

# Table of Contents

阿里云安全部署指南	1.1
第1章 云环境安全最佳实践	1.2
1-1 阿里云账号最佳安全实践	1.2.1
1-2 ECS安全部署方法	1.2.2
1-3 ECS Linux 系统 CPU 异常占用: minerd 、tplink 等挖矿进程	1.2.3
1-4 阿里云自定义镜像安全建议	1.2.4
1-5 安装Telnet Client	1.2.5
1-6 升级ECS OpenSSL	1.2.6
1-7 查看操作系统用户列表	1.2.7
1-8 加密勒索病毒处理方案	1.2.8
1-9 加密勒索实践防护方案	1.2.9
1-10 WannaCry一键解密和修复工具	1.2.10
1-11 ECS反射型DDoS攻击解决方法	1.2.11
1-12 获取访客真实 IP : ASP、PHP、ASPX、JSP	1.2.12
1-13 暴力破解攻击和防御	1.2.13
1-14 管理后台弱口令漏洞	1.2.14
1-15 阿里云DDoS攻击缓解最佳实践	1.2.15
第2章 操作系统安全加固	1.3
2-1 Windows操作系统安全加固	1.3.1
2-2 Linux操作系统加固	1.3.2
2-3 NFS服务安全加固	1.3.3
2-4 Rsync 服务安全加固	1.3.4
2-5 如何在 Windows 和 Windows Server 中启用/禁用 SMBv1、SMBv2 和 SMBv3	1.3.5
第3章 应用软件安全加固	1.4
3-1 FTP匿名登录或弱口令漏洞及服务加固	1.4.1
3-2 Docker服务安全加固	1.4.2
3-3 Jenkins服务安全加固	1.4.3
3-4 Kubernetes服务安全加固	1.4.4
3-5 Hadoop环境安全加固	1.4.5
3-6 FileZilla FTP Server 安全加固	1.4.6
3-7 Elasticsearch服务安全加固	1.4.7
3-8 phpMyadmin 服务安全加固	1.4.8
第4章 数据库服务安全加固	1.5
4-1 MongoDB服务安全加固	1.5.1
4-2 Memcached服务安全加固	1.5.2

4-3 Redis服务安全加固	1.5.3
4-4 MySQL服务安全加固	1.5.4
4-5 预防数据库勒索事件	1.5.5
第5章 语言运行环境安全加固	1.6
5-1 PHP环境安全加固	1.6.1
第6章 WEB应用安全加固	1.7
6-1 Apache服务安全加固	1.7.1
6-2 Tomcat服务安全加固	1.7.2
6-3 网站被植入WebShell的解决方案	1.7.3
第7章 Web应用安全漏洞	1.8
7-1 Web漏洞含义解释	1.8.1
7-2 挂马攻击和防御	1.8.2
7-3 URL跳转漏洞	1.8.3
7-4 CRLF HTTP头部注入漏洞	1.8.4
7-5 任意文件下载漏洞	1.8.5
7-6 域名未设置SPF解析记录	1.8.6
7-7 系统弱口令	1.8.7
7-8 后门文件漏洞	1.8.8
7-9 文件包含漏洞	1.8.9
7-10 SQL注入漏洞	1.8.10
7-11 网络钓鱼攻击和防御	1.8.11
7-12 越权漏洞	1.8.12
7-13 Crossdomain.xml配置不当	1.8.13
7-14 文件上传漏洞	1.8.14
7-15 应用越权漏洞	1.8.15
7-16 SEO暗链	1.8.16
7-17 目录遍历攻击	1.8.17
7-18 网站备份文件泄露	1.8.18
7-19 代码执行漏洞	1.8.19
7-20 跨站攻击	1.8.20
7-21 DNS区域传送漏洞	1.8.21
7-22 信息泄露漏洞	1.8.22

# 阿里云云安全部署指南

- 阿里云账号最佳安全实践
- ECS安全部署方法
- ECS Linux 系统 CPU 异常占用：minerd、tplink 等挖矿进程
- 阿里云自定义镜像安全建议
- 安装 Telnet Client
- 升级 ECS OpenSSL
- 查看操作系统用户列表
- 加密勒索病毒处理方案
- 加密勒索事件防护方案
- WannaCry一键解密和修复工具
- ECS反射型DDoS攻击解决方法
- 获取访客真实 IP：ASP、PHP、ASPx、JSP
- 暴力破解攻击和防御
- 管理后台弱口令漏洞
- 阿里云DDoS攻击缓解最佳实践

本文介绍了阿里云账号管理方面的最佳安全实践，主要从密钥（AccessKey，简称 AK）管理和访问控制（RAM）两方面提供相关实践建议。

## AccessKey 泄露处置指南

AccessKey 是阿里云颁发给用户的一种身份凭证，用于在 API 调用进行身份验证。AccessKey 相当于获取用户云资源的钥匙，一旦泄露，将带来云资源泄露以及被恶意利用等风险。建议您定期自查内部 AccessKey 是否存在泄漏。

如您发现包含 AccessKey 的敏感信息已在公网泄露，请尽快删除已泄露的代码或信息，并登录到阿里云控制台上禁用或删除 AccessKey。操作步骤如下：

1. 登录到 [阿里云控制台](#)，并在屏幕右上角 用户菜单 下单击 accesskeys。
2. 在 AccessKey 管理 页面，单击被泄露 AccessKey 操作 列表下的 禁用 或 删除。

## 云账号安全实践

- 尽量不要使用 Github 类代码托管服务。特殊情况下，一定要使用的话，建议您自建私有仓库，或搭建企业内部代码托管系统，以防敏感信息泄露，确保代码安全。
- 采用云上安全产品进行预警、检测，如阿里云提供的免费版云盾 [态势感知](#)。态势感知能够检测到您系统账号的安全漏洞，您可登录到云盾控制台免费开通该服务，并开启自动检测功能。
- 启用阿里云权限管理机制，包括 [访问控制](#)（Resource AccessManagement，简称 RAM）和 安全凭证管理（SecurityToken Service，简称 STS。根据需求使用不同权限的子账号来访问云资源（如 OSS），或为用户提供访问的临时授权。
- 遵循 [RAM 最佳实践](#)，从登录验证、账号授权、权限分配等方面配置 RAM，有效地使用 RAM 进行用户身份管理和资源访问控制。主要的访问控制策略包括：
  - 为主账号和 RAM 用户启用 MFA
  - 为用户登录配置强密码策略
  - 定期轮转用户登录密码和访问密钥
  - 遵循最小授权原则
  - 使用策略限制条件
  - 及时撤销用户不再需要的权限
  - 不要为主账号创建访问密钥
  - 使用群组给 RAM 用户分配权限
  - 将用户管理、权限管理与资源管理分离
  - 将控制台用户与 API 用户分离
- 遵循 OSS 安全实践，包括：
  - 不使用主账号访问 OSS
  - 读写分离
  - Bucket 权限隔离
  - 使用 STS 的临时凭证来访问 OSS

具体请参考 [阿里云 OSS Android SDK 开发文档](#)。

在企业内建立安全制度，开展必要的安全意识培训等工作，提升全员安全意识。

## 更多信息

- 安全上云实践
- 阿里云访问控制 RAM
- 阿里云 OSS 权限管理
- 阿里云 OSS Android SDK

# 操作系统安全加固

## 1.重置ECS实例。

注意：强烈建议您重置ECS实例前，备份您的数据。ECS实例重置完成后，对数据进行杀毒后再上传至ECS实例。

- i. 登录ECS管理控制台，单击实例。
- ii. 选择您的ECS实例，在更多下拉菜单中单击停止（如果您的实例当前为启动状态）。
- iii. 实例停止后，在更多下拉菜单中单击重新初始化磁盘。
- iv. 重新初始化磁盘完成后，操作系统将恢复回新购时的状态。

实例ID/名称	监控	所在可用区	IP地址	状态(全部)	网络类型(全部)	配置	付费方式(全部)	操作
1	青岛可用区B	运行中	CPU: 1核 内存: 1024 MB 带宽: 1Mbps	包年包月 16-01-11 00:00到期	经典网络	管理 变配 续费 更多		
2	青岛可用区B	已过期	CPU: 1核 内存: 1024 MB 带宽: 1Mbps	包年包月 还有2天保	经典网络	启动 停止		
3	青岛可用区B	运行中	CPU: 1核 内存: 2048 MB 带宽: 1Mbps	包年包月 16-09-16	经典网络	重启 重置密码 修改信息 连接管理终端...		
4	青岛可用区B	运行中	CPU: 1核 内存: 1024 MB 带宽: 1Mbps	包年包月 16-01-03	经典网络	修改帮助 重新初始化磁盘		
5	青岛可用区B	运行中	CPU: 1核 内存: 1024 MB 带宽: 1Mbps	包年包月 16-06-17	经典网络	更换系统盘 购买相同配置		
6	青岛可用区B	运行中	CPU: 1核 内存: 1024 MB 带宽: 1Mbps	包年包月 16-06-17	经典网络	编辑标签 安全组配置		
7	青岛可用区B	运行中	CPU: 1核 内存: 512 MB 带宽: 1Mbps	包年包月 16-03-15	经典网络	修改私网IP		

## 2.主机登录安全设置。

- i. 建议修改默认的登录端口（RDP、SSH）为其它端口。
- ii. 建议使用证书登录，设置可信登录主机的IP。
- iii. 如果您需要使用密码登录，请务必使用复杂密码（字母+数字+特殊符号，包含大小写，并保证10位及以上字符）。
- iv. 登录用户的权限使用普通用户权限，需要管理员权限时再进行提权。（Windows使用runas，linux使用sudo）

更多操作系统加固方法，请查看：

- [Windows操作系统安全加固](#)
- [Linux操作系统加固](#)

# 应用服务软件安全部署

## 常见Web应用安全部署设置

1.WDCP、TOMCAT、Apache、Nginx、Jenkins、PHPMyAdmin、WebLogic、Jboss等Web服务管理后台不要使用默认密码或空密码，务必使用复杂密码（字母+数字+特殊符号，包含大小写，并保证10位及以上字符）；不使用的管理后台建议直接关闭，否则黑客有可能直接控制您的ECS服务器。

2.保证Web应用升级到最新版本，如Struts、ElasticSearch非最新版都爆发过远程命令执行漏洞。务必及时更新Web应用版本，否则黑客有可能直接控制您的ECS服务器。

3.Redis、Memcached、MongoDB如设置无密码访问，可导致黑客直接远程登录并控制您的服务器。请务必设置为有密码访问，并使用复杂密码。另外，建议修改端口并设置绑定监听IP为127.0.0.1。

## 常见数据库应用安全部署设置

- Postgresql、Oracle、MySQL、SQLServer等默认连接端口建议修改为非常用端口。
- 根据不同的角色创建不同的账号，并精细化授权。切忌共享使用账号或使用系统账号登录数据库。
- 数据库连接密码使用复杂密码（字母+数字+特殊符号，包含大小写，并保证**10位及以上字符**）。

## 其它

- 请及时升级各种应用，以免安全漏洞被黑客利用入侵服务器。
- 安装云盾安全防护软件，做好杀毒防护工作。

# 说明

本文提到的挖矿程序排查场景，仅为技术人员提供故障排查思路，不保证与攻击者实际使用方式一致，具体场景以实际情况为准。

## 问题描述

云服务器 ECS Linux 服务器上 CPU 使用率超过 70%，严重时可达到 100%，或者服务器响应越来越慢。

## 原因分析

### 恶意 **miner**、**tplink** 进程

在服务器上运行 top 命令，结果如下：

```
[root@... /]# top
top - 18:04:11 up 59 min, 5 users, load average: 3.63, 3.58, 3.28
Tasks: 173 total, 2 running, 171 sleeping, 0 stopped, 0 zombie
Cpu(s): 23.6%us, 28.1%sy, 25.8%ni, 0.1%id, 0.0%wa, 0.0%hi, 0.5%si, 21.9%st
Mem: 8057972k total, 1675620k used, 6382352k free, 49256k buffers
Swap: 0k total, 0k used, 0k free, 382572k cached

      PID USER      PR  NI    VIRT    RES    SHR    S %CPU %MEM     TIME+   COMMAND
 1906 root      20   0  384m  86m 1112 S 64.5   1.1  57:22.76 miner
 144 root      20   0 36624 1972 1152 S 62.5   0.0  22:47.04 plymouthd
15203 git      20   0 94560 23m 4016 R 26.6   0.3  0:00.80 ruby
 1855 root      20   0 94660  924  468 S 9.6   0.0  4:54.19 k111
 2006 root      20   0  740  212  628 S 0.7   0.1  0:07.28 -
```

可以看到，有一个 **miner**（或 **tplink**）的异常进程，占用了大量 CPU 资源。该进程是服务器被入侵后，被恶意安装的比特币挖矿程序，一般存在于 `/tmp/` 目录下。

如果使用 top 命令查看不到所述进程，可以用 ps 命令检查相关进程。例如，

```
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root     1072  141  6.2 852688 504664 ?        S1   Nov30 17:14:05 /tmp/minerd -a scrypt -o 31.41.40.25:9327 -u LNzyZEbAfZDGwb3Cca13qjbcKJ2JfqTTkk:1
```

可以看到，服务器中存在这个进程。如果它不是您主动开启的，则很可能是被入侵所致。服务器被恶意利用来挖比特币。

## 隐藏的恶意模块

黑客通过驱动 rootkit 程序入侵主机，并部署隐藏挖矿程序，CPU 使用率可能达到 90-100%。该场景无法通过 top 命令和 ps 命令来检测确认。

## 处理方案

### 恶意 **miner**、**tplink** 进程

- 使用如下命令，通过 PID 获取对应文件的路径。然后，找到并删除对应的文件。

```
ls -l /proc/$PID/exe
```

其中，\$PID 为进程对应的 PID 号，可以通过 ps 或者 top 获取。

- 使用 kill 命令关闭进程。
- 建议您平时增强服务器的安全维护，优化代码，以避免因程序漏洞等导致服务器被入侵。

## 隐藏的恶意模块

被隐藏的恶意模块一般有：raid.ko、iptable\_mac.ko、snd\_pcs.ko、usb\_pcs.ko 和 ipv6\_kac.ko。您可以使用 file /lib/udev/usb\_control/... 命令，分别检查是否存在以上模块。

例如，使用以下命令查看是否存在 iptable\_mac.ko 模块：

```
file /lib/udev/usb_control/iptable_mac.ko
```

结果如下图所示，表明存在隐藏的 iptable\_mac.ko 模块。

```
[root@172.31.1.10 ~]# file /lib/udev/usb_control/iptable_mac.ko
/lib/udev/usb_control/iptable_mac.ko: ELF 64-bit LSB relocatable, x86-64, version 1 (SYSV), BuildID[sha1]=8582d597272300bb32eac5fe755579e6e98c0be1, not stripped
```

阿里云自定义镜像主要用于创建 ECS 实例，操作系统、及已经预安装的应用程序和数据，都可以通过自定义镜像自动复制到新实例中。您可以通过准备好的自定义镜像方便地创建具有相同配置环境的实例，从而提高工作交付效率。

通过阿里云自定义镜像创建的实例与通过正常官方镜像创建的实例一样，在创建过程中，可能会存在不同层面的操作系统自身的漏洞问题，如远程命令执行高危漏洞（导致 NSA 工具受影响的 Windows 0day 漏洞）、应用安全漏洞（弱口令、管理后台信息泄露、SQL 代码注入漏洞、Struts2 高危漏洞）。如果在镜像创建前能解决掉这些安全问题，将会提高您业务的安全性。阿里云安全团队提供以下安全最佳实践帮助您解决自定义镜像安全问题：

## 1. 操作系统漏洞

- 在使用官方标准镜像的自定义镜像后，建议您时刻关注 [安全漏洞情报](#)，当出现高风险漏洞时，及时更新操作系统所有补丁，并重新创建自定义镜像。
- 对于高危漏洞但暂时无法更新补丁的情况，建议您使用安全组访问控制策略、应用防护策略对该服务器进行实时检测、防御，防止被黑客成功入侵。

## 2. 软件配置加固

- 对于已自定义安装的应用服务软件（如 Tomcat、Apache、Nginx 等软件），建议使用官方最新版的软件，并对应用软件进行安全加固，禁止不必要的功能或组件，提高整体安全能力。您可参考阿里云安全团队提供的相关安全加固文档对应用服务进行安全加固。
- 定期关注安全漏洞情况，一旦发现高危漏洞，应及时更新到最新版本。

