Linux服务器构建与运维管理

从基础到实战（基于 openEuler）

第12章：openEuler的安全加固

阮晓龙

13938213680 / ruanxiaolong@hactcm.edu.cn

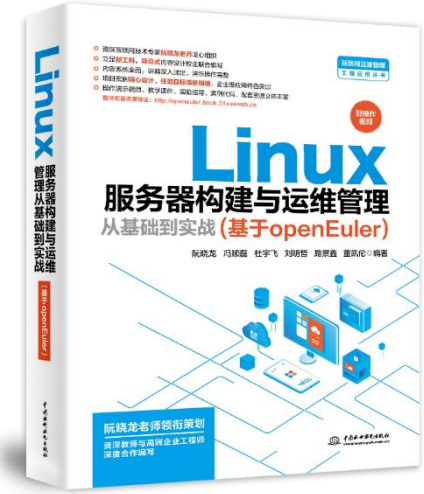
https://internet.hactcm.edu.cn
http://www.51xueweb.cn

河南中医药大学信息技术学院互联网技术教学团队
河南中医药大学医疗健康信息工程技术研究所

2024.11

提纲

- 进行操作系统安全加固
  - 操作系统的安全风险
  - openEuler安全加固指南
- 提升操作系统内核安全
  - 了解SELinux
  - 管理SELinux
- 保护操作系统业务安全
  - 理解主机防火墙
  - 管理Firewalld服务
  - 配置防火墙和数据包过滤器
- 审计操作系统安全漏洞
  - 了解安全检测
  - 使用Nmap进行安全检测

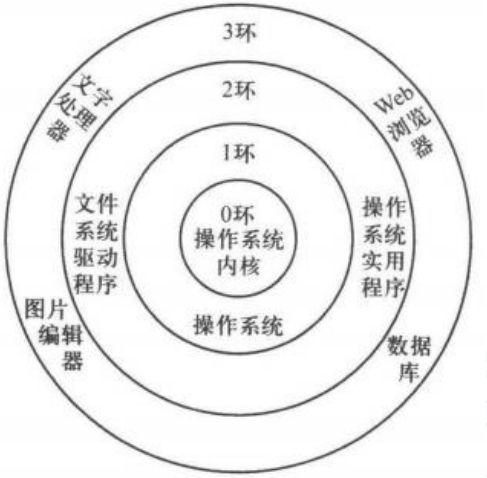


1. 进行操作系统安全加固

- 系统安全是指在系统生命周期内，应用系统安全工程和系统安全管理方法，辨识系统中的隐患，采取有效的控制措施使其危险性最小，从而使系统在规定的性能、时间和成本范围内达到最佳的安全程度。
- 操作系统是信息系统的重要组成部分，操作系统的安全在整个信息系统的安全性中起到至关重要的作用，没有操作系统的安全，信息系统的安全性将犹如建在沙丘上的城堡一样没有牢固的根基。
  - 操作系统位于软件系统的底层，需要为其上运行的各类应用服务提供支持
  - 操作系统是系统资源的管理者，对所有系统软、硬件资源实施统一管理
  - 作为软硬件的接口，操作系统起到承上启下的作用，应用软件对系统资源的使用与改变都是通过操作系统来实施

1. 进行操作系统安全加固

- 如果没有合理设置和防护，操作系统会成为计算机系统的薄弱点，在遭遇信息威胁时成为最脆弱的风险点。
- 为了实现安全目标：
  - 操作系统需要从用户管理、资源访问行为管理以及数据安全、网络访问安全等各个方面对系统行为进行控制，保证破坏系统安全的行为难以发生。
  - 操作系统需要对系统的所有行为进行记录，使攻击等恶意行为一旦发生就会留下痕迹，使安全管理人员有据可查。



1. 进行操作系统安全加固

- 操作系统的安全风险主要分为以下3个方面。
  - 硬件设备的安全风险。
    - 外部硬件设备的运行情况是否正常，硬件设备所处的环境是否长期正常稳定，在使用过程中应防止因异常关机或设备零件故障造成操作系统的无法正常使用。
  - 交互过程的安全风险。
    - 系统使用过程中，存在用户权限混乱、服务进程异常等安全风险。
  - 网络病毒漏洞的安全风险。
    - 当操作系统在网络中提供服务时，将会面临着服务攻击、口令破解攻击、欺骗用户攻击、网络监听攻击、端口扫描攻击等网络安全风险。

1. 进行操作系统安全加固

- openEuler中已经内置多种安全保护机制。
  - PAM机制。
    - PAM（Pluggable Authentication Modules）机制是一套共享库，其目的是提供一个框架和一套编程接口，将认证工作由程序员交给管理员。
    - PAM允许管理员在多种认证方法之间进行选择，它能够在不重新编译与认证相关应用程序的情况下改变本地认证方法。
  - 安全审计机制。
    - 虽然openEuler不能预测何时服务器会遭受攻击，但是可以记录入侵者的行踪，记录事件信息和网络连接情况，信息保存到日志文件中，为后续复查提供支持。
  - 强制访问控制机制。
    - 强制访问控制（Mandatory Access Control，MAC）是一种由系统管理员从全系统的角度定义和实施的访问控制机制，它通过标记系统中的主客体，强制性地限制信息的共享和流动，使用户只能访问与其相关的、指定范围的信息，防止信息泄密，杜绝访问权限的交叉混乱。
  - 防火墙机制。
    - 通过防火墙的控制策略、行为审计、抗攻击等功能，保障服务器的自身安全。

1. 进行操作系统安全加固

- 操作系统作为信息系统的核心，承担着管理硬件资源和软件资源的重任，是整个信息系统安全的基础。
- 操作系统之上的各种应用，要想获得信息的完整性、机密性、可用性和可控性，必须依赖于操作系统，脱离了对操作系统的安全保护，仅依靠其他层面的防护手段来阻止黑客和病毒等对网络信息系统的攻击，是无法满足安全需求的。
- 需要对操作系统进行安全加固，构建动态、完整的安全体系，增强业务的安全性。

1. 进行操作系统安全加固

- 在openEuler中，提供了两种安全加固的方式。
  - 手动修改配置或执行命令。
    - 可以通过修改/etc/openEuler_security/security.conf配置文件进行系统安全加固。
  - 使用工具批量修改加固项。
    - openEuler安全加固工具security-tool以openEuler-security.service服务运行。
    - 系统首次启动时会自动运行该服务去执行默认加固策略，且自动设置后续开机不启动该服务。
  - 由于安全加固对系统至关重要，只有root用户允许修改并应用安全加固策略。

操作系统加固概述

介绍对openEuler系统进行加固的目的和加固方案。

- 操作系统加固概述
  - 加固目的
  - 加固方案
  - 加固影响

须知

由于安全加固对系统至关重要，因此只有root用户允许修改并应用安全加固策略。

加固目的

操作系统作为信息系统的核心，承担着管理硬件资源和软件资源的重任，是整个信息系统安全的基础。操作系统之上的各种应用，要想获得信息的完整性、机密性、可用性和可控性，必须依赖于操作系统。脱离了对操作系统的安全保护，仅依靠其他层面的防护手段来阻止黑客和病毒等对网络信息系统的攻击，是无法满足安全需求的。

因此，需要对操作系统进行安全加固，构建动态、完整的安全体系，增强产品的安全性，提升产品的竞争力。

加固方案

本章描述openEuler的安全加固方案，包括加固方式和加固内容。

加固方式

用户可以通过手动修改加固配置或执行相关命令对系统进行加固，也可以通过加固工具批量修改加固项。openEuler的安全加固工具security tool以openEuler-security.service服务的形式运行。系统首次启动时会自动运行该服务去执行默认加固策略，服务运行后会将该服务自动设置为后续开机不启动。

