RHCSA 系列（十三）：在 RHEL 7 中使用 SELinux 进行强制访问控制

2015-10-3 09:07 收藏: 3

原文：http://www.tecmint.com/selinux-essentials-and-control-filesystem-access/
译文：LCTT https://linux.cn/article-6339-1.html

作者： Gabriel Cánepa
译者： FSSlc

在本系列的前面几篇文章中，我们已经详细地探索了至少两种访问控制方法：标准的 ugo/rwx 权限（**RHCSA 系列（三）：如何管理 RHEL7 的用户和组 <https://linux.cn/article-6187-1.html>** ）和访问控制列表（**RHCSA 系列（七）：使用 ACL（访问控制列表）和挂载 Samba/NFS 共享 <https://linux.cn/article-6263-1.html>** ）。

RHCSA 认证：SELinux 精要和控制文件系统的访问

尽管作为第一级别的权限和访问控制机制是必要的，但它们同样有一些局限，而这些局限则可以由 安全增强 Linux（Security Enhanced Linux，简称为 SELinux） 来处理。

这些局限的一种情形是：某个用户可能通过一个泛泛的 chmod 命令将文件或目录暴露出现了安全违例，从而引起访问权限的意外传播。结果，由该用户开启的任意进程可以对属于该用户的文件进行任意的操作，最终一个恶意的或有其它缺陷的软件可能会取得整个系统的 root 级别的访问权限。

考虑到这些局限性，美国国家安全局（NSA） 率先设计出了 SELinux，一种强制的访问控制方法，它根据最小权限模型去限制进程在系统对象（如文件，目录，网络接口等）上的访问或执行其他的操作的能力，而这些限制可以在之后根据需要进行修改。简单来说，系统的每一个元素只给某个功能所需要的那些权限。

在 RHEL 7 中，SELinux 被并入了内核中，且默认情况下以 强制模式（Enforcing） 开启。在这篇文章中，我们将简要地介绍有关 SELinux 及其相关操作的基本概念。

SELinux 的模式

SELinux 可以以三种不同的模式运行：

- 强制模式（Enforcing）：SELinux 基于其策略规则来拒绝访问，这些规则是用以控制安全引擎的一系列准则；
- 宽容模式（Permissive）：SELinux 不会拒绝访问，但对于那些如果运行在强制模式下会被拒绝访问的行为进行记录；
- 关闭（Disabled） (不言自明，即 SELinux 没有实际运行).

使用 `getenforce` 命令可以展示 SELinux 当前所处的模式，而 `setenforce` 命令（后面跟上一个 1 或 0） 则被用来将当前模式切换到 强制模式（Enforcing） 或 宽容模式（Permissive），但只对当前的会话有效。

为了使得在登出和重启后上面的设置还能保持作用，你需要编辑 `/etc/selinux/config` 文件并将 SELINUX 变量的值设为 enforcing，permissive，disabled 中之一：

```
# getenforce
# setenforce 0
```

```
# getenforce
# setenforce 1
# getenforce
# cat /etc/selinux/config
```

设置 SELinux 模式

通常情况下，你应该使用 `setenforce` 来在 SELinux 模式间进行切换（从强制模式到宽容模式，或反之），以此来作为你排错的第一步。假如 SELinux 当前被设置为强制模式，而你遇到了某些问题，但当你把 SELinux 切换为宽容模式后问题不再出现了，则你可以确信你遇到了一个 SELinux 权限方面的问题。

SELinux 上下文

一个 SELinux 上下文 (Context) 由一个访问控制环境所组成，在这个环境中，决定的做出将基于 SELinux 的用户，角色和类型（和可选的级别）：

- 一个 SELinux 用户是通过将一个常规的 Linux 用户账户映射到一个 SELinux 用户账户来实现的，反过来，在一个会话中，这个 SELinux 用户账户在 SELinux 上下文中被进程所使用，以便能够明确定义它们所允许的角色和级别。
- 角色的概念是作为域和处于该域中的 SELinux 用户之间的媒介，它定义了 SELinux 可以访问到哪个进程域和哪些文件类型。这将保护您的系统免受提权漏洞的攻击。
- 类型则定义了一个 SELinux 文件类型或一个 SELinux 进程域。在正常情况下，进程将会被禁止访问其他进程正使用的文件，并禁止对其他进程进行访问。这样只有当一个特定的 SELinux 策略规则允许它访问时，才能够进行访问。

下面就让我们看看这些概念是如何在下面的例子中起作用的。

例 1：改变 sshd 守护进程的默认端口

在 **RHCSA 系列（八）：加固 SSH，设定主机名及启用网络服务 <https://linux.cn/article-6266-1.html>** 中，我们解释了更改 sshd 所监听的默认端口是加固你的服务器免受外部攻击的首要安全措施。下面，就让我们编辑 `/etc/ssh/sshd_config` 文件并将端口设置为 9999：

```
Port 9999
```

保存更改并重启 sshd：

```
# systemctl restart sshd
# systemctl status sshd
```

重启 SSH 服务

正如你看到的那样，sshd 启动失败，但为什么会这样呢？

快速检查 `/var/log/audit/audit.log` 文件会发现 sshd 已经被拒绝在端口 9999 上开启（SELinux 的日志信息包含单词 "AVC"，所以这类信息可以被轻易地与其他信息相区分），因为这个端口是 JBoss 管理服务的保留端口：

```
# cat /var/log/audit/audit.log | grep AVC | tail -1
```

查看 SSH 日志

在这种情况下，你可以像先前解释的那样禁用 SELinux（但请不要这样做！），并尝试重启 sshd，且这种方法能够起效。但是，`semanage` 应用可以告诉我们在哪些端口上可以开启 sshd 而不会出现任何问题。

