银河麒麟高级服务器操作系统V10 SP3 2403

安全加固操作指南

麒麟软件有限公司

2024年2月

手册说明

麒麟V10 SP3 2403手册是对银河麒麟V10 SP3 2403服务器安全加固需求中184个加固项从级别、适用版本、检查方法、修复方法（加固方法，还原方法）、修改影响这5个方面做出的解释与说明。

加固项格式说明

加固项格式要求：

分类：用于表示当前加固项所属的大类（安全服务、内核参数、安全网络、系统命令、系统审计、系统设置、潜在风险、文件权限、风险账户、磁盘检查、密码强度、账户锁定、系统安全、系统维护、资源分配）；

标题：标题包含统一的编号 + 空格 + 加固项名

级别：表示当前条目所属的重要级别，当前分为两级：**"要求" 表示核心配置，通常情况下必须满足；"建议" 表示开启后对系统影响较大建议配置**，对于高安全要求的系统，建议满足；

适用版本：表示该条目适用哪些服务器版本，默认值是 "V10 SP3 2403"，可以是 "V10SP1、V10SP2、V10SP3" 等；

说明：用于说明该规则的要求、背景；

检查方法：用于表示在系统中通过什么方式检查该规则是否满足；

修改建议：加固方法用于表示在系统中通过什么方式修复、配置该规则，使之满足要求；还原方法用于表示在系统中通过什么方式将加固过后的系统恢复成默认设置；

修改影响：用于说明系统如果按照该规则进行配置，对系统或业务造成的影响，如果没有影响，则写"无"；

其他

安装软件包：使用系统默认yum源，通过yum install命令进行安装。

目录

1 安全服务

1.1 禁用不必要的系统服务

级别

建议

适用版本

V10 SP3 2403

说明

禁用不必要的系统服务：cups{cups.service, cups-lpd.socket}、sendmail{sendmail.service}、nfs{nfs.service,nfs-server.service}、ident{auth.socket}、ntalk{ntalk.socket,ntalk.service}、bootps{dhcpd.service}、ypbind{ypbind.service}、nfs-lock{nfs-lock.service}、tftp{tftp.service,tftp.socket}、rsync{rsyncd.service}。

检查方法

- 通过如下命令检查不必要的服务是否已禁用：

```
[root@localhost ~]# systemctl is-enabled <服务名>
disabled
[root@localhost ~]# systemctl is-active <服务名>
inactive
```

修改建议

加固方法

- 通过如下命令禁用不必要的服务：

```
[root@localhost ~]# systemctl disable <服务名>
[root@localhost ~]# systemctl stop <服务名>
```

还原方法

- 使用如下方法开启不必要的服务：

```
[root@localhost ~]# systemctl enable <服务名>
[root@localhost ~]# systemctl start <服务名>
```

修改影响

禁用不必要的系统服务可以减少攻击面，加固后无法使用如下服务：cups{cups.service, cups-lpd.socket}、sendmail{sendmail.service}、nfs{nfs.service, nfs-server.service}、ident{auth.socket}、ntalk{ntalk.socket,ntalk.service}、bootps{dhcpd.service}、ypbind{ypbind.service}、nfs-lock{nfs-lock.service}、tftp{tftp.service, tftp.socket}、rsync{rsyncd.service}，加固项功能如下表所示。

禁用服务的功能说明表

服务	功能
cups	CUPS服务支持本地打印机、网络打印机，并能够共享打印机。
sendmail	Sendmail服务是一种电子邮件传输代理程序，它允许用户在本地和远程主机之间发送和接收邮件。
nfs	NFS（Network File System）是一种分布式文件系统协议，它允许用户在客户端和服务器之间共享文件和目录。。
ident	Indent服务是一种认证机制，它允许用户通过网络协议（如SSH）访问服务器上的资源。
ntalk	Ntalk是一种基于文本的聊天服务，它允许用户通过网络协议（如TCP/IP）进行实时通信。
bootps	BOOTP是一种用于网络设备自动配置的协议，它允许无盘工作站从网络上获取IP地址、子网掩码、默认网关等信息。。
ypbind	ypbind（Yellow Pages Bind）是NIS（Network Information Service）的客户端守护进程。
nfs-lock	NFS-Lock服务是Network File System（NFS）的一部分，它负责在客户端和服务器之间协调文件锁定。
tftp	TFTP（Trivial File Transfer Protocol）是一种简单的文件传输协议，它允许用户在客户端和服务器之间上传或下载文件。
rsync	RSync服务是一种文件同步工具，它允许用户在本地和远程主机之间进行快速、安全的数据备份和迁移。

1.2 禁止匿名VSFTP用户登录

级别

要求

适用版本

V10 SP3 2403

说明

取消对匿名登录的支持，即不允许无账号密码的访客访问。

> **说明：**
> 为了保护您的数据和网络资源免受非法访问和破坏，请考虑禁用 VSFTPD 的匿名登录功能。

检查方法

- 检查/etc/vsftpd.conf文件中参数"anonymous_enable"是否为"NO"：

```
[root@localhost ~]# cat /etc/vsftpd/vsftpd.conf |grep anonymous_enable
anonymous_enable=NO
```

修改建议

加固方法

- 编辑/etc/vsftpd/vsftpd.conf文件找到"anonymous_enable"所在行，在行首添加"#"进行注释，在注释行之后添加以下内容：

```
anonymous_enable=NO
```

还原方法

- 编辑/etc/vsftpd/vsftpd.conf文件把加固时新增的行全部删除，再去掉查找到的"anonymous_enable"所在行的行首的"#"。

修改影响

设置anonymous_enable=NO后，无法通过匿名方式登录VSFTP。

1.3 禁止root登录VSFTP

级别

要求

适用版本

V10 SP3 2403

说明

检查是否禁止root用户登录VSFTP。

> **说明：**
> 为了保护您的数据和网络资源免受非法访问和破坏，请考虑禁止 root 用户登录 FTP。

检查方法

- 通过如下命令验证是否已禁止root登录VSFTP：

```
[root@localhost ~]# cat /etc/vsftpd/ftpusers|grep -w root
root
```

```
[root@localhost ~]# cat /etc/vsftpd/user_list|grep -w root
root
```

修改建议

加固方法

- 通过如下命令禁止root登录VSFTP：

```
[root@localhost ~]# echo "root" >> /etc/vsftpd/ftpusers
```

还原方法

- 修改/etc/vsftpd/ftpusers文件，删除添加的root。

修改影响

root用户将无法通过VSFTP登录到服务器。

1.4 设置ssh登录前警告Banner

级别

要求

适用版本

V10 SP3 2403

说明

检查系统openssh安全配置，设置ssh登录前警告Banner。

检查方法

- 通过如下命令检查/etc/ssh/sshd_config文件中Banner后路径是否为/etc/sshd.net：

```
[root@localhost ~]# grep "^Banner" /etc/ssh/sshd_config
Banner /etc/sshd.net
```

- 检查/etc/sshd.net文件中是否包含如下行：

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

修改建议

加固方法

- 编辑/etc/ssh/sshd_config文件，在Banner行首添加"#"进行注释，在注释行之后添加以下内容：

```
Banner /etc/sshd.net
```

- 通过如下命令创建/etc/sshd.net文件并设置相关警告信息：

```
[root@localhost ~]# touch  /etc/sshd.net
[root@localhost ~]# chmod 644 /etc/sshd.net
[root@localhost ~]# echo  "Authorized users only. All activities may be monitored and reported." > /etc/sshd.net
```

还原方法

- 编辑/etc/ssh/sshd_config文件将加固时新增的行删除，再去掉Banner行的注释，删除文件/etc/sshd.net：

```
[root@localhost ~]# rm -rf /etc/sshd.net
```

修改影响

虽然Banner不会直接阻止未经授权的访问，但它是一种很好的预防措施，有助于提高整个系统的安全性。

1.5 设置ssh成功登录后Banner信息

级别

要求

适用版本

V10 SP3 2403

说明

检查系统openssh安全配置，设置ssh登录后警告Banner。

检查方法

- 通过如下命令验证SSH服务是否应用pam_motd.so模块：

```
[root@localhost ~]# grep "pam_motd.so" /etc/pam.d/sshd|grep "session"|grep "optional"
session    optional     pam_motd.so
```

- 通过如下命令检查/etc/motd文件是否包含如下内容:

```
[root@localhost ~]# cat /etc/motd
Login success. All activity will be monitored and reported.
```

修改建议

加固方法

- 编辑/etc/pam.d/sshd文件，添加如下行确保ssh应用pam_motd.so模块：

```
session optional pam_motd.so
```

- 通过如下命令创建/etc/motd文件并设置相关警告信息：

```
[root@localhost ~]# touch /etc/motd
[root@localhost ~]# chmod 644 /etc/motd
[root@localhost ~]# echo "Login success. All activity will be monitored and reported." > /etc/motd
```

还原方法

- 编辑/etc/pam.d/sshd文件将加固时新增的行删除：

```
session optional pam_motd.so
```

- 编辑/etc/motd文件删除添加的信息：

```
Login success. All activity will be monitored and reported.
```

修改影响

虽然Banner不会直接阻止未经授权的访问，但它是一种很好的预防措施，有助于提高整个系统的安全性。

1.6 禁止root用户登录SSH

级别

要求

适用版本

V10 SP3 2403

说明

检查系统openssh安全配置，禁止root用户通过ssh登录。

检查方法

- 通过如下命令验证是否已配置ssh禁止root登录：

```
[root@localhost ~]# grep "^PermitRootLogin" /etc/ssh/sshd_config
PermitRootLogin no
```

修改建议

加固方法

- 编辑/etc/ssh/sshd_config文件找到PermitRootLogin yes所在行，在行首添加"#"进行注释，在注释行之后添加以下内容：

```
PermitRootLogin no
```

- 然后重启ssh服务：

```
[root@localhost ~]# systemctl restart sshd
```

还原方法

- 编辑/etc/ssh/sshd_config文件将加固时新增的行删除，再去掉查找到的PermitRootLogin所在行的行首的"#"，重启ssh服务。

```
[root@localhost ~]# systemctl restart sshd
```

修改影响

root用户无法通过ssh服务远程登录本服务器。

1.7 设置SSH安全协议

级别

要求

适用版本

V10 SP3 2403

说明

检查系统openssh安全配置，ssh服务使用安全的协议版本。

> **说明** Protocol参数控制SSH服务支持的协议版本。在SSH中，有两种主要的协议版本：SSH 1和SSH 2，分别是指Secure Shell协议的第一版和第二版，它们各自遵循不同的标准：
>
> SSH1： SSH1是最初的版本，主要由Tatu Ylönen在1995年开发。它并未严格遵循IETF（Internet Engineering Task Force）的标准制定流程，但它是最早被广泛采用的SSH协议版本。SSH1存在多个子版本，如1.3和1.5等。
>
> SSH2： SSH2是对SSH1的重大改进和重新设计，它是一个更加安全且功能更丰富的版本。SSH2通过一系列RFC文档进行了标准化，这些RFC包括但不限于以下几项：
>
> RFC 4250：定义了SSH协议的整体框架。 RFC 4251：描述了SSH协议的消息格式、数据类型以及通用约定。 RFC 4252：详细说明了SSH的身份验证协议，包括公钥认证机制。 RFC 4253：定义了SSH传输层协议，用于加密和压缩会话数据。 RFC 4254：规定了SSH连接协议，涉及通道、会话和远程命令执行等方面。 RFC 4255：定义了SSH公钥指纹格式。 它们在安全性和功能上有显著的区别：
>
> 安全性改进：
>
> SSH1存在一些已知的安全漏洞，尤其是在其加密和认证机制上。SSH1使用较弱的加密算法，并且在密钥交换和会话完整性保护方面不如SSH2成熟。 SSH2引入了更强大的加密算法，如AES（高级加密标准）替代了SSH1中的DES、3DES等，并且改进了密钥交换协议，提高了整体安全性。 SSH2不再使用CRC校验码，而是采用更为安全的消息认证码（MACs）来确保数据的完整性和防篡改。
>
> 兼容性与标准化：
>
> SSH2在设计时考虑了向后不兼容SSH1的问题，但为了增强安全性做出了这种权衡。这意味着SSH2服务器通常不会接受SSH1客户端连接，反之亦然。 SSH2被更加广泛地采纳并成为正式的IETF标准（RFC 4250至RFC 4256），而SSH1则没有得到这样的标准化。 功能扩展：
>
> SSH2提供了更多的网络服务支持，例如SFTP（安全文件传输协议）得到了增强，SCP也更加稳定可靠。 SSH2支持多通道功能，可以在单个TCP连接上同时进行多个会话，比如同时打开多个SFTP会话或远程shell窗口。 鉴于上述差异，自SSH2发布以来，业界普遍推荐并优先使用SSH2，许多系统和服务都已经不再支持SSH1以避免潜在的安全风险。

检查方法

- 通过如下命令验证是否使用安全的协议版本：

```
[root@localhost ~]# grep "^Protocol" /etc/ssh/sshd_config
Protocol 2
```

修改建议

加固方法

- 编辑/etc/ssh/sshd_config文件找到Protocol所在行，在行首添加"#"进行注释，在注释行之后添加以下内容：

```
Protocol 2
```

- 然后重启ssh服务：

```
[root@localhost ~]# systemctl restart sshd
```

还原方法

- 编辑/etc/ssh/sshd_config文件把加固时新增的行删除，再去掉查找到的Protocol所在行的行首的"#"，然后重启:

```
[root@localhost ~]# systemctl restart sshd
```

修改影响

可以提高系统的安全性，并且与现代的安全标准保持一致,但不兼容旧版本。

1.8 设置SSH日志等级

级别

要求

适用版本

V10 SP3 2403

说明

检查系统openssh安全配置，设置ssh日志等级。

> **说明** LogLevel控制在系统日志中记录的SSH服务事件的详细程度，设置为INFO会记录大部分重要的事件，但不会生成过于冗长的日志。

检查方法

- 通过如下命令验证是否配置安全的SSH日志等级：

```
[root@localhost ~]# grep "^LogLevel" /etc/ssh/sshd_config
LogLevel INFO
```

修改建议

加固方法

- 编辑/etc/ssh/sshd_config文件找到LogLevel所在行，在行首添加"#"进行注释，在注释行之后添加以下内容:

```
LogLevel INFO
```

- 然后重启ssh服务：

```
[root@localhost ~]# systemctl restart sshd
```

还原方法

- 编辑/etc/ssh/sshd_config文件把加固时新增的行删除，再去掉查找到的LogLevel所在行的行首的"#"，然后重启：

```
[root@localhost ~]# systemctl restart sshd
```

修改影响

记录一般性的信息，包括成功的登录。

1.9 设置SSH失败尝试次数

级别

要求

适用版本

V10 SP3 2403

说明

检查系统openssh安全配置，设置SSH失败尝试次数。

> **说明** MaxAuthTries控制在断开连接之前客户端尝试身份验证的最大次数。

检查方法

- 通过如下命令验证SSH失败尝试次数是否小于或等于4：

```
[root@localhost ~]# grep "^MaxAuthTries" /etc/ssh/sshd_config
MaxAuthTries 4
```

修改建议

加固方法

- 编辑/etc/ssh/sshd_config文件找到MaxAuthTries所在行，在行首添加"#"进行注释，在注释行之后添加以下内容:

```
MaxAuthTries 4
```

- 然后重启ssh服务：

```
[root@localhost ~]# systemctl restart sshd
```

还原方法

- 编辑/etc/ssh/sshd_config文件把加固时新增的行删除，再去掉查找到的MaxAuthTries所在行的行首的"#"，然后重启：

```
[root@localhost ~]# systemctl restart sshd
```

修改影响

如果设置为4，那么在4次尝试失败后，SSH服务会断开与客户端的连接，从而减少暴力破解密码的可能性。

1.10 禁用SSH空密码用户登录

级别

要求

适用版本

V10 SP3 2403

说明

允许空密码意味着任何人都可以无需身份验证就访问你的系统，这会极大地增加被恶意攻击的风险，麒麟建议关闭ssh空密码登录。

检查方法

- 通过如下命令验证是否禁止空密码用户通过SSH登录：

```
[root@localhost ~]# grep "^PermitEmptyPasswords" /etc/ssh/sshd_config
PermitEmptyPasswords no。
```

修改建议

加固方法

- 编辑/etc/ssh/sshd_config文件找到PermitEmptyPasswords所在行，在行首添加"#"进行注释，在注释行之后添加以下内容:

```
PermitEmptyPasswords no
```

- 然后重启ssh服务：

```
[root@localhost ~]# systemctl restart sshd
```

还原方法

- 编辑/etc/ssh/sshd_config文件把加固时新增的行删除，再去掉查找到的PermitEmptyPasswords所在行的行首的"#"，然后重启：

```
[root@localhost ~]# systemctl restart sshd
```

修改影响

不允许用户使用空密码进行登录。

1.11 禁用SSH环境处理

级别

要求

适用版本

V10 SP3 2403

说明

检查系统openssh安全配置，禁用SSH环境处理。

> **说明** 启用PermitUserEnvironment可能会带来一些安全风险，因为攻击者可能尝试通过修改用户的~/.ssh/environment文件来注入恶意环境变量。

检查方法

- 通过如下命令验证SSH环境处理是否禁用：

```
[root@localhost ~]# grep PermitUserEnvironment /etc/ssh/sshd_config
PermitUserEnvironment no
```

修改建议

加固方法

- 编辑/etc/ssh/sshd_config文件找到PermitUserEnvironment所在行，在行首添加"#"进行注释，在注释行之后添加以下内容:

```
PermitUserEnvironment no
```

- 然后重启ssh服务：

```
[root@localhost ~]# systemctl restart sshd
```

还原方法

- 编辑/etc/ssh/sshd_config文件把加固时新增的行全部删除，再去掉查找到的行的PermitUserEnvironment所在行首的"#"，然后重启：

```
[root@localhost ~]# systemctl restart sshd
```

修改影响

用户在~/.ssh/environment文件中定义的环境变量不会生效。

1.12 开启SSH强加密算法

级别

要求

适用版本

V10 SP3 2403

说明

检查系统openssh安全配置，启用强加密算法。

> **说明** 以下是一些公认的弱加密算法： Arcfour，也称为 RC4，是由Ron Rivest在1987年设计的一种流密码算法。 MD5：尽管MD5在哈希领域广泛使用，但由于其碰撞可能性较大，已被认为不适合用于密码哈希或任何需要保证数据完整性和唯一性的场景。 3DES（Triple DES/TDEA）：虽然比DES有所改进，通过三次使用DES算法增强了安全性，但仍因其较低的密钥效率和相对较短的有效密钥长度（实质上是168位，但因为重复使用部分密钥，实际强度低于理论值）而不被视为理想的加密手段。 RC4：一种流密码，曾广泛应用于SSL/TLS协议中，但由于多种安全问题，现在已经废弃不用。 现代加密实践中，推荐使用高级加密标准（AES）以及其他经过时间和攻击考验的安全算法，如RSA、ECC（椭圆曲线加密）、SHA-2或SHA-3系列散列函数等。此外，对于密钥交换，TLS协议中常用的是DH（Diffie-Hellman）或ECDH（Elliptic Curve Diffie-Hellman）算法。

检查方法

- 通过如下命令检查系统加密算法中不包含弱加密算法arcfour、3des、md5、rc4：

```
[root@localhost ~]# cat /etc/ssh/sshd_config |grep -v '^#' |grep Ciphers
Ciphers aes128-ctr,aes192-ctr,aes256-ctr,aes128-gcm@openssh.com,aes256-gcm@openssh.com,chacha20-poly1305@openssh.com
```

修改建议

加固方法

- 编辑/etc/ssh/sshd_config文件Ciphers所在行，在行首添加"#"进行注释，在注释行之后添加以下内容：

```
Ciphers aes128-ctr,aes192-ctr,aes256-ctr,aes128-cbc
```

