第3章

FusionOS安全加固

CHAPTER 3

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

3.1 操作系统加固概述

须知

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

3.1.1 加固方案

本节描述 FusionOS 的安全加固方案，包括加固方式和加固内容。

1. 加固方式

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

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

2. 加固内容

FusionOS 系统加固内容主要分为以下 5 部分。

- 系统服务；
- 文件权限；
- 内核参数；
- 授权认证；
- 账号口令。

3.1.2 加固影响

对文件权限、账户口令等安全加固，可能造成用户使用习惯变更，从而影响系统的易用性。影响系统易用性的常见加固项参见表 3-1。

表 3-1 加固影响说明

加 固 项	建 议 加 固	易用性影响	FusionOS 默认是否设置了该加固项
字符界面等待超时限制	当字符界面长时间处在空闲状态时，字符界面应该自动退出。 说明： • 当用户通过 SSH 登录，会话空闲时长上限应该由/etc/profile 文件的 TMOUT 变量控制，而不能由/etc/ssh/sshd_config 文件的 ClientAliveInterval 变量和 ClientAliveCountMax 两个值决定。 • 建议 TMOUT 变量为 300(300s)	当字符界面长时间处在空闲状态时，字符界面会自动退出。 说明： 从 openSSH 8.2 版本起，ClientAliveInterval 和 ClientAliveCountMax 两个变量不再控制客户端的空闲时限，而是控制两端的通信故障时限(即两端通信出现故障一段时间后，Server 会自动关闭 SSH 会话)	是

续表

加　固　项	建 议 加 固	易用性影响	FusionOS 默认是否设置了该加固项
口令复杂度限制	口令长度最小为 8 位，口令至少包含大写字母、小写字母、数字和特殊字符中的三种	系统中所有用户不能设置简单的口令，口令必须符合复杂度要求	是
限定登录失败时的尝试次数	当用户登录系统时，口令连续输错三次，账户将被锁定 60s，锁定期间不能登录系统	用户不能随意登录系统，账户被锁定后必须等待 60s	是
用户默认 umask 值限制	设置所有用户的默认 umask 值为 077，使用户创建文件的默认权限为 600、目录权限为 700	用户需要按照需求修改指定文件或目录的权限	否
口令有效期	口令有效期的设置通过修改/etc/login.defs 文件实现，加固默认值为口令最大有效期 90 天，两次修改口令的最小间隔时间为 0，口令过期前开始提示天数为 7	口令过期后用户重新登录时，提示口令过期并强制要求修改，不修改则无法进入系统	是
su 权限限制	su 命令用于在不同账户之间切换。为了增强系统安全性，有必要对 su 命令的使用权进行控制，只允许 root 和 wheel 群组的账户使用 su 命令，限制其他账户使用	普通账户执行 su 命令失败，必须加入 wheel 群组才可以 su 成功	是
禁止 root 账户直接 SSH 登录系统	设置/etc/ssh/sshd_config 文件的 PermitRootLogin 字段的值为 no，用户无法使用 root 账户直接 SSH 登录系统	用户需要先使用普通账户 SSH 登录后，普通用户需要加入 wheel 组后，才能切换至 root 账户	是
SSH 强加密算法	SSH 服务的 MACs 和 Ciphers 配置，禁止对 CBC、MD5、SHA1 算法的支持，修改为 CTR、SHA2 算法	当前 Windows 下使用的部分低版本的 Xshell、PuTTY 不支持 aes128-ctr、aes192-ctr、aes256-ctr、hmac-sha2-256、hmac-sha2-512 算法，可能会出现无法通过 SSH 登录系统的情况，请使用最新的 PuTTY(0.63 版本以上)、Xshell(5.0 版本及以上版本)登录	是

3.2　加固指导

用户可以通过修改加固策略配置文件或加固脚本进行系统加固。本节介绍各加固项的含义以及 FusionOS 是否已默认加固，并给出加固方法，指导用户进行安全加固。

3.2.1　账户口令

1. 屏蔽系统账户

1) 说明

除了用户账户外，其他账号称为系统账户。系统账户仅系统内部使用，禁止用于登录系

统或其他操作，因此屏蔽系统账户。

2）实现

执行如下命令，将系统账户的 Shell 修改为/sbin/nologin。

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

📖说明

$systemaccount 指系统账户。

2. 限制使用 su 命令的账户

1）说明

su 命令用于在不同账户之间切换。为了增强系统安全性，有必要对 su 命令的使用权进行控制，只允许 root 和 wheel 群组的账户使用 su 命令，限制其他账户使用。

2）实现

su 命令的使用控制通过修改/etc/pam.d/su 文件实现，配置如下。

```
auth        required      pam_wheel.so use_uid
```

其中，pam_wheel.so 配置项说明如表 3-2 所示。

表 3-2 pam_wheel.so 配置项说明

配 置 项	说 明
use_uid	基于当前账户的 uid

3. 设置口令复杂度

1）说明

用户可以通过修改对应配置文件设置口令的复杂度要求，建议用户根据实际情况设置口令复杂度。

2）实现

口令复杂度通过/etc/pam.d/password-auth 和/etc/pam.d/system-auth 文件中的 pam_pwquality.so 和 pam_pwhistory.so 模块实现。用户可以通过修改这两个模块中的配置项修改口令复杂度要求。

3）设置举例

这里给出一个配置口令复杂度的例子，供用户参考。

(1) 密码复杂度要求。

① 口令长度至少 8 个字符。

② 口令必须包含如下至少三种字符的组合。

- 小写字母。
- 大写字母。
- 数字。
- 特殊字符：`～！@＃$％^&＊()-_＝＋\|[{}]；：'",<.>/？和空格。

③ 口令不能和账号或者账号的倒写一样。

④ 新口令不能和当前口令之前的 5 个旧口令相似。

(2) 配置实现。

在/etc/pam.d/password-auth 和/etc/pam.d/system-auth 文件中添加如下两行配置内容。

```
password    requisite     pam_pwquality.so minlen = 8 minclass = 3 enforce_for_root try_first_pass
local_users_only retry = 3 dcredit = 0 ucredit = 0 lcredit = 0 ocredit = 0
password    required      pam_pwhistory.so use_authtok remember = 5 enforce_for_root
```

(3) 配置项说明。

pam_pwquality.so 和 pam_pwhistory.so 的配置项见表 3-3 和表 3-4。

表 3-3　pam_pwquality.so 配置项说明

配　置　项	说　　明
minlen＝8	口令长度至少包含 8 个字符
minclass＝3	口令至少包含大写字母、小写字母、数字和特殊字符中的任意三种
ucredit＝0	口令包含任意个大写字母
lcredit＝0	口令包含任意个小写字母
dcredit＝0	口令包含任意个数字
ocredit＝0	口令包含任意个特殊字符
retry＝3	每次修改最多可以尝试三次
enforce_for_root	本设置对 root 账户同样有效

表 3-4　pam_pwhistory.so 配置项说明

配　置　项	说　　明
remember＝5	表示当前口令之前的 5 个旧口令会被系统记录，新口令不能和这 5 个旧口令相似
enforce_for_root	本设置对 root 账户同样有效

4. 设置口令有效期

1) 说明

出于系统安全性考虑，建议设置口令有效期限，且口令到期前通知用户更改口令。

2) 实现

口令有效期的设置通过修改/etc/login.defs 文件实现，加固项如表 3-5 所示。表中所有的加固项都在文件/etc/login.defs 中。表中字段直接通过修改配置文件完成。

表 3-5　login.defs 加固项说明

加　固　项	加固项说明	建 议 加 固	FusionOS 默认是否已加固为建议值
PASS_MAX_DAYS	口令最大有效期	90	是
PASS_MIN_DAYS	两次修改口令的最小间隔时间	0	是
PASS_WARN_AGE	口令过期前开始提示天数	7	是

📖说明

- login.defs是设置用户账号限制的文件，可配置口令的最大过期天数、最大长度约束等。该文件里的配置对root用户无效。
- 如果/etc/shadow文件里有相同的选项，则以/etc/shadow配置为准，即/etc/shadow的配置优先级高于/etc/login.defs。
- 口令过期后用户重新登录时，提示口令过期并强制要求修改，不修改则无法进入系统。

5. 设置口令的加密算法

1）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，FusionOS默认已设置。通过上述设置可以有效防范口令泄露，保证口令安全。

2）实现

口令的加密算法设置通过修改/etc/pam.d/password-auth和/etc/pam.d/system-auth文件实现，添加如下配置。

```
password    sufficient    pam_unix.so sha512 shadow nullok try_first_pass use_authtok
```

其中，pam_unix.so配置项说明如表3-6所示。

表3-6 pam_unix.so配置项说明

配置项	说明
sha512	使用sha512算法对口令加密

6. 登录失败超过三次后锁定

1）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300s）。

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

FusionOS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60s。

2）实现

口令复杂度的设置通过修改/etc/pam.d/password-auth和/etc/pam.d/system-auth文件实现，设置口令最大的出错次数为3次，系统锁定后的解锁时间为300s的配置以下三行所示。

```
auth    required     pam_faillock.so preauth audit deny=3 even_deny_root unlock_time=300
auth    [default=die] pam_faillock.so authfail audit deny=3 even_deny_root unlock_time=300
auth    sufficient   pam_faillock.so authsucc audit deny=3 even_deny_root unlock_time=300
```

其中，pam_faillock. so 配置项说明如表 3-7 所示。

表 3-7　pam_faillock. so 配置项说明

配　置　项	说　　明
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
unlock_time＝300	普通用户自动解锁时间为 300s(5min)
even_deny_root	同样限制 root 账户

7. 加固 su 命令

1）说明

为了增强系统安全性，防止使用“su”切换用户时将当前用户环境变量带入其他环境，FusionOS 默认已做配置。总是在使用 su 切换用户时初始化 PATH。

2）实现

通过修改/etc/login. defs 实现，配置如下。

```
ALWAYS_SET_PATH = yes
```

8. 密码到期时禁用账户

1）说明

为了增强系统安全性，在用户密码超过 30 天未更换时，禁用账户。FusionOS 默认配置为 35 天。

2）实现

通过修改/etc/default/useradd 实现，将默认 35 天修改为 30 天，配置如下。

```
INACTIVE = 30
```

9. 修改 TMOUT 配置

1）说明

为了增强 FusionOS 的安全性，需要在用户输入空闲一段时间后自动断开，这个操作可以由设置 TMOUT 值来实现。并将其设为 readonly 防止用户修改。如果需要修改此变量，需要重新配置，请按如下步骤实现。

2）实现

通过修改/etc/profile 文件实现。

步骤 1　编辑/etc/profile。

```
vi /etc/profile
```

步骤 2　修改 TMOUT 的值。

步骤 3　重启系统，使修改生效。

```
reboot
```

3.2.2　授权认证

1. 设置网络远程登录的警告信息

1）说明

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

2）实现

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

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

2. 禁止通过 Ctrl＋Alt＋Del 重启系统

1）说明

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

2）实现

禁止通过 Ctrl＋Alt＋Del 重启系统的操作步骤如下。

步骤 1　删除两个 ctrl-alt-del.target 文件，参考命令如下。

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

步骤 2　修改/etc/systemd/system.conf 文件，将 # CtrlAltDelBurstAction＝reboot-force 修改为 CtrlAltDelBurstAction＝none。

步骤 3　重启 systemd，使修改生效，参考命令如下。

```
systemctl daemon-reexec
```

3. 设置终端的自动退出时间

1）说明

无人看管的终端容易被侦听或被攻击，可能会危及系统安全。因此需要终端在停止运行一段时间后能够自动退出。

2）实现

自动退出时间由/etc/profile 文件的 TMOUT 字段(单位为秒)控制，在/etc/profile 的

尾部添加如下配置。

```
export TMOUT = 300
```

4. 设置 GRUB2 加密口令

1）说明

GRUB(GRand Unified Bootloader)是一个操作系统启动管理器，用来引导不同系统(如 Windows、Linux)，GRUB2 是 GRUB 的升级版。

系统启动时，可以通过 GRUB2 界面修改系统的启动参数。为了确保系统的启动参数不被任意修改，需要对 GRUB2 界面进行加密，仅在输入正确的 GRUB2 口令时才能修改启动参数。

说明

GRUB2 默认没有设置密码，建议用户首次登录时设置密码并定期更新，避免密码泄露后，启动选项被篡改，导致系统启动异常。

2）实现

步骤 1　使用 grub2-mkpasswd-pbkdf2 命令生成加密的口令。

说明

GRUB2 加密算法使用 PBKDF2。

```
# grub2 - mkpasswd - pbkdf2
Enter password:
Reenter password:  PBKDF2 hash of your password is grub.pbkdf2.sha512.10000.ACB8EE2321839E
444D8DD1E34B364F57ECC46BEBA26FC2B6004B4E1DC72B04E0655E9E0B14CDBB0A9F865DF91D66AD1168F66
738C54465F3746A2D92CCDEF249.4348B8768F83295572B06F0BA781C3295AC17EAFC45E7D86E2108ED11E7
F21235656A176B6D12C6D3F9FC5E21CEFC1C13C010B16045FE56F8B44E95774FBC6D4
```

说明

在 Enter password 和 Reenter password 输入相同的口令。grub.pbkdf2.sha512.10000.ACB8EE2321839E444D8DD1E34B364F57ECC46BEBA26FC2B6004B4E1DC72B04E0655E9E0B14CDBB0A9F865DF91D66AD1168F66738C54465F3746A2D92CCDEF249.4348B8768F83295572B06F0BA781C3295AC17EAFC45E7D86E2108ED11E7F21235656A176B6D12C6D3F9FC5E21CEFC1C13C010B16045FE56F8B44E95774FBC6D4 为 FusionOS12 # $ 经过 grub2-mkpasswd-pbkdf2 加密后的输出，每次输出的密文不同。

步骤 2　使用 vi 工具打开 grub.cfg 的开始位置追加如下两行字段。

```
set superusers = "root" password_pbkdf2 root grub.pbkdf2.sha512.10000.ACB8EE2321839E444D8D
D1E34B364F57ECC46BEBA26FC2B6004B4E1DC72B04E0655E9E0B14CDBB0A9F865DF91D66AD1168F66738C54
465F3746A2D92CCDEF249.4348B8768F83295572B06F0BA781C3295AC17EAFC45E7D86E2108ED11E7F21235
656A176B6D12C6D3F9FC5E21CEFC1C13C010B16045FE56F8B44E95774FBC6D4
```

说明

- 不同模式下 grub.cfg 文件所在路径不同：x86 架构的 UEFI 模式下位于/boot/efi/EFI/FusionOS/grub.cfg，legacy BIOS 模式下位于/boot/grub2/grub.cfg。aarch64 架构下，

只有一种安装模式 UEFI，因此只有一种目录：/boot/efi/EFI/FusionOS/grub.cfg

- superusers 字段用于设置 GRUB2 的超级管理员的账户名。
- password_pbkdf2 字段后的参数，第 1 个参数为 GRUB2 的账户名，第 2 个为该账户的加密口令，两个参数配置在同一行。

5. 安全单用户模式

1）说明

单用户模式是以 root 权限进入系统的。

2）实现

重启系统，在出现 grub 界面时选择要启动的内核，按 E 键进入编辑模式，在以“linux”开始的那一行末尾添加空格和“single”，按 Ctrl＋X 组合键启动，如图 3-1 所示。

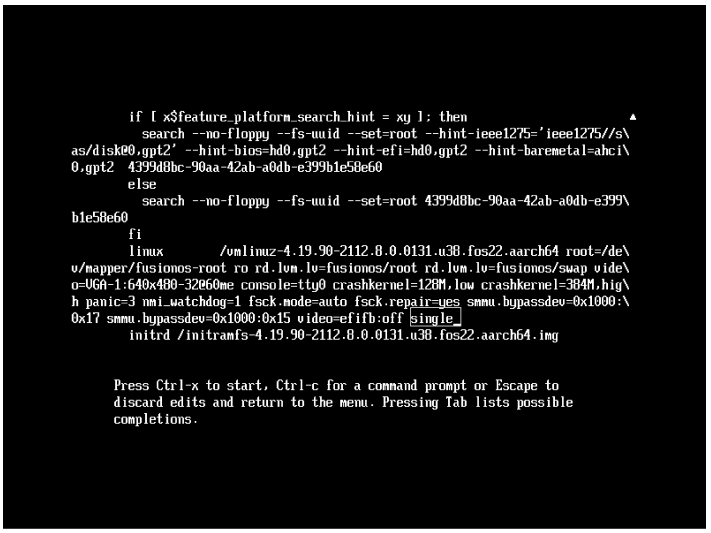

图 3-1　编辑模式

6. 禁止交互式启动

1）说明

使用交互式引导，控制台用户可以禁用审计、防火墙或其他服务，削弱了系统安全性。用户可以禁止使用交互式引导，提升安全性。FusionOS 默认已禁止。

2）实现

该设置可以通过修改/etc/sysconfig/init 文件内容实现。将 PROMPT 选项配置为 PROMPT＝no。

3.2.3　系统服务

1. 加固 SSH 服务

1）说明

SSH(Secure Shell)是目前较可靠，专为远程登录会话和其他网络服务提供安全性保障

的协议。利用 SSH 协议可以有效防止远程管理过程中的信息泄露问题。通过 SSH 可以对所有传输的数据进行加密，并防止 DNS 欺骗和 IP 欺骗。OpenSSH 是 SSH 协议的免费开源实现。

加固 SSH 服务，是指修改 SSH 服务中的配置来设置系统使用 OpenSSH 协议时的算法、认证等参数，从而提高系统的安全性。表 3-8 中详细说明了各加固项含义、建议加固值及其默认策略。

2）实现

服务端加固操作如下。

步骤 1　打开服务端 SSH 服务的配置文件/etc/ssh/sshd_config，在该文件中修改或添加对应加固项及其加固值。

步骤 2　保存/etc/ssh/sshd_config 文件。

步骤 3　重启 SSH 服务，命令如下。

```
systemctl restart sshd
```

客户端加固操作如下。

步骤 1　打开客户端 SSH 服务的配置文件/etc/ssh/ssh_config，在该文件中修改或添加对应加固项及其加固值。

步骤 2　保存/etc/ssh/ssh_config 文件。

步骤 3　重启 SSH 服务，命令如下。

```
systemctl restart sshd
```

3）加固项说明

（1）服务端加固策略。

SSH 服务的所有加固项均保存在配置文件/etc/ssh/sshd_config 中，服务端各加固项的含义、加固建议以及 FusionOS 默认是否已经加固为建议加固值见表 3-8。

表 3-8　SSH 服务端加固项说明

加　固　项	加固项说明	加 固 建 议	FusionOS 默认是否已加固为建议值
Protocol	设置使用 SSH 协议的版本	2	是
SyslogFacility	设置 SSH 服务的日志类型。加固策略将其设置为“AUTH”，即认证类日志	AUTH	是
LogLevel	设置记录 sshd 日志消息的层次	VERBOSE	是
X11Forwarding	设置使用 SSH 登录后，能否使用图形化界面	no	是
MaxAuthTries	最大认证尝试次数	3	否
PubkeyAuthentication	设置是否允许公钥认证	yes	是
RSAAuthentication	设置是否允许只有 RSA 安全验证	yes	是