用户可以通过修改security.conf，使用安全加固工具实现个性化安全加固的效果。

加固内容

openEuler系统加固内容主要分为以下5个部分：

须知
加固目的
加固方案
加固影响

1. 进行操作系统安全加固

- 在openEuler中，最基本的安全加固有5个方面。
  - 系统服务。
    - 将系统中运行的服务配置进行调整修改，提高服务配置的安全性。
  - 文件权限。
    - 通过修改文件和目录的权限和属主提升系统安全性。
  - 内核参数。
    - 内核参数决定配置和应用特权的状态，可通过参数配置进行提升系统的安全性。
  - 授权认证。
    - 将通过限制授权系统的操作权限、用户权限等内容提升系统的安全性。
  - 账号口令。
    - 通过删除所有测试账号、共享账号，设置合理的用户权限策略，制定复杂的用户密码并定期检查等提升系统的安全性。

帐户口令

具体操作建议

屏蔽系统帐户
限制使用su命令的帐户
设置口令复杂度
设置口令有效期
设置口令的加密算法
登录失败超过三次后锁定
加固su命令

- 帐户口令
  - 屏蔽系统帐户
  - 限制使用su命令的帐户
  - 设置口令复杂度
  - 设置口令有效期
  - 设置口令的加密算法
  - 登录失败超过三次后锁定
  - 加固su命令

屏蔽系统帐户

官方文档的加固指导

说明

除了用户帐户外，其他帐号称为系统帐户。系统帐户仅系统内部使用，禁止用于登录系统或其他操作，因此屏蔽系统帐户。

实现

将系统帐户的Shell修改为/sbin/nologin。

```
usermod -L -s /sbin/nologin $systemaccount
```

> 说明：$systemaccount 指系统帐户。

限制使用su命令的帐户

说明

授权认证

具体操作建议

设置网络远程登录的警告...
禁止通过Ctrl+Alt+Del重启...
设置终端的自动退出时间
设置用户的默认umask值...
设置GRUB2加密口令
安全单用户模式
禁止交互式启动

设置网络远程登录的警告信息

说明

设置网络远程登录的警告信息，用于在登录进入系统之前向用户提示警告信息，明示非法侵入系统可能受到的惩罚，吓阻潜在的攻击者。同时也可以隐藏系统架构及其他系统信息，避免招致对系统的目标性攻击。

实现

该设置可以通过修改/etc/issue.net文件的内容实现。将/etc/issue.net文件原有内容替换为如下信息（openEuler默认已设置）：

```
Authorized users only. All activities may be monitored and reported.
```

禁止通过Ctrl+Alt+Del重启系统

说明

操作系统默认能够通过"Ctrl+Alt+Del"进行重启，建议禁止该项特性，防止因为误操作而导致数据丢失。

实现

禁止通过"Ctrl+Alt+Del"重启系统的操作步骤如下：

1. 删除两个ctrl-alt-del.target文件，参考命令如下：

```
rm -f /etc/systemd/system/ctrl-alt-del.target
rm -f /usr/lib/systemd/system/ctrl-alt-del.target
```

加固影响

对文件权限、帐户口令等安全加固，可能造成用户使用习惯变更，从而影响系统的易用性。影响系统易用性的常见加固项请参见表1。

表 1 加固影响说明

加固项	建议加固	易用性影响	openEuler默认是否设置了该加固项
字符界面等待超时限制	当字符界面长时间处在空闲状态，字符界面会自动退出。 说明： 当用户通过SSH登录，超时时间由/etc/profile文件的TMOUT字段和/etc/ssh/sshd_config文件的ClientAliveInterval字段两个值中较小的值决定。建议加固为300秒。	用户长时间不操作字符界面，字符界面会自动退出。	否
口令复杂度限制	口令长度最小为8位，口令至少包含大写字母、小写字母、数字和特殊字符中的3种。	系统中所有用户不能设置简单的口令，口令必须符合复杂度要求。	否
限定登录失败时的尝试次数	当用户登录系统时，口令连续输错3次，帐户将被锁定60秒，锁定期间不能登录系统。	用户不能随意登录系统，帐户被锁定后必须等待60秒。	是
用户默认umask值限制	设置所有用户的默认umask值为077，使用户创建文件的默认权限为600、目录权限为700。	用户需要按照需求修改指定文件或目录的权限。	否
口令有效期	口令有效期的设置通过修改/etc/login.defs文件实现，加固默认值为口令最大有效期90天，两次修改口令的最小间隔时间为0，口令过期前开始提示天数为7。	口令过期后用户重新登录时，提示口令过期并强制要求修改，不修改则无法进入系统。	否
su权限限制	su命令用于在不同帐户之间切换。为了增强系统安全性，有必要对su命令的使用权进行控制，只允许root和wheel群组的帐户使用su命令，限制其他帐户使用。	普通帐户执行su命令失败，必须加入wheel群组才可以su成功。	是
禁止root帐户直接SSH登录系统	设置/etc/ssh/sshd_config文件的PermitRootLogin字段的值为no，用户无法使用root帐户直接SSH登录系统。	用户需要先使用普通帐户SSH登录后，再切换至root帐户。	否
SSH强加密算法	SSH服务的MACs和Ciphers配置，禁止对CBC、MD5、SHA1算法的支持，修改为CTR、SHA2算法。	当前 Windows下使用的部分低版本的Xshell、PuTTY不支持aes128-ctr、aes192-ctr、aes256-ctr、hmac-sha2-256、hmac-sha2-512算法，可能会出现无法通过SSH登录系统的情况，请使用最新的PuTTY（0.63版本以上）、Xshell（5.0版本及以上版本）登录。	是

