc a2=555ff4 a3=2008171 items=0 ppid=1902 pid=1904 auid=500 uid=48
 gid=48 euid=48 suid=48 fsuid=48 egid=48 sgid=48 fsgid=48 tty=(none) ses=1 comm="httpd" exe="/
usr/sbin/httpd" subj=unconfined_u:system_r:httpd_t:s0 key=(null)
```

También, un error similar al siguiente se registra en **`/var/log/httpd/error_log`**:

```
[Wed May 06 23:00:54 2009] [error] [client 127.0.0.1] (13)Permission denied: access to /
testfile denied
```

4.2. Procesos no Confinados

Los procesos no confinados corren en dominios no confinados, por ejemplo, los programas init (del arranque) corren en el dominio no confinado **`initrc_t`**, los procesos no confinados del kernel corren en el dominio **`kernel_t`** y los usuarios no confinados deLInux corren en el dominio **`unconfined_t`**. Para procesos no confinados, las reglas de la política de SELinux son aplicadas, pero hay reglas de la política que permiten que los procesos se ejecuten en dominios no confinados tengan casi todos los accesos. Los procesos que corren en dominios no confinados terminan usando exclusivamente las reglas DAC. Si un proceso no confinado es comprometido, SELinux no impide que un atacante gane acceso a los recursos del sistema y a los datos, pero, por supuesto, las reglas DAC todavía se usan. SELinux es una mejora de seguridad sobre las reglas DAC - no las reemplaza.

The following example demonstrates how the Apache HTTP Server (`httpd`) can access data intended for use by Samba, when running unconfined. Note: in Fedora, the `httpd` process runs in the confined **`httpd_t`** domain by default. This is an example, and should not be used in production. It assumes that the *httpd*, *wget*, *setroubleshoot-server*, *dbus* and *audit* packages are installed, that the SELinux targeted policy is used, and that SELinux is running in enforcing mode:

1. Ejecute el comando **`sestatus`** para confirmar que SELinux está activado, se ejecuta en modo obligatorio y que la política destinada se está usando:

   ```
   $ /usr/sbin/sestatus
   SELinux status:                 enabled
   SELinuxfs mount:                /selinux
   Current mode:                   enforcing
   Mode from config file:          enforcing
   Policy version:                 23
   Policy from config file:        targeted
   ```

   `SELinux status: enabled` is returned when SELinux is enabled. **`Current mode: enforcing`** is returned when SELinux is running in enforcing mode. **`Policy from config file: targeted`** is returned when the SELinux targeted policy is used.

2. Como usuario root de Linux, corra el comando **`touch /var/www/html/prueba2`** para crear un archivo.

3. Ejecute el comando **`ls -Z /var/www/html/prueba2`** para ver el contexto SELinux:

   ```
   -rw-r--r--  root root unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test2file
   ```

   By default, Linux users run unconfined in Fedora, which is why the **`test2file`** file is labeled with the SELinux **`unconfined_u`** user. RBAC is used for processes, not files. Roles do not have a meaning for files - the **`object_r`** role is a generic role used for files (on persistent storage and network file systems). Under the **`/proc/`** directory, files related to processes may use the **`system_r`** role.[3] The **`httpd_sys_content_t`** type allows the `httpd` process to access this file.

4. El comando **`chcon`** reetiqueta archivos; sin embargo, tales cambios de etiquetas no sobreviven cuando el sistema se reetiqueta. Para que los cambios sobrevivan un reetiquetado del sistema, use el comando **`semanage`**, que se discute más adelante. Como usuario root de Linux, corra el siguiente comando para cambiar el tipo a un tipo usado por Samba:

   `chcon -t samba_share_t /var/www/html/test2file`

   Ejecute el comando **`ls -Z /var/www/html/prueba2`** para ver los cambios:

   ```
   -rw-r--r--  root root unconfined_u:object_r:samba_share_t:s0 /var/www/html/test2file
   ```

5. Corra el comando **`service httpd status`** para confirmar que el proceso `httpd` no se está ejecutando:

   ```
   $ /sbin/service httpd status
   httpd is stopped
   ```

```
```

Si la salida difiere, ejecute **`service httpd stop`** como usuario root de Linux para detener el proceso `httpd`:

```
# /sbin/service httpd stop
Stopping httpd:                                            [  OK  ]
```

6. Para hacer que el proceso `httpd` corra no confinado, ejecute el siguiente comando como usuario root de Linux para cambiar el tipo de **`/usr/sbin/httpd`**, a un tipo que no transicione a un dominio confinado:

   `chcon -t unconfined_exec_t /usr/sbin/httpd`

7. Ejecute el comando **`ls -Z /usr/sbin/httpd`** para confirmar que **`/usr/sbin/httpd`** está etiquetado con el tipo **`unconfined_exec_t`**:

```
-rwxr-xr-x  root root system_u:object_r:unconfined_exec_t /usr/sbin/httpd
```

8. Como usuario root de Linux, ejecute el comando **`service httpd start`** para iniciar el proceso `httpd`. La salida escomo sigue si `httpd` inicia con éxito:

```
# /sbin/service httpd start
Starting httpd:                                            [  OK  ]
```

9. Ejecute el comando **`ps -eZ | grep httpd`** para ver si `httpd` está corriendo en el dominio **`unconfined_t`**:

```
$ ps -eZ | grep httpd
unconfined_u:system_r:unconfined_t 7721 ?      00:00:00 httpd
unconfined_u:system_r:unconfined_t 7723 ?      00:00:00 httpd
unconfined_u:system_r:unconfined_t 7724 ?      00:00:00 httpd
unconfined_u:system_r:unconfined_t 7725 ?      00:00:00 httpd
unconfined_u:system_r:unconfined_t 7726 ?      00:00:00 httpd
unconfined_u:system_r:unconfined_t 7727 ?      00:00:00 httpd
unconfined_u:system_r:unconfined_t 7728 ?      00:00:00 httpd
unconfined_u:system_r:unconfined_t 7729 ?      00:00:00 httpd
unconfined_u:system_r:unconfined_t 7730 ?      00:00:00 httpd
```

10. Cambie al directorio donde su usuario Linux tenga permiso de escritura y ejecute el comando **`wget http://localhost/prueba2`**. A menos que hayan cambios en la configuración predeterminada, este comando debería tener éxito:

```
--2009-05-07 01:41:10--  http://localhost/test2file
Resolving localhost... 127.0.0.1
Connecting to localhost|127.0.0.1|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 0 [text/plain]
Saving to: `test2file.1'

[ <=>                                 ]--.-K/s   in 0s
```

```
2009-05-07 01:41:10 (0.00 B/s) - `test2file.1' saved [0/0]
```

Aunque el proceso `httpd` no tiene acceso a archivos etiquetados con el tipo **`samba_share_t`**, `httpd` se ejecuta en el dominio **`unconfined_t`**, y termina usando las reglas DAC, como tal, el comando **`wget`** tiene éxito. Teniendo a `httpd` ejecutándose en el dominio **`httpd_t`**, el comando **`wget`** habría fallado.

11. El comando **`restorecon`** restaura el contexto SELinux predeterminado de los archivos. Como usuario root de Linux, ejecute el comando **`restorecon -v /usr/sbin/httpd`** para restaurar el contexto SELinux de **`/usr/sbin/httpd`**:

```
# /sbin/restorecon -v /usr/sbin/httpd
restorecon reset /usr/sbin/httpd context system_u:object_r:unconfined_notrans_exec_t:s0-
>system_u:object_r:httpd_exec_t:s0
```

Ejecute el comando **`ls -Z /usr/sbin/httpd`** para confirmar que **`/usr/sbin/httpd`** está etiquetado con el tipo **`httpd_exec_t`**:

```
$ ls -Z /usr/sbin/httpd
-rwxr-xr-x  root root system_u:object_r:httpd_exec_t   /usr/sbin/httpd
```

12. Como usuario root de Linux, corra el comando **`/sbin/service httpd restart`** para reiniciar `httpd`. Después de reiniciar, ejecute **`ps -eZ | grep httpd`** para confirmar que `httpd` se está ejecutando en el dominio confinado **`httpd_t`**:

```
# /sbin/service httpd restart
Stopping httpd:                                            [  OK  ]
Starting httpd:                                            [  OK  ]
# ps -eZ | grep httpd
unconfined_u:system_r:httpd_t    8880 ?        00:00:00 httpd
unconfined_u:system_r:httpd_t    8882 ?        00:00:00 httpd
unconfined_u:system_r:httpd_t    8883 ?        00:00:00 httpd
unconfined_u:system_r:httpd_t    8884 ?        00:00:00 httpd
unconfined_u:system_r:httpd_t    8885 ?        00:00:00 httpd
unconfined_u:system_r:httpd_t    8886 ?        00:00:00 httpd
unconfined_u:system_r:httpd_t    8887 ?        00:00:00 httpd
unconfined_u:system_r:httpd_t    8888 ?        00:00:00 httpd
unconfined_u:system_r:httpd_t    8889 ?        00:00:00 httpd
```

13. Como usuario root de Linux, corra el comando **`rm -i /var/www/html/prueba2`** para eliminar **`prueba2`**.

14. Si no necesita que corra `httpd`, como usuario root de Linux corra el comando **`service httpd stop`** para detener a `httpd`:

```
# /sbin/service httpd stop
Stopping httpd:                                            [  OK  ]
```

Los ejemplos en estas secciones muestran cómo proteger los datos desde un proceso confinado comprometido (protegido por SELinux), así como cuánto más accesible son los datos para un atacante si el proceso comprometido estaba no confinado (no protegido por SELinux).

4.3. Usuarios Confinados y no Confinados

Cada usuario Linux se mapea a un usuario SELinux vía la política de SELinux. Esto permite a los usuarios Linux heredar las restricciones sobre los usuarios SELinux. Este mapeo de usuarios Linux se ve ejecutando el comando **`semanage login -l`** como usuario root de Linux:

```
# /usr/sbin/semanage login -l

Login Name                SELinux User              MLS/MCS Range

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

In Fedora 13, Linux users are mapped to the SELinux **`__default__`** login by default (which is mapped to the SELinux **`unconfined_u`** user). The following defines the default-mapping:

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

The following example demonstrates adding a new Linux user, and that Linux user being mapped to the SELinux **`unconfined_u`** user. It assumes that the Linux root user is running unconfined, as it does by default in Fedora 13:

1. Como usuario root de Linux, ejecute el comando **`/usr/sbin/useradd usuarionuevo`** para crear un nuevo usuario Linux con nombre usuarionuevo.

2. As the Linux root user, run the **`passwd newuser`** command to assign a password to the Linux newuser user:

   ```
   # passwd newuser
   Changing password for user newuser.
   New UNIX password: Enter a password
   Retype new UNIX password: Enter the same password again
   passwd: all authentication tokens updated successfully.
   ```

3. Log out of your current session, and log in as the Linux newuser user. When you log in, pam_selinux maps the Linux user to an SELinux user (in this case, unconfined_u), and sets up the resulting SELinux context. The Linux user's shell is then launched with this context. Run the **`id -Z`** command to view the context of a Linux user:

   ```
   [newuser@localhost ~]$ id -Z
   unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
   ```

4. Log out of the Linux newuser's session, and log in with your account. If you do not want the Linux newuser user, run the **`/usr/sbin/userdel -r newuser`** command as the Linux root user to remove it, along with the Linux newuser's home directory.

Los usuarios Linux confinados y no confinados son sujeto a chequeso de memoria escribible y ejecutable, y también son restringidos por MCS (y MLS, si la política MLS está siendo usada). Si los usuarios Linux no confinados ejecutan una aplicación que la política de SELinux define que puede transicionar del dominio **unconfined_t** a su propio dominio confinado, los usuarios Linux no confinados están sujetos a restricciones de ese dominio confinado. El beneficio de seguridad de esto es que, aunque un usuario Linux corra no confinado, la aplicación permanece confinada, y por lo tanto, la explotación de una brecha en la aplicación está limitada por la política. Nota: esto no protege al sistema del usuario. En su defecto, el usuario y el sistema están siendo protegidos de posibles daños causados por una brecha en la aplicación.

The following confined SELinux users are available in Fedora 13:

Usuario	Dominio	Sistema de Ventanas X	su y sudo	Ejecute en el directorio de inicio y /tmp/	Red
guest_u	guest_t	no	no	optional	no
xguest_u	xguest_t	yes	no	optional	only **Firefox**
user_u	user_t	yes	no	optional	yes
staff_u	staff_t	yes	only **sudo**	optional	yes

Tabla 4.1. Capacidades del Usuario SELinux

- Los usuarios Linux en los dominios **guest_t**, **xguest_t** y **user_t** sólo pueden ejecutar aplicaciones con ID de usuario (setuid) si la política de SELinux lo permite (tal como **passwd**). No podrán ejecutar **su** y **/usr/bin/sudo**
- Los usuarios Linux en el dominio **guest_t** no tienen acceso a la red, y sólo pueden ingresar vía una terminal (incluyendo ssh; pueden ingresaro por ssh, pero no se pueden ssh conectar a otro sistema).
- El único acceso de red que tienen los usuarios en el dominoi **xguest_t** es con **Firefox** conectándose a páginas web.
- Los usuarios Linux en los dominios **xguest_t**, **user_t** y **staff_t** pueden ingresar vía el Sistema de Ventanas X o por una terminal.
- Por defecto, los usuarios Linux en el dominio **staff_t** no tienen permisos para ejecutar aplicaciones con **/usr/bin/sudo**. Estos permisos deben ser configurados por un administrador.

By default, Linux users in the **guest_t** and **xguest_t** domains can not execute applications in their home directories or **/tmp/**, preventing them from executing applications (which inherit users' permissions) in directories they have write access to. This helps prevent flawed or malicious applications from modifying files users' own.

By default, Linux users in the **user_t** and **staff_t** domains can execute applications in their home directories and **/tmp/**. Refer to *Sección 6.6, "Booleanos para que los Usuarios Ejecuten Aplicaciones"* for information about allowing and preventing users from executing applications in their home directories and **/tmp/**.

Trabajando con SELinux

The following sections give a brief overview of the main SELinux packages in Fedora; installing and updating packages; which log files are used; the main SELinux configuration file; enabling and disabling SELinux; SELinux modes; configuring Booleans; temporarily and persistently changing file and directory labels; overriding file system labels with the **mount** command; mounting NFS file systems; and how to preserve SELinux contexts when copying and archiving files and directories.

5.1. Paquetes de SELinux

In Fedora, the SELinux packages are installed by default, unless they are manually excluded during installation. By default, SELinux targeted policy is used, and SELinux runs in enforcing mode. The following is a brief description of the main SELinux packages:

policycoreutils: provee utilitarios, tales como **semanage**, **restorecon**, **audit2allow**, **semodule**, **load_policy** y **setsebool**, para la operación y administración de SELinux.

policycoreutils-gui: provee **system-config-selinux**, una herramienta gráfica para la administración de SELinux.

selinux-policy: provee una Política de Referencia de SELinux. La Política de Referencia de SELinux en una política de SELinux completa, y se usa como base para otras políticas, tales como la política destinada de SELinux. Vaya a la página *Política de Referencia de SELinux*[1] de Tresys Technology para más información. El paquete *selinux-policy-devel* provee herramientas de desarrollo, tales como **/usr/share/selinux/devel/policygentool** y **/usr/share/selinux/devel/policyhelp**, así como archivos de política ejemplos. Este paquete fue mezclado con el paquete *selinux-policy*.

selinux-policy-policy: provee las políticas de SELinux. Para la política destinada, instale *selinux-policy-targeted*. Para MLS, instale *selinux-policy-mls*. En Fedora 8, la política estricta fue mezclada con la política destinada, permitiendo a los usuarios confinados y no confinados coexistir en el mismo sistema.

setroubleshoot-server: traduce mensajes de negaciones, producidos cuando el acceso es denegado por SELinux, en descripciones detalladas que se ven con **sealert** (que se provee en este paquete).

setools, *setools-gui*, and *setools-console*: these packages provide the *Tresys Technology SETools distribution*[2], a number of tools and libraries for analyzing and querying policy, audit log monitoring and reporting, and file context management[3]. The *setools* package is a meta-package for SETools. The *setools-gui* package provides the **apol**, **seaudit**, and **sediffx** tools. The *setools-console* package provides the **seaudit-report**, **sechecker**, **sediff**, **seinfo**, **sesearch**, **findcon**, **replcon**, and **indexcon** command line tools. Refer to the *Tresys Technology SETools*[4] page for information about these tools.

libselinux-utils: provee las herramientas **avcstat**, **getenforce**, **getsebool**, **matchpathcon**, **selinuxconlist**, **selinuxdefcon**, **selinuxenabled**, **setenforce**, **togglesebool**.

mcstrans: traduce niveles, tales como **s0-s0:c0.c1023**, a una forma legible como **SystemLow-SystemHigh**. Este paquete no se instala por defecto.

[1] http://oss.tresys.com/projects/refpolicy
[2] http://oss.tresys.com/projects/setools
Brindle, Joshua. "Re: blurb for fedora setools packages" Email to Murray McAllister. 1 November 2008. Any edits or changes in this version were done by Murray McAllister.
[4] http://oss.tresys.com/projects/setools

To install packages in Fedora, as the Linux root user, run the **`yum install package-name`** command. For example, to install the *mcstrans* package, run the **`yum install mcstrans`** command. To upgrade all installed packages in Fedora, run the **`yum update`** command.

Vaya a *Administración de Software con yum*[56] para más información sobre el uso de **`yum`** para administrar paquetes.

Nota

En versiones anteriores de Fedora, el paquete *selinux-policy-devel* es necesario cuando se crea un módulo de política local con **`audit2allow -M`**.