续表

加 固 项	加固项说明	加 固 建 议	FusionOS 默认是否已加固为建议值
IgnoreRhosts	设置是否使用 rhosts 文件和 shosts 文件进行验证。rhosts 文件和 shosts 文件用于记录可以访问远程计算机的计算机名及关联的登录名	yes	是
RhostsRSAAuthentication	设置是否使用基于 rhosts 的 RSA 算法安全验证。rhosts 文件记录可以访问远程计算机的计算机名及关联的登录名	no	是
HostbasedAuthentication	设置是否使用基于主机的验证。基于主机的验证是指已信任客户机上的任何用户都可以使用 SSH 连接	no	是
PermitRootLogin	是否允许 root 账户直接使用 SSH 登录系统。 说明： 若需要直接使用 root 账户通过 SSH 登录系统，请修改/etc/ssh/sshd_config 文件的 PermitRootLogin 字段的值为 yes	no	是
PermitEmptyPasswords	设置是否允许用口令为空的账号登录	no	是
PermitUserEnvironment	设置是否解析～/. ssh/environment 和～/. ssh/authorized_keys 中设定的环境变量	no	是
Ciphers	设置 SSH 数据传输的加密算法	aes128-ctr，aes192-ctr，aes256-ctr，aes128-gcm @ openssh. com，aes256-gcm @ openssh. com，chacha20-poly1305@openssh. com	是
ClientAliveCountMax	设置超时次数。服务器发出请求后，客户端没有响应的次数达到一定值，连接自动断开	0	是
Banner	指定登录 SSH 前后显示的提示信息的文件	/etc/issue. net	是
MACs	设置 SSH 数据校验的哈希算法	hmac-sha2-512，hmac-sha2-512-etm @ openssh. com，hmac-sha2-256，hmac-sha2-256-etm@openssh. com	是

续表

加　固　项	加固项说明	加 固 建 议	FusionOS 默认是否已加固为建议值
StrictModes	设置 SSH 在接收登录请求之前是否检查用户 HOME 目录和 rhosts 文件的权限和所有权	yes	是
UsePAM	使用 PAM 登录认证	yes	是
AllowTcpForwarding	设置是否允许 TCP 转发	no	是
Subsystem sftp/usr/libexec/openssh/sftp-server	sftp 日志记录级别，记录 INFO 级别以及认证日志	-l INFO -f AUTH	是
AllowAgentForwarding	设置是否允许 SSH Agent 转发	no	是
GatewayPorts	设置是否允许连接到转发客户端端口	no	是
PermitTunnel	Tunnel 设备是否允许使用	no	是
KexAlgorithms	设置 SSH 密钥交换算法	-diffie-hellman-group1-sha1，diffie-hellman-group14-sha1，diffie-hellman-group14-sha256，diffie-hellman-group16-sha512，diffie-hellman-group18-sha512，diffie-hellman-group-exchange-sha1，diffie-hellman-group-exchange-sha256	是
LoginGraceTime	限制用户必须在指定的时限内认证成功，0 表示无限制。默认值是 60s	60	否

说明

默认情况下，登录 SSH 前后显示的提示信息保存在/etc/issue.net 文件中，/etc/issue.net 默认信息为“Authorized users only. All activities may be monitored and reported.”。

（2）客户端加固策略。

SSH 服务的所有加固项均保存在配置文件/etc/ssh/ssh_config 中，客户端各加固项的含义、加固建议以及 FusionOS 默认是否已经加固为建议加固值见表 3-9。

表 3-9　SSH 客户端加固项说明

加　固　项	加固项说明	加 固 建 议	FusionOS 默认是否已加固为建议值
KexAlgorithms	设置 SSH 密钥交换算法	ecdh-sha2-nistp256，ecdh-sha2-nistp384，ecdh-sha2-nistp521，diffie-hellman-group-exchange-sha256	否
VerifyHostKeyDNS	是否使用 DNS 或者 SSHFP 资源记录验证 HostKey	ask	否

说明

对于使用 dh 算法进行密钥交换的第三方客户端和服务端工具，要求允许建立连接的最低长度为 2048b。

2. 设置时间同步 chrony

1) 说明

chrony 是一个实现网络时间协议(NTP)的守护进程,使用高度准确的源跨各种系统同步系统时钟,FusionOS 默认启用 chrony 替代 ntp。

2) 实现

步骤 1 执行如下命令,检查 chrony 的远程服务器配置。

```
grep -E "^(server|pool)" /etc/chrony.conf
```

返回示例:

```
pool pool.ntp.org iburst
```

步骤 2 执行如下命令,检查 chronyd 进程。

```
ps -ef | grep chronyd
```

返回示例:

```
chrony 345809   1   0 16:33 ?     00:00:00 /usr/sbin/chronyd
```

步骤 3 根据需要将服务器或池行添加或编辑到/etc/chrony.conf。

```
vi /etc/chrony.conf
```

步骤 4 修改配置文件后需要重启生效。

```
systemctl restart chronyd
```

3. 其他安全建议

(1) SSH 服务仅侦听指定 IP 地址。

出于安全考虑,建议用户在使用 SSH 服务时,仅在必需的 IP 上进行绑定侦听,而不是侦听 0.0.0.0,可修改/etc/ssh/sshd_config 文件中的 ListenAddress 配置项。

① 打开并修改/etc/ssh/sshd_config 文件。

```
vi /etc/ssh/sshd_config
```

修改内容如下,表示绑定侦听 IP 为 192.168.1.100,用户可根据实际情况修改需要侦听的 IP。

```
...
ListenAddress 192.168.1.100
...
```

② 重启 SSH 服务。

```
systemctl restart sshd.service
```

(2) 限制 SFTP 用户向上跨目录访问。

SFTP 是 FTP over SSH 的安全 FTP 协议,对于访问 SFTP 的用户建议使用专用账号,只能上传或下载文件,不能用于 SSH 登录,同时对 SFTP 可以访问的目录进行限定,防止目录遍历攻击,具体配置如下。

说明

sftpgroup 为示例用户组,sftpuser 为示例用户名,sftpupload 为示例用户上传目录。

- 创建 SFTP 用户组。

```
groupadd sftpgroup
```

- 创建 SFTP 根目录。

```
mkdir /sftp
```

- 修改 SFTP 根目录属主和权限。

```
chown root:root /sftp
chmod 755 /sftp
```

- 创建 SFTP 用户。

```
useradd -g sftpgroup -s /sbin/nologin sftpuser
```

- 设置 SFTP 用户的口令。

```
passwd sftpuser
```

- 创建 SFTP 用户登录后的根目录。

```
mkdir /sftp/sftpuser
```

- 修改 SFTP 用户登录后根目录的属主和权限。

```
chown root:root /sftp/sftpuser
chmod 755 /sftp/sftpuser
```

- 创建 SFTP 用户上传目录。

```
mkdir /sftp/sftpuser/sftpupload
```

- 修改 SFTP 用户上传目录的属主。

```
chown sftpuser:sftpgroup /sftp/sftpuser/sftpupload
```

• 修改/etc/ssh/sshd_config 文件。

```
vi /etc/ssh/sshd_config
```

修改内容如下。

```
#Subsystem sftp /usr/libexec/openssh/sftp-server -l INFO -f AUTH
Subsystem sftp internal-sftp -l INFO -f AUTH
...

Match Group sftpgroup
    ChrootDirectory /sftp/%u
    ForceCommand internal-sftp
```

说明

• %u 代表当前 sftp 用户的用户名,这是一个通配符,用户原样输入即可。

• 以下内容必须加在/etc/ssh/sshd_config 文件的末尾。

```
Match Group sftpgroup
  ChrootDirectory /sftp/%u
  ForceCommand internal-sftp
```

表示当属于 sftpgroup 组的用户通过 sftp 连接到服务器时,他们将被限制在自己的 chroot 目录 /sftp/用户名中,而且他们只能执行 sftp 操作,无法执行其他 Shell 命令。这样可以确保用户只能在指定的目录内进行安全的文件传输操作,同时不会访问系统中的其他文件和目录。

• 重启 SSH 服务。

```
systemctl restart sshd.service
```

(3) SSH 远程执行命令。

OpenSSH 通用机制,在远程执行命令时,默认不开启 tty,如果执行需要密码的命令,密码会明文回显。出于安全考虑,建议用户增加-t 选项,确保密码输入安全,如下:

```
ssh -t testuser@192.168.1.100 su
```

说明

192.168.1.100 为示例 IP,testuser 为示例用户。

3.2.4 文件权限

1. 设置文件的权限和属主

1) 说明

Linux 将所有对象都当作文件来处理,即使一个目录也被看作包含有多个其他文件的大文件。因此,Linux 中最重要的就是文件和目录的安全性。文件和目录的安全性主要通过权限和属主来保证。

FusionOS 默认对系统中的常用目录、可执行文件和配置文件设置了权限和属主。

2）实现

以/bin 目录为例，修改文件权限和文件属主的操作如下。

（1）修改文件权限。例如，将/bin 目录权限设置为 755。

```
chmod 755 /bin
```

（2）修改文件属主。例如，将/bin 目录的拥有者和群组设置为 root:root。

```
chown root:root /bin
```

2. 删除无主文件

1）说明

系统管理员在删除用户/群组时，可能会出现忘记删除该用户/该群组所拥有文件的问题。如果后续新创建的用户/群组与被删除的用户/群组同名，则新用户/新群组会拥有部分不属于其权限的文件，建议将此类文件删除。

2）实现

删除用户 ID 不存在的文件。

步骤 1　查找用户 ID 不存在的文件。

```
find / -nouser
```

步骤 2　删除查找到的文件。其中，filename 为用户 ID 不存在文件的文件名。

```
rm -f filename
```

步骤 3　查找群组 ID 不存在的文件。

```
find / -nogroup
```

步骤 4　删除查找到的文件。其中，filename 为群组 ID 不存在文件的文件名。

```
rm -f filename
```

3. 处理空链接文件

1）说明

无指向的空链接文件，可能会被恶意用户利用，影响系统安全性。建议用户删除无效的空链接文件，提高系统安全性。

2）特殊场景

FusionOS 系统安装完成后，可能存在空链接文件，这些空链接文件可能有对应用途（有些空链接文件是预制的，会被其他组件依赖）。请用户根据实际环境进行处理，处理方式请参见实现。

例如，FusionOS 在 x86 架构下支持 UEFI 和 legacy BIOS 两种安装模式，两种引导场

景支持的 grub 相关包默认都安装。当用户选择 legacy BIOS 模式安装时，形成空链接文件“/etc/grub2.cfg”；当用户选择 UEFI 模式安装时，会形成空链接文件“/etc/grub2-uefi.cfg”，需要用户根据实际情况处理空链接。

3）实现

步骤 1 通过如下命令查找系统中的空链接文件。

```
find dirname -type l -follow 2>/dev/null
```

说明

dirname 为搜索目录的名称，通常需要关注系统关键目录：/bin、/boot、/usr、/lib64、/lib、/var 等。

步骤 2 如果此类文件无实际作用，可通过如下命令删除。

```
rm -f filename
```

说明

filename 为步骤 1 找出的文件名。

4. 设置守护进程的 umask 值

1）说明

umask 值用来为新创建的文件和目录设置默认权限。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，让系统可以接收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，建议设置其 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参见 umask 含义。

说明

FusionOS 默认设置守护进程的 umask 值为 0022。

2）实现

在配置文件/etc/sysconfig/init 中新增一行：umask 0027。

5. 为全局可写目录添加粘滞位属性

1）说明

任意用户可以删除、修改全局可写目录中的文件和目录。为了确保全局可写目录中的文件和目录不会被任意删除，需要为全局可写目录添加粘滞位属性。

2）实现

步骤 1 搜索全局可写目录。

```
find / -type d -perm -0002 ! -perm -1000 -ls | grep -v proc
```

步骤 2 为全局可写目录添加粘滞位属性。dirname 为实际查找到的目录名。

```
chmod +t dirname
```

6. 删除非授权文件的全局可写属性

1）说明

全局可写文件可被系统中的任意用户修改，影响系统完整性。

2）实现

步骤 1　列举系统中所有的全局可写文件。

```
find / -type d \( -perm -o+w \) | grep -v proc
find / -type f \( -perm -o+w \) | grep -v proc
```

步骤 2　查看步骤 1 列举的所有文件（粘滞位的文件和目录可以排除在外），删除文件或去掉其全局可写权限。使用以下命令去掉权限，其中，filename 为对应文件名。

```
chmod o-w  filename
```

说明

可通过如下命令确定对应文件或目录是否设置了粘滞位，若回显中包含 T 标记，则为粘滞位文件或目录。命令中的 filename 为需要查询文件或目录的名称。

```
ls -l filename
```

7. 限制 at 命令的使用权限

1）说明

at 命令用于创建在指定时间自动执行的任务。为避免任意用户通过 at 命令安排工作，造成系统易受攻击，需要指定可使用该命令的用户。

2）实现

步骤 1　删除/etc/at.deny 文件。

```
rm -f /etc/at.deny
```

步骤 2　创建/etc/at.allow 文件。

```
touch /etc/at.allow
```

步骤 3　将/etc/at.allow 的文件属主改为 root:root。

```
chown root:root /etc/at.allow
```

步骤 4　控制/etc/at.allow 的文件权限，仅 root 可操作。

```
chmod og-rwx /etc/at.allow
```

8. 限制 cron 命令的使用权限

1）说明

cron 命令用于创建例行性任务。为避免任意用户通过 cron 命令安排工作，造成系统易

受攻击，需要指定可使用该命令的用户。

2）实现

步骤 1　删除/etc/cron.deny 文件。

```
rm -f /etc/cron.deny
```

步骤 2　创建/etc/cron.allow 文件。

```
touch /etc/cron.allow
```

步骤 3　将/etc/cron.allow 的文件属主改为 root:root。

```
chown root:root /etc/cron.allow
```

步骤 4　控制/etc/cron.allow 的文件权限，仅 root 可操作。

```
chmod og-rwx /etc/cron.allow
```

9. 限制 sudo 命令的使用权限

1）说明

sudo 命令用于普通用户以 root 权限执行命令。为了增强系统安全性，有必要对 sudo 命令的使用权进行控制，只允许 root 使用 sudo 命令，限制其他账户使用。FusionOS 默认未限制 wheel 组内的非 root 用户使用 sudo 命令的权限。

2）实现

sudo 命令的使用控制通过修改/etc/sudoers 文件实现，需要注释掉如下配置行。

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

10. Capability 权能机制

1）简介

Capability 权能机制的主要思想在于分割 root 用户的特权，即将 root 的特权分割成不同的能力。Capabilities 作为线程（Linux 并不真正区分进程和线程）的属性存在，每个功能组都可以独立启用和禁用。其本质上就是将内核调用分门别类，具有相似功能的内核调用被分到同一组中。这样一来，权限检查的过程就变成了：在执行特权操作时，如果线程的有效身份不是 root，就去检查其是否具有该特权操作所对应的 Capabilities，并以此为依据，决定是否可以执行特权操作。Capabilities 可以在进程执行时赋予，也可以直接从父进程继承。

基于 POSIX1.e 中关于能力的定义，系统设计实现如下 38 种能力，如表 3-10 所示。

表 3-10　Capability 权能列表

能力名称	值	含　义
CAP_CHOWN	0	进程进行 chown()操作修改文件的属主 ID 和用户组 ID，覆盖进程属主 ID 等于文件属主 ID、进程用户组 ID 或进程附加组 ID 等于文件用户组 ID 的限制

续表

能 力 名 称	值	含　义
CAP_DAC_OVERRIDE	1	进程执行文件时，覆盖文件访问权限位中对执行访问的限制
CAP_DAC_READ_SEARCH	2	进程读文件或搜索目录时，覆盖文件访问权限位中对读和搜索的限制
CAP_FOWNER	3	在 CAP_FSETID 未设置时，可以覆盖要求进程属主 ID 等于文件属主 ID 的文件操作限制
CAP_FSETID	4	当文件的 S_ISUID 和 S_ISGID 位被置上时，可以覆盖要求进程属主 ID 等于文件属主 ID 的文件操作限制；当文件的 S_ISGID 被置上时，可以覆盖要求进程所属组 ID 等于文件属主 ID 的文件操作限制
CAP_KILL	5	覆盖信号发送时要求发送进程的用户 ID/有效用户 ID 等于接收进程的用户 ID/有效用户 ID 的限制
CAP_SETGID	6	覆盖进程进行 setgid()操作修改进程的真实用户组 ID 和只能修改有效用户组 ID 到真实用户组 ID 的限制；若系统支持保留 ID 时，覆盖只能修改保留的设置组 ID 为真实用户组 ID 或保留的设置组 ID 的限制
CAP_SETUID	7	覆盖进程进行 setgid()操作修改进程的真实属主 ID 和只能修改有效属主 ID 到真实属主 ID 的限制；若系统支持保留 ID 时，覆盖修改保留的设置属主 ID 的限制
CAP_SETPCAP	8	允许设置当前用户 permitted 集合中的任何权能到任何进程；允许删除任何进程中属于当前用户 permitted 集合的任何权能
CAP_LINUX_IMMUTABLE	9	允许修改文件的 S_IMMUTABLE 和 S_APPEND 属性
CAP_NET_BIND_SERVICE	10	允许绑定到 1024 以下的 TCP/UDP sockets；允许绑定到 32 以下的 ATM VCI
CAP_NET_BROADCAST	11	允许广播消息和监听组播消息
CAP_NET_ADMIN	12	允许接口配置；允许 IP 防火墙，伪装和记账功能；允许设置 socket 调试选项；允许修改路由表；允许设置任意进程或进程组对 socket 的所有权；允许绑定到任何开放代理；允许设置服务类型；允许设置混杂模式；允许清除驱动的统计信息；允许组播；允许读/写设备专用寄存器；允许激活 ATM 控制 socket
CAP_NET_RAW	13	允许使用 RAW socket；允许使用 PACKET socket
CAP_IPC_LOCK	14	允许对共享内存段加锁；允许 mlock 和 mlockall
CAP_IPC_OWNER	15	允许覆盖 IPC 所有权检测
CAP_SYS_MODULE	16	允许插入和删除内核模块，修改内核
CAP_SYS_RAWIO	17	允许 ioperm/iopl 访问；允许通过/proc/bus/usb 发送 USB 消息到任何设备
CAP_SYS_CHROOT	18	允许使用 chroot
CAP_SYS_PTRACE	19	允许 ptrace 任何进程
CAP_SYS_PACCT	20	允许设置进程记账