- 然后重启ssh服务：

```
[root@localhost ~]# systemctl restart sshd
```

还原方法

- 编辑/etc/ssh/sshd_config文件把以下加固时新增的行全部删除，再去掉查找到的Ciphers所在行的行首的"#"：

```
Ciphers aes128-ctr,aes192-ctr,aes256-ctr,aes128-cbc
```

- 然后重启ssh服务：

```
[root@localhost ~]# systemctl restart sshd
```

修改影响

强加密算法通常具有更高的加密强度和更复杂的加密过程，这使得数据在传输过程中更难被破解。可以提高SSH通信的安全性，并保持与现代安全标准的兼容性。这对于保护敏感信息，如用户凭据、命令执行结果和其他关键数据至关重要。

1.13 检查是否安装chkrootkit

级别

建议

适用版本

V10 SP3 2403

说明

chkrootkit 是一款强大而又易用的工具，可以帮助您加强网络安全，并保持系统健康稳定。麒麟建议安装软件。以系统环境**chkrootkit-0.55-4.ky10.x86_64**为例。

检查方法

- 通过如下命令验证是否已安装chkrootkit软件：

```
[root@localhost ~]# rpm -q chkrootkit
chkrootkit-0.55-4.ky10.x86_64
```

修改建议

加固方法

- 配置系统运行的网络环境，如未安装则提醒用户安装，通过如下命令安装chkrootkit软件：

```
[root@localhost ~]# yum install chkrootkit
```

还原方法

- 卸载chkrootkit软件:

```
[root@localhost ~]# rpm -e chkrootkit
```

修改影响

chkrootkit 可以有效地查找 rootkits和其他恶意软件，并提供详细的报告，提高系统安全性。但会影响主机性能（较少发生、不会产生严重影响），需要定期更新版本以保持最新的病毒库，在不同平台上可能需要调整配置文件或脚本。所以在使用 chkrootkit 前，请务必了解其兼容性和适用范围。

1.14 检查是否存在rootkit程序

级别

建议

适用版本

V10 SP3 2403

说明

通过chkrootkit工具进行系统监测。

> **说明：** 使用 -p 开关，只进行进程扫描部分。

检查方法

- 通过如下命令检测是否存在疑似感染的程序，存在则需要进行加固：

```
[root@localhost ~]# chkrootkit -p /home:/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin |grep '^/'
```

修改建议

加固方法

- 对检测出的异常文件通常需要根据具体情况采取不同的行动。以下是一些通用步骤，但请注意，具体措施应根据**文件的性质**、**上下文**和**安全策略**来决定：
    - 确认文件来源： 确认文件何时、何地以及为何被创建或修改，如果是系统或已知程序正常生成的文件，比如 .vmlinuz.hmac 这样的内核哈希文件，那么可以忽略。 查看文件的权限、属主和属组，以及修改时间，对比系统日志和其他相关记录，判断文件是否合理。
    - 分析文件内容： 对于文本文件，可以打开并阅读内容以初步判断其用途和性质，比如 /etc/passwd文件检测异常，需查看是具体哪些内容引起的，修改异常内容或重新安装该文件的安装包。 对于二进制文件，可以使用 file 命令来查看文件类型，或者使用反病毒软件进行扫描。
    - 检查文件关联： 检查文件是否与系统关键进程或服务有关联，以及这些关联是否合法和预期。
    - 隔离和备份: 在不确定文件是否安全的情况下，可以先将其备份，以防删除重要文件导致问题。如果怀疑是恶意文件，可以选择隔离或移至安全位置以断开其与系统的连接。通过以下命令备份：

      ```
      [root@localhost ~]# cp /etc/passwd /etc/passwd.bak
      ```

    - 删除或修复: 若确认文件为恶意文件或垃圾文件，可以安全地删除它。 如果文件是系统文件但损坏或被篡改，可以尝试用正确的文件替换它，或者根据错误日志和系统指南进行修复。
    - 跟进调查: 如果发现恶意文件，应对整个系统进行全面的安全审计，查找是否有其他受感染的地方，并加强系统安全防御措施。 报告给安全团队或专业人员，他们可以帮助分析攻击源和攻击手段，以便采取更针对性的防范措施。
    - 更新和补丁: 应确保系统和应用程序的最新更新已安装，以修复可能导致异常文件出现的安全漏洞。

还原方法

- 如果加固时对异常文件进行备份处理可通过如下命令进行恢复：

```
[root@localhost ~]# cp /etc/passwd.bak /etc/passwd
```

修改影响

- 经过上述处理异常文件的过程，可能会产生以下影响：
    - 安全性提升： 删除恶意文件可以消除潜在的安全威胁，阻止恶意代码执行，避免数据泄露或系统瘫痪。 安全审计和加强防护措施有助于发现系统弱点并加以修补，提高整体的安全性。
    - 系统稳定性增强： 修复或替换损坏的系统文件有助于恢复系统的正常运行，减少因文件异常导致的各种故障和错误。 更新和打补丁可以解决已知漏洞，保持系统稳定性和兼容性。
    - 资源占用降低： 清理无效或垃圾文件可以释放存储空间，减轻磁盘压力，提高系统性能。

- 业务连续性得到保障： 及时识别和处理异常文件减少了意外宕机的风险，保证了业务连续性及用户访问体验。
- 然而，在处理过程中也可能存在潜在的影响： 不当的操作可能误删重要文件，造成业务中断或其他不可预见的问题，因此在删除前务必做好备份和评估工作。 频繁更改系统设置和文件结构可能会影响其他依赖于该文件的应用程序或服务，需要在变更前后做好测试和验证。 综上所述，正确且谨慎地处理异常文件能够显著提高系统安全性与稳定性，但也需要注意操作风险，尽量减小对正常业务流程的干扰。

1.15 限制FTP用户登录后能访问的目录

级别

要求

适用版本

V10 SP3 2403

说明

检查并配置配置ftp服务器用户访问权限。

检查方法

- 检查/etc/vsftpd/vsftpd.conf文件是否配置FTP用户登录后能访问的目录：

```
[root@localhost ~]# cat /etc/vsftpd/vsftpd.conf |grep chroot_local_user
chroot_local_user=YES
```

```
[root@localhost ~]# cat /etc/vsftpd/vsftpd.conf |grep allow_writeable_chroot
allow_writeable_chroot = YES
```

修改建议

加固方法

- 编辑/etc/vsftpd/vsftpd.conf文件找到配置项所在行，把查找到的行在行首添加"#"进行注释，在注释行之后添加以下内容:

```
chroot_local_user=YES
allow_writeable_chroot = YES
```

- 如需要写权限需要再配置：

```
write_enable=YES
```

- 该项重启vsftp服务生效：

```
[root@localhost ~]# systemctl restart vsftp
```

还原方法

- 在/etc/vsftpd/vsftpd.conf文件中，找到加固配置项所在行，把查找到的行在行首前"#"进行删除，把加固时新增的内容删除，编辑文件后，该项重启vsftp服务生效：

```
[root@localhost ~]# systemctl restart vsftp
```

修改影响

当 chroot_local_user=YES 时，FTP用户在登录后将被限制在其宿主目录下，无法访问上级目录。这意味着用户只能看到和操作他们在系统中的个人 home 目录及其子目录，不能导航到文件系统的其他部分。同时，allow_writeable_chroot=YES 允许用户在他们的 chroot 目录中有写权限。

权限参数功能说明表

权限参数	功能
chroot_local_user	是否将本地用户限制在他们的主目录下。
allow_writeable_chroot	是否允许使用可写的chroot环境。

权限参数	功能
write_enable	是否启用FTP服务的写入功能。

1.16 开启SSH服务自启动

级别

要求

适用版本

V10 SP3 2403

说明

对于使用IP协议进行远程维护的设备,应配置使用SSH协议。

检查方法

- 通过如下命令验证ssh服务是否已开启：

```
[root@localhost ~]# systemctl is-enabled sshd.service
enable
[root@localhost ~]# systemctl is-active  sshd.service
active
```

修改建议

加固方法

- 通过如下命令开启ssh服务：

```
[root@localhost ~]# systemctl enable sshd.service
[root@localhost ~]# systemctl start sshd.service
```

还原方法

- 通过如下命令关闭ssh服务：

```
[root@localhost ~]# systemctl disable sshd.service
[root@localhost ~]# systemctl stop sshd.service
```

修改影响

启动sshd.service意味着系统开启了Secure Shell (SSH)服务，允许其他用户或管理员通过网络使用加密的连接方式登录到这台服务器。这意味着你可以在本地主机或其他任何地方通过SSH客户端软件（如PuTTY、OpenSSH等）远程执行命令、传输文件以及进行各种管理操作。系统会运行sshd守护进程，它将持续监听指定的端口（默认为22），等待并处理来自客户端的连接请求。这将占用一定的系统资源，包括CPU、内存和网络带宽，尤其是在高并发连接场景下。

1.17 禁止远程登录telnet服务

级别

要求

适用版本

V10 SP3 2403

说明

禁用telnet服务通常是为了增强系统的安全性，因为telnet协议以明文方式传输用户名和密码，存在较大的安全风险。

检查方法

- 通过如下命令验证telnet服务是否已关闭：

```
[root@localhost ~]# systemctl is-enabled telnet.service
enable
[root@localhost ~]# systemctl is-active  telnet.service
active
```

修改建议

加固方法

- 通过如下命令关闭telnet服务：

```
[root@localhost ~]# systemctl disable telnet.service
[root@localhost ~]# systemctl stop telnet.service
```

还原方法

- 通过如下命令开启telnet服务：

```
[root@localhost ~]# systemctl enable telnet.service
[root@localhost ~]# systemctl start telnet.service
```

修改影响

关闭Telnet服务器以阻止远程连接到此计算机的Telnet服务。

1.18 禁止root用户远程telnet登录

级别

要求

适用版本

V10 SP3 2403

说明

禁止root用户通过 Telnet 登录是一种有效的安全策略，可以帮助您提高系统的安全性、加强审计和控制以及改善性能。

检查方法

- 通过如下命令验证是否配置pam_succeed_if.so模块：

```
[root@localhost ~]# cat /etc/pam.d/remote |grep pam_succeed_if |grep requisite |grep root
auth requisite  pam_succeed_if.so   user!=root
```

加固方法

- 编辑/etc/pam.d/remote文件添加如下行：

```
auth requisite  pam_succeed_if.so   user!=root
```

还原方法

- 编辑/etc/pam.d/remote文件 删除加固时添加的内容：

```
auth requisite  pam_succeed_if.so   user!=root
```

修改影响

禁止root用户通过telnet服务登录。

1.19 设置telnet登录前警告Banner

级别

要求

适用版本

V10 SP3 2403

说明

设置 telnet登录前警告 Banner 可以有效地增强网络安全性，并让用户更加明确地了解自己的权限和责任。

检查方法

- 检查/etc/issue.net文件中是否已包含如下行：

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

修改建议

加固方法

- 编辑/etc/issue.net文件并添加如下警告信息：

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

还原方法

- 把加固时新增的行删除

修改影响

虽然Banner不会直接阻止未经授权的访问，但它是一种很好的预防措施，有助于提高整个系统的安全性。

1.20 设置telnet登录后警告Banner

级别

要求

适用版本

V10 SP3 2403

说明

检查是否设置telnet登录后警告Banner。

检查方法

- 检查/etc/motd文件是否包含如下行：

```
Login success. All activity will be monitored and reported.
```

修改建议

加固方法

- 编辑/etc/motd文件并添加如下警告信息：

```
Login success. All activity will be monitored and reported.
```

还原方法

- 编辑/etc/motd文件删除加固时添加的信息：

```
Login success. All activity will be monitored and reported.
```

修改影响

虽然Banner不会直接阻止未经授权的访问，但它是一种很好的预防措施，有助于提高整个系统的安全性。

1.21 禁用不必要的xinetd服务

级别

建议

说明

禁用不必要的xinetd服务：chargen-dgram、chargen-stream、daytime-stream、daytime-dgram、eklogin、echo-stream、echo-dgram、tcpmux-server、discard-dgram、discard-stream、klogin、krb5-telnet、ekrb5-telnet、cvs、kshell、time-dgram、time-stream、lpd、gssftp。上述不必要的xinetd服务存在安全漏洞有被Dos 攻击的风险，建议关闭。

检查方法

- 通过如下命令验证不必要的服务是否已关闭：

```
[root@localhost ~]#chkconfig --list

    运行结果（以部分 xinetd 的服务为例）：
        chargen-dgram:  关
        chargen-stream: 关
        cvs:            关
        daytime-dgram:  关
        daytime-stream: 关
        discard-dgram:  关
        discard-stream: 关
        echo-dgram:     关
        echo-stream:    关
        tcpmux-server:  关
        time-dgram:     关
        time-stream:    关
```

修改建议

加固方法

- 通过如下命令禁用chargen-dgram、chargen-stream、daytime-dgram、daytime-stream、eklogin、echo-dgram、echo-stream、tcpmux-server、discard-dgram、discard-stream、klogin、krb5-telnet、ekrb5-telnet、cvs、kshell、time-dgram、time-stream、lpd、gssftp服务：

```
[root@localhost ~]# chkconfig <服务名>  off
```

还原方法

- 通过如下命令将加固时设置为off的服务启用：

```
[root@localhost ~]# chkconfig <服务名>  on
```

修改影响

加固后无法使用所示的服务：chargen-dgram、chargen-stream、daytime-dgram、daytime-stream、eklogin、echo-dgram、echo-stream、tcpmux-server、discard-dgram、discard-stream、klogin、krb5-telnet、ekrb5-telnet、cvs、kshell、time-dgram、time-stream、lpd、gssftp。 **禁用服务的功能说明表**

服务	功能
chargen-dgram	产生随机字符序列的网络服务。
chargen-stream	连续发送生成的随机字符序列的网络服务。
daytime-dgram	返回当前日期时间的服务，使用 UDP 协议。
daytime-stream	返回当前日期时间的服务，使用 TCP 协议。
eklogin	eklogin 服务通常是指在 AIX 系统（IBM 的 Unix 操作系统）中的一个组件，它是 AIX Enhanced Kerberos 登录 (EKA) 功能的一部分。EKA 提供了基于 Kerberos v5 协议的身份验证和安全远程登录功能。
echo-dgram	回显客户端发来的消息的服务，使用 UDP 协议。
echo-stream	回显客户端发来的消息的服务，使用 TCP 协议。

服务	功能
tcpmux-server	TCP 服务多路复用器，为多个不同的服务提供统一端口号。
discard-dgram	丢弃接收到的数据包的服务，使用 UDP 协议。
discard-stream	丢弃接收到的数据包的服务，使用 TCP 协议。
klogin	远程登录 Kerberos V5 安全 shell。
krb5-telnet	支持 Kerberos V5 的 Telnet 协议。
ekrb5-telnet	加密的 Kerberos V5 Telnet 协议，旨在提供更安全的数据传输。
cvs	CVS (Concurrent Versions System) 有一些潜在的风险，比如容易遭受拒绝服务攻击和非授权访问等问题
kshell	kshell服务是指KShell守护进程。
time-dgram	返回当前日期时间的服务，使用 UDP 协议。
time-stream	返回当前日期时间的服务，使用 TCP 协议。
lpd	一种网络打印协议。
gssftp	基于 Kerberos V5 的 FTP 协议，支持文件传输。

1.22 关闭系统不必要的端口

级别

建议

适用版本

V10 SP3 2403

说明

关闭高危端口。以22端口为例。

检查方法

- 通过如下命令检查所有端口是否已关闭TCP传输:

```
[root@localhost ~]# firewall-cmd --query-port=22/tcp
no
```

- 通过如下命令检查所有端口是否已关闭UDP传输:

```
[root@localhost ~]firewall-cmd --query-port=22/udp
no
```

- 通过如下命令检查所有端口是否已关闭SCTP传输:

```
[root@localhost ~]# firewall-cmd --query-port=22/sctp
no
```

修改方法

加固方法

- 通过如下命令禁止端口TCP传输:

```
[root@localhost ~]# firewall-cmd --remove-port=22/tcp
success
```

- 通过如下命令禁止端口UDP传输：

```
[root@localhost ~]# firewall-cmd --remove-port=22/udp
success
```

- 通过如下命令禁止端口SCTP传输：

```
[root@localhost ~]# firewall-cmd --remove-port=22/sctp
success
```

- 通过如下命令重新加载设置：

```
[root@localhost ~]# firewall-cmd --runtime-to-permanent
success
```

还原方法

- 通过如下命令恢复端口可以进行TCP传输:

```
[root@localhost ~]# firewall-cmd --add-port=22/tcp
success
```

- 通过如下命令恢复端口可以进行UDP传输：

```
[root@localhost ~]# firewall-cmd --add-port=22/udp
success
```

- 通过如下命令恢复端口可以进行SCTP传输：

```
[root@localhost ~]# firewall-cmd --add-port=22/sctp
success
```

- 通过如下命令重新加载设置：

```
[root@localhost ~]# firewall-cmd --runtime-to-permanent
success
```

修改影响

将从防火墙规则中删除对高危端口的访问权限。

1.23 限制SSH服务可访问源

级别

建议

适用版本

V10 SP3 2403

说明

通过pam.d模块实现对ssh访问源的限制通，可以使 SSH 服务只能从指定的 IP 网段访问，示例以**192.168.201.0**网段为例。

检查方法

- 通过如下命令检测ssh服务是否已启用pam.d模块：

```
[root@localhost ~]# cat /etc/pam.d/sshd |grep account |grep required |grep pam_access.so
account required pam_access.so
```

- 通过如下命令检查文件/etc/security/access.conf中是否已配置允许192.168.201.0/24网段访问：

```
[root@localhost ~]# cat /etc/security/access.conf |grep +|grep ALL |grep "192.168.201.0/24"
+:ALL:192.168.201.0/24
```

- 通过如下命令检查文件/etc/security/access.conf中内容是否已配置禁止所有网段访问：

```
[root@localhost ~]# grep ':\bALL:ALL\b' /etc/security/access.conf
-:ALL:ALL
```

- 检查/etc/security/access.conf文件中-:ALL:ALL所在行要在所有+:ALL:ip段/24所在行的后面。

修改建议

加固方法

- 编辑/etc/pam.d/sshd文件添加如下内容：

```
account required pam_access.so
```

- 编辑/etc/security/access.conf文件添加内容，-:ALL:ALL要在所有+:ALL:ip段/24行之后：

```
+:ALL:192.168.201.0/24
-:ALL:-ALL
```

还原方法

- 编辑/etc/pam.d/sshd文件删除加固时添加的内容：

```
account required pam_access.so
```

- 编辑/etc/security/access.conf文件删除加固时添加的内容：

```
+:ALL:192.168.201.0/24
-:ALL:-ALL
```

修改影响

限制 SSH 服务仅允许特定ip段地址访问，可以有效地阻止未经授权的访问，提高了系统安全性。

1.24 设置登录后系统提示信息

级别

要求

适用版本

V10 SP3 2403

说明

检查系统openssh安全配置，用户成功登录时显示上次登录的信息。

检查方法

- 通过如下命令验证是否设置登录后系统提示信息：

```
[root@localhost ~]# grep "^PrintLastLog" /etc/ssh/sshd_config
PrintLastLog no
```

修改建议

加固方法

- 编辑/etc/ssh/sshd_config文件，找到PrintLastLog no所在行，在行首添加"#"进行注释，在注释行之后添加以下内容：

```
[root@localhost ~]PrintLastLog yes
```

- 然后重启ssh服务：

```
[root@localhost ~]# systemctl restart sshd
```

还原方法

把加固时新增的行全部删除，再去掉查找到的PrintLastLog no所在行的行首的"#"，重启ssh服务。

```
[root@localhost ~]# systemctl restart sshd
```

