强化 Linux 服务器安全性的技巧

1 介绍

Linux 提供一个全面的安全体系，从网络防火墙控制到访问控制安全策略，应有尽有。尽管 Linux 采用了“默认安全”的原则，但本文将探讨各种尽量减少漏洞的默认设置和管理办法。

请谨记，在这里讨论的策略要考虑的选项，而不是明确的执行规则，在这里一些应用系统的修改必须始终进行兼容性测试，并确认支持的应用程序。因为市场上运行着不同版本以及不同厂家的 Linux。

本文介绍以下一般强化 Linux 的策略和方法：

- 最少化软件或服务。消除不必要的软件程序包和服务，最大限度地减少可能的攻击途径。
- 拧紧网络和用户接入。网络是一个恶意用户和应用程序的主要入口点。微调网络配置以及所有用户的访问点，有助于防止未经授权的访问。
- 保护应用程序和数据。设置设备，加载适当的文件系统（在某些情况下，使用加密）有助于维护应用程序和数据。
- 实现安全功能的执行策略。在一些情况下，安全策略可能要求额外的机制，如 TCP 封装器，可插式验证模块（PAM），或执行安全增强性 Linux（SELinux）。
- 运行适当的可选程序。除了维持系统的物理安全，应用支持，补丁和安全更新及时。还要监视系统日志和审计跟踪，执行一些工具寻找可疑的迹象。此外，定期进行安全性评估，审查与安全有关的做法以及程序。

下面将重点介绍前 4 个策略来强化 Linux 操作系统安全。至于运行保障措施，如系统管理，审计和更新，这里暂时先不讨论这些，虽然他们是同样重要，等后续找个专题继续讨论。

2 最少化软件安装

当安装 Linux 时，只安装所需要的软件包，可以减少受攻击的几率。软件包是一个潜在来源的 setuid 程序，Network services, 和 Libraries 有可能被用来获取非法登录凭证危险系统。使用预先配置文件的 Kickstart 提供一致和精确控制的安装，降低安全风险，以及安装管理工作的自动化。

在已经安装的 Linux 系统上，尽量减少不必要的修剪 RPMS 软件痕迹。例如，在大多数的服务器上 X-Windows 系统不需要，可以卸载。

3 尽量减少活动的服务

默认情况下，Linux 系统配置最小的一组服务，print services（cupsd 和 lpd），sendmail，sshd 和 xinetd（启动其他联网服务）。软件服务最少化，服务器上配置最少的应用程序服务类型这样可以消除潜在的攻击途径。理想情况下一种方法（但并不总是可行的话）是同类服务器配置相应类型的服务（例如，一类服务器配置 Apache HTTP 服务，另一类配置 NFS 服务，第三类配置打印服务，依此类推）。这种配置的好处是，如果一类系统被攻破，风险不至于传递到其它类服务器上。

最理想的方案，如果服务不使用，删除软件包与服务。在某些情况下，因为软件的依赖关系，可能无法删除服务，如果这样的话，可以禁用这类服务，在不被删除的情况下可以使用 service 和 chkconfig 命令进行禁用。

对于正在使用的服务，一定要保持最新的套装软件，应用最新的支持补丁和安全更新。为了防止未经授权的更改，确保在文件/etc/services 文件，确保它的拥有者是 root，只能由 root 修改，并链接到它不能被创建。

4 锁定网络服务

由于网络服务是一种常见的攻击途径，需要特别关注他们。一种常用的策略是尽量减少由 xinetd 启动的网络服务，禁用那些不需要的。它也可以设置资源限制，xinetd 的服务，以阻止潜在的拒绝服务（DoS）攻击。例如，可以限制每个服务或连接速率连接实例的数量，由指定的配置文件/etc/xinetd.conf 中的限制。（对于资源的控制选项，请参阅手册页 xinetd 和/etc/xinetd.conf）。

5 端口扫描

另一种常见的策略，加强安全检查哪些服务正在运行的系统上使用端口扫描工具。这些命令有时被用来识别 TCP 端口及相关服务：

```
#netstat  - tulp
#lsof -iTCP -sTCP：LISTEN
#nmap  - sTU <hostname>
```

6 TCP Wrappers

为了保护系统免受通过网络服务的攻击，共同管理的做法是配置 TCP Wrappers 和设置防火墙用 netfilter 和 iptables。 TCP 封装器之间进行调解传入客户端请求，请求的服务，以及他们访问控制的基础上定义的规则。您可以通过编辑文件/etc/hosts.deny 和/etc/hosts.allow 文件，限制和允许访问的主机或网络服务。