续表

能力名称	值	含义
CAP_SYS_ADMIN	21	允许配置安全密码；允许管理随机设备；允许检查和配置磁盘配额；允许配置内核 syslog(例如 printk 操作)；允许设置域名；允许设置主机名；允许调用 bdflush()；允许 mount()和 umount()，设置新的 smb 连接；允许一些 nfsservctl；允许 VM86_REQUEST_IRQ；允许读/写 alpha 体系结构的 pci 配置；允许 mips 上的 irix_prctl；允许 m68k 上的 cache flush 操作；允许删除信号；允许加锁/解锁共享内存段 ；允许打开/关闭 swap；允许在块设备上设置 readahead 和缓冲 flush；允许在 socket 信任检测时伪造 pid；允许在软驱中设置几何；允许在 xd 驱动上开/关 DMA；允许管理 md 设备；允许调整 ide 驱动；允许访问 nvram 设备；允许管理 apm_bios、序列和 bttv 设备；允许 isdn CAPI 驱动的 manufacturer 命令；允许在 pci 配置空间读取非标准部分；允许在 sbpcd 驱动中 DDI 调试 ioctl；允许建立序列端口；允许发送 qic-117 命令；允许在 SCSI 控制器上打开/关闭标记队列，发送任意 SCSI 命令；允许在 loopback 文件系统中设置加密钥匙；允许设置 zone reclaim 策略
CAP_SYS_BOOT	22	允许使用 reboot
CAP_SYS_NICE	23	允许提升和设置其他进程的优先级；允许对自己的进程使用 FIFO 和 round-robin 调度；允许设置其他进程的调度算法；允许在其他进程上设置 cpu affinity
CAP_SYS_RESOURCE	24	允许覆盖资源限制，设置资源闲置；允许覆盖配额限制；允许为 ext2 文件系统保留空间；允许在 ext3 文件系统修改数据 journaling 模式；允许覆盖 IPC 消息队列的长度限制；允许使用实际时钟中大于 64Hz 的中断；允许覆盖控制台分配的最大数量；允许覆盖 keymap 的最大数量
CAP_SYS_TIME	25	允许操作系统时钟；允许在 mips 上 irix_stime；允许设置实际时钟
CAP_SYS_TTY_CONFIG	26	允许配置 tty 设备，允许 tty 上的 vhangup()
CAP_MKNOD	27	允许 mknod()的特权
CAP_LEASE	28	允许对文件采用 lease 操作
CAP_AUDIT_WRITE	29	允许对审计文件进行写操作
CAP_AUDIT_CONTROL	30	允许配置和管理审计
CAP_SETFCAP	31	允许为文件设置任意的 Capabilities
CAP_MAC_OVERRIDE	32	忽略文件的 DAC 访问限制
CAP_MAC_ADMIN	33	允许 MAC 配置或状态更改
CAP_SYSLOG	34	允许使用 syslog()系统调用
CAP_WAKE_ALARM	35	允许触发一些能唤醒系统的东西(如 CLOCK_BOOTTIME_ALARM 计时器)
CAP_BLOCK_SUSPEND	36	使用可以阻止系统挂起的特性
CAP_AUDIT_READ	37	允许通过 multicast netlink 套接字读取审计日

2) 进程及文件的权能

(1) 进程 Capabilities。

每个进程均关联 4 个能力位集，分别如下。

① Inheritable(继承能力位集，简写为 pI)，表示进程可以传递给子进程的能力集。

② Permitted(许可能力位集，简写为 pP)，表示进程所能够拥有的能力集上限。

③ Effective(有效能力位集,简写为 pE),表示当前进程可用的能力集。每个能力在位集中占一位,如果为 1,表示对应能力被设置;如果为 0,则表示对应能力被关。

④ Bounding(是 Inheritable 集合的超集),如果某个 Capability 不在 Bounding 集合中,即使它在 Permitted 集合中,该进程也不能将该 Capability 添加到它的 Inheritable 集合中。此外,Bounding 集合中的 Capabilities 在执行 fork()系统调用时会传递给子进程的 Bounding 集合,并且在执行 exec 系统调用后保持不变。

在进行访问控制时,用于访问判断的是 E 能力位集。当一个进程试图进行某特权操作时,操作系统将检查进程 E 能力位集中的相应位,只有 E 能力位集中的相应位被设置时,进程才被允许执行该特权操作,而不管另两个能力位集的相应位如何。

(2) 文件 Capabilities。

任何文件都可以赋予能力,但只有可执行文件的能力才有意义。可执行文件拥有三组能力集,分别称为 Effective、Inheritable 和 Permitted(分别简记为 fE,fI,fP)。Permitted 集合中包含的 Capabilities,在文件被执行时,自动加入进程的 Permitted 集合;Inheritable 集合与进程的 Inheritable 集合的交集,是执行完 exec 后实际集成的 Capabilities;Effective 仅是一个 bit,如果设置开启,那么在运行 exec 后,Permitted 集合中新增的 Capabilities 会自动出现在 Effective 集合中,否则不会出现在 Effective 集合中。

在系统中,主体是进程,进程只有具有了能力,才能够代表用户进行特权操作,新进程能力的计算方式如下,P 代表执行 exec 前的 Capabilities,P'代表执行 exec 后的 Capabilities,F 代表 exec 执行的文件的 Capabilities。

- P'(Permitted)=(P(Inheritable) & F(Inheritable)) | (F(Permitted) & P(Bounding))
- P'(Effective)=F(Effective)? P'(Permitted): 0
- P'(Inheritable)=P(Inheritable)

解释如下。

① 执行 exec 前进程的 Inheritable 集合与可执行文件的 Inheritable 集合取交集,会被添加到执行 exec 后进程的 Permitted 集合;可执行文件的 Permitted 集合与执行 exec 前进程的 Bounding 集合取交集,也会被添加到执行 exec 后进程的 Permitted 集合。

② 如果可执行文件开启了 Effective 标志位,那么在执行完 exec 后,进程 Permitted 集合中的 Capabilities 会自动添加到它的 Effective 集合中。

③ 执行 exec 前进程的 Inheritable 集合会继承给执行 exec 后进程的 Inheritable 集合。

说明

- 上面的公式是针对系统调用 exec 的,如果是 fork,那么子进程的 Capabilities 信息完全复制父进程的能力位集。
- 进程各能力集的直接操作。

系统提供了设置进程能力集的系统调用 capset()。进程可以调用 capset()来直接修改除 init 进程以外的任何进程的能力集,但只有拥有 CA_SETPCAP 能力的进程才有此特权操作,而且修改后的新能力集必须为该进程 pP 的子集。

3) 使用说明

(1) 接口。

FusionOS 提供 libcap 软件包用于获取和设置文件及进程的 cap,此外还提供编程接口

用于 cap 相关编程。下面介绍 libcap 软件包提供的若干关于 cap 的接口。

① 使用 setcap 设置文件 cap。

```
setcap (capabilities | -r) filename
```

- capabilities：所要设置的权能。
- -r：移除权能。
- filename：所要操作的可执行文件。

② 使用 getcap 查看文件 cap。

```
getcap filename
```

filename：需查询的文件名。

③ 使用 getpcaps 查看进程 cap。

```
getpcaps pid
```

pid：需查询的进程 id。

(2) 可执行文件 cap 的设置及执行过程。

以对可执行文件/bin/chown 设置 CAP_CHOWN 能力为例，CAP_CHOWN 允许改变文件的 owner。

步骤 1 执行如下命令设置可执行文件/bin/chown 的 cap。

```
setcap cap_chown = eip /bin/chown
```

说明

cap_chown＝eip 是将 chown 的能力以 e、i、p 三种位集的方式授权给相关的程序文件。

步骤 2 执行如下命令查看可执行文件的 cap。

```
getcap /bin/chown
```

回显如下。

```
/bin/chown = cap_chown + eip
```

步骤 3 切换到普通用户后，执行如下命令将测试文件的 owner 改为普通用户。

```
chown testuser.testuser ./test.file
```

回显如下。

```
-rw-------. 1 testuser testuser 3 Jul 23 21:39 test.file
```

(3) 系统预置的可执行文件 capability。

FusionOS 预置的/usr/bin 和/usr/sbin 下的二进制文件具备 capability 能力的罗列如下。

```
arping = cap_net_raw + p
clockdiff = cap_net_raw + p
newrole = cap_chown,cap_dac_override,cap_dac_read_search,cap_fowner,cap_setpcap,cap_sys_
admin,cap_audit_write + ep
ping = cap_net_admin,cap_net_raw + p
suexec = cap_setgid,cap_setuid + ep
```

11. 检查 SUID 和 SGID 可执行文件

1）说明

SUID(Set User ID)和 SGID(Set Group ID)是两种特殊权限，设置 SUID 权限的文件在执行时，调用者将获得该文件所有者的权限；设置 SGID 权限的文件在执行时，调用者将获得该文件所属组的权限。

文件的所有者可以将文件的权限设置为 SUID 或 SGID 的权限，使用户能够执行需要 root 权限或 root 用户组权限的功能(如修改密码)。

SUID 和 SGID 程序的存在有正当原因，但是要识别和审核此类程序以确保它们是合法的。

2）实现

步骤 1　执行如下命令，列出所有的 SUID 文件。

```
df --local -P | awk '{if (NR!=1) print $6}' | xargs -I '{}' find '{}' -xdev -type f -perm -
4000 | xargs ls -l
```

执行如下命令，列出所有的 SGID 文件。

```
df --local -P | awk '{if (NR!=1) print $6}' | xargs -I '{}' find '{}' -xdev -type f -perm -
2000 | xargs ls -l
```

上面的命令只搜索本地文件系统，可以省略--local 搜索系统上的所有文件系统，包含网络挂载的分区，或者为每个分区手工执行如下命令。

列出指定分区的 SUID 文件。

```
find <partition> -xdev -type f -perm -4000
```

列出指定分区的 SGID 文件。

```
find <partition> -xdev -type f -perm -2000
```

步骤 2　检查返回的文件，确保系统中没有引入任何恶意 SUID 和 SGID 程序。并检查系统二进制文件的 MD5 校验和是否与包中的一致，确认二进制文件没有被替换。

12. 删除隐藏的可执行文件

1）说明

恶意程序、代码和脚本通常以点“.”开头以隐藏自身。

2）实现

步骤 1 执行如下命令，查找出隐藏的可执行文件。

```
find / -type f -executable -name ".*"
```

步骤 2 检查结果中的文件，根据需要进行删除处理。

```
rm -f filename
```

说明

filename 为步骤 1 找出的文件名。

3.2.5 内核参数

1. 加固内核参数

1）说明

内核参数决定配置和应用特权的状态。内核提供用户可配置的系统控制，这一系统控制可微调或配置，该功能特性可通过控制各种可配置的内核参数，来提高操作系统的安全特性。例如，通过微调或配置网络选项，可有效提高系统的安全性。

2）实现

步骤 1 将表 3-11 中的加固项写入/etc/sysctl.conf 文件中。

说明

写入方式如下。

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

表 3-11 内核参数加固策略说明

加 固 项	加固项说明	加固建议	FusionOS 默认是否已加固为建议值
net.ipv4.icmp_echo_ignore_broadcasts	是否接收 ICMP 广播报文。加固策略为不接收	1	是
net.ipv4.conf.all.rp_filter	验证数据包使用的实际源地址是否与路由表相关，以及使用该特定源 IP 地址的数据包是否通过接口获取其响应。加固策略为启用该项	1	是
net.ipv4.conf.default.rp_filter		1	是
net.ipv4.ip_forward	IP Forwarding 可阻止未授权的 IP 数据包渗透至网络。加固策略为禁用该特性	0	是
net.ipv4.conf.all.accept_source_route	accept_source_route 指允许数据包的发送者指定数据包的发送路径，以及返回给发送者的数据包所走的路径。加固策略为禁用该特性	0	是
net.ipv4.conf.default.accept_source_route		0	是

续表

加　固　项	加固项说明	加固建议	FusionOS 默认是否已加固为建议值
net. ipv4. conf. all. accept_redirects	是否发送 ICMP 重定向报文。加固策略为禁止发送	0	是
net. ipv4. conf. default. accept_redirects		0	是
net. ipv6. conf. all. accept_redirects		0	是
net. ipv6. conf. default. accept_redirects		0	是
net. ipv4. conf. all. send_redirects	是否将 ICMP 重定向报文发送至其他主机。只有当主机作为路由时，应启用该策略。加固策略为禁用该项	0	是
net. ipv4. conf. default. send_redirects		0	是
net. ipv4. icmp_ignore_bogus_error_responses	忽略伪造的 ICMP 数据包，不会将其记录到日志，将节省大量的硬盘空间。加固策略为启用该项	1	是
net. ipv4. tcp_syncookies	SYN Attack 是一种通过占用系统资源迫使系统重启的 DoS 攻击。加固策略为开启 TCP-SYN cookie 保护	1	是
kernel. dmesg_restrict	加固 dmesg 信息，仅允许管理员查看	1	是
kernel. sched_autogroup_enabled	该选项决定内核是否对线程进行自动分组调度。开启后调度组之间互相竞争时间片，调度组内的线程再竞争调度组分配到的时间片。加固策略为不启用该项	0	是
kernel. sysrq	禁用魔术键。 说明： 建议禁用魔术键，避免由于直接发送命令到内核对系统造成影响，增强内核安全性	0	是
net. ipv4. conf. all. secure_redirects	设置系统是接收来自任何主机的 ICMP 重定向消息还是从默认网关列表中的网关处接收 ICMP 重定向消息。加固策略为采用前者	0	是
net. ipv4. conf. default. secure_redirects		0	是
net. ipv6. conf. all. accept_ra	设置禁用系统接收 IPv6 路由器通告的能力	0	否
net. ipv6. conf. default. accept_ra		0	否

步骤 2　加载 sysctl. conf 文件中设置的内核参数。

```
sysctl -p  /etc/sysctl.conf
```

2. 其他安全建议

(1) net. ipv4. icmp_echo_ignore_all：忽略 ICMP 请求。

出于安全考虑，建议开启此项(默认值为 0，开启时设为 1，关闭时设为 0)。

但开启后会忽略所有接收的 icmp echo 请求的包(会导致机器无法 ping 通)，建议用户根据实际组网场景决定是否开启此项。

(2) net. ipv4. conf. all. log_martians/net. ipv4. conf. default. log_martians：对于仿冒/

源路由/重定向数据包开启日志记录。

出于安全考虑，建议开启此项(默认值为 0，开启时设为 1，关闭时设为 0)。

但是开启后会记录带有不允许的地址的数据到内核日志中，存在冲日志风险，建议用户根据实际使用场景决定是否开启此项。

(3) net.ipv4.tcp_timestamps：关闭 tcp_timestamps。

出于安全考虑，建议关闭 tcp_timestamps(默认值为 1，开启时设为 1，关闭时设为 0)。

但是关闭此项会影响 TCP 超时重发的性能，建议用户根据实际使用场景决定是否关闭此项。

(4) net.ipv4.tcp_max_syn_backlog：决定了 SYN_RECV 状态队列的数量。

该参数决定了 SYN_RECV 状态队列的数量，超过这个数量，系统将不再接收新的 TCP 连接请求，一定程度上可以防止系统资源耗尽。建议由用户根据实际使用场景配置合适的值。

3.2.6 SELinux 配置

1. 概述

自主访问控制(Discretionary Access Control，DAC)基于用户、组和其他权限，决定一个资源是否能被访问的因素是某个资源是否拥有对应用户的权限，它不能使系统管理员创建全面和细粒度的安全策略。SELinux(Security-Enhanced Linux)是 Linux 内核的一个模块，也是 Linux 的一个安全子系统。SELinux 实现了强制访问控制(Mandatory Access Control，MAC)，每个进程和系统资源都有一个特殊的安全标签，资源能否被访问除了 DAC 规定的原则外，还需要判断每一类进程是否拥有对某一类资源的访问权限。

FusionOS 默认使用 SELinux 提升系统安全性。SELinux 分为以下三种模式。

(1) permissive：SELinux 仅打印告警而不强制执行。

(2) enforcing：SELinux 安全策略被强制执行。

(3) disabled：不加载 SELinux 安全策略。

2. 配置说明

(1) 获取当前 SELinux 运行状态。

```
# getenforce
Enforcing
```

(2) SELinux 开启的前提下，设置运行状态为 enforcing 模式。

```
# setenforce 1
# getenforce
Enforcing
```

(3) SELinux 开启的前提下，设置运行状态为 permissive 模式。

```
# setenforce 0
# getenforce
Permissive
```

(4) SELinux 开启的前提下,设置当前 SELinux 运行状态为 disabled(关闭 SELinux,需要重启系统)。

① 修改 SELinux 配置文件/etc/selinux/config,设置“SELINUX=disabled”。

```
# cat /etc/selinux/config | grep "SELINUX = "
SELINUX = disabled
```

② 重启系统。

```
# reboot
```

③ 状态切换成功。

```
# getenforce
Disabled
```

(5) SELinux 关闭的前提下,设置 SELinux 运行状态为 permissive。

① 修改 SELinux 配置文件/etc/selinux/config,设置“SELINUX=permissive”。

```
# cat /etc/selinux/config | grep "SELINUX = "
SELINUX = permissive
```

② 在根目录下创建.autorelabel 文件。

```
# touch /.autorelabel
```

③ 重启系统,此时系统会重启两次。

```
# reboot
```

④ 状态切换成功。

```
# getenforce
Permissive
```

(6) SELinux 关闭的前提下,设置 SELinux 运行状态为 enforcing。

① 按照上一步骤所述,设置 SELinux 运行状态为 permissive。

② 修改 SELinux 配置文件/etc/selinux/config,设置“SELINUX=enforcing”。

```
# cat /etc/selinux/config | grep "SELINUX = "
SELINUX = enforcing
```

③ 重启系统。

```
# reboot
```

④ 状态切换成功。

```
# getenforce
Enforcing
```

3. SELinux 相关命令

查询运行 SELinux 的系统状态。SELinux status 表示 SELinux 的状态，enabled 表示启用 SELinux，disabled 表示关闭 SELinux。Current mode 表示 SELinux 当前的安全策略。

```
# sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:             targeted
Current mode:                   enforcing
Mode from config file:          enforcing
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:     actual (secure)
Max kernel policy version:      31
```

4. 注意事项

(1) 如用户需使能 SELinux 功能，建议通过 dnf 升级方式将 selinux-policy 更新为最新版本，否则应用程序有可能无法正常运行。升级命令示例：

```
dnf update selinux-policy -y
```

(2) 如果用户由于 SELinux 配置不当(如误删策略或未配置合理的规则或安全上下文)导致系统无法启动，可以在启动参数中添加 selinux=0，关闭 SELinux 功能，系统即可正常启动。

3.2.7　日志审计

说明

详细的日志信息可以帮助回溯历史操作，提高现网问题定位的效率。

因此虽然 OS 的管理员有权限管理审计记录，但是在删除日志时应做好备份，谨慎操作。

1. rsyslog 配置为将日志发送到远程日志主机

1) 说明

rsyslog 程序支持将其收集的日志发送到运行 syslogd(8)的远程日志主机或从远程主机接收消息，统一管理从而减少管理开销。

如果攻击者获得本地系统的 root 访问权限，可以篡改或删除存储在本地系统上的日志数据。在远程主机上存储日志数据可以在本机被攻击时保护日志完整性。

2) 实现

步骤 1　执行如下命令，编辑/etc/rsyslog.conf 文件。

```
vi /etc/rsyslog.conf
```

并添加以下行(其中,loghost. example. com 为远程日志服务器名称,根据实际情况替换)。

```
*.* @@loghost.example.com
```