修改影响

当用户登录时，ssh服务器会显示他们的上一次登录时间和 IP 地址。

2 内核参数

2.1 禁止icmp重定向报文

级别

要求

适用版本

V10 SP3 2403

说明

ICMP重定向消息是传递路由信息并告诉系统通过备用路径发送数据包。这是一种允许外部路由设备更新系统路由表的方法。通过将net.ipv4.conf.all.accept_redirects和net.ipv6.conf.all.accept_redirects设置为0，系统不会接受任何ICMP重定向报文。通过将net.ipv4.conf.all.secure_redirects和net.ipv4.conf.default.send_redirects设置为0，系统不会从网关接收ICMP重定向报文（IPv6无此配置项）。

检查方法

- 通过如下命令验证是否禁止icmp重定向：

```
[root@localhost ~]# sysctl net.ipv4.conf.all.accept_redirects
net.ipv4.conf.all.accept_redirects = 0
```

```
[root@localhost ~]# sysctl net.ipv4.conf.default.accept_redirects
net.ipv4.conf.default.accept_redirects = 0
```

```
[root@localhost ~]# grep "net\.ipv4\.conf\.all\.accept_redirects" /etc/sysctl.conf
net.ipv4.conf.all.accept_redirects= 0
```

```
[root@localhost ~]# grep "net\.ipv4\.conf\.default\.accept_redirects" /etc/sysctl.conf
net.ipv4.conf.default.accept_redirects = 0
```

修改建议

加固方法

- 编辑/etc/sysctl.conf文件把查找到的行在行首添加"#"进行注释，在注释行之后添加以下内容：

```
net.ipv4.conf.all.accept_redirects = 0
net.ipv4.conf.default.accept_redirects = 0
```

- 通过如下命令载入sysctl配置文件，并设置活动内核参数：

```
[root@localhost ~]# sysctl -p
```

还原方法

- 编辑/etc/sysctl.conf文件把加固时新增的行删除，再去掉查找到的行的行首的"#"，通过如下命令以载入sysctl配置文件，并设置活动内核参数：

```
[root@localhost ~]# sysctl -p
```

修改影响

关闭accept_redirects参数后，内核将不再接收ICMP重定向报文，可能导致网络性能降低或延迟增加，也可能导致一些路由变更无法得到及时处理。

2.2 禁止send_redirects发送定向

级别

要求

适用版本

V10 SP3 2403

说明

检查系统内核参数配置，检查send_redirects配置。

> **说明** send_redirects是Linux系统中的一个配置选项，用于控制网络栈是否允许发送重定向包。当send_redirects设置为1时，系统允许发送重定向包；当设置为0时，系统禁止发送重定向包。 重定向包是一种网络协议包，用于将数据包从源主机重定向到目标主机。在某些情况下，源主机和目标主机之间的路由可能发生变化，导致数据包无法到达目标主机。此时，路由器可能会发送一个重定向包给源主机，指示它将数据包发送到新的路径。 在某些情况下，发送重定向包可能会对网络安全造成风险。

检查方法

- 通过如下命令验证是否禁止send_redirects发送定向：

```
[root@localhost ~]# sysctl net.ipv4.conf.all.send_redirects
net.ipv4.conf.all.send_redirects = 0
```

```
[root@localhost ~]# sysctl net.ipv4.conf.default.send_redirects
net.ipv4.conf.default.send_redirects = 0
```

```
[root@localhost ~]# grep "net\.ipv4\.conf\.all\.send_redirects" /etc/sysctl.conf
net.ipv4.conf.all.send_redirects = 0
```

```
[root@localhost ~]# grep "net\.ipv4\.conf\.default\.send_redirects" /etc/sysctl.conf
net.ipv4.conf.default.send_redirects = 0
```

修改建议

加固方法

- 编辑/etc/sysctl.conf文件把查找到的行在行首添加"#"进行注释，在注释行之后添加以下内容：

```
net.ipv4.conf.all.send_redirects = 0
net.ipv4.conf.default.send_redirects = 0
```

- 通过如下命令载入sysctl配置文件，并设置活动内核参数：

```
[root@localhost ~]# sysctl -p
```

还原方法

- 编辑/etc/sysctl.conf文件把加固时新增的行删除，再去掉查找到的行的行首的"#"，通过如下命令以载入sysctl配置文件，并设置活动内核参数：

```
[root@localhost ~]# sysctl -p
```

修改影响

关闭send_redirects参数后，内核将不再发送ICMP重定向报文，可能导致网络性能降低或延迟增加，也可能导致一些路由变更无法得到及时处理。

2.3 忽略icmp echo请求广播

级别

要求

适用版本

V10 SP3 2403

说明

ICMP是网络控制消息协议，主要用于传递查询报文与差错报文，通过设置是否接受ICMP广播报文对ICMP报文攻击进行防护。

该参数决定设备是否要回应ICMP echo消息和时间戳请求，对这些消息和请求来说，目的地址就是广播地址。无论是哪台设备发送的报文，报文都会发送到网络上的每一台设备上去。如果源地址是伪造的，就可能会导致网络上所有的设备发送恶意的echo报文给受害者（被伪造地址的设备）。

检查方法

- 通过如下命令验证是否忽略icmp echo请求广播：

```
[root@localhost ~]# sysctl net.ipv4.icmp_echo_ignore_broadcasts
net.ipv4.icmp_echo_ignore_broadcasts = 1
```

```
[root@localhost ~]# grep "net\.ipv4\.icmp_echo_ignore_broadcasts" /etc/sysctl.conf
net.ipv4.icmp_echo_ignore_broadcasts = 1
```

修改建议

加固方法

- 编辑/etc/sysctl.conf文件把查找到的行在行首添加"#"进行注释，在注释行之后添加以下内容：

```
net.ipv4.icmp_echo_ignore_broadcasts = 1
```

- 通过如下命令载入sysctl配置文件，并设置活动内核参数：

```
[root@localhost ~]# sysctl -p
```

还原方法

- 编辑/etc/sysctl.conf文件把加固时新增的行删除，再去掉查找到的行的行首的"#"，通过如下命令以载入sysctl配置文件，并设置活动内核参数：

```
[root@localhost ~]# sysctl -p
```

修改影响

禁用icmp_echo_ignore_broadcasts参数后，内核将不再响应来自广播地址的ping请求，而是直接丢弃这些请求

2.4 禁止icmp源路由

级别

要求

适用版本

V10 SP3 2403

说明

在网络中，源路由允许发送方部分或全部指定数据包通过网络的路由，而常规路由中，网络中的路由器根据数据包的目的地确定路径。大量报文被篡改后通过指定路由，则可以对内部网络进行定向攻击，可导致指定路由器负载过高，正常业务流量中断。

攻击者可以伪造一些合法的IP地址，通过合适的设置源路由选项及合法的路由器，蒙混进入网络。另外，如果允许源路由数据包，则通过构造中间路由地址，可以用于访问专用地址系统；如果攻击者对原始报文截取，并利用源路由进行地址欺骗，则可以强制指定回传的报文都通过攻击者的设备进行路由返回，这样攻击者就可以成功接收到双向的数据包。所以，应禁用报文源路由，减小攻击面。

检查方法

- 通过如下命令验证是否禁止icmp源路由：

```
[root@localhost ~]# sysctl net.ipv4.conf.all.accept_source_route
net.ipv4.conf.all.accept_source_route = 0
```

```
[root@localhost ~]# sysctl net.ipv4.conf.default.accept_source_route
net.ipv4.conf.default.accept_source_route = 0
```

```
[root@localhost ~]# grep "net\.ipv4\.conf\.all\.accept_source_route" /etc/sysctl.conf
net.ipv4.conf.all.accept_source_route= 0
```

```
[root@localhost ~]#grep "net\.ipv4\.conf\.default\.accept_source_route" /etc/sysctl.conf
net.ipv4.conf.default.accept_source_route= 0
```

修改建议

加固方法

- 编辑/etc/sysctl.conf文件把查找到的行在行首添加"#"进行注释，在注释行之后添加以下内容：

```
net.ipv4.conf.all.accept_source_route = 0
net.ipv4.conf.default.accept_source_route = 0
```

- 通过如下命令载入sysctl配置文件，并设置活动内核参数：

```
[root@localhost ~]# sysctl -p
```

还原方法

- 编辑/etc/sysctl.conf文件把加固时新增的行删除，再去掉查找到的行的行首的"#"，通过如下命令以载入sysctl配置文件，并设置活动内核参数：

```
[root@localhost ~]# sysctl -p
```

修改影响

禁用accept_source_route参数后，内核将不再接受含有源路由信息的IP数据包。

2.5 禁止ip_forward数据包转发

级别

要求

适用版本

V10 SP3 2403

说明

检查系统内核参数配置，检查ip_forward配置。

> **说明** net.ipv4.ip_forward参数是Linux系统中的一个网络参数，用于控制内核是否允许IP数据包进行路由转发。

检查方法

- 通过如下命令验证是否禁止ip_forward数据包转发：

```
[root@localhost ~]# sysctl net.ipv4.ip_forward
net.ipv4.ip_forward = 0
```

```
[root@localhost ~]# grep "net\.ipv4\.ip_forward" /etc/sysctl.conf
net.ipv4.ip_forward = 0
```

修改建议

加固方法

- 编辑/etc/sysctl.conf文件把查找到的行在行首添加"#"进行注释，在注释行之后添加以下内容：

```
net.ipv4.ip_forward = 0
```

- 通过如下命令以载入sysctl配置文件，并设置活动内核参数：

```
[root@localhost ~]# sysctl -p
```

还原方法

- 编辑/etc/sysctl.conf文件把加固时新增的行删除，再去掉查找到的行的行首的"#"，通过如下命令以载入sysctl配置文件，并设置活动内核参数：

```
[root@localhost ~]# sysctl -p
```

修改影响

当ip_forward参数设置为0时，内核将不允许IP数据包进行路由转发。也就是说，当内核接收到一个不属于本地主机的数据包时，它会直接丢弃数据包，而不是将其转发到适当的网卡上。

3 安全网络

3.1 修改snmp默认团体字

级别

要求

适用版本

V10 SP3 2403

说明

检查是否修改snmp默认团体字。

检查方法

- 通过如下命令验证snmp是否存在默认public、private团体字：

```
[root@localhost ~]# cat /etc/snmp/snmpd.conf |grep "public" |grep "default" | grep -v "^#"
com2sec notConfigUser  default       public
```

```
[root@localhost ~]# cat /etc/snmp/snmpd.conf |grep "private" |grep "default" | grep -v "^#"
com2sec notConfigUser  default       private
```

修改建议

加固方法

- 编辑/etc/snmp/snmpd.conf文件把查找到的行在行首添加"#"进行注释，把注释行的public改成：security_snmp1，private改成：security_snmp2并去除default关键字，添加在注释行的下面:

```
com2sec notConfigUser  security_snmp1
com2sec notConfigUser  security_snmp2
```

- 重启snmp服务

```
[root@localhost ~]# systemctl restart snmpd
```

还原方法

- 编辑/etc/snmp/snmpd.conf文件把加固时新增的行删除，再去掉查找到的行的行首的"#"，重启snmp服务：

```
[root@localhost ~]# systemctl restart snmpd
```

修改影响

禁用使用 "public" 和 "private" 共同体字符串的 SNMP 访问，使用这些共同体字符串的 SNMP 请求将无法通过身份验证。可使用security_snmp1、security_snmp2团体字。

3.2 关闭多IP绑定

级别

要求

适用版本

V10 SP3 2403

说明

检查是否配置关闭多IP绑定。

检查方法

- 通过如下命令并验证是否已关闭IP绑定:

```
[root@localhost ~]# grep multi /etc/host.conf
multi off
```

修改建议

加固方法

- 编辑/etc/host.conf文件，把查找到的行在行首添加"#"进行注释，将如下行添加在注释行的下面:

```
multi off
```

还原方法

- 把加固时新增的行删除，再去掉查找到的行的行首的"#

修改影响

关闭IP多绑定，通常意味着不再允许一个网络接口同时使用多个IP地址。

3.3 关闭IP伪装

级别

要求

适用版本

V10 SP3 2403

说明

将net.ipv4.conf.all.rp_filter和net.ipv4.conf.default.rp_filter设置为1，强制Linux内核对接收到的数据包使用反向路径过滤，检查报文源地址的合法性，如果反查源地址的路由表，发现源地址下一跳的最佳出接口并不是收到报文的入接口，则将报文丢弃。

攻击者可以实施IP地址欺骗，在目前网络攻击中使用比较多。通过反向地址过滤在收到数据包时，取出源IP地址，然后查看该路由器的路由表中是否有该数据包的路由信息。如果路由表中没有其用于数据返回的路由信息，那么极有可能是某人伪造了该数据包，于是路由便把它丢弃。

检查方法

- 通过如下命令验证是否启用反向路由筛选:

```
[root@localhost ~]# sysctl -n net.ipv4.conf.all.rp_filter
1
```

```
[root@localhost ~]# sysctl -n net.ipv4.conf.default.rp_filter
1
```

修改建议

加固方法

- 编辑/etc/sysctl.conf文件，找到查找到的行，在查找到的行前加"#"注释，然后添加以下参数：

```
net.ipv4.conf.all.rp_filter = 1
net.ipv4.conf.default.rp_filter = 1
```

- 通过如下命令设置活动内核参数：

```
[root@localhost ~]# sysctl -w net.ipv4.conf.all.rp_filter=1
net.ipv4.conf.all.rp_filter = 1
```

```
[root@localhost ~]# sysctl -w net.ipv4.conf.default.rp_filter=1
net.ipv4.conf.default.rp_filter = 1
```

还原方法

- 编辑/etc/sysctl.conf文件中找到查找行，在查找行前去掉"#"注释，删除以下行参数：

```
net.ipv4.conf.all.rp_filter = 1
net.ipv4.conf.default.rp_filter = 1
```

- 通过如下命令以设置活动内核参数：

```
[root@localhost ~]# sysctl -w net.ipv4.conf.all.rp_filter=0
net.ipv4.conf.all.rp_filter = 0
```

```
[root@localhost ~]# sysctl -w net.ipv4.conf.default.rp_filter=0
net.ipv4.conf.default.rp_filter = 0
```

修改影响

禁止了用户或应用程序更改他们的网络接口的IP地址。

4 系统命令

4.1 启用sudo日志

级别

要求

适用版本

V10 SP3 2403

说明

启用sudo日志可以审计用户在使用sudo时执行了什么命令等相关信息。

检查方法

- 查询文件/etc/sudoers是否包含如下参数，不包含则需加固

```
[root@localhost ~]# vim /etc/sudoers
Defaults logfile=/var/log/sudo.log
```

- 查询文件/etc/rsyslog.conf是否包含包含如下参数，不包含则需加固

```
[root@localhost ~]# cat /etc/rsyslog.conf
local2.debug            /var/log/sudo.log
```

> **注意** 空白处用"Tab"键补齐，不可以用空格

修改建议

加固方法

- 创建sudo.log文件

```
[root@localhost ~]# touch /var/log/sudo.log
```

- 修改文件权限

```
[root@localhost ~]# chmod root:root /var/log/sudo.log
```

- 修改rsyslog配置文件,添加以下内容

```
[root@localhost ~]# vim /etc/rsyslog.conf
local2.debug     /var/log/sudo.log
```

- 重启rsyslog服务

```
[root@localhost ~]# systemctl restart rsyslog
```

- 修改sudo配置文件，添加以下内容：

```
[root@localhost ~]# visudo
Defaults logfile=/var/log/sudo.log
```

- 修改sudo配置文件，删除以下内容

```
[root@localhost ~]# visudo
Defaults logfile=/var/log/sudo.log
```

还原方法

- 修改rsyslog配置文件,删除以下内容

```
[root@localhost ~]# vim /etc/rsyslog.conf
local2.debug     /var/log/sudo.log
```

- 重启rsyslog服务

```
[root@localhost ~]# systemctl restart rsyslog
```

- 删除sudo.log文件

```
[root@localhost ~]# rm /var/log/sudo.log
```

修改影响

配置将这些日志消息定向到 /var/log/sudo.log 文件。所有匹配 local2.debug 级别或更高级别的消息都将被写入这个文件。

4.2 设置sudo命令使用伪终端执行

设置sudo命令使用伪终端执行

级别

要求

适用版本

V10 SP3 2403

说明

检查方法

1.查询文件/etc/sudoers是否存在以下行

```
Defaults use_pty
```

修改建议

加固方法

- 修改sudo配置文件，添加以下内容：

```
[root@localhost ~]# visudo
Defaults use_pty
```

还原方法

- 修改sudo配置文件，删除以下内容：

```
[root@localhost ~]# visudo
Defaults use_pty
```

修改影响

- 启用伪终端（Pseudo-Terminal，PTY）：Defaults use_pty 这一行指令告诉 sudo 在执行命令时使用一个伪终端。伪终端是一种模拟真实终端设备的软件机制，它允许进程之间进行交互式的输入和输出。
- 交互式命令支持：当 sudo 使用 PTY 时，它能够更好地支持需要交互式输入的命令。例如，如果一个命令提示用户进行确认或者输入密码，那么在使用 PTY 的情况下，这些交互将能够正常进行。
- 安全增强：启用 PTY 可以提高安全性，因为它可以防止某些类型的攻击，如 tty 认证绕过攻击。通过使用 PTY，sudo 能够更准确地模拟终端环境，使得攻击者更难以利用漏洞。

4.3 设置使用指定用户sudo提权需输入指定用户的密码

级别

要求

适用版本

V10 SP3 2403

说明

在Linux系统中，sudo（superuser do）命令允许普通用户以超级用户（或其他用户）的身份执行命令。默认情况下，当普通用户使用sudo时，系统会要求输入该用户自己的密码，而不是超级用户的密码。这是sudo安全机制的一部分，用于确保只有经过授权的用户才能以更高的权限执行命令。

检查方法

1.查询文件/etc/sudoers是否存在以下行

```
Defaults targetpw
```

修改建议

加固方法

- 修改sudo配置文件，添加以下内容：

```
[root@localhost ~]# visudo
Defaults targetpw
```

还原方法

- 修改sudo配置文件，删除以下内容：

```
[root@localhost ~]# visudo
Defaults targetpw
```

修改影响

- 密码提示：Defaults targetpw 这一行指令告诉 sudo 在用户以其他用户身份执行命令时，提示输入目标用户的密码，而不是当前用户的密码。
- 提高安全性：这个设置增强了系统的安全性，因为即使用户知道自己的密码，他们也需要知道目标用户的密码才能以该用户的身份执行命令。这可以防止未经授权的用户获取额外的权限。
- 管理员操作：对于系统管理员来说，这个设置可能会增加一些操作的复杂性，因为他们需要记住更多的密码。但是，这也鼓励了更好的权限管理实践，避免了不必要的权限提升。

4.4 配置su命令使用情况记录

级别

要求

适用版本

V10 SP3 2403

说明

检查是否配置su命令使用情况记录。

检查方法

- 检查/etc/rsyslog.conf文件是否配置了如下行：

```
[root@localhost ~]# cat /etc/rsyslog.conf
authpriv.*   /var/log/secure
```

修改建议

加固方法

- 编辑/etc/rsyslog.conf文件，添加如下行：

```
authpriv.*   /var/log/secure
```

还原方法

- 编辑/etc/rsyslog.conf文件，删除添加的行：

```
authpriv.*   /var/log/secure
```

修改影响

su命令使用的日志消息都会被写入到/var/log/secure文件中。

4.5 限制输出和保留历史命令的条数

级别

要求

适用版本

V10 SP3 2403

说明

检查保留历史命令的条数和保留历史命令的记录文件大小。

检查方法

- 通过如下命令验证查看保留历史命令的条数条数设置为5条以下:

```
[root@localhost ~]# grep "^HISTSIZE" /etc/profile
5
```

- 通过如下命令验证查看保留历史命令的记录文件大小条数设置为5条以下:

```
[root@localhost ~]# grep "^HISTFILESIZE" /etc/profile
5
```