操作系统安全的关键是人

严格按照指南做

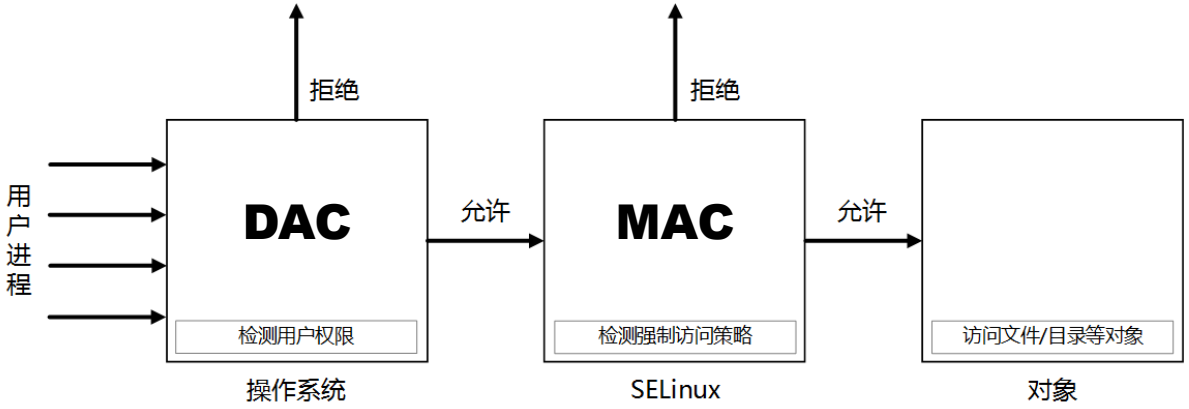
严格遵照制度做

2. 提升操作系统内核安全

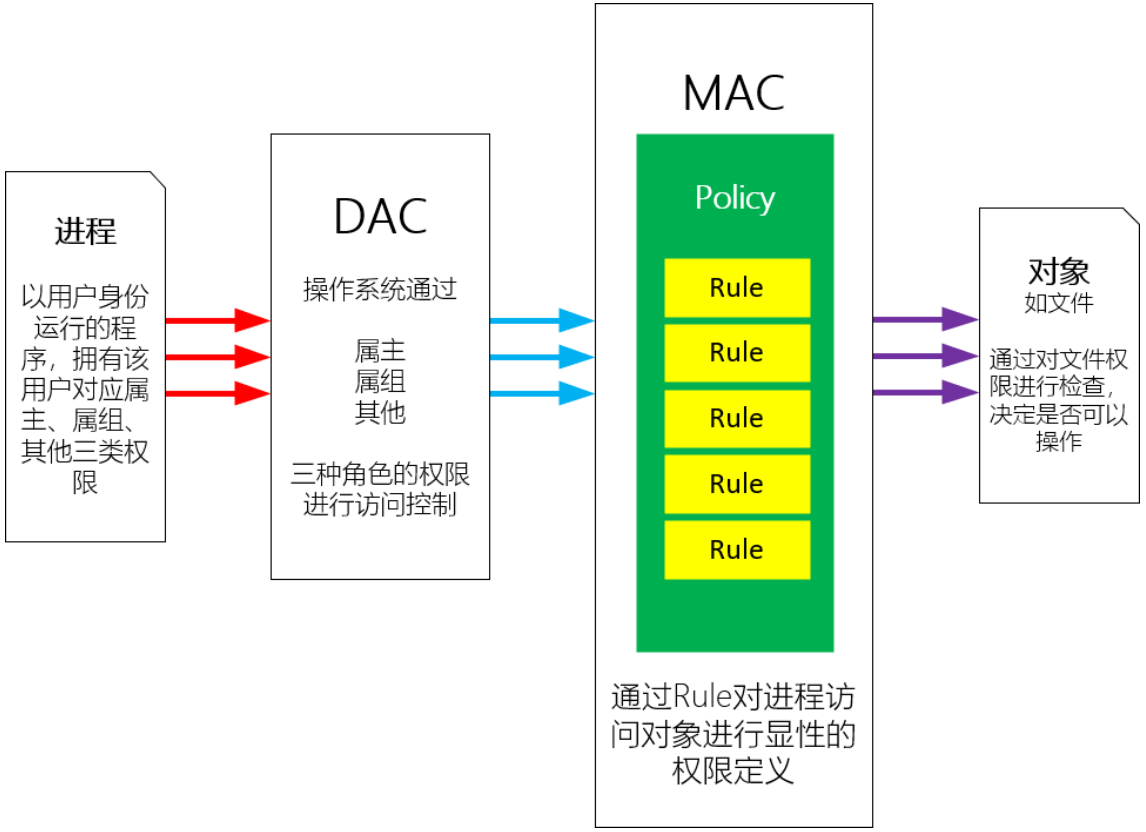
- SELinux（Security-Enhanced Linux）是强制访问控制机制在Linux内核上的实现，旨在提升Linux Kernel安全性。
  - SELinux（Security-Enhanced Linux）是Linux内核的一个模块，也是Linux的一个安全子系统。
  - SELinux的实现了强制访问控制MAC（Mandatory Access Control），每个进程和系统资源都有一个特殊的安全标签，资源能否被访问除了DAC规定的原则外，还需要判断每一类进程是否拥有对某一类资源的访问权限。
  - Linux Kernel 2.6及以上版本均集成SELinux模块。

2. 提升操作系统内核安全

- SELinux采用最小权限原则，最大限度地减小系统中服务进程可访问资源的范围，进而实现对系统安全的保护。
  - 启用SELinux后，用户进程不能直接访问到系统中的任何文件、目录、端口等。
  - 启用SELinux后，访问资源的流程：



1. **操作系统检查用户权限是否允许访问（DAC控制权限）。**
2. **如果允许，继续检测SELinux强制访问控制策略是否允许（MAC访问控制）。**
3. **如果允许，用户进程可访问系统内的对象。**



强制访问控制（Mandatory Access Control，MAC）：
是一种由系统管理员从全系统的角度定义和实施的访问控制机制，它通过标记系统中的主客体，强制性地限制信息的共享和流动，使用户只能访问与其相关的、指定范围的信息，防止信息泄密，杜绝访问权限的交叉混乱。

自主访问控制（Discretionary Access Control，DAC）：
是一种由客体的属主对自己的客体进行管理，由属主自行决定是否将自己的客体访问权或部分访问权授予其他主体，这种控制方式是自主的。

2. 提升操作系统内核安全

- SELinux is not:
  - antivirus software
  - replacement for passwords, firewalls, and other security systems
  - all-in-one security solution

- SELinux is：
  - 旨在增强现有的安全解决方案。
  - 即使运行 SELinux，仍需要遵循好的安全实践，如保持软件更新、使用安全的密码、使用防火墙。

2. 提升操作系统内核安全

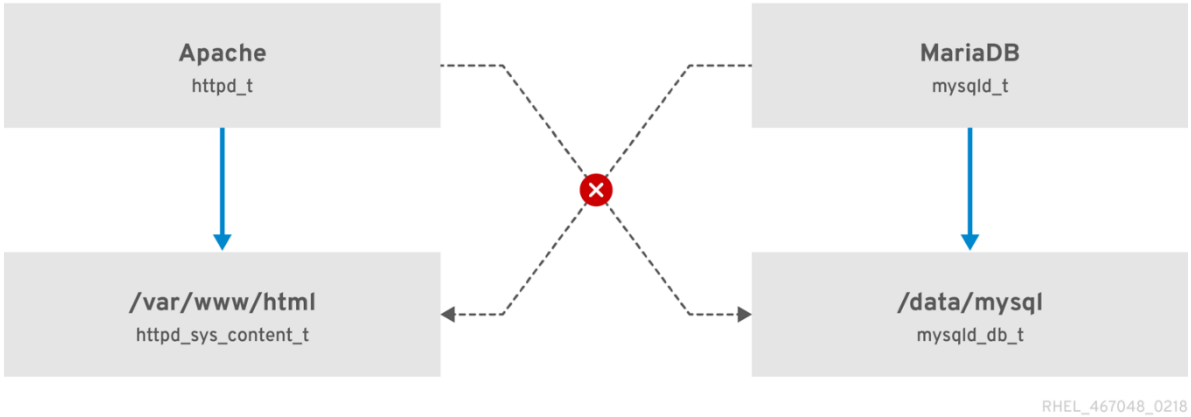
- SELinux的工作主要是通过安全策略和安全上下文协同实现。
  - 安全策略。
    - 定义主体（进程）读取对象（系统中文件、目录、端口等均可）的规则类数据库，规则中记录了哪个类型的主体使用哪个方法读取哪一个对象是允许还是拒绝，并定义了哪种行为是允许或拒绝。
  - 安全上下文（Security Context）是SELinux的核心。
    - 安全上下文由4个部分组成。它们分别是user、role、type和security level。
      - a．user：SELinux的用户类型，如user_u（普通用户登录系统后的预设）、system_u（开机过程中系统进程的预设）、root（root用户登录后的预设）、unconfined_u（多数本地进程运行的预设）。
      - b．role：定义文件（object_r）、进程和用户（system_r）的角色，角色可以限制“type”的使用。
      - c．type：数据类型，是定义何种进程类型访问何种文件对象目标的策略。
      - d．security level：安全等级，每个对象有且只有一个级别，等级为s0～s15，s0等级最低。策略默认等级为s0。

2. 提升操作系统内核安全

- Policy and Rule：政策 与 规则
  - Policy就是规则库，许多的Rule集合在一起就形成了Policy。
  - 按照MAC的定义，最佳方案就是系统上所有的程序都能够受到保护。
    - 操作系统运行的程序非常多，为所有程序撰写Policy不现实。
    - Policy的撰写难度非常高，让使用Linux操作系统的人都掌握撰写方法不可能。
    - 综合考虑安全性和易用性，只保护重要程序是最佳的选择。
  - 内置了三种政策。
    - 提高了SELinux易用性，让SELinux能够广泛应用。
    - 保护了关键的业务和程序。
    - 操作系统使用人员不需要掌握撰写Policy和Rule的专业技能。