步骤 2　重启 rsyslogd 以加载配置。

```
systemctl restart rsyslog
```

2. 设置仅在指定的日志主机上接收远程 rsyslog 消息

1) 说明

默认情况下,rsyslog 不监听来自远程系统的日志消息。rsyslog 可以通过配置 ModLoad 加载 imtcp. so 模块, 设定 InputTCPServerRun 参数,使 rsyslogd 监听指定的 TCP 端口。

应确保将远程日志主机配置为仅接收来自指定域内的主机的 rsyslog 数据,非日志主机的系统应不接收任何远程 rsyslog 消息。这样可以确保系统管理员在一个位置查看正确完整的系统日志数据。

2) 实现

步骤 1　执行如下命令,查看当前配置信息。

```
grep '$ ModLoad imtcp' /etc/rsyslog.conf
grep '$ InputTCPServerRun' /etc/rsyslog.conf
```

步骤 2　对于日志主机,编辑/etc/rsyslog. conf 文件。

```
vi /etc/rsyslog.conf
```

取消注释或添加以下行。

```
$ ModLoad imtcp
$ InputTCPServerRun 514
```

步骤 3　对于非日志主机,编辑/etc/rsyslog. conf 文件。

```
vi /etc/rsyslog.conf
```

注释或删除以下行。

```
# $ ModLoad imtcp
# $ InputTCPServerRun 514
```

步骤 4　重启 rsyslogd 以加载配置。

```
systemctl restart rsyslog
```

3. rsyslog 配置 daemon. debug 日志选项

1) 说明

默认情况下,rsyslog 不记录守护进程的 debug 级日志信息。设置 daemon. debug 提供

debug 级的日志输出,可以帮助管理员监管和调试守护进程。FusionOS 作为通用操作系统一般不进行调试工作环境,所以默认不开启此项设置。

2) 实现

步骤 1 执行如下命令,查看当前配置信息。

```
grep 'daemon.debug' /etc/rsyslog.conf
```

步骤 2 编辑/etc/rsyslog.conf 文件。

```
vi /etc/rsyslog.conf
```

取消注释或添加以下行。

```
daemon.debug FILE
```

步骤 3 重启 rsyslogd 以加载配置。

```
systemctl restart rsyslog
```

4. rsyslog 配置 kern.* 日志选项

1) 说明

默认情况下,rsyslog 不记录 kernel 日志信息。设置 kern.* 提供 kernel 日志输出,可以帮助管理员监管 kernel。FusionOS 默认不开启此项设置。

2) 实现

步骤 1 执行如下命令,查看当前配置信息。

```
grep 'kern.*' /etc/rsyslog.conf
```

步骤 2 编辑/etc/rsyslog.conf 文件。

```
vi /etc/rsyslog.conf
```

取消注释或添加以下行。

```
kern.* FILE
```

步骤 3 重启 rsyslogd 以加载配置。

```
systemctl restart rsyslog
```

3.2.8 防 DoS 攻击

网络服务可对 Linux 系统造成很多危险,其中一种是 DoS 攻击。

拒绝服务攻击(DoS):通过向服务发出大量请求,拒绝服务攻击可让系统无法使用,因为它会尝试记录并回应每个请求。

分布拒绝服务攻击(DDoS):一种 DoS 攻击类型,可使用多台被入侵的机器(经常是几

千台或者更多)对某个服务执行联合攻击,向其发送海量请求并使其无法使用。

FusionOS 主要通过防火墙技术(firewalld),SYN cookie 等技术来防止 DoS 攻击。

1. 防火墙

FusionOS 的防火墙基于开源的 firewalld 进行构建,防火墙是抵御网络攻击的第一道防线,它坐落于网络之间的枢纽点,保护某个网络以抵御来自其他网络的攻击。

通过以下命令可以操作防火墙服务。

(1) 查看防火墙状态。

```
systemctl status firewalld
```

(2) 开启防火墙。

```
systemctl start firewalld
```

(3) 关闭防火墙。

```
systemctl stop firewalld
```

2. SYN cookie

SYN Attack 是一种通过占用系统资源迫使系统重启的 DoS 攻击,FusionOS 可以开启 TCP-SYN cookie 保护。通过在内核加固参数配置文件/etc/sysctl. conf 中加入 net. ipv4. tcp_syncookies=1 开启,默认是开启状态。

3.2.9　安全启动

1. 特性描述

1) 背景

安全启动是统一可扩展固件接口(UEFI)规范的启动路径验证组件。

(1) 安全启动第一阶段。

shim. efi、grub. efi 及 kernel 由公司签名平台采用 signcode 方式进行签名,公钥证书由 BIOS 集成到签名数据库 DB 中,启动过程中 BIOS 对 shim 进行验证,shim 和 grub 组件对下一级组件进行验证。

(2) 安全启动第二阶段。

内核模块由公司签名平台采用 ELF 方式进行签名,公钥证书集成在内核中,内核加载 ko 模块时进行验证。

FusionOS 包括对 UEFI 安全启动功能的支持,这意味着 FusionOS 可以在启用了 UEFI 安全启动的系统上安装和运行。在启用了安全启动技术的基于 UEFI 的系统上,加载的所有驱动程序都必须使用有效证书进行签名,否则系统将不接收它们。

2) 定义

安全启动技术确保系统固件检查系统引导加载程序是否使用固件中包含的数据库授权

的加密密钥进行签名。通过在下一阶段的引导加载程序、内核以及可能的用户空间中进行签名验证,可以防止执行未签名的代码。

3) 目的和受益

安全启动特性的目的和受益如表 3-12 所示,FusionOS 22.0.1 版本及其后续版本支持该特性。

表 3-12 安全启动特性的目的和受益

目的和受益	详 细 说 明
防止启动非法文件	安全启动经过签名验证的文件,防止启动未验证通过的文件

2. 约束与限制

需要服务器厂商的 BIOS 支持 UEFI 安全启动和 FusionOS 证书。

3. 配置使用

BIOS 下配置。

3.2.10 初始化设置

本节中的建议适用于所有系统,但在系统初始设置后可能比较困难或需要大量准备工作。

1. 为全局使用的目录设置单独分区

1) 说明

对于全局使用的目录可以通过将其放在单独的分区来提供进一步的保护。这控制了这些目录资源耗尽的影响,并允许使用适用于目录预期用途的挂载选项。本节中的建议在初始系统安装期间更容易执行。

建议为如表 3-13 所示的全局目录设置单独的分区。

表 3-13 全局目录

目 录	说 明
/boot	用于存放内核文件与启动所需要的文件
/tmp	目录是一个全局可写目录,用于所有用户和某些应用程序的临时存储
/var	目录被守护进程和其他系统服务用来临时存储动态数据。这些进程创建的一些目录可能是全局可写的
/var/tmp	目录是一个全局可写目录,用于所有用户和某些应用程序的临时存储
/var/log	目录被系统服务用来存储日志数据
/var/log/audit	目录被审计守护进程 auditd 用来存储日志数据
/home	目录用于支持本地用户的磁盘存储需求

2) 实现

以/var 为例说明。

(1) 执行如下命令,检查/var 是否已设置单独分区。

```
mount | grep -E '\s/var\s'
```

(2) 对于新安装,在安装期间创建自定义分区设置并为/var 指定单独的分区。

(3) 对于已安装的系统,创建一个新分区并根据需要配置/etc/fstab。修改/var 时,建议将系统置于紧急模式(此时 auditd 不运行),重命名现有目录,挂载新文件系统,迁移数据然后返回多用户模式。

2. 加固分区挂载选项

1) 说明

安全相关挂载选项如表 3-14 所示。

表 3-14　安全相关挂载选项

选　　项	说　　明
nodev	挂载选项指定文件系统不能包含特殊设备
nosuid	挂载选项指定文件系统不能包含 setuid 文件
noexec	挂载选项指定文件系统不能包含可执行二进制文件

分区加载项建议采用如表 3-15 所示的方法设置。

表 3-15　分区加载项建议设置

分　　区	建议设置的选项	FusionOS 默认是否已加固为建议值
/tmp	nodev,nosuid,noexec	否
/dev/shm	nodev,nosuid,noexec	否
/home	nodev,nosuid,noexec	否
/var	nodev,nosuid,noexec	否
/var/tmp	nodev,nosuid,noexec	否
/var/log	nodev,nosuid,noexec	否
/var/log/audit	nodev,nosuid,noexec	否

2) 实现

以下示例以已挂载单独分区的/var,检查其 nodev 选项。

步骤 1　/var 分区存在时,使用以下命令验证是否设置 nodev,未设置返回分区信息,已设置不返回内容。

```
mount | grep -E '\s/var\s' | grep -v nodev
```

步骤 2　编辑/etc/fstab 文件并将 nodev 添加到/var 分区的第 4 个字段(安装选项)。

```
vi /etc/fstab
```

步骤 3　运行以下命令重新挂载/var。

```
mount -o remount,nodev /var
```

3. 安装 AIDE

1）说明

AIDE 是一款功能强大的开源入侵检测工具，它使用预定义的规则来检查 Linux 操作系统中文件和目录的完整性。AIDE 是 Tripwire 的简单的开源替代。

2）实现

步骤 1 执行如下命令，检查是否安装 AIDE。

```
rpm -qa | grep aide
```

步骤 2 根据需要执行如下命令进行安装。

```
dnf install aide
```

步骤 3 根据环境配置 AIDE。有关选项，请参阅 aide --help。

执行如下命令，初始化 AIDE。

```
aide -- init
mv /var/lib/aide/aide.db.new.gz /var/lib/aide/aide.db.gz
```

3.2.11 网络用户认证

须知

FusionOS 默认使用基于用户名密码本地认证机制，并进行了默认加固。若客户配置使用第三方认证机制，需要确认是否存在口令复杂度校验、防暴力破解、防 DoS 攻击等机制，建议进行重复测试验证。

1. 特性描述

1）背景

当网络中的多个系统都要访问公共资源时，所有用户和组身份对于该网络中的所有计算机而言是否相同就显得极其重要。网络应该对用户透明：不管用户实际正在使用哪台计算机，其环境都不应该有变化。可以通过 NIS、LDAP 服务完成此操作。Kerberos 是一个网络身份验证协议，同时还提供加密，可以与 LDAP 集成使用。

2）定义

（1）NIS（Network Information Service）最早被称为 Sun Yellow Pages（简称 YP）。NIS 服务的应用结构分为 NIS 服务器和 NIS 客户机两种角色，NIS 服务器集中维护用户的账号信息（数据库）供 NIS 客户机进行查询，用户登录任何一台 NIS 客户机都会从 NIS 服务器进行登录认证。NIS 服务器也可以使用 master/slave 架构，一方面分散 master NIS 服务器的负载，也可以避免因 master NIS 服务器异常而导致的无法登录的风险。

（2）LDAP（Lightweight Directory Access Protocol）是一种目录访问协议，LDAP 服务器提供目录存储及访问服务。LDAP 目录结构树如图 3-2 所示。

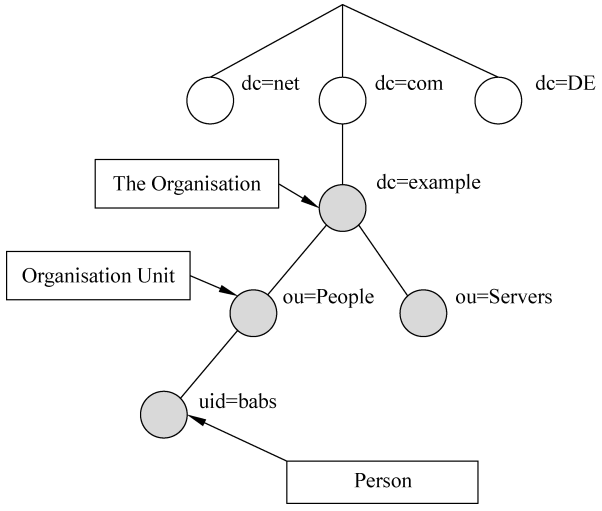

图 3-2　LDAP 目录结构树示意图

dc：domain component。

ou：organization unit。

uid：user id。

cn：common name。

根据需求可以新增和修改相应的节点。每个节点都是一个条目，每个条目通过 DN（Distingguished Name）进行区分。

（3）Kerberos：Kerberos 是一种计算机网络的授权协议，用在非安全网络环境中，对个人通信以安全的手段进行身份认证。用户端和服务器端均可向对方进行身份认证。Kerberos 使用对称加密。在整个 Kerberos 系统中，有一个公认的第三方被称为 KDC（Key Distribution Center），所有的主机（不论是服务提供端还是用户端）都需要加入 Kerberos 服务的域内，所有主机都必须向 KDC 请求一个 ticket，根据这个 ticket 来跟 KDC 验证，验证后给予加密的密钥。KDC 除了发放票据 ticket，还负责身份验证 AS（Authentication Server）。

Kerberos 认证流程如图 3-3 所示。

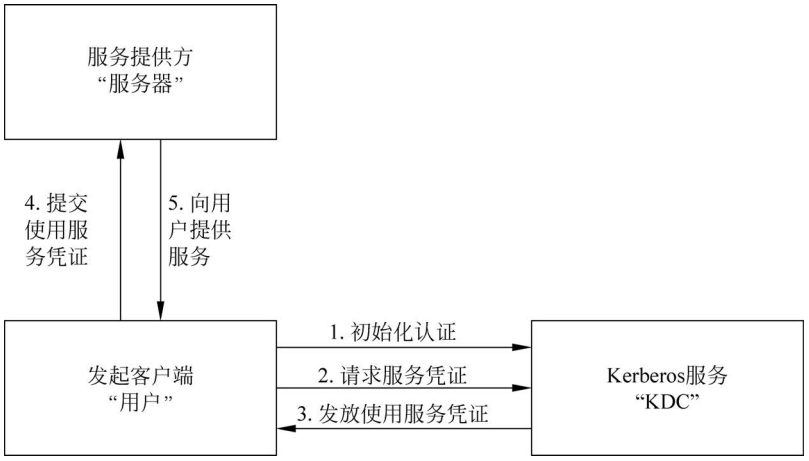

图 3-3　Kerberos 认证流程

说明

- principal：参加认证的实体，相当于用户名，用来表示一个 client 或 service server 的唯一的身份。principal 的格式是：name/instance@REALM。
- Kerberos 不做用户信息管理。krb5 网络认证本地用户和 LDAP 用户使用 krb5 认证下的用户端设置将以用户信息管理使用 LDAP，认证使用 Kerberos5 为例讲述搭建过程。

3）目的和受益

网络用户认证特性的目的和受益如表 3-16 所示，FusionOS 22.0.1 版本及其后续版本支持该特性。

表 3-16 网络用户认证特性的目的和受益

目的和受益	详 细 说 明
集中管理网络中的用户	网络用户的信息在集中的服务器上进行管理，用户可在任一用户端上登录

2. NIS 配置使用

1）事先规划及设定

NIS 网域名称：domaintest。

NIS master server：假设 IP 地址为 90.90.112.66，hostname 为 server.domaintest。

NIS slave server：假设 IP 地址为 90.90.113.193，hostname 为 slave.domaintest。

NIS client：假设 IP 地址为 90.90.114.246，hostname 为 client.domaintest。

hostname 设置方法举例：

```
hostnamectl set-hostname server.domaintest
```

2）NIS 服务器搭建

步骤 1 关闭防火墙。

```
systemctl stop firewalld
```

说明

如需开启防火墙，请参照上面的服务器端防火墙设置进行配置。

步骤 2 安装 ypserv、rpcbind、ypbind、yp-tools。

```
yum install -y ypserv rpcbind ypbind yp-tools
```

步骤 3 设置 NIS 的网域名称。

（1）执行如下命令，查询 NIS 的网域名称。

```
nisdomainname
```

（2）执行如下命令，设置 NIS 的网域名称。

```
nisdomainname domaintest
vi /etc/sysconfig/network
```

在最下面加入一行：

```
NISDOMAIN = domaintest
```

步骤 4　设置服务器端配置文件。

```
vi /etc/ypserv.conf
```

在最下面设置 NIS 域的查询权限，其他设置为拒绝，如下所示。

```
90.90.112.66       : *        : *                  : none
90.90.113.193      : *        : *                  : none
90.90.114.246      : *        : *                  : none
*                  : *        : *                  : deny
```

步骤 5　设置 hostname 和 IP 对应。

```
vi /etc/hosts
新增 NIS 域中 IP 和 hostname 对应关系：
90.90.112.66   server server.domaintest
90.90.114.246 client client.domaintest
90.90.113.193 slave   slave.domaintest
```

步骤 6　启动 rpcbind、ypserv、yppasswdd 服务。

```
systemctl start rpcbind ypserv yppasswdd
systemctl enable rpcbind ypserv yppasswdd
```

步骤 7　添加账户及创建 NIS 数据库。

```
useradd nisuser1 -d /home/rhome/nisuser1
passwd nisuser1
```

输入 nisuser1 的密码如 123。

```
touch /etc/netgroup
touch /etc/publickey
/usr/lib64/yp/ypinit -m
```

依次输入 NIS server hosts：server、slave(根据实际是否存在决定是否输入)；Ctrl+D；y。设置 NIS 域的查询权限示意图如图 3-4 所示。

说明

- 账户及密码等信息变更后，需要重新创建数据库，并重启服务。
- 数据库执行命令 ypinit -m 创建，无法修改名称。
- Ctrl+D：在交互界面输入 NIS server 的列表后，结束输入。
- y：再次确认输入的列表是否正确。

3）NIS 主从服务器搭建

(1) 主服务器端开启推送及 ypxfrd 服务。

先按 NIS 服务器搭建设置好主服务器，再进行如下操作。

图 3-4　设置 NIS 域的查询权限示意图

步骤 1　开启推送。

```
vi /var/yp/Makefile
```

设置如下内容。

```
NOPUSH = false
```

步骤 2　指定 slave 主机。

```
vi /var/yp/ypservers
```

添加 slave。

步骤 3　创建数据库同时向从服务器推送数据库。

```
cd /var/yp
make
```

须知

从服务器端需关闭防火墙或者按下一步设置。

主服务器端在/var/yp/Makefile 中设置 YPPUSH_ARGS 为固定端口，从服务器端设置防火墙放行该端口。

ypinit 时不启用推送功能。

步骤 4　启动 ypxfrd 服务，允许从服务器主动同步数据库。

```
systemctl start ypxfrd
systemctl enable ypxfrd
```

(2) 从服务器端搭建。

步骤 1　关闭防火墙。

```
systemctl stop firewalld
```

步骤 2　安装 ypserv、rpcbind、ypbind、yp-tools。

```
yum install -y ypserv rpcbind ypbind yp-tools
```

步骤 3　设置 NIS 的网域名称。

① 执行如下命令，查询 NIS 的网域名称。

```
nisdomainname
```

② 执行如下命令，设置 NIS 的网域名称。

```
nisdomainname domaintest
vi /etc/sysconfig/network
```

在最下面加入一行：

```
NISDOMAIN = domaintest
```

步骤 4　设置服务器端配置文件。

```
vi /etc/ypserv.conf
```

在最下面设置 NIS 域的查询权限，其他设置为拒绝，如下所示。

```
90.90.112.66      : *       : *                  : none
90.90.113.193     : *       : *                  : none
90.90.114.246     : *       : *                  : none
*                 : *       : *                  : deny
```