修改建议

加固方法

- 编辑/etc/profile文件，将查找到的行在行首添加"#"进行注释，在注释行下添加如下：

```
HISTSIZE=5
```

- 编辑/etc/profile文件，将查找到的行在行首添加"#"进行注释，在注释行下添加如下:

```
HISTFILESIZE=5
```

还原方法

- 编辑/etc/profile文件，查找HISTSIZE所在的行，在行首去掉"#"注释，删除加固时添加的行:

```
HISTSIZE=5
```

- 编辑/etc/profile文件，查找HISTFILESIZE所在的行,在行首去掉"#"注释,删除加固时添加的行:

```
HISTFILESIZE=5
```

修改影响

配置后只能输出和保留最近的5条历史命令。

5 系统审计

5.1 swatch软件加固

级别

建议

适用版本

V10 SP3 2403

说明

审计日志分析软件swatch未安装，建议安装。

检查方法

- 通过如下命令检查是否安装了swatch软件：

```
[root@localhost ~]# rpm -qa |grep swatch
```

修改建议

加固方法

- 配置软件安装环境，通过如下命令安装swatch软件：

```
[root@localhost ~]# yum install swatch
```

还原方法

- 通过如下命令卸载swatch软件：

```
[root@localhost ~]# rpm -e swatch
```

修改影响

SWATCH 可以帮助监控系统日志、网站和文件的变化，及时发现并响应潜在的安全威胁，如未经授权的访问、恶意攻击等。

5.2 开启审计机制

级别

建议

适用版本

V10 SP3 2403

说明

保证审计机制默认处于开启状态,且对审计日志的开启和关闭进行保护，以x86_64架构为例。

检查方法

- 通过如下命令，查看系统架构：

```
[root@localhost ~]# arch
x86_64
```

- 系统是bios引导，检查/boot/grub2/grub.cfg文件，匹配"vmlinuz"参数，匹配上一步arch命令查出的"[架构]"参数，检查audit是否已开启，1开启，0未开启：

```
[root@localhost ~]# cat -n /boot/grub2/grub.cfg |grep "vmlinuz" |grep [架构]
linux   /vmlinuz-4.19.90-83.1.v2307.ky10.x86_64 root=/dev/mapper/klas-root ro resume=/dev/mapper/klas-swap rd.lvm.lv=klas/rc
```

- 系统是uefi引导，检查/boot/efi/EFI/kylin/grub.cfg文件中内容，匹配"vmlinuz"参数，匹配上一步arch命令查出的"[架构]"参数，检查audit是否已开启，1开启，0未开启：

- 通过如下命令检查auditd服务是否开启：

```
[root@localhost ~]# systemctl is-enabled  auditd.service
enabled
[root@localhost ~]# systemctl is-active  auditd.service
active
```

加固方法

- 根据检查方法，确认要更改的文件，如果目标行无audit参数则在行尾添加audit=1参数，如果目标行存在audit=0，将其改为audit=1：

```
audit=1
```

> **注意** 添加的audit=1与相邻参数间要有空格

- 设置auditd服务自启动：

```
[root@localhost ~]# systemctl enable  auditd.service
```

- 配置完以上步骤**重启系统生效**

还原方法

- 根据检查方法，确认要更改的文件，如果目标行无audit参数删除添加的audit=1参数，如果目标行存在audit=0，将其改为audit=0改为audit=1：

- 禁止auditd服务自启动：

```
[root@localhost ~]# systemctl disable  auditd.service
```

- 配置完以上步骤**重启系统生效**

修改影响

- 系统性能 overhead： 启动audit服务会增加系统的负载，因为它需要监控和记录系统的各种活动。这包括文件访问、系统调用、用户登录注销等事件。这些额外的记录操作可能会占用一定的CPU、内存和磁盘I/O资源，从而可能对系统性能产生一定影响。
- 安全性增强： audit服务的主要目的是提高系统的安全性。通过审计，系统可以跟踪和记录用户的操作，这对于安全分析和入侵检测非常有用。审计日志可以帮助管理员发现潜在的安全威胁，识别未经授权的访问或恶意行为。
- 存储需求增加： audit服务生成的审计日志需要存储空间。随着系统活动的增加，审计日志的大小也会增长。如果不合理管理审计日志，例如设置适当的日志rotate策略，存储空间可能会被迅速填满。

5.3 设置系统审计日志规则

级别

建议

适用版本

V10 SP3 2403

说明

1、身份鉴别、自主访问控制、标记和强制访问控制等安全功能的使用 2、创建、删除客体的操作 3、所有管理员的操作 4、每条审计记录应包括：事件类型、事件发生的日期和时间、触发事件的用户、事件成功或失败等字段 5、创建和删除客体的事件审计记录还应包括客体的名字、客体的安全属性 6、网络会话事件审计记录还应包括：网络程序名称、协议类型、源ＩＰ地址、目的ＩＰ地址、源端口、目的端口、会话总字节数等字段。

检查方法

- 通过如下命令验证是否安装audit包：

```
[root@localhost ~]# rpm -qa audit
audit-3.0-6.se.06.ky10.x86_64
```

- 通过如下命令查看是否设置了规则：

```
[root@localhost ~]# auditctl -l
No rules
```

修改建议

加固方法

- 通过如下命令开启auditd服务：

```
[root@localhost ~]# systemctl enable auditd
[root@localhost ~]# systemctl restart auditd
```

- 查看是否设置了规则（输出为空则未添加规则），如未设置规则则提示用户添加规则：

```
[root@localhost ~]# auditctl -l
No rules
```

- 建议1.内核的审计规则：如果未添加规则，编辑/etc/audit/rules.d/audit.rules文件添加如下规则：

```
[root@localhost ~]# vim /etc/audit/rules.d/module.rules  // 内核的审计规则
-w /sbin/insmod -p x -k kmodule
-w /sbin/rmmod -p x -k kmodule
-w /sbin/modprobe -p x -k kmodule
-a always,exit -F arch=b32 -S init_module -S delete_module -k kmodule
```

- 若是是64位系统，需要再添加arch=b64相关配置：

```
-a always,exit -F arch=b64 -S init_module -S delete_module -k kmodule
```

- 建议2.修改账号信息的审计规则：在/etc/audit/rules.d/目录下新建规则文件，例如usermgn.rules，在文件中添加审计规则：

```
[root@localhost ~]# vim /etc/audit/rules.d/usermgn.rules // 修改账号信息的审计规则
-w /etc/group -p wa -k usermgn
-w /etc/passwd -p wa -k usermgn
-w /etc/gshadow -p wa -k usermgn
-w /etc/shadow -p wa -k usermgn
-w /etc/security/opasswd -p wa -k usermgn
```

- 建议3.配置时间修改的审计规则 在/etc/audit/rules.d/目录下新建规则文件，例如time.rules，在文件中添加审计规则：

```
[root@localhost ~]# vim /etc/audit/rules.d/time.rules // 配置时间修改的审计规则
-a always,exit -F arch=b32 -S stime -S settimeofday -S adjtimex -S clock_settime -k time
-w /etc/localtime -p wa -k time
```

- 如果是64位系统，需要再添加arch=b64相关配置：

```
-a always,exit -F arch=b64 -S settimeofday -S adjtimex -S clock_settime -k time
```

- 考虑兼容性，64位系统中arch=b32相关配置必须保留。
- 重启auditd服务，使规则生效：

```
[root@localhost ~]# systemctl restart  auditd.service
```

还原方法

- 通过如下命令关闭auditd服务:

```
[root@localhost ~]# systemctl disable auditd
[root@localhost ~]# systemctl stop auditd
```

- 删除规则（加固时设置的规则）

```
[root@localhost ~]# auditctl -D  // 删除auditctl -l 查出的所有规则
```

- 删除加固时创建的.rules文件（module.rules，usermgn.rules，time.rules）

修改影响

监控和跟踪用户活动，以及识别潜在的安全威胁，例如未经授权的访问尝试、异常行为和恶意操作。

5.4 设置审计存储阈值

级别

建议

适用版本

V10 SP3 2403

说明

审计日志应存储在掉电非遗失性存储媒体中。系统管理员应能定义超过审计跟踪存储极限的阈值,当超过阈值时将向管理员报警。当审计存储空间被耗尽时，覆盖所存储的最早的审计记录。

> **说明** 当分配的审计记录存储量达到存储库最大审计记录存储容量的 75% 时，操作系统必须立即通知系统管理员 （SA） 和信息系统安全官 ISSO（至少）为例

检查方法

- 获取总容量:

```
[root@localhost ~]# df -h /var/log/audit/ | awk '{if (NR>1){print $2}}'|awk -F 'G' '{print $1}'
64
```

结果为64，单位为G，再将这个值*1024*0.25，获取到剩余大小16384，单位为M

- 执行以下命令获取/etc/audit/auditd.conf文件中参数space_left值，若不等于上一步骤获取值（16384）则需加固

```
[root@localhost ~]# cat /etc/audit/auditd.conf|grep -w space_left
space_left = 75
```

加固方法

- 设置/etc/audit/auditd.conf文件中的space_left值

```
[root@localhost ~]# vim /etc/audit/auditd.conf
space_left = 16384
```

- 重启audit服务

```
systemctl restart  auditd.service
```

还原方法

- 设置/etc/audit/auditd.conf文件中的space_left值为初始值

```
[root@localhost ~]#  vim /etc/audit/auditd.conf
space_left = 75
```

- 重启audit服务

```
systemctl restart  auditd.service
```

修改影响

space_left参数定义了当磁盘空间剩余量低于这个值（以KB为单位）时，auditd服务应该如何响应。例如，你设置的值是16384，这意味着当用于存储审计日志的分区空闲空间小于16MB时，auditd服务将采取预定义的动作。

5.5 限制用户使用计划任务

级别

建议

适用版本

V10 SP3 2403

说明

限制指定的管理员账户运行计划任务，防止开机运行不明任务。

检查方法

- 在终端中输入命令：

```
[root@localhost ~]# more /etc/cron.allow
```

- 根据现实结果查看当前允许运行计划任务的用户

加固方法

- 在/etc/cron.allow文件中添加需要使用计划任务的用户

还原方法

- 在/etc/cron.allow文件中删除添加的用户

修改影响

只有在 /etc/cron.allow 文件中列出的用户才能使用 crontab 命令来创建和管理计划任务。

5.6 启动日志服务rsyslog

级别

要求

适用版本

V10 SP3 2403

说明

通过启动日志服务即使记录相关行为的操作用于取证与溯源。

检查方法

- 通过如下命令检查rsyslog服务是否开启：

```
[root@localhost ~]# systemctl is-enabled rsyslog.service
enable
[root@localhost ~]# systemctl is-active  rsyslog.service
active
```

修改建议

加固方法

- 通过如下命令开启rsyslog服务：

```
[root@localhost ~]# systemctl enable rsyslog.service
[root@localhost ~]# systemctl start rsyslog.service
```

还原方法

- 通过如下命令禁用rsyslog服务：

```
[root@localhost ~]# systemctl disable rsyslog.service
[root@localhost ~]# systemctl stop rsyslog.service
```

修改影响

- 日志记录：启动 rsyslog.service 服务会启用系统的日志记录功能。系统活动、应用程序消息、错误信息和其他重要事件将被收集和记录下来。
- 系统监控：日志是系统监控的重要组成部分。通过分析 rsyslog 记录的日志，管理员可以了解系统的运行状态、识别潜在问题和安全威胁。
- 故障排查：当系统出现问题或故障时，日志文件是重要的故障排查工具。通过查看相关日志，管理员可以追踪问题的根源，帮助修复故障。
- 资源使用：运行 rsyslog.service 会占用一定的系统资源，包括 CPU、内存和磁盘空间。日志记录和存储过程可能会对系统性能产生轻微影响，尤其是在高负载或大量日志生成的情况下。
- 安全性和合规性：许多安全策略和法规要求系统必须记录和保留特定类型的日志。启用 rsyslog.service 可以帮助满足这些要求，提高系统的安全性并确保合规性。
- 网络通信（如果配置为远程日志服务器）：如果 rsyslog 配置为接收来自其他系统或设备的日志，那么开启该服务将导致网络通信。这可能包括监听特定端口（默认为 UDP 端口 514）以及发送和接收日志数据。
- 隐私考虑：日志记录可能会涉及到用户隐私和敏感信息。管理员需要确保日志配置符合隐私政策和法规要求，并采取适当的安全措施来保护日志数据。
- 日志管理和存储：开启 rsyslog.service 会导致日志文件的增长。如果不合理管理日志文件，例如设置适当的日志rotate策略和清理旧日志，磁盘空间可能会被迅速填满。

5.7 记录用户对设备的操作

级别

建议

适用版本

V10 SP3 2403

说明

检查是否记录用户对设备的操作。

检查方法

- 通过如下命令，检查是否安装了psacct软件:

```
[root@localhost ~]# rpm -qa |grep psacct
psacct-6.6.4-5.ky10.x86_64
```

- 通过如下命令，检查psacct服务是否开启:

```
[root@localhost ~]# systemctl is-enabled psacct.service
enable
[root@localhost ~]# systemctl is-active  psacct.service
active
```

修改建议

加固方法

- 通过如下命令，安装psacct软件：

```
[root@localhost ~]# yum install  psacct
Last metadata expiration check: 1:11:27 ago on 2023年12月29日 星期五 10时06分22秒.
Dependencies resolved.
================================================================================================================================
 Package                                          Architecture                                  Version
================================================================================================================================
Installing:
 psacct                                           x86_64                                        6.6.4-5.ky10
Installing weak dependencies:
 psacct-help                                      x86_64                                        6.6.4-5.ky10

Transaction Summary
================================================================================================================================
Install  2 Packages

Total download size: 85 k
Installed size: 230 k
Is this ok [y/N]: y
```

- 通过如下命令，开启psacct.service服务：

```
[root@localhost ~]# systemctl enable psacct.service
[root@localhost ~]# systemctl start psacct.service
```

还原方法

- 通过如下命令，卸载psacct软件：

```
[root@localhost ~]# rpm -e psacct
```

```
[root@localhost ~]# rpm -e psacct-help
```

- 通过如下命令，关闭psacct.service服务：

```
[root@localhost ~]# systemctl disable psacct.service
[root@localhost ~]# systemctl stop psacct.service
```

修改影响

通过安装并启用 psacct，管理员可以更好地了解系统的资源使用情况，帮助识别潜在的性能瓶颈，并进行相应的优化。

5.8 记录用户登录日志

级别

要求

适用版本

V10 SP3 2403

说明

检查是否对登录进行日志记录。

检查方法

- 通过如下命令，检查是否存在/var/log/wtmp、/var/run/utmp文件:

```
[root@localhost ~]# ll /var/log/wtmp
-rw-rw-r-- 1 root utmp 10368 12月 18 15:16 /var/log/wtmp
[root@localhost ~]# ll /var/run/utmp
-rw-rw-r-- 1 root utmp 1536 12月 18 15:16 /var/run/utmp
```

修改建议

加固方法

- 通过如下命令，创建文件/var/log/wtmp：

```
[root@localhost ~]# touch /var/log/wtmp
```

- 通过如下命令，创建文件/var/run/utmp：

```
[root@localhost ~]# touch /var/run/utmp
```

还原方法

- 通过如下命令，删除文件/var/log/wtmp：

```
[root@localhost ~]# rm  /var/log/wtmp
```

- 通过如下命令，删除文件/var/run/utmp：

```
[root@localhost ~]# rm  /var/run/utmp
```

修改影响

用户登录和注销信息记录在/var/log/wtmp和/var/run/utmp文件。

5.9 配置安全事件日志

级别

要求

适用版本

V10 SP3 2403

说明

检查安全事件日志配置。

检查方法

- 检查/etc/rsyslog.conf文件是否配置了以下内容：

```
[root@localhost ~]# cat /etc/rsyslog.conf
*.err /var/adm/messages
kern.debug /var/adm/messages
daemon.notice /var/adm/messages
```

修改建议

加固方法

- 编辑/etc/rsyslog.conf文件在相关文件中配置添加如下行：

```
*.err /var/adm/messages
kern.debug /var/adm/messages
daemon.notice /var/adm/messages
```

还原方法

删除加固时添加的行。

修改影响

- *.err /var/adm/messages： 将所有错误级别的日志（err）都记录到"/var/adm/messages"文件中。这里的"*"表示所有设备或所有日志消息。
- kern.debug /var/adm/messages： 这个规则将内核（kern）产生的所有调试（debug）级别消息写入到/var/adm/messages 文件中。 调试级别的消息通常包含非常详细的信息，可能包括系统内部的操作和状态信息。
- daemon.notice /var/adm/messages： 这个规则将守护进程（daemon）产生的通知（notice）级别消息写入到/var/adm/messages 文件中。 通知级别的消息通常表示值得注意但不紧急的情况，如服务启动、停止等。

5.10 启用cron行为日志功能

级别

要求

适用版本

V10 SP3 2403

说明

检查安全事件日志配置。

检查方法

- 检查/etc/rsyslog.conf文件是否配置了以下内容：

```
[root@localhost ~]# cat /etc/rsyslog.conf
cron.* /var/log/cron
```

修改建议

加固方法

- 编辑/etc/rsyslog.conf或/etc/syslog.conf文件添加如下行：

```
cron.* /var/log/cron
```

- 通过如下命令，重启rsyslog或syslog服务:

```
[root@localhost ~]# systemctl restart rsyslog.service
```

- 如果/var/log/cron文件不存在，创建文件并设置755权限（文件存在无需执行创建和设置权限步骤）：

```
# touch /var/log/cron
# chmod 755 /var/log/cron
```

还原方法

- 编辑/etc/syslog.conf文件删除加固时添加的行，恢复文件默认权限，重启对应服务：

```
[root@localhost ~]# systemctl restart rsyslog.service
或
[root@localhost ~]# systemctl restart syslog.service
```

修改影响

- 该规则将所有这些级别的 cron 日志都写入到指定的文件/var/log/cron中。

5.11 检查是否安装入侵检测工具AIDE

级别

建议

适用版本

V10 SP3 2403

说明

基于模板的异常检测:根据用户的历史使用模式建立模板，用置疑等级表示用户当前活动与模板中已建立的使用模式不一致的程度,当用户的置疑等级超过门限条件时,能指出对操作系统的可能侵害即将发生。当检测到潜在的安全侵害时,应生成实时报警。

> **说明** 文件完整性工具必须至少每周验证一次基线操作系统配置。

检查方法

- 查看是否安装，安装了提示无需加固，未安装提示需要手动加固。

```
[root@localhost ~]# rpm -qa |grep aide
```

安装方法

- 配置软件安装环境，通过如下命令安装aide软件：

```
[root@localhost ~]# yum install aide
```

修改影响

AIDE，全称为Advanced Intrusion Detection Environment，是一个主要用于检测文件完整性的入侵检测工具。它能够构建一个指定文件的数据库，并使用aide.conf作为其配置文件。AIDE数据库能够保存文件的各种属性，包括权限、索引节点序号、所属用户、所属用户组、文件大小、最后修改时间、创建时间、最后访问时间、增加的大小以及连接数等。此外，AIDE还支持多种算法，如sha1、md5、rmd160、tiger等，以密文形式建立每个文件的校验码或散列号。安装入侵检测工具AIDE可以帮助企业或组织提高安全性，但也会带来一些潜在的影响，如需要配置和管理数据库，以及需要定期更新和维护。

5.12 检查文件完整性检测配置

级别

建议

适用版本

V10 SP3 2403

说明

需要定期检查文件系统完整性，以检测文件系统的更改。

> **说明** 定期文件检查允许系统管理员定期确定是否以未经授权的方式更改了关键文件。

检查方法

- 使用命令crontab -u root -l | grep aide | grep check 查询，如果存在，提示无需加固，否则需要加固

```
[root@localhost ~]# crontab -u root -l | grep aide | grep check
no crontab for root
```

