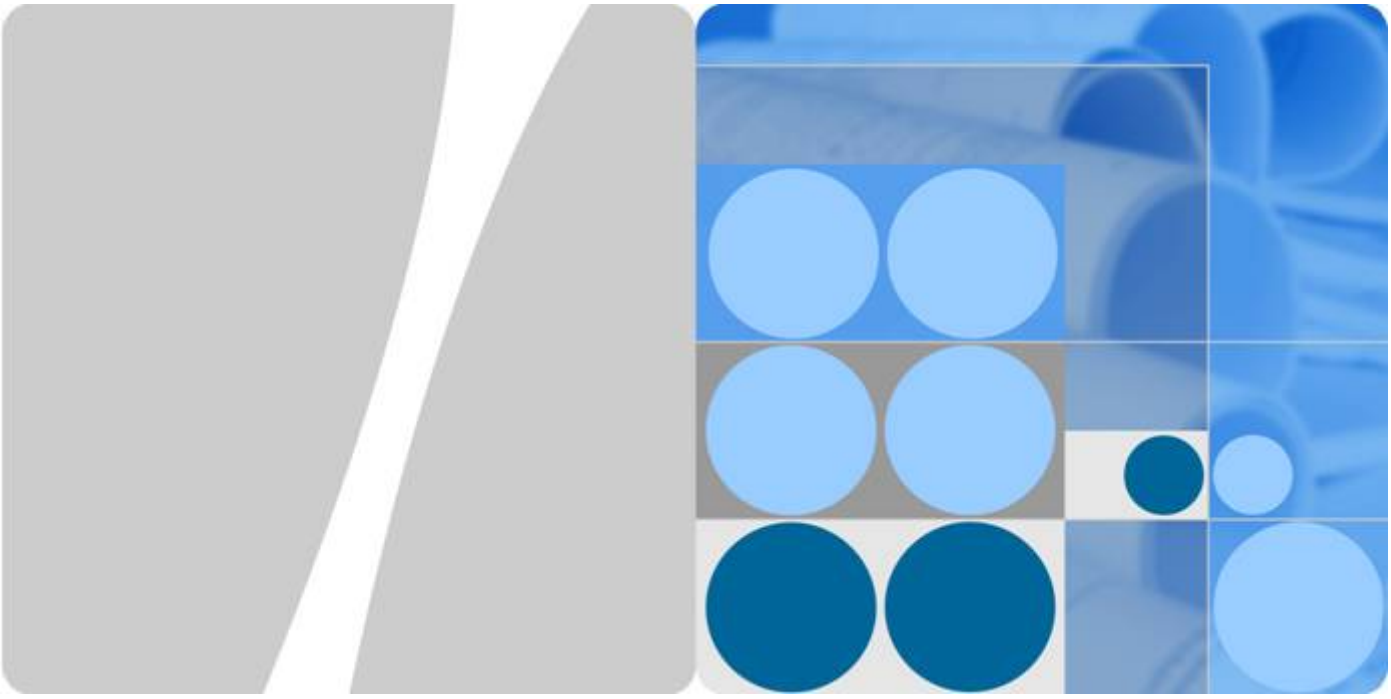


华为欧拉服务器操作系统软件 V2.0 安全说明手册

文档版本　01

发布日期　2019-08-12

华为技术有限公司

商标声明


和其他华为商标均为华为技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编：518129

网址： http://www.huawei.com

客户服务邮箱： support@huawei.com

客户服务电话： 4008302118

目 录

1 安全技术白皮书

本手册介绍EulerOS系统的安全架构和安全技术，主要从管理层、系统层和网络层进行描述。

1.1 概述
介绍EulerOS面临的安全威胁、采用的安全策略和安全流程。

1.2 安全技术
本节主要描述EulerOS三个层级，即管理层、系统层和网络层所使用的安全技术。

1.3 客户价值
随着网络应用的蓬勃发展，信息系统面临的安全威胁越来越多，单靠一种简单的安全技术不能完全保障系统的安全。

1.4 缩略语表
对文档中出现的术语与缩略语进行解释说明。

1.1 概述

介绍EulerOS面临的安全威胁、采用的安全策略和安全流程。

1.1.1 面临的安全威胁

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，但是由于技术和管理制度上的缺陷，使操作系统面临着诸多安全威胁，如管理层安全威胁、系统层安全威胁、网络层安全威胁。

1.1.2 安全策略

EulerOS操作系统的安全策略可以从三个层面考虑：安全技术、流程管理、人员。

图 1-1 EulerOS 操作系统安全策略

- 安全技术

  主要指的是OS自身的安全，也是产品安全的重点。安全技术主要包括操作系统的安全层、传输通信的网络安全层以及在操作系统之上的应用安全层。
- 流程管理

  主要指的是安全管理规章制度和安全资料上的安全问题。
- 人员

  主要指的是人本身的意识层面和执行层面的安全问题。

1.2 安全技术

本节主要描述EulerOS三个层级，即管理层、系统层和网络层所使用的安全技术。

1.2.1 管理层安全

管理层主要描述如何通过管理制度确保人员正确并安全的使用产品，降低人为因素给产品带来的安全威胁。

组织和过程

确定组织的范围，明确组织中的成员，并确定成员使用产品的规范，规定产品的使用过程，对产品的使用信息进行记录，以便在产生安全威胁时进行定位和回溯。从流程和规范上确保用户的行为不存在安全风险，保障产品的安全性。

账户和权限管理

按照账户职责的不同对系统中的账户进行严格的区分，为每个账户分配适当的权限，确保能完成其职责范围内的工作，做到账户的权限不过大、不相互交叉，从而满足最小权限原则。

同时用户的职责要做到明确和互斥，用户只能获取到与其职责相匹配的账户，账户不能交叉使用。

日志检查和稽核

日志检查和稽核是一项非常重要的工作，通过该项工作可以检查到系统是否遭受攻击。对日志进行分类，主要关注系统登录登出、用户删除创建、关键目录存取、关键文件权限变革、特权操作等事件。

稽核的事件可分为两类：成功的事件与失败事件。成功的事件是指使用者已经成功进行操作，而失败的事件则是指使用者曾经试图操作但是失败了。失败的事件对于追踪试图攻击环境的行为很有帮助，然而要分析成功的事件要困难很多。虽然绝大部分成功的稽核事件都只是系统中一般的正常活动，但处心积虑的攻击者在成功控制系统后也会产生成功的事件，因此事件的模式和事件本身一样重要。例如，在连续不断失败之后获取的成功，就表示可能有人试图攻击而且最终得逞。

无论什么情况，都应该将稽核事件与其他相关的使用者的信息结合起来。例如，某用户对应的账户被锁定，这时就可以稽查该用户是否在锁定期间有非法登录的情况。

应该指派特定的用户进行日志检查和稽核工作，并设定只有该用户可以执行这项操作，确保日志的安全；同时还要保证管理员每天进行一次日志检查和稽核操作，并给出检查和稽核结果。

补丁管理

对补丁进行分类管理，记录补丁的使用过程，确保任何补丁的升级都有日志记录，以便于后期检查。

系统备份

系统备份是一项及其重要的工作，操作得当的话，它们是防范灾害的最后一道防线。即使主业务系统全部被摧毁，也可以在其他的计算机上通过正确生成和保护完好的备份进行恢复。

基本的备份方针有很多。但是，在设计和搭建备份环境时，需要注意以下几点：

1. 镜像不能代替备份。镜像可以防范硬盘存储器的失效，但是它对于已删除的或损坏的文件无能为力。如果文件在镜像中删除了，那么它就同时在镜像和原始位置消失了，因此它必须通过某些外部的方法才可以取回。最常用的（但不是唯一的）外部恢复方法是恢复存储在备份磁盘上的数据。
2. 定期测试可恢复性。如果您无法恢复备份数据，那么就在创建备份上浪费了大量的时间。定期测试恢复性并不要求测试所创建的每一个备份，但是必须定期检测每一个磁盘驱动器以便确保备份是可读的，并且必须进行随机抽样测试以便确保它们可以正确地读取和恢复。
3. 保持磁头清洁。弄脏的磁头可能让备份看起来似乎成功地完成了，然而事实上磁带上只是写入了无用信息。
4. 注意备份介质的平均故障时间（Mean Time Between Failure，MTBF）。如果制造商建议存储介质的有效使用寿命是1000次备份，那么就只使用它1000次，然后就不要它。
5. 重要的数据要作双重备份。为了更好地确保数据的寿命和安全，明智的做法是存储一个异地备份。

安全培训

定期对目标用户进行安全培训，培养用户的安全意识，建立安全使用产品的理念。此外，分析系统可能会面临的安全风险，指导用户对基本安全故障进行定位和修正。

1.2.2 系统层安全

系统层安全是EulerOS安全的核心，主要包括操作系统层面采用的各种安全技术。

1.2.2.1 系统层安全架构

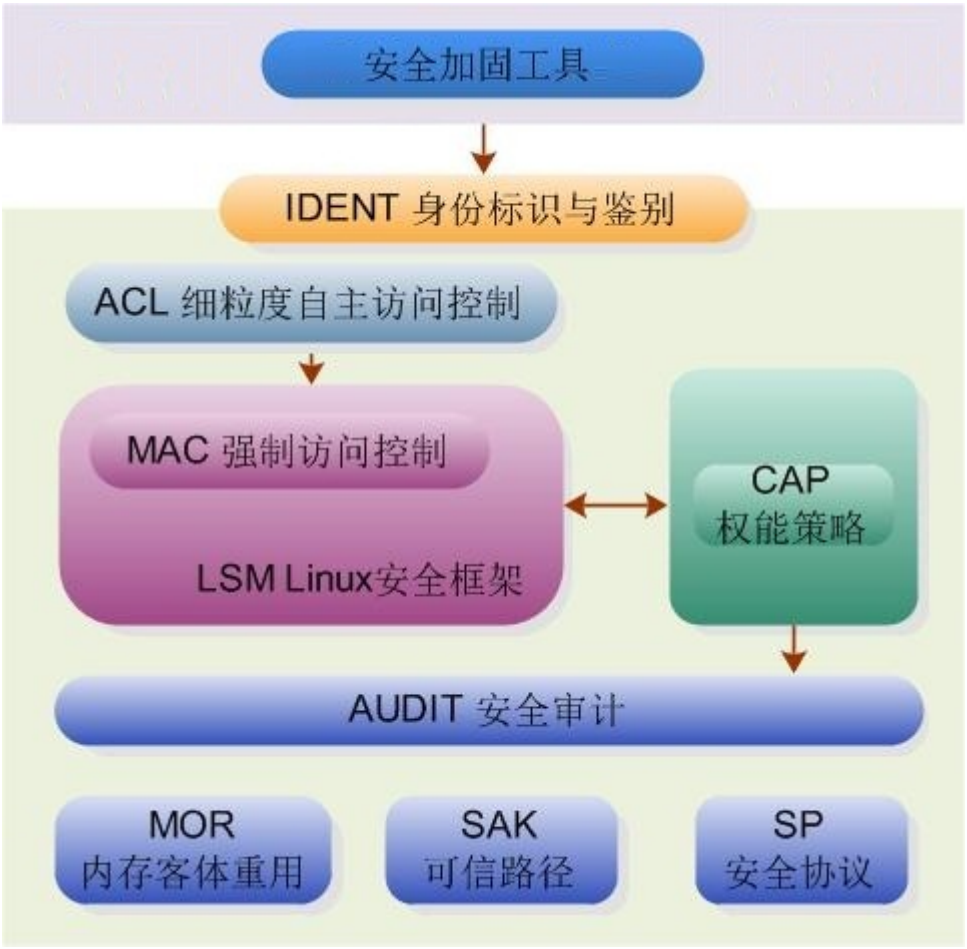
操作系统作为信息系统的核心，为网络服务、数据库系统等上层应用的正常运行提供了基本保障。

然而仅依赖应用空间的安全机制，无法从根本上解决信息系统的安全问题。没有操作系统安全机制的保障，应用空间的安全机制容易遭受破坏、旁路和欺骗攻击。上层应用的安全机制，诸如访问控制和加密等必须依赖操作系统安全机制的支持，才能实现其安全功能。

EulerOS操作系统提供身份标识与鉴别、安全协议、细粒度访问控制、强制访问控制、文件完整性检查、安全审计、内存客体重用、可信路径等安全机制，保障操作系统的安全性，为各类上层应用提供安全基础。

EulerOS操作系统的安全架构如图1-2所示。

图 1-2 EulerOS 操作系统安全架构



相关说明如下：

- 安全加固工具，提供方便的安全配置与管理，实现对系统服务、文件权限、内核参数、日志审计、账户口令等的安全加固。
- IDENT完成用户的身份标识与鉴别；可信路径提供安全注意键，启动可信的登录流程。
- ACL通过访问控制列表实现细粒度的自主访问控制。
- LSM（Linux Security Module）内核安全框架
- CAP基于LSM框架实现MAC强制访问控制。
- AUDIT负责进行安全审计。
- MOR禁止内存客体重用。
- SP为系统集成的安全协议。

1.2.2.2 身份标识与鉴别

EulerOS操作系统通过PAM机制来实现用户的身份标识与鉴别。

可插入式验证模块PAM（Pluggable Authentication Module），是SUN公司最早提出和开发的一套为系统登录应用程序提供验证和相关的安全服务的套件。主要功能包括认证管理、账户管理、会话管理和口令管理。EulerOS操作系统默认使用Linux-PAM。

1.2.2.3 安全协议

安全协议，通常也被称作口令协议，它是以口令学为基础的消息交换协议，其目的是在网络环境中提供各种安全服务。口令学是网络安全的基础，但网络安全不能单纯依靠安全的口令算法。安全协议是网络安全的一个重要组成部分，需要通过安全协议进行实体之间的认证、在实体之间安全地分配密钥或其他各种秘密、确认发送和接收的消息的非否认性等。

安全目标是多种多样的。例如，认证协议的目标是认证参加协议的实体的身份。此外，许多认证协议还有一个附加的目标，即在主体之间安全地分配密钥或其他各种秘密。

EulerOS操作系统支持的安全协议如下：

- SSH
- SSL
- IPSec
- SFTP

1.2.2.4 自主访问控制

访问控制，是指控制系统中主体对客体的访问权限。其中主体是指引起信息在客体间交换或者改变系统状态的主动实体，通常是发出访问请求的对象，例如进程；客体是指包含或接收数据的被动实体，是信息的载体，通常是被访问的对象，例如文件；而权限是指对客体进行特定模式访问的操作许可。

自主访问控制DAC（Discretionary Access Control）是操作系统必不可少的安全机制之一，在自主访问控制机制下，客体的属主承担着分派权限的任务，客体属主可以将其所拥有的权限任意组织并授予其他用户。例如：对某个文件A的属主UserA可以将它拥有的对文件A的读、写、执行权限授予其他用户，并且UserA可以在任何时间将这些权限收回。

EulerOS操作系统通过UGO和ACL机制实现自主访问控制。

1.2.2.5 强制访问控制

强制访问控制MAC（Mandatory Access Control）的基本思想是：每个主体、客体（文件、消息队列、共享区域、信号量）都赋予相应的安全属性（标记），该属性由管理员或系统按严格的规则设置，用户不能修改。

主体对客体的访问由强制安全控制机制按照某种安全策略，根据主、客体安全属性，确定是否允许访问。若系统判断不许访问，任何人（包括客体主）也不能访问。

1.2.2.6 内存客体重用

客体重用机制保证在主体活动结束后，主体占用的存储客体中的信息将不能被另一个主体使用。

在计算机信息系统可信计算机的空闲存储客体空间中，对客体初始指定、分配或再分配一个主体之前，撤消该客体所含信息的所有授权。当主体获得对一个已被释放的客体的访问权时，当前主体不能获得原主体活动所产生的任何信息。

EulerOS操作系统实现了内存的客体重用。

内存客体重用是用于防止新主体获得先前主体残留在内存中的信息。内存客体重用在分配内存时实施，它对内存进行覆写，以达到禁止重用目的。

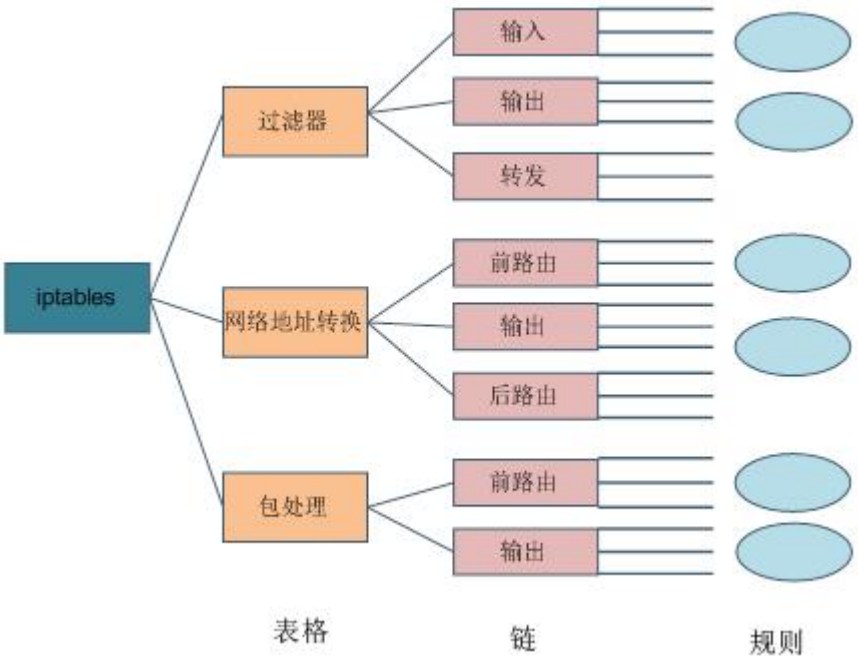
1.2.2.7 主机防火墙

EulerOS操作系统默认安装iptables防火墙。

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。它放置的位置必须是受保护网络与其他网络的唯一进出口点。如果有其他入口节点可以进入受保护网络，防火墙将毫无作用。