步骤 5　设置 hostname 和 IP 对应。

```
vi /etc/hosts
```

新增 NIS 域中 IP 和 hostname 对应关系。

```
90.90.112.66 server server.domaintest
90.90.114.246 client client.domaintest
90.90.113.193 slave slave.domaintest
```

步骤 6 设置 NIS server。

```
vi /etc/yp.conf
```

添加如下内容。

```
domain domaintest server 90.90.112.66
```

步骤 7 启动 rpcbind、ypserv、yppasswdd 服务。

```
systemctl start rpcbind ypserv yppasswdd
systemctl enable rpcbind ypserv yppasswdd
```

步骤 8 从主服务器获取数据库。

```
/usr/lib64/yp/ypinit -s server.domaintest
```

查看是否获取数据。

```
ypcat -h slave.domaintest passwd.byname
```

步骤 9 设置定时跟主服务器同步数据库。

① 设置 1 小时更新一次。

```
vi /usr/lib64/yp/ypxfr_1perhour
```

将 $YPBINDIR/ypxfr $map 修改成 $YPBINDIR/ypxfr $map -h server.domaintest。

MAPS_TO_GET=修改为实际要同步的文件。

```
vi /etc/crontab
```

添加如下内容。

```
@hourly root /usr/lib64/yp/ypxfr_1perhour > /dev/null 2 > &1
```

② 设置 5 分钟同步一次。

```
vi /etc/crontab
```

添加如下内容。

```
*/5 * * * *  /usr/lib64/yp/ypinit -s server.domaintest
```

说明

- 执行如下命令，可以查看从服务器上文件是否有同步。

```
ll /var/yp/domaintest/
```

- 通过 ypxfr 更新数据库文件，需要主从服务器上的文件不一致，如果一致主服务器上没有更新，则不会启动同步。

4）NIS 用户端搭建

步骤 1　安装 rpcbind、ypbind、yp-tools、authselect。

```
yum install -y rpcbind ypbind yp-tools authselect
```

步骤 2　设置 NIS 的网域名称。

（1）执行如下命令，查询 NIS 的网域名称。

```
nisdomainname
```

（2）执行如下命令，设置 NIS 的网域名称。

```
nisdomainname domaintest
vi /etc/sysconfig/network
```

在最下面加入一行。

```
NISDOMAIN=domaintest
```

步骤 3　设置 hostname 和 IP 对应。

```
vi /etc/hosts
```

新增 NIS 域中 IP 和 hostname 对应关系。

```
90.90.112.66 server server.domaintest
90.90.114.246 client client.domaintest
90.90.113.193 slave slave.domaintest
```

步骤 4　设置 NIS server。

```
vi /etc/yp.conf
```

添加 domain domaintest server 90.90.112.66。

如果设置了从服务器，则把从服务器也添加进去：domain domaintest server 90.90.113.193。

步骤 5　启动 rpcbind、ypbind 服务。

（1）启动 rpcbind、ypbind 服务，并设置为开机启动。

```
systemctl start rpcbind ypbind
systemctl enable rpcbind ypbind
```

（2）执行如下命令，测试是否能连接 NIS 服务器。

```
yptest
```

（3）执行如下命令，测试连接哪台 NIS 服务器。

```
ypwhich
```

步骤 6 选择使用 NIS 进行用户认证。

```
mkdir -p /etc/authselect
authselect select nis --force
```

说明

NIS client 会先去本机/etc/passwd、/etc/shadow 进行查询，查询不到才会到 NIS server 上进行查询。

步骤 7 测试。

(1) 本地登录。

使用用户名 nisuser1，输入密码 123 可以登录。

(2) ssh 登录。

执行如下命令，使用 ssh 可以登录。

```
ssh nisuser1@90.90.114.246
```

5) 家目录设置

家目录服务器需用户根据实际进行规划，下面以主服务器作为家目录服务器进行设置。

假设网络用户 nisuser1 的家目录为/home/rhome/nisuser1。

(1) 主服务器端设置。

步骤 1 执行如下命令，创建共享目录和用户的家目录。

```
mkdir -p /home/nfs-server/
mkdir -p /home/nfs-server/nisuser1
```

并设置/home/nfs-server/nisuser1 目录的属主为 nisuser1。

步骤 2 设置服务器端共享目录。

```
vi /etc/exports
```

增加/home/nfs-server 90.90.114.246(rw,sync)。

共享服务器端的/home/nfs-server 目录可被用户端 90.90.114.246 访问。

步骤 3 执行如下命令，启动 NFS。

```
systemctl start nfs
systemctl enable nfs
```

(2) 用户端设置。

手动挂载 NFS 服务器分享的资源。

步骤 1 查询 NFS 服务器提供哪些资源。

```
showmount -e 90.90.112.66
```

其中，90.90.112.66 为 NFS 服务器端 IP。

步骤 2　建立挂载点，并挂载。

```
mkdir -p /home/rhome/
```

该挂载点须与网络用户的家目录保持一致。

```
mount -t nfs 90.90.112.66:/home/nfs-server /home/rhome/
```

步骤 3　查询是否挂载成功，使用 df 或者 mount 命令。

```
df
```

步骤 4　网络用户登录，查看家目录是否正确，是否能访问。

```
ssh nisuser1@90.90.114.246
```

执行如下命令，显示当前目录的绝对路径。

```
pwd
```

当前用户家目录为“/home/rhome/nisuser1”。

使用 autofs 自动挂载：持续侦测某个指定的目录，并预先设定当使用到该目录下的某个次目录时，将会取得来自服务器端的 NFS 文件系统资源，并进行自动挂载的动作。

步骤 1　执行如下命令，安装 autofs。

```
yum install -y autofs
```

步骤 2　建立主配置文件/etc/auto.master，并指定侦测的特定目录。

```
mkdir -p /home/rhome/
```

该侦测目录须与网络用户的家目录保持一致。

```
vi /etc/auto.master
```

增加/home/rhome、/etc/auto.nfs。

步骤 3　建立数据对应文件内(/etc/auto.nfs)的挂载信息与服务器对应资源。

格式为

```
[本地端次目录] [-挂载参数] [服务器所提供的目录]
```

选项与参数如下。

[本地端次目录]：指在/etc/auto.master 内指定的目录之次目录。

[-挂载参数]：rw、bg、soft 等的参数，可选。

[服务器所提供的目录]：例如 192.168.100.254:/home/public。

```
vi /etc/auto.nfs
```

打开配置文件：

```
vi /etc/auto.nfs
```

配置文件中增加如下配置：

```
*  -fstype=nfs,rw,local_lock=all,vers=3 10.90.112.66:/home/nfs-server/&
```

步骤 4 执行如下命令，启动 autofs 服务。

```
systemctl start autofs
systemctl enable autofs
```

步骤 5 网络用户登录，查看家目录是否正确，是否能访问。

```
ssh nisuser1@90.90.114.246
```

执行如下命令，显示当前目录的绝对路径。

```
pwd
```

当前用户家目录为“/home/rhome/nisuser1”。

说明

创建用户时的家目录路径跟用户端 mount 的路径要一致，否则用户登录时也会报无法找到对应的家目录路径。

6）服务器端防火墙设置

（1）设置 NIS 相关服务为固定端口。

步骤 1 设置 ypserv 为固定端口。

① 配置 network 文件。

```
vi /etc/sysconfig/network
```

添加 YPSERV_ARGS="-p 1011"。

② 执行如下命令，重启 ypserv 服务。

```
systemctl restart ypserv
```

步骤 2 设置 yppasswdd 为固定端口。

① 配置 yppasswdd 文件。

```
vi /etc/sysconfig/yppasswdd
```

修改 YPPASSWDD_ARGS="--port 1013"。

② 执行如下命令，重启 yppasswdd 服务。

```
systemctl restart yppasswdd
```

步骤 3　设置 ypxrfd 为固定端口。

① 配置 network 文件。

```
vi /etc/sysconfig/network
```

添加 YPXFRD_ARGS="-p 1015"。

须知

/etc/ypserv.conf 中 xfr_check_port 为 yes 则只允许 xfr 请求端口小于 1024。

② 执行如下命令，重启 ypxfrd 服务。

```
systemctl restart ypxfrd
```

步骤 4　执行如下命令，查询 NIS 服务相关端口。

```
rpcinfo -p localhost
```

(2) 防火墙增加对 NIS 相关端口的放行。

```
firewall-cmd --permanent --add-service=rpc-bind
firewall-cmd --permanent --add-port=1011/tcp
firewall-cmd --permanent --add-port=1011/udp
firewall-cmd --permanent --add-port=1013/tcp
firewall-cmd --permanent --add-port=1013/udp
firewall-cmd --permanent --add-port=1015/tcp
firewall-cmd --permanent --add-port=1015/udp
```

执行如下命令，重启 firewalld 服务。

```
systemctl restart firewalld
```

该部分仅需要在家目录服务器上设置。

```
firewall-cmd --permanent --add-port=20048/tcp
firewall-cmd --permanent --add-port=20048/udp
firewall-cmd --permanent --add-port=2049/tcp
firewall-cmd --permanent --add-port=2049/udp
```

执行如下命令，重启 firewalld 服务。

```
systemctl restart firewalld
```

3. LDAP 配置使用

1) 事先规划及设定

(1) LDAP 域名 baseDN(Suffix)：test.com。

(2) LDAP 管理员账户 RootDN：Manager.test.com。

(3) LDAP master server：假设 IP 地址为 90.90.112.64，hostname 为 ldapserver.test.com。

(4) LDAP slave server：假设 IP 地址为 90.90.114.165，hostname 为 ldapconsumerserver.test.com。

(5) LDAP client：假设 IP 地址为 90.90.112.65，hostname 为 ldapclient.test.com。

(6) hostname 设置。

```
hostnamectl set-hostname ldapserver.test.com
```

(7) 设置 hostname 与 IP 的对应关系，域内的每个机器上/etc/hosts 都需要设置。

```
vi /etc/hosts
```

每行添加 IP 地址及对应的 hostname，需要将域内的服务器、用户端都加上。

2) LDAP 服务器搭建

步骤 1 关闭防火墙。

```
systemctl stop firewalld
```

步骤 2 安装 openldap、openldap-servers、openldap-clients。

```
yum install -y openldap openldap-servers openldap-clients
```

步骤 3 开启 slapd 服务。

```
systemctl start slapd
systemctl enable slapd
```

步骤 4 设置域名、管理员账户和密码。

```
mkdir ldap
cd ldap
slappasswd
```

输入管理员密码如 123，slappasswd 会生成加密后的密码{SSHA}IbjjYQw+bZxaCsBGaLuWsONiwWQPvurs。

说明

LDAP 支持多种密码存储方式，如{CRYPT}，{MD5}，{SMD5}，{SSHA}，and{SHA}，slappasswd 可通过-h 来指定生成密码的方式，默认使用{SSHA}。

建立一个 basedn.ldif 文件。

```
vi basedn.ldif
```

basedn.ldif 文件内容如下，basedn.ldif 文件配置项说明如表 3-17 所示。

```
dn: olcDatabase={2}mdb,cn=config
changetype: modify
replace: olcSuffix
olcSuffix: dc=test,dc=com

dn: olcDatabase={2}mdb,cn=config
changetype: modify
```

```
replace: olcRootDN
olcRootDN: cn = Manager,dc = test,dc = com

dn: olcDatabase = {2}mdb,cn = config
changetype: modify
replace: olcRootPW
olcRootPW: {SSHA}IbjjYQw + bZxaCsBGaLuWsONiwWQPvurs

dn: olcDatabase = {1}monitor,cn = config
changetype: modify
replace: olcAccess
olcAccess:{0}to * by dn.base = "gidNumber = 0 + uidNumber = 0,cn = peercred,cn = external,cn =
auth" read by dn.base = "cn = Manager,dc = test,dc = com" read by * none
```

表 3-17　basedn.ldif 文件配置项说明

配　置　项	说　　明
olcSuffix	LDAP 数据库的后缀。 示例中后缀设置为"dc＝test,dc＝com",意味着所有以"dc＝test,dc＝com"结尾的条目都将存储在这个数据库中
olcRootDN	LDAP 数据库的根 DN(Distinguished Name,区分名)。 示例中根 DN 设置为"cn＝Manager,dc＝test,dc＝com"
olcRootPW	LDAP 数据库的根密码。 示例中根密码为"IbjjYQw＋bZxaCsBGaLuWsONiwWQPvurs"
olcAccess	定义了谁可以访问 OpenLDAP 数据库中的条目。 示例中允许"gidNumber＝0＋uidNumber＝0,cn＝peercred,cn＝external,cn＝auth"和"cn＝Manager,dc＝test,dc＝com"访问所有条目,其他用户不能访问

说明

如果原数据库无管理员密码即无 olcRootPW 则 replace：olcRootPW 修改为 add：olcRootPW。

内容跟第一部分事先规划及设定中保持一致。

执行如下命令,将修改内容更新到数据库中。

```
ldapmodify -Y EXTERNAL  -H ldapi:/// -f basedn.ldif
```

说明

- slapd 的配置文件在/etc/openldap/slapd.d/下,cn＝config 是根节点,包含全局性的设置,下面可以有 cn＝Module(dynamically loaded modules)、cn＝Schema(schema definitions)、olcBackend ＝ xxx(backend-specific settings)、olcDatabase ＝ xxx(database-specific settings)4 类子节点。
- 配置文件的修改通过 ldif 文件及相关命令完成,修改后不需要重启 slapd 服务。
- ldif 文件中每个条目之间用一个空行分隔。

步骤 5　添加一些必需的模式库,依次添加 cosine、nis、inetorgperson。

```
ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/cosine.ldif
ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/nis.ldif
ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/inetorgperson.ldif
```

步骤 6 设置 Linux 账户所需要的节点：账户和群组。

账户的 dn 为 ou＝People，dc＝test，dc＝com。

群组的 dn 为 ou＝Group，dc＝test，dc＝com。

执行如下命令，建立一个 base.ldif 文件。

```
vi base.ldif
```

base.ldif 文件内容如下。

```
dn: dc = test,dc = com
objectClass: top
objectClass: dcObject
objectClass: organization
o: test com
dc: test

dn: cn = Manager,dc = test,dc = com
objectClass: organizationalRole
cn: Manager
description: Directory Manager

dn: ou = People,dc = test,dc = com
objectClass: organizationalUnit
ou: People

dn: ou = Group,dc = test,dc = com
objectClass: organizationalUnit
ou: Group
```

执行如下命令，将修改内容更新到数据库中。

```
ldapadd -x -W -D "cn = Manager,dc = test,dc = com" -f base.ldif
```

输入上面设定的管理员密码 123。

步骤 7 添加账户和群组。

新建一个用户 ldapuser1，密码为 mypassword123。

执行如下命令，建立一个 user.ldif 文件。

```
vi user.ldif
```

user.ldif 文件内容如下。

```
dn: uid = ldapuser1,ou = People,dc = test,dc = com
uid:ldapuser1
cn: ldapuser1
sn: ldapuser1
mail: ldapuser1@test.com
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: posixAccount
```

```
objectClass: top
objectClass: shadowAccount
userPassword: {SSHA}qvpw8W7IOdCDMQaGQf9BczBjwMFD5n3n
shadowLastChange: 18585
shadowMin: 0
shadowMax: 99999
shadowWarning: 7
loginShell: /bin/bash
uidNumber: 60001
gidNumber: 60001
homeDirectory: /home/rhome/ldapuser1
```

执行如下命令，将修改内容更新到数据库中。

```
ldapadd -x -W -D "cn=Manager,dc=test,dc=com" -f user.ldif
```

输入上面设定的管理员密码 123。

新建一个群组 ldapuser1。

执行如下命令，建立一个 group.ldif 文件。

```
vi group.ldif
```

group.ldif 文件内容如下。

```
dn: cn=ldapuser1,ou=Group,dc=test,dc=com
objectClass: posixGroup
objectClass: top
cn: ldapuser1
userPassword: {crypt}x
gidNumber: 6001
```

执行如下命令，将修改内容更新到数据库中。

```
ldapadd -x -W -D "cn=Manager,dc=test,dc=com" -f group.ldif
```

输入上面设定的管理员密码 123。

步骤 8　查询测试。

```
ldapsearch -x -H ldap://ldapserver.test.com -b "dc=test,dc=com"
ldapsearch -x -H ldap://ldapserver.test.com -b "dc=test,dc=com" uid=ldapuser1
```

说明

LDAP 服务器是否运行匿名连接(无用户名和无密码，即任何人可以去查询 LDAP 数据库，但是要访问数据库中的 entry 需要输入对应的凭证)；匿名连接是通过 access control 控制的，就是数据库中的 olcAccess 属性。

如果不允许匿名连接则需要提供 bind DN 和 bind password，在/etc/openldap/ldap.conf 中指定 bindDN 或在命令行中-D 指定。

3) LDAP 复制目录搭建

复制目录存在提供者/多提供者和消费者：提供者可以接受外部写入操作，并使它们可供消费者检索；消费者向提供者请求复制更新。下面介绍使用 Sync Replication engine(简称 Syncrepl)来复制目录的方法。

(1) 提供者服务器端配置。

在 LDAP 服务器搭建完成之后增加如下步骤。

步骤 1 导出提供者服务器上数据，并传到消费者服务器上。

执行如下命令，导出提供者服务器上数据。

```
slapcat -n 0 -l ldap-config.ldif
slapcat -n 2 -l ldap-data.ldif
```

执行如下命令，传到消费者服务器上。

```
scp ldap-config.ldif root@90.90.114.165:/root/
scp ldap-data.ldif root@90.90.114.165:/root/
```

步骤 2 使能索引。

执行如下命令，新建 enable_indexing.ldif 文件。

```
vi enable_indexing.ldif
```

enable_indexing.ldif 文件内容如下。

```
dn: olcDatabase={2}mdb,cn=config
changetype: modify
add: olcDbIndex
olcDbIndex: entryCSN eq

add: olcDbIndex
olcDbIndex: entryUUID eq
```

执行如下命令，添加到数据库中。

```
ldapadd -Y EXTERNAL -H ldapi:/// -f enable_indexing.ldif
```

步骤 3 添加 syncprov 模块。

执行如下命令，新建 addmod_syncprov.ldif 文件。

```
vi addmod_syncprov.ldif
```

addmod_syncprov.ldif 文件内容如下。

```
dn: cn=module,cn=config
objectClass: olcModuleList
cn: module
olcModulePath: /usr/lib64/openldap
olcModuleLoad: syncprov.la
```

执行如下命令，添加到数据库中。

```
ldapadd -Y EXTERNAL -H ldapi:/// -f addmod_syncprov.ldif
```

步骤 4　配置 syncprov 模块。

执行如下命令，新建 syncprov.ldif 文件。

```
vi syncprov.ldif
```

syncprov.ldif 文件内容如下。

```
dn: olcOverlay=syncprov,olcDatabase={2}mdb,cn=config
objectClass: olcOverlayConfig
objectClass: olcSyncProvConfig
olcOverlay: syncprov
olcSpCheckpoint: 100 10
olcSpSessionLog: 100
```