加固方法

- 手动创建检查文件系统完整性的定时任务,例如：

```
[root@localhost ~]# crontab -e
# 每天凌晨1点检查文件系统完整性
0 1 * * * /usr/bin/fsck /dev/sda1
```

修改影响

手动创建检查文件系统完整性的定时任务是一个良好的系统管理实践，它可以增强系统的可靠性。

6 系统设置

6.1 设置命令行超时退出

级别

要求

适用版本

V10 SP3 2403

说明

检查是否设置命令行界面超时退出。

检查方法

- 通过如下命令验证TMOUT不超过600秒：

```
[root@localhost ~]# grep "^TMOUT" /etc/profile
export TMOUT=600
```

修改建议

加固方法

- 编辑/etc/profile文件添加如下行：

```
export TMOUT=600
```

- 运行如下命令，功能立即生效：

```
[root@localhost ~]# source /etc/profile
```

还原方法

- 编辑/etc/profile文件删除加固时添加的TMOUT行:

```
export TMOUT=600
```

- 运行如下命令，设置立即生效：

```
[root@localhost ~]# source /etc/profile
```

修改影响

如果用户在命令行上没有任何输入长达10分钟（即600秒），那么他们的会话将被自动注销。

6.2 设置系统引导管理器密码

级别

要求

适用版本

V10 SP3 2403

说明

Grub是Linux的默认引导程序，通过引导程序可以设置系统的启动模式，而设置Grub口令可以防御攻击者通过修改Grub设置进入单用户模式。 如果没有设置Grub口令，攻击者可以轻易进入Grub编辑菜单，通过修改启动参数进行攻击行为，例如：进入单用户模式修改root口令，窃取数据。 UEFI和boot是两种不同的引导方式，对应的Grub配置文件路径会存在差异。efi引导为/boot/efi/EFI/kylin/grub.cfg文件，boot引导为/boot/grub2/grub.cfg文件。 **以x86系统为例**

检查方法

- 若系统为efi引导，检查/boot/efi/EFI/kylin/user.cfg中是否存在"GRUB2_PASSWORD"：

```
[root@localhost ~]#  cat /boot/efi/EFI/kylin/user.cfg |grep GRUB2_PASSWORD
RUB2_PASSWORD=grub.pbkdf2.sha512.10000.C0A907E3C7721CADF87F7B25CB3D68A3E6DDE415ACB57631E4821393E927A4D9D9C87B14A5C201CDAD2F1
```

- 检查/boot/efi/EFI/kylin/grub.cfg文件中是否存在匹配"set""superusers"的行：

```
[root@localhost ~]#  cat /boot/efi/EFI/kylin/user.cfg |grep set |grep superusers
set superusers="root"
```

- 检查是否存在匹配"password"、且不存在"GRUB2_PASSWORD"关键字的行：

```
[root@localhost ~]#  cat /boot/efi/EFI/kylin/user.cfg |grep password |grep -v GRUB2_PASSWORD
```

- 若系统为boot引导，检查/boot/grub2/user.cfg中是否存在"GRUB2_PASSWORD"：

```
[root@localhost ~]#  cat /boot/grub2/user.cfg |grep GRUB2_PASSWORD
GRUB2_PASSWORD=grub.pbkdf2.sha512.10000.C0A907E3C7721CADF87F7B25CB3D68A3E6DDE415ACB57631E4821393E927A4D9D9C87B14A5C201CDAD2F
```

- 检查/boot/grub2/grub.cfg文件中是否存在匹配"set""superusers"的行：

```
[root@localhost ~]# cat  /boot/grub2/grub.cfg |grep set |grep superusers
set superusers="root"
```

- 检查是否存在匹配"password"、且不存在"GRUB2_PASSWORD"关键字的行：

```
[root@localhost ~]# cat  /boot/grub2/user.cfg |grep password |grep -v GRUB2_PASSWORD
```

修改方法

加固方法

- 通过如下命令设置grub密码：

```
[root@localhost ~]# grub2-setpassword
    Enter password:  《《《这里输入你的grub密码
    Reenter password:  《《《这里输入你的grub密码
```

- 通过如下命令查看密码是否生成：

```
[root@localhost ~]# cat /boot/grub2/user.cfg
GRUB2_PASSWORD=grub.pbkdf2.sha512.10000.9D8FD7A8D39E4397CFA159588CD71A68DDC8B6A07D0E1B3BD35DA42D89B1DC0809F2A9CA78614C384324
```

还原方法

- 删除/boot/grub2/user.cfg文件:

```
[root@localhost ~]# rm /boot/grub2/user.cfg
```

修改影响

设置后进入系统引导管理器需要输入密码。

6.3 管理sudo权限

级别

要求

适用版本

V10 SP3 2403

说明

限制用户使用sudo权限，防止用户对系统做出破坏性更改或恶意提权操作。

检查方法

- 在终端中输入命令：

```
[root@localhost ~]# visudo
```

- 查找未被"#"注释掉的部分，是否存在类似于以下行，有即为定义的sudo用户

```
root ALL=(ALL) ALL
```

- 查找未被"#"注释掉的部分，是否存在类似于以下行，即为定义wheel组为sudo用户组

```
%wheel ALL=(ALL) ALL
```

若存在首元素既不是root也不是%wheel，则需加固。

修改建议

加固方法

- 编辑/etc/sudoers文件删除或注释检查到的行：

```
[root@localhost ~]# visudo
#test ALL=(ALL) ALL
```

还原方法

- 编辑/etc/sudoers文件添加或对检查到的行取消注释：

```
[root@localhost ~]# visudo
test ALL=(ALL) ALL
```

修改影响

test用户无法使用sudo命令。

6.4 限制su命令的访问

级别

要求

适用版本

V10 SP3 2403

说明

使用PAM认证模块禁止wheel组之外的用户su为root。

> **说明** 该项PAM 配置条目是为了实现只允许 root 用户和 wheel 组成员通过 su 命令成为其他用户的目的，同时，当这些用户使用 su 命令时，将以他们的 UID 而非 root 的 UID 执行命令。

检查方法

- 通过如下命令检查是否使用PAM认证模块禁止wheel组之外的用户su为root：

```
[root@localhost ~]# grep -v "deny" /etc/pam.d/su |grep  "group=wheel" |grep "use_uid" |grep "root_only"
auth required pam_wheel.so root_only group=wheel use_uid
```

修改建议

加固方法

- 编辑/etc/pam.d/su文件，注释匹配"required、 pam_wheel"关键字，且不存在"deny"关键字的行，增加以下内容：

```
auth required pam_wheel.so root_only group=wheel use_uid
```

还原方法

- 删除/etc/pam.d/su文件中的注释，删除增加的行。

修改影响

禁止wheel组之外的用户su为root，可以在一定程度上提高系统的安全性、增强审计能力、维护系统的稳定性等方面带来诸多益处。
PAM参数说明表

配置项	说明
root_only	仅允许 root 用户使用 su 或 sudo 命令。
group=wheel	只有属于 wheel 组的用户才能使用 su。
use_uid	使用当前用户的 UID 而不是 root 用户的 UID 执行命令。

6.5 检查是否安装时间同步软件包

级别

建议

适用版本

V10 SP3 2403

说明

> **说明**

检查方法

*查看是否安装chrony和ntp，安装了提示无需加固，若均未安装提示需要手动加固。

```
[root@localhost ~]#rpm -qa chrony ntp
```

安装方法

- 配置软件安装环境，运行以下命令安装chrony软件：

```
[root@localhost ~]#  yum install chrony
```

修改影响

安装时间同步软件包可以确保系统时间的准确性，但需要注意网络依赖性、安全性、系统资源占用和权限管理等方面的影响。在安装之前，需要进行充分的评估和准备，以确保系统的正常运行和安全性。

6.6 设置系统时间同步

级别

要求

适用版本

V10 SP3 2403

说明

设置时间同步保证记录日志的时间的准确性。

检查方法

- 首先查看配置文件/etc/chrony.conf和/etc/sysconfig/chronyd是否存在。如果都不存在,无需加固，如果其中一个文件不存在，则反馈为需要加固

```
[root@localhost ~]# ll /etc/chrony.conf
-rw-r--r-- 1 root root 1169  4月  8  2021 /etc/chrony.conf

[root@localhost ~]# ll /etc/sysconfig/chronyd
-rw-r--r-- 1 root root 46  4月  8  2021 /etc/sysconfig/chronyd
```

- 查看/etc/chrony.conf文件里是否有server或pool配置行，如不存在则需加固

```
[root@localhost ~]# cat /etc/chrony.conf|grep -E 'server|pool'
# Use public servers from the pool.ntp.org project.
# Please consider joining the pool (http://www.pool.ntp.org/join.html).
#pool pool.ntp.org iburst
server ntp.ntsc.ac.cn iburst
server ntp1.aliyun.com iburst
#server cn.pool.ntp.org iburst
```

- 查看/etc/sysconfig/chronyd是否有OPTIONS=，如不存在则需加固

```
[root@localhost ~]# cat /etc/sysconfig/chronyd|grep OPTIONS=
OPTIONS=""
```

以上步骤运行结果均匹配无需加固，否则需要加固。

修改方法

加固方法

- 编辑/etc/chrony.conf配置文件

```
[root@localhost ~]# vim /etc/chrony.conf
```

- 添加相应的NTP服务器信息

```
server ntp1.aliyun.com iburst
或
pool pool.ntp.org iburst
```

- 重启chronyd服务

```
[root@localhost ~]# systemctl restart chronyd
```

还原方法

- 删除添加的NTP服务器信息

```
server ntp1.aliyun.com iburst
或
pool pool.ntp.org iburst
```

- 重启chronyd服务

```
[root@localhost ~]# systemctl restart chronyd
```

修改影响

从 pool.ntp.org 提供的 NTP 服务器池中选择时间源，并使用 "iburst" 参数，Chrony 就会开始使用添加的 NTP 服务器或服务器池进行时间同步。

6.7 关闭系统core dump

级别

建议

适用版本

V10 SP3 2403

说明

关闭系统core dump设置。

> **说明：** Core dump文件可能包含应用程序运行时的内存状态，包括一些潜在的敏感数据，例如密码、密钥或其他用户信息。

检查方法

- 通过如下命令，检查/etc/sysctl.conf文件中存在"kernel.core_pattern"关键词，查看其后路径是否只为"|"管道字符:

```
[root@localhost ~]# cat /etc/sysctl.conf |grep /proc/sys/kernel/core_pattern
```

- 通过如下命令，检查/proc/sys/kernel/core_pattern文件中是以"|"管道字符开头，且"|"管道字符后面路径不包含/usr/lib/systemd/systemd-coredump有效程序:

```
[root@localhost ~]# cat /proc/sys/kernel/core_pattern
|
```

修改方法

加固方法

- 编辑/etc/sysctl.conf文件中把查找到的行在行首添加"#"进行注释，在注释行之后添加以下内容：

```
kernel.core_pattern=|
```

- 编辑/etc/sysctl.conf文件将含有"kernel.core_pattern"但其后路径不只是为"|"管道字符的行，用以下内容替换：

```
kernel.core_pattern=|
```

- 通过如下命令立即生效：

```
[root@localhost ~]# sysctl -p
```

还原方法

将/etc/sysctl.conf文件中修改过的行复原，通过如下命令立即生效：

```
[root@localhost ~]# sysctl -p
```

修改影响

关闭core dump有助于保护敏感信息，但它也可能会对你的系统管理和调试工作增加难度。

6.8 禁用ctrl+alt+del组合键

级别

要求

适用版本

V10 SP3 2403

说明

检查系统是否禁用ctrl+alt+del组合键

检查方法

- 按下组合键后系统会不会重启

修改方法

加固方法

- 通过如下命令，设置图形界面禁用组合键：

```
[root@localhost ~]# gsettings set org.mate.SettingsDaemon.plugins.media-keys logout ''
```

- 通过如下命令，设置终端禁用组合键，配置内核参数为"0"：

```
[root@localhost ~]# echo 0 > /proc/sys/kernel/ctrl-alt-del
```

- 编辑/etc/systemd/system.conf文件把查找到的行在行首添加"#"进行注释，在注释行之后添加以下内容,设置用户在一定时间内连续按下Ctrl+Alt+Del组合键时系统的无响应行为：

```
CtrlAltDelBurstAction=none
```

- 通过如下命令，删除/usr/lib/systemd/system/ctrl-alt-del.target、/etc/systemd/system/ctrl-alt-del.target文件：

```
[root@localhost ~] rm -rf  /usr/lib/systemd/system/ctrl-alt-del.target
[root@localhost ~] rm -rf  /etc/systemd/system/ctrl-alt-del.target
```

- 通过如下命令，设置功能立即生效：

```
[root@localhost ~] systemctl daemon-reexec
```

还原方法

- 通过如下命令，设置图形界面启用组合键：

```
[root@localhost ~]# gsettings set org.mate.SettingsDaemon.plugins.media-keys logout '<Control><Alt>Delete'
```

- 通过如下命令，设置终端启用组合键，配置内核参数为"1"：

```
[root@localhost ~]# echo 1 > /proc/sys/kernel/ctrl-alt-del
```

- 编辑/etc/systemd/system.conf文件删除添加的行：

```
CtrlAltDelBurstAction=none
```

- 通过如下命令，恢复/usr/lib/systemd/system/ctrl-alt-del.target、/etc/systemd/system/ctrl-alt-del.target文件：

```
[root@localhost ~]#  ln -sf /usr/lib/systemd/system/reboot.target /usr/lib/systemd/system/ctrl-alt-del.target
[root@localhost ~]#  ln -sf /usr/lib/systemd/system/reboot.target /etc/systemd/system/ctrl-alt-del.target
```

- 通过如下命令，设置为立即生效：

```
[root@localhost ~] systemctl daemon-reexec
```

修改影响

按下组合键后系统不再重启。

6.9 启用空闲锁屏时间

级别

要求

适用版本

V10 SP3 2403

说明

检查是否配置定时自动屏幕锁定（适用于具备图形界面的设备）。

> **说明** 开启空闲锁屏时间的一个主要好处是：它有助于防止未经授权的访问和保护个人信息不被泄露，从而提高设备的安全性和个人隐私保护。

检查方法

检查锁屏时间是否为15分钟。

修改方法

加固方法

- 通过如下命令，设置空闲启动屏保时间:

```
[root@localhost ~]# touch /etc/dconf/db/local.d/locks/02-lockidledelay
[root@localhost ~]# echo /org/mate/desktop/session/idle-delay >> /etc/dconf/db/local.d/locks/02-lockidledelay
```

- 通过如下命令，创建配置文件，将全局（所有用户）空闲启动屏保时间设定为15分钟，并添加"[org/mate/desktop/session]"和"idle-delay=15"两行内容到文件内:

```
[root@localhost ~]# touch /etc/dconf/db/local.d/02-set-idledelay
```

- 通过如下命令，功能立即生效:

```
[root@localhost ~]# dconf update
```

还原方法

- 通过如下命令，删除加固时创建的/etc/dconf/db/local.d/locks/02-lockidledelay文件:

```
[root@localhost ~]# rm -rf /etc/dconf/db/local.d/locks/02-lockidledelay
```

- 通过如下命令，删除加固时创建的/etc/dconf/db/local.d/02-set-idledelay文件:

```
[root@localhost ~]# rm -rf /etc/dconf/db/local.d/02-set-idledelay
```

- 通过如下命令，功能立即生效:

```
[root@localhost ~]# dconf update
```

修改影响

一定时间内无人操作系统会自动锁屏，再次进入系统需要用户重新登陆。

6.10 启用屏保

级别

要求

适用版本

V10 SP3 2403

说明

检查是否配置定时自动屏幕锁定（适用于具备图形界面的设备）。

> **说明** 在一定程度上降低被他人注意到屏幕内容的机会，从而提供一定的隐私保护。

检查方法

- 通过如下命令验证是否启用屏保：

```
[root@localhost ~]# gsettings get org.ukui.screensaver mode
'blank-only'
```

修改方法

加固方法

- 通过如下命令将屏保设置为blank-only：

```
[root@localhost ~]# gsettings set org.ukui.screensaver mode blank-only
```

还原方法

- 通过如下命令将屏保设置为原始值：

```
[root@localhost ~]# gsettings set org.ukui.screensaver mode <原始值>
```

修改影响

锁屏后屏保显示为黑屏。

6.11 禁止系统自动登录

级别

要求

适用版本

V10 SP3 2403

说明

采用强化管理的口令鉴别/基于令牌的动态口令鉴别/生物特征鉴别/数字证书鉴别等相结合的方式，使用多鉴别机制实现对用户身份的真实性鉴别，并在每次用户登录系统时和系统重新连接时进行鉴别。

> **说明** 配置该参数后autologin-user，Linux 系统将在下次引导时自动使用指定的用户名进行登录。以**root**用户为例。

检查方法

- 检查/etc/lightdm/lightdm.conf文件里的autologin-user=参数后是否存在有效用户：

```
[root@localhost ~]# cat /etc/lightdm/lightdm.conf | grep '^autologin-user'|awk -F "=" '{print $2}'
root
```

加固方法

- 编辑/etc/lightdm/lightdm.conf文件找到autologin-user=用户名行所在行，在行首添加"#"进行注释，在注释行之后添加以下内容：

```
autologin-user=
```

还原方法

- 编辑/etc/lightdm/lightdm.conf文件把加固时新增的行删除，再去掉查找到的autologin-user=用户名所在行的行首的"#"

修改影响

禁止系统自动登录可以增强安全性，但可能会对用户的便捷性和管理复杂性产生一定影响。在决定是否禁用自动登录时，需要根据具体需求和场景进行权衡和评估。

6.12 禁止SSH免密登录

级别

要求

适用版本

V10 SP3 2403

说明

检查系统openssh安全配置，禁止SSH免密登录。

检查方法

- 检查是否禁止SSH免密登录：

```
[root@localhost ~]# grep "^PubkeyAuthentication" /etc/ssh/sshd_config
PubkeyAuthentication no
```

修改建议

加固方法

- 编辑/etc/ssh/sshd_config文件找到PubkeyAuthentication所在行，在行首添加"#"进行注释，在注释行之后添加以下内容：

```
PubkeyAuthentication no
```

- 然后重启ssh服务：

```
[root@localhost ~]# systemctl restart sshd
```

还原方法

- 编辑/etc/ssh/sshd_config文件把加固时新增的行删除，再去掉查找到的PrintLastLog所在行的行首的"#"，重启ssh服务：

```
[root@localhost ~]# systemctl restart sshd
```

修改影响

禁止SSH 客户端将尝试使用用户的私钥与本服务器的公钥进行匹配。即不可以无密码地登录到本服务器。

6.13 设置守护进程的umask值

级别

要求

适用版本

V10 SP3 2403

说明

主体对客体的访问应遵循该客体的自主访问控制权限属性;将访问控制客体的颗粒度控制在文件和目录。

> **说明** umask值用来为新创建的文件和目录设置缺省权限。如果没有设定umask值，则生成的文件具有全局可写权限，存在一定的风险。守护进程负责系统上某个服务，让系统可以接受来自用户或者是网络客户的要求。为了提高守护进程所创建文件和目录的安全性，设置其umask值为0027。umask值代表的是权限的"补码"，umask值和权限的换算方法请参见umask值含义。

检查方法

- 判断/etc/sysconfig/init文件里是否包含umask，且其值为027，若不符合则需要加固

```
[root@localhost ~]# cat /etc/sysconfig/init | grep 'umask'|awk -F " " '{print $2}'
022
```

加固方法

- 在/etc/sysconfig/init文件里添加或修改如下行

```
[root@localhost ~]# vim /etc/sysconfig/init
 umask 027
```

还原方法

- 在/etc/sysconfig/init文件里删除或修改umask为初始值

```
[root@localhost ~]# vim /etc/sysconfig/init
 umask 027
```