5.2. Qué Archivo Log se usa

In Fedora 13, the *dbus*, *setroubleshoot-server* and *audit* packages are installed if packages are not removed from the default package selection.

Los mensajes de negación de SELinux, tales como el siguiente, se escriben por defecto en **`/var/log/audit/audit.log`**:

```
type=AVC msg=audit(1223024155.684:49): avc:  denied  { getattr } for  pid=2000 comm="httpd"
 path="/var/www/html/file1" dev=dm-0 ino=399185 scontext=unconfined_u:system_r:httpd_t:s0
 tcontext=system_u:object_r:samba_share_t:s0 tclass=file
```

Also, if `setroubleshootd` is running, denial messages from **`/var/log/audit/audit.log`** are translated to an easier-to-read form and sent to **`/var/log/messages`**:

```
May  7 18:55:56 localhost setroubleshoot: SELinux is preventing httpd (httpd_t) "getattr"
 to /var/www/html/file1 (samba_share_t). For complete SELinux messages. run sealert -l
 de7e30d6-5488-466d-a606-92c9f40d316d
```

In Fedora 13, `setroubleshootd` no longer constantly runs as a service, however it is still used to analyze the AVC messages. Two new programs act as a method to start setroubleshoot when needed: `sedispatch` and `seapplet`. `sedispatch` runs as part of the audit subsystem, and via `dbus`, sends a message when an AVC denial occurs, which will go straight to `setroubleshootd` if it is already running, or it will start `setroubleshootd` if it is not running. `seapplet` is a tool which runs in the system's toolbar, waiting for dbus messages in `setroubleshootd`, and will launch the notification bubble, allowing the user to review the denial.

Los mensajes de negación se envían a una ubicación distinta, dependiendo de cuáles demonios se están ejecutando:

Daemon	Log Location
auditd on	**`/var/log/audit/audit.log`**
auditd off; rsyslogd on	**`/var/log/messages`**
rsyslogd and auditd on	**`/var/log/audit/audit.log`**. Easier-to-read denial messages also sent to **`/var/log/messages`**

[5] http://docs.fedoraproject.org/yum/en/
Administración de Software con yum, escrito por Stuart Ellis, editado por Paul W. Frields, Rodrigo Menezes y Hugo Cisneiros.

Iniciando Demonios Automáticamente

Para configurar los demonios `auditd`, `rsyslogd`, y `setroubleshootd` para que inicien automáticamente al arrancar, corra los siguientes comandos como usuario root de Linux:

```
/sbin/chkconfig --levels 2345 auditd on
```

```
/sbin/chkconfig --levels 2345 rsyslog on
```

Use el comando **`service` *`nombre-de-servicio`* `status`** para chequear si estos servicios se están ejecutando, por ejemplo:

```
$ /sbin/service auditd status
auditd (pid  1318) is running...
```

Si los servicos de arriba no se están ejecutando (***`nombre-de-servicio`* `está detenido`**), use el comando **`service` *`nombre-de-servicio`* `start`** como usuario root de Linux para iniciarlos. Por ejemplo:

```
# /sbin/service auditd start
Starting auditd:                                   [  OK  ]
```

5.3. Archivo de Configuración Principal

El archivo **`/etc/selinux/config`** es el archivo de configuración principal de SELinux. Controla el modo de SELinux y la política de SELinux a usar:

```
# 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
```

`SELINUX=enforcing`

La opción **`SELINUX`** pone el modo en el que corre SELinux. SELinux tiene tres modos: obediente, permisivo y deshabilitado. Cuando se usa modo obediente, la política de SELinux es aplicada y SELinux niega el acceso basándose en las reglas de políticas de SELinux. Los mensajes de negación se guardan. Cuando se usa modo permisivo, la política de SELinux no es obediente. Los mensajes son guardados. SELinux no niega el acceso, pero se guardan las negaciones de acciones que hubieran sido negadas si SELinux estaba en modo obediente. Cuando se usa el modo deshabilitado, SELinux está deshabilitado (el módulo de SELinux no se registra con el kernel de Linux), y sólo se usan las reglas DAC.

SELINUXTYPE=targeted

La opción **SELINUXTYPE** pone la política SELinux a usar. La política Destinada es la predeterminada. Sólo cambie esta opción si quiere usar la política MLS. Para usar la política MLS, instale el paquete *selinux-policy-mls*; configure **SELINUXTYPE=mls** en **/etc/selinux/config**; y reinicie su sistema.

Importante

Cuando los sistemas corren con SELinux en modo permisivo o deshabilitado, los usuarios tiene permiso para etiquetar los archivos incorrectamente. También, los archivos creados con SELinux deshabilitado no son etiquetados. Esto causa problemas cuando se cambia a modo obediente. Para prevenir el etiquetado incorrecto o la falta de etiquetado, los sistemas de archivos son automáticamente reetiquetados cuando se cambie desde el modo deshabilitado al modo permisivo u obediente.

5.4. Habilitando y Deshabilitando SELinux

Use los comandos **/usr/sbin/getenforce** o **/usr/sbin/sestatus** para chequear el estado de SELinux. El comando **getenforce** devuelve **Obediente**, **Permisivo**, o **Deshabilitado**. El comando **getenforce** devuelve **Obediente** cuando SELinux está habilitado (las reglas de la política de SELinux son aplicadas):

```
$ /usr/sbin/getenforce
Enforcing
```

El comando **getenforce** devuelve **Permissive** cuando SELinux está activado, pero las reglas de políticas de SELinux no están en obligatorio, y sólo se usan las reglas DAC. El comando **getenforce** devuelve **Disabled** si SELinux está deshabilitado.

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

```
$ /usr/sbin/sestatus
SELinux status:                 enabled
SELinuxfs mount:                /selinux
Current mode:                   enforcing
Mode from config file:          enforcing
Policy version:                 23
Policy from config file:        targeted
```

SELinux status: enabled is returned when SELinux is enabled. **Current mode: enforcing** is returned when SELinux is running in enforcing mode. **Policy from config file: targeted** is returned when the SELinux targeted policy is used.

5.4.1. Habilitando SELinux

En sistemas con SELinux deshabilitado, la opción **SELINUX=disabled** se configura en **/etc/selinux/config**:

```
# 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
```

También, el comando **`getenforce`** devuelve **`Disabled`**:

```
$ /usr/sbin/getenforce
Disabled
```

Para habilitar SELinux:

1. Use los comandos **`rpm -qa | grep selinux`**, **`rpm -q policycoreutils`** y **`rpm -qa | grep setroubleshoot`** para confirmar que los paquetes de SELinux están instalados. esta guía asume que los siguientes paquetes están instalados: *selinux-policy-targeted*, *selinux-policy*, *libselinux*, *libselinux-python*, *libselinux-utils*, *policycoreutils*, *setroubleshoot*, *setroubleshoot-server*, *setroubleshoot-plugins*. Si estos paquetes no están instalados, como usuario root de Linux, debe instalarlos con el comando **`yum install`** ***`nombre-de-paquete`***. Los siguientes paquetes son opcionales: *policycoreutils-gui*, *setroubleshoot*, *selinux-policy-devel* y *mcstrans*.

2. Antes de activar SELinux, cada archivo en el sistema de archivo debe ser etiquetado con un contexto de SELinux. Antes que esto ocurra, los dominios confinados pueden tener el acceso denegado, impidiendo de que su sistema se inicie correctamente. Para prevenir esto, configure **`SELINUX=permissive`** en **`/etc/selinux/config`**:

   ```
   # 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. Como usuario root de Linux, corra el comando **`reboot`** para reiniciar el sistema. Durante el siguiente arranque, los sistemas de archivo son etiquetados. El proceso etiqueta todos los archivos con un contexto de SELinux:

   ```
   *** Warning -- SELinux targeted policy relabel is required.
   *** Relabeling could take a very long time, depending on file
   *** system size and speed of hard drives.
   ****
   ```

   Cada carácter **`*`** en la línea de abajo representa 1000 archivos que han sido etiquetados. En el ejemplo de arriba, cuatro caracteres **`*`** representan 4000 archivos etiquetados. El tiempo que toma reetiquetar todos los archivos depende del número de archivos del sistema, y la velocidad de los discos rígidos. En sistemas modernos, este proceso puede tomar 10 minutos.

4. In permissive mode, SELinux policy is not enforced, but denials are still logged for actions that would have been denied if running in enforcing mode. Before changing to enforcing mode, as the Linux root user, run the **`grep "SELinux is preventing" /var/log/messages`** command as the Linux root user to confirm that SELinux did not deny actions during the last boot. If SELinux did not deny actions during the last boot, this command does not return any output. Refer to *Capítulo 7, Solución a Problemas* for troubleshooting information if SELinux denied access during boot.

5. Si no hay mensajes de negación en **`/var/log/messages`**, configure **`SELINUX=enforcing`** en **`/etc/selinux/config`**:

```
# 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
```

6. Reinicie su sistema. Después de reiniciar, confirme que **`getenforce`** devuelve **`Enforcing`**:

```
$ /usr/sbin/getenforce
Enforcing
```

7. Como usuario root de Linux, corra el comando **`/usr/sbin/semanage login -l`** para ver el mapeo entre usuarios de SELinux y de Linux. La salida debe ser como la siguiente:

```
Login Name                SELinux User              MLS/MCS Range

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

Si éste no es el caso, ejecute los siguientes comandos como usuario root de Linux para corregir los mapeos de usuario. Es seguro ignorar los mensajes **`El usuario SELinux`** ***`nombre-de-usuario`*** **`ya está definido`** si es que aparecen, donde *`nombre-de-usuario`* puede ser **`unconfined_u`**, **`guest_u`**, o **`xguest_u`**:

1.
```
/usr/sbin/semanage user -a -S targeted -P user -R "unconfined_r system_r" -r s0-
s0:c0.c1023 unconfined_u
```

2.
```
/usr/sbin/semanage login -m -S targeted -s "unconfined_u" -r s0-s0:c0.c1023 __default__
```

3.
```
/usr/sbin/semanage login -m -S targeted -s "unconfined_u" -r s0-s0:c0.c1023 root
```

4.
```
/usr/sbin/semanage user -a -S targeted -P user -R guest_r guest_u
```

5.
```
/usr/sbin/semanage user -a -S targeted  -P user -R xguest_r xguest_u
```

Importante

Cuando los sistemas corren con SELinux en modo permisivo o deshabilitado, los usuarios tiene permiso para etiquetar los archivos incorrectamente. También, los archivos creados con SELinux deshabilitado no son etiquetados. Esto causa problemas cuando se cambia a modo obediente. Para prevenir el etiquetado incorrecto o la falta de etiquetado, los sistemas de archivos son automáticamente reetiquetados cuando se cambie desde el modo deshabilitado al modo permisivo u obediente.

5.4.2. Deshabilitando SELinux

Para deshabilitar SELinux, configure **`SELINUX=disabled`** en **`/etc/selinux/config`**:

```
# 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
```

Reinicie su sistema. Después de reiniciar, confirme que **`getenforce`** devuelve **`Disabled`**:

```
$ /usr/sbin/getenforce
Disabled
```

5.5. Modos de SELinux

SELinux tiene tres modos:

- Obligatorio: la política de SELinux es obligatoria. SELinux niega el acceso basado en las reglas de políticas de SELinux.
- Permisivo: la política de SELinux no es obligatoria. SELinux no niega el acceso, pero se guardan las negaciones para acciones que hubieran sido negadas si el modo obligatorio estaba activado.
- Deshabilitado: SELinux está deshabilitado. Sólo se usan las reglas DAC.

Use el comando **`/usr/sbin/setenforce`** para cambiar entre los modos obediente y permisivo. Los cambios hechos con **`/usr/sbin/setenforce`** no sobreviven a una reiniciada. Para cambiar a modo obediente, como usuario root de Linux, ejecute el comando **`/usr/sbin/setenforce 1`**.

Para cambiar a modo permisivo, ejecute el comando **`/usr/sbin/setenforce 0`**. Use el comando **`/usr/sbin/getenforce`** para ver el modo de SELinux actual.

Persistent mode changes are covered in *Sección 5.4, "Habilitando y Deshabilitando SELinux"*.

5.6. Booleanos

Los booleanos permiten cambiar partes de la política de SELinux en tiempo de ejecución, sin ningún conocimiento sobre la escritura de políticas de SELinux. Esto permite cambios, como permitir el acceso de servicios a sistemas de archivo NFS, sin recargar o recompilar la política de SELinux.

5.6.1. Listando los Booleanos

Para una lista de los Booleanos, una explicación de lo que son y de si están activos o inactivos, ejecute el comando **`semanage boolean -l`** como usuario root de Linux. El siguiente ejemplo no lista todos los Booleanos:

```
# /usr/sbin/semanage boolean -l
SELinux boolean                          Description

ftp_home_dir                   -> off   Allow ftp to read and write files in the user home
 directories
xen_use_nfs                    -> off   Allow xen to manage nfs files
xguest_connect_network         -> on    Allow xguest to configure Network Manager
```

La columna **`SELinux boolean`** lista los nombres de Booleanos. La columna **`Description`** lista si el booleano está activo (on) o inactivo (off) y lo que hacen.

En el siguiente ejemplo, el Booleano **`ftp_home_dir`** está apagado, impidiendo al demonio FTP (`vsftpd`) la lectura y escritura de archivos en los directorios de inicio de los usuarios:

```
ftp_home_dir                   -> off   Allow ftp to read and write files in the user home
 directories
```

El comando **`getsebool -a`** lista los Booleanos, ya sea que estén activos o inactivos, pero no da una descripción de cada uno. El siguiente ejemplo no lista todos los booleanos:

```
$ /usr/sbin/getsebool -a
allow_console_login --> off
allow_cvs_read_shadow --> off
allow_daemons_dump_core --> on
```

Ejecute el comando **`getsebool`** ***`nombre-de-booleano`*** para listar solamente el estado del booleano *`nombre-de-booleano`*:

```
$ /usr/sbin/getsebool allow_console_login
allow_console_login --> off
```

Una lista separada por espacio para listar los Booleanos múltiples:

```
$ getsebool allow_console_login allow_cvs_read_shadow allow_daemons_dump_core
allow_console_login --> off
```

```
allow_cvs_read_shadow --> off
allow_daemons_dump_core --> on
```

5.6.2. Configurando los Booleanos

El comando **setsebool *nombre-de-booleano x*** activa o desactiva Booleanos, donde *nombre-de-booleano* es un nombre de Booleano, y *x* es **on** para activar, u **off** para desactivar.

El siguiente ejemplo muestra la configuración de Booleano **httpd_can_network_connect_db**:

1. Por defecto, el booleano **httpd_can_network_connect_db** está apagado, impidiendo a los scripts y módulos del Servidor HTTP Apache conectarse a servidores de bases de datos:

   ```
   $ /usr/sbin/getsebool httpd_can_network_connect_db
   httpd_can_network_connect_db --> off
   ```

2. Para permitir temporalmente a los scripts y módulos del Servidor HTTP Apache conectarse a servidores de bases de datos, ejecute el comando **setsebool httpd_can_network_connect_db on** como usuario root de Linux.

3. Use el comando **getsebool httpd_can_network_connect_db** para verificar que el Booleano está activado:

   ```
   $ /usr/sbin/getsebool httpd_can_network_connect_db
   httpd_can_network_connect_db --> on
   ```

   Esto permite a los scripts y módulos del Servidor HTTP Apache conectarse a servidores de bases de datos.

4. Este cambio no es persistente entre reinicios. Para hacer los cambios persistentes, corra el comando **setsebool -P *boolean-name* on** como usuario root de Linux:

   ```
   # /usr/sbin/setsebool -P httpd_can_network_connect_db on
   ```

5. Para revertir temporalmente el comportamiento por defecto, como usuario root de Linux, corra el comando **setsebool httpd_can_network_connect_db off**. Para que los cambios sean persistentes entre reiniciadas, ejecute el comando **setsebool -P httpd_can_network_connect_db off**.

5.6.3. Booleanos para NFS y CIFS

Por defecto, los montajes NFS en el lado del cliente se etiquetan con el contexto predeterminado definido por la política para sistemas de archivos NFS. En políticas comúnes, este contexto predeterminado usa el tipo **nfs_t**. También, por defecto, los compartidos de Samba en el lado del cliente se etiquetan con el contexto predeterminado definido por la política. En políticas comúnes, este contexto predeterminado usa el tipo **cifs_t**.

Dependiendo en la configuración de la política, los servicios pueden tener bloqueado la lectura a archivos con la etiqueta de los tipos **nfs_t** o **cifs_t**. Esto puede prevenir que los sistemas de archivo etiquetados con estas etiquetas se monten y sean leídos o exportados por otros servicios.

Hay Booleanos que se pueden poner en 1 o 0 para controlar qué servicios pueden acceder los tipos **nfs_t** y **cifs_t**.

Los comandos **setsebool** y **semanage** se deben ejecutar como usuario root de Linux. El comando **setsebool -P** hace persistentes a los cambios. No use la opción **-P** si no quiere que los cambios persistan entre reiniciadas:

Servidor HTTP Apache

Para permitir el acceso a sistemas de archivo NFS (archivos etiquetados con el tipo **nfs_t**):

```
/usr/sbin/setsebool -P httpd_use_nfs on
```

Para permitir el acceso a sistemas de archivos SAMBA (archivos etiquetados con el tipo **cifs_t**):

```
/usr/sbin/setsebool -P httpd_use_cifs on
```

Samba

Para exportar sistemas de archivo NFS:

```
/usr/sbin/setsebool -P samba_share_nfs on
```

FTP (vsftpd)

Para permitir el acceso a sistemas de archivo NFS:

```
/usr/sbin/setsebool -P allow_ftpd_use_nfs on
```

Para permitir el acceso a sistemas de archivo Samba:

```
/usr/sbin/setsebool -P allow_ftpd_use_cifs on
```

Otros Servicios

Para una lista de los Booleanos relacionados con NFS para otros servicios:

```
/usr/sbin/semanage boolean -l | grep nfs
```

Para una lista de los Booleanos relacionados con SAMBA para otros servicios:

```
/usr/sbin/semanage boolean -l | grep cifs
```

Nota

These Booleans exist in SELinux policy as shipped with Fedora 13. They may not exist in policy shipped with other versions of Fedora or other operating systems.