执行如下命令，添加到数据库中。

```
ldapadd -Y EXTERNAL -H ldapi:/// -f syncprov.ldif
```

(2) 消费者服务器端搭建。

步骤 1　关闭防火墙。

```
systemctl stop firewalld
```

步骤 2　安装 openldap、openldap-servers、openldap-clients。

```
yum install -y openldap openldap-servers openldap-clients
```

步骤 3　使用提供者服务器上的数据配置消费者服务器。

执行如下命令，清除消费者服务器上的数据。

```
rm -rf /etc/openldap/slapd.d/*
rm -rf /var/lib/ldap/*
```

执行如下命令，配置消费者服务器。

```
slapadd -n 0 -l ldap-config.ldif -F /etc/openldap/slapd.d/
slapadd -n 2 -l ldap-data.ldif -F /etc/openldap/slapd.d/
```

执行如下命令，修改属主。

```
chown -R ldap:ldap /etc/openldap/slapd.d/ /var/lib/ldap/
```

步骤 4　开启 slapd 服务。

```
systemctl start slapd
systemctl enable slapd
```

步骤 5 增加一个 syncrepl。

执行如下命令，新建一个 syncrepl.ldif 文件。

```
vi syncrepl.ldif
```

syncrepl.ldif 文件内容如下。

```
dn: olcDatabase={2}mdb,cn=config
changetype: modify
add: olcSyncRepl
olcSyncRepl: rid=001
  provider=ldap://ldapserver.test.com
  bindmethod=simple
  binddn="cn=Manager,dc=test,dc=com"
  credentials=123
  searchbase="dc=test,dc=com"
  scope=sub
  schemachecking=on
  type=refreshAndPersist
  attrs="*,+"
  retry="30 5 300 3"
  interval=00:00:05:00
```

说明

- rid 表示 syncrepl 的编号，有多个时编号不能重复。
- provider 开始的内容每行空两个空格。
- credentials：binddn 的密码。
- retry：格式为[retry interval] [retry times] [interval of re-retry] [re-retry times]。
- type：同步类型，支持 pull-base 和 push-base 方式，type＝refreshOnly 表示使用 pull-base 方式，消费者定期轮询提供者以获取更新。type＝refreshAndPersist 表示使用 push-base 方式，消费者监听提供者实时发送的更新。

执行如下命令，将修改内容更新到数据库中。

```
ldapadd -Y EXTERNAL -H ldapi:/// -f syncrepl.ldif
```

(3) 提供者和消费者同步测试。

步骤 1 在提供者服务器上新增一个账户。

执行如下命令，新建一个 ldaprptest.ldif 文件。

```
vi ldaprptest.ldif
```

ldaprptest.ldif 文件内容如下。

```
dn: uid=ldaprptest,ou=People,dc=test,dc=com
objectClass: top
objectClass: account
objectClass: posixAccount
objectClass: shadowAccount
cn: ldaprptest
```

```
uid: ldaprptest
uidNumber: 9988
gidNumber: 100
homeDirectory: /home/rhome/ldaprptest
loginShell: /bin/bash
gecos: LDAP Replication Test User
userPassword: {crypt}x
shadowLastChange: 17058
shadowMin: 0
shadowMax: 99999
shadowWarning: 7
```

执行如下命令，将修改内容更新到数据库中。

```
ldapadd -x -w 123 -D "cn=Manager,dc=test,dc=com" -f ldaprptest.ldif
```

步骤 2　在消费者服务器上搜索新增的账户。

```
ldapsearch -x cn=ldaprptest -b dc=test,dc=com
```

能查询到主服务上新增的 ldaprptest 信息，说明提供者和消费者数据库同步成功。

(4) 用户端配置。

① LDAP 配置文件增加消费者服务器。

```
vi /etc/openldap/ldap.conf
```

在 URI 那一行后面增加空格 ldap://ldapconsumerserver.test.com。

② 如果用户端使用 nss-pam-ldapd 搭建，按如下步骤增加消费者服务器。

```
vi /etc/nslcd.conf
```

在 URI 那一行后面增加空格 ldap://ldapconsumerserver.test.com。

执行如下命令，重启 nslcd 服务。

```
systemctl restart nslcd
```

③ 如果用户端使用 sssd 搭建，按如下步骤增加消费者服务器。

```
vi /etc/sssd/sssd.conf
```

在 ldap_uri 那一行后面增加，ldap://ldapconsumerserver.test.com。

执行如下命令，重启 sssd 服务。

```
systemctl restart sssd
```

4) LDAP 用户端搭建

用户端有两种方案可选择，一种是使用 nss-pam-ldapd，另一种是使用 sssd。两者的区别是 nss-pam-ldapd 没有本地缓存功能，而 sssd 有。

(1) 使用 nss-pam-ldapd 搭建流程。

步骤 1 安装 nss-pam-ldapd、openldap-clients。

```
yum install -y nss-pam-ldapd openldap-clients
```

步骤 2 设置 LDAP 服务器地址及 basedn。

```
vi /etc/openldap/ldap.conf
```

设置 URI 为 ldap://ldapserver.test.com。

设置 BASE 为 dc=test,dc=com。

```
vi /etc/nslcd.conf
```

设置 URI 为 ldap://ldapserver.test.com。

设置 BASE 为 dc=test,dc=com。

步骤 3 修改账户查询顺序。

```
vi  /etc/nsswitch.conf
```

将所有 sss 修改为 ldap。

步骤 4 修改账户认证方式。

```
vi  /etc/pam.d/system-auth
```

将所有 pam_sss.so 所在行的最前面的-去掉,并将 sss 修改为 ldap。

```
vi/etc/pam.d/password-auth
```

将所有 pam_sss.so 所在行的最前面的-去掉,并将 sss 修改为 ldap。

说明

如果 password-auth 不修改,ssh 登录会出现 Permission denied。原因是/etc/pam.d/sshd 中包含 password-auth。

步骤 5 启动 nslcd 服务。

```
systemctl start nslcd
systemctl enable nslcd
```

步骤 6 验证登录。

执行如下命令,显示用户信息。

```
id ldapuser1
```

① 本地登录。

输入用户名 ldapuser1,输入密码 mypassword123,可以登录。

② ssh 登录。

执行如下命令,使用 ssh 可以登录。

```
ssh ldapuser1@90.90.112.65
```

(2) 使用 sssd 搭建流程。

步骤 1　安装 authselect、sssd、openldap、openldap-clients。

```
yum install -y authselect sssd openldap openldap-clients
```

步骤 2　配置。

① 配置 ldap.conf 文件,设置 LDAP 服务器地址及 basedn。

```
vi /etc/openldap/ldap.conf
```

设置 URI 为 ldap://ldapserver.test.com。

设置 BASE 为 dc=test,dc=com。

② 配置 sssd.conf 文件。

```
vi /etc/sssd/sssd.conf
```

设置使用的 domains 为 LDAP。

```
[sssd]
domains = LDAP
```

设置[domain/LDAP]为如下内容。

```
[domain/LDAP]
id_provider = ldap
auth_provider = ldap
chpass_provider = ldap
access_provider = permit
ldap_uri = ldap://ldapserver.test.com
ldap_search_base = dc=test,dc=com
ldap_default_authtok_type = password
ldap_tls_cacert = /etc/openldap/certs/cacert.pem
ldap_tls_reqcert = hard
```

步骤 3　选择使用 sssd。

```
mkdir -p /etc/authselect
authselect select sssd --force
```

步骤 4　启动 sssd 服务。

```
systemctl start sssd
systemctl enable sssd
```

步骤 5　验证登录。

执行如下命令,显示用户信息。

```
id ldapuser1
```

① 本地登录。

输入用户名 ldapuser1,输入密码 mypassword123,可以登录。

② ssh 登录。

执行如下命令,使用 ssh 可以登录。

```
ssh ldapuser1@90.90.112.65
```

说明

- 如果要通过用户端的 sssd 进行 LDAP 认证,必须设置 TLS 加密,sssd 不支持不加密的认证通道。
- 在/etc/sssd/sssd.conf 配置文件中对应 section 下添加 debug_level = 6 然后重启 sssd 服务,可以在/var/log/sssd/文件夹下看到对应的日志信息。

5) 用户家目录设置

(1) 使用 NFS。

详细步骤请参见前面的目录设置。

说明

服务器和用户端均需要开启 rpcbind 服务。

(2) 使用 oddjob。

只需要在用户端执行,目前仅支持用户端使用 sssd 的方式下使用。

① 安装 oddjob。

```
yum install -y oddjob
```

② 启动 oddjobd 服务。

```
systemctl start oddjobd
systemctl enable oddjobd
```

③ 选择 sssd 时使用 with-mkhomedir 功能。

```
mkdir -p /etc/authselect
authselect select sssd with-mkhomedir --force
```

依赖 oddjob 提供的/usr/lib64/security/pam_oddjob_mkhomedir.so。

④ 测试。

使用用户名 ldapuser,输入密码 mypassword123 登录。

```
pwd
```

显示 ldapuser1 的家目录为/home/rhome/ldapuser1。

6) LDAP 服务器端防火墙设置

(1) 防火墙增加对 LDAP 相关端口的放行。

```
firewall-cmd --permanent --add-service={ldap,ldaps}
```

执行如下命令，重启 firewalld 服务。

```
systemctl restart firewalld
```

(2) 防火墙增加对 NFS 相关端口的放行。

说明

该部分仅在使用 NFS 设置家目录时需要设置，并且仅需要在家目录服务器上设置。

```
firewall-cmd --permanent --add-port=20048/tcp
firewall-cmd --permanent --add-port=20048/udp
firewall-cmd --permanent --add-port=2049/tcp
firewall-cmd --permanent --add-port=2049/udp
```

执行如下命令，重启 firewalld 服务。

```
systemctl restart firewalld
```

7) 其他设置

(1) 设置 SSL/TLS。

OpenLDAP 用户端和服务器能够使用 Transport Layer Security(TLS)框架来提供完整性和机密性保护，并支持使用 SASL EXTERNAL 机制的 LDAP 身份验证。TLS 是 Secure Socket Layer(SSL)的标准名称。

① 自建 CA 中心。

- 执行如下命令，CA 中心生成自身私钥。

```
cd /etc/pki/CA
(umask 077; openssl genrsa -out private/cakey.pem 2048)
```

- 执行如下命令，CA 签发自身公钥。

```
openssl req -new -x509 -key private/cakey.pem -out cacert.pem -days 365
```

- 输入相关信息后，生成证书。

```
[root@lfbn-idf1-1-1427-165 CA]# openssl req -new -x509 -key private/cakey.pem -out
cacert.pem -days 365
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value, If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:CN
State or Province Name (full name) [Some-State]:ZheJiang
Locality Name (eg, city) []:Hangzhou Organization Name (eg, company) [Internet Widgits Pty
Ltd]:test.com
Organizational Unit Name (eg, section) []:ca
Common Name (e.g. server FQDN or YOUR name) []:caroot
Email Address []:ca@test.com
```

• 执行如下命令，查询根证书信息。

```
openssl x509 -noout -text -in /etc/pki/CA/cacert.pem
```

② 生成 LDAP 服务器证书。

• 执行如下命令，openLDAP 服务器端生成私钥。

```
cd /etc/openldap/certs
(umask 077; openssl genrsa -out ldapkey.pem 1024)
```

• 执行如下命令，openLDAP 服务器向 CA 申请证书签署请求。

```
openssl req -new -key ldapkey.pem -out ldap.csr -days 3650
```

• 输入相关信息后，生成证书。

```
[root@lfbn-idf1-1-1427-165 certs]# openssl req -new -key ldapkey.pem -out ldap.csr
-days 3650
Ignoring -days; not generating a certificate
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value, If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:CN
State or Province Name (full name) [Some-State]:ZheJiang
Locality Name (eg, city) []:Hangzhou Organization Name (eg, company) [Internet Widgits Pty
Ltd]:test.com
Organizational Unit Name (eg, section) []:ca
Common Name (e.g. server FQDN or YOUR name) []:ldapserver.test.com
Email Address []:ldapserver@test.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:.
An optional company name []:.
```

说明

除 Common Name 和 Email Address 外，其他信息必须和 CA 中心的证书所填信息保持一致，否则无法得到验证。

• 执行如下命令，CA 核实并签发证书。

```
openssl x509 -req -in ldap.csr -out ldapcert.pem -days 3650 -CA /etc/pki/CA/cacert.pem
-CAkey /etc/pki/CA/private/cakey.pem -CAcreateserial
```

须知

如果 CA 中心为独立的服务器，则需要将证书颁发请求文件 ldap.csr 传至 CA 中心服务器上，当 CA 中心服务器完成签发后，将 ldapcert.pem 传回 LDAP 服务器，且需要将 CA 证书 cacert.pem 文件从 CA 服务器复制到 LDAP 服务器的/etc/openldap/certs/目录下。

• 执行如下命令，修改证书权限。

```
cp /etc/pki/CA/cacert.pem ./
chown ldap:ldap cacert.pem ldapkey.pem ldapcert.pem
```

• 执行如下命令，进行查询。

```
ll /etc/openldap/certs
```

③ 服务器端设置证书信息。

• 执行如下命令，新建 certs.ldif。

```
vi certs.ldif
```

certs.ldif 内容如下。

```
dn: cn=config
add: olcTLSCACertificateFile
olcTLSCACertificateFile: /etc/openldap/certs/cacert.pem

dn: cn=config
add: olcTLSCertificateKeyFile
olcTLSCertificateKeyFile: /etc/openldap/certs/ldapkey.pem

dn: cn=config
add: olcTLSCertificateFile
olcTLSCertificateFile: /etc/openldap/certs/ldapcert.pem
```

• 执行如下命令，修改数据库中证书信息。

```
ldapmodify -Y EXTERNAL -H ldapi:/// -f certs.ldif
```

• 执行如下命令，确认 slapd 服务已开启加密端口。

```
cat /usr/lib/systemd/system/slapd.service
```

ExecStart=/usr/sbin/slapd -u ldap -h " ldap:/// ldaps:/// ldapi:///" 中包含 ldaps:///即为开启成功。

• 执行如下命令，重启 slapd 服务。

```
systemctl restart slapd
```

④ 使用加密通道进行信息查询。

• 执行如下命令，设置 CA 证书信息。

```
vi /etc/openldap/ldap.conf
```

配置项 TLS_CACERT /etc/openldap/certs/cacert.pem 指定了用于验证 LDAP 服务器证书的 CA(Certificate Authority)证书文件的路径。配置项 TLS_REQCERT hard 指示客户端在使用 TLS 连接时，要求对服务器证书进行严格的验证。

说明

如果是在用户端，需要先将 CA 证书复制到用户端/etc/openldap/certs/目录下。

```
scp root@90.90.112.64:/etc/openldap/certs/cacert.pem /etc/openldap/certs
```

- 执行如下命令，使用加密通道进行信息查询，能正常输出信息。

```
ldapsearch -x -H ldaps://ldapserver.test.com -b "dc=test,dc=com" uid=ldapuser1
```

⑤ 用户端设置证书信息并要求服务器端提供证书。

如果用户端没有 CA 证书，执行如下命令，先从服务器端将证书复制到用户端。

```
scp root@90.90.112.64:/etc/openldap/certs/cacert.pem /etc/openldap/certs
```

- 如果用户端使用 nss-pam-ldapd 搭建，则执行以下流程。

执行如下命令，在/etc/nslcd.conf 中添加 CA 证书信息。

```
vi /etc/nslcd.conf
```

增加如下内容。

```
#uri 修改为使用 ldaps
uri = ldaps://ldapserver.test.com

tls_cacertfile /etc/openldap/certs/cacert.pem
tls_reqcert hard
```

执行如下命令，重启 nslcd 服务。

```
systemctl restart nslcd
```

- 如果用户端使用 sssd 搭建，则执行以下流程。

执行如下命令，在/etc/sssd/sssd.conf 中添加证书信息。

```
vi /etc/sssd/sssd.conf
```

设置如下内容，如果已添加则忽略该步骤。

```
#ldap_uri 修改为使用 ldaps
ldap_uri = ldaps://ldapserver.test.com

ldap_tls_cacert = /etc/openldap/certs/cacert.pem
ldap_tls_reqcert = hard
```

执行如下命令，重启 sssd 服务。

```
systemctl restart sssd
```

(2) 使用 Kerberos5 认证。

详细请参见 LDAP 用户使用 krb5 认证下的服务器端设置和 LDAP 用户使用 krb5 认

证下的用户端设置。

(3) 开启日志功能。

① 执行如下命令,创建日志文件,调整日志文件权限,修改 rsyslog.conf。

```
mkdir -p /var/log/slapd
chown ldap:ldap /var/log/slapd
touch /var/log/slapd/slapd.log
chown ldap . /var/log/slapd/slapd.log
echo "local4.* /var/log/slapd/slapd.log" >> /etc/rsyslog.conf
```

② 执行如下命令,重启 rsyslog。

```
systemctl restart rsyslog
```

③ 执行如下命令,新建 log.ldif,修改数据库配置文件。

```
vi log.ldif
```

log.ldif 内容如下。

```
dn: cn=config
changetype: modify
add: olcLogLevel
olcLogLevel: 256
```

日志级别是可相加的,可用级别如下。

1(0x1 跟踪):跟踪函数调用。

2(0x2 数据包):调试数据包处理。

4(0x4 args):重跟踪调试(函数参数)。

8(0x8 conns):连接管理。

16(0x10 BER):打印发送的数据包和已收到的数据包。

32(0x20 filter):搜索过滤处理。

64(0x40 config):配置文件处理。

128(0x80 ACL):访问控制列表处理。

256(0x100 stats):状态日志:连接/操作/结果。

512(0x200 stats2):stats2 日志条目已发送。

1024(0x400 shell):打印与 Shell 的通信后端。

2048(0x800 解析):入口解析。

16384(0x4000 同步):LDAPSync 复制。

32768(0x8000 无):无论何种日志等级,仅记录 messages。

④ 执行如下命令,使数据库配置文件生效。

```
ldapmodify -Y EXTERNAL -H ldapi:/// -f log.ldif
```

4. Kerberos5 配置使用

1) 事先规划及设定

(1) Kerberos 域 realm 为:TEST.COM。

(2) 主 KDC：IP 为 90.90.114.169，hostname 为 kdc.test.com。
(3) 从 KDC：IP 为 90.90.113.82，hostname 为 kdcslave.test.com。
(4) krbclient：IP 为 90.90.115.201，hostname 为 client.test.com。
(5) LDAP 用户端：IP 为 90.90.112.65，hostname 为 ldapclient.test.com。
(6) LDAP 服务器：假设 IP 地址为 90.90.112.64，hostname 为 ldapserver.test.com。
(7) 所有主机开启 NTP 服务，假设当前使用的是 chrony。
根据实际配置 NTP 服务器：

```
vi /etc/chrony.conf
```

启动 chrony 服务：

```
systemctl start chronyd
systemctl enable chronyd
```

(8) hostname 设置。

```
hostnamectl set-hostname kdc.test.com
```

(9) 设置 hostname 与 IP 地址的对应关系。

```
vi /etc/hosts
```

每行添加 IP 地址及对应的 hostname。
2) KDC 服务器搭建
步骤 1 关闭防火墙。

```
systemctl stop firewalld
```