修改影响

- 默认权限设置： umask 值决定了用户创建新文件或目录时的默认权限。将 umask 设置为 027，意味着新创建的文件将具有 070（即 rwx------）的权限，而新创建的目录将具有 077 （即 rwx------）的权限。
- 安全性增强： 更严格的 umask 设置（如 027）可以增强系统的安全性，因为它限制了新创建的文件和目录对其他用户的访问权限。这有助于防止未经授权的访问和数据泄露。
- 可能的兼容性问题： 如果某些应用程序或服务依赖于特定的文件权限设置，过于严格的 umask 可能会导致这些应用程序或服务无法正常运行。在这种情况下，您可能需要为这些特定的应用程序或服务单独设置 umask。
- 系统范围的影响： 修改 /etc/sysconfig/init 文件中的 umask 设置将影响整个系统的 umask 值，包括所有用户和进程。这可能会对系统中所有新建的文件和目录的权限产生全局性的影响。 请注意，由于 /etc/sysconfig/init 文件在现代 Linux 发行版中的弃用，建议您查阅您的系统文档以确定正确的 umask 设置方法。在 RHEL/CentOS 7 及更高版本中，如前所述，应修改 /etc/bashrc 或其他相应的 shell 配置文件来设置 umask。

6.14 限制多重并发会话数

级别

要求

适用版本

V10 SP3 2403

说明

多重并发会话限定的要求，限制系统并发会话的最大数量,并利用默认值作为会话次数的限定数。

> **说明** 操作系统必须将所有帐户和/或帐户类型的并发会话数限制为10

检查方法

- 查看/etc/security/limits.conf文件中是否包含maxlogins字段

```
[root@localhost ~]# cat /etc/security/limits.conf |grep maxlogins|grep -v "^#"
*        hard       maxlogins       10
```

修改建议

加固方法

- 编辑/etc/security/limits.conf并添加或编辑umask参数如下：

```
[root@localhost ~]# vim /etc/security/limits.conf
*        hard       maxlogins       10
```

还原方法

- 编辑/etc/security/limits.conf并删除或编辑maxlogins参数为之前的修改值

修改影响

设置了所有用户（* 表示所有用户）的最大并发登录数为 10。这意味着任何用户同时登录的数量不能超过 10 个。

7 潜在风险

7.1 检查空链接文件

级别

要求

适用版本

V10 SP3 2403

说明

如果系统存在空链接文件，可能会对网站或应用程序的正常运行产生负面影响。空链接文件是指那些已经失效或不存在的文件，这些链接无法正常访问，会导致用户无法正常访问网站或应用程序的某些功能。

> **说明** 及时发现和修复空链接文件是维护网站或应用程序正常运行的重要工作之一。通过使用工具、检查代码和文件路径、更新服务器配置以及定期维护和更新等措施，可以有效地解决空链接问题，提高网站或应用程序的用户体验。

检查方法

- 执行如下命令查看系统中是否存在空链接文件

```
[root@localhost ~]# find / ! -path '/proc/*' -xtype l ! -exec test -e {} \; -print
```

修改建议

加固方法

- 使用命令检查出来的文件可能会存在系统需要使用的情况，需要管理员判断相应文件是否为无用文件再删除。

修改影响

根据删除的文件情况而定。

7.2 检查不安全组件

级别

建议

适用版本

V10 SP3 2403

说明

建议不要集成net-snmp和tomcat

检查方法

- 执行如下命令查看系统中是否存在net-snmp和tomcat

```
[root@localhost ~]# rpm -qa net-snmp tomcat
net-snmp-5.9-4.p01.ky10.x86_64
tomcat-9.0.10-26.ky10.noarch
```

修改建议

加固方法

- 执行如下命令卸载net-snmp和tomcat

```
[root@localhost ~]# rpm -e net-snmp
[root@localhost ~]# rpm -e tomcat
```

还原方法

- 执行如下命令安装net-snmp和tomcat

```
[root@localhost ~]# yum install net-snmp
[root@localhost ~]# yum install tomcat
```

修改影响

- 建议不要集成net-snmp，snmp服务会明文传输信息
- 系统集成的tomcat包不使用不安全Realms，建议从集成镜像中移除tomcat包

7.3 检查可调试组件

级别

要求

适用版本

V10 SP3 2403

说明

系统中不应存在调试组件：tcpdump、gdb、strace、readelf、cpp、gcc、netcat、arecord、vnstat、vnStatsvg、nload、atop、iftop、objdump、aplay、eu-readelf、eu-objdump

检查方法

- 判断在/usr/bin/和usr/sbin/目录下是否存在相关命令文件，若存在则需手动加固，否则无需加固

```
[root@localhost ~]# ll /usr/bin/gdb
[root@localhost ~]# ll /usr/sbin/gdb
```

修改建议

加固方法

- 执行如下命令查询出命令所属包，然后卸载对应包

```
[root@localhost ~]# rpm -qf /usr/bin/gdb
gdb-9.2-3.p01.ky10.x86_64
[root@localhost ~]# rpm -e gdb
```

还原方法

- 执行如下命令安装之前卸载的rpm包

```
[root@localhost ~]# yum install gdb
```

修改影响

在卸载任何组件之前，建议先备份重要的数据和配置文件，以防意外情况发生。

7.4 /etc/aliases禁用不必要的别名

级别

要求

适用版本

V10 SP3 2403

说明

检查/etc/aliases是否禁用不必要的别名。

> **说明** 在 /etc/aliases 文件中，不必要的别名通常是指那些不再被使用或不再需要的别名。这些别名可能已经过时，或者由于某些原因被移除或更改。

检查方法

- 通过如下命令，检查/etc/aliases文件中除了"postmaster:root"行外，是否存在其他含"root"的配置项:

```
[root@localhost ~]# cat /etc/aliases|grep root |grep -v postmaster |grep -v '^#'
bin:        root
daemon:     root
adm:        root
lp:     root
sync:       root
shutdown:   root
halt:       root
mail:       root
news:       root
uucp:       root
operator:   root
games:      root
gopher:     root
ftp:        root
nobody:     root
radiusd:    root
nut:        root
(列举部分文件内容)
...
```

修改建议

加固方法

- 编辑/etc/aliases文件将除了"postmaster: root"行外含有root的行注释：

还原方法

- 去除加固时注释行前面的 "#"。

修改影响

在/etc/aliases文件中被注释掉的别名失效。

7.5 /etc/mail/aliases禁用不必要的别名

级别

要求

适用版本

V10 SP3 2403

说明

检查/etc/mail/aliases是否禁用不必要的别名。

> **说明** 在 /etc/mail/aliases 文件中，不必要的别名通常是指那些不再被使用或不再需要的别名。这些别名可能已经过时，或者由于某些原因被移除或更改。

检查方法

- 通过如下命令，检查/etc/mail/aliases文件中除了"postmaster:root"行外，是否存在其他含"root"的配置项:

```
[root@localhost ~]# cat etc/mail/aliases|grep root |grep -v postmaster |grep -v '^#'
```

修改建议

加固方法

- 编辑/etc/mail/aliases文件将除了"postmaster:root"行外含有root的行注释。

还原方法

- 去除加固时注释行前面的 "#"。

修改影响

在/etc/mail/aliases文件中被注释掉的别名失效。

7.6 删除潜在危险.netrc文件

级别

要求

适用版本

V10 SP3 2403

说明

是否删除.netrc 文件。

> **说明** 由于.netrc文件存储了敏感信息（如密码），如果未正确保护该文件的权限，可能会导致安全风险。

检查方法

- 通过如下命令验证是否存在.netrc文件：

```
[root@localhost ~]# find / -name .netrc 2>/dev/null
/home/KylinUser/.netrc
```

修改建议

加固方法

- 通过如下命令将.netrc文件名加固成.netrc.bak：

```
[root@localhost ~]# find / -name .netrc 2>/dev/null|xargs -I {} mv {} {}.bak
```

还原方法

- 通过如下命令将加固为.netrc.bak文件名还原成.netrc：

```
[root@localhost ~]# mv  /home/KylinUser/.netrc.bak  /home/KylinUser/.netrc
```

修改影响

- 删除.netrc文件后，支持.netrc文件的程序将无法自动提供登录所需的凭据，需要手动输入用户名和密码进行登录。

7.7 删除潜在危险hosts.equiv文件

级别

要求

适用版本

V10 SP3 2403

说明

是否删除hosts.equiv文件。

> **说明** 在hosts.equiv文件中，你可以指定允许访问本地主机的其他主机名或IP地址。

检查方法

- 通过如下命令验证是否存在hosts.equiv文件：

```
[root@localhost ~]#find / -name hosts.equiv 2>/dev/null
/home/KylinUser/hosts.equiv
```

修改建议

加固方法

- 通过如下命令将hosts.equiv文件名加固成hosts.equiv.bak：

```
[root@localhost ~]# find / -name hosts.equiv 2>/dev/null|xargs -I {} mv {} {}.bak
```

还原方法

- 通过如下命令将加固为hosts.equiv.bak文件名还原成hosts.equiv：

```
[root@localhost ~]# mv  /home/KylinUser/hosts.equiv.bak  /home/KylinUser/hosts.equiv
```

修改影响

修改hosts.equiv文件将会阻止任何远程主机无需密码就可以执行远程命令，不会影响登录到远程系统的功能。

7.8 删除潜在危险.rhosts文件

级别

要求

适用版本

V10 SP3 2403

说明

是否删除.rhosts文件。

> **说明** .rhosts 文件是一种 UNIX/Linux 系统中的特殊文本文件，它可以指定哪些用户可以从哪些主机无密码地访问本机资源。

检查方法

- 通过如下命令验证是否存在.rhosts文件：

```
[root@localhost ~]# find / -name .rhosts 2>/dev/null
/home/KylinUser/.rhosts
```

修改建议

加固方法

- 通过如下命令将.rhosts文件名加固成.rhosts.bak：

```
[root@localhost ~]# find / -name .rhosts 2>/dev/null|xargs -I {} mv {} {}.bak
```

还原方法

- 通过如下命令将加固为.rhosts.bak文件名还原成.rhosts：

```
[root@localhost ~]# mv  /home/KylinUser/.rhosts.bak  /home/KylinUser/.rhosts
```

修改影响

删除.rhosts文件将阻止从其他主机无密码地访问本系统。

7.9 关闭系统信任机制equiv

级别

要求

适用版本

V10 SP3 2403

说明

是否删除equiv文件。

> **说明** equiv文件主要用于信任远程登录的身份验证。具体来说，有两个重要的文件涉及到equiv： /etc/hosts.equiv： 这个文件定义了哪些计算机可以无密码地通过rlogin、rcp和rsh等命令访问本地计算机。它包含了一个主机名列表，这些主机被认为是可信任的，因此它们的用户可以不提供口令就执行远程操作。 ~/.rhosts： 对于每个用户，他们的家目录下都有一个.rhosts文件（通常是隐藏的），这个文件也用于无密码的远程访问。但是，与/etc/hosts.equiv不同的是，.rhosts文件只对单个用户生效，并且它列出了可以访问该用户的特定远程用户或主机。 这两个文件一起构成了"信任"机制的一部分，使得在某些情况下，系统管理员和普通用户可以在没有密码的情况下执行远程操作。然而，由于这种机制可能带来安全风险，许多现代系统已经不再使用这种方法，转而使用更安全的身份验证协议，如SSH。

检查方法

- 通过如下命令验证是否存在equiv文件：

```
[root@localhost ~]# find / -name equiv 2>/dev/null
/home/KylinUser/equiv
```

修改建议

加固方法

- 通过如下命令将equiv文件名加固成equiv.bak：

```
[root@localhost ~]# find / -name equiv 2>/dev/null|xargs -I {} mv {} {}.bak
```

还原方法

- 通过如下命令将加固为equiv.bak文件名还原成equiv：

```
[root@localhost ~]# mv  /home/KylinUser/equiv.bak  /home/KylinUser/equiv
```

修改影响

删除equiv文件将阻止从其他主机无密码地访问本系统。

7.10 关闭系统信任机制rhosts

级别

要求

适用版本

V10 SP3 2403

说明

是否删除rhosts文件。

> **说明** rhosts 文件是远程主机信任文件，用于指定哪些主机可以无密码登录到系统。如果这个文件被篡改或包含恶意主机，可能会导致安全风险。

检查方法

- 通过如下命令验证是否存在rhosts文件：

```
[root@localhost ~]# find / -name rhosts 2>/dev/null
/home/KylinUser/rhosts
```

修改建议

加固方法

- 通过如下命令将rhosts文件名加固成rhosts.bak：

```
[root@localhost ~]# find / -name rhosts 2>/dev/null|xargs -I {} mv {} {}.bak
```

还原方法

- 通过如下命令将加固为rhosts.bak文件名还原成rhosts：

```
[root@localhost ~]# mv  /home/KylinUser/rhosts.bak  /home/KylinUser/rhosts
```

修改影响

删除rhosts文件将阻止从其他主机无密码地访问本系统。

8 文件权限

8.1 删除无属组属主的文件或文件夹

级别

要求

适用版本

V10 SP3 2403

说明

查找系统中无用的文件，不再使用的文件可进行删除，以后需要使用的文件表明属主属组。

检查方法

- 在终端中输入命令：

```
[root@localhost ~]# find / \( -nouser -o -nogroup \) -exec ls -al {} \;
```

- 查看执行结果，是否存在可删除的文件

修改建议

加固方法

- 使用命令检查出来的文件可能会存在系统需要使用的情况，需要管理员判断相应文件是否为无用文件再删除。

修改影响

根据系统情况而定。在删除无属组属主的文件或文件夹之前，需要谨慎评估其影响，并采取相应的措施来降低风险。这包括备份重要数据、确认删除操作不会影响系统的正常运行、检查是否存在依赖关系等。

8.2 设置用户目录缺省访问权限

级别

要求

适用版本

V10 SP3 2403

说明

检查用户目录缺省访问权限设置。

检查方法

- 使用umask命令查看当前umask值

```
[root@localhost ~]# umask
0027
```

- 若umask值不为027，则查看/etc/bashrc文件中是否包含对应umask值:

```
[root@localhost ~]# cat /etc/bashrc |grep umask
umask 0027
```

修改建议

加固方法

- 编辑/etc/bashrc并添加或编辑umask参数如下：

```
[root@localhost ~]# vim /etc/bashrc
umask 027
```

还原方法

- 编辑/etc/bashrc并删除或编辑umask参数为之前的初始值

修改影响

- 对于新建的文件：

文件的所有者将具有读、写和执行（rwx）权限。 文件所在的组将具有读和执行（rx）权限。 其他用户将没有任何权限（---）。

- 对于新建的目录： 目录的所有者将具有读、写和执行（rwx）权限。 目录所在的组将具有读、写和执行（rwx）权限。 其他用户将只有执行（x）权限，不能读取或写入目录中的内容。

8.3 限制重要目录或文件权限

级别

要求

适用版本

V10 SP3 2403

说明

设置目录权限/etc/rc.d/init.d/ 750、/etc/xinetd.conf 600、/etc/rc5.d/ 750、/etc/rc4.d 750、/etc/security 600、/etc/shadow 600、/etc/passwd 644、/etc/grub.conf 600、/boot/grub/grub.conf 600、/etc/lilo.conf 600、/etc/grub2.cfg 600、/etc/rc0.d/ 750、/etc/rc6.d 750、/etc/rc1.d/ 750、/etc/rc2.d/ 750、/etc/group 644、/etc/services 644、/etc/rc3.d 750、/boot/efi/EFI/kylin/grub.cfg 600。

检查方法

- 运行以下命令并访问权限是否符合上述文件目录权限或者更严，如果与说明不匹配则需要加固：

```
[root@localhost ~]#  stat <目录>
权限：(0755/drwxr-xr-x)
```

修改建议

加固方法

- 运行以下命令设置文件目录的权限：

```
[root@localhost ~]#  chmod <权限>  <文件目录名称>
```

还原方法

- 运行以下命令设置成加固前stat命令查询出来的文件目录的权限：

```
[root@localhost ~]#  chmod <权限>  <文件目录名称>
```

修改影响

在限制重要目录或文件的权限时，需要权衡安全性和功能性。应确保必要的操作能够正常进行，同时提高系统的安全性。在设置权限时，应遵循最小权限原则，即只赋予用户和应用程序执行任务所需的最小权限。

8.4 限制日志文件权限

级别

要求

适用版本

V10 SP3 2403

说明

检查用户目录缺省访问权限设置。

检查方法

- 如果文件/etc/rsyslog.conf存在，检查参数$FileCreateMode及权限是否小于等于0640：

```
[root@localhost ~]# cat /etc/rsyslog.conf |grep "$FileCreateMode"
$FileCreateMode 0640
```

- 如果文件/etc/rsyslog.conf存在，检查参数$DirCreateMode及权限是否小于等于0755：

```
[root@localhost ~]#  cat /etc/rsyslog.conf |grep "$DirCreateMode"
$DirCreateMode 0755
```

- 如果文件/etc/rsyslog.conf不存在，文件/etc/syslog.conf存在检查参数$FileCreateMode及权限是否小于等于0640：

```
[root@localhost ~]#  cat/etc/syslog.conf |grep "$FileCreateMode"
$FileCreateMode 0640
```

- 如果文件/etc/rsyslog.conf不存在，文件/etc/syslog.conf存在，检查参数$DirCreateMode及权限是否小于等于0755：

```
[root@localhost ~]#  cat /etc/syslog.conf |grep "$DirCreateMode"
$DirCreateMode 0755
```

修改建议

加固方法

- 如果文件/etc/rsyslog.conf存在，用#注释$FileCreateMode行，添加如下内容：

```
$FileCreateMode 0640
```

- 如果文件/etc/rsyslog.conf存在，用#注释$DirCreateMode行，添加如下内容：

```
$DirCreateMode 0755
```

- 如果文件/etc/rsyslog.conf不存在，文件/etc/syslog.conf存在，用#注释$FileCreateMode行，添加如下内容：

```
$FileCreateMode 0640
```

- 如果文件/etc/rsyslog.conf不存在，文件/etc/syslog.conf存在，用#注释$DirCreateMode行，添加如下内容：

```
$DirCreateMode 0755
```

还原方法

- 编辑/etc/rsyslog.conf文件，根据加固情况删除"#"注释，删除新增的行。
- 编辑/etc/syslog.conf文件，根据加固情况删除"#"注释，删除新增的行。

修改影响

- $FileCreateMode 0640会产生以下影响： 文件的所有者将具有读、写和执行（rwx）权限。 文件所在的组将具有读取权限（r--）。 其他用户将只有读取权限（r--），不能修改或执行文件。
- $DirCreateMode 0755会产生以下影响： 目录的所有者将具有读、写和执行（rwx）权限。 目录所在的组将具有读取、写入和执行（rwx）权限。 其他用户将只能读取和执行（rx），不能写入目录中的内容。

8.5 限制FTP用户上传的文件所具有的权限

级别

要求

适用版本

V10 SP3 2403

说明

检查FTP用户上传的文件所具有的权限。

检查方法

- 运行以下命令，查看/etc/vsftpd/vsftpd.conf文件中参数local_umask是否大于等于022。

```
[root@localhost ~]# cat /etc/vsftpd/vsftpd.conf |grep local_umask
local_umask 022
```

修改建议

加固方法

- 如果/etc/vsftpd/vsftpd.conf文件中不存在local_umask参数添加如下行，如果存在local_umask参数将其用#注释，在注释后面添加如下行：

```
[root@localhost ~]# vim /etc/vsftpd/vsftpd.conf
local_umask 022
```

还原方法

- 根据加固情况删除加固时添加的如下行，并取消注释。

```
[root@localhost ~]# vim /etc/vsftpd/vsftpd.conf
local_umask 022
```