## 3. 上线前最后一步

- 在您已完成前两步后，建议您使用 [安全扫描工具](#)（例如，操作系统漏洞扫描工具：Nessus、Nexpose，Web 漏洞扫描工具：Appscan、WVS）扫描自定义镜像是否仍存在高风险漏洞。如存在安全漏洞，强烈建议您修复完漏洞后再发布使用。

Telnet 是一款功能全面的端口测试工具，常用于网络管理和服务器调试。但是，Windows 操作系统默认未安装该功能。本文介绍了在 Windows 中安装 Telnet Client 的方法。

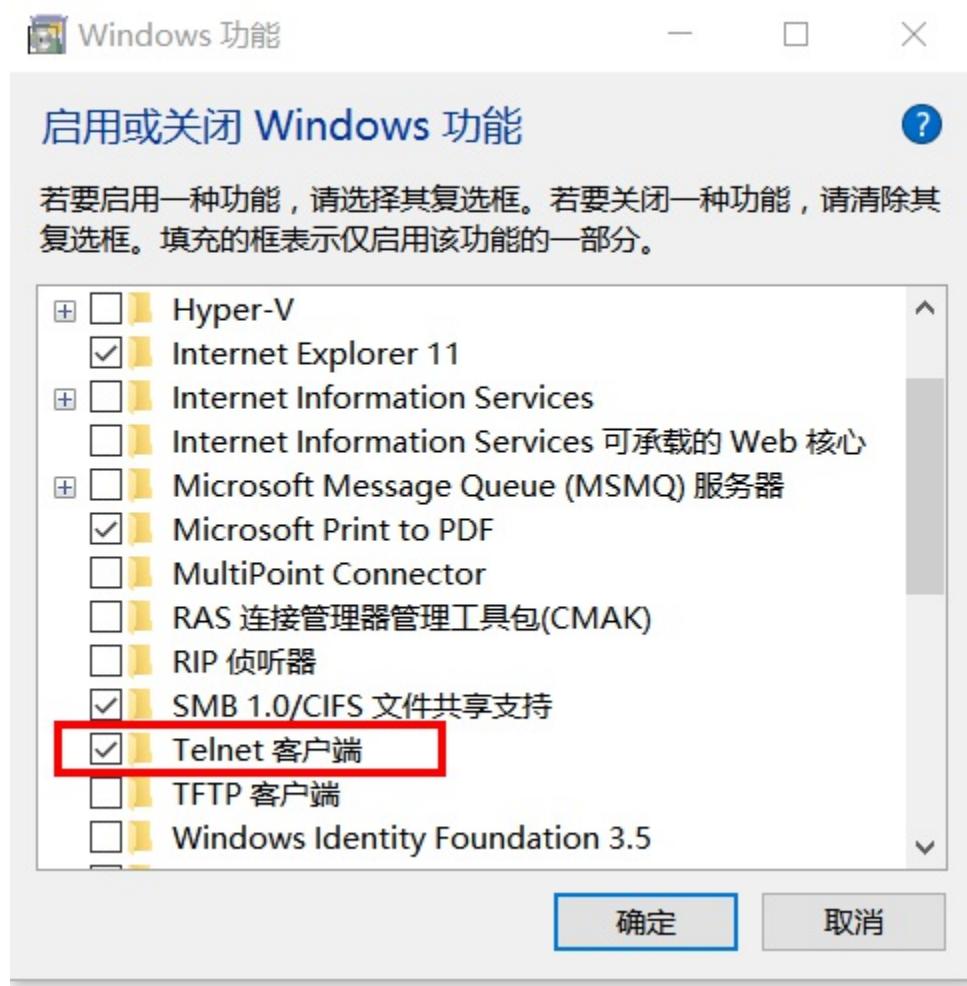
## 使用命令行安装

以管理员的身份打开 cmd 窗口（win + R），执行下面的命令：

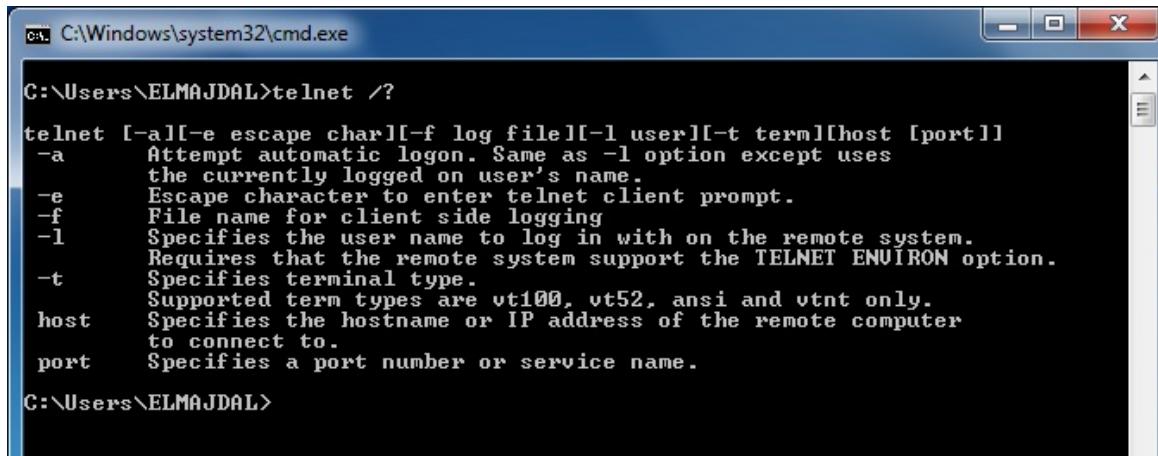
```
dism /online /Enable-Feature /FeatureName:TelnetClient
```

## 在窗口模式下开启

1. 打开 控制面板 > 程序 > 启用或关闭 Windows 功能。
2. 勾选 Telnet 客户端 选项。



3. 单击确定，等待安装完毕。



```
C:\Windows\system32\cmd.exe
C:\Users\ELMAJDAL>telnet /?
telnet [-a][-e escape char][-f log file][-l user][-t term][host [port]]
-a      Attempt automatic logon. Same as -l option except uses
        the currently logged on user's name.
-e      Escape character to enter telnet client prompt.
-f      File name for client side logging
-l      Specifies the user name to log in with on the remote system.
        Requires that the remote system support the TELNET ENVIRON option.
-t      Specifies terminal type.
        Supported term types are vt100, vt52, ansi and vtnt only.
-host   Specifies the hostname or IP address of the remote computer
        to connect to.
-port   Specifies a port number or service name.

C:\Users\ELMAJDAL>
```

为了确保 OpenSSL 的安全，建议云上用户升级 ECS 的 OpenSSL 版本到官方最新版本。本文介绍了具体的升级方法。

## 升级 ECS OpenSSL

连接到 ECS 实例，打开 shell 运行命令行。

## 使用源更新 OpenSSL

对于阿里云的 Linux/CentOS 服务器，以 root 权限运行以下命令：

```
sudo yum update openssl
```

对于 Ubuntu Server/Debian 服务器，以 root 权限运行以下命令：

```
sudo apt-get update  
sudo apt-get upgrade
```

## 使用编译安装更新 OpenSSL

下载最新版本 OpenSSL（以 openssl-1.1.0e 为例）。

注意：以下编译操作存在风险，建议由专业技术人员来操作。

以 root 权限运行以下命令：

```
wget https://www.openssl.org/source/openssl-1.1.0e.tar.gz  
tar zxvf openssl-1.1.0e.tar.gz  
cd openssl-1.1.0e  
../config shared zlib  
make  
make install  
# 替换旧版 OpenSSL  
mv /usr/bin/openssl /usr/bin/openssl.old  
mv /usr/include/openssl /usr/include/openssl.old  
ln -s /usr/local/bin/openssl /usr/bin/openssl  
ln -s /usr/local/include/openssl/ /usr/include/openssl
```

## 检查 OpenSSL 版本

以 root 权限运行 openssl version -a 命令，系统会返回 OpenSSL 版本信息，如下所示。

```
OpenSSL 1.1.0e 16 Feb 2017  
built on: reproducible build, date unspecified  
platform: linux-x86_64  
compiler: gcc -DZLIB -DDSO_DLFCN -DHAVE_DLFCN_H -DNDEBUG -DOPENSSL_THREADS -DOPENSSL_NO_STATIC_ENGINE -DOPENSSL_PIC -  
DOPENSSL_IA32_SSE2 -DOPENSSL_BN_ASM_MONT -DOPENSSL_BN_ASM_MONT5 -DOPENSSL_BN_ASM_GF2m -DSHA1_ASM -DSHA256_ASM -DSHA51  
2_ASM -DRC4_ASM -DMD5_ASM -DAES_ASM -DVPAES_ASM -DBSAES_ASM -DGHASH_ASM -DECP_NISTZ256_ASM -DPADLOCK_ASM -DPOLY1305_A  
SM -DOPENSSLDIR="/usr/local/ssl"" -DENGINESDIR="/usr/local/lib/engines-1.1"" -Wa,--noexecstack  
OPENSSLDIR: "/usr/local/ssl"
```

```
ENGINESEDIR: "/usr/local/lib/engines-1.1"
```

了解更多：[OpenSSL 官方漏洞信息公告](#)。

查看操作系统用户列表：

- Windows 系统，在命令行下执行 net user，回车。

```
Microsoft Windows [版本 6.1.7601]
C:\Users>net user
\\ALI-074000N 的用户帐户

Administrator          Guest
命令成功完成。
```

- Linux 系统，在命令行下执行 cat /etc/passwd，回车。

```
[root@f20 html]# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
polkitd:x:999:998:User for polkitd:/:/sbin/nologin
tss:x:59:59:Account used by the trousers package to sandbox the tcscd daemon:/dev/null:/sbin/nologin
avahi-autoipd:x:170:170:Avahi IPv4LL Stack:/var/lib/avahi-autoipd:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
mysql:x:27:27:MySQL Server:/var/lib/mysql:/sbin/nologin
usbmuxd:x:113:113:usbmuxd user:/:/sbin/nologin
geoclue:x:998:997:User for geoclue:/var/lib/geoclue:/sbin/nologin
abrt:x:173:173::/etc/abrt:/sbin/nologin
```

勒索软件是一种恶意软件（木马或其他类型的病毒），它能锁定用户设备或加密用户文件，然后通知用户必须支付赎金才能拿回自己的数据。所要求赎金一般很高昂，且不能保证一定可以成功解密。

如果在您的系统中存在文件被恶意加密或者有勒索信息，说明该系统感染了勒索软件。目前已知的勒索软件多数是通过邮件进行传播，本文提供了相应的解决方案。

## 拒绝交付赎金

如果您的业务不幸遭遇密勒索软件，建议您不要向勒索者交付赎金以恢复数据。

## 恢复数据

在文件被恶意加密时，您可以尝试通过之前的备份恢复数据。如果没有额外的数据备份，您可以使用安全厂商发布的工具集尝试解密数据。

注意：随着加密勒索软件对抗的不断升级，加密勒索软件可以在很短的时间内出现变种，使用工具不一定保障数据能够恢复成功。

## 解密工具

作为安全厂商，卡巴斯基实验室破解了世界上大部分勒索病毒，更多勒索软件破解器请随时关注：

- [卡巴斯基：勒索软件解密工具集](#)
- [Trendmicro：勒索软件解密方案](#)
- [Avast：勒索软件解密工具集](#)

## XTBL 和 Wallet 加密勒索软件

以下举例说明常见的 Xtbl 和 Wallet 加密勒索软件。

病毒名称：XTBL（可以解密），Wallet（暂时无法解密）

病毒类型：勒索病毒

勒索动机：黑客勒索受利益驱使，不接受现金，使用 bitcoin 进行勒索交易。

利用手法：使用 AES 或 RSA 算法，批量加密上百种后缀文件类型或者应用程序，并且把原来的文件名后缀为：  
xxxxx@aol.com、xxxxx@india.com.wallet。

危险等级：高危

入侵手段：远程控制协议漏洞（RDP 弱口令）、远程密码泄露，或其他升级演变的入侵方式。

病毒特征：黑客会在所有被加密文件上留下其联系方式。

- Xtbl 病毒感染样例：

名称	修改日期	类型	大小
mego.mdf.id-74948504.centurion_legion@aol.com.xtbl	2016/9/12 20:42	Xtbl 文件	28,417 KB
megolog.ldf.id-74948504.centurion_legion@aol.com.xtbl	2016/9/12 20:42	Xtbl 文件	7,681 KB
meg02.mdf.id-74948504.centurion_legion@aol.com.xtbl	2016/9/12 20:42	Xtbl 文件	3,841 KB
meg02_log.ldf.id-74948504.centurion_legion@aol.com.xtbl	2016/9/12 20:42	Xtbl 文件	1,025 KB
model.mdf.id-74948504.centurion_legion@aol.com.xtbl	2016/9/12 20:42	Xtbl 文件	3,073 KB

- Wallet 病毒感染样例：

20160921145744369.pdf.[amagnus@india.com].wallet	27/11/2016 18:49	72 KB
20160922102331345.pdf.[amagnus@india.com].wallet	27/11/2016 18:49	191 KB
20160922102736460.pdf.[amagnus@india.com].wallet	27/11/2016 18:49	191 KB
20160923114259166.pdf.[amagnus@india.com].wallet	27/11/2016 18:49	356 KB
20160923114944265.pdf.[amagnus@india.com].wallet	27/11/2016 18:49	216 KB
20160924175751586.pdf.[amagnus@india.com].wallet	27/11/2016 18:49	222 KB

破解工具：[Xtbl 解密工具](#)

## 数据恢复安全规范

- 建议重新部署系统或回滚之前正常状态的快照，并对操作系统和业务代码进行安全加固。
- 修改所有管理端口的账号和密码，并配置强口令。建议使用安全组策略限制访问，并禁止将管理端口和管理后台开放到互联网，仅开放必要的业务端口。
- 数据恢复完毕后，立即对 ECS 配置定期快照策略，同时做好异地数据备份工作。
- 对业务作全面的安全检查，及时修复发现的漏洞。如果您没有专业的技术人员，可以选用阿里云云盾 安全管家服务 协助您对业务进行排查和加固。

了解更多：[企业用户应该如何抵御加密勒索事件？](#)

## 数据安全防范措施

- 备份重要文档。备份的最佳做法是采取“3-2-1 规则”，即至少做三个副本，用两种不同格式保存，并将副本放在异地存储。如果有条件，建议全量备份，如数据量过大，可以选择不定期的实时全量和增量备份混合方案，推荐您使用快照和 OSS 异地备份方案。
- 如果需要远程管理，可以使用 ECS 安全组来限制访问。把 RDP、SSH、FTP 或其他重要内网管理后台隐藏起来，仅限于内网访问，降低暴露在互联网被攻击的风险。
- 安装企业级防病毒软件。推荐您使用赛门铁克、卡巴斯基商用防病毒软件，Linux 服务器可以安装 Clam。
- 安装最新版本的安全补丁。过时的操作系统或软件相对容易成为勒索软件的攻击目标，因此，定期运行软件及操作系统可强化你电脑的安全性。
- 恶意软件经常将使用远程桌面协议（Remote Desktop Protocol, RDP）的系统作为主要目标。因此，不需要远程访问时就停用 RDP，可以有效阻挡来自恶意软件的攻击。
  - 配置好密码策略。
  - 修改远程控制账户密码，妥善管理密码。
  - 定期更新管理员账号名称和密码的更新策略。
- 提高安全意识，做好数据安全防护措施。

了解更多：[加密勒索事件防护方案](#)

越来越多的用户受加密勒索事件困扰。加密勒索软件利用系统漏洞，成功入侵用户业务服务器对全盘数据进行加密勒索，导致用户业务突然中断、数据泄露和数据丢失，带来严重业务风险。

本文分析了导致加密勒索病毒发生的不安全因素，并提供了相应防护方案，帮助您防护云服务器，远离加密勒索软件。

## 不安全来源

通过对云上用户的调查分析，大部分用户未按照最佳的安全使用方式来使用云服务器资源，主要问题有：

- 关键账号存在弱口令或无认证机制。
  - 服务器关键账号（root、administrator）密码简单或无密码。
  - 数据库（Redis、MongoDB、MySQL、MSsql Server）等重要业务使用弱密码或无密码。
- 无访问控制策略，业务暴露在互联网上。RDP、SSH、Redis、MongoDB、MySQL、MSsql Server 等高危服务可以通过互联网直接访问。
- 服务器操作系统和软件存在高危漏洞。恶意攻击者可以利用服务器操作系统和应用服务软件存在的高危漏洞，上传加密勒索软件或执行勒索操作，实现远程攻击。