Linux内核内置了一套防火墙机制，称为Netfilter。而设置和控制Netfilter的工具是iptables。本系统就是使用iptables制定的防火墙规则。iptables的规则链组织结构如**图1-3**所示。

图 1-3 iptables 规则链组织结构



1.2.3 网络层安全

网络层安全就是通过采用一系列安全措施，使得网络系统得到应有的安全保护，为在该网络平台上运行的业务系统提供应用的支持，包括一切访问网络资源或使用网络服务相关的安全保护。

网络层安全技术包括网络拓朴安全设计、网络设备保护、网络隔离、网络边界保护（如防火墙）、网络安全检测（如IDS）、网络数据加密（如VPN）、网络安全扫描、网络安全管理和二层安全（如IP/MAC绑定和DHCP隔离）等多个方面。EulerOS在网络安全方面着重关注服务器内部网络安全问题，主要通过防火墙技术来保证服务器内部网络安全。iptables内容请参见主机防火墙。

1.3 客户价值

随着网络应用的蓬勃发展，信息系统面临的安全威胁越来越多，单靠一种简单的安全技术不能完全保障系统的安全。

EulerOS在保证产品可用性的同时，通过三个层次的安全技术，对产品进行全方位的保护，增强产品在管理层、系统层、网络层的安全性，降低产品遭受攻击几率，确保产品在整个生命周期中的安全，切实保障用户的利益，降低用户的安全风险。

管理层为用户提供产品使用及管理规范上的指导，确保用户正常、安全的使用产品，通过管理规范将人为因素对安全的影响降至最低。

系统层通过提供身份标识与鉴别、安全协议、细粒度访问控制、强制访问控制、文件完整性检查、安全审计、内存客体重用、可信路径、安全加固等安全机制，为上层应用提供安全支撑，确保产品的安全性，有效防止木马、病毒及网络攻击，有效的将客户的安全风险控制在合理的范围之内。此外，EulerOS定期对系统日志进行收集和备份，便于用户进行安全审计。

网络层通过安全组网，保障整个产品在网络中的安全。

1.4 缩略语表

对文档中出现的术语与缩略语进行解释说明。

本文出现的每个缩略语的英文全名和中文全名请参见表1-1。

表 1-1 缩略语清单

英文缩写	英文全称	中文全称
ACL	Access Control List	访问控制列表
CRC32	Cyclic Redundancy Check	循环冗余校验
DAC	Discretionary Access Control	自主访问控制
HTTPS	Hypertext Transfer Protocol over Secure Socket Layer	安全HTTP
IDS	Intrusion Detection Systems	入侵检测系统
IPSEC	Internet Protocol Security	Internet安全协议
LSM	Linux Security Module	Linux安全框架
MAC	Mandatory Access Control	强制访问控制
MD5	Message Digest Algorithm	消息摘要算法第五版
MOR	Memory Object Reuse	内存客体重用
MTBF	Mean Time Between Failure	平均故障时间
PAM	Pluggable Authentication Module	可插入验证模块
RAID	Redundant Array of Independent Disk	独立冗余磁盘阵列
SAK	Secure Attention Key	安全注意键
SHA	Secure Hash Algorithm	安全Hash算法
SSL	Secure Socket Layer	安全套接层
TLS	Transport Layer Security	传输层安全

2 安全加固操作指南

本指南描述EulerOS系统的安全加固策略和加固方法。

2.1 操作系统加固概述
介绍对EulerOS系统进行加固的目的、方案和影响。

2.2 默认加固策略
加固策略配置文件及加固脚本中已配置了默认的加固策略。本节点介绍各项默认策略的含义，默认加固策略不建议用户修改。

2.3 附录
介绍文件权限的含义和umask值的含义。

2.1 操作系统加固概述

介绍对EulerOS系统进行加固的目的、方案和影响。

2.1.1 加固目的

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

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

2.1.2 加固方案

本章描述EulerOS的安全加固方案，包括加固方式和加固内容。

- 加固方式

  EulerOS默认搭载系统安全加固包，在安装时生成安全加固服务，系统在首次启动时自动启动安全加固服务对系统进行整体加固。

- 加固内容

  EulerOS系统加固内容主要分为以下5个部分：

  - 系统服务

- 文件权限
- 内核参数
- 授权认证
- 帐号口令

2.1.3 加固影响

由于对文件权限、账户口令进行了安全加固，从而造成用户使用习惯上的变更，影响了系统的易用性，对易用性的影响如表2-1所示。

表 2-1 加固影响说明

加固项	加固描述	易用性影响
字符界面等待超时限制	当字符界面长时间处在空闲状态，字符界面会自动退出。 **说明** 当用户通过SSH登录，超时时间由/etc/profile文件的TMOUT字段和/etc/ssh/sshd_config文件的ClientAliveInterval字段两个值中较小的值决定，当前为300秒。	用户长时间不操作字符界面，字符界面会自动退出。
口令复杂度限制	口令长度最小为8位，口令至少包含大写字母、小写字母、数字和特殊字符中的3种。	系统中所有用户不能设置简单的口令，口令必须符合复杂度要求。
限定登录失败时的尝试次数	当用户登录系统时，口令连续输错3次，账户将被锁定300秒，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300秒。
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。
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。
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。
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，再切换至root账户。

加固项	加固描述	易用性影响
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。

2.2 默认加固策略

加固策略配置文件及加固脚本中已配置了默认的加固策略。本节点介绍各项默认策略的含义，默认加固策略不建议用户修改。

2.2.1 系统服务

2.2.1.1 加固 SSH 服务

说明

设置系统使用OpenSSH协议时的算法、认证等参数。

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。透过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，来提高系统的安全性。表2-2中详细说明各加固项的默认策略，并给出各加固项的取值范围。

说明

默认加固策略不建议用户修改。

实现

- 服务端加固策略

  当前EulerOS系统对SSH服务的加固策略如表2-2所示。表中所有加固项均在SSH的配置文件/etc/ssh/sshd_config中。

  表 2-2 SSH 服务默认加固策略说明

策略说明	加固项	加固默认值
设置使用SSH协议的版本	Protocol	2

策略说明	加固项	加固默认值
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	SyslogFacility	AUTH
设置记录sshd日志消息的层次	LogLevel	VERBOSE
设置使用SSH登录后，能否使用图形化界面	X11Forwarding	no
最大认证尝试次数	MaxAuthTries	3
设置是否允许公钥认证。	PubkeyAuthentication	yes
设置是否允许只有RSA安全验证	RSAAuthentication	yes
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	IgnoreRhosts	yes
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	RhostsRSAAuthentication	no
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	HostbasedAuthentication	no
禁止root账户直接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。	PermitRootLogin	no
设置是否允许用口令为空的帐号登录	PermitEmptyPasswords	no
设置是否解析 ~/.ssh/environment和~/.ssh/authorized_keys中设定的环境变量	PermitUserEnvironment	no
设置SSH数据传输的加密算法	Ciphers	aes128-ctr,aes192-ctr,aes256-ctr
设置系统等待的超时时间（单位秒）。超过指定时间未收到来自客户端的数据，则断开连接	ClientAliveInterval	300
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	ClientAliveCountMax	0
登录SSH前后显示的提示信息	Banner	/etc/issue.net

策略说明	加固项	加固默认值
设置SSH数据校验的哈希算法	MACs	hmac-sha2-256,hmac-sha2-512
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	StrictModes	yes
使用PAM登录认证	UsePAM	yes
设置是否允许TCP转发	AllowTcpForwarding	no
sftp日志记录级别，记录INFO级别以及认证日志。	Subsystem sftp /usr/libexec/openssh/sftp-server	-l INFO -f AUTH
设置是否允许SSH Agent转发	AllowAgentForwarding	no
设置是否允许连接到转发客户端端口	GatewayPorts	no
Tunnel设备是否允许使用	PermitTunnel	no
设置SSH密钥交换算法	KexAlgorithms	ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,diffie-hellman-group-exchange-sha256,diffie-hellman-group-exchange-sha1,diffie-hellman-group14-sha1

说明

/etc/issue.net默认为“Authorized users only. All activities may be monitored and reported.”

- 客户端加固策略

  当前EulerOS系统对SSH客户端的加固策略如**表2-3**所示。表中所有加固项均在SSH的配置文件“/etc/ssh/ssh_config”中。

表 2-3 SSH 客户端默认加固策略说明

策略说明	加固项	加固默认值
设置SSH密钥交换算法	KexAlgorithms	ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,diffie-hellman-group-exchange-sha256,diffie-hellman-group-exchange-sha1,diffie-hellman-group14-sha1
是否使用DNS或者SSHFP资源记录验证HostKey	VerifyHostKeyDNS	ask

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1536bits。

其他安全建议

- SSH服务仅监听指定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上进行绑定监听，而不是监听0.0.0.0，可修改/etc/ssh/sshd_config文件中的ListenAddress配置项。

  a. 修改/etc/ssh/sshd_config文件

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

  修改内容如下：

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

  说明

  此处以192.168.1.100为例，产品可根据实际情况配置。

  b. 重启SSH服务

  ```
  systemctl restart sshd.service
  ```

- 限制SFTP用户其向上跨目录访问

  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：

  a. 创建SFTP用户组

  ```
  groupadd sftpgroup
  ```

  b. 创建SFTP根目录

  ```
  mkdir /sftp
  ```

  c. 修改SFTP根目录属主和权限

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

  d. 创建SFTP用户

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

  e. 设置SFTP用户的口令

  ```
  passwd sftpuser
  ```

  f. 创建SFTP用户上传目录


```
mkdir /sftp/sftpuser
```

g. 修改SFTP用户上传目录属主和权限

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

h. 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 AUTHH
Subsystem sftp internal-sftp -l INFO -f AUTH
...
#End of sshd_config

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

说明

1. %u代表当前用户的用户名。

i. 重启SSH服务

```
systemctl restart sshd.service
```

- 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。

2.2.2 文件权限

2.2.2.1 设置文件的权限和属主

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是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。

该默认策略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。

实现

步骤1 修改文件权限。以/bin目录为例，默认策略将该路径权限设置为755。

```
chmod 755 /bin
```

步骤2 修改文件属主。

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

----结束

2.2.2.2 删除无主文件

说明

系统管理员在删除用户时，存在着忘记删除该用户所拥有的文件的问题。如果后续新创建的用户与被删除的用户同名，则新用户会拥有部分不属于其权限的文件，建议将此类文件删除。

实现

步骤1 通过命令“find / -nouser” 查找出用户ID不存在的文件，将此类文件删除。

步骤2 通过命令“find / -nogroup” 查找出群组ID不存在的文件，将此类文件删除。

----结束

2.2.2.3 处理空链接文件

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。

实现

步骤1 通过如下命令查找系统中的空链接文件。

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

说明

dir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 如果此类文件为无实际作用，可通过如下命令删除。

```
rm -f fileX
```

说明

fileX即为**步骤1**找出的文件名。

----结束

2.2.2.4 设置守护进程的 umask 值

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**2.3.2 umask值含义**。

实现

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

2.2.2.5 删除非授权文件的全局可写属性

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。

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(设置粘贴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：

```
chmod o-w 〈filename〉
```

----结束

2.2.2.6 限制 at 命令的使用权限

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。

实现

步骤1 删除/etc/at.deny文件。

步骤2 将at.allow的文件属主改为root:root。

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

步骤3 控制/etc/at.allow的文件权限，仅root可操作。

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

----结束

2.2.2.7 限制 cron 命令的使用权限

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。

实现

步骤1 删除/etc/cron.deny文件。

步骤2 将cron.allow的文件属主改为root:root。

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

步骤3 控制/etc/cron.allow的文件权限，仅root可操作。

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

----结束

2.2.2.8 为全局可写目录添加黏着位属性

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黏着位属性。

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黏着位属性。

```
chmod +t $dirname
```

----结束

2.2.3 内核参数

2.2.3.1 加固内核参数

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。比如：通过微调或配置网络选项，可有效提高系统的安全性。

实现

步骤1 将表2-4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 2-4 内核参数加固策略说明

策略说明	加固项	默认值
是否接受ICMP广播报文。加固策略为不接受。	net.ipv4.icmp_echo_ignore_broadcasts	1
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。	net.ipv4.conf.all.rp_filter	1
	net.ipv4.conf.default.rp_filter	1

策略说明	加固项	默认值
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。	net.ipv4.ip_forward	0
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。	net.ipv4.conf.all.accept_source_route	0
	net.ipv4.conf.default.accept_source_route	0
是否发送ICMP重定向报文。加固策略为禁止发送。	net.ipv4.conf.all.accept_redirects	0
	net.ipv4.conf.default.accept_redirects	0
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。	net.ipv4.conf.all.send_redirects	0
	net.ipv4.conf.default.send_redirects	0
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。	net.ipv4.icmp_ignore_bogus_error_responses	1
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。	net.ipv4.tcp_syncookies	1
加固dmesg信息，仅允许管理员查看。	kernel.dmesg_restrict	1
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。	net.ipv4.conf.all.secure_redirects	0
	net.ipv4.conf.default.secure_redirects	0

步骤2 加载sysctl.conf文件中设置的内核参数

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

----结束

其它安全建议

- net.ipv4.icmp_echo_ignore_all：忽略ICMP请求。

  出于安全考虑，建议开启此项（当前默认值为0，开启将值设为1）。

  但开启后会忽略所有接收到的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。

- net.ipv4.conf.all.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（当前默认值为0，开启将值设为1）。

  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。

- net.ipv4.tcp_timestamps：关闭tcp_timestamps。
  出于安全考虑，建议关闭tcp_timestamps（当前默认值为1，关闭将值设为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。

2.2.4 授权认证

2.2.4.1 设置网络远程登录的警告信息

说明

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

实现

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

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

2.2.4.2 禁止通过 CTRL+ALT+DEL 组合键重启系统

说明

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

实现

通过屏蔽内核keyboard中的Ctrl+Alt+Del组合键响应函数解决。

说明

如下文件保留的原因是XEN驱动需要调用，系统已无法响应Ctrl+Alt+Del组合键操作，因此无影响：

/etc/systemd/system/ctrl-alt-del.target

/usr/lib/systemd/system/ctrl-alt-del.target

2.2.4.3 设置终端的自动退出时间

说明

无人看管的终端容易被监听或被攻击，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。

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
```

2.2.4.4 设置用户的默认 umask 值为 077

说明

umask值用于为用户新创建的文件和目录设置缺省权限。如果umask的值设置过小，会使群组用户或其他用户的权限过大，给系统带来安全威胁。因此设置所有用户默认的umask值为0077，即用户创建的目录默认权限为700，文件的默认权限为600。umask值代表的是权限的“补码”，umask值和权限的换算方法请参见**2.3.2 umask值含义**。

实现

步骤1 分别在/etc/bashrc、/etc/profile.d/目录下所有文件中加入“umask 0077”。

```
echo “umask 0077” >> $FILE
```

说明

*$FILE*指上面所描述的文件名。

步骤2 设置步骤1中所列文件的属主为root，群组为root。

```
chown root.root $FILE
```

说明

*$FILE*指步骤1所描述的文件名。

----结束

2.2.4.5 设置 grub2 加密口令

说明

GRUB是GRand UnifiedBootloader的缩写，它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口令的情况下才能修改启动参数。

说明

grub2默认设置的口令为Huawei#12，强烈建议用户登录系统后第一时间修改grub2的口令，修改方法请参考**实现**。

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sha512。

```
[root@localhost ~]# grub2-mkpasswd-pbkdf2
Enter password:
Reenter password:
PBKDF2 hash of your password is
grub.pbkdf2.sha512.10000.D0649C25C42DA547B79AD683974DE105D0F0899C1EFA4AD34BE49E87C5A41C89F9C1E29A88
BBDC05584706C89F3FEB5A284A8A85738058A15F21A862A0464E34.CC0D536DF0418B75C34351B635BF209E05B02503B244
0C7FE290843287925E0221860080318EC5A8C8D796B29A2C73C6623B6EAAE04FF8840EE6FCEDA3C29C46
```

说明

在Enter password和Reenter password输入相同的口令；

*grub.pbkdf2.sha512.10000.D0649C25C42DA547B79AD683974DE105D0F0899C1EFA4AD34BE49E87C5A41C89F9C1E29A88BBDC05584706C89F3FEB5A284A8A85738058A15F21A862A0464E34.CC0D536DF0418B75C34351B635BF209E05B02503B2440C7FE290843287925E0221860080318EC5A8C8D796B29A2C73C6623B6EAAE04FF8840EE6FCEDA3C29C46*为Huawei#12经过grub2-mkpasswd-pbkdf2加密后的输出，每次输出的密文不同。

步骤2 向/etc/grub.d/00_header末尾追加如下字段：