修改影响

用户登录 FTP 服务器时创建文件和目录将具有以下初始权限：

- 文件：644（所有者可读写，同组和其他用户只读）
- 目录：755（所有者可读写执行，同组和其他用户可读执行）

8.6 禁止日志文件全局可读写

级别

要求

适用版本

V10 SP3 2403

说明

禁止日志文件全局可读写。

检查方法

- 运行以下命令查看/var/log目录下组有wx权限或者other有rwx权限的文件，如果有这样的文件存在，提示需要加固，并将文件名添加到vector中。否则无需加固。

```
[root@localhost ~]# find /var/log -type f -perm /g+wx,o+rwx -exec ls -l {} \;| awk '{print $1,$9}'
-rw-rw-r-- /var/log/lastlog
-rw-r--r-- /var/log/README
-rw-rw-r-- /var/log/wtmp
-rw-rw---- /var/log/btmp
-rw-rw-r-- /var/log/firebird/firebird.log
-rw-r--r-- /var/log/tuned/tuned.log
-rw-r--r-- /var/log/sa/sa10
-rw-r--r-- /var/log/sa/sa29
-rw-r--r-- /var/log/yhkydefenderd_monitor.log
-rw-r--r-- /var/log/yhkydefenderd.log
-rw-r--r-- /var/log/dracut.log
-rw-r--r-- /var/log/kydima.log
-rw-r--r-- /var/log/ky_kms_activation.log
-rw-r--r-- /var/log/Xorg.9.log
-rw-r--r-- /var/log/systemtap.log
-rw-r--r-- /var/log/kysec.log
-rw-r--r-- /var/log/Xorg.0.log
-rw-r--r-- /var/log/activation.log
-rw-r--r-- /var/log/dnf.log
-rw-r--r-- /var/log/dnf.librepo.log
-rw-r--r-- /var/log/dnf.rpm.log
-rw-r--r-- /var/log/hawkey.log
-rw-r--r-- /var/log/rpmpkgs
```

修改建议

加固方法

- 运行以下命令去除组的wx权限和other的所有权限：

```
[root@localhost ~]# find /var/log -type f -exec chmod g-wx,o-rwx {} \;
```

还原方法

- 根据系统初始权限值，使用chmod命令恢复：

```
[root@localhost ~]# chmod <权限> <文件>
```

修改影响

在/var/log目录下的文件的组无wx权限、other无rwx权限。

9 风险账户

9.1 删除与设备运行、维护等工作无关的账号

级别

要求

适用版本

V10 SP3 2403

说明

检查是否删除与设备运行、维护等工作无关的账号。

检查方法

- 检查/etc/shadow中是否存在用户（adm，lp，mail，uucp，operator，games，gopher，ftp，nobody，nobody4，noaccess，listen，webservd，rpm，dbus，avahi，mailnull，smmsp，nscd，vcsa，rpc，rpcuser，nfs，sshd，pcap，ntp，haldaemon，distcache，apache，webalizer，squid，xfs，gdm，sabayon，named等）且未锁定。

```
[root@localhost ~]# passwd -S dbus |awk '{if($2 !="LK" && $2 != "L")print $1}'|grep -v "#"
dbus
```

修改建议

加固方法

- 运行以下命令进行用户锁定：

```
[root@localhost ~]# usermod -L <用户名>
```

还原方法

- 对之前加固的用户，运行以下命令解除锁定

```
[root@localhost ~]# passwd -u -f <用户名>
```

修改影响

加固后被锁定用户无法登录。

9.2 删除空口令账号

级别

要求

适用版本

V10 SP3 2403

说明

检查是否存在空口令账号，对存在的空口令账号加固，可防止系统被黑客所创建的账户登录或遭到黑客暴力破解。

检查方法

- 在/etc/passwd查找secadm、auditadm，root,和 uid大于1000的用户:

```
[root@localhost ~]# cat /etc/passwd|awk -F ':' \
'{if(($3>=1000 || $3==0 || $1=="auditadm" || $1=="secadm") && $1 != "nobody") print $1}'
```

- 检查/etc/shadow，查看之前获取的用户是否有设置密码（$user换成具体要检查的用户）：

```
[root@localhost ~]# grep "$user" /etc/shadow|awk -v str="$user" -F ':' '{if($1==str && $2==NULL) print $1}'
```

修改建议

加固方法

- 运行以下命令对空密码账户进行锁定并提示用户进行密码设置

```
[root@localhost ~]# usermod -L <用户名>
```

还原方法

- 对之前加固的用户运行以下命令解除锁定：

```
[root@localhost ~]# passwd -u -f <用户名>
```

修改影响

加固后被锁定用户无法登录。

9.3 禁止系统账号进行交互式登录

级别

要求

适用版本

V10 SP3 2403

说明

禁止系统账号进行交互式登录。

检查方法

- 检查/etc/passwd文件，除了root/secadm/auditadm/sync/shutdown/halt账户外，uid小于1000，登录权限不是"/usr/sbin/nologin","/bin/false","/sbin/nologin"的账户
- 检查上述用户passwd -S检测未锁定的账户。

```
[root@localhost ~]# passwd -S mail
mail PS 1969-12-31 0 99999 7 -1 (密码已设置，使用 SHA512 算法。)
```

修改建议

加固方法

- 对于检查出的需要加固的用户，替换最后一个字段为/sbin/nologin。

还原方法

- 使用备份的原有数据替换/etc/passwd中的现有数据。

修改影响

禁止系统账号进行交互式登录的好处主要集中在增强安全性、简化管理、减少不必要的登录活动和提高资源利用率等方面。但需要注意的是，对于需要频繁与系统交互的用户或开发者，这可能会带来一些不便。因此，在实施禁止交互式登录的措施时，需要权衡安全性和功能性需求。

9.4 删除UID重复账号

级别

要求

适用版本

V10 SP3 2403

说明

禁止用户存在相同的uid

> **说明** 禁用相同用户的 UID 不仅可以提升系统的可靠性和安全性，还能简化管理任务，有利于保证整个系统的高效稳定运行。以自建用户**kylinuser01:1002（用户名:UID）**为例。

检查方法

- 查看/etc/passwd中uid>=1000和uid=0的用户:

```
[root@localhost ~]# awk -F ":" '$3 >= 1000 || $3 == "0"{ print $1":"$3 }' /etc/passwd
kylinuser01:1002
```

- 查看/etc/uid_list中uid>=1000和uid=0的用户:

```
[root@localhost ~]# awk -F ":" '$2 >= 1000 ||  $3 == "0"{ print $1":"$2 }' /etc/uid_list
kylinuser01:1002
```

- 根据上述运行结果，对比两个文件相同名字的用户uid是否相同，如果运行结果中所有相同名字的用户uid均一致，无需加固；如果存在用户名相同，uid不用同的用户，则需要加固。

加固方法

*执行如下命令锁定/etc/passwd与/etc/uid_list中所有用户名相同，uid不用同的用户：

```
[root@localhost ~]# usermod -L <用户名>
```

还原方法

- 将加固时锁定的用户解锁：

```
[root@localhost ~]# passwd -u -f <用户名>
```

修改影响

被锁定的用户将不能通过常规的方式登录到系统中，即使输入正确的用户名和密码也无法成功登录。

9.5 启用用户标识唯一性

级别

要求

适用版本

V10 SP3 2403

说明

操作系统用户标识应使用用户名和 UID,并在操作系统的整个生存周期实现用户的唯一性标识，以及用户名或别名、UID 等之间的一致性。以**off**为例。

检查方法

- 查看/etc/chkuid_state的中开关的状态，off需要加固，on无需加固:

```
[root@localhost ~]#  awk -F "=" '{ print $2 }' /etc/chkuid_state
off
```

加固方法

- 将/etc/chkuid_state的中开关的状态改为on：

```
[root@localhost ~]# sed -i 's/state=off/state=on/g' /etc/chkuid_state
```

还原方法

- 将/etc/chkuid_state的中开关的状态改为off：

```
[root@localhost ~]# sed -i 's/state=on/state=off/g' /etc/chkuid_state
```

修改影响

开启开关后usermod -u uid <用户名> -o命令无法设置相同uid的用户。

10 磁盘检查

10.1 检查系统磁盘分区使用率

级别

要求

适用版本

V10 SP3 2403

说明

检查系统磁盘分区使用率。

> **说明** 麒麟建议将磁盘使用率控制在80%以内，如果超过80%则需要扩容。以系统情况**30%**为例。

检查方法

- 运行以下命令，如果运行结果大于80%，则需要加固；小于80%无需加固：

```
[root@localhost ~]# df -h |grep -w '/' |awk '{print $5}'|awk -F '%' '{print $1}'
14
```

修改建议

加固方法

- 当磁盘使用率超过80%时，提醒用户扩容。

修改影响

定期检查和控制磁盘分区使用率有助于维护系统健康状态，提高其可用性和安全性。

11 密码强度

11.1 设置系统密码复杂度

级别

要求

适用版本

V10 SP3 2403

说明

检查设备密码复杂度策略。

检查方法

- 运行以下命令，检查/etc/security/pwquality.conf中minclass后的参数是否为3：

```
[root@localhost ~]# cat /etc/security/pwquality.conf |grep minclass
minclass = 3
```

修改建议

加固方法

- 在/etc/security/pwquality.conf文件中，将查找到的行在行首添加"#"进行注释，在注释行之后添加以下内容：

```
minclass = 3
```

还原方法

- 把新增的行全部删除，再去掉查找到的行的行首的"#"。

修改影响

配置minclass = 3后表示一个合法的用户密码至少需要包含来自三个不同字符类别的字符。这些类别通常是： 小写字母（lowercase letters） 大写字母（uppercase letters） 数字（digits） 特殊字符（special characters）

11.2 设置口令过期前警告天数

级别

要求

适用版本

V10 SP3 2403

说明

检查是否设置口令过期前警告天数。

> **说明** PASS_WARN_AGE是一个配置选项，它定义了密码过期前多少天开始提示用户需要更改密码。

检查方法

- 运行以下命令并验证PASS_WARN_AGE是否为30或更长：

```
[root@localhost ~]# grep PASS_WARN_AGE /etc/login.defs
PASS_WARN_AGE 30
```

- chage -l <用户名>查看密码过期前警告天数是否为30或更长

```
[root@localhost ~]# chage -l <user>
在密码过期之前警告的天数  : 30
```

修改建议

加固方法

- 在/etc/login.defs中将查找到的行在行首添加"#"进行注释，在注释行之后添加以下内容：

```
PASS_WARN_AGE 30
```

- 修改密码设置为匹配的所有用户的密码过期前警告天数

```
[root@localhost ~]# chage --warndays 30 <user>
```

还原方法

- 把加固时新增的行全部删除，再去掉查找到的行的行首的"#"。
- 先取得检查方法中chage -l获取的用户密码过期前警告天数，调用chage --warndays命令重新设置时间。

修改影响

从密码即将过期的第30天开始，系统会向用户发出警告，提醒他们尽快更改密码。

11.3 开启密码复杂度策略

级别

要求

适用版本

V10 SP3 2403

说明

检查设备是否开启密码复杂度策略（密码强度功能、密码禁止包含用户名功能、密码回文检查功能、密码相似性检查功能、密码字典检查功能）功能。

检查方法

- 运行以下命令，并验证是否开启密码强度功能，配置则符合:

```
[root@localhost ~]# cat /etc/pam.d/system-auth|grep enforce_for_root |grep pam_pwquality
password requisite pam_pwquality.so try_first_pass local_user_only enforce_for_root
```

- 检查/etc/security/pwquality.conf中参数"palindromic"是否未屏蔽，未屏蔽则符合。
- 检查/etc/security/pwquality.conf中参数"usercheck"是否为"1"，1符合，0不符合，屏蔽也不符合。
- 检查/etc/security/pwquality.conf中参数"no_similar_check"是否屏蔽，屏蔽则符合。
- 检查/etc/security/pwquality.conf中参数""dictcheck"是否为"1"，1符合，0不符合,屏蔽也不符合。

修改建议

加固方法

- 在/etc/pam.d/system-auth中找到对应行，然后在后面增加字段enforce_for_root：

```
password requisite pam_pwquality.so try_first_pass local_user_only enforce_for_root
```

- 在/etc/security/pwquality.conf中把查找到的行在行首添加"#"进行注释，在注释行之后添加以下内容：

```
dictcheck=1
usercheck=1
palindromic
#no_similar_check   //屏蔽no_similar_check
```

还原方法

- 在/etc/pam.d/system-auth中找到对应行，删除字段enforce_for_root：

```
password requisite pam_pwquality.so try_first_pass local_user_only
```

- 把加固时新增的行全部删除，去掉查找到的行的行首的"#"。

修改影响

设置密码时，密码禁止包含用户名功能、密码回文检查功能、密码相似性检查功能、密码字典检查功能。

11.4 限制口令生存周期

级别

要求

适用版本

V10 SP3 2403

说明

检查是否设置口令过期前警告天数。

> **说明** PASS_MAX_DAYS是一个配置选项，它定义了用户密码的最大有效期。

检查方法

- 运行以下命令并验证PASS_MAX_DAYS是否为30或更长：

```
[root@localhost ~]# grep PASS_MAX_DAYS /etc/login.defs
PASS_MAX_DAYS 90
```

- chage -l <用户名>查看警告天数是否为30或更长

```
[root@localhost ~]# chage -l <user>
两次改变密码之间相距的最大天数      : 90
```

修改建议

加固方法

- 在/etc/login.defs中将查找到的行在行首添加"#"进行注释，在注释行之后添加以下内容：

```
[root@localhost ~]#vim /etc/login.defs
PASS_MAX_DAYS 90
```

- 修改密码设置为匹配的所有用户的密码的最大有效期

```
[root@localhost ~]#  chage --maxdays 90 <user>
```

还原方法

- 把加固时新增的行全部删除，再去掉查找到的行的行首的"#"。
- 先取得检查方法中chage -l获取的密码的最大有效期，调用chage --maxdays命令重新设置时间。

修改影响

当PASS_MAX_DAYS被设置为90时，意味着从密码更改之日起算，用户必须在90天内更改他们的密码。

11.5 限制口令最小长度

级别

要求

适用版本

V10 SP3 2403

说明

检查口令最小长度。

检查方法

- 运行命令验证/etc/security/pwquality.conf的参数minlen是否为8或8位以上，且未屏蔽。

```
[root@localhost ~]# cat /etc/security/pwquality.conf |grep minlen
minlen = 8
```

修改建议

加固方法

- 在/etc/security/pwquality.conf中把查找到的行在行首添加"#"进行注释，在注释行之后添加以下内容：

```
minlen = 8
```

还原方法

- 把加固时新增的行全部删除，再去掉查找到的行的行首的"#"。

修改影响

用户设置的密码必须至少包含 8 个字符。

11.6 限制口令更改最短间隔

级别

要求

适用版本

V10 SP3 2403

说明

检查是否设置口令生存周期。

> **说明** PASS_MIN_DAYS 是一个配置选项，它定义了一个密码最小使用天数，即创建或更改密码后必须等待的最短天数。

检查方法

- 运行以下命令并验证PASS_MIN_DAYS是否为6天或6天以上：

```
[root@localhost ~]# grep PASS_MIN_DAYS /etc/login.defs
PASS_MIN_DAYS 6
```

- chage -l <用户名>查看两次改变密码之间相距的最小天数是否为6天或6天以上

```
[root@localhost ~]# chage -l <user>
两次改变密码之间相距的最小天数      : 6
```

修改建议

加固方法

- 在/etc/login.defs中将查找到的行在行首添加"#"进行注释，在注释行之后添加以下内容：

```
[root@localhost ~]#vim /etc/login.defs
PASS_MIN_DAYS 6
```

- 修改密码设置为匹配的所有用户的两次改变密码之间相距的最小天数

```
[root@localhost ~]#  chage --mindays 6 <user>
```

还原方法

- 把加固时新增的行全部删除，再去掉查找到的行的行首的"#"。
- 先取得检查方法中chage -l获取的两次改变密码之间相距的最小天数，调用chage --mindays命令重新设置时间。

修改影响

用户需要至少等待6天才能更改他们的密码。

11.7 限制密码重复使用次数

级别

要求

适用版本

V10 SP3 2403

说明

检查密码重复使用次数限制。

检查方法

- 运行以下命令，在/etc/pam.d/system-auth查找password ， pam_pwhistory和remember所在行：

```
[root@localhost ~]# egrep "pam_pwhistory.so" /etc/pam.d/system-auth|grep password|grep remember
password requisite pam_pwhistory.so remember=5 enforce_for_root
```

修改建议

加固方法

- 编辑/etc/pam.d/system-auth，找到类似行password sufficient pam_unix.so sha512 shadow nullok try_first_pass use_authtok，在该行上一行添加或修改密码历史策略：

```
[root@localhost ~]# vim /etc/pam.d/system-auth
password requisite pam_pwhistory.so remember=5 enforce_for_root
```

还原方法

- 编辑/etc/pam.d/system-auth，删除或修改密码历史策略：

```
[root@localhost ~]# vim /etc/pam.d/system-auth
password requisite pam_pwhistory.so remember=5 enforce_for_root
```

修改影响

- pam_pwhistory.so：这是一个PAM模块，用于检查用户的新密码是否与最近使用的某些密码相同。如果新密码在历史记录中，则会拒绝更改密码。
- remember=5：这意味着系统将记住用户最近的5个密码。当用户尝试设置新密码时，如果新密码与这5个历史密码中的任何一个相同，系统将拒绝更改密码。
- enforce_for_root：这个选项表示root用户也必须遵守上述规则。如果不包含这个选项，root用户可能会绕过这些密码历史规则。

11.8 加强口令的密码算法

级别

要求

适用版本

V10 SP3 2403

说明

口令应是不可见的，在存储和传输时进行安全保护,确保其不被非授权的访问、修改和删除

> **说明:**
> 出于系统安全考虑，口令不允许明文存储在系统中，应该加密保护。在不需要还原口令的场景，必须使用不可逆算法加密。设置口令的加密算法为sha512。通过上述设置可以有效防范口令泄露，保证口令安全。

检查方法

- 判断/etc/pam.d/system-auth文件中password和pam_unix.so所在行是否包含sha512，若包含则无需加固

```
[root@localhost ~]# cat /etc/pam.d/system-auth |grep password|grep pam_unix
password    sufficient    pam_unix.so sha512 shadow nullok try_first_pass use_authtok
```

- 判断/etc/login.defs文件中ENCRYPT_METHOD参数值是否为SHA512，若是则无需加固

```
[root@localhost ~]# cat /etc/login.defs|grep ENCRYPT_METHOD
ENCRYPT_METHOD SHA512
```

- 两者都不符合则需加固

修改建议

加固方法

- 在/etc/pam.d/system-auth文件中添加或修改如下行

```
[root@localhost ~]# vim /etc/pam.d/system-auth
password    sufficient    pam_unix.so sha512 shadow nullok try_first_pass use_authtok
```

还原方法

- 在/etc/pam.d/system-auth文件中删除或修改为初始值

修改影响

加强口令的密码算法的加固可以提高系统的安全性、保护数据、提高用户体验和降低风险。

12 账户锁定

12.1 设置账户登录失败锁定功能

级别

要求

适用版本

V10 SP3 2403

说明

为了保障用户系统的安全，建议用户设置口令出错次数的阈值（建议3次），以及由于口令尝试被锁定用户的自动解锁时间（建议600秒）。

用户锁定期间，任何输入被判定为无效，锁定时间不因用户的再次输入而重新计时；解锁后，用户的错误输入记录被清空。通过上述设置可以有效防范口令被暴力破解，增强系统的安全性。

> **说明**
> kylin默认口令出错次数的阈值为3次，系统被锁定后自动解锁时间为600秒，在/etc/pam.d/system-auth文件与/etc/pam.d/password-auth文件中deny、unlock_time参数需保持同样的配置。以两个文件均不存在该项配置为例。