以上漏洞的利用成本较低，经常被黑客使用来发动数据库删除等勒索攻击，攻击者不需要获取账号密码就可以对业务造成重创。

## 安全防护方案

为使云上用户尽量少受加密勒索软件影响，我们建议您参照并部署以下安全防护方案。

### 数据备份与恢复

可靠的数据备份可以将勒索软件带来的损失最小化，但您也要对这些数据备份进行安全防护，避免数据感染和损坏。

- 建议您至少备份两份数据：本地备份和异地备份。
- 也建议您采用多种不同的备份方式，以确保在发生勒索事件后，尽可能地挽回损失。

推荐您使用以下方法：

- ECS快照功能，具体操作请参考 [ECS快照使用配置手册](#)
- RDS提供的数据备份功能，具体操作请参考 [RDS数据备份功能配置手册](#)
- 使用OSS存储服务备份重要数据文件
- 由用户制定的数据备份策略或方案，如自建数据库MySQL备份

关键业务账号安全策略

- 阿里云主账号

阿里云为您分配的账号是所有云上业务的“关键钥匙”，一旦您所拥有的最高权限的“钥匙”泄露，黑客将从根本上掌握支撑云上业务的所有云服务资源，从而将直接威胁整体的云上业务安全。

阿里云为您提供账号登录多因素验证机制（MFA）、密码安全策略，和审计功能，您可以在控制台方便地启用和设置以上功能，确保云服务账号安全。

针对组织内部多角色场景，企业需要使用**RAM**服务为不同角色合理分配账号并授权，以防止在运维管理活动中，出现意外操作而带来安全风险。

- 业务最高权限账号

当您选用了云服务器，并在其中部署了关键业务（例如：数据库服务、文件服务、缓存等与数据强相关的核心重要服务），这些服务的最高管理员账号的安全是保证业务持续可靠运行的必要条件。

您需要妥善设置好账号和密码。

- 不要将这些高危服务公开在互联网上，您可以参见 强化网络访问控制 部分配置强访问控制策略。
- 启用认证鉴权功能。
- 禁止使用**root**账号直接登录。
- 如果您使用的是**Windows**系统，建议您修改 **administrator** 默认名称。
- 为所有服务配置强密码。强密码要求至少8个字符以上，包含大小写字母、数字、特殊符号，不包含用户名、真实姓名或公司名称，不包含完整的单词。

推荐工具：阿里云访问控制**RAM**服务、系统和软件加固。

具体操作请参考 [RAM服务配置手册](#)。

## 强化网络访问控制

精细化的网络管理是业务的第一道屏障。很多企业的网络安全架构缺少业务分区分段，一旦遭遇入侵，其影响面往往是全局的。在这种情况下，通过有效的安全区域划分、访问控制和准入机制可以防止或减缓渗透范围，阻止不必要的人员进入业务环境。

例如，您可以限制**SSH**、**RDP**等管理协议，并对**FTP**、**Redis**、**MongoDB**、**Memcached**、**MySQL**、**MSSQL-Server**、**Oracle**等数据相关服务的连接源IP进行访问控制，实现最小化访问范围，仅允许受信IP地址访问，并对出口网络行为实时分析和审计。

推荐您 使用安全的**VPC**网络。

- 通过**VPC**和安全组，划分不同安全等级的业务区域，让不同的业务处在不同的隔离空间。
- 配置入口/出口过滤安全组防火墙策略，同时在入口和出口进行过滤。例如，
  - 常用的数据库服务不需要在互联网直接管理或访问，可以通过配置入方向的访问控制策略防止数据库服务暴露在互联网上被黑客利用。
  - 也可以配置更严格的内网访问控制策略，例如：在内网入方向配置仅允许内网某IP访问内网的某台数据库服务器。

推荐工具：[VPC网络](#)、[安全组策略](#)

## 搭建具有容灾能力的基础架构

高性能、具有冗余的基础架构能力是保障业务强固的基础条件。在云环境下，您可以使用**SLB**集群搭建高可用架构。当某一个节点发生紧急问题，高可用架构可以无缝切换至备用节点，既防止业务中断，也防止数据丢失。

在资源允许的条件下，企业或组织可以搭建同城或异地容灾备份系统，当主系统出现发生勒索事件后，可以快速切换到备份系统，从而保证业务的连续性。

推荐工具：阿里云**SLB**、阿里云**RDS**等高性能服务组合而成的容灾架构

## 定期进行外部端口扫描

端口扫描可以用来检验企业的弱点暴露情况。如果企业有一些服务连接到互联网，需要确定哪些业务是必须要发布到互联网上，哪些仅需要内部访问。公开到互联网的服务数量越少，攻击者的攻击范围就越窄，从而遭受的安全风险就越小。

推荐工具：[阿里云云盾安全管家服务](#)

## 定期进行安全测试

企业IT管理人员需要定期对业务软件资产进行安全漏洞探测，一旦确定有公开暴露的服务，应使用漏洞扫描工具对其进行扫描，尽快修复扫描发现的漏洞。同时，日常也应该不定期关注软件厂商发布的安全漏洞信息和补丁信息，及时做好漏洞修复管理工作。

推荐工具：[VPC网络](#)、[安全组](#)、[阿里云云盾安全管家服务](#)、主机系统和服务软件安全加固

## 基础安全运维

- 制定并实施IT软件安全配置，对操作系统（Windows、Linux）和软件（FTP、Apache、Nginx、Tomcat、Mysql、MS-Sql Server、Redis、MongoDB、Memcached等服务）初始化安全加固，并定期核查其有效性。
- 为Windows操作系统云服务器安装防病毒软件，并定期更新病毒库。
- 定期更新补丁。
- 修改 administrator 默认名称，为登录账号配置强口令。
- 开启日志记录功能，集中管理日志和审计分析。
- 合理地分配账号、授权，并开启审计功能，例如：为服务器、RDS数据库建立不同权限账号并启用审计功能；如果有条件，可以实施类似堡垒机、VPN等更严格的访问策略。
- 实施强密码策略，并定期更新维护，对于所有操作行为严格记录并审计。
- 对所有业务关键点进行实时监控，当发现异常时，立即介入处理。

推荐工具：[阿里云云盾安骑士](#)

## 应用系统代码安全

大部分安全问题来自程序代码不严谨，代码安全直接影响到业务风险。根据经验，代码层的安全需要程序员从一开始就将安全架构设计纳入到整体软件工程内，按照标准的软件开发流程，在每个环节内关联安全因素。

对于一般企业，需要重点关注开发人员或软件服务提供上的安全编码和安全测试结果，尤其是对开发完成的业务代码进行代码审计评估和上线后的黑盒测试（也可以不定期地进行黑盒渗透测试）。

推荐工具：[阿里云云盾先知计划](#)、[阿里云云盾web应用防火墙\(WAF\)](#)、SDL标准流程

## 建立全局的外部威胁和情报感知能力

安全是动态对抗的过程，在安全事件发生之前，就要时刻了解和识别外部不同类型的风险。做安全的思路应该从防止安全入侵这种不可能的任务转到了防止损失这一系列关键任务上。防范措施必不可少，但是基于预警、响应的时间差也同样关键。而实现这种快速精准的预警能力需要对外面的信息了如指掌，所以建立有效的监控和感知体系是实现安全管控措施是不可少的环节，更是安全防护体系策略落地的基础条件。

您可以登录阿里云控制台，到云盾菜单中免费开通阿里云态势感知服务，查看实时的外部攻击行为和内部漏洞（弱点）情况。

推荐工具：[阿里云云盾态势感知](#)、[大数据安全分析平台](#)

## 建立安全事件应急响应流程和预案

在安全攻防的动态过程中，建议您为突发的安全事件准备好应急策略。在安全事件发生后，要通过组织快速响应、标准化的应急响应流程、规范的事件处置规范来降低安全事件带来的损失。

推荐工具：可管理的安全服务（MSS）、安全事件应急响应服务

## 更多参考

请参考以下相关加固文档：

[Windows操作系统加固手册](#) [Linux操作系统加固手册](#) [FTP服务加固手册](#) [MySQL服务加固手册](#) [Redis服务加固手册](#)  
[MongoDB服务加固手册](#) [Memcached服务加固手册](#)

# **WannaCry 勒索病毒**

WannaCry（又名 Wanna Decryptor），是一种“蠕虫式”勒索病毒软件。WannaCry 勒索病毒在全球范围内爆发，至少150个国家、30万名用户收到影响，已造成损失达80亿美元。

WannaCry 利用 Windows 操作系统445端口存在的漏洞进行传播，并具有自我复制、主动传播的特性。被该勒索病毒入侵后，用户主机系统内的照片、图片、文档、音频、视频等几乎所有类型的文件都将被加密，加密文件的后缀名被统一修改为“. WCRY”，并在桌面弹出勒索对话框，要求受害者支付比特币。

## **WannaCry 解密修复工具**

阿里云安全团队经过分析研究，找到 WannaCry 加密勒索病毒的解密方式，发布针对 WannaCry 勒索病毒的一键解密和修复工具。经反复测试验证，该工具可以恢复已被 WannaCry 勒索病毒加密的文件。

### **前提条件**

感染 WannaCry 勒索病毒后，未重启操作系统。

### **适用范围**

该工具适用于 Windows 云服务器和本地服务器，支持的操作系统版本包括：Windows Server 2003、Windows Server 2008。

### **修复步骤**

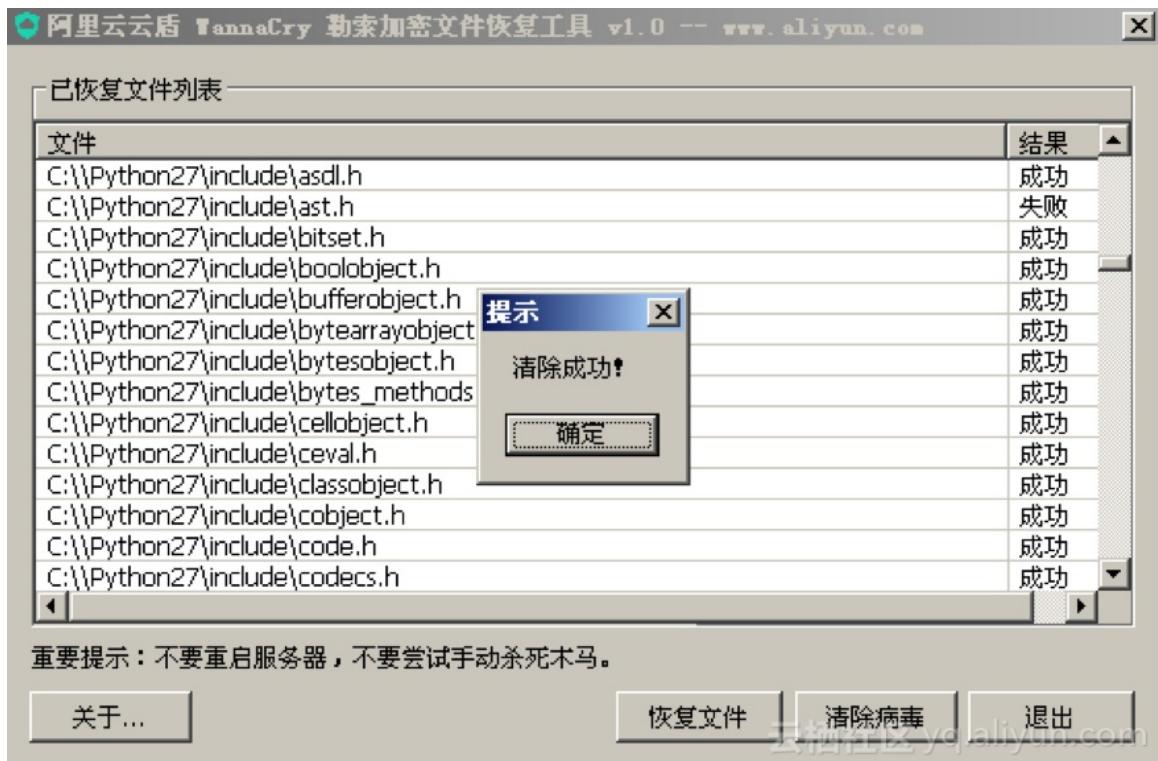
1. 单击 WannaCry 修复工具，将修复工具下载到被感染的 Windows 服务器或 PC 机上。
2. 双击 WannaCry-Decrypt-Tool.exe 文件，运行修复工具。



3. 单击恢复文件，执行文件恢复功能。执行时间较长，请耐心等待。



4. 单击清除病毒。执行时间较长，请耐心等待。



#### 注意事项

- 大多数情况下，被加密的文件可以被成功恢复。但可能因内存数据被二次写入，覆盖原有加密状态时的数据，导致数据恢复不成功。解密和修复文件失败，不会对操作系统造成任何影响。
- 阿里云安全团队强烈建议，在感染 WannaCry 勒索病毒后，不要关闭或重启操作系统，也不要手工查杀病毒，建议优先使用该修复工具尝试恢复数据。
- 该修复工具针对 WannaCry 勒索病毒加密方式研发，Windows 系统均可使用。

# 问题现象

由于某些服务配置不当，导致服务器被黑客利用进行DDoS攻击。具体表现为机器对外带宽占满；使用抓包工具检测，可看到大量同一源端口的包对外发出。

# 解决方案

## Linux系统

### 1. 加固NTP服务

1. 通过Iptables配置只允许信任的IP访问本机UDP的123端口。

```
修改配置文件，然后执行以下命令：  
echo "disable monitor" >> /etc/ntp.conf  
执行以下命令重启NTP服务：  
service ntpd restart
```

2. 我们建议您直接关闭掉NTP服务，并禁止其开机自启动。

```
执行service ntpd stop命令。  
执行chkconfig ntpd off命令。
```

### 2. 加固Chargen服务

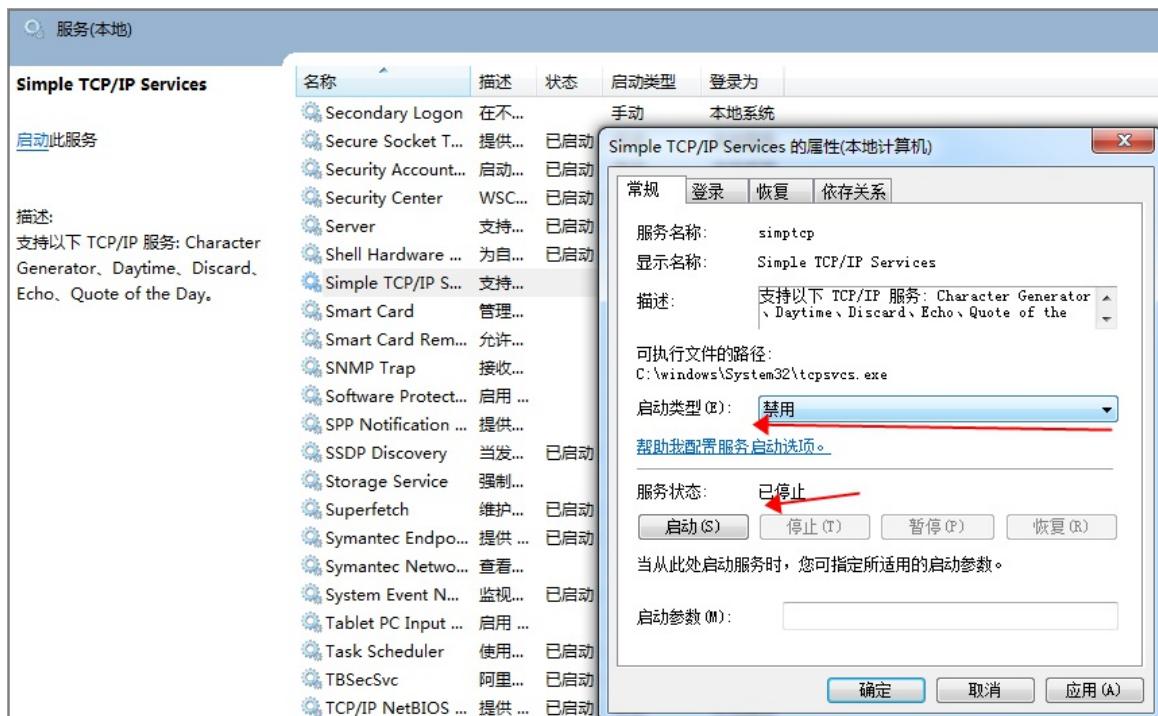
1. 通过Iptables配置只允许信任的IP访问本机UDP的19端口。
2. 我们建议您直接关闭掉chargen服务。编辑配置文件"/etc/inetd.conf"，用#号注释掉chargen服务，然后重启inetd服务。