说明
如需开启防火墙，请参见 KDC 服务器端防火墙设置。
步骤 2 安装 krb5、krb5-libs、krb5-server、krb5-client。

```
yum install -y krb5 krb5-libs krb5-server krb5-client
```

步骤 3 设置 Kerberos 域。
执行如下命令，配置 krb5.conf 文件。

```
vi /etc/krb5.conf
```

修改 default_realm = TEST.COM。
修改[realms]中域名和其中的 kdc 与 admin_server 为 KDC 实际的 hostname。

```
[realms]
TEST.COM = {
kdc = kdc.test.com
admin_server = kdc.test.com
}
```

修改[domain_realm]中的.域名小写＝域名,域名小写＝域名。

```
[domain_realm]
.test.com = TEST.COM
test.com = TEST.COM
```

步骤 4　创建 KDC 数据库。

执行如下命令,查看数据库目录。

```
ll /var/kerberos/krb5kdc/
```

执行如下命令,创建数据库及设置数据库 Master password。

```
kdb5_util create -s
```

输入密码如 manager@123,再输入一次确认。

执行如下命令,查看数据库目录。

```
ll /var/kerberos/krb5kdc/
```

多了 principal、principal.kadm5、principal.kadm5.lock、principal.ok 等文件。

步骤 5　放行所有管理员权限。

```
vi /var/kerberos/krb5kdc/kadm5.acl
```

修改为 */admin@TEST.COM　　*。

步骤 6　注释掉使用 KCM 作为凭据缓存。

```
vi /etc/krb5.conf.d/kcm_default_ccache
```

注释掉最后两行,如下所示。

```
#[libdefaults]
#    default_ccache_name = KCM:
```

步骤 7　新建管理员 root/admin。

```
kadmin.local
```

输入 addprinc root/admin：新建管理员 root/admin,允许用户端使用 kadmin 使用 root/admin 账户密码登录。

输入 root/admin 的密码如 root@123,并再次输入确认。

输入 listprincs：查看是否存在新建的管理员信息。

输入 exit：退出。

步骤 8　执行如下命令,启动 kadmin、krb5kdc 服务。

```
systemctl start kadmin krb5kdc
systemctl enable kadmin krb5kdc
```

步骤 9 管理员登录测试。

```
kadmin
```

输入 root/admin 密码。

输入 listprincs。

输入 exit 退出。

步骤 10 新增 host 和用户 principal。

```
kadmin.local
```

输入 addprinc -randkey host/kdc.test.com：添加 host/kdc.test.com，密码为随机。

输入 addprinc krbuser1：添加一个普通账户 krbuser1。

输入 krbuser1 在 Kerberos 上的密码如 123，再次输入确认。

输入 exit 退出。

说明

想要管理 KDC 的数据库有两种方式，一种是直接在 KDC 上面直接执行，可以不需要密码就可以管理数据库；另一种则是需要输入密码才能管理。这两种方式分别如下。

- kadmin.local：需要在 KDC server 上面运行，不需要密码即可管理数据库。
- kadmin：可以在任何一台 KDC 领域的系统上面运行，但是需要输入管理员密码。

3）KDC 主从服务器搭建

（1）主 KDC 端设置。

步骤 1 Kerberos 域中增加从 KDC 主机内容。

```
vi /etc/krb5.conf
```

在[realms]中添加从 KDC 的 hostname，见加粗部分。

```
[realms]
TEST.COM = {
kdc = kdc.test.com
admin_server = kdc.test.com
kdc = kdcslave.test.com
}
```

说明

- admin_server 只有一个，一般是主 KDC。
- 这一步需要在 KDC 端步骤 1～步骤 4 完成后再从此步骤开始往下操作。

步骤 2 添加从 KDC 服务器 principal。

```
kadmin.local
```

输入“addprinc -randkey host/kdcslave.test.com”。

输入“exit”退出。

步骤 3　将主 KDC 的配置文件复制到从 KDC 上。

```
scp /etc/krb5.conf root@90.90.113.82:/etc
scp /var/kerberos/krb5kdc/kdc.conf root@90.90.113.82:/var/kerberos/krb5kdc/
scp /var/kerberos/krb5kdc/kadm5.acl root@90.90.113.82:/var/kerberos/krb5kdc/
scp /var/kerberos/krb5kdc/. k5. TEST. COM root @ 90. 90. 113. 82:/var/kerberos/krb5kdc/. k5.
TEST.COM
```

步骤 4　创建本机的票据资料。

```
kadmin.local
```

输入“ktadd host/kdc. test. com@TEST. COM:”添加票据到文件/etc/krb5. keytab 中。
输入“exit”退出。

步骤 5　同步数据库到从 KDC 上。

说明

这一步需要从 KDC 端完成步骤 7。

```
kdb5_util dump /var/kerberos/krb5kdc/kdc.dump
kprop - f /var/kerberos/krb5kdc/kdc.dump kdcslave.test.com
```

(2) 从 KDC 端设置。

步骤 1　关闭防火墙。

```
systemctl stop firewalld
```

说明

如需开启防火墙,请参见 KDC 服务器端防火墙设置进行配置。

步骤 2　安装 krb5、krb5-libs、krb5-server、krb5-client。

```
yum install - y krb5 krb5 - libs krb5 - server krb5 - client
```

步骤 3　创建 KDC 数据库。
执行如下命令,查看数据库目录。

```
ll /var/kerberos/krb5kdc/
```

创建数据库及设置数据库 Master password。

```
kdb5_util create - s
```

输入密码如 manager@123,再输入一次确认。
执行如下命令,查看数据库目录。

```
ll /var/kerberos/krb5kdc/
```

多了 principal、principal. kadm5、principal. kadm5. lock、principal. ok 等文件。

步骤 4 注释掉使用 KCM 作为凭据缓存。

```
vi /etc/krb5.conf.d/kcm_default_ccache
```

注释掉最后两行，如下所示。

```
#[libdefaults]
#     default_ccache_name = KCM:
```

步骤 5 创建本机的票据资料。

说明

这一步需要在主 KDC 端步骤 1～步骤 4 完成后再从此步骤开始往下操作。

```
kadmin
```

输入 root/admin 的密码。

输入“ktadd host/kdcslave.test.com@TEST.COM”，添加票据到文件/etc/krb5.keytab 中。

输入“exit”退出。

步骤 6 设置 kpropd 权限。

```
vi /var/kerberos/krb5kdc/kpropd.acl
```

添加如下内容。

```
host/kdc.test.com@TEST.COM
host/kdcslave.test.com@TEST.COM
```

步骤 7 执行如下命令，启动 kprop 服务。

```
systemctl start kprop
systemctl enable kprop
```

步骤 8 等主 KDC 完成步骤 5 同步数据库到从 KDC 上，执行如下命令，查看从 KDC 上数据库文件是否更新。

```
ll /var/kerberos/krb5kdc/
```

步骤 9 执行如下命令，启动 krb5kdc 服务。

```
systemctl start krb5kdc
systemctl enable krb5kdc
```

4）krb5 用户端搭建

步骤 1 安装 krb5、krb5-libs、krb5-client。

```
yum install -y krb5 krb5-libs krb5-client
```

步骤 2　设置 Kerberos 域，同 KDC 服务器端设置，也可以直接从 KDC 服务器端复制。

```
vi /etc/krb5.conf
```

修改 default_realm = TEST.COM。

修改[realms]中域名和其中的 kdc 和 admin_server 为 KDC 实际的 hostname。

```
[realms]
TEST.COM = {
kdc = kdc.test.com
admin_server = kdc.test.com
kdc = kdcslave.test.com
}
```

修改[domain_realm]中的.域名小写 = 域名，域名小写 = 域名。

```
[domain_realm]
.test.com = TEST.COM
test.com = TEST.COM
```

步骤 3　注释掉使用 KCM 作为凭据缓存。

```
vi /etc/krb5.conf.d/kcm_default_ccache
```

注释掉最后两行，如下所示。

```
#[libdefaults]
#    default_ccache_name = KCM:
```

步骤 4　创建本机的票据资料。

```
kadmin
```

输入 root/admin 的密码。

输入“addprinc -randkey host/client.test.com”，添加“host/client.test.com”，密码为随机。

输入“ktadd host/client.test.com@TEST.COM”，添加票据到文件/etc/krb5.keytab 中。

输入“exit”退出。

步骤 5　执行如下命令，查看票据信息。

```
klist -t -k
```

说明

service server 和 client 都是 KDC 服务器的用户端，都需要按上述步骤完成 KDC 服务器和票据资料的设置。

5. krb5 网络认证本地用户

使用 krb5 认证本地用户的步骤如下。

步骤 1　参见 krb5 用户端搭建，搭建一台 krb5 用户端。

步骤 2 执行如下命令，安装 pam_krb5。

```
yum install -y pam_krb5
```

步骤 3 修改账户认证方式。

```
vi  /etc/pam.d/system-auth
```

将所有 pam_sss.so 所在行最前面的“-”去掉，并将 pam_sss.so 修改为 pam_krb5.so。

```
vi/etc/pam.d/password-auth
```

将所有 pam_sss.so 所在行最前面的“-”去掉，并将 pam_sss.so 修改为 pam_krb5.so。

说明

如果 password-auth 不修改，ssh 登录会出现 Permission denied，原因是/etc/pam.d/sshd 中包含 password-auth。

步骤 4 本地创建一个用户 krbuser2。

```
useradd krbuser2 -d /home/krbuser2
```

步骤 5 在 krb 数据库中添加用户 krbuser2 及其密码。

```
kadmin
```

输入 root/admin 的密码“root@123”。

输入“addprinc krbuser2”。

输入密码如 123，再次输入确认。

输入“exit”。

步骤 6 登录测试。

输入用户名“krbuser2”，输入密码“123”可以登录。

```
ssh krbuser2@90.90.115.201
```

ssh 登录成功。

6. LDAP 用户使用 krb5 认证下的服务器端设置

步骤 1 参见 LDAP 服务器搭建，完成 LDAP 服务器端搭建。

步骤 2 参见 krb5 用户端搭建，完成在 LDAP 服务器上的 krb5 用户端搭建。

步骤 3 为 LDAP 服务创建 principal 和 keytab。

```
kadmin
```

如果默认管理员不是 root/admin，则使用如下命令。

```
kadmin -p root/admin@TEST.COM
```

输入 root/admin 的密码“123”。

输入“addprinc -randkey ldap/ldapserver.test.com”。

输入“ktadd ldap/ldapserver.test.com@TEST.COM”，添加密钥到文件/etc/krb5.keytab 中。

输入“exit”退出。

执行如下命令，修改/etc/krb5.keytab 文件的权限，使 LDAP 可以访问。

```
chown ldap:ldap /etc/krb5.keytab
```

步骤 4　执行如下命令，查看密钥信息。

```
klist -t -k
```

步骤 5　执行如下命令，重启 slapd 服务。

```
systemctl restart slapd
```

7. LDAP 用户使用 krb5 认证下的用户端设置

步骤 1　参见使用 sssd 搭建流程，完成配置。

需修改认证方式为 krb5。

```
vi /etc/sssd/sssd.conf
```

设置[sssd]下使用的 domains 为 test.com。

```
domains = test.com
```

设置[domain/test.com]为如下内容。

```
[domain/test.com]
id_provider = ldap
auth_provider = krb5
chpass_provider = krb5

ldap_uri = ldap://ldapserver.test.com
ldap_search_base = dc=test,dc=com
ldap_sasl_mech = GSSAPI

krb5_server = kdc.test.com
krb5_realm = TEST.COM

timeout = 300
krb5_auth_timeout = 300
```

如需要设置目录，使用 oddjob 完成设置。

执行如下命令，重启 sssd 服务。

```
systemctl restart sssd
```

步骤 2　参见 krb5 用户端搭建，完成配置。

步骤 3 设置 LDAP 用户的密码到 krb5 中，进行登录测试。

```
kadmin
```

输入 root/admin 的密码“root@123”。

输入“addprinc ldapuser1”，添加 ldapuser1 普通用户。

输入 ldapuser1 的密码如“123”。

再次输入密码。

输入“exit”退出。

(1) 本地登录测试。

输入用户名“ldapuser1”。

输入密码“123”。

登录成功。

(2) ssh 登录测试。

```
ssh ldapuser1@90.90.112.65
```

输入密码“123”。

登录成功。

说明

如果登录失败，可尝试修改/etc/sssd/sssd.conf 文件，将参数 krb5_server 修改为 KDC 服务器的 IP 地址，重启 sssd 服务。

8. KDC 服务器端防火墙设置

(1) 主 KDC 服务器端防火墙设置。

① 执行如下命令，防火墙增加对 Kerberos 相关端口的放行。

```
firewall-cmd --permanent --add-service=kerberos
```

② 执行如下命令，重启 firewalld 服务。

```
systemctl restart firewalld
```

(2) 从 KDC 服务器端防火墙设置。

① 防火墙增加对 Kerberos 相关端口的放行，需要额外放行 kprop 服务的端口 754/tcp。

```
firewall-cmd --permanent --add-service=kerberos
firewall-cmd --permanent --add-port=754/tcp
```

② 执行如下命令，重启 firewalld 服务。

```
systemctl restart firewalld
```

9. 常用操作

(1) 增加条目。

```
ldapadd -x -w 123 -D "cn=Manager,dc=test,dc=com" -f ldaprptest.ldif
```

说明

- -x 为进行简单认证。
- -w 123 -D "cn=Manager,dc=test,dc=com"为连接 slapd 的密码和 binddn。
- 也可以不通过命令行指定密码(即去掉-w 123),通过-w 选项来根据提示输入密码。

(2) 修改条目。

```
ldapmodify -x -w 123 -D "cn=Manager,dc=test,dc=com" -f ldaprptest.ldif
```

说明

此外,ldapmodify -a 选项表示新增条目,功能同上一条操作: ldapadd。

(3) 删除条目。

```
ldapdelete -x -w 123 -D "cn=Manager,dc=test,dc=com"
"uid=ldapuser2,ou=People,dc=test,dc=com"
```

说明

要删除的条目信息由"uid=ldapuser2,ou=People,dc=test,dc=com"指定。

(4) 查询条目。

```
ldapsearch -x cn=ldapuser2 -b dc=test,dc=com
```

说明

- -b 指定要查询的根节点。
- cn=ldapuser2 为要查询的内容。

(5) 修改密码。

执行如下命令,修改 uid=ldapuser2,ou=People,dc=test,dc=com 的密码为 456。

```
ldappasswd -x  -D "cn=Manager,dc=test,dc=com" -w 123
"uid=ldapuser2,ou=People,dc=test,dc=com" -s 456
```

说明

也可以通过 ldapmodify 结合 ldif 文件进行修改。

(6) 导出数据库内容。

```
slapcat -l export.ldif
```

说明

将数据导出到 export.ldif 文件中,服务器端命令。

(7) 调试模式启动 slapd。

```
slapd -d 256
```

10. 常见问题处理

1) NIS

(1) The local host's domain name hasn't been set. Please set it.

使用 nisdomainname 查询和设置。

(2) failed to send 'clear' to local ypserv: RPC: Program not registered.

rpcbind、ypserv 服务没有开启。

(3) gmake[1]: *** No rule to make target '/etc/netgroup', needed by 'netgroup'. Stop.

/etc/netgroup 不存在,新建该文件即可。

(4) yptest 报 ypbind failed 及 Internal NIS error。

检查 ypbind 服务是否开启。

2) LDAP

(1) ldap_add: Invalid syntax (21)。

additional info: objectClass: value #0 invalid per syntax

问题原因: ldif 文件格式存在问题,行尾不能有空格等。

(2) ldap_modify: Other (e.g., implementation specific) error (80)。

使用 ldapmodify -Y EXTERNAL -H ldapi:/// -f certs.ldif 来修改证书信息时报错,检查证书文件的所属是否是 LDAP。

(3) ldap_bind: Invalid credentials (49)。

管理员 DN 或者用户 DN 或者管理员密码错误。

(4) Result: No such object (32)。

条目不存在,检查 DN。

(5) Result: Strong(er) authentication required (8)。

需要使用管理员及管理员密码才能操作。常见于 ldapdelete、ldappasswd 等操作。

(6) ldap_sasl_interactive_bind_s: Can't contact LDAP server (-1)。

检查 slapd 服务是否开启。

命令行中可通过增加-d 1 打开调试信息来查看具体问题。

(7) 用户登录时报 Permission denied。

设置/etc/sssd/sssd.conf 中 access_provider=permit。

(8) id ldapuser1 报错 no such user。

检查 sssd.conf 文件中 ldap_uri 和 ldap_search_base 配置是否正确,域名是否能解析。

(9) ldap_modify: Other (e.g., implementation specific) error (80)。

常见于在修改 cn=config 中的证书文件时出现,与证书文件的顺序有关,可将多个修改分割为多个 ldif 文件进行修改,单独执行每个 ldif 文件进行修改。

3) Lerberos5

(1) kadmin: No KCM server found while opening default credentials cache

vi /etc/krb5.conf.d/kcm_default_ccache.

注释掉最后两行。

说明

开源 Kerberos 实现不含有 KCM server。

(2) kadmin: Cannot contact any KDC for realm 'TEST.COM' while initializing kadmin interface.

① 检查/etc/krb5.conf 中 Kerberos 域及 kdc 和 admin_server 是否配置正确。

② 检查/etc/hosts 中 hostname 和主机配置是否正确。

③ 检查服务器端 kadmin、krb5kdc 服务是否开启。

(3) kadmin: Communication failure with server while initializing kadmin interface.

① etc/krb5.conf 中 admin_server 只能有一个。

② 检查 admin_server 设置的域名是否正确,是否能解析,防火墙设置。

(4) get_principals: Operation requires "list" privilege while retrieving list.

检查/var/kerberos/krb5kdc/kadm5.acl 中的权限设置。

(5) SASL [conn=1028] Failure: GSSAPI Error: Unspecified GSS failure. Minor code may provide more information (Permission denied).