Refer to the SELinux Managing Confined Services Guide:*http://docs.fedoraproject.org/selinux-managing-confined-services-guide* for more information relating to SELinux Booleans.

5.7. Contextos de SELinux - Etiquetado de Archivos

On systems running SELinux, all processes and files are labeled in a way that represents security-relevant information. This information is called the SELinux context. For files, this is viewed using the **ls -Z** command:

```
$ ls -Z file1
-rw-rw-r--  user1 group1 unconfined_u:object_r:user_home_t:s0 file1
```

En este ejemplo, SELinux provee un usuario (**unconfined_u**), un rol (**object_r**), un tipo (**user_home_t**) y un nivel (**s0**). Esta información se usa para tomar decisiones sobre el control de acceso. En sistemas DAC, el control de acceso se basa en los IDs de usuario y grupo de Linux. Las reglas de la política de SELinux se chequean después de las reglas DAC. Las reglas de la política de SELinux no se usan si las reglas DAC niegan el acceso antes.

Hay muchos comandos para la administración del contexto de archivos de SELinux, como por ejemplo **chcon**, **semanage fcontext**, y **restorecon**.

5.7.1. Cambios Temporales: chcon

The **chcon** command changes the SELinux context for files. However, changes made with the **chcon** command do not survive a file system relabel, or the execution of the **/sbin/restorecon** command. SELinux policy controls whether users are able to modify the SELinux context for any given file. When using **chcon**, users provide all or part of the SELinux context to change. An incorrect file type is a common cause of SELinux denying access.

Referencia Rápida

- Ejecute el comando **chcon -t *tipo nombre-de-archivo*** para cambiar el tipo de archivo, donde *tipo* es el tipo, por ejemplo **httpd_sys_content_t**, y *nombre-de-archivo* es un nombre de archivo o de directorio.
- Ejecute el comando **chcon -R -t *tipo nombre-de-directorio*** para cambiar el tipo de un directorio y su contenido, donde *tipo* es el tipo, por ejemplo **httpd_sys_content_t**, y *nombre-de-directorio* es un nombre de directorio.

Changing a File's or Directory's Type

El siguiente ejemplo muestra el cambio de tipo solamente en el contexto de SELinux:

1. Ejecute el comando **cd** sin argumentos para cambiar a su directorio de inicio.
2. Ejecute el comando **touch archivo1** para crear un archivo nuevo. Use el comando **ls -Z archivo1** para ver el contexto de SELinux del **archivo1**:

   ```
   $ ls -Z file1
   -rw-rw-r--  user1 group1 unconfined_u:object_r:user_home_t:s0 file1
   ```

   In this example, the SELinux context for **file1** includes the SELinux **unconfined_u** user, **object_r** role, **user_home_t** type, and the **s0** level. For a description of each part of the SELinux context, refer to *Capítulo 3, Contextos de SELinux*.
3. EJecute el comando **chcon -t samba_share_t archivo1** para cambiar el tipo a **samba_share_t**. La opción **-t** sólo cambia el tipo. Vea el cambio con **ls -Z archivo1**:

   ```
   $ ls -Z file1
   -rw-rw-r--  user1 group1 unconfined_u:object_r:samba_share_t:s0 file1
   ```

4. Use el comando **`/sbin/restorecon -v archivo1`** para restaurar el contexto de SELinux del **`archivo1`**. Use la opción **`-v`** para ver qué cambia:

```
$ /sbin/restorecon -v file1
restorecon reset file1 context unconfined_u:object_r:samba_share_t:s0-
>system_u:object_r:user_home_t:s0
```

   En este ejemplo, el tipo previo **`samba_share_t`**, se restaura al tipo correcto **`user_home_t`**. Cuando se usa la política destinada (la política SELinux predeterminada en Fedora 11), el comando **`/sbin/restorecon`** lee los archivos en el directorio **`/etc/selinux/targeted/contexts/files/`** para ver qué contexto de SELinux deben tener los archivos.

El ejemplo en esta sección funciona igual para directorios, por ejemplo, si **`archivo1`** fuera un directorio.

Cambio de un Directorio y sus Tipos de Contenidos

The following example demonstrates creating a new directory, and changing the directory's file type (along with its contents) to a type used by the Apache HTTP Server. The configuration in this example is used if you want Apache HTTP Server to use a different document root (instead of **`/var/www/html/`**):

1. Como usuario root de Linux, ejecute el comando **`mkdir /web`** para crear un directorio nuevo, y luego el comando **`touch /web/archivo{1,2,3}`** para crear 3 archivos vacíos (**`archivo1`**, **`archivo2`** y **`archivo3`**). El directorio **`/web/`** y los archivos en él son etiquetados con el tipo **`default_t`**:

```
# ls -dZ /web
drwxr-xr-x  root root unconfined_u:object_r:default_t:s0 /web
# ls -lZ /web
-rw-r--r--  root root unconfined_u:object_r:default_t:s0 file1
-rw-r--r--  root root unconfined_u:object_r:default_t:s0 file2
-rw-r--r--  root root unconfined_u:object_r:default_t:s0 file3
```

2. Como usuario root de Linux, corra el comando **`chcon -R -t httpd_sys_content_t /web/`** para cambiar el tipo del directorio **`/web/`** (y su contenido) a **`httpd_sys_content_t`**:

```
# chcon -R -t httpd_sys_content_t /web/
# ls -dZ /web/
drwxr-xr-x  root root unconfined_u:object_r:httpd_sys_content_t:s0 /web/
# ls -lZ /web/
-rw-r--r--  root root unconfined_u:object_r:httpd_sys_content_t:s0 file1
-rw-r--r--  root root unconfined_u:object_r:httpd_sys_content_t:s0 file2
-rw-r--r--  root root unconfined_u:object_r:httpd_sys_content_t:s0 file3
```

3. Como usuario root de Linux, corra el comando **`/sbin/restorecon -R -v /web/`** para restaurar los contextos de SELinux predeterminados:

```
# /sbin/restorecon -R -v /web/
```

```
restorecon reset /web context unconfined_u:object_r:httpd_sys_content_t:s0-
>system_u:object_r:default_t:s0
restorecon reset /web/file2 context unconfined_u:object_r:httpd_sys_content_t:s0-
>system_u:object_r:default_t:s0
restorecon reset /web/file3 context unconfined_u:object_r:httpd_sys_content_t:s0-
>system_u:object_r:default_t:s0
restorecon reset /web/file1 context unconfined_u:object_r:httpd_sys_content_t:s0-
>system_u:object_r:default_t:s0
```

Refer to the chcon(1) manual page for further information about **chcon**.

Nota

La Obligación de Tipo es el control de permisos principal usado en la política destinada de SELinux. Para la mayor parte, los usuarios y roles de SELinux se pueden ignorar.

5.7.2. Cambios Persistentes: semanage fcontext

El comando **/usr/sbin/semanage fcontext** cambia el contexto SELinux de los archivos. Cuando se usa la política destinada, los cambios hechos con este comando se agregan al archivo **/etc/selinux/targeted/contexts/files/file_contexts** si los cambios so para archivos que están en **file_contexts**, se agregan a **file_contexts.local** para archivos nuevos y directorios, como sería al crear un directorio **/web/** nuevo. **setfiles**, que se usa cuando el sistema de archivo es reetiquetado, y **/sbin/restorecon**, que restaura los contextos de SELinux predeterminados, leen estos archivos, Lo que significa que los cambios hechos por **/usr/sbin/semanage fcontext** son persistentes, aún si el sistema de archivo es reetiquetado. La política de SELinux controla si los usuarios pueden modificar el contexto de SELinux para cualquier archivo dado.

Referencia Rápida

Para hacer que los cambios de contexto de SELinux sobrevivan un reetiquetado del sistema de archivo:

1. Ejecute el comando **/usr/sbin/semanage fcontext -a *opciones nombre-de-archivo|nombre-de-directorio***, recuerde usar la dirección completa del archivo o del directorio.

2. Ejecute el comando **/sbin/restorecon -v *nombre-de-archivo|nombre-de-directorio*** para aplicar los cambios de contexto.

Changing a File's Type

The following example demonstrates changing a file's type, and no other attributes of the SELinux context:

1. Como usuario root de Linux, ejecute el comando **touch /etc/archivo1** para crear un archivo nuevo. Por defecto, los archivos recién creados en el directorio **/etc/** se etiquetan con el tipo **etc_t**:

```
# ls -Z /etc/file1
-rw-r--r--  root root unconfined_u:object_r:etc_t:s0       /etc/file1
```

2. Como usuario root de Linux, ejecute el comando **`/usr/sbin/semanage fcontext -a -t samba_share_t /etc/archivo1`** para cambiar el tipo del **`archivo1`** a **`samba_share_t`**. La opción **`-a`** agrega un registro nuevo, y la opción **`-t`** define un tipo (**`samba_share_t`**). Nota: al ejecutar este comando no se cambia directamente el tipo - el **`archivo1`** todavía es del tipo **`etc_t`**:

```
# /usr/sbin/semanage fcontext -a -t samba_share_t /etc/file1
# ls -Z /etc/file1
-rw-r--r--  root root unconfined_u:object_r:etc_t:s0       /etc/file1
```

   El comando **`/usr/sbin/semanage fcontext -a -t samba_share_t /etc/archivo1`** agrega la siguiente entrada a **`/etc/selinux/targeted/contexts/files/file_contexts.local`**:

```
/etc/file1    unconfined_u:object_r:samba_share_t:s0
```

3. Como usuario root de Linux, ejecute el comando **`/sbin/restorecon -v /etc/archivo1`** para cambiar el tipo. Dado que el comando **`semanage`** agregó una entrada a **`file.contexts.local`** para **`/etc/archivo1`**, el comando **`/sbin/restorecon`** cambia el tipo a **`samba_share_t`**:

```
# /sbin/restorecon -v /etc/file1
restorecon reset /etc/file1 context unconfined_u:object_r:etc_t:s0-
>system_u:object_r:samba_share_t:s0
```

4. Como usuario root de Linux, ejecute el comando **`rm -i /etc/archivo1`** para borrar el **`archivo1`**.

5. Como usuario root de Linux, ejecute el comando **`/usr/sbin/semanage fcontext -d /etc/archivo1`** para eliminar el contexto agregado para **`/etc/archivo1`**. Cuando el contexto se elimina, ejecutando **`restorecon`** cambia el tipo a **`etc_t`**, en vez de **`samba_share_t`**.

Changing a Directory's Type

The following example demonstrates creating a new directory and changing that directory's file type, to a type used by Apache HTTP Server:

1. Como usuario root de LInux, ejecute el comando **`mkdir /web`** para crear un directorio nuevo. Este directorio se etiqueta con el tipo **`default_t`**:

```
# ls -dZ /web
drwxr-xr-x  root root unconfined_u:object_r:default_t:s0 /web
```

   La opción **`-d`** de **`ls`** hace que **`ls`** liste la información de un directorio, en vez de su contenido, y la opción **`-Z`** hace que **`ls`** muestre el contexto de SELinux (en este ejemplo, **`unconfined_u:object_r:default_t:s0`**).

2. Como usuario root de Linux, ejecute el comando **`/usr/sbin/semanage fcontext -a -t httpd_sys_content_t /web`** para cambiar el tipo de **`/web/`** a **`httpd_sys_content_t`**. La

opción **-a** agrega un nuevo registro, y la opción **-t** define un tipo (**httpd_sys_content_t**). Nota: la ejecución de este comando no cambia el tipo directamente - **/web/** todavía tiene la etiqueta de tipo **default_t**:

```
# /usr/sbin/semanage fcontext -a -t httpd_sys_content_t /web
# ls -dZ /web
drwxr-xr-x  root root unconfined_u:object_r:default_t:s0   /web
```

El comando **/usr/sbin/semanage fcontext -a -t httpd_sys_content_t /web** agrega la siguiente entrada a **/etc/selinux/targeted/contexts/files/file_contexts.local**:

```
/web    unconfined_u:object_r:httpd_sys_content_t:s0
```

3. Como usuario root de Linux, ejecute el comando **/sbin/restorecon -v /web** para cambiar el tipo. Como el comando **semanage** agregó una entrada a **file.contexts.local** para **/web**, el comando **/sbin/restorecon** cambia el tipo a **httpd_sys_content_t**:

```
# /sbin/restorecon -v /web
restorecon reset /web context unconfined_u:object_r:default_t:s0-
>system_u:object_r:httpd_sys_content_t:s0
```

Por defecto, los archivos y directorios recién creados heredan el tipo SELinux de sus carpetas padres. Cuando se usó este ejemplo, y antes de eliminar el contexto SELinux agregado para **/web/**, los archivos y directorios creados en el directorio **/web/** fueron etiquetados con el tipo **httpd_sys_content_t**.

4. Como usuario root de Linux, ejecute el comando **/usr/sbin/semanage fcontext -d /web** para borrar el contexto agregado para **/web/**.

5. Como usuario root de Linux, ejecute el comando **/sbin/restorecon -v /web** para restaurar el contexto predeterminado de SELinux.

Cambio de un Directorio y sus Tipos de Contenidos

The following example demonstrates creating a new directory, and changing the directory's file type (along with its contents) to a type used by Apache HTTP Server. The configuration in this example is used if you want Apache HTTP Server to use a different document root (instead of **/var/www/html/**):

1. Como usuario root de Linux, ejecute el comando **mkdir /web** para crear un directorio nuevo, y luego el comando **touch /web/archivo{1,2,3}** para crear 3 archivos vacíos (**archivo1**, **archivo2** y **archivo3**). El directorio **/web/** y los archivos en él son etiquetados con el tipo **default_t**:

```
# ls -dZ /web
drwxr-xr-x  root root unconfined_u:object_r:default_t:s0 /web
# ls -lZ /web
-rw-r--r--  root root unconfined_u:object_r:default_t:s0 file1
-rw-r--r--  root root unconfined_u:object_r:default_t:s0 file2
-rw-r--r--  root root unconfined_u:object_r:default_t:s0 file3
```

```
```

2. As the Linux root user, run the **/usr/sbin/semanage fcontext -a -t httpd_sys_content_t "/web(/.*)?"** command to change the type of the **/web/** directory and the files in it, to **httpd_sys_content_t**. The **-a** option adds a new record, and the **-t** option defines a type (httpd_sys_content_t). The **"/web(/.*)?"** regular expression causes the **semanage** command to apply changes to the **/web/** directory, as well as the files in it. Note: running this command does not directly change the type - **/web/** and files in it are still labeled with the **default_t** type:

```
# ls -dZ /web
drwxr-xr-x  root root unconfined_u:object_r:default_t:s0 /web
# ls -lZ /web
-rw-r--r--  root root unconfined_u:object_r:default_t:s0 file1
-rw-r--r--  root root unconfined_u:object_r:default_t:s0 file2
-rw-r--r--  root root unconfined_u:object_r:default_t:s0 file3
```

   The **/usr/sbin/semanage fcontext -a -t httpd_sys_content_t "/web(/.*)?"** command adds the following entry to **/etc/selinux/targeted/contexts/files/file_contexts.local**:

```
/web(/.*)?    system_u:object_r:httpd_sys_content_t:s0
```

3. Como usuario root de LInux, ejecute el comando **/sbin/restorecon -R -v /web** para cambiar elt ipo del directorio **/web/**, junto con los archivos dentro de él. La opción **-R** significa recursivo, es decir, todos los archivos y directorios dentro del directorio **/web/** se etiquetarán con el tipo **httpd_sys_content_t**. Dado que el comando **semanage** agregó una entrada en **file.contexts.local** para **/web(/.*)?**, el comando **/sbin/restorecon** los tipos a **httpd_sys_content_t**:

```
# /sbin/restorecon -R -v /web
restorecon reset /web context unconfined_u:object_r:default_t:s0-
>system_u:object_r:httpd_sys_content_t:s0
restorecon reset /web/file2 context unconfined_u:object_r:default_t:s0-
>system_u:object_r:httpd_sys_content_t:s0
restorecon reset /web/file3 context unconfined_u:object_r:default_t:s0-
>system_u:object_r:httpd_sys_content_t:s0
restorecon reset /web/file1 context unconfined_u:object_r:default_t:s0-
>system_u:object_r:httpd_sys_content_t:s0
```

   Por defecto, los archivos y directorios recién creados heredan el tipo SELinux de sus padres. En este ejemplo, los archivos y directorios creado en el directorio **/web/** se etiquetarán con el tipo **httpd_sys_content_t**.

4. As the Linux root user, run the **/usr/sbin/semanage fcontext -d "/web(/.*)?"** command to remove the context added for **"/web(/.*)?"**.

5. Como usuario root de Linux, ejecute el comando **/sbin/restorecon -R -v /web** para restaurar el contexto predeterminado de SELinux.