运行：

```
# semanage port -l | grep ssh
```

便可以得到一个 SELinux 允许 sshd 在哪些端口上监听的列表：

Semanage 工具

所以让我们在 `/etc/ssh/sshd_config` 中将端口更改为 9998 端口，增加这个端口到 sshportt 的上下文，然后重启 sshd 服务：

```
# semanage port -a -t ssh_port_t -p tcp 9998
# systemctl restart sshd
# systemctl is-active sshd
```

semanage 添加端口

如你所见，这次 sshd 服务被成功地开启了。这个例子告诉我们一个事实：SELinux 用它自己的端口类型的内部定义来控制 TCP 端口号。

例 2：允许 httpd 访问 sendmail

这是一个 SELinux 管理一个进程来访问另一个进程的例子。假如在你的 RHEL 7 服务器上，你要为 Apache 配置 mod_security 和 mod_evasive <https://linux.cn/article-5639-1.html> ，你需要允许 httpd 访问 sendmail，以便在遭受到 (D)DoS 攻击时能够用邮件来提醒你。在下面的命令中，如果你不想使得更改在重启后仍然生效，请去掉 `-P` 选项。

```
# semanage boolean -l | grep httpd_can_sendmail
# setsebool -P httpd_can_sendmail 1
# semanage boolean -l | grep httpd_can_sendmail
```

允许 Apache 发送邮件

从上面的例子中，你可以知道 SELinux 布尔设定（或者只是布尔值）分别对应于 true 或 false，被嵌入到了 SELinux 策略中。你可以使用 `semanage boolean -l` 来列出所有的布尔值，也可以管道至 grep 命令以便筛选输出的结果。

例 3：在一个特定目录而非默认目录下提供一个静态站点服务

假设你正使用一个不同于默认目录（ `/var/www/html` ）的目录来提供一个静态站点服务，例如 `/websites` 目录（这种情形会出现在当你把你的网络文件存储在一个共享网络设备上，并需要将它挂载在 /websites 目录时）。

a). 在 /websites 下创建一个 index.html 文件并包含如下的内容：

```
<html>
<h2>SELinux test</h2>
</html>
```

假如你执行

```
# ls -lZ /websites/index.html
```

你将会看到这个 index.html 已经被标记上了 default_t SELinux 类型，而 Apache 不能访问这类文件：

检查 SELinux 文件的权限

b). 将 `/etc/httpd/conf/httpd.conf` 中的 DocumentRoot 改为 /websites，并不要忘了更新相应的 Directory 块。然后重启 Apache。

c). 浏览 `http://<web server IP address>` ，则你应该会得到一个 503 Forbidden 的 HTTP 响应。

d). 接下来，递归地改变 /websites 的标志，将它的标志变为 `httpd_sys_content_t` 类型，以便赋予 Apache 对这些目录和其内容的只读访问权限：

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

e). 最后，应用在 d) 中创建的 SELinux 策略：

```
# restorecon -R -v /websites
```

现在重启 Apache 并再次浏览到 `http://<web server IP address>`，则你可以看到被正确展现出来的 html 文件：

确认 Apache 页面

总结

在本文中，我们详细地介绍了 SELinux 的基础知识。请注意，由于这个主题的广泛性，在单篇文章中做出一个完全详尽的解释是不可能的，但我们相信，在这个指南中列出的基本原则将会对你进一步了解更高级的话题有所帮助，假如你想了解的话。

假如可以，请让我推荐两个必要的资源来入门 SELinux：**NSA SELinux 页面 <https://www.nsa.gov/research/selinux/index.shtml>** 和 **针对用户和系统管理员的 RHEL 7 SELinux 指南 <https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/SELinux_Users_and_Administrators_Guide/part_I-SELinux.html>** 。

假如你有任何的问题或评论，请不要犹豫，让我们知晓吧。

---

via: **http://www.tecmint.com/selinux-essentials-and-control-filesystem-access/ <http://www.tecmint.com/selinux-essentials-and-control-filesystem-access/>**

作者：**Gabriel Cánepa <http://www.tecmint.com/author/gacanepa/>** 译者：**FSSlc <https://github.com/FSSlc>** 校对：**wxy <https://github.com/wxy>**

本文由 **LCTT <https://github.com/LCTT/TranslateProject>** 原创翻译，**Linux中国 <file:///root/github/my-notes/Manual/Linux.cn/RHCSA/RHCSA%E7%B3%BB%E5%88%9713%E5%9C%A8%20RHEL%207%20%E4%B8%AD%E4%BD%BF%E7%94%A8%20S.html>** 荣誉推出

原文：**http://www.tecmint.com/selinux-essentials-and-control-filesystem-access/ <http://www.tecmint.com/selinux-essentials-and-control-filesystem-access/>**

作者： Gabriel Cánepa

译文：**LCTT <http://lctt.github.io/>** **https://linux.cn/article-6339-1.html <https://linux.cn/article-6339-1.html>** 译者： FSSlc

本文导航

相关阅读

RHCSA

- RHCSA 系列（十二）：使用 Kickstart 完成 RHEL 7 的自动化安装 2015-10-2
- RHCSA 系列（十四）： 在 RHEL 7 中设置基于 LDAP 的认证 2015-10-4
- RHCSA 系列（七）：使用 ACL（访问控制列表) 和挂载 Samba/NFS 共享 2015-9-22
- RHCSA 系列（八）：加固 SSH，设定主机名及启用网络服务 2015-9-23
- RHCSA 系列（九）：安装、配置及加固一个 Web 和 FTP 服务器 2015-9-24
- RHCSA 系列（十）：Yum 包管理、Cron 自动任务计划和监控系统日志 2015-9-26