2. 提升操作系统内核安全

- 举例：
  - 通过 SELinux 以安全的方式运行 Apache 和 MariaDB 的示例。

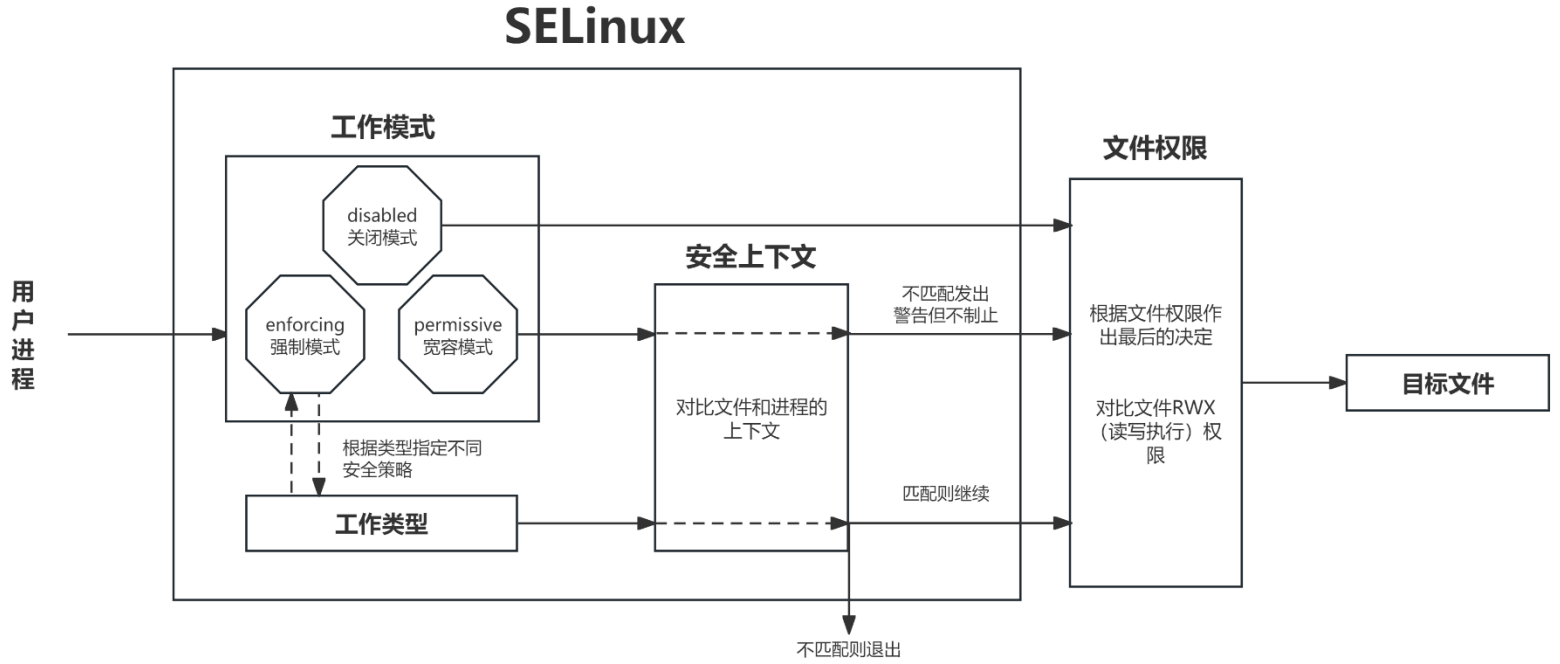


SELinux 允许作为 httpd_t 运行 Apache 进程访问 /var/www/html/ 目录，并且拒绝同一进程访问 /data/mysql/ 目录，因为 httpd_t 和 mysqld_db_t 类型上下文没有允许规则。

另一方面，作为 mysqld_t 运行的 MariaDB 进程可以访问 /data/mysql/ 目录，SELinux 也会正确地拒绝使用 mysqld_t 类型的进程来访问标记为 httpd_sys_content_t 的 /var/www/html/ 目录。

2. 提升操作系统内核安全

- SELinux的工作原理。



2. 提升操作系统内核安全

- SELinux的工作模式与类型。
  - SELinux在工作过程中分为3种模式，用于不同层次的系统安全。

表 12-0-1　SELinux 运行模式设置说明

模式	说明
enforcing	强制模式 该模式是默认和推荐的操作模式，在强制模式下，SELinux 正常运行，在整个系统上强制加载安全策略
permissive	许可模式，又叫宽容模式 该模式启用 SELinux，但不阻止任何操作，只提出警告信息和进行记录。该模式下策略规则不被强制执行，只接收到审核拒绝的信息，不做任何安全策略加固
disabled	停用模式 该模式下，SELinux 是完全关闭的。关闭 SELinux 后，系统不再强制执行 SELinux 策略，还会停止标记任何对象，如果业务系统为正式服务的系统，在关闭 SELinux 的情况下运行一段时间后，由于大量的文件没有进行标记，未来启用 SELinux 是非常困难的 强烈建议不要关闭 SELinux，如不需要使用 SELinux，可将工作模式调整为许可模式

2. 提升操作系统内核安全

- SELinux的工作模式与类型。
  - 工作类型指定SELinux使用的安全策略，openEuler内置了3种安全策略。

表 12-0-2　SELinux 安全策略类型

类型	说明
targeted	默认值，表示部分程序受到 SELinux 的保护 对系统中目标网络的进程进行访问控制，如 dhcpd、httpd、named、nscd、ntpd、portmap、snmpd、squid 以及 syslogd 等
minimum	targeted 的简化版，仅选定的程序受到保护
mls（strict）	Multi-Level Security，多级安全限制 对系统中所有进程与操作进行严格访问控制，属于较严格的规则集合

工作模式
SELINUX

工作模式决定SELinux是否启用

- enforcing：强制模式，启用SELinux
- permissive：宽容模式，启用SELinux，但不阻止任何操作，只提出警告信息
- disabled：关闭模式，关闭SELinux

工作类型
SELINUXTYPE

工作类型指定SELinux使用的安全政策

- targeted：默认值，有限程序收到SELinux的保护
- minimum：targeted的简化版，仅选定程序受保护
- mls：Multi-Level Security，多级安全限制，较严格

2. 提升操作系统内核安全

- openEuler内置SELinux并默认为开机自启动。
- 管理SELinux的工作模式和运行状态
  - 查看SELinux的运行状态
    - #sestatus
  - 查看SELinux的工作模式
    - #getenforce
  - 设置SELinux的工作模式为强制模式
    - #setenforce 1
  - 设置SELinux的工作模式为宽容模式
    - #setenforce 0
  - 禁用/启动SELinux
    - 修改SELinux的配置文件　　#vi /etc/selinux/config
    - 重启操作系统　　#reboot

Use the setenforce utility to change between enforcing and permissive mode.

Changes made with setenforce do not persist across reboots.

2. 提升操作系统内核安全

- 管理SELinux的软件包
  - 策略：
    - selinux-policy-targeted
    - selinux-policy-mls
  - 工具：
    - policycoreutils
    - policycoreutils-gui
    - libselinux-utils
    - policycoreutils-python-utils
    - setools-console
    - checkpolicy

对焦模式

使用 SELinux

对红帽文档提供反馈

1. SELinux 入门

2. 更改 SELinux 状态和模式

3. 管理限制和未限制的用户

管理限制和未限制的用户

3.1. SELinux 中的限制和未限制的用户

3.2. SELinux 用户的角色和权限

3.3. SELinux 中约束的非管理员角色

3.4. SELinux 中受限的管理员角色

3.5. 添加一个新用户会自动映射到 SELinux unconfined_u 用户

3.6. 以 SELinux 限制的用户身份添加新用户

3.7. 在 SELinux 中限制常规用户

3.8. 通过映射到 sysadm_u来限制管理员

3.9. 使用 sudo 和 sysadm_r 角色约束管理员

3.10. 其他资源

4. 为使用非标准配置的应用程序和服务配置 SELinux

5. 故障排除与 SELinux 相关的问题

6. 使用多级别安全（MLS）

7. 使用多类别安全(MCS)进行数据保密性

8. 编写自定义 SELinux 策略

9. 为容器创建 SELinux 策略

10. 在多个系统中部署相同的 SELinux 配置

3.2. SELinux 用户的角色和权限

PDF

SELinux 策略将每个 Linux 用户映射到 SELinux 用户。这允许 Linux 用户继承 SELinux 用户的限制。

您可以通过调整策略中的布尔值来自定义 SELinux 策略中受限用户的权限。您可以使用 `semanage boolean -l` 命令确定这些布尔值的当前状态。要列出所有 SELinux 用户、MLS 和 MCS 的级别和范围，请以 `root` 用户身份使用 `semanage user -l` 命令。