Borrado de un Contexto agregado

El siguiente ejemplo muestra el agregado y su eliminación del contexto de SELinux:

1. Como usuario root de Linux, ejecute el comando **/usr/sbin/semanage fcontext -a -t httpd_sys_content_t /prueba**. El directorio **/prueba/** no tiene que existir. Este comando agrega el siguiente contexto a **/etc/selinux/targeted/contexts/files/file_contexts.local**:

   ```
   /test    system_u:object_r:httpd_sys_content_t:s0
   ```

2. Para eliminar el contexto, como usuario root de Linux, ejecute el comando **/usr/sbin/semanage fcontext -d *nombre-de-archivo|nombre-de-directorio***, donde *nombre-de-archivo|nombre-de-directorio* es la primera parte en **file_contexts.local**. El siguiente es un ejemplo de un contexto en **file_contexts.local**:

   ```
   /test    system_u:object_r:httpd_sys_content_t:s0
   ```

   Siendo la primera parte **/prueba**. Para evitar que el directorio **/prueba/** se etiquete con **httpd_sys_content_t** después de ejecutar **/sbin/restorecon**, o después de un reetiquetado del sistema, ejecute el siguiente comando como usuario root de Linux para eliminar el contexto de **file_contexts.local**:

   /usr/sbin/semanage fcontext -d /test

Si el contexto es parte de una expresión regular, por ejemplo **/web(/.*)?**, use las comillas para encerrar la expresión regular:

/usr/sbin/semanage fcontext -d "/web(/.*)?"

Refer to the semanage(8) manual page for further information about **/usr/sbin/semanage**.

Importante

Cuando se cambia el contexto de SELinux con **/usr/sbin/semanage fcontext -a**, use la dirección completa del archivo o directorio para evitar etiquetar mal los archivos después de un reetiquetado del sistema de archivo, o después ejecutar el comando **/sbin/restorecon**.

5.8. Los tipos file_t y default_t

For file systems that support extended attributes, when a file that lacks an SELinux context on disk is accessed, it is treated as if it had a default context as defined by SELinux policy. In common policies, this default context uses the **file_t** type. This should be the only use of this type, so that files without a context on disk can be distinguished in policy, and generally kept inaccessible to confined domains. The **file_t** type should not exist on correctly-labeled file systems, because all files on a system running SELinux should have an SELinux context, and the **file_t** type is never used in file-context configuration[7].

[7] Files in **/etc/selinux/targeted/contexts/files/** define contexts for files and directories. Files in this directory are read by **restorecon** and **setfiles** to restore files and directories to their default contexts.

The **default_t** type is used on files that do not match any other pattern in file-context configuration, so that such files can be distinguished from files that do not have a context on disk, and generally kept inaccessible to confined domains. If you create a new top-level directory, such as **/mydirectory/**, this directory may be labeled with the **default_t** type. If services need access to such a directory, update the file-contexts configuration for this location. Refer to *Sección 5.7.2, "Cambios Persistentes: semanage fcontext"* for details on adding a context to the file-context configuration.

5.9. Montaje de Sistemas de Archivos

Por defecto, cuando un sistema de archivo que soporta atributos extendidos se monta, el contexto de seguridad para cada archivo se obtiene de atributo extendido *security.selinux* del archivo. A los archivos en sistemas de archivo que no dan soporte a atributos extendidos se les asigna un único contexto de seguridad predeterminado desde la configuración de la política, basada en el tipo de sistema de archivo.

Use el comando **mount -o context** para superponer los atributos extendidos actuales, o para especificar uno distinto y por defecto para sistemas de archivo que no dan soporte a atributos extendidos. Esto es útil si no confía en que un sistema de archivo provea los atributos correctos, por ejemplo, medios removibles en sistemas múltiples. El comando **mount -o context** también se puede ussar para dar soporte al etiquetado de sistemas de archivos que no soportan atributos extendidos, tales como la Tabla de Ubicación de Archivos (FAT) o los sistemas de archivo NFS. El contexto especificado con la opción **context** no se escribe al disco: los contextos originales son preservados, y se ven cuando se lo monta sin la opción **context** (si el sistema de archivo ya tenía sopoorte para atributos extendidos).

For further information about file system labeling, refer to James Morris's "Filesystem Labeling in SELinux" article: *http://www.linuxjournal.com/article/7426*.

5.9.1. Montajes de Contexto

Para montar un sistema de archivo con el contexto especificado, superponiendo los contextos existentes si existieran, o para especificar uno predeterminado distinto para un sistema de archivo que no da soporte para atributos extendidos, como usuario root de Linux, use el comando **mount -o context=*SELinux_user:role:type:level*** cuando monte el sistema de archivo deseado. Los cambios de contexto no se graban en el disco. Por defecto, los montajes NFS en el lado del cliente se etiquetan con un contexto distinto definido por una política para sistemas de archivo NFS. En políticas comunes, este contexto predeterminado usa el tipo **nfs_t**. Sin las opciones de montaje adicionales, esto podría evitar el que sistemas de archivo NFS sean compartidos vía otros servicios, como el Servidor HTTP Apache. El siguiente ejemplo monta un sistema de archivo NFS para que se pueda acceder a través del Servidor HTTP Apache:

```
# mount server:/export /local/mount/point -o\
context="system_u:object_r:httpd_sys_content_t:s0"
```

Los archivos y directorios recién creados en este sistema de archivo parecen tener un contexto SELinux especificado con **-o contexto**; sin embargo, dado que los cambios del contexto no se escriben en el disco en estas situaciones, el contexto especificado por la opción **-o contexto** sólo se mantiene si se usa la misma opción en la siguiente montada, y si además se especifica el mismo contexto.

La Obligación de Tipo es el control de permiso principal en la política destinada de SELinux. Para la mayor parte, los usuarios y roles de SELinux se pueden ignorar, por lo que, cuando se superponga

el contexto de SELinux con **-o context**, use el usuario SELinux **system_u** y el rol **object_r**, y concéntrese en el tipo. Si no está usando la política MLS o seguridad multi-categoría, use el nivel **s0**.

Nota

Cuando se monta un sistema de archivo con la opción **context**, los cambios de contexto (por usuarios y procesos) son prohibidos. Por ejemplo, ejecutando **chcon** en un sistema de archivo montado con la opción **context** resulta en un error de **Operación no soportada**.

5.9.2. Cambio del Contexto Predeterminado

As mentioned in *Sección 5.8, "Los tipos file_t y default_t"*, on file systems that support extended attributes, when a file that lacks an SELinux context on disk is accessed, it is treated as if it had a default context as defined by SELinux policy. In common policies, this default context uses the **file_t** type. If it is desirable to use a different default context, mount the file system with the **defcontext** option.

El siguiente ejemplo monta un sistema de archivo recién creado (en **/dev/sda2**) en el directorio recién creado **/prueba/**. Asume que no hay reglas en **/etc/selinux/targeted/contexts/files/** que definan el contexto del directorio **/prueba/**:

```
# mount /dev/sda2 /test/ -o defcontext="system_u:object_r:samba_share_t:s0"
```

En este ejemplo:

- the **defcontext** option defines that **system_u:object_r:samba_share_t:s0** is "the default security context for unlabeled files"[8].
- cuando sea montado, el directorio raíz (**/prueba/**) del sistema de archivo se trata como si estuviera etiquetado con el contexto especificado por **defcontext** (esta etiqueta no se guarda en el disco). Esto afecta el etiquetado de archvios creados en **/prueba/**: los archivos nuevos heredan el tipo **samba_share_t**, y estas etiquetas se guardan en el disco.
- los archivos creados bajo **/prueba/** mientras el sistema de archivo estaba montado con la opción **defcontext** retendrán sus etiquetas.

5.9.3. Montando un Sistema de Archivos NFS

Por defecto, los montajes NFS en el lado del cliente son etiquetados con un contexto predeterminado por la política para los sistemas de archivo NFS. En políticas comunes, este contexto predeterminado usa el tipo **nfs_t**. Dependiendo de la configuración de la política, los servicios, como el Servidor HTTP Apache y MySQL, pueden no poder leer archivos etiquetados con el tipo **nfs_t**. Esto puede prevenir que los sistemas de archivos etiquetados con este tipo se monten y sean leídos o exportados por otros servicios.

Si desea montar un sistema de archivo NFS y leer o exportar ese sistema de archivo con otro servicio, use la opción **contexto** cuando monte para anular el tipo **nfs_t**. Use la siguiente opción de contexto para montar sistemas de archivo NFS para que puedan compartirse vía el Servidor HTTP Apache:

```
mount server:/export /local/mount/point -o\
context="system_u:object_r:httpd_sys_content_t:s0"
```

Dado que los cambios de contexto no se escriben al disco para estas situaciones, el contexto especificado con la opción **context** sólo se retiene si la opción **context** se usa en el siguiente montaje, y si el mismo contexto se especifica.

As an alternative to mounting file systems with **context** options, Booleans can be turned on to allow services access to file systems labeled with the **nfs_t** type. Refer to *Sección 5.6.3, "Booleanos para NFS y CIFS"* for instructions on configuring Booleans to allow services access to the **nfs_t** type.

5.9.4. Montajes NFS Múltiples

Cuando se monten múltiples montajes desde el mismo NFS exportado, el intento de sobreescribir el contexto de SELinux e cada montaje con un contexto diferente, resulta en fallos de los comandos de montaje subsecuentes. En el siguiente ejemplo, el servidor NFS tiene un exportado único, **/export**, que tiene dos subdirectorios, **web/** and **database/**. El siguiente comando intenta dos montajes desde un único export NFS e intenta sobreescribir el contexto para cada uno:

```
# mount server:/export/web /local/web -o\
context="system_u:object_r:httpd_sys_content_t:s0"

# mount server:/export/database /local/database -o\
context="system_u:object_r:mysqld_db_t:s0"
```

El segundo comando mount falla, y se graba lo siguiente en **/var/log/messages**:

```
kernel: SELinux: mount invalid.  Same superblock, different security settings for (dev 0:15,
 type nfs)
```

Para montar montajes múltiples de un exportado NFS único, con cada montaje teniendo un contexto diferente, use las opciones **-o nosharecache,context**. El siguiente ejemplo monta montajes múltiples de un único export de NSF, con un contexto diferente para cada montaje (permitiendo un único acceso de servicio a cada uno):

```
# mount server:/export/web /local/web -o\
nosharecache,context="system_u:object_r:httpd_sys_content_t:s0"

# mount server:/export/database /local/database -o\
nosharecache,context="system_u:object_r:mysqld_db_t:s0"
```

En este ejemplo, **server:/export/web** se monta localmente en **/local/web/**, con todos los archivos etiquetados con el tipo **httpd_sys_content_t**, lo que permite el acceso al Servidor HTTP Apache. **server:/export/database** está montado localmente en **/local/database**, con los archivos etiquetados con el tipo **mysqld_db_t**, lo que permite a MySQL el acceso. Estos cambios de tipo no se escriben en el disco.

Importante

Las opciones **nosharecache** le permiten montar el mismo subdirectorio de un exportado varias veces con distintos contextos (por ejemplo, montar **/export/web** varias veces). No monte el mismo directorio de un exportado varias veces con distintos contextos, dado que esto crea un montado solapado, donde los archivos se pueden acceder con dos contextos diferentes.

5.9.5. Haciendo Persistente los Contextos de Montajes

Para hacer que los contextos de montajes persistentes entre remontadas y reiniciadas, agregue las entradas de los sistemas de archivos en **/etc/fstab** o un mapa de automontador, y use el contexto deseado como una opción de montaje. El siguiente ejemplo agrega una entrada en **/etc/fstab** para un montaje de contexto NFS:

```
server:/export /local/mount/ nfs context="system_u:object_r:httpd_sys_content_t:s0" 0 0
```

Refer to the *Red Hat Enterprise Linux 5 Deployment Guide, Section 19.2. "NFS Client Configuration"*[9] for information about mounting NFS file systems.

5.10. Mantención de las Etiquetas de SELinux

Estas secciones describen qué les pasa a los contextos SELinux cuando se copia, mueve y compacta archivos y directorios. También explica cómo preservar los contextos cuando se copia o se compacta.

5.10.1. Copia de Directorios y Archivos

When a file or directory is copied, a new file or directory is created if it does not exist. That new file or directory's context is based on default-labeling rules, not the original file or directory's context (unless options were used to preserve the original context). For example, files created in user home directories are labeled with the **user_home_t** type:

```
$ touch file1
$ ls -Z file1
-rw-rw-r--  user1 group1 unconfined_u:object_r:user_home_t:s0 file1
```

Si un archivo se copia a otro directorio, tal como **/etc/**, el archivo nuevo se crea de acuerdo a las reglas de etiquetado predeterminado del directorio **/etc/**. El copiado de un archivo (sin opciones adicionales) puede no preservar el contexto original:

```
$ ls -Z file1
-rw-rw-r--  user1 group1 unconfined_u:object_r:user_home_t:s0 file1
# cp file1 /etc/
$ ls -Z /etc/file1
-rw-r--r--  root root unconfined_u:object_r:etc_t:s0   /etc/file1
```

[9] http://www.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5.2/html/Deployment_Guide/s1-nfs-client-config.html

Cuando el **archivo1** se copia a **/etc/**, si **/etc/archivo1** no existe, **/etc/archivo1** se crea como un archivo nuevo. Como se muestra en el ejemplo de arriba, **/etc/archivo1** se etiqueta con el tipo **etc_t**, de acuerdo con las reglas de etiquetado predeterminadas.

When a file is copied over an existing file, the existing file's context is preserved, unless the user specified **cp** options to preserve the context of the original file, such as **--preserve=context**. SELinux policy may prevent contexts from being preserved during copies.

Copia sin Preservar los Contextos de SELinux

Cuando se copia un archivo con el comando **cp**, si no se dan opciones, el tipo se hereda desde el directorio padre destino:

```
$ touch file1
$ ls -Z file1
-rw-rw-r--  user1 group1 unconfined_u:object_r:user_home_t:s0 file1
$ ls -dZ /var/www/html/
drwxr-xr-x  root root system_u:object_r:httpd_sys_content_t:s0 /var/www/html/
# cp file1 /var/www/html/
$ ls -Z /var/www/html/file1
-rw-r--r--  root root unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/file1
```

In this example, **file1** is created in a user's home directory, and is labeled with the **user_home_t** type. The **/var/www/html/** directory is labeled with the **httpd_sys_content_t** type, as shown with the **ls -dZ /var/www/html/** command. When **file1** is copied to **/var/www/html/**, it inherits the **httpd_sys_content_t** type, as shown with the **ls -Z /var/www/html/file1** command.

Preservación de los Contextos de SELinux cuando se copia

Use el comando **cp --preserve=context** para preservar los contextos cuando se copia:

```
$ touch file1
$ ls -Z file1
-rw-rw-r--  user1 group1 unconfined_u:object_r:user_home_t:s0 file1
$ ls -dZ /var/www/html/
drwxr-xr-x  root root system_u:object_r:httpd_sys_content_t:s0 /var/www/html/
# cp --preserve=context file1 /var/www/html/
$ ls -Z /var/www/html/file1
-rw-r--r--  root root unconfined_u:object_r:user_home_t:s0 /var/www/html/file1
```

In this example, **file1** is created in a user's home directory, and is labeled with the **user_home_t** type. The **/var/www/html/** directory is labeled with the **httpd_sys_content_t** type, as shown with the **ls -dZ /var/www/html/** command. Using the **--preserve=context** option preserves SELinux contexts during copy operations. As shown with the **ls -Z /var/www/html/file1** command, the **file1** **user_home_t** type was preserved when the file was copied to **/var/www/html/**.

Copiado y Cambio del Contexto

Use the **cp -Z** command to change the destination copy's context. The following example was performed in the user's home directory:

```
$ touch file1
$ cp -Z system_u:object_r:samba_share_t:s0 file1 file2
$ ls -Z file1 file2
-rw-rw-r--  user1 group1 unconfined_u:object_r:user_home_t:s0 file1
-rw-rw-r--  user1 group1 system_u:object_r:samba_share_t:s0 file2
$ rm file1 file2
```

En este ejemplo, el contexto se define en la opción **-Z**. Sin la opción **-Z**, **archivo2** se etiquetaría con el contexto **unconfined_u:object_r:user_home_t**.

Copia de un Archivos sobre un otro existente

When a file is copied over an existing file, the existing file's context is preserved (unless an option is used to preserve contexts). For example:

```
# touch /etc/file1
# ls -Z /etc/file1
-rw-r--r--  root root unconfined_u:object_r:etc_t:s0   /etc/file1
# touch /tmp/file2
# ls -Z /tmp/file2
-rw-r--r--  root root unconfined_u:object_r:user_tmp_t:s0 /tmp/file2
# cp /tmp/file2 /etc/file1
# ls -Z /etc/file1
-rw-r--r--  root root unconfined_u:object_r:etc_t:s0   /etc/file1
```

En este ejemplo, se crean dos archivos: **/etc/archivo1**, etiquetado con el tipo **etc_t**, y **/tmp/archivo2**, etiquetado con el tipo **user_tmp_t**. Ell comando **cp /tmp/archivo2 /etc/archivo1** sobreescribe **archivo1** con **archivo2**. Después de copiar, el comando **ls -Z /etc/archivo1** muestra a **archivo1** etiquetado con el tipo **etc_t**, en vez del **user_tmp_t** de **/tmp/archivo2** que reemplazó a **/etc/archivo1**.

Importante

Copie archivos y directorios, en vez de moverlos. Esto ayuda a asegurar que se etiquetan con los contextos de SELinux correctos. Los contextos SELinux incorrectos pueden hacer que los procesos no puedan acceder a esos archivos y directorios.