如果 LDAP 用户端使用 id 命令查询不到账户信息,而 LDAP 服务器端 slapd 的状态又出现 SASL [conn=1028] Failure: GSSAPI Error: Unspecified GSS failure. Minor code may provide more information (Permission denied)信息,则检查 LDAP 服务器端的 keytab 文件的权限是否允许 LDAP 访问,keytab 文件是否存在 LDAP 服务的票据。

3.2.12　FusionOS 鉴权 ESN 信息获取

1. 特性描述

1) 背景

鉴权 ESN 服务生成唯一性 ID,是获取 FusionOS 技术支持的重要凭据。

2) 定义

FusionOS 的商用版本需收集 ESN(Equipment Serial Number,设备序列号),用于支撑技术支持服务鉴权。FusionOS 使用过程中,如需要通过 400 电话或电子邮箱联系超聚变技术支持,请提供收集到的序列号。

第一次部署后,请将收集到的 ESN 信息反馈给超聚变接口人,作为商业技术支持鉴权依据。

3) 约束与限制

安装环境在选择环境语言时,须选择英文,否则会影响后续数据的收集。

2. 安装

FusionOS 系统安装时自动安装,或通过 rpm 包手动安装。

3. 配置使用

1）使用场景一：通过执行脚本逐台主机生成 License 文件

说明

- 适用少量节点，环境变量 p 和 f 无冲突场景。
- 如果使用 ARM 镜像，则只须执行步骤 2 获取 License 文件即可。

步骤 1 每台主机独立执行脚本创建 License 文件。使用 root 登录，并执行如下脚本。

须知

脚本是整体命令行，请先将下方脚本复制到文本编辑器中，确认没有换行，再复制到 Shell 中执行。

```
p="/etc/FusionOS_Verify/";f="/etc/FusionOS_Verify/FusionOS-license";mkdir $p;chattr -ia $f;echo "type is `dmidecode -s system-manufacturer`">$f;echo "Socket_num is `lscpu|grep -i "socket(s)"|awk -F " " '{print $NF}'`">>$f;system_uuid=`dmidecode -s system-uuid`;system_uuid_trimmed=${system_uuid//-/};rootfs_uuid=`lsblk -o uuid,mountpoint -x mountpoint|grep '/$'`;uuid="${system_uuid_trimmed:12:6}${rootfs_uuid:0:4}${system_uuid_trimmed:0:6}";echo "UUID is $uuid">>$f
```

步骤 2 获取 License 文件内容，并复制进行反馈。文件路径为“/etc/FusionOS_Verify/FusionOS-license”。文件内容如下。

```
[root@FusionOS ~]# cat /etc/FusionOS_Verify/FusionOS-license
Type is QEMU
Socket_num is 1
UUID is cce613ce20c94a14
```

2）使用场景二：安装 FusionOS_Verify 软件包，通过服务逐台主机生成 License 文件

说明

- 适用少量节点，可能存在环境变量冲突场景。
- 如果使用任意版本的 ARM 镜像或 22.0.4 版本的 x86 镜像，则无须安装 FusionOS_Verify 软件包，只须执行步骤 3 获取 License 文件即可。

步骤 1 手动安装 FusionOS_Verify 软件包。详细的安装步骤参见安装 FusionOS_Verify 软件包。

步骤 2 使用 root 账号登录，执行 systemctl start fusionos-verify.service 命令，生成 /etc/FusionOS_Verify/FusionOS-license 文件。

步骤 3 获取 License 文件内容，并复制进行反馈。文件路径为“/etc/FusionOS_Verify/FusionOS-license”。文件内容如下。

```
[root@FusionOS ~]# cat /etc/FusionOS_Verify/FusionOS-license
Type is QEMU
Socket_num is 1
UUID is cce613ce20c94a14
```

3）使用场景三：安装 FusionOS_Verify 软件包，生成 License 文件，并批量收集。

说明

- 适用节点数较多场景。
- 如果使用任意版本的 ARM 镜像或 22.0.4 版本的 x86 镜像，则无须安装 FusionOS_Verify 软件包，从步骤 2 开始执行即可。

步骤 1　手动逐台主机安装 FusionOS_Verify 软件包。详细的安装步骤参见安装 FusionOS_Verify 软件包。

步骤 2　选择一台主机作为收集机，安装 expect、dos2unix、sqlite 软件包。详细的安装步骤参见安装 expect、dos2unix、sqlite。可通过如下命令查看是否已经安装这三个包，如果包已安装则此步忽略。

```
rpm -qa | grep expect
rpm -qa | grep dos2unix
rpm -qa | grep sqlite
```

步骤 3　在收集机上创建配置文件，记录其他被收集主机的登录信息，用于批量登录收集。文件名称及路径“/root/log”。详细格式如下。

```
[root@localhost ~]# vim /root/log
node_ip         ssh_user   ssh_pass
90.90.114.247   root       fusion@123   #节点的 IP 信息,连接节点的用户,连接节点用户的密码
90.90.115.197   root       fusion@123
```

说明

操作完成后，请删除此配置文件，避免信息安全问题。

步骤 4　在收集机上执行命令 **fusionos_idcollect.sh --collect /root/log**，命令执行成功，其他 *N* 台机器的 ESN 信息被保存在/root/all_license_info.csv 文件中。/root/all_license_info.csv 文件内容如下。

```
[root@FusionOS ~]# cat /root/all_license_info.csv
date,Node_ip,UUID,Env_type,Socket_num
"2022-06-07 10:16:28
",,,,
,90.90.114.247,2fd75724-990f-4d05-92a6-a810ce641fbe,QEMU,4
,90.90.113.2,f5c5278e-f7ff-4663-93bb-705b09922fc1,QEMU,1
```

步骤 5　复制 all_license_info.csv 文件，并反馈给超聚变接口人。

4. 常用操作

1）安装 FusionOS_Verify 软件包

说明

任意版本的 ARM 镜像、FusionOS 22.0.4 版本的 x86 镜像已集成 FusionOS_Verify 软件包，可忽略安装 FusionOS_Verify 软件包。

步骤 1 x86 下载 FusionOS 发布的 FusionOS-22_22.0.1.SPC1_everything_x86-64.tar.gz 交付件，并将交付件上传到 Linux 环境的/root/目录下。

步骤 2 解压压缩包。

```
cd /root/
tar -xof /root/FusionOS-22_22.0.1.SPC1_everything_x86-64.tar.gz
```

步骤 3 创建本地 repo 源。

```
cd /root/
createrepo_c FusionOS-22_22.0.1.SPC1_everything_x86-64/
```

步骤 4 编辑/etc/yum.repos.d/FusionOS.repo 文件并添加以下内容。

```
vim /etc/yum.repos.d/FusionOS.repo
[new]
name=new
baseurl=file:///root/FusionOS-22_22.0.1.SPC1_everything_x86-64
enabled=1
gpgcheck=0
```

步骤 5 安装 FusionOS_Verify。

```
yum install FusionOS_Verify -y
```

2）安装 expect、dos2unix、sqlite

步骤 1 把 FusionOS 22 22.0.1 的 everything iso 上传到 Linux 环境的/root/目录下。

步骤 2 创建挂载点/root/iso 并挂载 ISO，示例如下。

```
mkdir /root/iso
mount /root/FusionOS-22_22.0.1_everything_x86-64.iso /root/iso
```

步骤 3 编辑/etc/yum.repos.d/FusionOS.repo 文件并添加以下内容。

```
vim /etc/yum.repos.d/FusionOS.repo
[all]
name=all
baseurl=file:///root/iso
enabled=1
gpgcheck=0
```

步骤 4 安装 expect、dos2unix、sqlite(FusionOS 默认已安装 sqlite)。

```
yum install expect dos2unix sqlite -y
```

步骤 5 安装完成之后请执行下方命令卸载挂载点。

```
umount /root/iso
```

3.3　安全加固工具

3.3.1　加固操作

1. 概述

安全加固工具会根据 usr-security.conf 设置加固策略，使用加固工具设置加固策略需要用户修改 usr-security.conf。本节介绍 usr-security.conf 的修改规则。用户可配置的加固项请参见 3.2 节对应内容。

2. 注意事项

(1) 修改配置后，需要重启安全加固服务使配置生效。重启方法请参见 3.3.2 节对应内容。

(2) 用户修改加固配置时，仅修改/etc/openEuler_security/usr-security.conf 文件，不建议修改/etc/openEuler_security/security.conf。security.conf 中为基本加固项，仅运行一次。

(3) 当重启安全加固服务使配置生效后，在 usr-security.conf 中删除对应加固项并重启安全加固服务并不能清除之前的配置。

(4) 安全加固操作记录在日志文件/var/log/openEuler-security.log 中。

3. 配置格式

usr-security.conf 中的每一行代表一项配置，根据配置内容的不同有不同配置格式，这里给出各类配置的格式说明。

说明

- 所有配置项以执行 ID 开头，执行 ID 仅为了方便用户识别配置内容，取值为正整数，由用户自行定义。
- 配置项的各内容之间使用@作为分隔符。
- 若实际配置内容中包含@，需要使用@@表示以和分隔符区分，例如，实际内容为 xxx@yyy，则配置为 xxx@@yyy。目前不支持@位于配置内容的开头和结尾。

1) d：注释

格式：执行 ID@d@对象文件@匹配项

功能：将对象文件中以匹配项开头(行首可以有空格)的行注释(在行首添加#)。

示例：执行 ID 为 401，注释/etc/sudoers 文件中以%wheel 开头的行。

```
401@d@/etc/sudoers@ % wheel
```

2) m：替换

格式：执行 ID@m@对象文件@匹配项@替换目标值

功能：将对象文件中以匹配项开头(行首可以有空格)的行替换为“匹配项加替换目标值”。若匹配行开头有空格，替换后将删除这些空格。

示例：执行 ID 为 101，将/etc/ssh/sshd_config 文件中以 Protocol 开头的行替换为 Protocol 2。匹配和替换时也会考虑 Protocol 后的空格。

```
101@m@/etc/ssh/sshd_config@Protocol @2
```

3) sm：精确修改

格式：执行 ID@sm@对象文件@匹配项@替换目标值

功能：将对象文件中以匹配项开头(行首可以有空格)的行替换为“匹配项加替换目标值”。若匹配行开头有空格，替换后将保留这些空格，这是 sm 和 m 的区别。

示例：执行 ID 为 201，将/etc/audit/hzqtest 文件中以 size 开头的行替换为 size 2048。

```
201@sm@/etc/audit/hzqtest@size@ 2048
```

4) M：修改子项

格式：执行 ID@M@对象文件@匹配项@匹配子项[@匹配子项的值]

功能：匹配对象文件中以匹配项开头(行首可以有空格)的行，并将该行中以匹配子项开始的内容替换为“匹配子项和匹配子项的值”，其中，匹配子项的值可选。

示例：执行 ID 为 101，找到 file 文件中以 key 开头的行，并将这些行中以 key2 开始的内容替换为 key2value2。

```
101@M@file@key@key2@value2
```

5) systemctl：管理服务

格式：执行 ID@systemctl@对象服务@具体操作

功能：使用 systemctl 管理对象服务，具体操作可取值为 start、stop、restart、disable 等 systemctl 所有可用的命令。

示例：执行 ID 为 218，停止 cups.service 服务，等同于 systemctl stop cups.service 的配置行。

```
218@systemctl@cups.service@stop
```

6) 其他命令

格式：执行 ID@命令@对象文件

功能：执行对应命令，即执行命令行“命令 对象文件”。

示例一：执行 ID 为 402，使用 rm-f 命令删除文件/etc/pki/ca-trust/extracted/pem/email-ca-bundle.pem。

```
402@rm -f @/etc/pki/ca-trust/extracted/pem/email-ca-bundle.pem
```

示例二：执行 ID 为 215，使用 touch 命令创建文件/etc/cron.allow。

```
215@touch @/etc/cron.allow
```

示例三：执行 ID 为 214，使用 chown 命令将文件/etc/at.allow 的属主改为 root:root。

```
214@chown root:root @/etc/at.allow
```

示例四：执行 ID 为 214，使用 chmod 命令去除文件/etc/at.allow 属主所在群组及其他非属主用户的 rwx 权限。

```
214@chmod og-rwx @/etc/at.allow
```

3.3.2　加固生效

完成修改 usr-security.conf 文件后，请运行如下命令使新添加的配置生效。

```
systemctl restart openEuler-security.service
```

小结

本章深入探讨了操作系统的加固概述。通过介绍加固方案以及加固影响，读者可以更好地理解在 FusionOS 中如何实施安全加固措施，以保障系统的安全性和可靠性。同时，本章提供了加固指导，详细说明了各项安全加固的内容、实现方法和影响，为用户提供了实际操作的指引。最后，介绍了 FusionOS 中的安全加固工具，这些工具可以帮助用户根据配置对系统进行加固，以提升系统的安全性。通过本章的内容，用户可以了解如何使用安全加固工具进行系统加固，并在加固配置后使其生效，从而提高系统的整体安全性。

习题

1. 为什么需要对操作系统进行安全加固？加固的目的是什么？
2. FusionOS 的安全加固方案包括哪些方面？
3. FusionOS 的安全加固工具是如何运行的？用户可以如何通过工具进行加固？
4. FusionOS 的安全加固内容分为哪 5 部分？
5. 安全加固对文件权限和账号口令的修改可能会造成什么影响？
6. FusionOS 默认是否设置了口令复杂度限制？如何限制登录失败时的尝试次数以及账户锁定时间？如何设置口令的加密算法？如何设置口令的有效期？
7. 什么是 FusionOS 中加固 su 命令的目的和实现方法？
8. 如何设置密码到期时禁用账户，并在 FusionOS 中将默认设置从 35 天修改为 30 天？
9. 如何修改 TMOUT 配置以在用户输入空闲一段时间后自动断开连接，防止无人看管的终端被攻击？
10. 如何设置网络远程登录的警告信息，以及隐藏系统架构和其他系统信息，避免目标性攻击？
11. 如何禁止通过 Ctrl+Alt+Del 组合键重启系统？

12. 如何设置 GRUB2 的加密口令以增强系统启动的安全性?

13. 如何进入安全的单用户模式?

14. 如何禁止交互式引导,以增强系统的安全性?

15. 如何加固 SSH 服务以提高系统安全性?

16. 如何设置时间同步并使用 chrony 来同步系统时钟?

17. Linux 中文件和目录的安全性主要通过什么来保证? 请以/bin 目录为例,说明如何修改文件的权限。请以/bin 目录为例,说明如何修改文件的属主。

18. 如何查找用户 ID 不存在的文件并删除? 如何查找群组 ID 不存在的文件并删除?

19. 为什么建议删除无指向的空链接文件? 如何查找系统中的空链接文件并进行处理?

20. 什么是 umask 值? 为什么建议为守护进程设置 umask 值? 如何设置守护进程的 umask 值为 0027?

21. 什么是粘滞位属性? 为什么要为全局可写目录添加粘滞位属性?

22. 什么是 Capability 权能机制的主要思想? Capability 权能机制如何实现权限检查?

23. 进程的 Capabilities 分为哪 4 个属性集? 它们分别是什么? 文件的 Capabilities 分为哪三组能力集? 分别是什么?

24. 什么是 SUID 和 SGID 权限? 它们如何影响文件和进程的执行? 如何检查系统中的 SUID 和 SGID 可执行文件?

25. 为什么要删除隐藏的可执行文件? 如何找到隐藏的可执行文件并删除?

26. 什么是内核参数? 如何通过内核参数提高操作系统的安全特性?

27. 如何加固内核参数以提高系统安全性?

28. 什么是 SELinux? 它是如何实现强制访问控制的? SELinux 有哪些运行模式? 它们分别是什么意义? 如何查看当前系统的 SELinux 运行状态?

29. 如何将 SELinux 设置为 enforcing 模式? 如何将其设置为 permissive 模式? 如果要关闭 SELinux,应该如何操作? 如何查询系统的 SELinux 状态和当前模式?

30. 如何通过 dnf 升级方式更新 selinux-policy 为最新版本?

31. 在 SELinux 关闭的情况下,如果系统无法启动,应该如何解决?

32. 什么是 TCP SYN Attack? 为了防止这种攻击,应该采取什么加固策略?

33. 什么是 kernel.dmesg_restrict 参数? 它的作用是什么?

34. 什么是魔术键(Magic SysRq Key)? 为什么建议禁用它?

35. 什么是 SELinux 的强制访问控制(MAC)? 它与 DAC(Discretionary Access Control)有何不同?

36. 在关闭 SELinux 的前提下,如果要将 SELinux 运行状态切换为 permissive 模式,应该执行哪些步骤?

37. 为什么详细的日志信息对于回溯历史操作和问题定位很重要?

38. 如何将 rsyslog 配置为将日志发送到远程日志主机?

39. 如何设置仅在指定的日志主机上接收远程 rsyslog 消息?

40. 默认情况下,rsyslog 是否记录守护进程的 debug 级日志信息? 为什么要配置 daemon.debug 选项?

41. 如何配置 rsyslog 以记录 kern. * 日志信息？

42. 在防 DoS 攻击中，什么是拒绝服务攻击（DoS）？什么是分布的拒绝服务攻击（DDoS）？

43. FusionOS 如何防止 DoS 攻击？

44. FusionOS 的防火墙是基于什么构建的？防火墙在网络中的作用是什么？

45. 如何查看防火墙的状态？如何启动和关闭防火墙？

46. 什么是 TCP-SYN cookie 保护？它对什么类型的攻击有帮助？

47. 如何在 FusionOS 中加固全局使用的目录？

48. 如何为分区设置安全相关的挂载选项？

49. 如何安装 AIDE（开源入侵检测工具）？

50. 什么是 NIS 和 LDAP？它们在网络中的作用是什么？

51. Kerberos 是什么？它如何在网络环境中进行身份认证？

52. 在 NIS 服务器搭建过程中，需要进行哪些步骤？

53. 如何在服务器端设置家目录，并使用 NFS 和 autofs 进行自动挂载？

54. 如何安装和配置 OpenLDAP 服务器？如何设置 SSL/TLS 来保护 OpenLDAP 用户端和服务器之间的通信？

55. 如何配置 LDAP 服务器的访问控制以及防火墙设置？

56. 什么是 LDAP 复制目录？如何在提供者和消费者之间配置同步？

57. 如何在 Linux 客户端配置访问 LDAP 服务器？

58. 请解释 nss-pam-ldapd 和 sssd 之间的区别是什么。请描述使用 nss-pam-ldapd 搭建 LDAP 用户端的步骤。

59. 请描述使用 sssd 搭建 LDAP 用户端的步骤。

60. 请描述 KDC 服务器的搭建步骤。

61. 请解释 kadmin. local 和 kadmin 之间的区别。

62. 如何为 Kerberos Realm 选择合适的命名方案？

63. 请描述 LDAP 用户使用 krb5 认证下的服务器端设置的步骤。

64. 请描述如何将 LDAP 用户的密码设置到 krb5 中并进行登录测试。

65. 如何在 KDC 服务器端设置防火墙？

66. 请解释常用操作中的 LDAP 增加、修改、删除、查询条目和修改密码的命令是什么。

67. 如何以调试模式启动 slapd 服务？

68. 安全启动是什么？它的第一阶段和第二阶段分别涵盖了哪些组件的验证和验证方式？

69. 安全启动技术的目的和受益是什么？

70. FusionOS 鉴权 ESN 信息是什么？它的主要作用是什么？

71. FusionOS 鉴权 ESN 信息获取可以通过哪些不同的场景进行？

72. 安装 FusionOS_Verify 软件包的步骤是什么？

73. 安装 expect、dos2unix、sqlite 的步骤是什么？

74. FusionOS 鉴权 ESN 信息收集的结果保存在哪个文件中？请提供该文件的示例内容。

75. FusionOS 鉴权 ESN 信息收集的脚本 fusionos_idcollect.sh 如何使用?

76. 安全加固工具是如何根据 usr-security.conf 设置加固策略的? 何时需要重启安全加固服务?

77. usr-security.conf 的配置格式有哪些? 当修改了 usr-security.conf 配置后,为什么需要重启安全加固服务? usr-security.conf 中的配置项的执行 ID 起什么作用? 如何使用执行 ID?