表 3.1. SELinux 用户的角色

User	默认角色	其他角色
`unconfined_u`	`unconfined_r`	`system_r`
`guest_u`	`guest_r`	
`xguest_u`	`xguest_r`	
`user_u`	`user_r`	
`staff_u`	`staff_r`	`sysadm_r`
		`unconfined_r`
		`system_r`
`sysadm_u`	`sysadm_r`	
`root`	`staff_r`	`sysadm_r`
		`unconfined_r`
		`system_r`

对焦模式
使用 SELinux
对红帽文档提供反馈
1. SELinux 入门
2. 更改 SELinux 状态和模式
3. 管理限制和未限制的用户
4. 为使用非标准配置的应用程序和服务配置 SELinux
为使用非标准配置的应用程序和服务配置 SELinux
4.1. 在非标准配置中为 Apache HTTP 服务器自定义 SELinux 策略
4.2. 使用 SELinux 布尔值调整用于共享 NFS 和 CIFS 卷的策略
4.3. 查找用于管理对非标准目录的访问的正确 SELinux 类型
4.4. 为非特权用户管理对非标准共享目录的访问
5. 故障排除与 SELinux 相关的问题
6. 使用多级别安全（MLS）
7. 使用多类别安全(MCS)进行数据保密性
8. 编写自定义 SELinux 策略
9. 为容器创建 SELinux 策略
10. 在多个系统中部署相同的 SELinux 配置
11. 配置 polyinstantiated 目录
法律通告

4.1. 在非标准配置中为 Apache HTTP 服务器自定义 SELinux 策略

选择页面格式
Multi-page

您可以将 Apache HTTP 服务器配置为在不同端口中侦听，并在非默认目录中提供内容。要防止 SELinux 拒绝带来的后果，请按照以下步骤调整系统的 SELinux 策略。

先决条件

- 已安装 `httpd` 软件包，并将 Apache HTTP 服务器配置为侦听 TCP 端口 3131，并使用 `/var/test_www/` 目录而不是默认的 `/var/www/` 目录。
- `policycoreutils-python-utils` 和 `setroubleshoot-server` 软件包已安装在您的系统中。

步骤

1. 启动 `httpd` 服务并检查状态：

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

2. SELinux 策略假设 `httpd` 在端口 80 上运行：

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

3. 更改 SELinux 类型端口 3131 使其与端口 80 匹配：

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

4. 再次启动 `httpd` ：

```
# systemctl start httpd
```

5. 但是，内容仍无法访问：

```
# wget localhost:3131/index.html
...
```

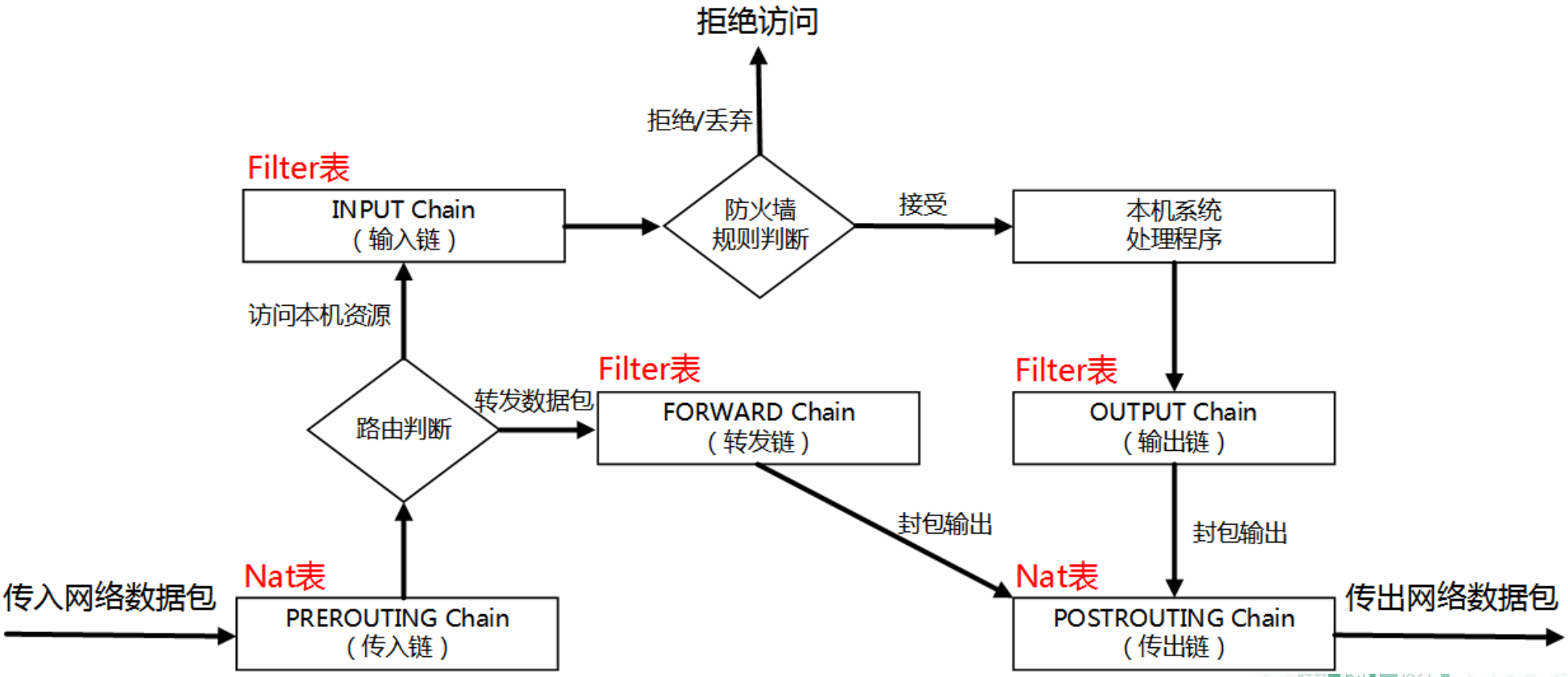
3. 保护操作系统业务安全

- 防火墙是服务器安全的重要保障系统，遵循允许和业务来往的网络通信机制，提供网络通信过滤服务。
- 从保护对象上区分，防火墙可分为主机防火墙和网络防火墙。
  - 主机防火墙是部署在一台计算机系统上的软件，针对单个主机进行防护。
  - 网络防火墙是部署在两个网络之间的设备或一整套装置，针对一个网络进行防护。通常部署在网络边界以加强访问控制，其将网络划分为可信与不可信区域，对流入流出的网络流量进行过滤，实现对可信网络的防护。

3. 保护操作系统业务安全

- 不管是Linux、Unix、Mac还是Windows操作系统，主机防火墙都是设置操作系统与外界网络之间的一系列软件组合。
- 主机防火墙通过检测、限制通过的数据流，尽可能地对外屏蔽操作系统的信息、结构和运行状态，有选择地接受外部网络的访问请求，进而实现提升主机安全性的目的。
- 主机防火墙工作在网络层，属于典型的包过滤防火墙。
  - 把网络层作为数据监控对象，对每个数据包的头部、协议、地址端口及类型信息进行分析。
  - 如果数据包的某个或多个部分与预先设定的防火墙规则（Filtering Rule）匹配，则按照防火墙规则进行处理，否则直接丢弃。