## Windows系统

### 1. 加固Simple TCP/IP服务

注意： Windows系统默认不安装Simple TCP/IP服务，如果您无需使用此服务，可跳过此步骤。

1. 通过防火墙配置，只允许信任的IP访问本机UDP、TCP的19、17端口。
2. 我们建议您直接关闭Simple TCP/IP服务，并禁止自启动。



## 2. Web应用的加固

### Wordpress的Pingback

1. 您可以通过增加Wordpress插件来防止Pingback被利用，加入如下过滤器：

```
add_filter( 'xmlrpc_methods', function( $methods ) {
    unset( $methods['pingback.ping'] );
    return $methods;
} );
```

2. 建议您直接删除xmlrpc.php文件。

在您的网站接入 Web 应用防火墙（WAF）防护后，您的网站接收到的请求来源 IP 将会是 WAF 的 IP 地址。

如果您的网站有获取真实的访客 IP 的需求，请参考以下方法：

## ASP

```
Request.ServerVariables("HTTP_X_REAL_IP")
```

## ASP.NET(C#)

```
Request.ServerVariables["HTTP_X_REAL_IP"]
```

## PHP

```
$_SERVER["HTTP_X_REAL_IP"]
```

## JSP

```
request.getHeader("HTTP_X_REAL_IP")
```

# 什么是暴力破解攻击

暴力破解攻击是指攻击者通过系统地组合并尝试所有的可能性以破解用户的用户名、密码等敏感信息。攻击者往往借助自动化脚本工具来发动暴力破解攻击。

## 攻击行为类型

根据暴力破解的穷举方式，其攻击行为可以分为：

- 字典攻击法。大多攻击者并没有高性能的破解算法和CPU/GPU，为节省时间和提高效率，会利用社会工程学或其它方式建立破译字典，使用字典中已存在的用户名、密码进行猜破。
- 穷举法。攻击者首先列出密码组合的可能性（如数字、大写字母、小写字母、特殊字符等），然后按密码长度从1位、2位....构成不同的账号和密码对，然后逐个猜试。该方法需要高性能的破解算法和CPU/GPU作支持。
- 组合式攻击法。使用字典攻击和穷举法的组合攻击方式。

理论上，只要拥有性能足够强的计算机和足够长的时间，大多密码均可以被破解出来。

## 攻击业务类型

- 针对Windows操作系统的远程桌面管理协议（RDP）、Linux操作系统的管理协议（SSH）的暴力破解攻击
- 针对具有登录认证机制的软件服务（如Mysql、SQLserver、FTP、Web前后端登录接口等应用服务）的暴力破解攻击

对于防御者而言，给攻击者留得时间越长，其组合出正确的用户名和密码的可能性就越大。因此，时间在检测暴力破解攻击时很重要。

## 暴力破解攻击有什么危害

通过自动化工具发起的暴力破解攻击可以获取用户账号和密码。

## 如何防御暴力破解攻击

- 制定密码复杂度策略，并进行服务加固。密码的长度要大于 8 位，且最好大于 20 位；密码应由数字、大小写字母和特殊符号混合组成；密码的最长有效期为 90 天。
- 配置好网络访问控制。严格限制将高危服务管理端口直接发布到互联网；建议您使用 VPN 和堡垒机的方式集中管理和审计。
- 提高内部全员安全意识，禁止借用或共享使用账号。



## 漏洞描述

管理后台存在弱口令，可导致攻击者直接访问网站后台，并进一步入侵网站。

## 修复方案

在管理页面修改访问密码，建议使用 10 位以上数字 + 字母 + 特殊符号的强密码。

注意：在修改前请做好备份，或为 ECS 建立硬盘快照。

# DDoS攻击简介

分布式拒绝服务攻击（DDoS攻击）是一种针对目标系统的恶意网络攻击行为，DDoS攻击经常会导致被攻击者的业务无法正常访问，也就是所谓的拒绝服务。

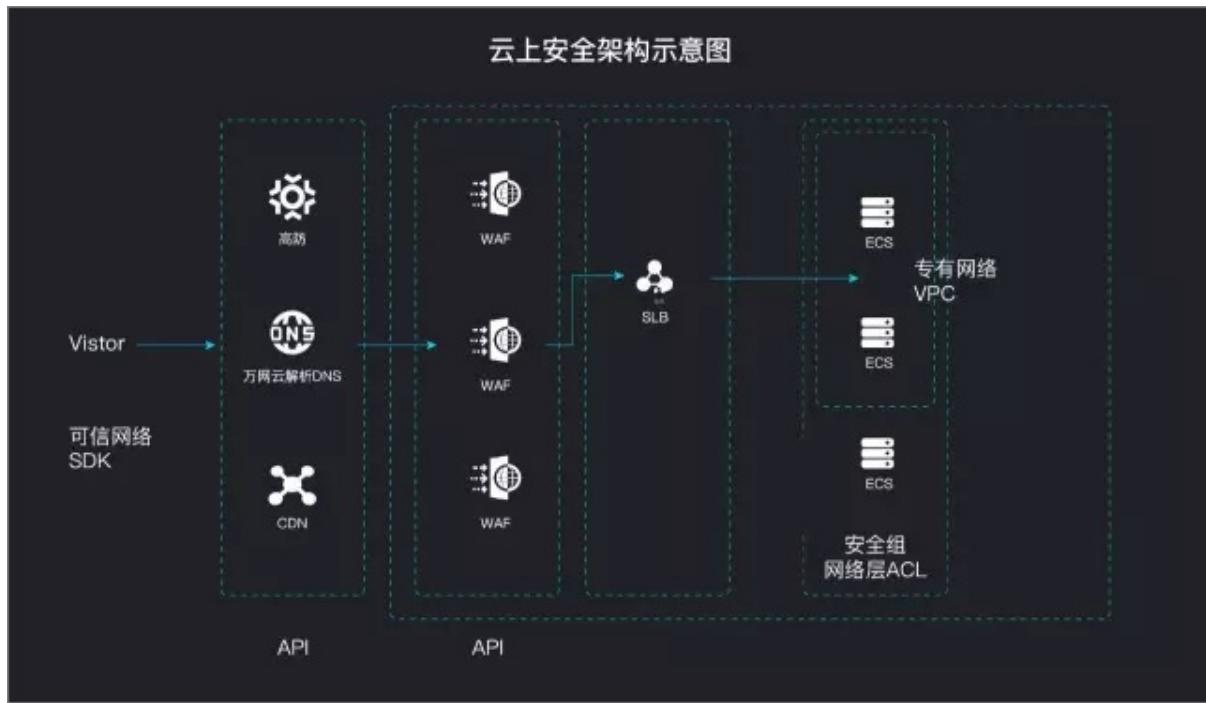
常见的DDoS攻击包括以下几类：

- 网络层攻击：比较典型的攻击类型是**UDP反射攻击**，这类攻击主要利用大流量拥塞被攻击者的网络带宽，导致被攻击者的业务无法正常响应客户访问。
- 连接层攻击：比较典型的攻击类型包括**SYN flood**攻击、连接数攻击等，这类攻击通过占用服务器的连接池资源从而达到拒绝服务的目的。
- 会话层攻击：比较典型的攻击类型是**SSL连接攻击**，这类攻击占用服务器的SSL会话资源从而达到拒绝服务的目的。
- 应用层攻击：比较典型的攻击类型包括**DNS flood**攻击、**HTTP flood**攻击、游戏假人攻击等，这类攻击占用服务器的应用处理资源极大的消耗服务器处理性能从而达到拒绝服务的目的。

# DDoS攻击缓解最佳实践

建议阿里云用户从以下几个方面着手缓解DDoS攻击的威胁：

1. 缩小暴露面，隔离资源和不相关的业务，降低被攻击的风险。
2. 优化业务架构，利用公共云的特性设计弹性伸缩和灾备切换的系统。
3. 服务器安全加固，提升服务器自身的连接数等性能。
4. 做好业务监控和应急响应。
5. 选择合适的商业安全方案。阿里云既提供了免费的基础DDoS防护，也提供了**BGP防护包**、**高防IP**、**游戏盾**等商业安全方案，用户也可以选择其他厂商的安全方案。



## 1. 缩小暴露面

### 1.1 配置安全组

尽量避免将非业务必须的端口暴露在公网上，从而避免与业务无关的请求和访问。通过配置安全组可以有效防止系统被扫描或者意外暴露。

关于安全组的详细介绍，请查看[安全组使用指南](#)。

### 1.2 使用专有网络（VPC）

通过专有网络VPC实现网络内部逻辑隔离，防止来自内网肉鸡的攻击。

关于专有网络VPC的详细介绍，请查看[专有网络VPC使用指南](#)。

## 2. 优化业务架构

### 2.1 部署负载均衡

通过部署负载均衡（SLB）实例来负载多台服务器的方式，可以有效缓解一定流量范围内的连接层DDoS攻击。

同时，部署负载均衡方案后，也能将用户访问流量均匀分配到各个服务器上，减少单台服务器的负担，加快访问速度。

关于负载均衡（SLB）的详细介绍，请查看[负载均衡使用指南](#)。

### 2.2 部署弹性伸缩

弹性伸缩（Auto Scaling），是根据用户的业务需求和策略，经济地自动调整弹性计算资源的管理服务。通过部署弹性伸缩，系统可以有效的缓解会话层和应用层攻击，在遭受攻击时自动增加服务器，提升处理性能，避免业务遭受严重影响。

关于弹性伸缩的详细介绍，请查看[弹性伸缩使用指南](#)。

## 2.3 部署DNS智能解析

通过智能解析的方式优化DNS解析，可以有效避免DNS流量攻击产生的风险。同时，建议您将业务托管至多家DNS服务商。

- 屏蔽未经请求发送的DNS响应信息
- 丢弃快速重传数据包
- 启用TTL
- 丢弃未知来源的DNS查询请求和响应数据
- 丢弃未经请求或突发的DNS请求
- 启动DNS客户端验证
- 对响应信息进行缓存处理
- 使用ACL的权限
- 利用ACL，BCP38及IP信誉功能

## 2.4 提供余量带宽

通过服务器性能测试，评估正常业务环境下所能承受的带宽和请求数。在购买带宽时确保有一定的余量带宽，可以避免遭受攻击时带宽大于正常使用量而影响正常用户的情况。

# 3. 服务器安全加固

对服务器进行安全加固，减少可被攻击的点，增大攻击方的攻击成本：

- 确保服务器的系统文件是最新的版本，并及时更新系统补丁。
- 对所有服务器主机进行检查，清楚访问者的来源。
- 过滤不必要的服务和端口。例如，对于WWW服务器，只开放80端口，将其他所有端口关闭，或在防火墙上设置阻止策略。
- 限制同时打开的SYN半连接数目，缩短SYN半连接的timeout时间，限制SYN/ICMP流量。
- 仔细检查网络设备和服务器系统的日志。一旦出现漏洞或是时间变更，则说明服务器可能遭到了攻击。
- 限制在防火墙外进行网络文件共享。降低黑客截取系统文件的机会，若黑客以特洛伊木马替换它，文件传输功能将会陷入瘫痪。
- 充分利用网络设备保护网络资源。在配置路由器时应考虑针对流控、包过滤、半连接超时、垃圾包丢弃、来源伪造的数据包丢弃、SYN阀值、禁用ICMP和UDP广播的策略配置。
- 通过iptables之类的软件防火墙限制疑似恶意IP的TCP新建连接，限制疑似恶意IP的连接、传输速率。

# 4. 业务监控和应急响应

## 4.1 关注基础DDoS防护监控

当您的业务遭受DDoS攻击时，基础DDoS默认会通过短信和邮件方式发出告警信息，针对大流量攻击基础DDoS防护也支持电话报警，建议您在接受到告警的第一时间进行应急处理。

关于配置告警消息接收人和语音告警方式，请查看[DDoS基础防护消息接收人设置](#)。

## 4.2 云监控

云监控服务可用于收集、获取阿里云资源的监控指标或用户自定义的监控指标，探测服务的可用性，并支持针对指标设置警报。

关于云监控的详细介绍，请查看[云监控用户指南](#)。

# 5. 商用安全方案

## 5.1 Web应用防火墙（WAF）

针对网站类应用，WAF可以提供针对连接层攻击、会话层攻击和应用层攻击的有效防御。

关于WAF的详细介绍，请查看[WAF用户指南](#)。

## 5.2 DDoS防护包

DDoS防护包为云产品IP提供防御100G以内的DDoS攻击的防护能力，配置简单，即时生效。

关于DDoS防护包的详细介绍，请查看[DDoS防护包用户指南](#)。

## 5.3 高防IP

针对大流量DDoS攻击，建议使用阿里云高防IP服务。

关于高防IP的详细介绍，请查看[DDoS高防IP用户指南](#)。

## 5.4 游戏盾

游戏盾是针对游戏行业常见的DDoS攻击、CC攻击推出的行业解决方案。相比于高防IP服务，游戏盾解决方案的针对性更强，针对游戏行业的攻击防御效果更好、成本更低。

关于游戏盾的详细介绍，请查看[游戏盾用户指南](#)。

# 应当避免的事项

DDoS攻击是业内公认的行业公敌，DDoS攻击不仅影响被攻击者，同时也会对服务商网络的稳定性造成影响，从而对处于同一网络下的其他用户业务也会造成损失。

计算机网络是一个共享环境，需要多方共同维护稳定，部分行为可能会给整体网络和其他租户的网络带来影响，需要您注意：

- 避免使用阿里云产品机制搭建DDoS防护平台
- 避免释放处于黑洞状态的实例
- 避免为处于黑洞状态的服务器连续更换、解绑、增加SLB IP、弹性公网IP、NAT网关等IP类产品
- 避免通过搭建IP池进行防御，避免通过分摊攻击流量到大量IP上进行防御
- 避免利用阿里云非网络安全防御产品（包括但不限于CDN、OSS），前置自身有攻击的业务
- 避免使用多个账号的方式绕过上述规则



- Windows 操作系统安全加固
- Linux 操作系统加固
- NFS 服务安全加固
- Rsync 服务安全加固
- 如何在 Windows 和 Windows Server 中启用/禁用 SMBv1、SMBv2 和 SMBv3

本文档旨在指导系统管理人员或安全检查人员进行Windows操作系统的安全合规性检查和配置。

## 1. 账户管理和认证授权

### 1.1 账户

默认账户安全

- 禁用Guest账户。
- 禁用或删除其他无用账户（建议先禁用账户三个月，待确认没有问题后删除。）

操作步骤

打开 控制面板 > 管理工具 > 计算机管理，在 系统工具 > 本地用户和组 > 用户 中，双击 Guest 帐户，在属性中选中 帐户已禁用，单击 确定。

按照用户分配帐户

按照用户分配帐户。根据业务要求，设定不同的用户和用户组。例如，管理员用户，数据库用户，审计用户，来宾用户等。

操作步骤

打开 控制面板 > 管理工具 > 计算机管理，在 系统工具 > 本地用户和组 中，根据您的业务要求设定不同的用户和用户组，包括管理员用户、数据库用户、审计用户、来宾用户等。