5.10.2. Movimiento de Archivos y Directorios

File and directories keep their current SELinux context when they are moved. In many cases, this is incorrect for the location they are being moved to. The following example demonstrates moving a file from a user's home directory to **/var/www/html/**, which is used by the Apache HTTP Server. Since the file is moved, it does not inherit the correct SELinux context:

1. Ejecute el comando **cd** sin ningún argumento para cambiar a su directorio de inicio. Una vez ahí, ejecute el comando **touch archivo1** para crear un archivo. Este archivo se etiqueta con el tipo **user_home_t**:

   ```
   $ ls -Z file1
   ```

```
-rw-rw-r--  user1 group1 unconfined_u:object_r:user_home_t:s0 file1
```

2. Ejecute el comando **ls -dZ /var/www/html/** para ver el contexto de SELinux del directorio **/var/www/html/**:

```
$ ls -dZ /var/www/html/
drwxr-xr-x  root root system_u:object_r:httpd_sys_content_t:s0 /var/www/html/
```

   Por defecto, el directorio **/var/www/html/** se etiqueta con el tipo **httpd_sys_content_t**. Los archivos y directorios creados bajo el directorio **/var/www/html/** heredan este tipo, y como tal, son etiquetados con este tipo.

3. Como usuario root de Linux, ejecute el comando **mv archivo1 /var/www/html/** para mover el **archivo1** al directorio **/var/www/html/**. Dado que el archivo es movido, mantiene su tipo **user_home_t** actual:

```
# mv file1 /var/www/html/
# ls -Z /var/www/html/file1
-rw-rw-r--  user1 group1 unconfined_u:object_r:user_home_t:s0 /var/www/html/file1
```

Por defecto, el Servidor HTTP Apache no puede leer archivos etiquetados con el tipo **user_home_t**. Si todos los archivos de una página web se etiquetaron con **user_home_t**, u otro tipo al que el Servidor HTTP Apache no puede leer, el permiso es negado cuando intente accederlo vía Firefox o algún otro navegador web basado en texto.

Importante

Mover archivos y directorios con el comando **mv** puede resultar en el contexto SELinux incorrecto, evitando que los procesos tales como el Servidor HTTP Apache y Samba puedan acceder a tales archivos y directorios.

5.10.3. Chequeando el Contexto SELinux Predeterminado

Use the **/usr/sbin/matchpathcon** command to check if files and directories have the correct SELinux context. From the matchpathcon(8) manual page: "**matchpathcon** queries the system policy and outputs the default security context associated with the file path."[10]. The following example demonstrates using the **/usr/sbin/matchpathcon** command to verify that files in **/var/www/html/** directory are labeled correctly:

1. Como usuario root de Linux, ejecute el comando **touch /var/www/html/archivo{1,2,3}** para crear tres archivos (**archivo1**, **archivo2** y **archivo3**). Estos heredan el tipo **httpd_sys_content_t** del directorio **/var/www/html/**:

```
# touch /var/www/html/file{1,2,3}
# ls -Z /var/www/html/
-rw-r--r--  root root unconfined_u:object_r:httpd_sys_content_t:s0 file1
-rw-r--r--  root root unconfined_u:object_r:httpd_sys_content_t:s0 file2
```

[10] The matchpathcon(8) manual page, as shipped with the *libselinux-utils* package in Fedora, is written by Daniel Walsh. Any edits or changes in this version were done by Murray McAllister.

```
-rw-r--r--  root root unconfined_u:object_r:httpd_sys_content_t:s0 file3
```

2. Como usuario root de Linux, ejecute el comando **chcon -t samba_share_t /var/www/html/archivo1** para cambiar el tipo del **archivo1** a **samba_share_t**. Nota: El Servidor HTTP Apache no puede leer archivos o directorios etiquetados con el tipo **samba_share_t**.

3. La opción **/usr/sbin/matchpathcon -V** compara el contexto SELinux actual con el contexto predeterminado correcto dado por la política de SELinux. Ejecute el comando **/usr/sbin/matchpathcon -V /var/www/html/*** para chequear todos los archivos del directorio **/var/www/html/**:

```
$ /usr/sbin/matchpathcon -V /var/www/html/*
/var/www/html/file1 has context unconfined_u:object_r:samba_share_t:s0, should be
 system_u:object_r:httpd_sys_content_t:s0
/var/www/html/file2 verified.
/var/www/html/file3 verified.
```

La siguiente salida del comando **/usr/sbin/matchpathcon** explica que el **archivo1** está etiquetado con el tipo **samba_share_t**, pero debería estar etiquetado con el tipo **httpd_sys_content_t**:

```
/var/www/html/file1 has context unconfined_u:object_r:samba_share_t:s0, should be
 system_u:object_r:httpd_sys_content_t:s0
```

Para resolver el problema de etiqueta y permitir al Servidor HTTP Apache acceder a **archivo1**, como usuario root de Linux corra el comando **/sbin/restorecon -v /var/www/html/archivo1**:

```
# /sbin/restorecon -v /var/www/html/file1
restorecon reset /var/www/html/file1 context unconfined_u:object_r:samba_share_t:s0-
>system_u:object_r:httpd_sys_content_t:s0
```

5.10.4. Archivando archivos con tar

tar no retiene los atributos extendidos por defecto. Dado que los contextos SELinux se almacenan en los atributos extendidos, los contextos se pueden perder cuando se compactan archivos. Use **tar --selinux** para crear archivos que retengan los contextos. Si un archivo Tar contiene archivos sin los atributos extendidos, o si quiere que los atributos extendidos coincidan con los predeterminados del sistema, ejecute el archivado a través de **/sbin/restorecon**:

```
$ tar -xvf archive.tar | /sbin/restorecon -f -
```

Nota: dependiendo del directorio, puede necesitar ser el usuario root de Linux para ejecutar el comando **/sbin/restorecon**.

El siguiente ejemplo muestra la creación de un archivo Tar que mantiene sus contextos SELinux:

1. Como usuario root de Linux, ejecute el comando **`touch /var/www/html/archivo{1,2,3}`** para crear tres archivos (**`archivo1`**, **`archivo2`** y **`archivo3`**). Estos heredan el tipo **`httpd_sys_content_t`** del directorio **`/var/www/html/`**:

```
# touch /var/www/html/file{1,2,3}
# ls -Z /var/www/html/
-rw-r--r--  root root unconfined_u:object_r:httpd_sys_content_t:s0 file1
-rw-r--r--  root root unconfined_u:object_r:httpd_sys_content_t:s0 file2
-rw-r--r--  root root unconfined_u:object_r:httpd_sys_content_t:s0 file3
```

2. Ejecute el comando **`cd /var/www/html/`** para cambiar al directorio **`/var/www/html/`**. Una vez en este directorio, como usuario root de Linux ejecute el comando **`tar --selinux -cf prueba.tar archivo{1,2,3}`** para crear un archivo Tar con nombre **`prueba.tar`**.

3. Como usuario root de Linux, corra el comando **`mkdir /prueba`** para crear un directorio nuevo, y luego ejecute el comando **`chmod 777 /prueba/`** para permitir a los usuarios acceso total al directorio **`/prueba/`**.

4. Ejecute el comando **`cp /var/www/html/prueba.tar /prueba/`** para copiar el archivo **`prueba.tar`** en el directorio **`/prueba/`**.

5. Ejecute el comando **`cd /prueba/`** para cambiar al directorio **`/test/`**. Una vez ahí, ejecute el comando **`tar -xvf prueba.tar`** para extraer el archivo Tar.

6. Ejecute el comando **`ls -lZ /prueba/`** para ver los contextos SELinux. El tipo **`httpd_sys_content_t`** fue retenido, en vez de haberse cambiado al **`default_t`**, lo que hubiera pasado si la opción **`--selinux`** no se hubiera usado:

```
$ ls -lZ /test/
-rw-r--r--  user1 group1 unconfined_u:object_r:httpd_sys_content_t:s0 file1
-rw-r--r--  user1 group1 unconfined_u:object_r:httpd_sys_content_t:s0 file2
-rw-r--r--  user1 group1 unconfined_u:object_r:httpd_sys_content_t:s0 file3
-rw-r--r--  user1 group1 unconfined_u:object_r:default_t:s0 test.tar
```

7. Si el directorio **`/prueba/`** no se necesita más, como usuario root de Linux ejecute el comando **`rm -ri /prueba/`** para eliminarlo, así como todos los archivos en él.

Refer to the tar(1) manual page for further information about **`tar`**, such as the **`--xattrs`** option that retains all extended attributes.

5.10.5. Archivando archivos con tar

`star` no retiene los atributos extendidos por defecto. Dado que los contextos SELinux se almacenan en los atributos extendidos, los contextos se pueden perder cuando se crean esos archivos. Use **`star -xattr -H=exustar`** para crear archivos que retengan los contextos. El paquete *star* no se instala por defecto. Para instalar **`star`**, ejecute el comando **`yum install star`** como usuario root de Linux.

El siguiente ejemplo muestra la creación de un archivo Star que retiene los contextos SELinux:

1. Como usuario root de Linux, ejecute el comando **`touch /var/www/html/archivo{1,2,3}`** para crear tres archivos (**`archivo1`**, **`archivo2`** y **`archivo3`**). Estos heredan el tipo **`httpd_sys_content_t`** del directorio **`/var/www/html/`**:

```
# touch /var/www/html/file{1,2,3}
# ls -Z /var/www/html/
-rw-r--r--  root root unconfined_u:object_r:httpd_sys_content_t:s0 file1
-rw-r--r--  root root unconfined_u:object_r:httpd_sys_content_t:s0 file2
-rw-r--r--  root root unconfined_u:object_r:httpd_sys_content_t:s0 file3
```

2. Ejecute el comando **`cd /var/www/html/`** para cambiar al directorio **`/var/www/html/`**. Una vez en este directorio, como usuario root de Linux ejecute el comando **`star -xattr -H=exustar -c -f=prueba.star archivo{1,2,3}`** para crear un archivo Star llamado **`prueba.star`**:

```
# star -xattr -H=exustar -c -f=test.star file{1,2,3}
star: 1 blocks + 0 bytes (total of 10240 bytes = 10.00k).
```

3. Como usuario root de Linux, corra el comando **`mkdir /prueba`** para crear un directorio nuevo, y luego ejecute el comando **`chmod 777 /prueba/`** para permitir a los usuarios acceso total al directorio **`/prueba/`**.

4. Ejecute el comando **`cp /var/www/html/prueba.star /prueba/`** para copiar el archivo **`prueba.star`** al directorio **`/prueba/`**.

5. Ejecute el comando **`cd /prueba/`** para cambiar al directorio **`/prueba/`**. Una vez ahí, ejecute el comando **`star -x -f=prueba.star`** para extraer el archivo Star:

```
$ star -x -f=test.star
star: 1 blocks + 0 bytes (total of 10240 bytes = 10.00k).
```

6. Ejecute el comando **`ls -lZ /prueba/`** para ver los contextos SELinux. El tipo **`httpd_sys_content_t`** fue retenido, en vez de haberse cambiado al **`default_t`**, lo que hubiera pasado si la opción **`--selinux`** no se hubiera usado:

```
$ ls -lZ /test/
-rw-r--r--  user1 group1 unconfined_u:object_r:httpd_sys_content_t:s0 file1
-rw-r--r--  user1 group1 unconfined_u:object_r:httpd_sys_content_t:s0 file2
-rw-r--r--  user1 group1 unconfined_u:object_r:httpd_sys_content_t:s0 file3
-rw-r--r--  user1 group1 unconfined_u:object_r:default_t:s0 test.star
```

7. Si el directorio **`/prueba/`** no se necesita más, como usuario root de Linux ejecute el comando **`rm -ri /prueba/`** para eliminarlo, así como todos los archivos en él.

8. Si **`star`** ya no se necesita, como usuario root de Linux, ejecute el comando **`yum remove star`** para eliminar el paquete.

Refer to the star(1) manual page for further information about **`star`**.

Confinando a los Usuarios

A number of confined SELinux users are available in Fedora 13. Each Linux user is mapped to an SELinux user via SELinux policy, allowing Linux users to inherit the restrictions placed on SELinux users, for example (depending on the user), not being able to: run the X Window System; use networking; run setuid applications (unless SELinux policy permits it); or run the **su** and **sudo** commands. This helps protect the system from the user. Refer to *Sección 4.3, "Usuarios Confinados y no Confinados"* for further information about confined users.

6.1. Linux y los Mapeos de Usuarios de SELinux

Como usuario root de Linux, corra el comando **semanage login -l** para ver el mapeo entre los usuarios de Linux y los usuarios de SELinux:

```
# /usr/sbin/semanage login -l

Login Name                SELinux User              MLS/MCS Range

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

In Fedora 13, Linux users are mapped to the SELinux **__default__** login by default (which is in turn mapped to the SELinux **unconfined_u** user). When a Linux user is created with the **useradd** command, if no options are specified, they are mapped to the SELinux **unconfined_u** user. The following defines the default-mapping:

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

6.2. Confinando Usuarios Nuevos de Linux: useradd

Los usuarios Linux mapeados al usuario SELinux **unconfined_u** corren en el dominio **unconfined_t**. Esto se ve ejecutando el comando **id -Z** luego de haber ingresado como el usuario Linux que se mapea a **unconfined_u**:

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

Cuando los usuarios linux ejecuten en el dominio **unconfined_t**, se aplican las reglas de la política de SELinux, pero las reglas de políticas que existen para usuarios Linux que corren en el dominio **unconfined_t** permiten casi todos los accesos. Si los usuarios Linux no confinados ejecutan una aplicación que la política de SELinux define pueden transicionar desde el dominio **unconfined_t** a su propio dominio confinado, los usuarios Linux no confinados todavía pueden ser sujetos a restricciones del dominio confinado. El beneficio de seguridad de esto es que, aunque el usuario Linux corre en un dominio confinado, la aplicación permanece confinada, y por lo tanto, la explotación de una brecha en la aplicación se puede limitar por la política. Nota: esto no protege al sistema del usuario. En su lugar, el usuario y el sistema están siendo protegido de posibles daños causados en alguna debilidad en la aplicación.

Cuando se crean usuarios LInux con **useradd**, use la opción **-Z** para especificar a qué usuario SELinux se debe mapear. El siguiente ejemplo crea un usuario Linux nuevo, useruuser, y mapea ese usuario al usuario SELinux **user_u**. Los usuarios Linux mapeados al usuario SELinux **user_u** corren en el dominio **user_t**. En este dominio, los usuarios Linux no pueden correr aplicaciones setuid a menos que la política de SELinux lo permita (tal como **passwd**), y tampoco pueden correr **su** o **sudo**, lo que evita que se puedan volver usuarios root de Linux con estos comandos.

1. Como usuario root de Linux, corra el comando **/usr/sbin/useradd -Z user_u useruuser** para crear el usuario Linux nuevo (useruuser) que se mapeará al usuario SELinux **user_u**.

2. Como usuario root de Linux, corra el comando **semanage login -l** para ver el mapeo entre el usuario Linux **useruuser** y **user_u**:

```
# /usr/sbin/semanage login -l

Login Name                SELinux User              MLS/MCS Range

__default__               unconfined_u              s0-s0:c0.c1023
root                      unconfined_u              s0-s0:c0.c1023
system_u                  system_u                  s0-s0:c0.c1023
useruuser                 user_u                    s0
```

3. Como usuario root de Linux, corra el comando **passwd useruuser** para asignar una contraseña para el usuario useruuser de Linux:

```
# passwd useruuser
Changing password for user useruuser.
New UNIX password: Enter a password
Retype new UNIX password: Enter the same password again
passwd: all authentication tokens updated successfully.
```

4. Log out of your current session, and log in as the Linux useruuser user. When you log in, pam_selinux maps the Linux user to an SELinux user (in this case, **user_u**), and sets up the resulting SELinux context. The Linux user's shell is then launched with this context. Run the **id -Z** command to view the context of a Linux user:

```
[useruuser@localhost ~]$ id -Z
user_u:user_r:user_t:s0
```

5. Log out of the Linux useruuser's session, and log back in with your account. If you do not want the Linux useruuser user, run the **/usr/sbin/userdel -r useruuser** command as the Linux root user to remove it, along with its home directory.

6.3. Confinando Usuarios Linux Existentes: semanage login

Si un usuario Linux se mapea al usuario **unconfined_u** (el comportamiento predeterminado), y desea cambiar le usuario SELinux al que se mapea, use el comando **semanage login**. El siguiente

ejemplo crea un usuario de Linux nuevo llamado usuarionuevo, luego lo mapea al usuario SELinux **user_u**:

1. Como usuario root de Linux, ejecute el comando **/usr/sbin/useradd usuarionuevo** para crear un nuevo usuario (usuarionuevo). Dado que este usuario usa el mapeo por defecto, no aparece en la salida de **/usr/sbin/semanage login -l**:

```
# /usr/sbin/semanage login -l

Login Name                SELinux User              MLS/MCS Range

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

2. Para mapear un usuario usuarionuevo de Linux al usuario **user_u** de SELinux, corra el siguiente comando como usuario root de Linux:

   /usr/sbin/semanage login -a -s user_u newuser

   La opción **-a** agrega un registro nuevo y la opción **-s** especifica el usuario SELinux al que mapea el usuario Linux. El último argumento **usuarionuevo**, es el usuario Linux al que quiere que se mapee el usuario SELinux especificado.