包过滤防火墙过滤过程

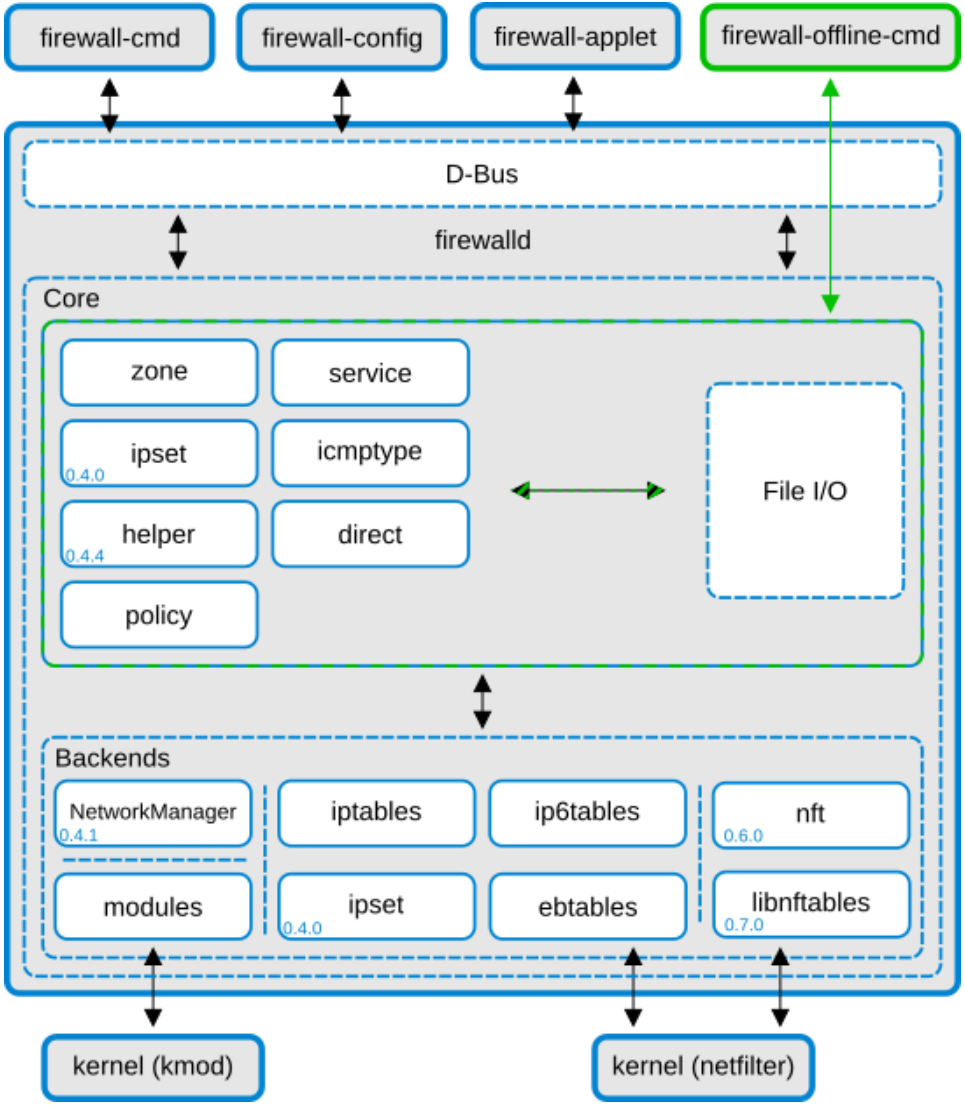

拒绝访问
拒绝/丢弃
Filter表
INPUT Chain
（输入链）
防火墙
规则判断
接受
本机系统
处理程序
访问本机资源
Filter表
转发数据包
路由判断
FORWARD Chain
（转发链）
Filter表
OUTPUT Chain
（输出链）
封包输出
封包输出
Nat表
传入网络数据包
PREROUTING Chain
（传入链）
Nat表
POSTROUTING Chain
（传出链）
传出网络数据包

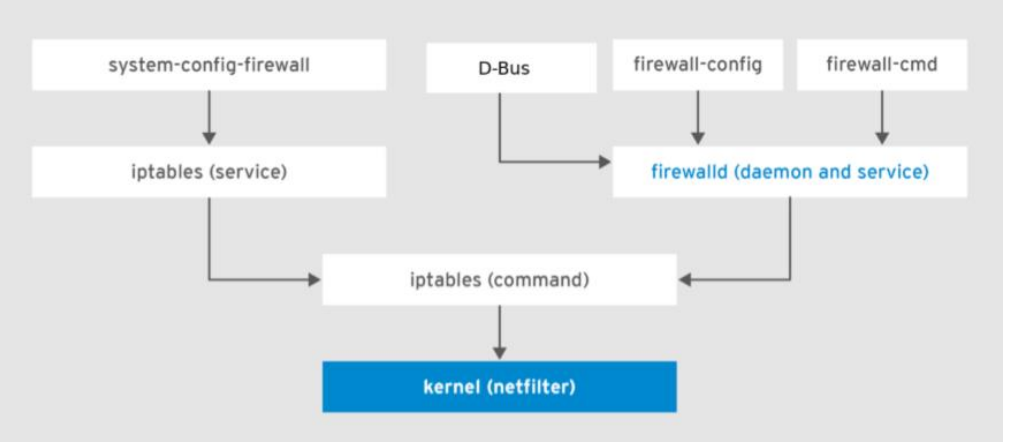
3. 保护操作系统业务安全

- 包过滤防火墙中存在“三表五链”用于防火墙的数据通信过滤与处理。
  - Filter表：对数据包进行过滤
  - Nat表：用于地址转换和端口转发
  - Mangle表：用于对数据包进行修改
  - PREROUTING链：路由之前的数据包传入链，目的地址转换
  - INPUT链：输入数据包链，数据包流入内核空间
  - FORWARD链：转发路由数据包链，数据包在端口间转发
  - OUTPUT链：输出数据包链，数据包流出内核空间
  - POSTROUTING：路由之后数据包，源地址转换

3. 保护操作系统业务安全

- 防火墙是重要的系统安全防护措施之一，但也不能过分依赖防火墙，因为防火墙自身具有一定的局限性。
  - 防火墙可以阻断攻击，但不能消灭攻击源。
  - 防火墙不能抵抗最新的未设置策略的攻击漏洞。
  - 防火墙的并发连接数限制容易导致服务拥塞或溢出。
  - 防火墙对针对服务器合法开放的端口的攻击无法阻止。
  - 防火墙对系统内部发起的攻击无法阻止。
  - 防火墙本身也会出现问题或受到攻击。
  - 防火墙无法防御病毒。





Firewalld 是由Red Hat发起的支持网络/防火墙区域(zone)定义网络链接以及接口安全等级的动态防火墙管理工具。

支持 IPv4、IPv6 防火墙设置以及以太网桥接，并且拥有运行时配置和永久配置选项。

支持允许服务或者应用程序直接添加防火墙规则的接口。

3. 保护操作系统业务安全

- firewalld 是一个防火墙服务守护进程。
  - 提供一个带有D-Bus接口的、动态可定制的、基于主机的防火墙。
  - 动态的是说启用、修改和删除规则时不需要重启防火墙守护进程。
- firewalld 使用区和服务的概念来简化流量管理。
  - zones是预定义的规则集。
  - 服务是预定义的规则，并在区中应用。

表 12-3-8　Firewalld 防火墙区域名称及策略规则

区域	默认策略规则
trusted	信任区域：信任该区域，接受来自该区域的所有网络连接
home	家庭区域：基本信任该区域，接受规则过滤的连接 默认开启 ssh、mdns、ipp-client、amba-client 与 dhcpv6-client 服务允许对外访问
internal	内部区域：基本信任该区域，接受规则过滤的连接
work	工作区域：基本信任该区域，接受规则过滤的连接 默认开启 ssh、ipp-client 与 dhcpv6-client 服务允许对外访问
public	公共区域：不信任该区域，接受规则过滤的连接 默认开启 ssh、dhcpv6-client 服务允许对外访问
external	外部区域：不信任该区域，对路由器隐藏信息，接受规则过滤的连接 默认开启 ssh 服务允许对外访问
dmz	非军事区域：信任该区域，该区域内主机可以访问其他区域，接受规则过滤的连接 默认开启 ssh 服务允许对外访问
block	限制区域：接收的任何数据包都被拒绝，且返回 icmp 信息 IPv4 返回 icmp-host-prohibited 信息，IPv6 返回 icmp6-adm-prohibited 信息
drop	丢弃区域：接收的任何数据包都被丢弃，不做任何回复