```
cat <<EOF
set superusers="root"
password_pbkdf2 root
grub.pbkdf2.sha512.10000.D0649C25C42DA547B79AD683974DE105D0F0899C1EFA4AD34BE49E87C5A41C89F9C1E29A88BBDC05584706C89F3FEB5A284A8A85738058A15F21A862A0464E34.CC0D536DF0418B75C34351B635BF209E05B02503B2440C7FE290843287925E0221860080318EC5A8C8D796B29A2C73C6623B6EAAE04FF8840EE6FCEDA3C29C46
EOF
```

说明

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个为该账户的口令加密密文。

步骤3 执行grub2-mkconfig命令使上述修改生效。

```
grub2-mkconfig -o /boot/grub2/grub.cfg
```

说明

/boot/grub2/grub.cfg是grub2的配置文件。

----结束

2.2.5 账户口令

2.2.5.1 屏蔽系统账户

说明

除了用户账户外，其他帐号称为系统账户。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。

实现

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。

2.2.5.2 限制使用 su 命令的账户

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。

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：

```
#%PAM-1.0
auth         sufficient    pam_rootok.so
# Uncomment the following line to implicitly trust users in the wheel group.
#auth        sufficient    pam_wheel.so trust use_uid
# Uncomment the following line to require a user to be in the wheel group.
auth         required      pam_wheel.so use_uid
auth         substack      system-auth
auth         include       postlogin
account      sufficient    pam_succeed_if.so uid = 0 use_uid quiet
account      include       system-auth
password     include       system-auth
session      include       system-auth
session      include       postlogin
session      optional      pam_xauth.so
```

表 2-5 pam_wheel.so 配置项说明

配置项	说明
use_uid	基于当前账户的uid。

2.2.5.3 设置口令复杂度

说明

设置口令的复杂度的要求如下：

1. 口令长度至少8个字符；
2. 口令必须包含如下至少3种字符的组合：
   －至少一个小写字母；
   －至少一个大写字母；
   －至少一个数字；
   －至少一个特殊字符：`~!@#$%^&*()-_=+\|[{}];:'",<.>/?和空格
3. 口令不能和帐号一样；
4. 口令不能使用字典词汇。

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，配置如下：

```
#%PAM-1.0
# User changes will be destroyed the next time authconfig is run.
auth        required      pam_env.so
auth        required      pam_faillock.so preauth audit deny=3 even_deny_root unlock_time=300
auth        sufficient    pam_fprintd.so
auth        sufficient    pam_unix.so nullok try_first_pass
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
auth        requisite     pam_succeed_if.so uid >= 1000 quiet_success
auth        required      pam_deny.so
```



```
account     required      pam_unix.so
account     required      pam_faillock.so
account     sufficient    pam_localuser.so
account     sufficient    pam_succeed_if.so uid < 1000 quiet
account     required      pam_permit.so

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
password    sufficient    pam_unix.so sha512 shadow nullok try_first_pass use_authtok
password    required      pam_deny.so

session     optional      pam_keyinit.so revoke
session     required      pam_limits.so
-session     optional      pam_systemd.so
session     [success=1 default=ignore] pam_succeed_if.so service in crond quiet use_uid
session     required      pam_unix.so
```

表 2-6 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3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3次
enforce_for_root	本设置对root账户同样有效

表 2-7 pam_pwhistory.so 配置项说明

配置项	说明
remember=5	口令不能修改为过去5次使用过的旧口令
enforce_for_root	本设置对root账户同样有效

2.2.5.4 设置口令有效期

说明

出于系统安全的考虑，设置口令有效期限为90天，口令到期前7天通知用户更改口令。

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2-8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 2-8 login.defs 配置项说明所示

策略说明	加固项	加固默认值
口令最大有效期	PASS_MAX_DAYS	90
两次修改口令的最小间隔时间	PASS_MIN_DAYS	0
口令过期前开始提示天数	PASS_WARN_AGE	7

说明

login.defs是设置用户帐号限制的文件，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。

2.2.5.5 设置口令的加密算法

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。

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，配置如下：

```
#%PAM-1.0
# User changes will be destroyed the next time authconfig is run.
auth        required      pam_env.so
auth        required      pam_faillock.so preauth audit deny=3 even_deny_root unlock_time=300
auth        sufficient    pam_fprintd.so
auth        sufficient    pam_unix.so nullok try_first_pass
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
auth        requisite     pam_succeed_if.so uid >= 1000 quiet_success
auth        required      pam_deny.so

account     required      pam_unix.so
account     required      pam_faillock.so
account     sufficient    pam_localuser.so
account     sufficient    pam_succeed_if.so uid < 1000 quiet
account     required      pam_permit.so

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
password    sufficient    pam_unix.so sha512 shadow nullok try_first_pass use_authtok
password    required      pam_deny.so

session     optional      pam_keyinit.so revoke
```



```
session     required      pam_limits.so
-session     optional      pam_systemd.so
session     [success=1 default=ignore] pam_succeed_if.so service in crond quiet use_uid
session     required      pam_unix.so
```

表 2-9 pam_unix.so 配置项说明

配置项	说明
sha512	使用sha512算法对口令加密。

2.2.5.6 登录失败超过三次后锁定

说明

为了保障用户安全，设置口令出错锁定阈值为3次，设置由于口令尝试被锁定的用户的自动解锁时间为5分钟（300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暴力破解，增强系统的安全性。

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，配置如下：

```
#%PAM-1.0
# User changes will be destroyed the next time authconfig is run.
auth        required      pam_env.so
auth        required      pam_faillock.so preauth audit deny=3 even_deny_root unlock_time=300
auth        sufficient    pam_fprintd.so
auth        sufficient    pam_unix.so nullok try_first_pass
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
auth        requisite     pam_succeed_if.so uid >= 1000 quiet_success
auth        required      pam_deny.so

account     required      pam_unix.so
account     required      pam_faillock.so
account     sufficient    pam_localuser.so
account     sufficient    pam_succeed_if.so uid < 1000 quiet
account     required      pam_permit.so

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
password    sufficient    pam_unix.so sha512 shadow nullok try_first_pass use_authtok
password    required      pam_deny.so

session     optional      pam_keyinit.so revoke
session     required      pam_limits.so
-session     optional      pam_systemd.so
session     [success=1 default=ignore] pam_succeed_if.so service in crond quiet use_uid
session     required      pam_unix.so
```

表 2-10 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。

配置项	说明
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300秒（即5分钟）。
even_deny_root	同样限制root账户。

2.3 附录

介绍文件权限的含义和umask值的含义。

2.3.1 文件和目录权限含义

Linux系统中文件和目录权限用于限定谁能通过何种方式对文件和目录进行访问和操作。文件和目录的访问权限分为只读，只写和可执行三种。

有三种不同类型的用户可对文件和目录进行访问：

- 文件所有者：文件的创建者。
- 同组用户：与文件所有者在同一个属组的用户。
- 其他用户：与文件所有者不在同一个属组的用户。

文件和目录的权限含义通过以下例子说明：

假设/usr/src的权限为755，将每位数字转化为二进制后为：111101101，含义如下：

- 左侧三位111表示文件所有者的权限为：可读、可写、可执行。
- 中间三位101表示同组用户的权限为：可读、不可写、可执行。
- 右侧三位101表示其他用户的权限为：可读、不可写、可执行。

2.3.2 umask 值含义

当用户新创建文件或目录时，该文件或目录具有一个缺省权限。该缺省权限由umask值来指定。

umask值代表的是权限的“补码”，即用缺省最大权限值减去umask值得到实际权限值。文件的缺省最大权限为可读可写，目录的缺省最大权限为可读可写可执行。即一个文件的实际缺省权限为666减去umask值。目录的实际缺省权限为777减去umask值。

3 安全维护手册

本手册介绍对EulerOS系统进行的日常安全维护操作，指导维护人员例行进行安全维护。

3.1 安全维护概述

介绍安全维护的目的和三个层次。

3.1.1 安全维护的目的

随着信息技术及网络应用的发展，信息系统面临的安全威胁越来越严重，操作系统作为整个信息系统的核心，一旦出现问题，将面临业务中断、系统崩溃等威胁，给客户带来不可估量的损失。因此，需要从多个层次构建、维护整个信息系统的安全屏障，提前发现、快速定位并及时处理各种可能存在的安全问题，从而降低客户面临的安全风险。

由于安全隐患层出不穷，完全依赖技术很难全面保证应用系统的安全。因此，客户需要根据安全维护建议和日常发现的问题，建立安全管理的制度，来保证系统安全、正常的运行。

3.1.2 分层的安全维护

根据安全维护的对象和目的，维护人员需要从不同层次对业务系统进行安全维护。

系统层

系统层针对EulerOS自研OS版本，系统层安全维护的目的是保障操作系统可以正常运行，以支撑应用层各个应用软件的运行。

系统层维护的内容主要包括系统服务、账户口令、文件权限、内核参数、系统访问认证授权五项内容。

网络层

网络层安全维护的目的是保障防火墙网络安全工具的正常运行，并提供远程登录管理、远程登录日志审计、网络扫描、网络监测等功能，便于客户在网络环境下维护操作系统的安全。

网络层的安全维护一般是基于维护对象所对应的维护终端或维护工具来实施。

管理层

管理层安全维护的目的是加强人为管理，防范于未然。管理层维护涉及上述各个层面。

3.2 系统层安全

介绍对EulerOS系统进行的安全维护操作。

3.2.1 系统层账户清单

3.2.1.1 主机账户清单

说明

当前所列的账户是“基础系统”的账户，用户安装EulerOS时，在“软件选择”中选择其他选项，会导致系统账户不同。

安装时已创建的主机账户清单如表3-1所示。

表 3-1 主机账户清单

账户	属组	账户描述
root	root	超级管理员账户
bin	bin	bin账户
daemon	daemon	daemon账户
adm	adm	adm账户
lp	lp	打印服务账户
sync	root	同步服务账户
shutdown	root	关机服务账户
halt	root	关机服务账户

账户	属组	账户描述
mail	mail	邮件服务账户
operator	root	操作账户
games	users	games账户
ftp	ftp	FTP账户
nobody	nobody	nobody账户
dbus	dbus	dbus服务账户
polkitd	polkitd	polkit服务账户
avahi-autoipd	avahi-autoipd	Avahi IPv4LL账户
tss	tss	TrouSerS组件账户，用于可信计算
sshd	sshd	SSH服务账户
ntp	ntp	ntp服务账户
systemd-bus-proxy	systemd-bus-proxy	systemd Bus Proxy 账户。systemd-bus-proxyd进程用于连接标准IO或者socket到特定的总线地址。
systemd-network	systemd-network	systemd Network Management账户。systemd-networkd是管理网络的系统服务。它检测和配置出现的网络设备，以及创建虚拟网络设备。
libstoragemgmt	libstoragemgmt	libStorageMgmt守护进程账户

3.2.2 账户口令维护

3.2.2.1 账户维护策略

管理建议

- 进行严格的账户管理，实施严格的账户策略。
- 严格控制增加、修改、删除系统中的账户、群组。
- 删除所有系统上不使用的账户。
- 管理员创建账户时，需要明确权限和职责、操作建议。
- 管理员代行root权限时，需要先自行登录再通过su切换到root账户。
- 采用sha512对系统口令进行加密。

3.2.2.2 创建账户

在遵从最小账户原则的基础上，管理员可根据需要创建账户。使用系统命令useradd来创建新的账户。

说明

仅root用户可以创建和删除账户。

操作说明

```
useradd [-u UID] [-g GID] [-d HOME] [-mM] [-s shell] username [-p PASSSWORD]
```

相关参数说明请参见表3-2。

表 3-2 创建账户参数说明

参数	参数说明
-u	直接给予一个UID。
-g	直接给予一个GID。
-d	直接将该用户的根目录指向已经存在的目录（系统不会再建立）。
-m	新建该用户的根目录，并将/etc/skel中的文件复制到用户根目录。
-M	不建立用户的根目录。
-s	定义其使用的shell。
-p	设置用户的密码。 **说明** 使用useradd、usermod、groupadd、groupmod命令时，不推荐使用-p或--password参数。设置、修改口令建议使用passwd命令，具体请参照3.2.2.8 修改口令章节。

操作举例

- 创建一个账户名称为test的普通用户。
  ```
  useradd test
  ```
- 创建一个普通账户，其根目录为home/test。
  ```
  useradd -d /home/test test
  ```

3.2.2.3 删除账户

若账户长期未使用，或者账户的口令超过有效期，管理员可删除该账户。使用系统命令userdel来删除账户。

参数说明

```
userdel [-r] account
```

相关参数说明请参见表3-3。

表 3-3 删除账户参数说明

参数	参数说明
-r	删除该用户目录下的所有目录。

操作举例

删除用户test。

```
userdel test
```

3.2.2.4 设置账户的有效期

为保证账户的安全性，应设置账户的有效期，使用系统命令chage来设置账户的有效期。

操作说明

```
chage [-m mindays] [-M maxdays] [-d lastday] [-I inactive] [-E expiredate] [-W warndays] user
```

相关参数说明请参见表3-4。

表 3-4 设置账户有效期

参数	参数说明
-m	口令可更改的最小天数。为零时代表任何时候都可以更改口令。
-M	口令保持有效的最大天数。为-1时可删除这项口令的检测。
-d	上一次更改的日期。
-l	列出当前的设置。由非特权用户来确定他们的口令或账户何时过期。
-I	停滞时期。过期指定天数后，设定密码为失效状态。
-E	账户到期的日期。过了这天，此账户将不可用。
-W	用户口令到期前，提前收到警告信息的天数。

说明

- 日期格式为YYYY-MM-DD，如chage -E 2013-12-01 test表示账户test的口令在2013年12月1日过期。
- User不写的话默认是root用户。

操作举例

修改用户test的有效期为2013年12月31日。

```
chage -E 2013-12-31 test
```


说明

对于超过期限的用户，应该检查是否允许继续使用，如果不使用则应该将这些用户删除，防止用户被非授权使用，删除账户请参考删除账户小节。

3.2.2.5 变更账户权限

管理员可通过添加用户到不同的组或从组中删除，来达到权限变更的目的。

操作说明

```
groupmems -g group [-a|-d] user
```

相关参数说明请参见表3-5。

表 3-5 变更账户群组

参数	参数说明
-g	指定群组名称。
-a	将指定账户加到指定群组。
-d	将指定账户加从指定群组中移除。

操作举例

将用户test增加到特权组root。

```
groupmems -g root -a test
```

3.2.2.6 锁定账户

为保证系统安全性，管理员可锁定暂时不使用的账户，之后根据实际情况再进行解锁。通过系统命令passwd或usermod来锁定账户。

操作说明

锁定账户可使用如下命令：

- passwd -l user
- usermod -L user

操作举例

锁定test账户。

```
passwd -l test
```

3.2.2.7 解锁账户

通过系统命令passwd或usermod来解锁账户。

操作说明

解锁账户可使用如下命令：

- passwd -u user
- usermod -U user

操作举例

解锁test账户。

```
passwd -u test
```

3.2.2.8 修改口令

如果用户忘记口令，或者账户的口令有效期快到了，管理员可以重置账户口令。普通用户可修改自己的口令来延长使用期限。使用系统命令passwd来修改账户口令。

操作说明

```
passwd [-g] [name]
```

说明

不加任何参数表示修改当前账户的口令。普通账户修改口令时都要求先输入旧口令，旧口令不匹配则无法修改口令，新口令需要满足口令复杂度要求，具体请参照2.2.5.3 设置口令复杂度。

相关参数说明请参表3-6。

表 3-6 修改口令参数说明

参数	参数说明
-g	表示修改指定群组的口令

说明

- chpasswd命令是批量更新用户口令的工具，把一个文件内容重新定向添加到/etc/shadow中，不会对口令复杂度进行校验，且可以配置空口令。管理员在使用此命令时，请确保口令复杂度满足业务安全要求，避免风险。
- EulerOS和业界主流发行版本一致，提供开源bash支持，history记录所有bash输入。对于涉及bash命令行明文密码的非安全操作方式，会导致明文密码被记录到history中，存在密码泄露风险，如示例1、2所示。建议用户优先使用对应程序提供的安全方式修改密码，如passwd交互式方式则不会记录密码到history。

  eg 1：

  ```
  passwd test << EOF
  >明文密码
  >明文密码
  >EOF
  ```

  eg 2：

  ```
  echo "明文密码" | passwd --stdin test
  ```

  EulerOS不建议用户使用非安全方式，如果用户使用场景涉及非安全方式，或者有些开源命令只接收明文命令行密码，请及时清理history以防密码泄露，清理命令如下：

  ```
  history -d 行号
  ```


操作举例

linux通用机制，命令输入支持管道的方式，但是以这种方式修改密码，会导致密码明文记录。出于安全考虑，不建议使用这种方式。为确保密码安全，请选择安全的修改方式。

例如，修改test账户的口令：