定期检查并删除与无关帐户

定期删除或锁定与设备运行、维护等与工作无关的帐户。

操作步骤

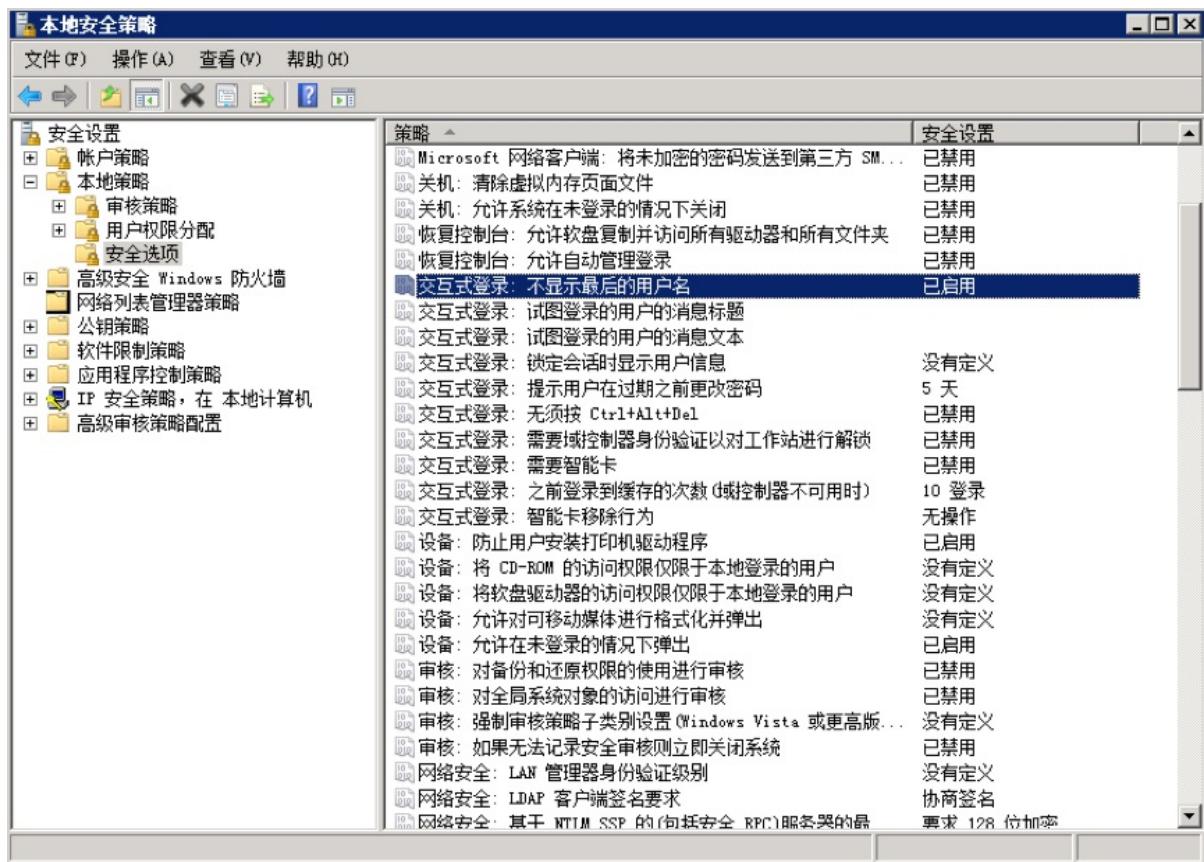
打开 控制面板 > 管理工具 > 计算机管理，在 系统工具 > 本地用户和组 中，删除或锁定与设备运行、维护等与工作无关的帐户。

不显示最后的用户名

配置登录登出后，不显示用户名。

操作步骤：

打开 控制面板 > 管理工具 > 本地安全策略，在 本地策略 > 安全选项 中，双击 交互式登录:不显示最后的用户名，选择 已启用 并单击 确定。



## 1.2 口令

### 密码复杂度

密码复杂度要求必须满足以下策略：

- 最短密码长度要求八个字符。
- 启用本机组策略中密码必须符合复杂性要求的策略。即密码至少包含以下四种类别的字符中的两种：
  - 英语大写字母 A, B, C, ... Z
  - 英语小写字母 a, b, c, ... z
  - 西方阿拉伯数字 0, 1, 2, ... 9
  - 非字母数字字符，如标点符号，@, #, \$, %, &, \*等

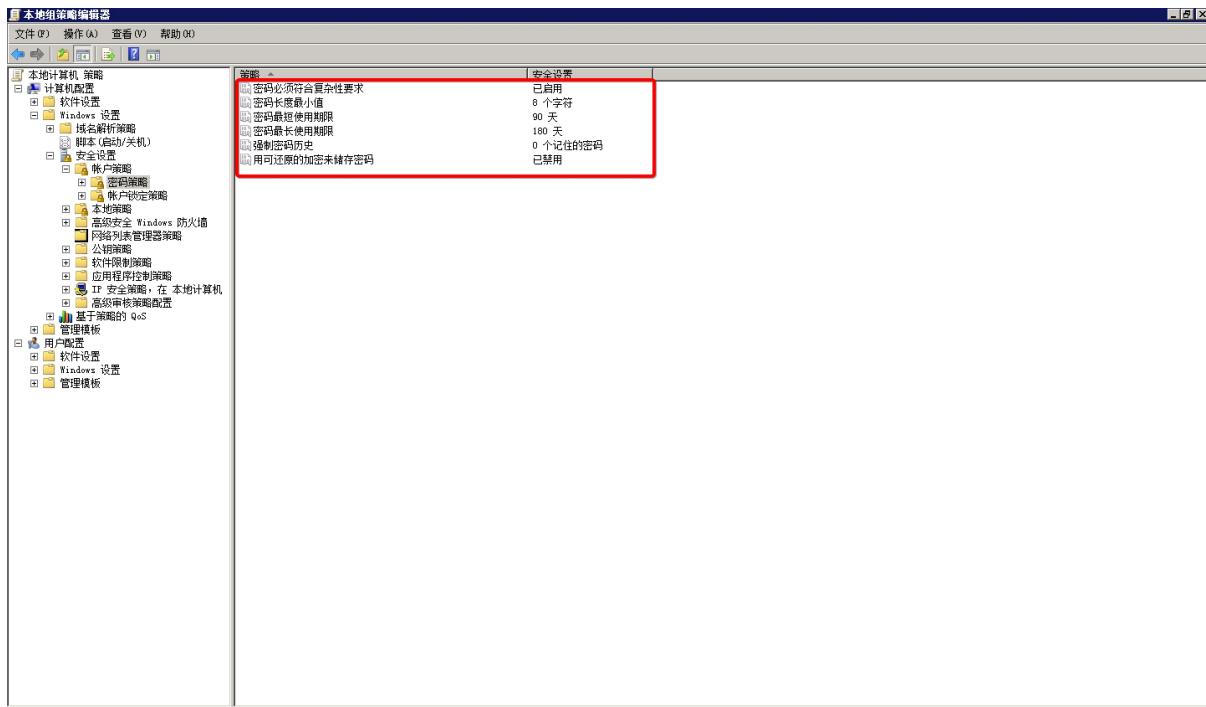
### 操作步骤

打开 控制面板 > 管理工具 > 本地安全策略，在 帐户策略 > 密码策略 中，确认 密码必须符合复杂性要求 策略已启用。

### 密码最长留存期

对于采用静态口令认证技术的设备，帐户口令的留存期不应长于90天。

操作步骤 打开 控制面板 > 管理工具 > 本地安全策略，在 帐户策略 > 密码策略 中，配置 密码最长使用期限 不大于 90天。



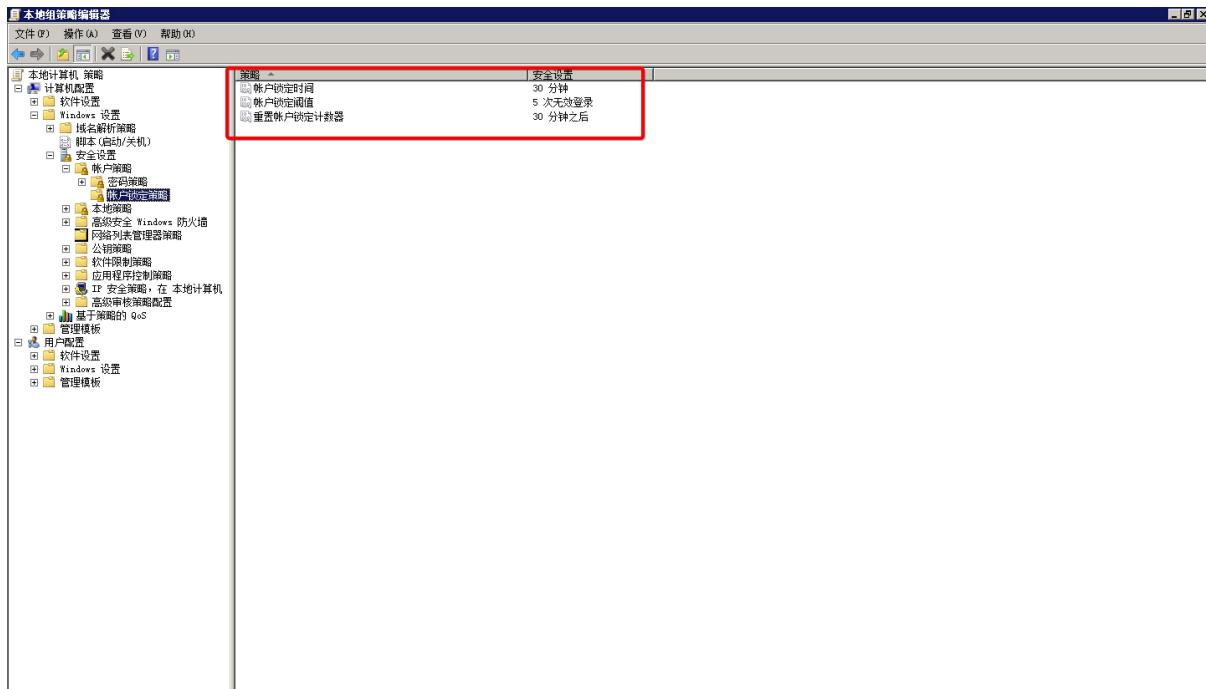
### 帐户锁定策略

对于采用静态口令认证技术的设备，应配置当用户连续认证失败次数超过10次后，锁定该用户使用的帐户。

#### 操作步骤

打开 控制面板 > 管理工具 > 本地安全策略，在 帐户策略 > 帐户锁定策略 中，配置 帐户锁定阈值 不大于10次。

#### 配置样例：



### 1.3 授权

#### 远程关机

在本地安全设置中，从远端系统强制关机权限只分配给**Administrators**组。

#### 操作步骤

打开 控制面板 > 管理工具 > 本地安全策略，在 本地策略 > 用户权限分配 中，配置 从远端系统强制关机 权限只分配给**Administrators**组。

#### 本地关机

在本地安全设置中关闭系统权限只分配给**Administrators**组。

#### 操作步骤

打开 控制面板 > 管理工具 > 本地安全策略，在 本地策略 > 用户权限分配 中，配置 关闭系统 权限只分配给**Administrators**组。

#### 用户权限指派

在本地安全设置中，取得文件或其它对象的所有权权限只分配给**Administrators**组。

#### 操作步骤

打开 控制面板 > 管理工具 > 本地安全策略，在 本地策略 > 用户权限分配 中，配置 取得文件或其它对象的所有权 权限只分配给**Administrators**组。

#### 授权帐户登陆

在本地安全设置中，配置指定授权用户允许本地登陆此计算机。

#### 操作步骤

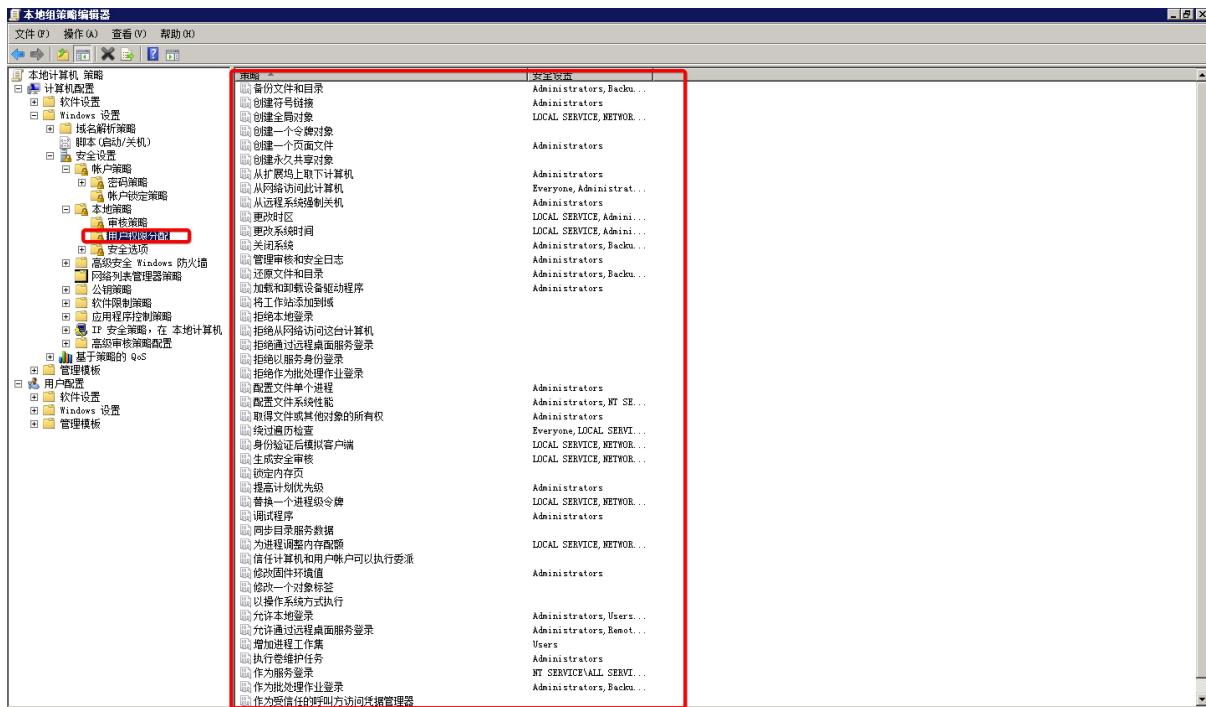
打开 控制面板 > 管理工具 > 本地安全策略，在 本地策略 > 用户权限分配 中，配置 允许本地登录 权限给指定授权 用户。