使用 TCP 包装的一种方法是明确了已知的恶意源信号入侵企图。例如，如果一个已知的恶意主机或网络的尝试攻击系统，您可以配置/etc/hosts.deny 文件拒绝访问，在同一时间，有关该事件的日志文件发送警告消息。例如，在/etc/hosts.deny 文件，这个项目在试图访问的 IP 地址 192.1.168.73 到一个已知的文件记录：

```
ALL:192.1.168.73:spawn /bin/echo date`%c %d >>/var/log/alert-admin
```

7 Netfilter and IPtables

Linux 附带一个内置的内核子系统称作为一个数据包过滤防火墙。Netfilter 执行以下三个操作：

- 包过滤
- 隐藏 IP 地址的网络地址转换（NAT）：后面一个公共的 IP 地址
- IP 伪装：更改为路由的数据包的 IP 包头信息

过滤规则(在内核表中存储所有这些操作)确定是否通过 Netfilter 可以让数据包接收，丢弃或转发。 Netfilter 适用对每个数据包的规则链。 iptables 命令是配置规则链的主要接口，或者可以使用防火墙配置工具（system-config-securitylevel）。需要注意的是直接修改规则文件/etc/sysconfig/iptables 或 ip6tables 这种方式不推荐使用。 iptables 文件中包含的标准适用于包过滤决定，如协议进行过滤，数据包的源或目标，而目标应采取的行动（例如，删除，接受等等）类型。

包过滤服务是使用 service 或 chkconfig 命令激活。放 IPTABLES_SAVE_ON_STOP 在文件/etc/sysconfig/iptables-config,保存过滤规则，然后必须重新启动，或通过运行下面的命令：

```
#/sbin/service iptables save
```

Netfilter 和 Iptables 可以跟踪运行的网络服务的连接状态，也可以使用该状态为基础的筛选策略。跟踪状态，这样允许配置代理建立连接，甚至本质上是无状态的协议（如 UDP）配置转发。例如，iptables 命令设置了一个已建立的连接规则来转发数据包：

```
# iptables -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT
```

Netfilter 和 Iptables 支持广泛的和灵活的防火墙配置。管理员可以配置 NAT 和 IP 伪装来保护系统控制路由到外部网络的端口转发通信。也可以设定规则的数据包记录在/ etc/ syslog.conf 文件中定义一个特定的日志文件。更多的资料，查看 Netfilter 和 iptables 相关的技术文档。

8 SSH

管理员通常使用的 Secure Shell（SSH）与其他系统进行加密的通讯。由于 SSH 是一个系统的入口点，如果是不需要的，它应该被禁用。如果需要，管理员可以通过编辑 /etc/ssh/sshd_config 加强连续的限制。限制 ssh 连接包括以下内容：

PubkeyAuthentication yes

PasswordAuthentication no

PermitRootLogin no

HostbasedAuthentication no

StrictModes yes

技术限制通过 SSH 远程访问配置一组或特定用户的访问。添加几行的/etc/ssh/ sshd_config 文件来允许或拒绝 ssh 访问：

- 分别添加一行 AllowUsers 或 AllowGroups 限制 ssh 到特定的用户或组。
- 分别添加一行 DenyUsers 或 DenyGroups 拒绝特定的用户或组。

其他一些有助于保护系统的设置导致的 ssh 客户端的时间不活动后自动退出：

ClientAliveInterval 600

ClientAliveCountMax 0

配置文件进行更改后，一定要重新启动该服务。

9 配置安装，文件权限和所有权

在安装 Linux 操作系统时，一些简单的步骤，就可以保护数据的完整性。首先，使用单独的磁盘分区的操作系统和用户数据（也就是单独的分区/home，/tmp，/var /tmp，/ ORACLE，等等）此策略可以防止出现“文件系统已满”的问题，影响系统运行。建立磁盘配额，也可以防止用户有意或无意地填补了文件系统。

为了防止操作系统文件和实用程序被篡改，mount /usr 文件系统为只读。当更新操作系统 RPM 包是，只需重新 mount /usr 文件作为读/写 使用-o remount ,rw 选项，（remount allows you to change mount flags without taking down the system）。执行更新后，不要忘记切换回只读模式。

要限制用户访问某些非 root 用户的本地文件系统（如/ tmp 或可移动的存储分区），设置使用 noexec，nosuid 和 nodev 安装选项。选项 noexec 防止执行的二进制文件（而不是脚本），nosuid 将防止 setuid 位生效，而 nodev 禁止使用设备文件。