```
passwd test
```

3.2.2.9 监控账户操作

管理员需要定期检查各个账户的操作，包括账户是否操作异常或者非法操作。

操作说明

打开账户根目录下的“.bash_history([home]/.bash_history)”文件，查看账户的操作历史，确认是否存在非法或异常操作。

3.2.2.10 检查账户

管理员应对运行账户进行定期检查，检查内容包括是否存在未知账户、不合理的账户。

账户清单请参见3.2.1.1 主机账户清单。

操作说明

步骤1 打开文件/etc/passwd，比较文件中的账户与3.2.1.1 主机账户清单描述的账户是否一致。若不一致，建议查出多余账户存在的原因，若无故增加，则应删除。

步骤2 检查系统账户是否可登录，即账户的shell是否为/sbin/nologin（/etc/passwd文件每行最后一个字段值）；若不是，则应改为/sbin/nologin。

----结束

3.2.2.11 系统口令加密算法维护

管理员可以对系统的口令加密算法进行定期检查，查看当前系统的加密算法是否与设定的算法一致。默认情况下的加密算法是sha512。

表 3-7 口令加密算法列表

算法名称	识别码	是否可逆	描述
des	无	是	一种对称加密算法，采用64位密钥技术，实际只有56位有效，8位用来校验的。
md5	1	否	Message-Digest Algorithm 5。md5的作用是让大容量信息在用数字签名软件签署私人密钥前被"压缩"成一种保密的格式（就是把一个任意长度的字符串变换成一定长度的十六进制数字串）

算法名称	识别码	是否可逆	描述
sha256	5	否	Secure Hash Algorithm，安全散列算法。sha256适用于长度不超过2^64二进制位的消息，sha512适用于长度不超过2^128二进制位的消息。
sha512	6	否	
blowfish	2/2a	是	BlowFish算法用来加密64Bit长度的字符串。

操作说明

步骤1 打开/etc/pam.d/password-auth和/etc/pam.d/system-auth文件，确认password sufficient pam_unix.so后的参数中的口令加密算法：

```
#%PAM-1.0
# User changes will be destroyed the next time authconfig is run.
auth        required      pam_env.so
auth        required      pam_faillock.so preauth audit deny=3 even_deny_root unlock_time=300
auth        sufficient    pam_fprintd.so
auth        sufficient    pam_unix.so nullok try_first_pass
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
auth        requisite     pam_succeed_if.so uid >= 1000 quiet_success
auth        required      pam_deny.so

account     required      pam_unix.so
account     required      pam_faillock.so
account     sufficient    pam_localuser.so
account     sufficient    pam_succeed_if.so uid < 1000 quiet
account     required      pam_permit.so

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
password    sufficient    pam_unix.so sha512 shadow nullok try_first_pass use_authtok
password    required      pam_deny.so

session     optional      pam_keyinit.so revoke
session     required      pam_limits.so
-session     optional      pam_systemd.so
session     [success=1 default=ignore] pam_succeed_if.so service in crond quiet use_uid
session     required      pam_unix.so
```

步骤2 到口令加密算法列表中查找对应算法的识别码。

步骤3 打开/etc/shadow文件，查看当前系统账户行中的第二字段的起始值是否与识别码一致。

----结束

操作举例

步骤1 要确认账户root的口令加密算法，打开/etc/pam.d/password-auth和/etc/pam.d/system-auth文件，查看password sufficient pam_unix.so后的参数中的口令加密算法为sha512。

步骤2 到口令加密算法中找到sha512对应的识别码是6。

步骤3 打开/etc/shadow文件，发现root账户第二字段起始字符为“6”，说明与设定的算法一致。

----结束

3.2.3 系统服务维护

3.2.3.1 服务维护策略

管理建议

- 最小服务和组件。
- 区分服务器的用途和角色，尽量避免安装不必要的服务和组件。
- 关闭未使用的服务。
- 服务内部组件也应采用上述原则进行裁减。

3.2.3.2 检查进程

管理员进行安全维护时，需要检查是否有多余的进程，及时发现不合理的进程并处理，避免风险。

操作说明

步骤1 用系统命令“ps -aux”查看系统当前所有进程。

步骤2 查看是否有多余进程（根据用户安装的组件不同，运行的进程有差异，具体由用户根据实际情况甄别）。

步骤3 若发现有多余进程，则用系统命令“kill -9 PID”删除该进程。

----结束

3.2.3.3 检查服务/端口

管理员应检查是否有多余的服务，如果有多余服务及时停止，以避免安全风险。

操作说明

1. 用户执行systemctl list-units --type=service命令查看系统所有服务。
2. 查看是否有不需要使用的服务（根据用户安装的组件不同，开启服务有差异，具体由用户根据实际情况甄别）。
3. 若存在不需要的服务，可用如下方式删除服务。

   以rsyslog服务为例，如下操作：

   a. 执行systemctl stop rsyslog命令停止该服务运行。

   b. 执行systemctl disable rsyslog命令关闭该服务随机启动。

> **警告**
>
> 本操作前请务必先确认该服务为不需要的服务。

3.2.3.4 检查主机间通信

管理员应检查是否有多余的、非法的主机间通信，避免风险。

操作说明

步骤1 用系统命令“netstat -an”查看当前系统中开放的所有端口。

步骤2 参考通信矩阵中的端口列表，检查当前是否有多余开放的端口。

步骤3 若存在多余端口，确认是否有必要开放，若无必要，则应关闭。

----结束

3.2.4 日志审计系统维护

3.2.4.1 日志文件列表

为保证应用系统安全，系统管理员需要定期检查系统的日志信息，日志信息请参见表3-8。

表 3-8 日志信息

日志文件	路径	内容概要
messages	/var/log/	记录除私人认证消息外的其他系统日志信息
secure	/var/log/	记录验证及授权等信息
maillog	/var/log/	记录mail日志
cron	/var/log/	记录定时任务的日志
spooler	/var/log/	记录news日志
boot.log	/var/log/	记录系统启动日志
dmesg	/var/log/	记录系统内核启动日志
wtmp	/var/log/	记录用户登录日志
lastlog	/var/log/	记录用户登录日志
tallylog	/var/log/	记录PAM_TALLY锁定日志
euleros-security.log	/var/log/	安全加固日志

说明

此表仅作参考，用户可能会根据实际需求重新规划日志目录，请以用户的实际使用情况为准。

3.2.4.2 系统日志类别与等级

本系统使用syslog记录系统日志。syslog能通过产生日志的类别（facility）和等级（level）对日志做分类处理，将日志写到文件或设备。它既可以记录本地日志，也可以通过网络记录另一个主机上的日志。

日志等级

日志等级level请参见表3-9（优先级由上而下降低）。

表 3-9 日志等级

level	介绍
emerg	最严重的错误，将导致系统不可用，如系统panic
alert	需要立即处理的错误
crit	关键错误
err	一般错误，等同error
warn	警告信息，等同于warning
notice	重要信息
info	普通信息
debug	调试信息

3.2.4.3 检查系统日志

开启关闭系统日志

- 开启系统日志，使用命令
  ```
  #systemctl start rsyslog
  ```
- 关闭系统日志，使用命令：
  ```
  #systemctl stop rsyslog
  ```

检查日志开关的状态

检查日志开关的状态，使用命令：

```
#systemctl status rsyslog
```

- 输出为active (running)，则为开启状态。
- 输出为inactive (dead)，则为关闭状态。

检查日志存储空间

建议系统管理员定期清理/var/log/中的日志文件，备份、删除旧的日志信息，释放空间。

3.2.4.4 开启关闭审计系统

- 开启
  ```
  systemctl start auditd
  auditctl -e 1
  ```
- 关闭
  ```
  systemctl stop auditd
  ```


3.2.4.5 检查审计开关的状态

```
systemctl status auditd
```

- 输出为active (running)，则为开启状态。
- 输出为inactive (dead)，则为关闭状态。

3.2.4.6 定制审计策略

审计系统可以使用auditctl命令来动态管理审计参数和审计规则，也可以将审计规则静态的写入到/etc/audit/audit.rules文件中。

操作说明

```
auditctl [options]
```

options取值请参见表3-10。

表 3-10 auditctl 参数说明

选项名	说明
-b <backlog>	设置内核允许的缓冲区数，默认值为64。
-e [0|1]	关闭或启动内核审计系统。
-f [0..2]	设置失败标识0=silent 1=printk 2=panic，默认值为1。设置内核如何处理临界错误，如：backlog限制超出、内存错误等。
-h	帮助信息。
-i	当从文件中读取规则时忽略错误。
-l	列出所有的规则，每行一条规则。
-k <key>	设置审计规则上的过滤关键词key，key是不超过32字节长的任意字符串，它能唯一鉴别由watch产生的审计记录。
-m text	仅由root用户发送用户空间消息到审计系统。为文件系统watch设置许可过滤器。r=read，w=write，x=execute，a=attribute change。这些许可不是文件的标准许可，而是系统调用使用的，read和write系统调用将忽略这种设置，否则它们将淹没log。
-r <rate>	设置每秒传输的消息数限制，默认值为0，表示无限制。
-R <file>	从file文件中读取规则。
-s	报告状态。
-a <l,a>	追加规则到l链表，a表示规则的动作。
-A <l,a>	添加规则到l链表头，动作为a。
-d <l,a>	从带有a动作的l链表删除规则。
-D	删除所有的规则和watch。
-S [系统调用名或号|all]	如果程序使用指定的系统调用，则它启动一项审计记录。如果给出域规则而没有指定系统调用，它将默认为所有系统调用。

表 3-11 链表名和规则说明

分类	选项	说明
有效链表名l	task	追加规则到每个任务链表AUDIT_FILTER_TASK 。域应用任务创建时的uid、gid等。
	entry	追加规则到系统调用进入链表AUDIT_FILTER_ENTRY，用于决定进入到系统调用时是否创建审计事件。
	exit	追加规则到系统调用退出链表AUDIT_FILTER_EXIT 。用于决定退出系统调用时是否创建审计事件。
	user	追加规则到用户消息过滤链表AUDIT_FILTER_USER，内核在转播用户空间产生的事件到审计后台之前，用这个链表过滤这些事件。仅域为uid、auid、gid和pid时有效。
	exclude	用于过滤不想看到的事件，对应内核消息过滤链表AUDIT_FILTER_TYPE。
规则的有效动作a	never	不产生审计记录。
	always	分配一个审计上下文，在系统调用退出时填充。

操作举例

- 查看所有不成功的open系统调用。
  ```
  auditctl -a entry,always -S open -F success!=0
  ```
- 监控/etc/audit/audit.rules的变化，将以下内容加入到audit.rules中即可实现。
  ```
  -w /etc/audit/audit.rules -k TEST_audit_rules -p rxwa
  ```

3.2.4.7 检查审计日志

工具ausearch用于查询审计后台的日志，它能基于不同搜索规则的时间查询审计后台日志。

操作说明

```
ausearch [options]
```

说明

options参数说明请自行查阅相关man帮助信息。

特别指出的，每个系统调用进入内核空间运行时有个唯一的事件ID，系统调用在进入内核后的运行过程的审计事件共享这个ID。也就是说，一个审计事件，可能包含几条审计记录。

操作举例

查询操作/etc/audit/rules.d/audit.rules文件的审计日志。

```
ausearch -k TEST_audit_rules
```

3.2.4.8 生成审计报告

audit审计系统使用aureport工具分析审计日志，对分析结果做出总结，并生成审计报告。

系统管理员可以根据维护的需要定期生成审计报告，用于分析异常的审计信息。

操作说明

```
aureport [options]
```

说明

options参数说明请自行查阅相关man帮助信息。

操作举例

报告失败事件。

```
aureport --failed
```

3.2.5 认证与授权维护

3.2.5.1 维护 PAM 策略

本系统使用Linux-PAM实现用户的身份鉴别、口令复杂度控制、登录阈值设置等。PAM主要是由一组共享库文件（在/lib64/security/目录下）和一些配置文件（在/etc/pam.d/目录下）组成的。

系统管理员可以根据维护需要对PAM策略进行修改。

口令复杂度维护

系统默认的设置口令复杂度如下：

（1）口令长度最短应不少于6位（管理员用户至少8位）。

（2）口令至少包含“小写字母（a-z）、大写字母（A-Z）、数字（0-9）、特殊字符”中的3种。

（3）口令不能和账户或者账户的倒写一样。

（4）不能修改为过去5次使用过的旧口令。

用户在设置root账户口令或者新创建账户口令时，强烈建议按照密码复杂度要求设置。根据维护需要，系统管理员通过修改/etc/pam.d/system-auth和/etc/pam.d/password-auth的pam_pwquality.so和pam_pwhistory.so配置来重新设置口令复杂度，但是如果关闭复杂度配置会导致安全风险。修改策略根据实际情况谨慎决策，建议用户使用EulerOS如下默认配置：

表 3-12 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

配置项	说明
minclass=3	口令至少包含大写字母、小写字母、数字任意两种和特殊字符中的任意3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3次
enforce_for_root	本设置对root用户同样有效

表 3-13 pam_pwhistory.so 配置项说明

配置项	说明
remember=5	口令不能修改为过去5次使用过的旧口令
enforce_for_root	本设置对root用户同样有效

口令复杂度维护操作说明

以修改口令长度为例，操作步骤如下

步骤1 修改/etc/pam.d/system-auth和/etc/pam.d/password-auth文件的pam_pwquality.so后的minlen字段的值，如下：

```
minlen=10
```

口令长度至少为10位。

步骤2 保存并退出/etc/pam.d/system-auth和/etc/pam.d/password-auth文件。

----结束

登录出错阈值维护

系统设置默认的登录出错阈值为3次（若连续登录出错次数超过3次，则账户被锁定5分钟）。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刷新；解锁后，用户的错误输入记录被清空。

根据维护需要，系统管理员通过修改/etc/pam.d/system-auth和/etc/pam.d/password-auth的pam_faillock.so配置来重新设置登录出错阈值，pam_faillock.so配置项如下。

表 3-14 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。

配置项	说明
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300秒（即5分钟）。
even_deny_root	同样限制root用户。

系统管理员在登录系统后可以通过执行“faillock --user *XXXX* --reset”命令来清空用户登录出错的次数记录（XXXX是对应的用户名）。

登录出错阈值维护操作说明

口令输错次数记录清空操作步骤如下：

步骤1 使用root用户登录系统。

步骤2 执行如下命令，清空口令输错次数记录。也可以清除指定用户的错误次数记录。

----结束

```
faillock --user XXXX --reset
```

说明

如果root用户因为登录口令输错3次被锁定，必须等待300秒才能再次登录系统，登录成功后pam_faillock会清空root用户的登录出错信息。

修改出错阈值的操作步骤如下：

步骤1 修改/etc/pam.d/system-auth和/etc/pam.d/password-auth文件的pam_faillock.so后的deny字段的值，如下：

deny=5

设置出错阈值为5次，即连续登录出错次数超过5次后账户被锁定300秒。

步骤2 保存并退出/etc/pam.d/system-auth和/etc/pam.d/password-auth文件。

----结束

su 权限维护

系统默认只允许root和wheel群组的用户使用su命令，限制其他用户使用su命令。

根据维护需要，系统管理员可以授予、回收其他用户使用su命令的权限，操作如下：

- 查看当前具有su权限的用户。
  ```
  cat /etc/group | grep wheel | cut -d : -f 4
  ```
  各用户间以“,”分隔。
- 授予su权限。
  ```
  groupmems -g wheel -a user
  ```
  user为允许使用su命令的用户。
- 回收su权限。
  ```
  groupmems -g wheel -d user
  ```


3.2.5.2 维护 SSH

SSH是安全shell的简写。SSH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。透过SSH可以对所有传输的数据进行加密，也防止DNS欺骗和IP。加固SSH服务，是指修改SSH服务中的一些配置，来提高系统的安全性。

操作说明

步骤1 打开SSH配置文件/etc/ssh/sshd_config，查看文件中的字段值与安全加固设定的默认值是否一致。若不一致，查明原因，并修改。

步骤2 重启SSH服务：

```
systemctl restart sshd.service
```

----结束

3.2.6 文件权限维护

3.2.6.1 检查文件权限

系统管理员必须保证系统配置文件，信息文件，系统设备文件和系统二进制文件不能被非系统管理员修改。所有文件都是在特定条件下作为通用可写被创建出来的。

操作说明

使用系统命令“ls -al [filename | path]”查看指定文件的权限。

3.2.6.2 修改文件权限

出于安全需要，管理员可以对文件权限进行设置和修改。

操作说明

步骤1 使用系统命令chmod修改文件权限。

步骤2 使用系统命令chown修改文件或目录的拥有者或组。

----结束

操作举例

步骤1 修改目录/bin的权限为755。