3. Para ver el mapeo entre el usuario usarionuevo de Linux y **user_u**, corra el comando **semanage login -l** como usuario root de Linux:

```
# /usr/sbin/semanage login -l

Login Name                SELinux User              MLS/MCS Range

__default__               unconfined_u              s0-s0:c0.c1023
newuser                   user_u                    s0
root                      unconfined_u              s0-s0:c0.c1023
system_u                  system_u                  s0-s0:c0.c1023
```

4. As the Linux root user, run the **passwd newuser** command to assign a password to the Linux newuser user:

```
# passwd newuser
Changing password for user newuser.
New UNIX password: Enter a password
Retype new UNIX password: Enter the same password again
passwd: all authentication tokens updated successfully.
```

5. Log out of your current session, and log in as the Linux newuser user. Run the **id -Z** command to view the newuser's SELinux context:

```
[newuser@rlocalhost ~]$ id -Z
```

```
user_u:user_r:user_t:s0
```

6. Log out of the Linux newuser's session, and log back in with your account. If you do not want the Linux newuser user, run the **userdel -r newuser** command as the Linux root user to remove it, along with its home directory. Also, the mapping between the Linux newuser user and **user_u** is removed:

```
# /usr/sbin/userdel -r newuser
# /usr/sbin/semanage login -l

Login Name                SELinux User              MLS/MCS Range

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

6.4. Cambiando el Mapeo Predeterminado

In Fedora 13, Linux users are mapped to the SELinux **__default__** login by default (which is in turn mapped to the SELinux **unconfined_u** user). If you would like new Linux users, and Linux users not specifically mapped to an SELinux user to be confined by default, change the default mapping with the **semanage login** command.

Por ejemplo, corra el siguiente comando como usuario root de Linux para cambiar el mapeo predeterminado de **unconfined_u** a **user_u**:

/usr/sbin/semanage login -m -S targeted -s "user_u" -r s0 __default__

Corra el comando **semanage login -l** como usuario root de Linux para verificar que el ingreso **__default__** se mapea a **user_u**:

```
# /usr/sbin/semanage login -l

Login Name                SELinux User              MLS/MCS Range

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

Si un usuario Linux nuevo se crea y el usuario SELinux no se especifica, o si un usuario Linux existente ingresa y no coincide una entrada específica de la salida de **semanage login -l**, se mapean a **user_u**, según el ingreso **__default__**.

Para volver al comportamiento predeterminado, corra el siguiente comando como usuario root de Linux para mapear el ingreso **__default__** al usuario SELinux **unconfined_u**:

```
/usr/sbin/semanage login -m -S targeted -s "unconfined_u" -r\
s0-s0:c0.c1023 __default__
```

6.5. xguest: Modo Kiosk

The *xguest* package provides a kiosk user account. This account is used to secure machines that people walk up to and use, such as those at libraries, banks, airports, information kiosks, and coffee shops. The kiosk user account is very limited: essentially, it only allows users to log in and use **Firefox** to browse Internet websites. Any changes made while logged in with his account, such as creating files or changing settings, are lost when you log out.

Para configurar la cuenta kiosk:

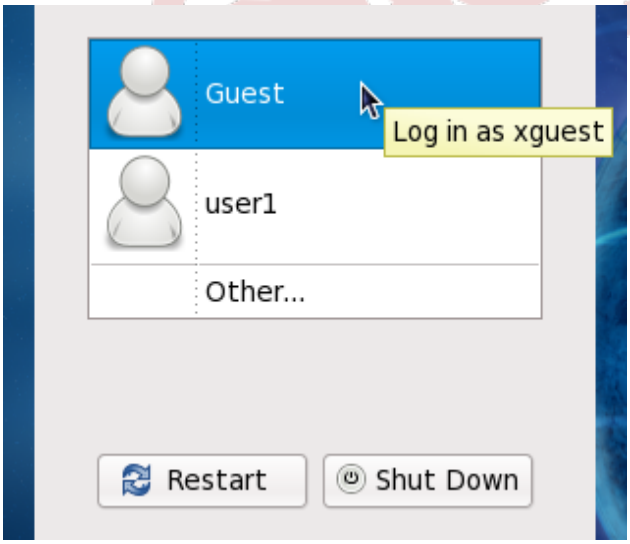
1. Como usuario root de Linux, corra el comando **`yum install xguest`** para instalar el paquete *xguest*. Instale las dependencias requeridas.

2. Para permitir a la cuenta kiosk usarse para una variedad de personas, la cuenta no se protege con contraseña, y como tal, la cuenta sólo se puede proteger si SELinux está funcionando en modo obediente. Antes de ingresar con esta cuenta, use el comando **`getenforce`** para confirmar que SELinux está funcionando en modo obediente:

   ```
   $ /usr/sbin/getenforce
   Enforcing
   ```

   If this is not the case, refer to *Sección 5.5, "Modos de SELinux"* for information about changing to enforcing mode. It is not possible to log in with this account if SELinux is in permissive mode or disabled.

3. Solamente puede ingresar a esta cuenta a través del Administración de Pantalla de GNOME (GDM). Una vez que el paquete *xguest* se instala, se agrega una cuenta **`Invitado`** a GDM. Para ingresar, haga clic en la cuenta **`Invitado`**:



6.6. Booleanos para que los Usuarios Ejecuten Aplicaciones

Not allowing Linux users to execute applications (which inherit users' permissions) in their home directories and **`/tmp/`**, which they have write access to, helps prevent flawed or malicious applications

from modifying files that users own. In Fedora 13, by default, Linux users in the **guest_t** and **xguest_t** domains can not execute applications in their home directories or **/tmp/**; however, by default, Linux users in the **user_t** and **staff_t** domains can.

Hay booleanos disponibles para cambiar este comportamiento, y se configuran con el comando **setsebool**. El comando **setsebool** se debe usar con el usuario root de Linux. El comando **setsebool -P** hace los cambios persistentes. No use la opción **-P** si no quiere que los cambios persistan entre reiniciadas:

guest_t

Para *permitir* a los usuarios Linux en el dominio **guest_t** que ejecuten aplicaciones en sus directorios de inicio y en **/tmp/**:

```
/usr/sbin/setsebool -P allow_guest_exec_content on
```

xguest_t

Para *permitir* a los usuarios Linux en el dominio **xguest_t** ejecutar aplicaciones en sus directorios inicios y **/tmp/**:

```
/usr/sbin/setsebool -P allow_xguest_exec_content on
```

user_t

Para *impedir* que los usuarios Linux en el dominio **user_t** ejecuten aplicaciones en sus directorios de inicio y **/tmp/**:

```
/usr/sbin/setsebool -P allow_user_exec_content off
```

staff_t

Para *impedir* que los usuarios Linux en el dominio **staff_t** ejecuten aplicaciones en sus directorios de inicio y en **/tmp/**:

```
/usr/sbin/setsebool -P allow_staff_exec_content off
```

Solución a Problemas

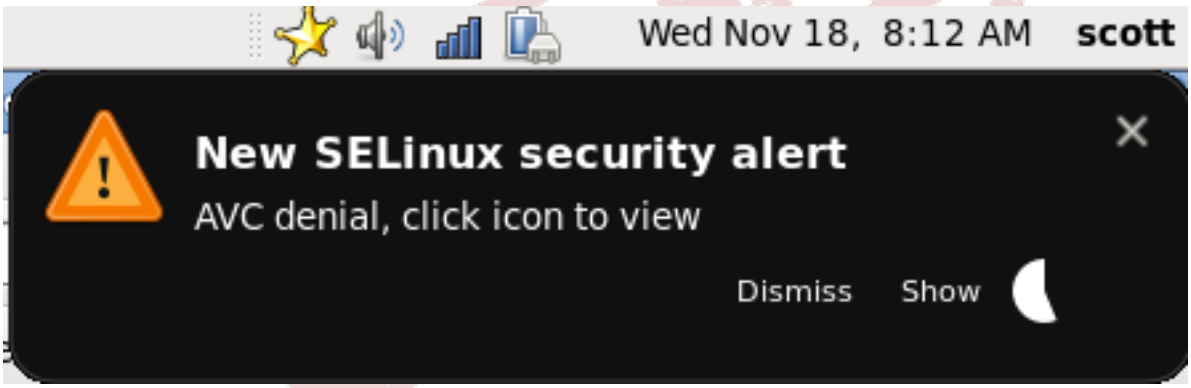
El siguiente capítulo describe qué pasa cuando SELinux niega el acceso; las principales tres causas de problemas; dónde encontrar información acerca del correcto etiquetado; análisis de las negaciones de SELinux; y creación de módulos de políticas personalizados con **`audit2allow`**.

7.1. Qué pasa cuando el Acceso es Denegado

SELinux decisions, such as allowing or disallowing access, are cached. This cache is known as the Access Vector Cache (AVC). Denial messages are logged when SELinux denies access. These denials are also known as "AVC denials", and are logged to a different location, depending on which daemons are running:

Daemon	Log Location
auditd on	**`/var/log/audit/audit.log`**
auditd off; rsyslogd on	**`/var/log/messages`**
setroubleshootd, rsyslogd, and auditd on	**`/var/log/audit/audit.log`**. Easier-to-read denial messages also sent to **`/var/log/messages`**

If you are running the X Window System, have the *setroubleshoot* and *setroubleshoot-server* packages installed, and the `setroubleshootd` and `auditd` daemons are running, a warning is displayed when access is denied by SELinux:



Clicking on 'Show' presents a detailed analysis of why SELinux denied access, and a possible solution for allowing access. If you are not running the X Window System, it is less obvious when access is denied by SELinux. For example, users browsing your website may receive an error similar to the following:

```
Forbidden

You don't have permission to access file name on this server
```

For these situations, if DAC rules (standard Linux permissions) allow access, check **`/var/log/messages`** and **`/var/log/audit/audit.log`** for **`"SELinux is preventing"`** and **`"denied"`** errors respectively. This can be done by running the following commands as the Linux root user:

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

grep "denied" /var/log/audit/audit.log
```

7.2. Tres Principales Causas de Problemas

Las siguientes secciones describen las tres principales causas de problemas: problemas de etiquetados, configuración de Booleanos y puertos para servicios, y la evolución de las reglas SELinux.

7.2.1. Problemas de Etiquetados

En sistemas que corren SELinux, todos los procesosy archivos se etiquetan con una etiqueta que contiene información de seguridad relevante. Esta información se llama contexto de SELinux. Si estas etiquetas están mal, el acceso puede ser negado. Si una aplicación se etiqueta incorrectamente, el proceso al que transiciona puede no tener la etiqueta correcta, causando negaciones de acceso de SELinux, y los procesos pueden crear archivo con las etiquetas incorrectas.

A common cause of labeling problems is when a non-standard directory is used for a service. For example, instead of using **/var/www/html/** for a website, an administrator wants to use **/srv/myweb/**. On Fedora 13, the **/srv/** directory is labeled with the **var_t** type. Files and directories created and **/srv/** inherit this type. Also, newly-created top-level directories (such as **/myserver/**) may be labeled with the **default_t** type. SELinux prevents the Apache HTTP Server (`httpd`) from accessing both of these types. To allow access, SELinux must know that the files in **/srv/myweb/** are to be accessible to `httpd`:

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

Este comando **semanage** agrega el contexto para el directorio **/srv/miweb/** (y todos los archivos dentro de él) a la configuración de contexto de archivos de SELinux [1]. El comando **semanage** no cambia el contexto. Como usuario root de Linux, ejecute el comando **restorecon** para aplicar los cambios:

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

Refer to *Sección 5.7.2, “Cambios Persistentes: semanage fcontext”* for further information about adding contexts to the file-context configuration.

7.2.1.1. ¿Cuál es el contexto correcto?

El comando **matchpathcon** chequea el contexto de un nombre completo de archivo y lo compara con la etiqueta por defecto para esa dirección. El siguiente ejemplo muestra el uso de **matchpathcon** en un directorio con archivos etiquetados incorrectamente:

```
$ /usr/sbin/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
```

Los archivos en **/etc/selinux/targeted/contexts/files/** definen los contextos de archivos y directorios. Los archivos en este directorio son leídos por **restorecon** y **setfiles** para restaurar archivos y directorios a sus contextos predeterminados.

En este ejemplo, los archivos **index.html** and **pagina1.html** se etiquetan con el tipo **user_home_t**. Este tipo se usa para archivos en los directorios de inicio de los usuarios. Usando el comando **mv** para mover archivos puede resultar en archivos etiquetados con el tipo **user_home_t**. Este tipo no debería existir fuera de los directorios home. Use el comando **restorecon** para restaurar tales archivos a su tipo correcto:

```
# /sbin/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 bajo un directorio, use la opción **-R**:

```
# /sbin/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
```

Refer to *Sección 5.10.3, “Chequeando el Contexto SELinux Predeterminado”* for a more detailed example of **matchpathcon**.

7.2.2. ¿Cómo se Ejecutan los Servicios Confinados?

Los servicios se pueden ejecutar en una variedad de formas. Para cambiar esto, debe decirle a SELinux cómo correrá los servicios. Esto se puede conseguir vía los Booleanos que permiten que parte de las políticas de SELinux se cambien en tiempo de ejecución, sin ningún conocimiento sobre la escritura de políticas de SELinux. Esto permite cambios, tales como permitir a servicios que accedan a sistemas de archivo NFS, sin recargar o recompilar una política SELinux. También, correr servicios en números de puerto no predeterminados requiere que la configuración de la política se actualice vía el comando **semanage**.

Por ejemplo, para permitir al Servidor HTTP Apache comunicarse con MySQL, active el Booleano **httpd_can_network_connect_db**:

```
# /usr/sbin/setsebool -P httpd_can_network_connect_db on
```

Si el acceso es denegado para un servicio particular, use los comandos **getsebool** y **grep** para ver si algún Booleano está disponible para permitir el acceso. Por ejemplo, use el comando **getsebool -a | grep ftp** para buscar un Booleano relacionado con FTP:

```
$ /usr/sbin/getsebool -a | grep ftp
allow_ftpd_anon_write --> off
allow_ftpd_full_access --> off
allow_ftpd_use_cifs --> off
allow_ftpd_use_nfs --> off
ftp_home_dir --> off
```

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

For a list of Booleans and whether they are on or off, run the **/usr/sbin/getsebool -a** command. For a list of Booleans, an explanation of what each one is, and whether they are on or off, run the **/usr/sbin/semanage boolean -l** command as the Linux root user. Refer to *Sección 5.6, "Booleanos"* for information about listing and configuring Booleans.

Números de Puertos

Dependiendo de la configuración de la política, los servicios pueden tener permitido correr sobre ciertos números de puerto. Intentar cambiar el puerto en el que corre un servicio sin cambiar la política puede resultar en un fallo al iniciar el servicio. Por ejemplo, ejecute el comando **semanage port -l | grep http** como usuario root de Linux para listar los puertos relacionados con `http`:

```
# /usr/sbin/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 el Servidor HTTP Apache puede escuchar, 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 fue actualizada para reflejar esto, el comando **service httpd start** falla:

```
# /sbin/service httpd start
Starting httpd: (13)Permission denied: make_sock: could not bind to address [::]:9876
(13)Permission denied: make_sock: could not bind to address 0.0.0.0:9876
no listening sockets available, shutting down
Unable to open logs
                  [FAILED]
```

Una negación de SELinux es similar a la siguiente y se guarda 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 a `httpd` escuchar en un puerto que no está listado en el tipo de puerto **http_port_t**, ejecute el comando **semanage port** para agregar un puerto a la configuración de la política [2]:

```
# /usr/sbin/semanage port -a -t http_port_t -p tcp 9876
```

El comando **semanage port -a** agrega una entrada al archivo **/etc/selinux/targeted/modules/active/ports.local**. Nota: por defecto, este archivo sólo puede ser visto por el usuario root de Linux.

La opción **-a** agrega 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 agregar.

7.2.3. Evolucionando las Reglas y las Aplicaciones Rotas

Las aplicaciones se pueden romper, provocando que SELinux niegue el acceso. También, las reglas de SELinux evolucionan - SELinux no se debe ver como una aplicación que se ejecuta en una cierta forma, haciendo que deniegue el acceso, aún cuando la aplicación está funcionado como se espera que lo haga. Por ejemplo, si una nueva versión de PostgreSQL se lanza, puede realizar acciones sobre la política actual que no han sido vistas antes, haciendo que el acceso sea denegado, aún cuando el acceso debería ser permitido.

For these situations, after access is denied, use **audit2allow** to create a custom policy module to allow access. Refer to *Sección 7.3.8, "Permitiendo el Acceso: audit2allow"* for information about using **audit2allow**.

7.3. Corrección de Problemas

Las siguientes secciones ayudan a resolver problemas. Cubren los temas: chequeo de los permisos de Linux, que se chequean antes que las reglas de SELinux; posibles causas de negaciones de acceso de SELinux, pero no negaciones que se estén guardando; páginas man de los servicios, que contienen información sobre etiquetado y Booleanos; dominios permisivos, para permitir a un proceso correr en modo permisivo, en vez de todo el sistema; cómo buscar y encontrar mensajes; análisis de negaciones; y creación de módulos de políticas personalizados con **audit2allow**.

7.3.1. Permisos de Linux

When access is denied, check standard Linux permissions. As mentioned in *Capítulo 2, Introducción*, most operating systems use a Discretionary Access Control (DAC) system to control access, allowing users to control the permissions of files that they own. SELinux policy rules are checked after DAC rules. SELinux policy rules are not used if DAC rules deny access first.

Si el acceso es denegado y no hay negaciones SELinux guardadas, use el comando **ls -l** para ver los permisos estándares de Linux:

```
$ ls -l /var/www/html/index.html
-rw-r----- 1 root root 0 2009-05-07 11:06 index.html
```

En este ejemplo, **index.html** pertenece al usuario y al grupo root. El usuario root tiene permisos de lectura y escritura (**-rw**), y los miembros del grupo root tienen permisos de lectura (**-r-**). Cualquier otro no tiene acceso (**---**). Por defecto, tales permisos no permiten a `httpd` leer este archivo. Para resolver esto, use el comando **chown** el dueño y el grupo. Este comando se debe ejecutar como usuario root de Linux:

```
# chown apache:apache /var/www/html/index.html
```

Esto asume la configuración predeterminada, en la que `httpd` corre como usuario apache de Linux. Si corre `httpd` con un usuario diferente, reemplace **apache:apache** con ese usuario.

Refer to the *Fedora Documentation Project "Permissions"*[3] draft for information about managing Linux permissions.

7.3.2. Posibles Causas de las Negaciones Silenciosas

En ciertas situaciones, las negaciones AVC pueden no ser guardadas cuando SELinux niega el acceso. Las aplicaciones y las funciones de las bibliotecas del sistema a menudo prueban más accesos que los pedidos para realizar sus tareas. Para mantener el menor privilegio sin llenar los informes de auditoría con negaciones AVC para pruebas sin peligro de las aplicaciones, la política puede silenciar las negaciones AVC sin permitir el uso de reglas **dontaudit**. Estas reglas son comúnes en la política estándar. La contraparte de **dontaudit** es que, aunque SELinux niega el acceso, los mensajes no se guardan, lo que dificulta resolver el problema.

Deshabilite temporalmente las reglas **dontaudit**, permitiendo que se guarden todas las negaciones, ejecute el siguiente comando como usuario root de Linux:

/usr/sbin/semodule -DB

La opción **-D** deshabilita las reglas **dontaudit**; la opción **-B** reconstruye la política. Después de ejcutar **semodule -DB**, pruebe ejercitar la aplicación que tuvo problemas de permisos, y vea si ahora se guardan negaciones de SELinux relacionadas con la aplicación. Tenga cuidado con la decición de qué negaciones se deben permitir, dado que algunas se deben ignorar y manejarse vía reglas **dontaudit**. Si tiene duda, o busca alguna guía, contacte a otros usuarios y desarrolladores de SELinux en una lista de SELinux, tal como *fedora-selinux-list*[4].

Para reconstruir la política y habilitar las reglas **dontaudit**, ejecute el siguiente comando como usuario root de Linux:

/usr/sbin/semodule -B

Esto restaura la política a su estado original. Para una lista completa de las reglas **dontaudit**, corra el comando **sesearch --dontaudit**. Búsquedas más refinadas usando la opción **-s *dominio*** y el comando **grep**. Por ejemplo:

```
$ sesearch --dontaudit -s smbd_t | grep squid
WARNING: This policy contained disabled aliases; they have been removed.
dontaudit smbd_t squid_port_t : tcp_socket name_bind ;
dontaudit smbd_t squid_port_t : udp_socket name_bind ;
```

Refer to *Sección 7.3.6, “Raw Audit Messages”* and *Sección 7.3.7, “Mensajes sealert”* for information about analyzing denials.

7.3.3. Páginas de Manual para Servicios

Las páginas de manual para los servicios conteniendo información valiosa, tal como qué tipo de archivo usar para una situación dada, y los Booleanos para cambiar el acceso que un servicio tiene (tal como `httpd` para acceder sistemas de archivos NFS). Esta información puede estar en la página de manual estándar o una página de manual con **selinux** como prefijo o sufijo.

[3] http://fedoraproject.org/wiki/Docs/Drafts/AdministrationGuide/Permissions
[4] http://www.redhat.com/mailman/listinfo/fedora-selinux-list

For example, the httpd_selinux(8) manual page has information about what file type to use for a given situation, as well as Booleans to allow scripts, sharing files, accessing directories inside user home directories, and so on. Other manual pages with SELinux information for services include:

- Samba: the samba_selinux(8) manual page describes that files and directories to be exported via Samba must be labeled with the **samba_share_t** type, as well as Booleans to allow files labeled with types other than **samba_share_t** to be exported via Samba.
- NFS: the nfs_selinux(8) manual page describes that, by default, file systems can not be exported via NFS, and that to allow file systems to be exported, Booleans such as **nfs_export_all_ro** or **nfs_export_all_rw** must be turned on.
- Berkeley Internet Name Domain (BIND): the named(8) manual page describes what file type to use for a given situation (see the **Red Hat SELinux BIND Security Profile** section). The named_selinux(8) manual page describes that, by default, named can not write to master zone files, and to allow such access, the **named_write_master_zones** Boolean must be turned on.

La información en las páginas del manual le ayudan a configurar los tipos de archivos correctos y los Booleanos, ayudándolo a prevenir las negaciones de acceso por parte de SELinux.

7.3.4. Dominios Permisivos

Cuando SELinux se ejecuta en modo permisivo, SELinux no niega el acceso, sino que las negaciones para las acciones se guardan como si fuera que corre en modo obediente. Previamente, no era posible hacer permisivo un único dominio (recuerde: los procesos corren en dominios). En ciertas situaciones, esto llevó a hacer el sistema permisivo para poder corregir los problemas.

Fedora 13 includes permissive domains, where an administrator can configure a single process (domain) to run permissive, rather than making the whole system permissive. SELinux checks are still performed for permissive domains; however, the kernel allows access and reports an AVC denial for situations where SELinux would have denied access. Permissive domains are also available in Fedora 9 (with the latest updates applied).

En el Linux para Empresas de Red Hat 4 y 5, los Booleanos ***dominio*_disable_trans** están disponibles para prevenir que una aplicación transicione a un dominio confinado, y por lo tanto, el proceso se ejecute en un dominio no confinado, tal como **initrc_t**. Poniendo en 1 tales booleanos pueden causar problemas serios. Por ejemplo, si el Booleano **httpd_disable_trans** se pone en 1:

- httpd corre en el dominio no confinado **initrc_t**. Los archivos creados por los procesos en el dominio **initrc_t** puede no tener aplicadas las mismas reglas de etiquetados como los archivos creados por el proceso corriendo en el dominio **httpd_t**, permitiendo que los procesos puedan potencialmente crear archivos mal etiquetados. Esto causa problemas más adelante.
- dominios confinados que pueden comunicarse con **httpd_t** no pueden comunicarse con **initrc_t**, posiblemente causan fallas adicionales.

Los Booleanos ***domain*_disable_trans** fueron eliminados de Fedora 7, y no se pusieron reemplazos. Los dominios permisivos pueden resolver esos problemas: se aplican las reglas de transición y los archivos se crean con las etiquetas correctas.

Los dominios permisivos se pueden usar para:

- hacer que un único proceso (dominio) corra permisivo para solucionar alguna cuestión, en vez de poner todo el sistema en riesgo haciendo permisivo a todo el sistema.

- creación de políticas para nuevas aplicaciones. Previamente, era recomendado crear una política mínima, y luego poner la máquina completa en modo permisivo, para que la aplicación pudiera funcionar, pero las negaciones de SELinux eran igualmente grabadas. **audit2allow** podría usarse luego para ayudar a escribir la política. Esto pone todo el sistema en riesgo. Con dominios permisivos, sólo el dominio en la nueva política puede marcarse como permisivo, sin poner en riesgo todo el sistema.

7.3.4.1. Creando un Dominio Permisivo

Para hacer un dominio permisivo, ejecute el comando **semanage permissive -a *dominio***, donde *dominio* es el dominio que quiere hacer permisivo. Por ejemplo, ejecute el siguiente comando como usuario root de Linux para hacer permisivo el dominio **httpd_t** (el dominio en el que corre el Servidor HTTP Apache):

/usr/sbin/semanage permissive -a httpd_t

Para ver una lista de los dominios que hizo permisivos, corra el comando **semodule -l | grep permissive** como usuario root de Linux. Por ejemplo:

```
# /usr/sbin/semodule -l | grep permissive
permissive_httpd_t      1.0
```

Si ya no quiere que un dominio sea permisivo, corra el comando **semanage permissive -d *dominio*** como usuario root de Linux. Por ejemplo:

/usr/sbin/semanage permissive -d httpd_t

7.3.4.2. Negaciones para Dominios Permisivos

El mensaje **SYSCALL** es diferente para dominios permisivos. El siguiente es un ejemplo de una negación de AVC (y la llamada a sistema asociada) desde el Servidor HTTP Apache:

```
type=AVC msg=audit(1226882736.442:86): avc:  denied  { getattr } for  pid=2427 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

type=SYSCALL msg=audit(1226882736.442:86): arch=40000003 syscall=196 success=no exit=-13
 a0=b9a1e198 a1=bfc2921c a2=54dff4 a3=2008171 items=0 ppid=2425 pid=2427 auid=502 uid=48
 gid=48 euid=48 suid=48 fsuid=48 egid=48 sgid=48 fsgid=48 tty=(none) ses=4 comm="httpd" exe="/
usr/sbin/httpd" subj=unconfined_u:system_r:httpd_t:s0 key=(null)
```

Por defecto, el dominio **httpd_t** es no permisivo, y como tal, la acción es negada, y el mensaje **SYSCALL** contiene **success=no**. El siguiente es un ejemplo de negación AVC para la misma situación, excepto que el comando **semanage permissive -a httpd_t** se ejecutó para hacer el dominio **httpd_t** permisivo:

```
type=AVC msg=audit(1226882925.714:136): avc:  denied  { read } for  pid=2512
 comm="httpd" name="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

type=SYSCALL msg=audit(1226882925.714:136): arch=40000003 syscall=5 success=yes exit=11
 a0=b962a1e8 a1=8000 a2=0 a3=8000 items=0 ppid=2511 pid=2512 auid=502 uid=48 gid=48 euid=48
```

```
suid=48 fsuid=48 egid=48 sgid=48 fsgid=48 tty=(none) ses=4 comm="httpd" exe="/usr/sbin/httpd"
subj=unconfined_u:system_r:httpd_t:s0 key=(null)
```

En este caso, aunque la negación AVC fue grabada, el acceso no fue negado, como se muestra en el mensaje **SYSCALL success=yes**.

Refer to Dan Walsh's *"Permissive Domains"*[5] blog entry for further information about permissive domains.

7.3.5. Búsqueda y Revisión de Negaciones

This section assumes the *setroubleshoot*, *setroubleshoot-server*, *dbus* and *audit* packages are installed, and that the `auditd`, `rsyslogd`, and `setroubleshootd` daemons are running. Refer to *Sección 5.2, "Qué Archivo Log se usa"* for information about starting these daemons. A number of tools are available for searching for and viewing SELinux denials, such as **ausearch**, **aureport**, and **sealert**.

ausearch

The *audit* package provides **ausearch**. From the ausearch(8) manual page: "**ausearch** is a tool that can query the audit daemon logs based for events based on different search criteria"[6]. The **ausearch** tool accesses **/var/log/audit/audit.log**, and as such, must be run as the Linux root user:

Buscando	Comando
todas las negaciones	**/sbin/ausearch -m avc**
negaciones de hoy	**/sbin/ausearch -m avc -ts today**
negaciones desde los últimos 10 minutos	**/sbin/ausearch -m avc -ts recent**

To search for SELinux denials for a particular service, use the **-c *comm-name*** option, where *comm-name* "is the executable's name"[7], for example, `httpd` for the Apache HTTP Server, and `smbd` for Samba:

```
/sbin/ausearch -m avc -c httpd
```

```
/sbin/ausearch -m avc -c smbd
```

Refer to the ausearch(8) manual page for further **ausearch** options.

aureport

The *audit* package provides **aureport**. From the aureport(8) manual page: "**aureport** is a tool that produces summary reports of the audit system logs"[8]. The **aureport** tool accesses **/var/log/audit/audit.log**, and as such, must be run as the Linux root user. To view a list of SELinux denials and how often each one occurred, run the **aureport -a** command. The following is example output that includes two denials:

```
# /sbin/aureport -a
```

[5] http://danwalsh.livejournal.com/24537.html
From the ausearch(8) manual page, as shipped with the *audit* package in Fedora 13.
From the ausearch(8) manual page, as shipped with the *audit* package in Fedora 13.
From the aureport(8) manual page, as shipped with the *audit* package in Fedora 13.

```
AVC Report
==============================================================
# date time comm subj syscall class permission obj event
==============================================================
1. 05/01/2009 21:41:39 httpd unconfined_u:system_r:httpd_t:s0 195 file getattr
 system_u:object_r:samba_share_t:s0 denied 2
2. 05/03/2009 22:00:25 vsftpd unconfined_u:system_r:ftpd_t:s0 5 file read
 unconfined_u:object_r:cifs_t:s0 denied 4
```

Refer to the aureport(8) manual page for further **`aureport`** options.

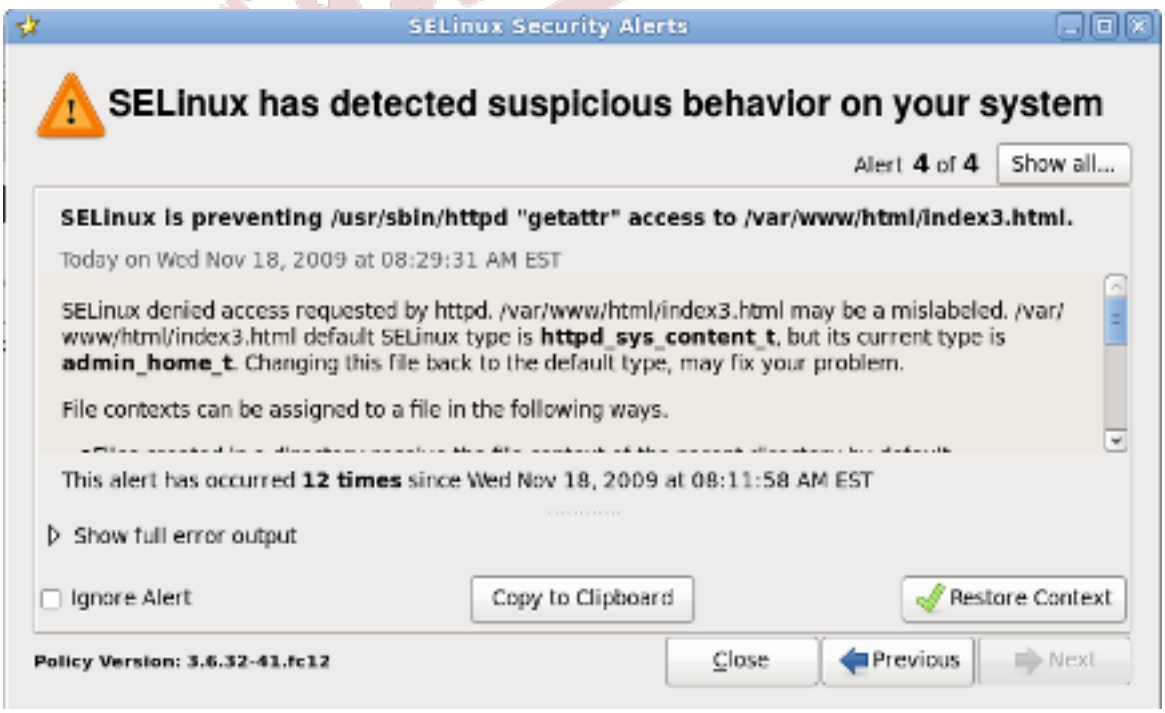
sealert

El paquete *setroubleshoot-server* provee **`sealert`**, que lee los mensajes de negación traducidos por *setroubleshoot-server*. A las negaciones se le asignan IDs, como se ve en **`/var/log/messages`**. El siguiente es un ejemplo de negación en **`messages`**:

```
setroubleshoot: SELinux is preventing httpd (httpd_t) "getattr" to /var/www/html/
file1 (samba_share_t). For complete SELinux messages. run sealert -l 84e0b04d-
d0ad-4347-8317-22e74f6cd020
```

En este ejemplo, el ID de negación es **`84e0b04d-d0ad-4347-8317-22e74f6cd020`**. La opción **`-l`** toma un ID como argumento. Ejecutando el comando **`sealert -l 84e0b04d-d0ad-4347-8317-22e74f6cd020`** le presenta un análisis detallado de por qué SELinux negó el acceso, y una posible solución para permitir el acceso.

If you are running the X Window System, have the *setroubleshoot* and *setroubleshoot-server* packages installed, and the `setroubleshootd`, `dbus` and `auditd` daemons are running, a warning is displayed when access is denied by SELinux. Clicking on 'Show' launches the **`sealert`** GUI, and displays denials in HTML output:



- Ejecute el comando **sealert -b** para lanzar la GUI de **sealert**.
- Ejecute el comando **sealert -l *** para ver un análisis detallado de todas las negaciones.
- As the Linux root user, run the **sealert -a /var/log/audit/audit.log -H > audit.html** command to create a HTML version of the **sealert** analysis, as seen with the **sealert** GUI.

Refer to the sealert(8) manual page for further **sealert** options.

7.3.6. Raw Audit Messages

Los mensajes crudos de auditoría se guardan en **/var/log/audit/audit.log**. El siguiente es un ejemplo de negación AVC (y su llamada a sistema asociado) que ocurrío cuando el Servidor HTTP Apache (corriendo en el dominio **httpd_t**) intentó acceder el **/var/www/html/archivo1** (etiquetado con el tipo **samba_share_t**):