#### 授权帐户从网络访问

在本地安全设置中，只允许授权帐号从网络访问（包括网络共享等，但不包括终端服务）此计算机。

#### 操作步骤

打开 控制面板 > 管理工具 > 本地安全策略，在 本地策略 > 用户权限分配 中，配置 从网络访问此计算机 权限给指定授权 用户。



## 2. 日志配置操作

### 2.1 日志配置

#### 审核登录

设备应配置日志功能，对用户登录进行记录。记录内容包括用户登录使用的帐户、登录是否成功、登录时间、以及远程登录时、及用户使用的IP地址。

#### 操作步骤

打开 控制面板 > 管理工具 > 本地安全策略，在 本地策略 > 审核策略 中，设置 审核登录事件。

#### 审核策略

启用本地安全策略中对Windows系统的审核策略更改，成功和失败操作都需要审核。

#### 操作步骤

打开 控制面板 > 管理工具 > 本地安全策略，在 本地策略 > 审核策略 中，设置 审核策略更改。

#### 审核对象访问

启用本地安全策略中对Windows系统的审核对象访问，成功和失败操作都需要审核。

#### 操作步骤

打开 控制面板 > 管理工具 > 本地安全策略，在 本地策略 > 审核策略 中，设置 审核对象访问。

#### 审核事件目录服务访问

启用本地安全策略中对Windows系统的审核目录服务访问，仅需要审核失败操作。

#### 操作步骤

打开 控制面板 > 管理工具 > 本地安全策略，在 本地策略 > 审核策略 中，设置 审核目录服务器访问。

## 审核特权使用

启用本地安全策略中对Windows系统的审核特权使用，成功和失败操作都需要审核。

### 操作步骤

打开 控制面板 > 管理工具 > 本地安全策略，在 本地策略 > 审核策略 中，设置 审核特权使用。

## 审核系统事件

启用本地安全策略中对Windows系统的审核系统事件，成功和失败操作都需要审核。

### 操作步骤

打开 控制面板 > 管理工具 > 本地安全策略，在 本地策略 > 审核策略 中，设置 审核系统事件。

## 审核帐户管理

启用本地安全策略中对Windows系统的审核帐户管理，成功和失败操作都要审核。

### 操作步骤

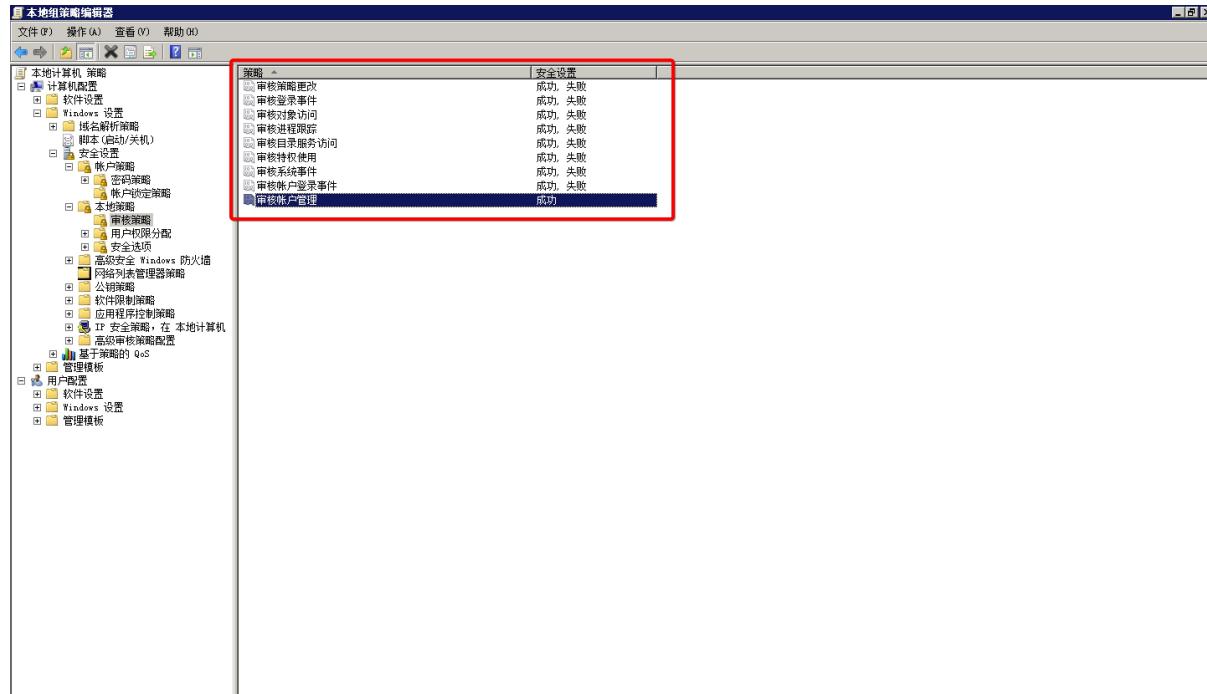
打开 控制面板 > 管理工具 > 本地安全策略，在 本地策略 > 审核策略 中，设置 审核帐户管理。

## 审核过程追踪

启用本地安全策略中对Windows系统的审核进程追踪，仅失败操作需要审核。

### 操作步骤

打开 控制面板 > 管理工具 > 本地安全策略，在 本地策略 > 审核策略 中，设置 审核进程追踪。

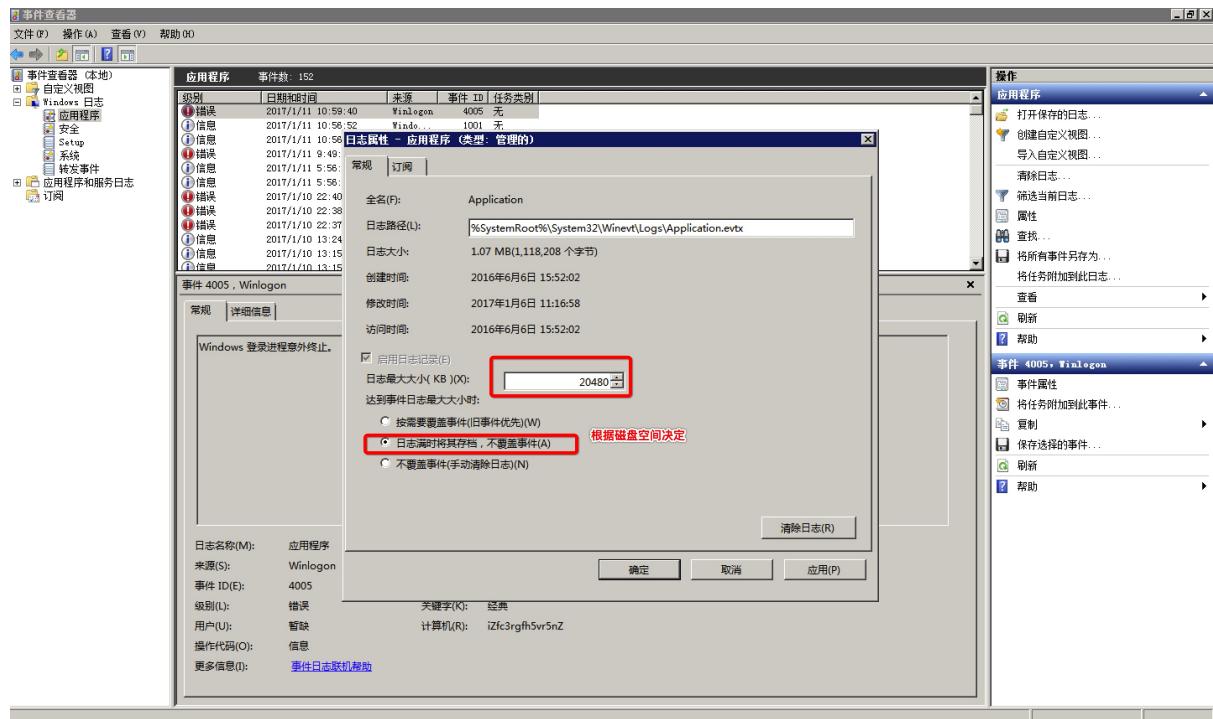


## 日志文件大小

设置应用日志文件大小至少为 8192 KB，可根据磁盘空间配置日志文件大小，记录的日志越多越好。并设置当达到最大的日志尺寸时，按需要轮询记录日志。

### 操作步骤

打开 控制面板 > 管理工具 > 事件查看器，配置 应用日志、系统日志、安全日志 属性中的日志大小，以及设置当达到最大的日志尺寸时的相应策略。



### 3. IP协议安全配置

#### 3.1 IP协议安全

启用SYN攻击保护

启用SYN攻击保护。

- 指定触发SYN洪水攻击保护所必须超过的TCP连接请求数阈值为5。
- 指定处于 SYN\_RECV 状态的 TCP 连接数的阈值为500。
- 指定处于至少已发送一次重传的 SYN\_RECV 状态中的 TCP 连接数的阈值为400。

操作步骤

打开 注册表编辑器，根据推荐值修改注册表键值。

Windows Server 2012

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\SynAttackProtect
推荐值: 2
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\TcpMaxHalfOpen
推荐值: 500
```

Windows Server 2008

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SynAttackProtect
推荐值: 2
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\TcpMaxPortsExhausted
推荐值: 5
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\TcpMaxHalfOpen
```

推荐值: 500  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\TcpMaxHalfOpenRetried  
推荐值: 400

## 4. 文件权限

### 4.1 共享文件夹及访问权限

关闭默认共享

非域环境中，关闭Windows硬盘默认共享，例如C\$，D\$。

操作步骤

打开注册表编辑器，根据推荐值修改注册表键值。

注意：Windows Server 2012版本已默认关闭Windows硬盘默认共享，且没有该注册表键值。

HKLM\System\CurrentControlSet\Services\LanmanServer\Parameters\AutoShareServer 推荐值: 0

共享文件夹授权访问

每个共享文件夹的共享权限，只允许授权的帐户拥有共享此文件夹的权限。

操作步骤

每个共享文件夹的共享权限仅限于业务需要，不要设置成为Everyone。打开控制面板 > 管理工具 > 计算机管理，在共享文件夹中，查看每个共享文件夹的共享权限。

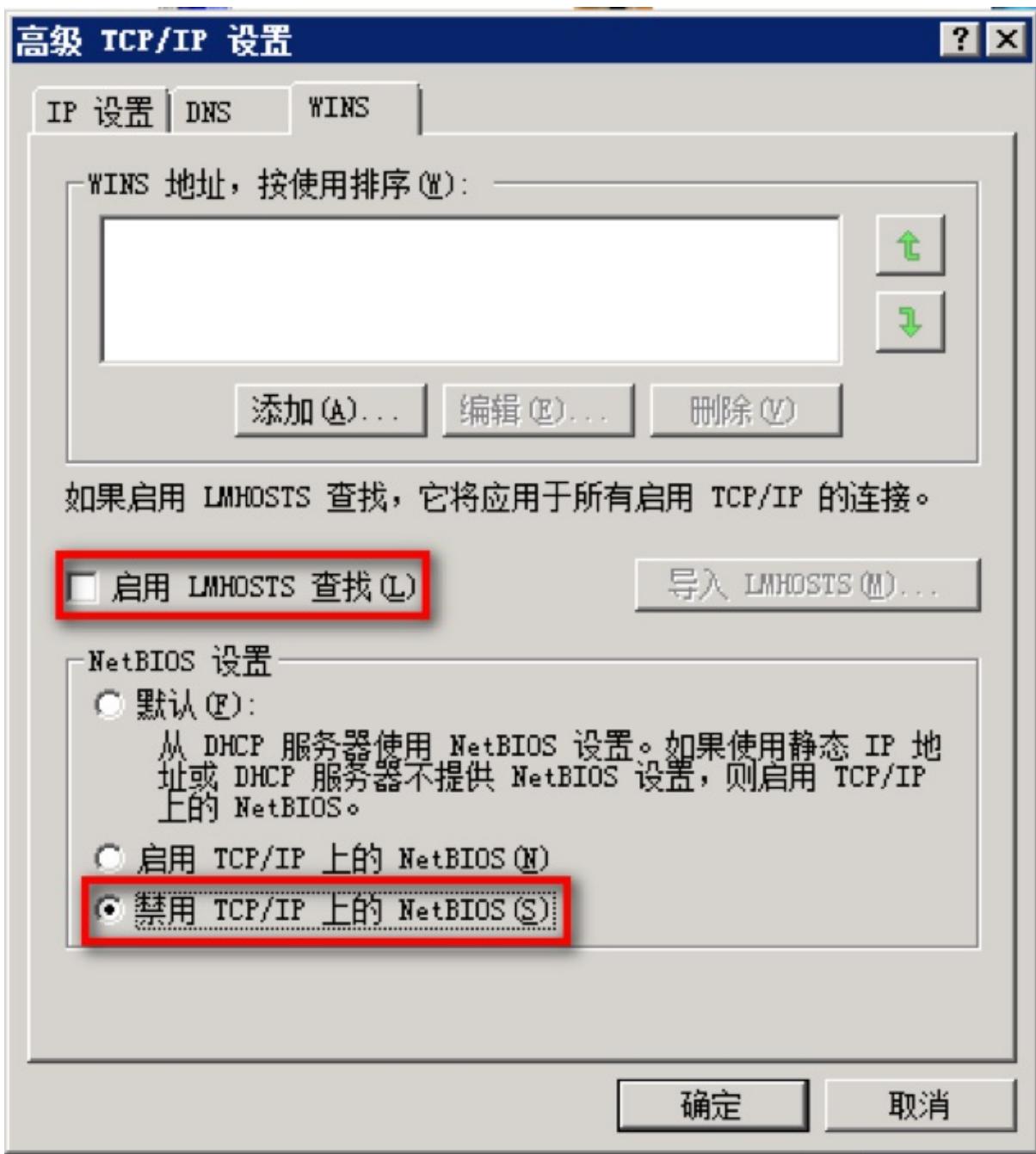
## 5. 服务安全

### 5.1 禁用TCP/IP上的NetBIOS

禁用TCP/IP上的NetBIOS协议，可以关闭监听的 UDP 137 (netbios-ns)、UDP 138 (netbios-dgm) 以及 TCP 139 (netbios-ssn) 端口。

操作步骤

在计算机管理 > 服务和应用程序 > 服务 中禁用 TCP/IP NetBIOS Helper 服务。在网络连接属性中，双击 Internet 协议版本4 (TCP/IPv4)，单击 高级。在 WINS 页签中，进行如下设置：



禁用不必要的服务

禁用不必要的服务, 请参考:

服务名称	建议
DHCP Client	如果不使用动态IP地址，就禁用该服务
Background Intelligent Transfer Service	如果不启用自动更新，就禁用该服务
Computer Browser	禁用
Diagnostic Policy Service	手动
IP Helper	禁用。该服务用于转换IPv6 to IPv4
Print Spooler	如果不需要打印，就禁用该服务
Remote Registry	禁用。Remote Registry主要用于远程管理注册表
Server	如果不使用文件共享，就禁用该服务。禁用本服务将关闭默认共享，如ipc\$、admin\$和c\$等
TCP/IP NetBIOS Helper	禁用
Windows Remote Management (WS-Management)	禁用
Windows Font Cache Service	禁用
WinHTTP Web Proxy Auto-Discovery Service	禁用
Windows Error Reporting Service	禁用

## 6. 安全选项

### 6.1 启用安全选项

操作步骤

打开 控制面板 > 管理工具 > 本地安全策略，在 本地策略 > 安全选项 中，进行如下设置：

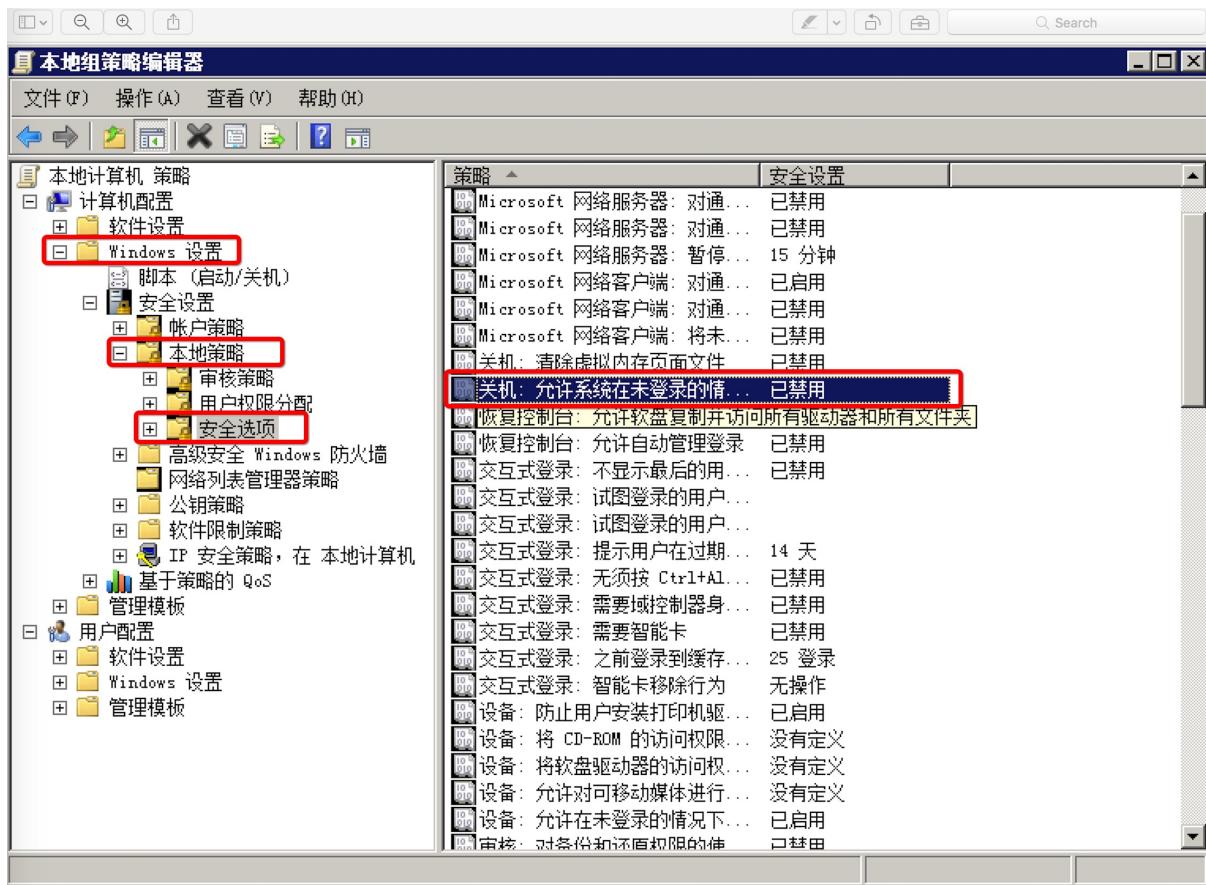
安全选项	配置内容
交互式登录: 尝试登录的用户的消息标题	注意
交互式登录: 尝试登录的用户的消息文本	内部系统只能因业务需要而使用，经由管理层授权。 管理层将随时监测此系统的使用。
Microsoft 网络服务器: 对通信进行数字签名(如果客户端允许)	启用
Microsoft 网络服务器: 对通信进行数字签名(始终)	启用
Microsoft 网络客户端: 对通信进行数字签名(如果服务器允许)	启用
Microsoft 网络客户端: 对通信进行数字签名(始终)	启用
网络安全: 基于 NTLM SSP 的(包括安全 RPC)服务器的最小会话安全	要求 NTLMv2 会话安全 要求 128 位加密
网络安全: 基于 NTLM SSP 的(包括安全 RPC)客户端的最小会话安全	要求 NTLMv2 会话安全 要求 128 位加密
网络安全: LAN 管理器身份验证级别	仅发送 NTLMv2 响应\拒绝 LM & NTLM
网络访问: 不允许 SAM 帐户的匿名枚举	启用 (默认已启用)
网络访问: 不允许 SAM 帐户和共享的匿名枚举	启用
网络访问: 可匿名访问的共享	清空 (默认为空)
网络访问: 可匿名访问的命名管道	清空 (默认为空)
网络访问: 可远程访问的注册表路径	清空, 不允许远程访问注册表
网络访问: 可远程访问的注册表路径和子路径	清空, 不允许远程访问注册表