```
chmod 755 /bin
```

步骤2 把目录/bin的属主和组都改为root。

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

----结束

3.2.7 内核参数维护

3.2.7.1 检查与修改内核参数

内核参数决定配置和应用特权的状态，管理员应定期查看系统中的内核参数的值是否有变更。内核提供用户可配置的系统控制，这一系统控制可微调至提高系统的安全性。

操作说明

步骤1 通过系统命令“sysctl -a”列出当前系统中的所有内核参数。

步骤2 将当前系统参数值与安全加固中的设置一一比较，观察其值有无更改。

步骤3 若有更改，则打开内核参数配置文件/etc/sysctl.conf，查看有无参数在文件中配置。

- 若有，则确定参数值是否与表格中一致。
- 若不存在，则按配置文件中的格式添加对参数的配置。

步骤4 执行系统命令“sysctl -p”使配置的参数值立即生效。

----结束

3.3 网络层安全

介绍对防火墙网络安全工具的安全维护操作。

3.3.1 防火墙管理

3.3.1.1 防火墙规则配置

防火墙需要配置一系列的规则，以决定什么样的数据包能够通过。防火墙基于iptables实现，系统管理员可以通过iptables配置防火墙规则。

查看当前防火墙配置

```
iptables -S
```

更改防火墙规则配置

防火墙规则配置请参考iptables相关帮助，此处不提供。

说明

通过iptables更改防火墙规则后，仅当前有效，防火墙重启后失效。

3.3.2 远程接入控制

EulerOS远程接入控制主要是SSH远程接入，因此本章节主要描述SSH远程接入的维护内容。

3.3.2.1 设置账户密钥

SSH支持2种认证方式：基于口令的认证和基于密钥的认证。

- 基于口令的认证方式，只要知道账户和口令就可以登录远程主机，但是容易受到“中间人”攻击（仿冒目的服务器）。

- 基于密钥的认证需要生成公私密钥对，将公钥存放于服务器，登录服务器时使用密钥进行验证。与口令认证相比，密钥认证不需要在网络上传送口令，且可以防范“中间人”攻击。

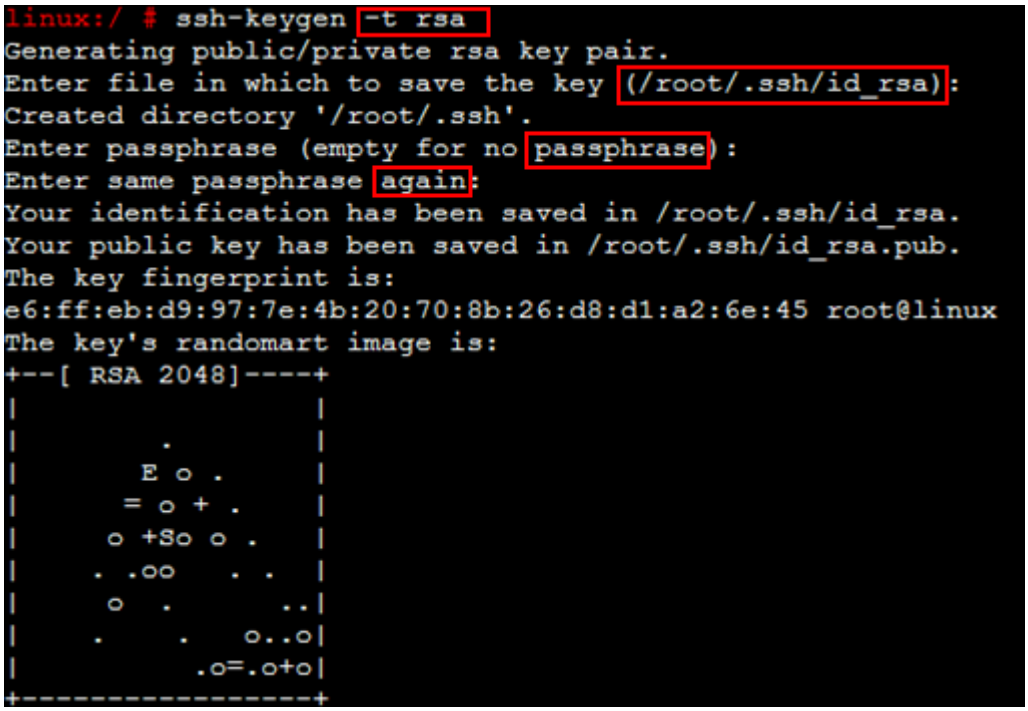
设置账户密钥的步骤如下：

步骤1 在服务器端，修改/etc/ssh/sshd_config配置文件如下字段的值：

```
RSAAuthentication yes
PubkeyAuthentication yes
AuthorizedKeysFile .ssh/authorized_keys
PasswordAuthentication no
ChallengeResponseAuthentication yes
StrictModes yes
```

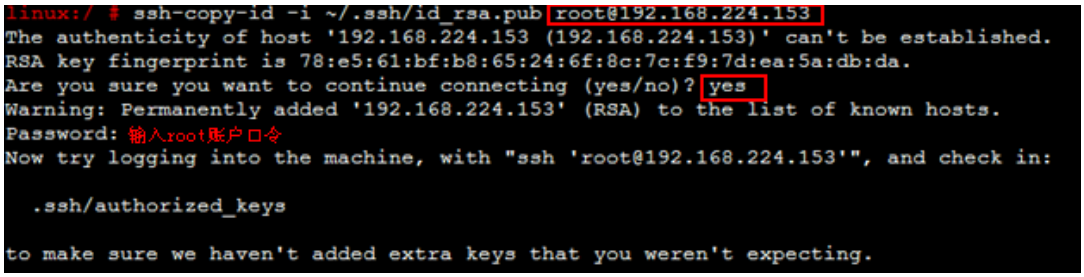
步骤2 在客户端使用ssh-keygen生成公私密钥对，生成文件存放于/root/.ssh/id_rsa，密钥口令为空，如图3-1所示。

图 3-1 生成公私密钥对



步骤3 在客户端使用ssh-copy-id将公钥传递至远程服务器，如图3-2所示。

图 3-2 传递公钥至远程服务器



说明

如果缺少ssh-copy-id命令请先安装ssh-copy-id工具，或者使用scp等将公钥拷贝至远程服务器

步骤4 在客户端使用密钥认证，登录远程服务器，如图3-3所示。

图 3-3 使用密钥认证登录远程服务器

```
linux:/ # ssh root@192.168.224.153
Last login: Mon Jun 18 19:29:51 2012 from 192.168.224.153
linux:~ #
```

----结束

3.3.2.2 远程连接管理

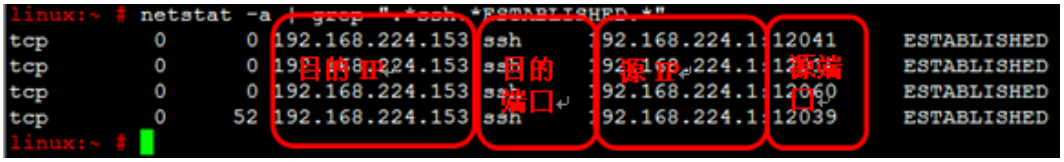
查看当前 SSH 服务连接

可以通过以下命令查看当前系统所有的SSH服务连接。

```
netstat -a | grep ".*ssh.*ESTABLISHED.*"
```

如图3-4所示。

图 3-4 SSH 服务链接



关闭异常 SSH 服务连接

- 使用以下命令关闭异常SSH服务连接。
  ```
  iptables -I INPUT -s 192.168.224.1 -m state --state ESTABLISHED -j DROP
  ```
  其中192.168.224.1为异常客户端的IP。
- 如果异常情况为误报或异常已经被清除，则使用如下命令删除上述限制。
  ```
  iptables -D INPUT -s 192.168.224.1 -m state --state ESTABLISHED -j DROP
  ```

3.3.2.3 远程接入日志审计

为保证系统安全，系统管理员需要每天检查远程接入日志，日志信息存放在/var/log/messages文件中。

查看SSH登录日志。命令示例如下：

```
cat /var/log/messages | grep ".*sshd.*for.*from.*"
```

3.4 管理层安全

介绍维护人员在人为管理上对系统应进行的安全维护操作。

3.4.1 账户维护建议

建议系统管理员对账户例行检查，检查的内容包括：

- 操作系统的账户是否必要，临时账户是否已删除。

- 账户的权限是否合理。
- 对账户的登录、操作日志进行检查和审计。

3.4.2 口令维护建议

用户身份验证是应用系统的门户。用户的账户和口令的复杂性、有效期等应根据运营商的安全要求进行配置。

对口令的维护建议如下：

- 专人保管主机root口令。
- 口令传递时注意加密，尽量避免通过邮件传递口令。
- 口令需要加密存储。
- 系统移交时提醒运营商更改口令。

3.4.3 日志维护建议

本节描述日常维护原则中日志相关的建议。

- 专人负责权限控制。
- 日志应当定期进行备份，同时将备份文件在外介质（磁盘、磁带、光盘等）上存档。

  详细描述参见单独的备份和恢复指导。
- 日志在进行备份后应当及时删除，以释放磁盘空间。
- 如有可能，需要进行日志集中审计。

利用审核和日志记录来帮助发现可疑的活动。系统对于重要业务（包括系统参数、资费配置与发布等）的操作需要记录日志。通过系统加固对日志文件进行保护。

定期检查日志

定期查看系统日志、应用程序日志及安全日志，若发现有异常日志出现，应及时向上级部门汇报，若不能定位原因或无法自行解决时，及时向当地华为办事处或拨打*4008302118*向华为公司求助。

定期备份日志

日志应当进行定期备份，同时将备份文件在外介质（磁盘、磁带、光盘等）上存档。日志在备份后应当及时删除，以释放日志空间。

3.4.4 安全评估建议

建议客户定期对系统进行安全评估，特别是在进行系统重大升级、网络搬迁、系统扩容等造成网络变更较大的情况下。

建议客户找具有安全评估资质的专业机构对系统进行安全评估，评估时请与华为技术服务工程师联系。

3.4.5 漏洞扫描建议

漏洞扫描通常是指基于漏洞数据库，通过扫描等手段，对指定的远程或者本地计算机系统的安全脆弱性进行检测，发现存在风险的漏洞的一种安全检测（渗透攻击）行为。

漏洞扫描是对当前系统进行全方位的扫描，检查您当前的系统是否有漏洞，如果有漏洞则需要马上进行修复，否则系统很容易受到网络的伤害甚至被黑客借助于系统的漏洞进行远程控制，那么后果将不堪设想，所以漏洞扫描对于保护系统安全和上网安全是必不可少的。

漏洞扫描建议用Nessus进行扫描。Nessus漏洞描述工具扩展性强，漏洞库全面。Nessus工具的使用指导可通过访问nessus官网**http://www.nessus.org/products/nessus**进行获取。

漏洞扫描频率建议为每星期一次。

- 如果扫描报告有一般的中低级漏洞，则管理员可尝试通过扫描报告的说明试着修复，也可收集漏洞信息后报告给华为技术工程师。
- 如果发现有高危漏洞，管理员需要马上通知华为技术工程师进行远程指导或现场支持，力求在最短的时间内修复漏洞。

3.4.6 备份建议

出于安全防护的需要，请在如下场景进行备份：

- 加固之前和之后进行系统数据的全量备份。
- 在进行日常安全配置维护、故障处理之前和之后进行备份。
- 安装补丁、升级时备份，请参见对应的指导书。

3.4.7 网络连接变更建议

当网络连接状况发生变化时（包括物理网络拓扑和服务器内虚拟网络拓扑，正常运行过程中发生的虚拟机热生命周期管理特别是虚拟机热迁移都会使网络拓扑发生变化），需要注意是否破坏了原有的安全策略。包括检查虚拟机是否被划在正确的vlan内，防火墙配置策略是否跟当前的网络拓扑匹配。

如果用户需要实施网络重新部署或安全组重新划分等对网络连接变动较大的动作时，建议先进行网络拓扑分析，避免引入安全漏洞。

3.4.8 缺陷报告建议

如果客户向华为报告系统遭到攻击，华为将根据攻击的具体情况采用如下两种处理方式：

- 如果现场发生了安全事故，华为技术支持工程师将提供远程或者现场支持，协同用户维护人员减轻系统遭受攻击的影响，并且完善现场事故报告的处理过程。
- 如果没有发生安全事故，华为技术支持工程师将把问题录入数据库并传给研发团队。研发团队找到解决方案后，技术支持工程师将分析方案实施对现场业务的影响，并提供建议的解决方法。

3.4.9 补丁管理建议

客户需要建立应用程序补丁的管理制度，设置相关人员，检视华为发布的补丁，并检查操作系统、数据库、中间件厂商发布的补丁。

注意

如果需要打补丁，请与华为技术工程师联系，切勿自行升级。

3.4.10 安全应急响应机制

用户需要建立应对安全事故的应急响应处理机制，以保证出现安全事故后，可以尽快恢复生产和解决问题，将损失降至最低。

3.5 系统服务安全

众所周知，用户访问管理控制对机构管理员来说是个重要问题，但监控哪些网络处于活跃状态对任何一位管理员以及Linux系统操作者来说都更为重要。

EulerOS中的很多服务都类似网络服务器，如果在一个机器上运行网络服务，那么服务器应用程序（亦称为daemon），就会侦听一个或者多个网络端口的连接。这些服务器被视为潜在的攻击手段。

3.5.1 服务的风险

网络服务可对Linux系统造成很多危险。以下是一些主要问题列表：

- 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。
- 分布的拒绝服务攻击（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。
- 脚本漏洞攻击：网页服务器通常使用脚本执行服务器端动作，破解者就可以攻击没有正确编写的脚本。这些脚本漏洞攻击导致缓存溢出，或者允许攻击者更改系统中的文件。
- 缓存溢出攻击：连接到特权端口为1023的服务器必须作为管理用户来运行。如果应用程序有可利用的缓存溢出，那么攻击者就可作为运行该应用程序的用户访问系统。因为有可利用的缓存溢出存在，破解者可使用自动工具来识别有漏洞的系统，并在获得访问后，使用自动工具套件保持其对该系统的访问。

说明

要限制通过网络进行攻击，应该将所有不使用的服务关闭。

3.5.2 识别并配置服务

要提高安全性，默认关闭在EulerOS中安装的大多数服务。但有些是例外：

- cups：EulerOS的默认打印服务器。
- cups-lpd：备用打印服务器。
- xinetd：控制与一系列下级服务器连接的超级服务器，比如gssftp和telnet超级服务器。
- sshd：OpenSSH服务器，是Telnet的安全替代产品。

管理员需要合理识别并配置服务，确保服务正常运行。例如：若不使用打印机，则不需要cups继续运行，此原理同样适用于portreserv。如果未挂载NFSv3卷或者未使用NIS（ypbind服务），则应该禁用 rpcbind。不但要检查开机启动时可用的网络服务，而且应该检查已打开并在侦听的端口。

3.5.3 不安全的服务

任何网络服务都是不安全的，所以需要关闭不使用的服务。管理员发现并修补服务漏洞，这些工作对更新与网络服务有关的软件包非常重要。

以下为安全级别较低的网络协议包含的服务：

- 以不加密的方式在网络中传输用户名和密码。很多老的协议，比如Telnet和FTP，它们对认证会话都不加密，应尽量避免使用。
- 以不加密方式传输敏感数据。很多协议在网络间传输数据时不加密，这些协议包括Telent、FTP、HTTP和SMTP。很多网络文件系统，比如NFS和SMB也以不加密的方式在网络间传输信息。用户在使用这些协议时有责任限制要传输的数据类型。

本身就不安全的服务示例包括rlogin、rsh、telnet、以及vsftpd。

所有远程登录和shell程序（rlogin、rsh、以及telnet）应避免使用以支持SSH。详情请参阅**第3.5.10节“保障SSH”**有关sshd。

FTP并不象远程shell那样天生对系统安全有威胁，但需要小心配置并监控FTP服务器以免出问题。有关保障FTP服务器安全的详情请参阅**第3.5.8 节“保障FTP安全”**。

应小心使用并在开启防火墙后使用的服务包括：

- auth
- nfs-server
- smb以及nbm（Samba）
- yppasswdd
- ypserv
- ypxfrd

3.5.4 保障 rpcbind

rpcbind服务是为NIS和NFS等RPC服务进行动态端口分配的守护进程。它的认证机制比较薄弱，并可以为其控制的服务大范围的分配端口，因此很难保障其安全。

说明

因为NFSv4不再需要rpcbind，所以保障rpcbind安全只影响NFSv2和NFSv3的执行。如果您要运行NFSv2或者NFSv3服务器，就需要rpcbind。

如果运行RPC服务，需要遵守以下基本规则。

使用 TCPWrapper 保护 rpcbind

因为TCP Wrapper没有内嵌的认证形式，所以使用TCP Wrapper限制网络或者限制主机访问rpcbind服务很重要。

另外，限制对服务的访问时，只需要使用IP地址。由于通过使DNS中毒和其它方法可以伪造主机名，所以请避免使用主机名。

使用防火墙保护 rpcbind

要进一步限制访问rpcbind服务，最好是为该服务器添加firewalld规则，并限制对具体网络的访问。

以下是firewalld Rich Text命令的两个示例。第一个是实现从网络192.168.0.0/24到111端口（rpcbind服务使用的端口）的TCP连接的示例。第二个是实现从本地主机到同一端口的TCP连接的示例。丢弃所有其它数据包。以下操作均是root用户。

```
# firewall-cmd --add-rich-rule='rule family="ipv4" port port="111" protocol="tcp" source
address="192.168.0.0/24" invert="True" drop'
```