3. 保护操作系统业务安全

- 使用防火墙规则来实现特定的配置，以允许或阻止网络流量。
- 防火墙规则通常根据各种属性定义某些条件。
  - 属性可以是：
    - 源 IP 地址
    - 目标 IP 地址
    - 传输协议（TCP、UDP、…）
    - 端口
    - 网络接口
  - 策略指定 --set-target 选项。可用的目标如下：
    - ACCEPT - 接受数据包
    - DROP - 丢弃不需要的数据包
    - REJECT - 拒绝不需要的数据包，并带有 ICMP 回复
    - CONTINUE（默认）- 数据包将遵循以下策略和区域中的规则。

3. 保护操作系统业务安全

- 管理Firewalld的方式有两种
  - 使用工具
    - 图形化工具：firewall-config
    - 命令行工具：firewall-cmd
    - 命令行工具：firewall-offline-cmd
  - 编辑/etc/firewalld/services/目录中的 XML 文件
    - 如果没有添加或更改服务，在/etc/firewalld/services 中没有相应的XML文件。
    - 可以使用 /usr/lib/firewalld/services/中的文件作为模板。

3. 保护操作系统业务安全

- 配置Firewalld的方式有两种
  - 运行时配置
    - 不中断现有连接
    - 不能修改服务配置
  - 永久配置
    - 不立即生效，除非Firewalld重新启动或重新加载配置
    - 中断现有连接
    - 可以修改服务配置

Firewalld provides a dynamically managed firewall with support for network/firewall zones that define the trust level of network connections or interfaces. It has support for IPv4, IPv6 firewall settings, ethernet bridges and IP sets. There is a separation of runtime and permanent configuration options. It also provides an interface for services or applications to add firewall rules directly.

Benefits of using firewalld

Changes can be done immediately in the runtime environment. No restart of the service or daemon is needed.

With the firewalld D-Bus interface it is simple for services, applications and also users to adapt firewall settings. The interface is complete and is used for the firewall configuration tools firewall-cmd, firewall-config and firewall-applet.

The separation of the runtime and permanent configuration makes it possible to do evaulation and tests in runtime. The runtime configuration is only valid up to the next service reload and restart or to a system reboot. Then the permanent configuration will be loaded again. With the runtime environment it is possible to use runtime for settings that should only be active for a limited amount of time. If the runtime configuration has been used for evaluation, and it is complete and working, then it is possible to save this configuration to the permanent environment.

Features

- Complete D-Bus API
- IPv4, IPv6, bridge and ipset support
- IPv4 and IPv6 NAT support
- Firewall zones
- Predefined list of zones, services and icmptypes
- Simple service, port, protocol, source port, masquerading, port forwarding, icmp filter, rich rule, interface and source address handlig in zones
- Simple service definition with ports, protocols, source ports, modules (netfilter helpers) and destination address handling
- Rich Language for more flexible and complex rules in zones
- Timed firewall rules in zones
- Simple log of denied packets
- Direct interface
- Lockdown: Whitelisting of applications that may modify the firewall
- Automatic loading of Linux kernel modules
- Integration with Puppet
- Command line clients for online and offline configuration
- Graphical configuration tool using gtk3
- Applet using Qt5

Who is using it?

firewalld is used in the following Linux distributions as the default firewall management tool:

- RHEL 7 and newer
- CentOS 7 and newer
- Fedora 18 and newer
- SUSE 15 and newer
- OpenSUSE 15 and newer

Search...

Recent Posts

Strict Forward Ports
Strict Filtering of Docker Containers
firewalld 2.1.0 release
firewalld 2.0.0 release
Software fastpath with nftables flowtable

Quick Links

Report a new issue
Browse issues

firewall-cmd

Name

firewall-cmd — firewalld command line client

Synopsis

`firewall-cmd` [OPTIONS...]

Description

firewall-cmd is the command line client of the firewalld daemon. It provides an interface to manage the runtime and permanent configurations.

The runtime configuration in firewalld is separated from the permanent configuration. This means that things can get changed in the runtime or permanent configuration.

Options

Sequence options are the options that can be specified multiple times, the exit code is 0 if there is at least one item that succeeded. The `ALREADY_ENABLED` (11), `NOT_ENABLED` (12) and also `ZONE_ALREADY_SET` (16) errors are treated as succeeded. If there are issues while parsing the items, then these are treated as warnings and will not change the result as long as there is a succeeded one. Without any succeeded item, the exit code will depend on the error codes. If there is exactly one error code, then this is used. If there are more than one then `UNKNOWN_ERROR` (254) will be used.

The following options are supported:

General Options

`-h`, `--help`
Prints a short help text and exits.

`-V`, `--version`
Print the version string of firewalld. This option is not combinable with other options.

`-q`, `--quiet`
Do not print status messages.

Status Options

`--state`
Check whether the firewalld daemon is active (i.e. running). Returns an exit code 0 if it is active, *`RUNNING_BUT_FAILED`* if failure occurred on startup, *`NOT_RUNNING`* otherwise. See the section called "Exit Codes". This will also print the state to *`STDOUT`*.

`--reload`
Reload firewall rules and keep state information. Current permanent configuration will become new runtime configuration, i.e. all runtime only changes done until reload are lost with reload if they have not been also in

Search...

Recent Posts

Strict Forward Ports
Strict Filtering of Docker Containers
firewalld 2.1.0 release
firewalld 2.0.0 release
Software fastpath with nftables flowtable

Quick Links

Report a new issue
Browse issues

3. 保护操作系统业务安全

- firewall-cmd is the command line client of the firewalld daemon.
- firewall-cmd provides an interface to manage the runtime and permanent configurations.

表 12-3-9　firewall-cmd 命令常用参数

参数	说明
--permanent	将策略写入到永久生效表中
--add-rich-rule	添加富规则
--add-interface	将指定网卡的所有流量都导向某区域
--add-port	设置允许的端口
--add-service	设置允许的服务
--add-source	将指定 IP 地址的所有流量都导向某区域
--change-interface	设置网卡与区域进行关联
--get-active-zones	显示当前正在使用的区域与网卡名称
--get-default-zone	显示默认的区域名称
--get-services	显示预先定义的服务
--get-zones	显示可用的区域列表
--list-all	显示当前区域的网卡配置参数、资源、端口及服务
--list-all-zones	显示区域信息情况
--list-ports	显示所有正在运行的端口
--panic-off	关闭紧急模式
--panic-on	开启紧急模式
--query-panic	显示是否被拒绝
--reload	立即加载永久生效策略，不重启服务
--remove-port	设置默认区域不再允许指定端口的流量
--remove-source	不要将指定 IP 地址的所有流量导向某区域
--remove-service	设置默认区域不再允许指定服务的流量
--set-default-zone	设置默认的区域
--state	显示当前服务运行状态

全局管理

```
firewall-cmd --reload
firewall-cmd --list-all

firewall-cmd --get-zones
firewall-cmd --set-default-zone=public
firewall-cmd --get-active-zones
firewall-cmd --get-zone-of-interface=eth0

firewall-cmd --panic-on
firewall-cmd --panic-off
firewall-cmd --query-panic
```

端口管理

```
firewall-cmd --add-port=8080/tcp --permanent
firewall-cmd --add-port=81-85/tcp --permanent
firewall-cmd --add-port={161/udp,5001-5005/udp,3389/tcp} --permanent
firewall-cmd --remove-port=8080/tcp --permanent
firewall-cmd --list-ports
```

服务管理

```
firewall-cmd --get-services
firewall-cmd --add-service=http
firewall-cmd --add-service={dns,mysql}
firewall-cmd --remove-service=http
firewall-cmd --list-service
```

3. 保护操作系统业务安全

- 针对复杂的防火墙策略可使用rich rule（富规则）配置
  - firewall-cmd --add-rich-rule='rule [富规则内容]’：
    - 添加富规则。
  - firewall-cmd --remove-rich-rule='rule [富规则内容]’：
    - 删除富规则。
  - firewall-cmd --query-rich-rule='rule [富规则内容]’：
    - 判断是否存在某条富规则。
  - firewall-cmd --list-rich-rule:
    - 列出富规则信息。