## 6.2 禁用未登录前关机

服务器默认是禁止在未登录系统前关机的。如果启用此设置，服务器安全性将会大大降低，给远程连接的黑客造成可乘之机，强烈建议禁用未登录前关机功能。

### 操作步骤

打开 控制面板 > 管理工具 > 本地安全策略，在 本地策略 > 安全选项 中，禁用 关机: 允许系统在未登录前关机 策略。



## 7. 其他安全配置

### 7.1 防病毒管理

Windows系统需要安装防病毒软件。

#### 操作步骤

安装企业级防病毒软件，并开启病毒库更新及实时防御功能。

### 7.2 设置屏幕保护密码和开启时间

设置从屏幕保护恢复时需要输入密码，并将屏幕保护自动开启时间设定为五分钟。

#### 操作步骤

启用屏幕保护程序，设置等待时间为5分钟，并启用在恢复时使用密码保护。

### 7.3 限制远程登陆空闲断开时间

对于远程登陆的帐号，设置不活动超过时间15分钟自动断开连接。

#### 操作步骤

打开控制面板>管理工具>本地安全策略，在本地策略>安全选项中，设置Microsoft网络服务器：暂停会话前所需的空闲时间数量属性为15分钟。

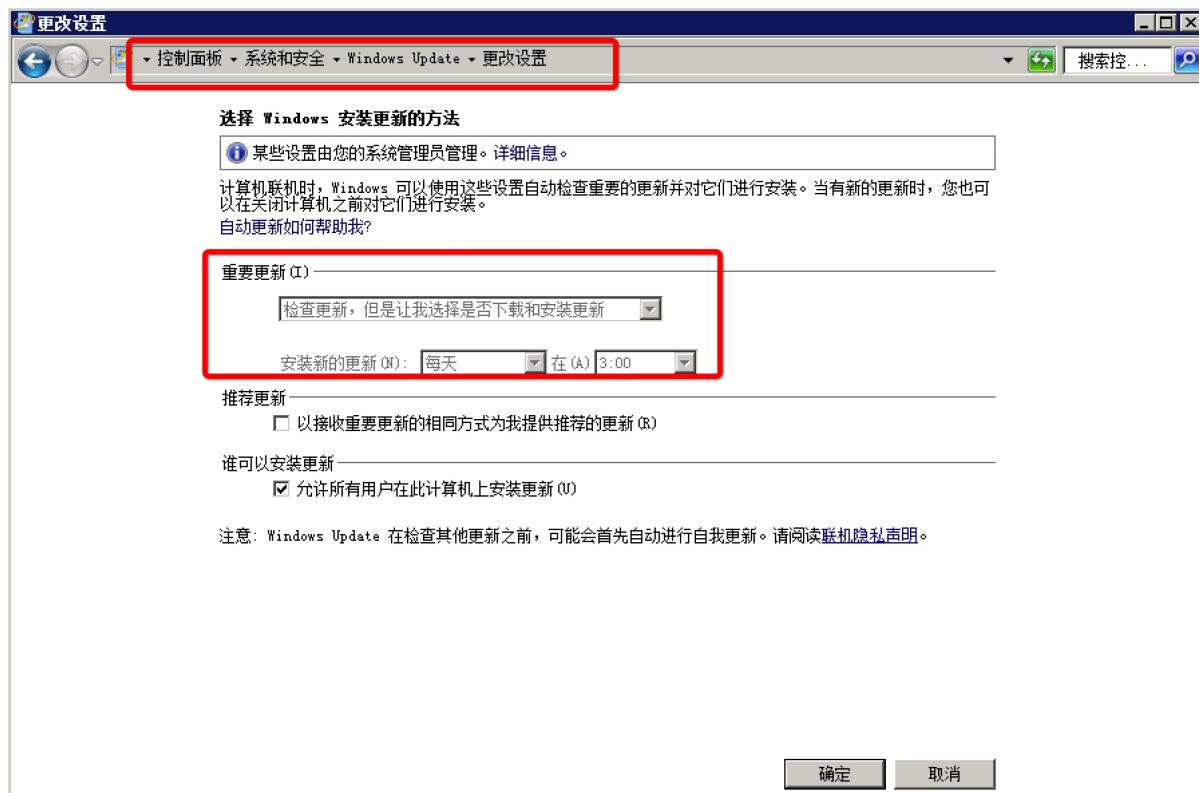
### 7.4 操作系统补丁管理

安装最新的操作系统Hotfix补丁。安装补丁时，应先对服务器系统进行兼容性测试。

## 操作步骤

安装最新的操作系统Hotfix补丁。安装补丁时，应先对服务器系统进行兼容性测试。

注意：对于实际业务环境服务器，建议使用通知并自动下载更新，但由管理员选择是否安装更新，而不是使用自动安装更新，防止自动更新补丁对实际业务环境产生影响。



本帮助手册旨在指导系统管理人员或安全检查人员进行Linux操作系统的安全合规性检查和加固。

## 1. 账号和口令

### 1.1 禁用或删除无用账号

减少系统无用账号，降低安全风险。

操作步骤

- 使用命令 `userdel <用户名>` 删除不必要的账号。
- 使用命令 `passwd -l <用户名>` 锁定不必要的账号。
- 使用命令 `passwd -u <用户名>` 解锁必要的账号。

### 1.2 检查特殊账号

检查是否存在空口令和root权限的账号。

操作步骤

- 查看空口令和root权限账号，确认是否存在异常账号：
  - 使用命令 `awk -F: '$2==""' /etc/shadow` 查看空口令账号。
  - 使用命令 `awk -F: '$3==0' /etc/passwd` 查看UID为零的账号。
- 加固空口令账号：
  - 使用命令 `passwd <用户名>` 为空口令账号设定密码。
  - 确认UID为零的账号只有root账号。

### 1.3 添加口令策略

加强口令的复杂度等，降低被猜解的可能性。

操作步骤

- 使用命令 `vi /etc/login.defs` 修改配置文件。
  - `PASS_MAX_DAYS 90` #新建用户的密码最长使用天数
  - `PASS_MIN_DAYS 0` #新建用户的密码最短使用天数
  - `PASS_WARN_AGE 7` #新建用户的密码到期前提醒天数
- 使用`chage`命令修改用户设置。
  - 例如，`chage -m 0 -M 30 -E 2000-01-01 -W 7 <用户名>` 表示将此用户的密码最长使用天数设为30，最短使用天数设为0，密码2000年1月1日过期，过期前七天警告用户。
- 设置连续输错三次密码，账号锁定五分钟。使用命令 `vi /etc/pam.d/common-auth` 修改配置文件，在配置文件中添加 `auth required pam_tally.so onerr=fail deny=3 unlock_time=300`。

### 1.4 限制用户su

限制能su到root的用户。

操作步骤

使用命令 `vi /etc/pam.d/su` 修改配置文件，在配置文件中添加行。例如，只允许test组用户su到root，则添加 `auth required pam_wheel.so group=test`。

## 2. 服务

## 2.1 关闭不必要的服务

关闭不必要的服务（如普通服务和`xinetd`服务），降低风险。

操作步骤

使用命令 `chkconfig --level <init级别> <服务名> on|off|reset` 设置服务在指定`init`级别下开机是否启动。

## 2.2 SSH服务安全

对SSH服务进行安全加固，防止暴力破解成功。

操作步骤

使用命令 `vim /etc/ssh/sshd_config` 编辑配置文件。

- 不允许root账号直接登录系统。设置 `PermitRootLogin` 的值为 `no`。
- 修改SSH使用的协议版本。设置 `Protocol` 的版本为 `2`。
- 修改允许密码错误次数（默认6次）。设置 `MaxAuthTries` 的值为 `3`。配置文件修改完成后，重启`sshd`服务生效。

# 3. 文件系统

## 3.1 设置umask值

设置默认的`umask`值，增强安全性。

操作步骤

使用命令 `vi /etc/profile` 修改配置文件，添加行 `umask 027`，即新创建的文件属主拥有读写执行权限，同组用户拥有读和执行权限，其他用户无权限。

## 3.2 设置登录超时

设置系统登录后，连接超时时间，增强安全性。

操作步骤

使用命令 `vi /etc/profile` 修改配置文件，将以 `TMOUT=` 开头的行注释，设置为 `TMOUT=180`，即超时时间为三分钟。

### 1. 日志

## 4.1 syslogd日志

启用日志功能，并配置日志记录。

操作步骤

Linux系统默认启用以下类型日志：

- 系统日志（默认）`/var/log/messages`
- `cron`日志（默认）`/var/log/cron`
- 安全日志（默认）`/var/log/secure`

注意：部分系统可能使用`syslog-ng`日志，配置文件为：`/etc/syslog-ng/syslog-ng.conf`。

您可以根据需求配置详细日志。

## 4.2 记录所有用户的登录和操作日志

通过脚本代码实现记录所以用户的登录操作日志，防止出现安全事件后无据可查。

#### 操作步骤

- 运行 [root@xxx ~]# vim /etc/profile 打开配置文件。
- 在配置文件中输入以下内容：

```
history
USER=`whoami`
USER_IP=`who -u am i 2>/dev/null| awk '{print $NF}'|sed -e 's/[()]///g'` 
if [ "$USER_IP" = "" ]; then
USER_IP=`hostname`
fi
if [ ! -d /var/log/history ]; then
mkdir /var/log/history
chmod 777 /var/log/history
fi
if [ ! -d /var/log/history/${LOGNAME} ]; then
mkdir /var/log/history/${LOGNAME}
chmod 300 /var/log/history/${LOGNAME}
fi
export HISTSIZE=4096
DT=`date +"%Y%m%d_%H:%M:%S"`
export HISTFILE="/var/log/history/${LOGNAME}/${USER}@${USER_IP}_${DT}"
chmod 600 /var/log/history/${LOGNAME}/*history* 2>/dev/null
```

- 运行 [root@xxx ~]# source /etc/profile 加载配置生效。注意： /var/log/history 是记录日志的存放位置，可以自定义。

通过上述步骤，可以在 /var/log/history 目录下以每个用户为名新建一个文件夹，每次用户退出后都会产生以用户名、登录IP、时间的日志文件，包含此用户本次的所有操作（root用户除外）。

同时，建议您使用OSS服务收集存储日志。

NFS（Network File System）是 FreeBSD 支持的一种文件系统，它允许网络中的计算机之间通过 TCP/IP 网络共享资源。不正确的配置和使用 NFS，会带来安全问题。

## 概述

NFS 的不安全性，主要体现于以下 4 个方面：

- 缺少访问控制机制
- 没有真正的用户验证机制，只针对 RPC/Mount 请求进行过程验证
- 较早版本的 NFS 可以使未授权用户获得有效的文件句柄
- 在 RPC 远程调用中，SUID 程序具有超级用户权限

## 加固方案

为有效应对以上安全隐患，推荐您使用下述加固方案。

### 配置共享目录（`/etc(exports)`）

使用 `anonuid`, `anongid` 配置共享目录，这样可以使挂载到 NFS 服务器的客户机仅具有最小权限。不要使用 `no_root_squash`。

### 使用网络访问控制

使用 [安全组策略](#) 或 `iptables` 防火墙限制能够连接到 NFS 服务器的机器范围。

```
iptables -A INPUT -i eth0 -p TCP -s 192.168.0.0/24 --dport 111 -j ACCEPT
iptables -A INPUT -i eth0 -p UDP -s 192.168.0.0/24 --dport 111 -j ACCEPT
iptables -A INPUT -i eth0 -p TCP -s 140.0.0.0/8 --dport 111 -j ACCEPT
iptables -A INPUT -i eth0 -p UDP -s 140.0.0.0/8 --dport 111 -j ACCEPT
```

### 账号验证

使用 Kerberos V5 作为登录验证系统，要求所有访问人员使用账号登录，提高安全性。

### 设置 NFSD 的 COPY 数目

在 Linux 中，NFSD 的 COPY 数目定义在启动文件 `/etc/rc.d/init.d/nfs` 中，默认值为 8。

最佳的 COPY 数目一般取决于可能的客户机数目。您可以通过测试来找到 COPY 数目的近似最佳值，并手动设置该参数。

### 选择传输协议

对于不同的网络情况，有针对地选择 UDP 或 TCP 传输协议。传输协议可以自动选择，也可以手动设置。

```
mount -t nfs -o sync,tcp,noatime,rsize=1024,wsize=1024 EXPORT_MACHINE:/EXPORTED_DIR /DIR
```

UDP 协议传输速度快，非连接传输时便捷，但其传输稳定性不如 TCP，当网络不稳定或者黑客入侵时很容易使 NFS 性能大幅降低，甚至导致网络瘫痪。一般情况下，使用 TCP 的 NFS 比较稳定，使用 UDP 的 NFS 速度较快。

- 在机器较少，网络状况较好的情况下，使用 UDP 协议能带来较好的性能。
- 当机器较多，网络情况复杂时，推荐使用 TCP 协议（V2 只支持 UDP 协议）。
- 在局域网中使用 UDP 协议较好，因为局域网有比较稳定的网络保证，使用 UDP 可以带来更好的性能。
- 在广域网中推荐使用 TCP 协议，TCP 协议能让 NFS 在复杂的网络环境中保持最好的传输稳定性。

## 限制客户机数量

修改 `/etc/hosts.allow` 和 `/etc/hosts.deny` 来限制客户机数量。

```
/etc/hosts.allow
portmap: 192.168.0.0/255.255.255.0 : allow
portmap: 140.116.44.125 : allow
/etc/hosts.deny
portmap: ALL : deny
```

## 改变默认的 NFS 端口

NFS 默认使用的是 111 端口，使用 `port` 参数可以改变这个端口值。改变默认端口值能够在一定程度上增强安全性。

## 配置 nosuid 和 noexec

SUID (Set User ID) 或 SGID (Set Group ID) 程序可以让普通用户以超过自己权限来执行。很多 SUID/SGID 可执行程序是必须的，但也可能被一些恶意的本地用户利用，获取本不应有的权限。

尽量减少所有者是 `root`，或是在 `root` 组中却拥有 SUID/SGID 属性的文件。您可以删除这样的文件或更改其属性，如：

- 使用 `nosuid` 选项禁止 set-UID 程序在 NFS 服务器上运行，可以在 `/etc/exports` 加入一行：

```
/www www.abc.com(rw, root_squash, nosuid)
```

- 使用 `noexec` 禁止直接执行其中的二进制文件。

**Rsync** 是一个通过检查文件的时间戳和大小，来跨计算机系统高效地传输和同步文件的工具。

通常情况下，管理程序在启动 **Rsync** 服务后，会直接运行传输任务。如果 **Rsync** 服务未经过安全加固，则很容易出现未授权访问等安全问题；其直接后果是传输数据裸露在互联网上，可以被任何人访问获取，带来严重的数据泄露风险。