```
# firewall-cmd --add-rich-rule='rule family="ipv4" port port="111" protocol="tcp" source
address="127.0.0.1" accept'
```

同样地，要限制UDP流量，则须使用以下命令：

```
# firewall-cmd --add-rich-rule='rule family="ipv4" port port="111" protocol="udp" source
address="192.168.0.0/24" invert="True" drop'
```

说明

将--permanent添加到firewalld Rich Text命令中，以实现永久设置。

3.5.5 保障 NIS 安全

“网络信息服务”（NIS）是一个RPC服务，亦称之为ypserv，可与rpcbind及其它相关服务一同使用，与声明在其域的中的所有计算机发布用户名、密码以及其它敏感信息进行映射。

NIS服务器由许多应用程序组成。它包括以下的应用程序：

- /usr/sbin/rpc.yppasswdd：也称为yppasswdd服务，这个守护进程允许用户更改其NIS密码。
- /usr/sbin/rpc.ypxfrd：也称为ypxfrd服务，这个守护进程负责通过网络的NIS映射传输。
- /usr/sbin/ypserv：这是NIS服务器守护进程。

就当今的标准而言，NIS在某种程度上并不安全。它没有主机认证机制，且所有通过网络的传输都是不加密的，包括哈希密码。因此设置使用NIS的网络时，要特别小心。事实上，NIS的默认配置本身就不安全，这也让情况变得更为复杂。

若管理员想要运行NIS服务器，先要保障rpcbind服务的安全，正如在**第3.5.4节“保障rpcbind”**中概括的那样，然后解决以下的问题。

谨慎规划网络

由于NIS通过网络传输敏感信息时未经加密，所以在防火墙后，且在隔离和安全的网络中运行就非常重要。使用不安全的网络传输NIS信息，无论何时都有被截获的风险。谨慎规划网络有助于防止严重的安全漏洞。

使用类似密码的 NIS 域名和主机名

只要用户知道NIS服务器的DNS主机名和NIS域名，那么在NIS域中的任何计算机都可以在未经认证的情况下使用命令从服务器中提取信息。

例如：如果有人是从笔记本电脑连接到网络或者从外部侵入（并要嗅探内部IP地址），那么以下命令就可揭示/etc/passwd映射：

```
ypcat -d <NIS_domain> -h <DNS_hostname> passwd
```

如果这个攻击者是root用户，那么他们就可通过以下命令获取/etc/shadow文件：

```
ypcat -d <NIS_domain> -h <DNS_hostname> shadow
```

注意

如果使用Kerberos，那么/etc/shadow文件就不会储存在NIS映射中。

要让攻击者更难访问NIS映射，则须让DNS主机名生成一个随机字符串，比如o7hfawtgmhwg.domain.com。同样地，也可创建一个“不同的”随机NIS域名。这就让攻击者访问该NIS服务器变得更加困难。

编辑/var/yp/securenets 文件

如果/var/yp/securenets文件是空白文件，或是根本不存在（默认安装后就是这种情况），那么NIS就会侦听所有网络。首先要做的就是在该文件中添加子网掩码/网络对，这样一来ypserv只会响应来自对应网络的请求。

以下是/var/yp/securenets文件的条目示例：

```
255.255.255.0       192.168.0.0
```

注意

首次启动NIS服务器时，一定要有已生成的/var/yp/securenets文件。

这个技术并不提供对IP嗅探式攻击的保护，但至少可以限制NIS服务器提供服务的网络。

分配静态端口并使用 Rich Text 规则

所有与NIS关联的服务器都可以分配到指定的端口，rpc.yppasswdd除外（该守护进程允许用户更改其登录密码）。其它两个NIS服务器守护进程rpc.ypxfrd和ypserv分配端口，这就可允许创建防火墙规则，以便进一步防止入侵者破坏NIS服务器守护进程。

要做到这一点，在/etc/sysconfig/network中添加以下命令行：

```
YPSERV_ARGS="-p 834"
YPXFRD_ARGS="-p 835"
```

以下rich text firewalld规则可用于强制设定服务器用这些端口进行侦听的网络：

```
# firewall-cmd --add-rich-rule='rule family="ipv4" source address="192.168.0.0/24" invert="True"
port port="834-835" protocol="tcp" drop'
# firewall-cmd --add-rich-rule='rule family="ipv4" source address="192.168.0.0/24" invert="True"
port port="834-835" protocol="udp" drop'
```

这就是说，如果请求来自192.168.0.0/24网络，那么服务器就只可连接到834和835端口。第一规则用于TCP，第二规则用于UDP。

使用 Kerberos 认证

NIS用于认证操作时，其中要考虑的问题是，无论用户何时登录机器，/etc/shadow映射上的哈希密码都是通过网络进行传送。如果入侵者可以访问NIS域或者探查网络流量，那么他们就可以收集用户名以及哈希密码。攻击者可以利用密码破译程序猜破译较弱的密码，继而可以访问网络上的有效账户。

因为Kerberos使用密钥加密，则不需要通过网络发送哈希密码，所以系统就更加安全。

3.5.6 保障 NFS 安全

NFS流量可通过使用不同版本的TCP进行传送，但它应在NFSv3下使用，而不是UDP；在使用NFSv4时，NFS流量是必要的。所有版本的NFS都支持Kerberos用户和分组认证，作为RPCSEC_GSS内核模块的一部分。因为EulerOS支持NFSv3使用rpcbind，所以有关rpcbind信息也包括在内。

谨慎规划网络

NFSv2和NFSv3传统上来说，不能安全地传输数据。现在所有版本的NFS都有能力对使用Kerberos的普通文件系统进行认证（且进行选择性加密）。在NFSv4下，可以使用Kerberos；在V2或V3下，锁定文件和挂载文件仍无法使用Kerberos。当使用NFSv4时，如果客户处于NAT或者防火墙的保护下，那么可能会关闭授权。

保障 NFS 挂载选项

从安全管理的角度来说，/etc/nfsmount.conf详细讲解了NFS挂载选项，这可用于设定客户默认选项。

1. 审查NFS服务器

   注意

   只能导出整个文件系统，导出文件系统的子目录成为一个安全问题。因为有些情况下，客户可能会“跳出”文件系统的导出的子目录，获取文件系统中未导出的目录。

   使用ro选项可使文件系统导出的属性为“只读取”，这在任何时候都会减少可对挂载文件系统进行写入操作的用户数量。只有在明确要求的情况下，才能使用rw选项。例如，允许写入访问，则会加大符号链接攻击的风险。这包括临时目录，如/tmp和/usr/tmp。

   用rw选项挂载目录时，要避免全域可写，这在任何时候都可降低风险。就像某些应用程序以明文储存密码或是储存加密强度较弱的密码，导出主目录也被视为有风险的操作。审查和改进应用代码可以减少这种风险。一些用户没有在他们的SSH密钥上设定密码，因此这也意味着主目录存在风险。强制使用密码或者使用Kerberos可以减少风险。

   限定只有需要访问权限的客户才能导出目录。在NFS服务器上使用showmount-e命令来审查该服务器导出的内容。请勿导出没有明确需求的任何内容。

   请勿使用no_root_squash选项，并且审查现有的安装程序，以确保并未使用该选项。更多信息，请参阅第**第3.5.6节“请勿使用no_root_squash选项”**。

   secure选项是服务器端导出选项，用于限定服务器端只能从“保留”端口进行导出。默认情况下，服务器只允许客户通过“保留”端口（端口编号不超过1024）进行通讯，因为传统上来说，客户只允许通过“可信”代码（例如内核NFS客户）来使用这些端口。然而在大部分网络上，普通用户要成为某些客户端的root并不难。因此，如果保留端口所进行的通讯拥有特权，对于服务器而言，通常都是不安全的。综上所述，限制保留端口并不能解决安全问题，最好还是依靠kerneros、防火墙、以及限定只有特定客户才能进行导出。

   如果可能的话，大多数的客户仍使用保留端口。然而，保留端口是有限的资源，因此客户（尤其是那些拥有大量NFS装载的客户）可以选择编号更高的端口。Linux客户可以通过使用“noresvport”挂载选项来完成。如果用户希望在导出目录中允许此运作，那么可以通过“insecure”导出选项来完成。

   禁止用户登录服务器是一个很好的做法。在审查NFS服务器的上述设置时，也审查能访问和进入服务器的人和内容。

2. 审查NFS客户

   使用nosuid选项来禁止使用setuid程序。nosuid选项可禁用set-user-identifier或set-group-identifier位，这可阻止远程用户通过运行setuid程序获取更高的特权，在客户端和服务器端使用该选项。

noexec选项可禁止客户端上的所有可执行文件。使用此选项可防止用户无意中执行了文件系统中所共享的文件，对于大多数的（即使不是全部的）文件系统而言，nosuid和noexec选项都是标准选项。

使用nodev选项可防止客户端将“device-files”作为硬件设备进行处理。

resvport选项是客户端挂载选项，secure是相应的服务器端导出选项。resvport限定只有使用“保留端口”才能进行通讯。“保留”端口或“知名”端口会保留给特权用户或程序，比如root用户。设置这个选项会促使客户使用保留的源端口与服务器进行通讯。

现在，所有版本的NFS都支持挂载Kerberos认证。启用这个挂载选项：sec=krb5。

NFSv4支持用Kerberos进行挂载，通过使用krb5i来确保完整性，使用krb5p来确保隐私保护。在使用sec=krb5进行挂载时，上述这些都会使用到，但需要在NFS服务器上配置。

注意语法错误

NFS服务器通过查阅/etc/exports文件，判断导出哪些文件系统以及将这些目录导出到哪些主机中。

注意

编辑此文件时，不要添加多余的空格。

例如，/etc/exports文件中的以下命令行可实现与主机bob.example.com共享/tmp/nfs/目录的读/写权限。

```
/tmp/nfs/     bob.example.com(rw)
```

另一方面，由于主机名的一个空格，这使/etc/exports中的以下命令行可实现与主机bob.example.com共享同一目录的只读权限，同时实现与“所有人”共享它的读 / 写权限。

```
/tmp/nfs/     bob.example.com (rw)
```

最好使用showmount命令检查所有已配置的NFS共享，以确定共享的内容：

```
showmount -e <hostname>
```

请勿使用 no_root_squash 选项

默认情况下，NFS共享会将root用户更改为一个非特权用户帐户，即nfsnobody用户。这会将所有root创建的文件的所有者更改为nfsnobody，这可防止用setuid位组来设置程序的上传。

注意

如果使用no_root_squash，那么远程root用户就可以更改共享文件系统中的任何文件，并留下感染木马的应用程序给其它用户去执行。

NFS 防火墙配置

NFSv4是EulerOS默认的NFS版本，且它要求只对TCP开放2049端口。如果使用NFSv3，那么就需要四个额外的端口，如下述说明。

为NFSv3配置端口。NFS使用的端口是由rpcbind进行动态分配，在创建防火墙规则时，可能会造成问题。要简化这个步骤，则须使用/etc/sysconfig/nfs文件指定要使用的端口：

- MOUNTD_PORT：用于挂载的TCP和UDP端口（rpc.mountd）。
- STATD_PORT：用于显示TCP和UDP状态的端口（rpc.statd）。
- LOCKD_TCPPORT：用于nlockmgr的TCP端口（rpc.lockd）。
- LOCKD_UDPPORT：用于nlockmgr的UDP端口（rpc.lockd）。

注意

指定的端口号绝对不能用于其它服务。对防火墙进行配置，可指定端口号以及TCP和UDP的2049端口（NFS）。

在NFS服务器上运行服务器上运行rpcinfo -p命令，可查看所使用的端口和RPC程序。

3.5.7 保障 Apache HTTP 服务器安全

ApacheHTTP有很多可用的选项和技术可用于保障Apache HTTP服务器安全。以下小节简要介绍了在运行Apache HTTP服务器时可采用的操作。

在投入生产之前，一定要核实所有脚本都可在系统中运行。另外，确保只有root用户才有权限写入含脚本或者CGI的任何目录。要做到这一点，则须作为root用户运行以下命令：

```
chown root <directory_name>
chmod 755 <directory_name>
```

系统管理员应谨慎使用以下配置选项（在/etc/httpd/conf/httpd.conf 进行配置）：

- FollowSymLinks

  此指令为默认启用，因此在创建符号链接到网页服务器的文档root目录时，请慎重行事。例如，请勿为“/”提供符号链接。

- Indexes

  虽然此指令为默认启用，但并非必要。要防止访问者浏览在服务器上的文件，则须删除这个指令。

- UserDir

  因为此指令可确认系统中用户帐户是否存在，所以要默认禁用UserDir指令。要在服务器上启用用户名目录浏览，则须使用以下指令：

  ```
  UserDir enabled
  UserDir disabled root
  ```

  这些指令用于/root/之外的所有用户目录，可激活其用户目录浏览这一功能。要在禁用帐户列表中添加用户，则须在UserDir disabled命令行添加以空格分隔的用户列表。

- ServerTokens

  ServerTokens指令控制着服务器响应标题头信息，这信息会传送回给客户。它包括不同的信息，通过使用下列参数，可以对其进行自定义操作。

  a. ServerTokens Full（默认选项），提供所有可用信息（OS类型以及所使用的模块），例如：

     ```
     Apache/2.0.41 (Unix) PHP/4.2.2 MyMod/1.2
     ```

  b. ServerTokens Prod或者ServerTokens ProductOnly，提供以下信息：

```
Apache
```

c. ServerTokens Major，提供以下信息：

```
Apache/2
```

d. ServerTokens Minor，提供以下信息：

```
Apache/2.0
```

e. ServerTokens Min或者ServerTokens Minimal，提供以下信息：

```
Apache/2.0.41
```

f. ServerTokens OS，提供以下信息：

```
Apache/2.0.41 (Unix)
```

建议使用ServerTokens Prod选项，这样一来，潜在攻击者就无法获取关于您系统的任何有用信息。

说明

请勿删除IncludesNoExec指令。默认情况下，“服务器端嵌入”（SSI）模块无法执行命令。除非绝对必要，建议您不要更改这个设置，因为它可能会允许攻击者在系统中执行命令。

删除 httpd 模式

在某些情况下，最好删除特定的httpd模式，以限制HTTP服务器的功能。要实现这一目的，只需为整个命令行添加注释，该命令行用于加载在/etc/httpd/conf/httpd.conf文件中需要删除的模块。例如，要删除代理模块，则须通过给下列命令行新增“#”字符，为下列命令行添加注释：

```
#LoadModule proxy_module modules/mod_proxy.so
```

请注意，/etc/httpd/conf.d/目录包含了可用于加载模块的配置文件。

3.5.8 保障 FTP 安全

“文件传输协议”（FTP）是一个比较旧的TCP协议，用来通过网络传输文件。因为服务器所处理的所有传输，包括用户认证，都是未经加密，所以认为FTP是一个不安全的协议，且应该谨慎地进行配置。

下列安全指南可用于设置vsftpd FTP服务。

FTP 登录信息

提交用户名和密码前，所有用户都会看到登录信息。默认情况下，这个信息包含了版本信息，这使得破解者能够轻易识别系统弱点。

要为vsftpd更改登录信息，则须在/etc/vsftpd/vsftpd.conf文件中添加以下指令：

```
ftpd_banner=<insert_greeting_here>
```

用登录信息文本替换上述指令中的<insert_greeting_here>。

对于多行信息而言，最好使用信息文件。要简化多提示信息管理，则须将所有提示信息放入名为/etc/banners/的新目录。在本示例中，用于FTP连接的提示信息文件为/etc/banners/ftp.msg。以下为此类文件的示例：

```
######### Hello, all activity on ftp.example.com is logged. #########
```

要在vsftpd中引用这个登录信息，则须在/etc/vsftpd/vsftpd.conf 文件中添加以下指令：

```
banner_file=/etc/banners/ftp.msg
```

还可以发送附加信息提示给使用TCP Wrapper的连入连接。

匿名访问

/var/ftp/目录的存在可激活匿名帐户。创建这个目录的最简单的方法是安装vsftpd软件包，这个软件包可为匿名用户建立目录树，并为匿名用户配置目录的只读权限。

默认情况下，匿名用户不能写入任何目录。

注意

如果启用对FTP服务器的匿名访问，那么就要注意保存敏感数据的位置。

- 匿名上传

  要允许匿名用户上传文件，那么建议在/var/ftp/pub/中生成只写目录。要完成此操作，则须作为root用户运行以下命令：

  ```
  # mkdir /var/ftp/pub/upload
  ```

  下一步，更改权限以防止匿名用户查看该目录中的内容：

  ```
  # chmod 730 /var/ftp/pub/upload
  ```

  该目录的详细格式列表应如下所示：

  ```
  # ls -ld /var/ftp/pub/upload
  drwx-wx---. 2 root ftp 4096 Nov 14 22:57 /var/ftp/pub/upload
  ```

  允许匿名用户在服务器目录中读取和写入，这是非常危险的操作。

  另外，在vsftpd下，在/etc/vsftpd/vsftpd.conf文件中添加以下行：