表 12-3-10　富规则常用参数内容

参数	说明	参数示例
family	指定规则是否应用于 IPv4 或 IPv6 数据包如果无此参数，则说明同时应用于 IPv4 和 IPv6	family=ipv4 或 family=ipv6
source address	指定针对某源地址匹配规则策略	source address=xx.xx.xx.xx/xx
destination address	指定针对某目的地址匹配规则策略	destination address=xx.xx.xx.xx/xx
service	指定规则匹配的服务类型，服务名称为系统支持的列表。可通过“firewall-cmd --get-services”命令查看服务名称列表信息	service name=ssh
port	指定规则匹配的端口，既可以是一个独立的端口，又可以是端口的范围	port port=53 或 port port=53-59
protocol	指定规则匹配的协议类型	protocol=tcp 或 protocol=udp
icmp-block	指定拒绝 ICMP 的类型。可使用“firewall-cmd --get-icmptypes”命令查看支持的 ICMP 类型列表信息	icmp-block name=host-unknown
masquerade	是否打开规则的 IP 地址伪装配置	masquerade
forward-port	指定匹配转发端口的规则内容	forward-port port=80 protocol=tcp to-port=8080 to-addr=xx.xx.xx.xx
log	记录防火墙规则匹配触发的日志信息 格式：log [prefix="<prefix text>"] [level="<log level>"] [limit value="rate/duration"] prefix：指定前缀加入记录日志信息中 level：记录日志信息的等级，等级类型为 emerg、alert、crit、error、warning、notice、info 或者 debug，等级类型含义可参照表 12-3-13 limit：执行记录日志信息的频率	log level=info prefix="HTTP" limit value="3/s"
accept|reject|drop	指定规则匹配时触发的动作内容，接收、拒绝、丢弃	accept 或 reject 或 drop

富规则管理

```
firewall-cmd --zone=public --add-rich-rule=" \
        rule family="ipv4" \
        source address="10.10.1.0/24" \
        service name="{http,dns,ftp}" \
        accept "
firewall-cmd --zone=public --remove-rich-rule=" \
        rule family="ipv4" \
        source address="10.10.1.0/24" \
        service name="{http,dns,ftp}" \
        accept "

firewall-cmd --zone=public --add-rich-rule=" \
        rule family="ipv4" \
        source address="10.10.1.0/24" \
        protocol value="icmp" \
        accept " \
        --timeout=20

firewall-cmd --zone=public --add-rich-rule=" \
        rule family="ipv4" \
        source address="10.10.3.100" \
        port protocol="tcp" port="22" \
        limit value="2/m" \
        audit \
        accept"
```

4. 审计操作系统安全漏洞

- 在网络技术高度成熟发展的今天，即便SELinux和防火墙同时使用，也无法保障操作系统无任何安全风险。
- 只有不断通过对操作系统进行安全审计评估，及时发现系统安全漏洞并进行修复，才能不断提高主机的安全性。

没有绝对的安全!

4. 审计操作系统安全漏洞

- 安全审计是对目标主机的整体审计，主要包含以下内容与步骤。
  - 实施端口扫描与服务探测。
    - 如果目标主机处于开机状态，通过扫描与探测，可得到目标主机的端口状态（监听/关闭）、目标主机中服务程序列表和版本信息以及目标主机操作系统版本和内核信息等。
  - 以攻击渗透等方式进行模拟探测。
    - 根据获取到目标主机上的服务列表和版本信息，查询安全漏洞数据库，获取有针对性的攻击脚本，开展对目标主机系统的尝试性攻击，并记录目标主机对攻击的响应信息。
  - 对数据进行分析并产生报告。
    - 对获取的响应信息进行分析，并比对安全漏洞信息数据库，明确目标主机确实存在的安全漏洞信息，形成安全审计报告。
  - 安全风险处理。
    - 系统管理员根据安全审计报告的内容，逐项对照解决安全风险。

4. 审计操作系统安全漏洞

表 12-0-3　主机安全扫描工具

工具	功能类别	官方网站
Nmap	安全审计	https://nmap.org
Snort	网络入侵扫描	https://www.snort.org
ClamAV	病毒检测	http://www.clamav.net
Nessus	漏洞扫描	https://www.swri.org/nessus
OpenVAS	漏洞评估系统	https://www.openvas.org
Nikto	Web 服务器扫描	https://cirt.net/Nikto2
Metasploit	渗透测试工具	https://www.metasploit.com

4. 审计操作系统安全漏洞

- 安全检测是使用工具对系统进行扫描检测，验证是否存在安全风险或漏洞，完成系统安全评估工作。
- Nmap是最常用的安全检测软件的之一，是开源软件且提供了强大的网络扫描功能，通过该工具可发现网络中在线主机、端口监听状态、主机上运行的应用程序与版本信息以及操作系统的类型和版本等。

4. 审计操作系统安全漏洞

- Nmap是一款开放源代码的网络探测和安全审核的工具。
  - 其设计目标是快速地扫描大型网络，也支持对单个主机的扫描。
  - 其使用原始 IP 报文来发现网络上有哪些主机，主机提供什么服务(应用程序名和版本)，主机运行什么操作系统(包括版本信息)，主机配置了什么类型的报文过滤器/防火墙，以及主机的其他信息。
  - 通常使用Nmap进行安全审核。运维工程师通过Nmap进行日常运维工作，例如查看整个网络的信息，管理服务升级计划，监视主机和服务的运行等。
  - Nmap输出的是扫描目标的列表，以及每个目标的补充信息，至于是哪些信息则依赖于所使用的选项。

4. 审计操作系统安全漏洞

- Nmap工具进行主机扫描或安全检测时，语法格式为：

nmap [选项] [对象]

- 选项：
  - 使用 “--script” 选项则说明指定检测脚本名称。
  - 未使用此选项，则默认使用“default”脚本类型，进行基本的应用服务信息收集。
- 对象：
  - 指定安全扫描与漏洞检测的主机IP地址或IP地址段。
- 安装：
  - 使用yum工具进行安装：yum install nmap

表 12-4-1　Nmap 脚本选项及说明

选项	说明
--script	指定使用的脚本文件名称或脚本类型信息
--script-args	为脚本文件提供参数信息
--script-args-file	提供脚本执行参数文件
--script-trace	显示发送和接收到的数据信息
--script-updatedb	更新脚本数据库
--script-help	查看脚本帮助信息

表 12-4-2　Nmap 脚本类别

类别	说明
default	默认脚本，提供基本的应用服务信息搜集功能
auth	处理鉴权证书，绕开权限校验进行检测
broadcast	在局域网内探查更多服务开启状况，如 dhcp/dns/sqlserver 等
brute	暴力破解方式进行检测，主要针对常见的应用，如 http/snmp 等
discovery	对网络服务进行更为详细的检测，如 SMB 枚举、SNMP 查询等
dos	进行拒绝服务攻击
exploit	利用已知漏洞入侵系统
External	利用第三方数据库或资源进行检测
fuzzer	模糊测试，通过发送异常包探测潜在漏洞
intrusive	入侵性脚本，此脚本可能触发 IDS/IPS
malware	探测目标机是否感染了病毒、是否存在后门等信息
safe	与 intrusive 相反，属于安全性脚本
version	增强的服务与软件版本检测脚本
vuln	检测是否存在常见的漏洞

表 12-4-3　Nmap 基础检测选项及说明

选项	说明
-sn	只进行主机发现扫描，不进行端口扫描
-sU	指定使用 UDP 方式检测
-Pn	跳过主机发现，将所有主机都视为开启状态，进行端口扫描
-sL	仅列出开启的主机 IP 地址，不进行主机端口发现等扫描
-F	快速扫描模式，仅扫描开放率最高的前 100 个端口
--top-ports <number>	指定扫描模式，仅扫描开放率最高的<number>个端口
-PO	指定使用 IP 数据报方式检测
-sV	进行应用服务版本检测
-O	进行操作系统版本检测
--osscan-guess	进行操作系统类型等详细信息检测

网络与信息系统智能运维课程体系学习平台

***

本课程体系由
河南中医药大学信息技术学院建设

课程体系学习平台由河南中医药大学医疗健康信息工程技术研究所开发与技术保障

网络与信息系统智能运维课程体系学习平台
https://internet.hactcm.edu.cn

互联网运维管理工程应用丛书
http://www.51xueweb.cn

扫码学习
并获取课程资源