POSIX 访问控制列表（ACL）提供了更丰富的访问控制模型比传统的 UNIX 自主访问控制（DAC）对应所有者，组和其他系统的用户设置读，写和执行权限。 ACL 可以定义不仅仅是一个单一的用户或组的访问权限，程序，进程，文件和目录的指定权。甚至可以在目录上设置一个默认的 ACL，其后代自动继承相同的权限。如何管理 ACL，请参见 setfacl 和 getfacl 命令的更多信息。内核提供了 ext3 和 NFS 导出的文件系统的 ACL 支持。

收紧文件的权限和检查所有权以尽量减少漏洞。检查通用可写目录和文件，以及任何不受控制的文件，如果这些这方面有问题，把相应的文件以及目录的权限尽量缩小。

Setuid 和 setgid 位有时设置为可执行的，使他们能够执行任务需要的其他权利，如 root 权限。然而 setuid 和 setgid 位可执行文件，通过使用权的开发，作为缓冲区溢出攻击。出于这个原因，要审查有 setuid 和 setgid 位可执行文件，下面一个例子用 find 命令显示 setuid 位权限的程序。

```
# find / \( -perm -4000 -o -perm -2000 \) -print
```

如果可执行文件的实际使用（这可能是许多实用程序，例如，/usr/bin 中/RCP/bin/ping6，等等，视系统而定），取出了 setuid 位从问题中的可执行文件：

```
# chmod -s /usr/bin/rcp
```

10 管理用户和权限

通常，一个简单的监督可以弥补一些巨大的安全漏洞。要定期检查系统未使用和未锁定的用户帐户，以及密码不安全的账户。确保没有非 root 用户帐户的用户 ID 为 0。

当安装软件，创建一个默认用户帐户和密码，一定要立即改变用户的默认密码。一个集中的用户身份验证方法(如 OpenLDAP 或其他 LDAP 实现)可以帮助简化用户认证和管理任务，这可能有助于降低风险对于未使用的帐户或帐户空密码。

明确用户的管理权限，不能直接以 root 身份登录。管理员应该在系统首先作为一个已命名的用户登录，然后使用 su 或 sudo 命令以 root 身份执行任务。为了防止用户以 root 身份登录，编辑/etc/passwd 文件，改变 shell /bin/bash 到/sbin/nologin。修改/etc/sudoers 文件中使用 visudo 来授予特定的用户权限来执行管理任务。

11 额外的安全功能和工具

Linux 高版本提供了更多的功能和工具，以增加内置的操作系统的安全控制。是否有必要执行这些功能依赖于安全要求，配置支持，以及和应用程序的兼容性。

SELinux 最初是由美国国家安全局开发,SELinux 添加了额外的安全层,超越基本的 UNIX 自主访问控制（DAC）机制。具体而言，SELinux 新增功能，支持强制访问控制（MAC）或基于角色的访问控制（RBAC）。**SELinux** 在类型强制服务器中合并了多级安全性或一种可选的多类策略，并采用了基于角色的访问控制概念。

SELinux 是一种基于 域-类型 模型（domain-type）的强制访问控制（MAC）安全系统，它由 NSA 编写并设计成内核模块包含到内核中，相应的某些安全相关的应用也被打了 SELinux 的补丁，最后还有一个相应的安全策略。

众所周知，标准的 UNIX 安全模型是"任意的访问控制"DAC。就是说，任何程序对其资源享有完全的控制权。假设某个程序打算把含有潜在重要信息的文件扔到/tmp 目录下，那么在 DAC 情况下没人能阻止他！

而 MAC 情况下的安全策略完全控制着对所有资源的访问。这是 MAC 和 DAC 本质的区别。SELinux 提供了比传统的 UNIX 权限更好的访问控制。

使用 sestatus 命令来显示是否运行 SELinux 是，当前的模式，和使用的策略：

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

```
Policy from config file:        targeted
```

在一些机密的环境中，站点的安全策略可能需要使用的 SELinux 毫升政策，提供严格的多级安全（MLS）保护。 MLS 配置通常需要特定的网站和安全，这需要大量的自定义的 MAC 标签。

12 最后的思考

系统的安全性远远超出只对操作系统本身的安全加固。它需要一个整体的安全架构和管理架构，包括一个定义良好的安全策略，系统化的管理程序，并定期进行安全评估，以确保系统和数据库的保密性，完整性和可用性。外部防火墙和入侵检测系统也有助于实现系统的安全性。

系统保护和操作系统加固是实现应用程序的可用性和数据保护的第一步。一般来说，Linux 服务器安装在默认安全的情况下，提供了一些安全选项，本文给系统管理员提供了一些额外的策略考虑。