  ```
  anon_upload_enable=YES
  ```

用户帐户

注意

因为FTP用不安全的网络对未经加密的用户名和密码进行认证，所以最好拒绝系统用户从其用户帐户访问服务器。

要禁用vsftpd中的所有用户帐户，则须在/etc/vsftpd/vsftpd.conf中添加以下指令：

```
local_enable=NO
```

限制用户帐户。要禁止FTP访问特殊账户或者特殊群组账户，例如root用户或者拥有sudo特权的用户，最简单的方法就是使用PAM列表文件。用于vsftpd的PAM配置文件是/etc/pam.d/vsftpd。

还可以在每个服务中直接禁用用户帐户。

要在vsftpd中禁用特定帐户，则需在/etc/vsftpd/ftpusers中添加用户名。

3.5.9 保障 Postfix 的安全

Postfix是邮件传输代理（MTA），它使用简单邮件传输协议（SMTP）在其它MTA和电子邮件客户端或者传递代理之间传递电子信息。虽然很多MTA都可以在彼此之间加密流量，但大多数并不这样做，因此使用任何公共网络发送电子邮件都被视为不安全的沟通形式。Postfix替代Sendmail成为EulerOS默认的MTA。

建议使用Postfix服务器的用户解决以下问题。

限制拒绝服务攻击

因为电子邮件的本质，攻击者可以很容易地使用邮件对服务器进行洪水攻击，导致拒绝服务。通过对/etc/postfix/main.cf文件中的指令进行限制设定，可以有效地阻止此类攻击。通过更改已经存在的指令赋值，或是以下列格式，将所要的值添加到所需的指令中：

```
<directive> = <value>
```

以下一系列指令可用于限制拒绝服务攻击：

- smtpd_client_connection_rate_limit：单位时间内，任何客户尝试与这个服务进行连接的最大次数。如果默认值是0，这就意味着在单位时间内，客户可进行的连接次数与Postfix能接收的连接次数一样多。默认情况下，可排除在信任网络中的客户。
- anvil_rate_time_unit：该时间单元被用于速率限制的计算。默认值是60秒。
- smtpd_client_event_limit_exceptions：从连接和速率限制命令中所排除的客户。默认情况下，也可排除在信任网络中的客户。
- smtpd_client_message_rate_limit：单位时间内，客户被允许进行请求传递信息的最大次数（不管Postfix是否真的接收这些信息）。
- default_process_limit：提供特定服务的Postfix子进程默认的最大值。这种限制可能因为在master.cf文件中的特定服务而取消。默认情况下，赋值为100。
- queue_minfree：在队列文件系统中，接收邮件所需的最小可用空间（以字节为单位）。Postfix SMTP服务器使用此指令来判断是否可以接收邮件。默认情况下，当可用空间的最小值小于message_size_limit的1.5倍时，Postfix SMTP服务器则会拒绝MAIL FROM指令。要具体制定一个更高的可用空间最小值限定，则须具体制定一个queue_minfree值，其大小至少是message_size_limit的1.5倍。默认情况下，queue_minfree值是0。
- header_size_limit：用于储存信息标题的最大内存（以字节为单位）。如果标题太大，那么超出的部分就会被舍弃。默认情况下，赋值为102400。
- message_size_limit：信息的最大值（以字节为单位），包括信封信息。默认情况下，赋值为10240000。

NFS 以及 Postfix

请勿将邮件spool目录，/var/spool/postfix/，放到NFS共享卷上。因为NFSv2和NFSv3不会保持对用户ID和组群ID的控制，所以两个或者更多用户可以有相同的UID，并接收和读取彼此的邮件。

说明

在NFSv4中使用Kerberos，就不会出现这种情况。因为SECRPC_GSS内核模块不会根据UID进行认证。但是，最好还是不要将邮件池目录放到NFS共享卷中。

只使用邮件的用户

为避免本地用户利用Postfix服务器上的漏洞，最好让邮件用户只能通过电子邮件程序访问Postfix服务器。应该禁止邮件服务器上的shell帐户访问，并且/etc/passwd文件中的所有shell用户都应设定到/sbin/nologin中（除了root用户之外）。

禁用 Postfix 网络侦听

默认情况下，Postfix被设定为只侦听本地回路地址，可以通过查看/etc/postfix/main.cf文件来核实。

查看/etc/postfix/main.cf文件，以确保只出现下列inet_interfaces命令行：

```
inet_interfaces = localhost
```

这样确保Postfix只接收来自本地系统而非来自网络的邮件信息（比如定时任务报告）。这是默认设置，并且保护Postfix免受网络攻击。

inet_interfaces=all设置可用于删除本地主机限制，并且允许Postfix侦听所有接口。

3.5.10 保障 SSH

Secure Shell（SSH）是一个强大的网络协议，可通过安全的渠道与其他系统进行通讯。通过SSH的传输的信息都经过加密，可避免被拦截。

加密登录

SSH支持使用加密密钥登录电脑，这比只使用密码要更安全。如果您可以把这种方法与其他受到认证的方法相结合，那么这就被认为是多因素认证。有关如何使用多种认证方法的更多信息，请参阅**第3.5.10节“多种认证方法”**。

为了启用加密密钥进行认证，在/etc/ssh/sshd_config文件中的PubkeyAuthentication配置指令需要设定为“yes”。请注意，这是默认设置。把PasswordAuthentication指令设定为“no”，则不支持使用密码登录。

使用ssh-keygen命令可以生成SSH密钥。如果在没有其它参数的情况下，调用SSH密钥，则会生成2048位RSA密钥集。在默认情况下，密钥储存在~/.ssh目录中。您可以使用-b切换更改密钥强度。正常情况下，使用2048位密钥就足够了。

在~/.ssh目录中，能够看到两个密钥。当运行ssh-keygen命令时，如果用户接受这种默认情况，那么所生成文件就会命名为id_rsa和id_rsa.pub，并且分别含有公钥和私钥。用户应当随时保护私钥，将其设置为除文件所有者外其他任何人都不可读取，使其免于暴露。然而，公钥则需要传送到用户将要登录的系统。用户可以使用ssh-copy-id命令来传送密钥至服务器：

```
$ ssh-copy-id -i [user@]server
```

这个命令会自动把公钥添加到服务器上的~/.ssh/authorized_key文件中。当用户试图登录服务器时，sshd守护进程就会检查此文件。

同样地，对于密码以及其他认证机制，用户也应该时常更改SSH密钥。这样做的时候，请确保从authorized_key文件中移除所有不用的密钥。

多种认证方法

使用多种认证方法或者多因素认证，会提升保护水平以防止未经授权的访问，强化系统以防止被入侵。尝试使用多因素认证登录系统的用户，必须成功通过所有指定的认证方法，才能得到授权进行访问。

使用/etc/ssh/sshd_config文件中的AuthenticationMethods配置指令，可指定要使用的认证方法。

说明

使用此指令可以定义多份所需的认证方法列表，用户必须完成至少一份列表上的每种方法。列表需用空格进行分隔，且列表中，每个认证方法的名称必须用逗号分隔。

例如：

```
AuthenticationMethods publickey,gssapi-with-mic publickey,keyboard-interactive
```

如果尝试登录成功的用户是通过publickey认证和gssapi-with-mic认证，或是publickey认证和keyboard-interactive认证，那么只有使用上述的AuthenticationMethods指令进行配置的sshd守护进程才能得到授权进行访问。请注意，每个所要求的认证方法都要使用对应的配置指令（例如，/etc/ssh/sshd_config文件中的PubkeyAuthentication），方可准确地启用。

其他方法保障 SSH 安全

- 协议版本

  由EulerOS所提供的SSH协议，即使此协议的运行支持SSH-1以及SSH-2版本的协议，但是尽量使用SSH-2版本的协议。SSH-2版本比起旧版SSH-1作了一些的改进，并且大多数高级配置选项只在使用SSH-2时才可用。

  建议用户使用SSH-2，这可使SSH协议对所使用的认证和通讯的保护范围达到最大化。通过使用/etc/ssh/sshd_config文件中的Protocol配置指令，可指定sshd守护进程所支持的版本协议或是其他版本的协议。默认设置是2。

- 密钥类型

  默认情况下，ssh-keygen命令会生成一对SSH-2RSA默认密钥；使用-t选项，通过指令它也可生成DSA或ECDSA密钥。ECDSA（EllipticCurveDigitalSignatureAlgorithm，椭圆曲线数字签名算法）在同等的密钥长度下可提供更好的操作。它也可生成较短的密钥。

- 非默认端口

  默认情况下，sshd守护进程会侦听22网络端口。更改端口会减少系统受到基于自动网络扫描而造成的攻击，从而增加其安全性。通过使用/etc/ssh/sshd_config配置文件中Port指令，可指定端口。请注意，要允许使用非默认端口，必须更改SELinux默认设置。通过作为root输入以下指令，修改ssh_port_tSELinux类型，可以完成此操作：