检查方法

*执行以下操作检查/etc/pam.d/system-auth文件用户锁定的当前设置，deny 小于等于3次，锁定时间大于等于600秒算符合：

```
[root@localhost ~]# grep "pam_faillock" /etc/pam.d/system-auth | grep deny|grep "unlock_time"
```

- 执行以下操作检查/etc/pam.d/password-auth文件用户锁定的当前设置，deny 小于等于3次，锁定时间大于等于600秒算符合：

```
[root@localhost ~]# grep "pam_faillock" /etc/pam.d/password-auth | grep deny|grep "unlock_time"
```

- 在/etc/pam.d/system-auth文件与/etc/pam.d/password-auth文件中配置都满足则无需加固，否则需要加固。

修改建议

加固方法

- 把查找到的行在行首添加"#"进行注释，在注释行之后添加如下内容；如果未查找到行则在文件开头添加下面的行，编辑/etc/pam.d/system-auth、/etc/pam.d/password-auth修改为如下策略：

```
  auth requisite      pam_faillock.so preauth  audit deny=3 even_deny_root fail_interval=900 unlock_time=600
  auth [default=die] pam_faillock.so authfail audit deny=3 even_deny_root fail_interval=900 unlock_time=600
  auth sufficient    pam_faillock.so authsucc audit deny=3 even_deny_root fail_interval=900 unlock_time=600
```

还原方法

把加固时新增的行全部删除，再去掉注释行的行首的"#"。

加固项影响

帐户被锁定后，必须等待60秒才能登录系统。

pam_faillock.so配置项说明表

配置项	说明
preauth	一个特殊的标志，可以用于配置 PAM 模块，使其在正式身份验证过程开始之前运行。
authfail	捕获用户登录失败的事件。
deny=3	用户连续登录失败次数超过3次即被锁定。
even_deny_root	同样限制root帐户。
fail_interval=900	两次认证失败事件之间的最小间隔时间900秒（即15分钟）。
unlock_time=600	普通用户自动解锁时间为600秒（即10分钟）。

13 系统安全

13.1 开启安全防护功能-开启管理员分权功能

级别

建议

适用版本

V10 SP3 2403

说明

检查系统管理员分权功能是否开启

检查方法

- 执行security-switch --get查看三权分立是否启用，若未启用，则需加固

```
[root@localhost ~]# security-switch --get

当前安全级别 : 安全优先(strict)
-------------------------------------------------
  安全模块      |      当前状态      |      默认状态
-------------------------------------------------
  Kysec           启用(Enforcing)     启用(Enforcing)
  SELinux         启用(Enforcing)     启用(Enforcing)
  三权分立        启用                启用
```

修改建议

加固方法

- 执行security-switch --set strict开启三权

```
[root@localhost ~]# security-switch --set strict
正在为 'Kysec' 检查内核配置... OK
正在为 'Kysec' 检查软件包安装状态... OK
正在为 'Kysec' 检查文件安装状态... OK
正在为开启 'Kysec' 检查系统配置状态... OK
正在为 'SELinux' 检查内核配置... OK
正在为 'SELinux' 检查软件包安装状态... OK
正在为 'SELinux' 检查文件安装状态... OK
正在为开启 'SELinux' 检查系统配置状态... OK
正在为 '三权分立' 检查内核配置... OK
正在为 '三权分立' 检查软件包安装状态... OK
正在为 '三权分立' 检查文件安装状态... OK
正在为开启 '三权分立' 检查系统配置状态... OK
正在启用 'Kysec' ... OK
正在启用 'SELinux' ... OK
root 用户密码需要手动设置
请输入新密码:
请再次输入新密码进行确认:
secadm 用户密码需要手动设置
请输入新密码:
请再次输入新密码进行确认:
auditadm 用户密码需要手动设置
请输入新密码:
请再次输入新密码进行确认:
正在启用 '三权分立' ... OK
正在更新系统启动配置... OK
正在写入系统安全配置... OK
系统安全级别切换成功，请立即重启系统生效!!!
```

还原方法

- 执行security-switch --set none关闭三权

```
[secadm@localhost ~]$ security-switch --set none
正在为关闭 'Kysec' 检查系统配置状态... OK
正在为关闭 'SELinux' 检查系统配置状态... OK
正在为关闭 '三权分立' 检查系统配置状态... OK
正在禁用 'Kysec' ... OK
正在禁用 'SELinux' ... OK
正在禁用 '三权分立' ... OK
正在更新系统启动配置... OK
正在写入系统安全配置... OK
系统安全级别切换成功，请立即重启系统生效!!!
```

修改影响

开启管理员分权功能可以在提高安全性和可审计性的同时，增加管理的复杂性和沟通成本。因此，在决定是否启用此功能时，需要综合考虑组织的安全需求、管理流程和人员能力等因素。

13.2 开启安全防护功能-开启应用执行控制功能

级别

建议

适用版本

V10 SP3 2403

说明

检查系统安全功能是否开启,检查应用执行控制功能是否开启

检查方法

- 执行getstatus -m kysec获取kysec状态，当kysec状态为disable时，需要加固

```
[root@localhost ~]# getstatus -m kysec
KySec status: disable
```

- 执行getstatus -m kysec获取kysec状态，当exectl状态不为on时，需要加固

```
[root@localhost 桌面]# getstatus -m kysec
KySec status: enforcing
     exectl  : on
     netctl  : on
     fpro    : on
     kmod    : on
     ppro    : on
```

修改建议

加固方法

- 执行如下命令开启kysec

```
[root@localhost ~]# setstatus kysec -s enforcing
安全模块状态设置改变，需要重启系统才能生效!
```

- 执行如下命令开启应用执行控制

```
[root@localhost ~]# setstatus kysec -f exectl -c on
```

还原方法

- 若之前未开启kysec，执行如下命令关闭kysec

```
[root@localhost ~]# setstatus kysec -s disable
kysec已成功设置为disable，此设置将使系统失去安全防护，极大增加系统安全风险!
```

- 若最初开启了kysec，但未开启执行控制，执行如下命令关闭应用执行控制

```
[root@localhost ~]# setstatus kysec -f exectl -c off
```

修改影响

开启应用执行控制功能可以在提高企业的安全性方面发挥重要作用，但同时也带来了管理复杂性、工作影响以及持续维护等方面的挑战。因此，在实施应用执行控制功能之前，需要充分评估其对企业的影响，并进行适当的规划、配置和培训。

13.3 开启安全防护功能-开启应用防护控制功能

级别

建议

适用版本

V10 SP3 2403

说明

检查系统安全功能是否开启,检查应用防护控制功能是否开启

检查方法

- 执行getstatus -m kysec获取kysec状态，当kysec状态为disable时，需要加固

```
[root@localhost ~]# getstatus -m kysec
KySec status: disable
```

- 执行getstatus -m kysec获取kysec状态，当kmod、fpro、ppro状态不为on时，需要加固

```
[root@localhost 桌面]# getstatus -m kysec
KySec status: enforcing
     exectl  : on
     netctl  : on
     fpro    : on
     kmod    : on
     ppro    : on
```

修改建议

加固方法

- 执行如下命令开启kysec

```
[root@localhost ~]# setstatus kysec -s enforcing
安全模块状态设置改变，需要重启系统才能生效!
```

- 执行如下命令开启应用防护控制

```
[root@localhost ~]# setstatus kysec -f kmod -c on
[root@localhost ~]# setstatus kysec -f fpro -c on
[root@localhost ~]# setstatus kysec -f ppro -c on
```

还原方法

- 若之前未开启kysec，执行如下命令关闭kysec

```
[root@localhost ~]# setstatus kysec -s disable
kysec已成功设置为disable，此设置将使系统失去安全防护，极大增加系统安全风险!
```

- 若最初开启了kysec，但未开启应用防护控制，执行如下命令关闭应用防护控制

```
[root@localhost ~]# setstatus kysec -f kmod -c off
[root@localhost ~]# setstatus kysec -f fpro -c off
[root@localhost ~]# setstatus kysec -f ppro -c off
```

修改影响

开启应用防护控制功能可以在提高电脑或企业安全性方面发挥重要作用，但同时也带来了管理复杂性、工作影响以及持续维护等方面的挑战。因此，在实施应用防护控制功能之前，需要充分评估其对企业的影响，并进行适当的规划、配置和培训。

13.4 开启安全防护功能-开启应用联网控制功能

级别

建议

适用版本

V10 SP3 2403

说明

检查系统安全功能是否开启,检查应用联网控制是否开启

检查方法

- 执行getstatus -m kysec获取kysec状态，当kysec状态为disable时，需要加固

```
[root@localhost ~]# getstatus -m kysec
KySec status: disable
```

- 执行getstatus -m kysec获取kysec状态，当netctl状态不为on时，需要加固

```
[root@localhost 桌面]# getstatus -m kysec
KySec status: enforcing
     exectl  : on
     netctl  : on
     fpro    : on
     kmod    : on
     ppro    : on
```

修改建议

加固方法

- 执行如下命令开启kysec

```
[root@localhost ~]# setstatus kysec -s enforcing
安全模块状态设置改变，需要重启系统才能生效!
```

- 执行如下命令开启应用联网控制

```
[root@localhost ~]# setstatus kysec -f netctl -c on
```

还原方法

- 若之前未开启kysec，执行如下命令关闭kysec

```
[root@localhost ~]# setstatus kysec -s disable
kysec已成功设置为disable，此设置将使系统失去安全防护，极大增加系统安全风险!
```

- 若最初开启了kysec，但未开启执行控制，执行如下命令关闭应用联网控制

```
[root@localhost ~]# setstatus kysec -f netctl -c off
```

修改影响

开启应用联网控制功能可以在提高电脑或企业安全性方面发挥作用，同时，开启应用联网控制功能也可能对用户体验产生影响，例如某些应用需要联网以提供服务或功能，如果限制了应用的联网权限，可能会影响其正常运行或使用。

13.5 开启保护箱开关功能

级别

建议

适用版本

V10 SP3 2403

说明

检查保护箱开关功能是否开启

检查方法

- 执行getstatus -m box获取kysec状态，当box状态为disable时，需要加固

```
[root@localhost ~]# getstatus -m box
box status: disable
```

修改建议

加固方法

- 执行如下命令开启box

```
[root@localhost ~]# setstatus box -s enable
安全模块状态设置改变，需要重启系统才能生效!
```

还原方法

- 执行如下命令关闭box

```
[root@localhost ~]# setstatus box -s disable
安全模块状态设置改变，需要重启系统才能生效!
```

修改影响

文件保护箱是一个非常有用的工具，可以有效地保护您的文件安全性和正确性，并适用于各种需要保护文件的应用场景。

13.6 开启防火墙功能

级别

建议

适用版本

V10 SP3 2403

说明

开启系统firewalld.service服务。

> **说明：**
> 开启系统firewalld.service服务，增强系统防御。

检查方法

- 通过如下命令验证firewalld.service服务是否自启动：

```
[root@localhost ~]#  systemctl is-enabled firewalld.service
enable
```

- 通过如下命令验证firewalld.service服务是否开启：

```
[root@localhost ~]#  systemctl is-active firewalld.service
active
```

修改建议

加固方法

- 通过如下命令设置firewalld.service服务自启动：

```
[root@localhost ~]#  systemctl enable firewalld.service
```

- 通过如下命令设置firewalld.service服务开启：

```
[root@localhost ~]#  systemctl start firewalld.service
```

还原方法

- 通过如下命令设置firewalld.service服务禁止自启动：

```
[root@localhost ~]# systemctl disable firewalld.service
```

- 通过如下命令设置firewalld.service服务禁止开启：

```
[root@localhost ~]# systemctl stop firewalld.service
```

修改影响

- 防火墙可以监控进出系统的网络流量，防止未经授权的访问和恶意活动。
- 它可以阻止不符合策略的网络流量，例如来自已知恶意IP地址或端口的通信。
- 防火墙通过配置规则来实施企业的安全政策，确保只有符合规定的网络连接被允许。
- 它可以帮助控制内部网络中不同区域之间的数据流动。
- 防火墙可以通过分析网络流量中的模式和签名来识别潜在的攻击行为。
- 一旦发现异常活动，防火墙可以采取行动，如阻断连接或发送警报。
- 防火墙可以防止未经授权的应用程序和服务访问用户的网络连接，从而保护个人隐私信息。

13.7 启用SELinux

级别

建议

适用版本

V10 SP3 2403

说明

SELinux是一个内核级别的安全机制，是对于强制访问控制的实现。在这种访问控制体系的限制下，进程只能访问那些在他的任务中所需要文件，提供了强有力的安全保护。

检查方法

- 在终端中输入命令：

```
[root@localhost ~]# getenforce
Disabled
```

- 根据显示结果检查当前SELinux是否开启

```
Disabled：关闭
Permissive：宽松模式（仅告警）
Enforcing：强制模式
```

修改建议

加固方法

- 执行以下命令开启selinux

```
[root@localhost ~]# setstatus selinux -s enforcing
```

- 重启系统生效

还原方法

- 执行以下命令关闭selinux

```
[root@localhost ~]# setstatus selinux -s disable
```

- 重启系统生效

修改影响

启用SELinux可以增强系统的安全性，但也可能对系统性能和功能产生一定的影响。在启用SELinux之前，需要仔细评估其对系统和应用程序的影响，并进行适当的配置和管理。

13.8 配置用户登录的访问规则

级别

建议

适用版本

V10 SP3 2403

说明

会话建立机制的要求，根据访问地址或端口，允许或拒绝用户的登录。鉴别机制不准许被旁路。系统应提供一种机制，按时间、进入方式、地点、网络地址或端口等条件规定哪些用户能进入系统。

检查方法

- 运行以下命令并验证firewalld是否运行

```
[root@localhost ~]# firewall-cmd --state
running
```

*若firewalld已运行，则运行以下命令查看默认模式是否设置了规则，主要是查看ports、services、protocols、forward-ports、source-ports、icmp-blocks、rich rules等是否存在规则，如果为空则无规则。

```
[root@localhost ~]# firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: ens33
  sources:
  services: cockpit dhcpv6-client mdns ssh
  ports:
  protocols:
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

修改建议

加固方法

- 如果firewalld服务未运行，执行如下命令进行加固，如已运行则直接进行第二步：

```
[root@localhost ~]# systemctl enable firewalld
[root@localhost ~]# systemctl start firewalld
```

- 使用firewall-cmd命令添加需要的规则，例如

```
[root@localhost ~]# firewall-cmd --permanent --add-port=<port>/<protocol> --source=<source_ip>
```

其中，是端口号，是协议类型（如tcp、udp），是源IP地址。

- 运行以下命令使更改生效，这将重新加载firewalld配置并应用新规则

```
[root@localhost ~]# firewall-cmd --reload
```

还原方法

- 根据系统初始状态，使用如下命令恢复系统初始状态,如过无需恢复继续执行下一步：

```
[root@localhost ~]# systemctl disable firewalld
[root@localhost ~]# systemctl stop firewalld
```

- 使用firewall-cmd命令删除之前添加的规则，例如

```
[root@localhost ~]# firewall-cmd --permanent --remove-port=<port>/<protocol> --source=<source_ip>
```

- 运行以下命令使更改生效，这将重新加载firewalld配置并应用新规则

```
[root@localhost ~]# firewall-cmd --reload
```

修改影响

合理运用防火墙规则，能更好地控制网络通信流，并为服务器提供更高的安全性和可定制性。

14 系统维护

14.1 限制仅允许系统管理员进入维护模式

级别

要求

适用版本

V10 SP3 2403

说明

应提供维护模式中运行系统的能力，在维护模式下各种安全功能全部失效。系统仅允许系统管理员进入维护模式。

检查方法

- 在/boot目录中搜索grub.cfg文件的路径，根据路径判断系统为bios引导还是uefi引导

```
[root@localhost ~]# find /boot/ -name grub.cfg
/boot/grub2/grub.cfg
bios引导
```

```
[root@localhost ~]# find /boot/ -name grub.cfg
/boot/efi/EFI/kylin/grub.cfg
uefi引导
```

- 若系统为bios引导，运行以下命令并验证运行结果是否是否存在匹配"rescue"、"vmlinuz"、"init="、"rw"、且不存在"security="的行，若存在匹配行，则为无需加固，否则为待手动加固：

```
[root@localhost ~]# cat -n /boot/grub2/grub.cfg|sed -n '/rescue/,/\}/p'|grep vmlinuz
118             linux   /vmlinuz-0-rescue-b72289a945e34f2096160c4160b1ade1 root=/dev/mapper/klas-root rw
 resume=/dev/mapper/klas-swap rd.lvm.lv=klas/root rd.lvm.lv=klas/swap rhgb quiet crashkernel=1024M,
 high audit=0 security= lsm=none init=/bin/bash
```

- 若系统为uefi引导，运行以下命令并验证运行结果是否是否存在匹配"rescue"、"vmlinuz"、"init="、"rw"、且不存在"security="的行，若存在匹配行，则为无需加固，否则为待手动加固：

```
[root@localhost ~]# cat -n /boot/efi/EFI/kylin/grub.cfg|sed -n '/rescue/,/\}/p'|grep vmlinuz
116             linux   /vmlinuz-0-rescue-58d88e695ada44f5b090b5ba7aae1193 root=/dev/mapper/klas-root rw
 rd.lvm.lv=klas/root rd.lvm.lv=klas/swap video=VGA-1:640x480-32@60me rhgb quiet console=tty0 crashkernel=1024M,
 high smmu.bypassdev=0x1000:0x17 smmu.bypassdev=0x1000:0x15 video=efifb:off audit=0  security= lsm=none init=/bin/bash
```

修改建议

加固方法

- 在/boot/grub2/grub.cfg或者/boot/efi/EFI/kylin/grub.cfg文件里，注释匹配行，然后复制匹配行将ro替换为rw，删除security=后内容，添加init=/bin/bash： 如：

```
[root@localhost ~]# vim /boot/grub2/grub.cfg 或者 /boot/efi/EFI/kylin/grub.cfg
linux   /vmlinuz-0-rescue-ea01c54431004d448beef40bfe14cdff root=/dev/mapper/klas-root rw
 resume=/dev/mapper/klas-swap rd.lvm.lv=klas/root rd.lvm.lv=klas/swap rhgb quiet crashkernel=1024M,
 high audit=0 init=/bin/bash
```

还原方法

- 在/boot/grub2/grub.cfg或者/boot/efi/EFI/kylin/grub.cfg文件里，查找到匹配行后，去掉行前的"#"，再把加固时增加的行删除

修改影响

开启维护模式可以进行一些需要高权限的任务，例如修复磁盘错误，或者恢复忘记的密码。

15 资源分配

15.1 检查系统资源使用控制

级别

建议

适用版本

V10 SP3 2403

说明

应以每个用户或每个用户组为基础，提供一种机制，控制其对磁盘的消耗和对ＣＰＵ等资源的使用。

> **说明** 麒麟系统推荐安装quota、libcgroup软件。不同系统的软件以rpm -qa 查到的为准。以**quota-4.06-7.ky10.x86_64、libcgroup-0.42.2-1.ky10.x86_64**为例。

检查方法

- 运行以下命令，并验证是否安装了quota、libcgroup，如果未安装提示用户手动安装：

```
[root@localhost ~]# rpm -qa|grep quota
quota-4.06-7.ky10.x86_64
```

```
[root@localhost ~]# rpm -qa|grep libcgroup
libcgroup-0.42.2-1.ky10.x86_64
```

修改建议

加固方法

- 安装quota、libcgroup。

```
[root@localhost ~]# yum install quota
[root@localhost ~]# yum install libcgroup
```

还原方法

- 运行以下命令，卸载quota、libcgroup：

```
[root@localhost ~]# rpm -e quota
[root@localhost ~]# rpm -e libcgroup
```

修改影响

安装quota和libcgroup软件可以有效控制系统资源使用，提高系统安全性，但也需要注意配置复杂性、依赖性和兼容性问题。在安装之前，需要进行充分的评估和准备，以确保系统的正常运行和安全性。