```
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

type=SYSCALL msg=audit(1226874073.147:96): arch=40000003 syscall=196 success=no exit=-13
 a0=b98df198 a1=bfec85dc a2=54dff4 a3=2008171 items=0 ppid=2463 pid=2465 auid=502 uid=48
 gid=48 euid=48 suid=48 fsuid=48 egid=48 sgid=48 fsgid=48 tty=(none) ses=6 comm="httpd" exe="/
usr/sbin/httpd" subj=unconfined_u:system_r:httpd_t:s0 key=(null)
```

{ getattr }
: The item in braces indicates the permission that was denied. **getattr** indicates the source process was trying to read the target file's status information. This occurs before reading files. This action is denied due to the file being accessed having the wrong label. Commonly seen permissions include **getattr**, **read**, and **write**.

comm="*httpd*"
: The executable that launched the process. The full path of the executable is found in the **exe=** section of the system call (**SYSCALL**) message, which in this case, is **exe="/usr/sbin/httpd"**.

path="*/var/www/html/file1*"
: La dirección al objeto (destino) al que quiere acceder el proceso.

scontext="*unconfined_u:system_r:httpd_t:s0*"
: El contexto de SELinux del proceso que intentó la acción denegada. En este caso, es el contexto SELinux del Servidor HTTP Apache, que core en el dominio **httpd_t**.

tcontext="*unconfined_u:object_r:samba_share_t:s0*"
: El contexto de SELinux del objeto (destino) al que intentó acceder el proceso. En este caso, es el contexto SELinux del **archivo1**. Nota: el tipo **samba_share_t** no es accesible para procesos que corren el dominio **httpd_t**.

  En ciertas cituaciónes, el **tcontext** puede coincidir con **scontext**, por ejemplo, cuando un proceso intenta ejecutar un servicio del sistema que cambiará las características de ese proceso en ejecución, tales como el ID del usuario. También el **tcontext** puede coincidir con el **scontext** cuando un proceso intenta usar más recursos (como la memoria) más allá de los límites normales permitidos, lo que resulta en un chequeo de seguridad para ver si el proceso tiene permitido romper esos límites.

Desde el mensaje de llamado al sistema (**SYSCALL**) nos interesan dos ítems:

- **success=*no***: indica si la negación (AVC) fue aplicada o no. **success=no** indica que la llamada al sistema no fue exitosa(SELinux negó el acceso). **success=yes** indica que la llamada al sistema fue exitosa - esto se puede ver en dominios permisivos o en dominios no confinados, tales como **initrc_t** y **kernel_t**.
- **exe="*/usr/sbin/httpd*"**: the full path to the executable that launched the process, which in this case, is **exe="/usr/sbin/httpd"**.

An incorrect file type is a common cause for SELinux denying access. To start troubleshooting, compare the source context (**scontext**) with the target context (**tcontext**). Should the process (**scontext**) be accessing such an object (**tcontext**)? For example, the Apache HTTP Server (**httpd_t**) should only be accessing types specified in the httpd_selinux(8) manual page, such as **httpd_sys_content_t**, **public_content_t**, and so on, unless configured otherwise.

7.3.7. Mensajes sealert

Las negaciones tienen IDs asignados, como se ve en **/var/log/messages**. El siguiente es un ejemplo de negación AVC (guardado en **messages**) que ocurrió cuando el Servidor HTTP Apache (corriendo en el dominio **httpd_t** domain) intentó acceder el **/var/www/html/archivo1** (etiquetado con el tipo **samba_share_t**):

```
hostname setroubleshoot: SELinux is preventing httpd (httpd_t) "getattr" to /var/www/
html/file1 (samba_share_t). For complete SELinux messages. run sealert -l 84e0b04d-
d0ad-4347-8317-22e74f6cd020
```

Como se sugirió, ejecute el comando **sealert -l 84e0b04d-d0ad-4347-8317-22e74f6cd020** para ver el mensaje completo. Este comando sólo funciona en la máquina local, y presenta la misma información que la interfase gráfica de **sealert**:

```
$ sealert -l 84e0b04d-d0ad-4347-8317-22e74f6cd020

Summary:

SELinux is preventing httpd (httpd_t) "getattr" to /var/www/html/file1
(samba_share_t).

Detailed Description:

SELinux denied access to /var/www/html/file1 requested by httpd.
/var/www/html/file1 has a context used for sharing by different program. If you
would like to share /var/www/html/file1 from httpd also, you need to change its
file context to public_content_t. If you did not intend to this access, this
could signal a intrusion attempt.

Allowing Access:

You can alter the file context by executing chcon -t public_content_t
'/var/www/html/file1'

Fix Command:

chcon -t public_content_t '/var/www/html/file1'
```

```
Additional Information:

Source Context                unconfined_u:system_r:httpd_t:s0
Target Context                unconfined_u:object_r:samba_share_t:s0
Target Objects                /var/www/html/file1 [ file ]
Source                        httpd
Source Path                   /usr/sbin/httpd
Port                          <Unknown>
Host                          hostname
Source RPM Packages           httpd-2.2.10-2
Target RPM Packages
Policy RPM                    selinux-policy-3.5.13-11.fc12
Selinux Enabled               True
Policy Type                   targeted
MLS Enabled                   True
Enforcing Mode                Enforcing
Plugin Name                   public_content
Host Name                     hostname
Platform                      Linux hostname 2.6.27.4-68.fc12.i686 #1 SMP Thu Oct
30 00:49:42 EDT 2008 i686 i686
Alert Count                   4
First Seen                    Wed Nov  5 18:53:05 2008
Last Seen                     Wed Nov  5 01:22:58 2008
Local ID                      84e0b04d-d0ad-4347-8317-22e74f6cd020
Line Numbers

Raw Audit Messages

node=hostname type=AVC msg=audit(1225812178.788:101): avc:  denied  { getattr }
 for  pid=2441 comm="httpd" path="/var/www/html/file1" dev=dm-0 ino=284916
 scontext=unconfined_u:system_r:httpd_t:s0 tcontext=unconfined_u:object_r:samba_share_t:s0
 tclass=file

node=hostname type=SYSCALL msg=audit(1225812178.788:101): arch=40000003 syscall=196 success=no
 exit=-13 a0=b8e97188 a1=bf87aaac a2=54dff4 a3=2008171 items=0 ppid=2439 pid=2441 auid=502
 uid=48 gid=48 euid=48 suid=48 fsuid=48 egid=48 sgid=48 fsgid=48 tty=(none) ses=3 comm="httpd"
 exe="/usr/sbin/httpd" subj=unconfined_u:system_r:httpd_t:s0 key=(null)
```

Resumen

Un breve resumen de la acción negada. Esto es lo mismo que la negación en **/var/log/messages**. En este ejemplo, al proceso `httpd` se le negó el acceso al (**archivo1**), que está etiquetado con el tipo **samba_share_t**.

Descripción Detallada

Una descripción más detallada. En este ejemplo **archivo1** está etiquetado con el tipo **samba_share_t**. Este tipo se usa para archivos y directorios que quiere exportar vía Samba. La descripción sugiere cambiar el tipo a un tipo que pueda ser accedido por Samba y por el Servidor HTTP Apache, si tal acceso es deseado.

Permitiendo Acceso

Una sugerencia sobre cómo permitir el acceso. Esto puede hacerse reetiquetando archivos, poniendo en 1 un Booleano, o creando un módulo de política local. En este caso, la sugerencia es etiquetar el archivo con un tipo accesible por el Servidor HTTP Apache y por Samba.

Comando para Corregir

Un comando sugerido para permitir el acceso y resolver la negación. En este ejemplo, se dá el comando para cambiar el tipo del **archivo1** a **public_content_t**, que es accesible por el Servidor HTTP Apache y por Samba.

Información Adicional
: Information that is useful in bug reports, such as the policy package name and version (**selinux-policy-3.5.13-11.fc12**), but may not help towards solving why the denial occurred.

Raw Audit Messages
: The raw audit messages from **/var/log/audit/audit.log** that are associated with the denial. Refer to *Sección 7.3.6, "Raw Audit Messages"* for information about each item in the AVC denial.

7.3.8. Permitiendo el Acceso: audit2allow

No use el ejemplo en esta sección en producción. Se usa sólo para mostrar el uso de **audit2allow**.

From the audit2allow(1) manual page: "**audit2allow** - generate SELinux policy allow rules from logs of denied operations"[9]. After analyzing denials as per *Sección 7.3.7, "Mensajes sealert"*, and if no label changes or Booleans allowed access, use **audit2allow** to create a local policy module. After access is denied by SELinux, running the **audit2allow** command presents Type Enforcement rules that allow the previously denied access.

El siguiente ejemplo muestra el uso de **audit2allow** para crear un módulo de política:

1. Una negación y la llamada al sistema asociado se graban en **/var/log/audit/audit.log**:

```
type=AVC msg=audit(1226270358.848:238): avc:  denied  { write }
 for  pid=13349 comm="certwatch" name="cache" dev=dm-0 ino=218171
 scontext=system_u:system_r:certwatch_t:s0 tcontext=system_u:object_r:var_t:s0 tclass=dir

type=SYSCALL msg=audit(1226270358.848:238): arch=40000003 syscall=39 success=no exit=-13
 a0=39a2bf a1=3ff a2=3a0354 a3=94703c8 items=0 ppid=13344 pid=13349 auid=4294967295
 uid=0 gid=0 euid=0 suid=0 fsuid=0 egid=0 sgid=0 fsgid=0 tty=(none) ses=4294967295
 comm="certwatch" exe="/usr/bin/certwatch" subj=system_u:system_r:certwatch_t:s0
 key=(null)
```

   In this example, **certwatch** (**comm="certwatch"**) was denied write access (**{ write }**) to a directory labeled with the **var_t** type (**tcontext=system_u:object_r:var_t:s0**). Analyze the denial as per *Sección 7.3.7, "Mensajes sealert"*. If no label changes or Booleans allowed access, use **audit2allow** to create a local policy module.

2. Con una negación grabada, tal como la negación de **certwatch** en el paso 1, corra el comando **audit2allow -w -a** para producir una descripción legible al humano sobre por qué el acceso fue negado. La opción **-a** hace que se lean todos los registros de autidoría. La opción **-w** produce una descripción legible al humano. La herramienta **audit2allow** accede a **/var/log/audit/audit.log**, y como tal, debe ser ejecutada como usuario root de Linux:

```
# audit2allow -w -a
type=AVC msg=audit(1226270358.848:238): avc:  denied  { write }
 for  pid=13349 comm="certwatch" name="cache" dev=dm-0 ino=218171
 scontext=system_u:system_r:certwatch_t:s0 tcontext=system_u:object_r:var_t:s0 tclass=dir
 Was caused by:
  Missing type enforcement (TE) allow rule.

 You can use audit2allow to generate a loadable module to allow this access.
```

[9] From the audit2allow(1) manual page, as shipped with the *policycoreutils* package in Fedora 13.

```
```

Como se muestra, el acceso fue negado debido a que falta una regla de Obligación de Tipo.

3. Ejecute el comando **audit2allow -a** para ver la regla de Obligación de Tipo que permite el acceso negado:

```
# audit2allow -a

#============= certwatch_t ==============
allow certwatch_t var_t:dir write;
```

Importante

La falta de reglas de Ejecución de Tipos son usualmente causados por errores en la política de SELinux, y deben ser informadas en el *Bugzilla de Red Hat*[10]. Para Fedora, crear informes sobre el producto **Fedora**, y seleccione el componente **selinux-policy**. Incluya la salida de los comandos **audit2allow -w -a** y **audit2allow -a** en el informe del error.

4. Para usar la regla mostrada por **audit2allow -a**, ejecute el comando **audit2allow -a -M *mycertwatch*** como usuario root de Linux. La opción **-M** crea un archivo de Obligación de Tipo (**.te**) con el nombre especificado en **-M**, en su directorio actual de trabajo:

```
# audit2allow -a -M mycertwatch

******************** IMPORTANT ***********************
To make this policy package active, execute:

semodule -i mycertwatch.pp

# ls
mycertwatch.pp  mycertwatch.te
```

También, **audit2allow** compila la regla de Obediencia de Tipo en un paquete de política (**.pp**). Para instalar el módulo, ejecute el comando **/usr/sbin/semodule -i *mycertwatch.pp*** como usuario root de Linux.

Importante

Los módulos creados con **audit2allow** pueden permitir más acceso que el requerido. Se recomienda que la política creada con **audit2allow** sea enviada a una lista de SELinux, tal como *fedora-selinux-list*[11], para su revisión. Si cree que hay un error en la política, informe un error en *Bugzilla de Red Hat*[12].

Si tiene múltiples negaciones de múltiples procesos, pero solo quiere crear una política personalizada para un proceso único, use el comando **grep** para una búsqueda más refinada de **audit2allow**.

El siguiente ejemplo muestra el uso de **`grep`** para sólo enviar negaciones de **`certwatch`** a **`audit2allow`**:

```
# grep certwatch /var/log/audit/audit.log | audit2allow -M mycertwatch2
******************* IMPORTANT ***********************
To make this policy package active, execute:

# /usr/sbin/semodule -i mycertwatch2.pp
```

Refer to Dan Walsh's *"Using audit2allow to build policy modules. Revisited."*[13] blog entry for further information about using **`audit2allow`** to build policy modules.

[13] http://danwalsh.livejournal.com/24750.html

Información Adicional

8.1. Contributors

- *Geert Warrink*[1] (translation - Dutch)
- *Domingo Becker*[2] (translation - Spanish)
- *Daniel Cabrera*[3] (translation - Spanish)

8.2. Other Resources

La Agencia de Seguridad Nacional (NSA)

De la página *Contribuyentes de SELinux*[4]:

Researchers in NSA's National Information Assurance Research Laboratory (NIARL) designed and implemented flexible mandatory access controls in the major subsystems of the Linux kernel and implemented the new operating system components provided by the Flask architecture, namely the security server and the access vector cache. The NSA researchers reworked the LSM-based SELinux for inclusion in Linux 2.6. NSA has also led the development of similar controls for the X Window System (XACE/XSELinux) and for Xen (XSM/Flask).

- Main SELinux website: *http://www.nsa.gov/research/selinux/index.shtml*.
- SELinux documentation: *http://www.nsa.gov/research/selinux/docs.shtml*.
- SELinux background: *http://www.nsa.gov/research/selinux/background.shtml*.

Tecnología de Tresys

Tresys Technology[5] son los desarrolladores de:

- *Herramientas y bibliotecas en el espacio del usuario para SELinux*[6].
- *Política de Referencia de SELinux*[7].

Noticias de SELinux

- News: *http://selinuxnews.org/wp/*.
- Planet SELinux (blogs): *http://selinuxnews.org/planet/*.

Wiki del Proyecto SELinux

- Main page: *http://selinuxproject.org/page/Main_Page*.
- User resources, including links to documentation, mailing lists, websites, and tools: *http://selinuxproject.org/page/User_Resources*.

[4] http://www.nsa.gov/research/selinux/contrib.shtml
[5] http://www.tresys.com/

Linux para Empresas de Red Hat

- La *Guía de Despliegue del Linux para Empresas de Red Hat*[8] contiene una sección de *Referencias*[9] SELinux, que tiene enlaces a tutoriales de SELinux, información general y la tecnología detrás de SELinux.
- La *Guía de SELinux del Linux para Empresas de Red Hat 4*[10].

Fedora

- Main page: *http://fedoraproject.org/wiki/SELinux*.
- Troubleshooting: *http://fedoraproject.org/wiki/SELinux/Troubleshooting*.
- Fedora Core 5 SELinux FAQ: *http://docs.fedoraproject.org/selinux-faq-fc5/*.
- SELinux Managing Confined Services Guide: *http://docs.fedoraproject.org/selinux-managing-confined-services-guide/*

Las Preguntas Frecuentes No Oficiales de SELinux

http://www.crypt.gen.nz/selinux/faq.html

IRC

En *Freenode*[11]:

- #selinux
- #fedora-selinux
- #security

[11] http://freenode.net/

Apéndice A. Revision History

Revisión 1.4 Mon Aug 31 2009 **Scott Radvan *sradvan@redhat.com***

Update and verification for Fedora 12

Revisión 1.3 Tue May 12 2009 **Scott Radvan *sradvan@redhat.com***

Update and verification for Fedora 11

Revisión 1.2 Mon Jan 19 2009 **Murray McAllister *mmcallis@redhat.com***

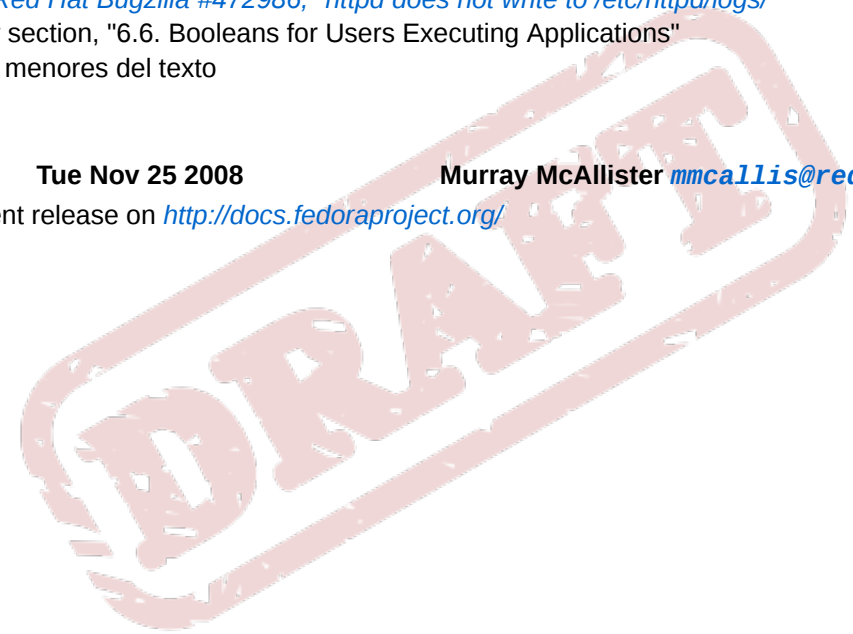
Actualización de los enlaces a sitios web de la NSA

Revisión 1.1 Sat Dec 6 2008 **Murray McAllister *mmcallis@redhat.com***

Resolving *Red Hat Bugzilla #472986, "httpd does not write to /etc/httpd/logs/"*[1]
Added new section, "6.6. Booleans for Users Executing Applications"
Revisiones menores del texto

Revisión 1.0 Tue Nov 25 2008 **Murray McAllister *mmcallis@redhat.com***

Initial content release on *http://docs.fedoraproject.org/*




DRAFT