  ```
  # semanage -a -t ssh_port_t -p tcp port_number
  ```

  在上述命令中，用Port指令指定的新端口号代替port_number。

- 非root登录

  如果无需作为root用户登录，应该在/etc/ssh/sshd_config文件中把PermitRootLogin配置指令设置成no。通过禁止作为root用户登录，管理者可以审核常规用户登录后运行了什么特权命令，且之后获取了root权限。

3.6 附录

3.6.1 安全维护任务列表

3.6.1.1 日维护表

表 3-15 日维护表

维护项	具体操作	对应措施
账户口令维护	查看当前系统的账户中是否有多余账户，打开文件/etc/passwd，并与账户清单比较。	若发现异常账户，确定是否为管理员认可的临时账户，若是，将其设置使用的有效期，有效期设置请参见**3.2.2.4 设置账户的有效期**。
	查看系统账户是否可登录。打开文件/etc/passwd，检查系统账户的Shell是否为/sbin/nologin。	若发现有系统账户的shell不是/sbin/nologin，则应将其修改。
日志审计维护	请参见**3.2.4.3 检查系统日志**，查看系统日志功能是否被关闭。	若发现系统日志功能被关闭，请参见**3.2.4.3 检查系统日志**章节开启日志系统。
	请参见**3.2.4.3 检查系统日志**，检查日志是否输出。	如果未输出日志，请参见**3.2.4.3 检查系统日志**章节开启日志系统。
	请参见**3.2.4.5 检查审计开关的状态**，查看审计系统是否被关闭。	若发现审计系统被关闭，确认是否需要重新开启审计系统，若需开启，请参见**3.2.4.4 开启关闭审计系统**。
	使用auditctl -l或查看/etc/audit/audit.rules文件，确认审计策略是否有变更。	如果审计策略有变更，则确认变更是否合理，不合理的删除该审计策略。
内核参数维护	检查当前系统内核参数值是否与内核参数列表一致。	若不一致，建议将参数值改成初始值，具体操作请参见**3.2.7.1 检查与修改内核参数**。
防火墙策略维护	再用命令“iptables -S”查看防火墙规则是否合理。	若防火墙规则中有不合理的规则，如接受不明网络的数据包，则应删除该规则，具体操作详见iptables指南。

3.6.1.2 周维护表

表 3-16 周维护表

维护项	具体操作	对应措施
账户口令维护	账户的使用有效期是否快到期或已经到期。	若有账户快到期，管理员需要通过实际情况确认账户是否有需要继续使用，若有需要，则管理员应将其使用期限延长，具体操作请参见**3.2.2.4 设置账户的有效期**。
	请参见**3.2.2.9 监控账户操作**，监控各账户的日常操作是否异常。	若有账户进行异常操作，应予以警告，或终止账户的操作权限。

维护项	具体操作	对应措施
系统服务维护	请参见**3.2.3.2 检查进程**，检查是否有不合理的进程出现。	若发现有多余的进程，则将其删除。
	请参见**3.2.3.3 检查服务/端口**，检查是否有多余服务。	若发现有多余服务，则将其删除。
	请参见**3.2.3.4 检查主机间通信**，检查是否有多余的、非法的主机间通信。	若发现有多余端口，先确认是否有必要开放，若无必要，则将其关闭。
日志审计维护	请参见**3.2.4.3 检查系统日志**，检查日志存储空间。	如果日志文件过大，需要定期转储或删除。
认证授权维护	请参见**口令复杂度维护**，检查口令复杂度设置是否符合要求。	若口令复杂度设置不满足要求，则按照**口令复杂度维护**修改口令复杂度，使其满足要求。
	请参**见登录出错阈值维护**，检查登录出错阈值设置是否符合规定。	若登录出错阈值设置不满足要求，则按照**登录出错阈值维护**修改系统登录出错阈值，使其满足要求。
	请参见**su权限维护**，检查当前系统具有su权限的用户是否应该具备su权限。	若用户不应该具备su权限，则按照**su权限维护**回收相应用户的su权限。
	请参见**3.2.5.2 维护SSH**，检查SSH的配置是否与**2.2.1.1 加固SSH服务**中一致。	若不一致，则查明不一致的原因，若被恶意修改，则按照**操作说明**将配置还原。
文件权限维护	请参见**3.2.6.1 检查文件权限**，查看文件权限是否满足安全需求。	若文件权限遭修改，则先查明原因，若是遭恶意修改，请参见**3.2.6.1 检查文件权限**章节修改文件权限。

3.6.2 脚本&命令清单

EulerOS_脚本&命令清单.xlsx

说明

EulerOS操作系统基于开源标准内核构建，兼容Redhat的外围包。这里介绍EulerOS自研的脚本和命令，对于自带的开源脚本和命令，用户可以查询对应开源社区文档或者man帮助信息，不在清单中说明。

4 通信矩阵

通信矩阵.xlsx

5 内核协议说明

EulerOS使用标准内核，默认支持所有标准内核协议，产品可根据业务需要使用iptables裁剪不需要的协议，协议列表如下：

表 5-1 EulerOS 支持协议列表

支持协议	协议号	说明
ICMP	1	网络控制消息协定（Internet Control Message Protocol）是网路协议族的核心协议之一，用于TCP/IP网络中发送控制消息。
IGMP	2	因特网组管理协议（Internet Group Management Protocol）是用于管理因特网协议多播组成员的一种通信协议。
TCP	6	传输控制协议（Transmission Control Protocol）是一种面向连接的、可靠的、基于字节流的传输层通信协议，由RFC 793定义。
UDP	17	用户数据报协议（User Datagram Protocol）是一个简单的面向数据报的传输层协议，由RFC 768定义。
UDPLite	136	轻量级用户数据包协议，由RFC 3828定义。
PIM	103	协议无关多播（Protocol Independent Multicast）是多播路由协议簇，可以利用静态路由或者任意单播路由协议（包括RIP、OSPF、IS-IS、BGP等）所生成的单播路由表为IP组播提供路由。

Nmap协议扫描结果如下：

表 5-2 Nmap 协议扫描结果

PROTOCOL	STATE	SERVICE	REASON
1	open	icmp	port-unreach
2	open|filtered	igmp	no-response
6	open	tcp	proto-response
17	open	udp	port-unreach
103	open|filtered	pim	no-response
136	open|filtered	udplite	no-response

原始扫描报告：pro.rar

说明

2、103、136协议的状态为open|filtered，是由于Nmap构造的探测报文仅包含报文头，不包含任何有效负载，内核协议栈将其丢弃导致。

6 SecureCAT 扫描结果及分析报告

EulerOS使用SecureCAT工具扫描得分超过70，详细参见如下分析报告

EulerOS SecureCAT Report.xlsm

7 漏洞分析报告

7.1 33929 - PCI DSS compliance

漏洞描述

Name	33929 - PCI DSS compliance
Synopsis	The remote host has been found to be NOT COMPLIANT with the PCI DSS external scanning requirements.
Description	The remote host is vulnerable to one or more conditions that are considered to be 'automatic failures' according to the PCI DSS Approved Scanning Vendors Program Guide (version 2.0). These failures include one or more of the following : - Vulnerabilities with a CVSS base score greater than or equal to 4.0 - Unsupported operating systems - Internet reachable database servers (must validate whether cardholder data is stored) - Presence of built-in or default accounts - Unrestricted DNS Zone transfers - Unvalidated parameters leading to SQL injection attacks - Cross-Site Scripting (XSS) flaws - Directory traversal vulnerabilities - HTTP response splitting/header injection - Detection of backdoor applications (malware, trojan horses, rootkits, backdoors) - Use of older, insecure SSL/TLS versions (TLS v1.1 is the minimum standard) Details of the failed items may be found in the 'Output' section of this plugin result. These vulnerabilities and/or failure conditions will have to be corrected before you are able to submit your scan results for validation by Tenable to meet your quarterly external scanning requirements. If you are conducting this scan via Nessus Cloud and either disagree with any of the results, believe there are false-positives, or must rely on compensating controls to mitigate the vulnerability then you may proceed with submitting this report to our PCI portal by clicking on 'Submit for PCI Validation'. You may login to the Tenable PCI portal using your Nessus Cloud credentials and dispute or provide mitigation evidence for each of the residual findings.
See Also	**http://www.pcisecuritystandards.org** **https://discussions.nessus.org/community/pci**
Solution	N/A

Risk Factor	High
Plugin Information:	Publication date: 2008/08/07, Modification date: 2015/07/23
Ports	tcp/0
CVSS Base Score	N/A

分析结果

这个是PCI DSS合规检查，是第三方支付行业的数据安全标准， EulerOS不涉及此漏洞。

7.2 56209 - PCI DSS Compliance : Remote Access Software Has Been Detected

漏洞描述

Name	56209 - PCI DSS Compliance : Remote Access Software Has Been Detected
Synopsis	A remote access software has been detected.
Description	Due to increased risk to the cardholder data environment when remote access software is present, please 1) justify the business need for this software to the ASV and 2) confirm it is either implemented securely per Appendix D in the ASV Program Guide, or disabled / removed. Please consult your ASV if you have questions about this Special Note.
See Also	N/A
Solution	N/A
Risk Factor	Medium
CVE No.	N/A
Plugin Information:	Publication date: 2011/09/15, Modification date: 2015/03/23
Ports	tcp/0
CVSS Base Score	N/A

分析结果

这个是PCI DSS合规检查，是第三方支付行业的数据安全标准， EulerOS不涉及此漏洞。

7.3 17704 - OpenSSH S/KEY Authentication Account Enumeration

漏洞描述

Name	17704 - OpenSSH S/KEY Authentication Account Enumeration
Description	OpenSSH 4.6 and earlier, when ChallengeResponseAuthentication is enabled, allows remote attackers to determine the existence of user accounts by attempting to authenticate via S/KEY, which displays a different response if the user account exists, a similar issue to CVE-2001-1483.
CVE No.	CVE-2007-2243
Base Score	5.0

分析结果

EulerOS中的OpenSSH不包含S/KEY支持，不受此漏洞影响。

7.4 17705 - OPIE w/ OpenSSH Account Enumeration

漏洞描述

Name	17705 - OPIE w/ OpenSSH Account Enumeration
Description	OpenSSH, when using OPIE (One-Time Passwords in Everything) for PAM, allows remote attackers to determine the existence of certain user accounts, which displays a different response if the user account exists and is configured to use one-time passwords (OTP), a similar issue to CVE-2007-2243.
CVE No.	CVE-2007-2768
Base Score	4.3

分析结果

EulerOS中的PAM不包含OPIE，不受此漏洞影响。

7.5 17744 - OpenSSH >= 2.3.0 AllowTcpForwarding Port Bouncing

漏洞描述

Name	17744 - OpenSSH >= 2.3.0 AllowTcpForwarding Port Bouncing
Description	The default configuration for OpenSSH enables AllowTcpForwarding, which could allow remote authenticated users to perform a port bounce, when configured with an anonymous access program such as AnonCVS.
CVE No.	CVE-2004-1653
Base Score	6.4

分析结果

EulerOS默认设置AllowTcpForwarding为no，不受此漏洞影响。

7.6 78655 - OpenSSH SSHFP Record Verification Weakness

漏洞描述

Name	78655 - OpenSSH SSHFP Record Verification Weakness
Description	It was discovered that OpenSSH clients did not correctly verify DNS SSHFP records. A malicious server could use this flaw to force a connecting client to skip the DNS SSHFP record check and require the user to perform manual host verification of the DNS SSHFP record.
CVE No.	CVE-2014-2653
Base Score	4.3

分析结果

EulerOS已经在openssh-6.6.1p1-11修复，此处为误报。

7.7 86328 - SSH Diffie-Hellman Modulus <= 1024 Bits (Logjam)

漏洞描述

Name	86328 - SSH Diffie-Hellman Modulus <= 1024 Bits (Logjam)
Description	A flaw was found in the way the TLS protocol composes the Diffie-Hellman exchange (for both export and non-export grade cipher suites). An attacker could use this flaw to downgrade a DHE connection to use export-grade key sizes, which could then be broken by sufficient pre-computation. This can lead to a passive man-in-the-middle attack in which the attacker is able to decrypt all traffic.
CVE No.	CVE-2015-4000
Base Score	4.3

分析结果

已经在openssh-6.6.1p1-25.4修复，此处为误报。

7.8 85382 - OpenSSH < 7.0 Multiple Vulnerabilities

漏洞描述

Name	85382 - OpenSSH < 7.0 Multiple Vulnerabilities
Description	The kbdint_next_device function in auth2-chall.c in sshd in OpenSSH through 6.9 does not properly restrict the processing of keyboard-interactive devices within a single connection, which makes it easier for remote attackers to conduct brute-force attacks or cause a denial of service (CPU consumption) via a long and duplicative list in the ssh -oKbdInteractiveDevices option, as demonstrated by a modified client that provides a different password for each pam element on this list.
CVE No.	CVE-2015-5600
Base Score	8.5

分析结果

EulerOS已经在openssh-6.6.1p1-22修复，此处为误报。

7.9 84638 - OpenSSH < 6.9 Multiple Vulnerabilities

漏洞描述

Name	84638 - OpenSSH < 6.9 Multiple Vulnerabilities
Description	The x11_open_helper function in channels.c in ssh in OpenSSH before 6.9, when ForwardX11Trusted mode is not used, lacks a check of the refusal deadline for X connections, which makes it easier for remote attackers to bypass intended access restrictions via a connection outside of the permitted time window.
CVE No.	CVE-2015-5352
Base Score	4.9

分析结果

EulerOS自带xorg-x11-server软件禁用了XSECURITY扩展，也就是OpenSSH无法实施不可信转发，无安全风险，因此不涉及。

7.10 85690 - OpenSSH < 7.1 PermitRootLogin Security Bypass

漏洞描述

85690 - OpenSSH < 7.1 PermitRootLogin Security Bypass

分析结果

EulerOS已设置PermitRootLogin no，此处为误报。

7.11 90022 - OpenSSH < 7.2 Untrusted X11 Forwarding Fallback Security Bypass

漏洞描述

Name	90022 - OpenSSH < 7.2 Untrusted X11 Forwarding Fallback Security Bypass

Description	According to its banner, the version of OpenSSH running on the remote host is prior to 7.2. It is, therefore, affected by a security bypass vulnerability due to a flaw in ssh(1) that is triggered when it falls back from untrusted X11 forwarding to trusted forwarding when the SECURITY extension is disabled by the X server. This can result in untrusted X11 connections that can be exploited by a remote attacker.
CVE No.	CVE-2016-1908
Base Score	4.3

分析结果

EulerOS已经在openssh-6.6.1p1-25.2修复，此处为误报。

7.12 90023 - OpenSSH < 7.2p2 X11Forwarding xauth Command Injection

漏洞描述

Name	90023 - OpenSSH < 7.2p2 X11Forwarding xauth Command Injection
Description	According to its banner, the version of OpenSSH running on the remote host is prior to 7.2p2. It is, therefore, affected by a security bypass vulnerability due to improper sanitization of X11 authentication credentials. An authenticated, remote attacker can exploit this, via crafted credentials, to inject arbitrary xauth commands, resulting in gaining read and write access to arbitrary files, connecting to local ports, or performing further attacks on xauth itself. Note that exploiting this vulnerability requires X11Forwarding to have been enabled.
CVE No.	CVE-2016-3115
Base Score	4.9

分析结果

EulerOS已经在openssh-6.6.1p1-25.2修复，此处为误报。

7.13 900104 - SFTP cd / Command Privilege Escalation

漏洞描述

Name	900104 - SFTP cd / Command Privilege Escalation

Description	Applicable OS: Linux/SunOS/AIX/FreeBSD/HP-UX Dependent Service Credential: SSH Check Method: step1: Check the remote server to see whether it provide SSH service. step2: Use correct username and password to login the SFTP server. step3: Try to switch the directory to root directory ' cd / ', and get the file list. If succeed then means there is vulnerability. Vulnerability Detail: SFTP service should provide limited privilege for the users. User is not allowed to get the privilege of the root directory. Vulnerability Source: ICSL Ticket
CVE No.	N/A
CVSS Base Score	6.0

分析结果

本检查项为判断SFTP用户是否具有根目录读写权限。

如果扫描过程中所配置的用户为“SFTP专用用户”，则判断为问题，需要进行修改，限制该用户的目录权限。若为其他类型用户， 请结合业务中该用户的权限需要，手工检查是否该用户有必要访问到根目录，若无必要，则需限制该用户的目录权限；若有必要，例如某些高权限的执行系统命令的用户，则请忽略该漏洞报出。

EulerOS本身不承载具体业务，不存在“SFTP专用用户”，不涉及此漏洞。

产品需要根据实际业务场景进行分析，如果存在SFTP专用用户”，需限制其向上跨目录访问，只能访问指定目录下的文件，配置方法见**2.2.1.1 加固SSH服务**。

7.14 900208 - The private key encryption check

漏洞描述

Name	900208 - The private key encryption check
Description	Applicable OS: Linux/SunOS/AIX/FreeBSD/HP-UX Dependent Service Credential: SSH Check Method: step1: Login the system. step2: Find certificate files in specified directory '/opt' and '/export'. File type: 'cer|crt|der|pem'. step3: Finde key fils ,File type : '.key | _key' . step3: Check each file to see whether its an un-encrpyted private Key . Vulnerability Detail: Private Key which is not encrpyted is considered to be unsafe. Vulnerability Source: ICSL Ticket/Red Line 2.0 The private key of the certificate is not encrypted, does not conform to safety regulations. The private key of the certificate is not encrypted : File: /etc/ssh/ssh_host_rsa_key File: /etc/ssh/ssh_host_ecdsa_key File: /etc/ssh/ssh_host_ed25519_key
CVE No.	NA

Base Score	4.0

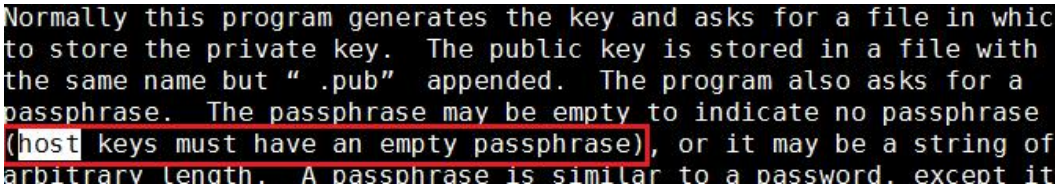
分析结果

EulerOS使用开源OpenSSH开源软件，默认情况下/etc/ssh/目录下存放SSH协议使用的HostKey，具体如下：

表 7-1 SSH 协议与 HostKey 文件对应关系

SSH协议	HostKey文件
SSH v1	/etc/ssh/ssh_host_key
SSH v2	/etc/ssh/ssh_host_dsa_key
SSH v2	/etc/ssh/ssh_host_ecdsa_key
SSH v2	/etc/ssh/ssh_host_ed25519_key
SSH v2	/etc/ssh/ssh_host_rsa_key

OpenSSH软件明确要求HostKey不能加密，如下：



因此此项检查对OpenSSH软件使用的HostKey不适用，非问题。

7.15 OpenSSH <=6.6 SFTP misconfiguration exploit for 64bit Linux - Remote

漏洞描述

Name	OpenSSH <=6.6 SFTP misconfiguration exploit for 64bit Linux - Remote
Description	A user would be able to access all parts of the filesystem that he has access to - including procfs if given SFTP access on a server that is misconfigured and didn't use ChrootDirectory.
CVE No.	N/A
Base Score	N/A

分析结果

OpenSSH <=6.6 SFTP misconfiguration exploit for 64bit Linux - Remote插件仅针对专用sftp帐号需要配置，EulerOS不存在专用sftp用户，不涉及此漏洞。

7.16 Openssh MaxAuthTries 限制绕过漏洞

漏洞描述

Name	Openssh MaxAuthTries限制绕过漏洞
Description	The kbdint_next_device function in auth2-chall.c in sshd in OpenSSH through 6.9 does not properly restrict the processing of keyboard-interactive devices within a single connection, which makes it easier for remote attackers to conduct brute-force attacks or cause a denial of service (CPU consumption) via a long and duplicative list in the ssh -oKbdInteractiveDevices option, as demonstrated by a modified client that provides a different password for each pam element on this list.
CVE No.	CVE-2015-5600
Base Score	8.5

分析结果

EulerOS已经在openssh-6.6.1p1-22修复，此处为误报。

7.17 OpenSSH roaming_common.c 堆缓冲区溢出漏洞

漏洞描述

Name	OpenSSH roaming_common.c堆缓冲区溢出漏洞
Description	A buffer overflow flaw was found in the way the OpenSSH client roaming feature was implemented. A malicious server could potentially use this flaw to execute arbitrary code on a successfully authenticated OpenSSH client if that client used certain non-default configuration options.
CVE No.	CVE-2016-0778
Base Score	5.1

分析结果

EulerOS已经在openssh-6.6.1p1-25修复，此处为误报。

7.18 OpenSSH sshd mm_answer_pam_free_ctx 释放后重利用漏洞

漏洞描述

Name	OpenSSH sshd mm_answer_pam_free_ctx释放后重利用漏洞
Description	A use-after-free flaw was found in OpenSSH. An attacker able to fully compromise a non-privileged pre-authentication process using a different flaw could possibly cause sshd to crash or execute arbitrary code with root privileges.
CVE No.	CVE-2015-6564
Base Score	4

分析结果

EulerOS已经在openssh-6.6.1p1-22修复，此处为误报。

7.19 OpenSSH 'x11_open_helper()'函数安全限制绕过漏洞

漏洞描述

Name	OpenSSH 'x11_open_helper()'函数安全限制绕过漏洞
Description	The x11_open_helper function in channels.c in ssh in OpenSSH before 6.9, when ForwardX11Trusted mode is not used, lacks a check of the refusal deadline for X connections, which makes it easier for remote attackers to bypass intended access restrictions via a connection outside of the permitted time window.
CVE No.	CVE-2015-5352
Base Score	4.9

分析结果

EulerOS自带xorg-x11-server软件禁用了XSECURITY扩展，也就是OpenSSH无法实施不可信转发，无安全风险，因此不涉及。

7.20 CVE-2015-6565

漏洞描述

Name	CVE-2015-6565
Description	sshd in OpenSSH 6.8 and 6.9 uses world-writable permissions for TTY devices, which allows local users to cause a denial of service (terminal disruption) or possibly have unspecified other impact by writing to a device, as demonstrated by writing an escape sequence.
CVE No.	CVE-2015-6565
Base Score	7.2

分析结果

该漏洞影响的版本为openssh 6.8和6.9版本，EulerOS使用的6.6p1，不受该漏洞影响。

7.21 93194 - OpenSSH < 7.3 Multiple Vulnerabilities

漏洞描述

Name	93194 - OpenSSH < 7.3 Multiple Vulnerabilities
Description	According to its banner, the version of OpenSSH running on the remote host is prior to 7.3. It is, therefore, affected by multiple vulnerabilities: - A flaw exists that is due to the program returning shorter response times for authentication requests with overly long passwords for invalid users than for valid users. This may allow a remote attacker to conduct a timing attack and enumerate valid usernames. (CVE-2016-6210) - A denial of service vulnerability exists in the auth_password() function in auth-passwd.c due to a failure to limit password lengths for password authentication. An unauthenticated, remote attacker can exploit this, via a long string, to consume excessive CPU resources, resulting in a denial of service condition. (CVE-2016-6515)
CVE No.	CVE-2016-6210 CVE-2016-6515
Base Score	8.5

分析结果

EulerOS openssh默认使用unix_chkpwd验证口令的有效性，会对口令长度进行限制不受此处所列的漏洞的影响，此处为误报。

7.22 96151 - OpenSSH < 7.4 Multiple Vulnerabilities

漏洞描述

Name	OpenSSH < 7.4 Multiple Vulnerabilities
Description	- A flaw exists in ssh-agent due to loading PKCS#11 modules from paths that are outside a trusted whitelist.A local attacker can exploit this, by using a crafted request to load hostile modules via agent forwarding, to execute arbitrary code. To exploit this vulnerability, the attacker would need to control the forwarded agent-socket (on the host running the sshd server) and the ability to write to the file system of the host running ssh-agent.（CVE-2016-10009） - A flaw exists in sshd due to creating forwarded Unix-domain sockets with 'root' privileges whenever privilege separation is disabled. A local attacker can exploit this to gain elevated privileges.(CVE-2016-10010) - authfile.c in sshd in OpenSSH before 7.4 does not properly consider the effects of realloc on buffer contents, which might allow local users to obtain sensitive private-key information by leveraging access to a privilege-separated child process.（CVE-2016-10011） - A flaw exists in sshd within the shared memory manager used by pre-authenticating compression support due to a bounds check being elided by some optimizing compilers and due to the memory manager being incorrectly accessible when pre-authenticating compression is disabled. A local attacker can exploit this to gain elevated privileges.(CVE-2016-10012)
CVE No.	CVE-2016-10009 CVE-2016-10010 CVE-2016-10011 CVE-2016-10012

分析结果

- CVE-2016-10009：已经在penssh-6.6.1p1-28.h7版本修复。
- CVE-2016-10010：EulerOS使用的openssh，基于CentOS 7中openssh6.6源码基线，centos从openssh-3.3/3.3p1开始启用了特权分离，而且 sshd socket转发是比较新的功能，在openssh6.6版本还未集成，因此不涉及此漏洞。
- CVE-2016-10011：已经在openssh-6.6.1p1-28.h8中修复，此处为误报。
- CVE-2016-10012：已经在openssh-6.6.1p1-28.h11中修复，此处为误报。