建议您在使用 **Rsync** 服务端时，参考本文对 **Rsync** 服务进行安全加固，保障数据安全。

## 加固方案

隐藏 **module** 信息

将配置文件修改为以下内容：

```
list = false
```

## 使用权限控制

将不需要写入权限的 **module** 设置为只读：

```
read only = true
```

## 限制网络访问

使用 [安全组策略](#) 或白名单，限制允许访问主机的 IP 地址。

```
hosts allow = 123.123.123.123
```

## 启用账户认证

只允许指定的用户，使用指定的密码，来调用 **Rsync** 服务。

- 服务端配置

```
auth users = ottocho
secrets file = /etc/rsyncd.secrets
```

在文件 `/etc/rsyncd.secrets` 中写入使用的账号密码，格式为：`username:password`，支持多行。  
注意：密码要求满足强密码策略，必须是 8 位以上，且包括大小写字母、数字、特殊字符的字符串。此处的 `password` 使用明文。

- 客户端配置

在客户端，使用 `--password-file=/etc/rsyncd.secrets` 参数，在 `/etc/rsyncd.secrets` 中写入密码。

```
Rsync -av --password-file=/etc/rsyncd.secrets test.host.com::files /des/path
```

在上述 `/etc/rsyncd.secrets` 密码文件中，用户或用户组必须和实际使用者保持一致，且权限必须是 600。

## 数据加密传输

`Rsync` 默认不支持加密传输，如果需要使用 `Rsync` 传输重要性很高的数据，可以使用 `SSH` 模式。

`Rsync` 支持以下两种同步模式：

- 当源路径或目的路径的主机名后面包含一个冒号分隔符时，`Rsync` 使用 `SSH` 传输。
- 当源路径或目的路径的主机名后面包含两个冒号，或使用 `Rsync://URL` 时，`Rsync` 使用 `TCP` 直接连接 `Rsync daemon`。

在配置好 `SSH` 后，推荐参照以下方式来使用：

```
Rsync -av test.host.com:/path/to/files /des/path
```

本文介绍如何在 SMB 客户端和服务器组件上启用/禁用服务器消息块 SMBv1、SMBv2 和 SMBv3。

注意：建议由专业技术工程师完成以下操作。

## 禁用 **SMBv2** 和 **SMBv3** 的影响

我们建议不要禁用 SMBv2 或 SMBv3。禁用 SMBv2 或 SMBv3 只能作为临时故障排除措施。请勿使 SMBv2 或 SMBv3 保持禁用状态。

### 禁用 **SMBv2** 的影响

在 Windows 7 和 Windows Server 2008 R2 中，禁用 SMBv2 会停用以下功能：

- 请求复合 - 允许发送多个 SMB 2 请求作为单个网络请求
- 大型读写 - 更好地利用更快速的网络
- 文件夹和文件属性缓存 - 客户端保留文件夹和文件的本地副本
- 持久句柄 - 如果临时断开连接，则允许连接以透明方式重新连接到服务器
- 改进的消息签名 - HMAC SHA-256 代替 MD5 作为哈希算法
- 改进的文件共享扩展性 - 每个服务器的用户数量、共享数量和打开文件数量大大增加
- 支持符号链接
- 客户端 **oplock** 租赁模式 - 限制在客户端和服务器之间传输的数据，从而提高高延迟网络性能并增强 SMB 服务的扩展性
- 大型 MTU 支持 - 可充分利用 10 千兆字节 (GB) 以太网
- 改进的能效 - 向服务器打开文件的客户端可以睡眠

### 禁用 **SMBv3** 的影响

在 Windows 8、Windows 8.1、Windows 10、Windows Server 2012 和 Windows Server 2016 中，禁用 SMBv3 会停用以下功能（以及以上列表中所述的 SMBv2 功能）：

- 透明故障转移 - 在维护或故障转移期间，客户端会重新连接，不会干扰群集节点
- 扩展 - 并发访问所有文件群集节点上的共享数据
- 多通道 - 如果客户端和服务器之间有多个路径可用时，则聚合网络带宽和容错
- SMB 直通 – 增加 RDMA 网络支持，实现极高的性能、低延迟和低 CPU 利用率
- 加密 – 提供端到端加密，并防止不可靠网络上的窃听
- 目录租赁 - 通过缓存改进分支机构中应用程序的响应时间
- 性能优化 - 对小型随机读/写 I/O 的优化

## 在 **SMB** 服务器上启用/禁用 **SMB** 协议

### Windows 8 和 Windows Server 2012

Windows 8 和 Windows Server 2012 引入了新的 Set-SMBServerConfiguration Windows PowerShell cmdlet。通过此 cmdlet，你可以在服务器组件上启用或禁用 SMBv1、SMBv2 和 SMBv3 协议。

注意：因为 SMBv2 和 SMBv3 共用一个堆叠，所以在 Windows 8 或 Windows Server 2012 中启用或禁用 SMBv2 时，也会启用或禁用 SMBv3。

## 使用 PowerShell cmdlet

运行 Set-SmbServerConfiguration cmdlet 后，无须重启计算机。

- 若要获取 SMB 服务器协议配置的当前状态，请运行以下 cmdlet：

```
Get-SmbServerConfiguration | Select EnableSMB1Protocol, EnableSMB2Protocol
```

- 若要在 SMB 服务器上禁用 SMBv1，请运行以下 cmdlet：

```
Set-SmbServerConfiguration -EnableSMB1Protocol $false
```

- 若要在 SMB 服务器上禁用 SMBv2 和 SMBv3，请运行以下 cmdlet：

```
Set-SmbServerConfiguration -EnableSMB2Protocol $false
```

- 若要在 SMB 服务器上启用 SMBv1，请运行以下 cmdlet：

```
Set-SmbServerConfiguration -EnableSMB1Protocol $true
```

- 若要在 SMB 服务器上启用 SMBv2 和 SMBv3，请运行以下 cmdlet：

```
Set-SmbServerConfiguration -EnableSMB2Protocol $true
```

## Windows 7、Windows Server 2008 R2、Windows Vista 和 Windows Server 2008

若要在运行 Windows 7、Windows Server 2008 R2、Windows Vista 或 Windows Server 2008 的 SMB 服务器上启用或禁用 SMB 协议，请使用 Windows PowerShell 或注册表编辑器。

## 使用 Windows PowerShell 2.0 或更高版本的 PowerShell

- 若要在 SMB 服务器上禁用 SMBv1，请运行以下 cmdlet：

```
Set-ItemProperty -Path "HKLM:\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters" SMB1 -Type DWORD -Value 0 -Force
```

- 若要在 SMB 服务器上禁用 SMBv2 和 SMBv3，请运行以下 cmdlet：

```
Set-ItemProperty -Path "HKLM:\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters" SMB2 -Type DWORD -Value 0 -Force
```

- 若要在 SMB 服务器上启用 SMBv1，请运行以下 cmdlet：

```
Set-ItemProperty -Path "HKLM:\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters" SMB1 -Type DWORD -Value 1 -Force
```

- 若要在 SMB 服务器上启用 SMBv2 和 SMBv3，请运行以下 cmdlet:

```
Set-ItemProperty -Path "HKLM:\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters" SMB2 -Type DWORD -Value 1 -Force
```

注意: 进行这些更改后, 必须重启计算机。

## 使用注册表编辑器

注意: 以下内容包含有关如何修改注册表的信息。修改注册表之前, 一定要先对其进行备份。并且一定要知道在发生问题时如何还原注册表。有关如何备份、还原和修改注册表的更多信息, 请查看 如何在 Windows 中备份和还原注册表。

- 若要在 SMB 服务器上启用或禁用 SMBv1, 请配置以下注册表项:

- 注册表子项:

HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters 注册表项:  
SMB1

- REG\_DWORD: 0 = 已禁用
- REG\_DWORD: 1 = 已启用
- 默认值: 1 = 已启用

- 若要在 SMB 服务器上启用或禁用 SMBv2, 请配置以下注册表项:

- 注册表子项:

HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters 注册表项:  
SMB2

- REG\_DWORD: 0 = 已禁用
- REG\_DWORD: 1 = 已启用
- 默认值: 1 = 已启用

## 在 **SMB** 客户端上启用/禁用 **SMB** 协议

Windows Vista、Windows Server 2008、Windows 7、Windows Server 2008 R2、Windows 8 和 Windows Server 2012

注意: 因为 SMBv2 和 SMBv3 共用一个堆叠, 所以在 Windows 8 或 Windows Server 2012 中启用或禁用 SMBv2 时, 也会启用或禁用 SMBv3。

- 若要在 SMB 客户端上禁用 SMBv1, 请运行以下命令:

```
sc.exe config lanmanworkstation depend= bowser/mrxsmb20/nsi  
sc.exe config mrxsmb10 start= disabled
```

- 若要在 SMB 客户端上启用 SMBv1, 请运行以下命令:

```
sc.exe config lanmanworkstation depend= bowser/mrxsmb10/mrxsmb20/nsi  
sc.exe config mrxsmb10 start= auto
```

- 若要在 SMB 客户端上禁用 SMBv2 和 SMBv3, 请运行以下命令:

```
sc.exe config lanmanworkstation depend= bowser/mrxsmb10/nsi  
sc.exe config mrxsmb20 start= disabled
```

- 若要在 SMB 客户端上启用 SMBv2 和 SMBv3，请运行以下命令：

```
sc.exe config lanmanworkstation depend= bowser/mrxsmb10/mrxsmb20/nsi  
sc.exe config mrxsmb20 start= auto
```

注意：

- 必须在提升的命令提示符中运行这些命令。
- 进行这些更改后，必须重启计算机。

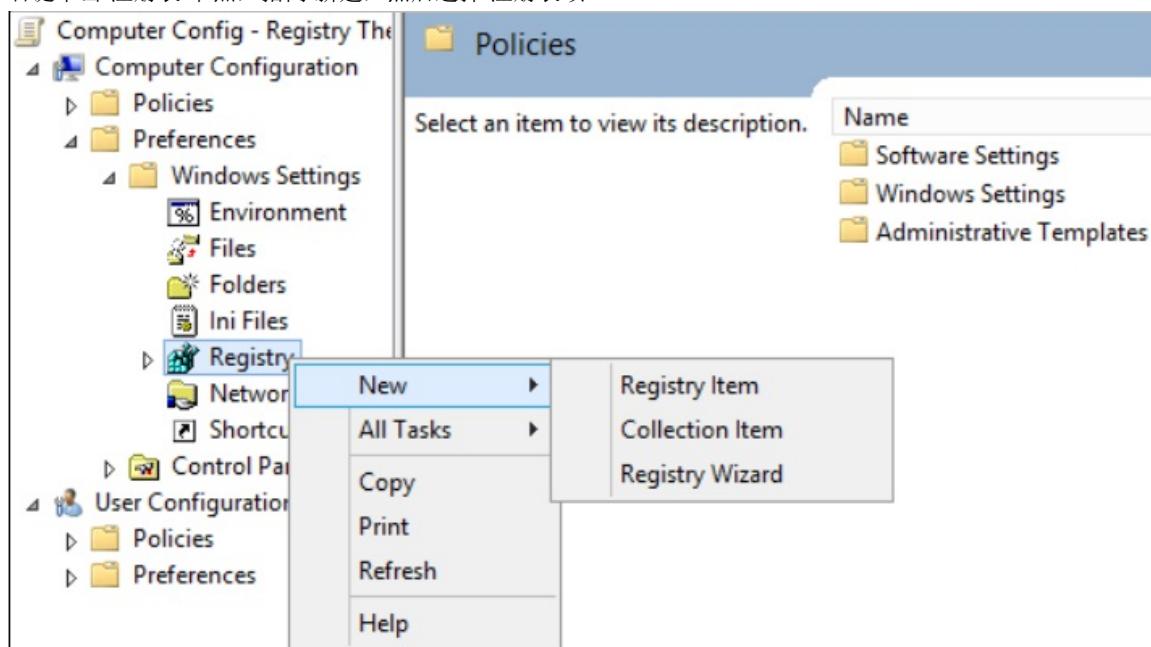
## 使用组策略禁用 **SMBv1** 服务器

这将在注册表中配置以下新项：

HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters 注册表项： SMB1 REG\_DWORD: 0 = Disabled

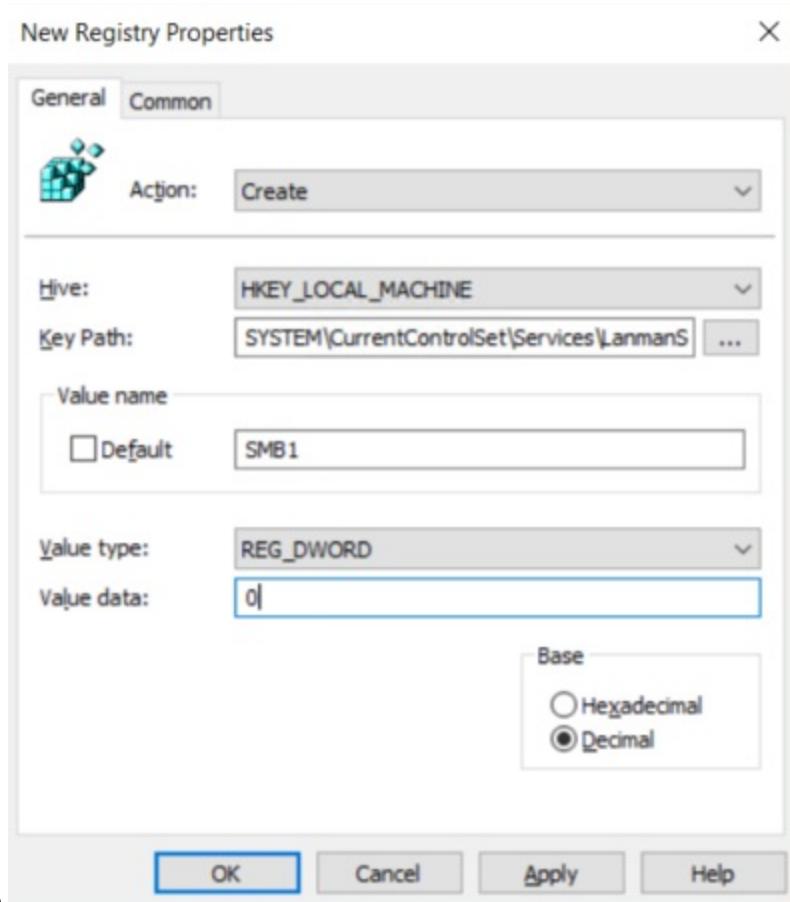
## 使用组策略配置流程

- 打开组策略管理控制台。右键单击应包含新首选项的组策略对象 (GPO)，然后单击 编辑。
- 在 计算机配置 下的控制台树中，展开 首选项 文件夹，然后展开 Windows 设置 文件夹。
- 右键单击 注册表 节点，指向 新建，然后选择 注册表项。



- 在 新建注册表属性 对话框中，选择以下内容：
- 操作： 创建
- Hive: HKEY\_LOCAL\_MACHINE
- 注册表项路径: SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters

8. 值名称: SMB1
9. 值类型: REG\_DWORD



10. 值数据: 0

11. 将此组策略应用到域中所有必需的工作站、服务器和域控制器，以禁用 SMBv1 服务器组件。也可以将 WMI 筛选器设置为不包含不受支持的操作系统或选中的排除项（如 Windows XP）。

注意：在旧版 Windows XP 或 Linux 早期版本以及第三方系统（不支持 SMBv2 或 SMBv3）需要访问 SYSVOL 或其他文件共享（已启用 SMB v1）的域控制器上进行这些更改时要谨慎小心。

## 使用组策略禁用 **SMBv1** 客户端

若要禁用 **SMBv1** 客户端，需要将服务注册表项更新为禁止 MRxSMB10 启动，然后还需要将 MRxSMB10 的依赖项从 LanmanWorkstation 项中删除，以便它可以正常启动（无需首先启动 MRxSMB10）。

这将更新和替换注册表以下 2 个项中的默认值

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\mrxsmb10 注册表项: Start REG_DWORD: 4 = Disabled  
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanWorkstation 注册表项: DependOnService REG_MULTI_SZ: "Bowers","MRxSmb20","NSI"
```

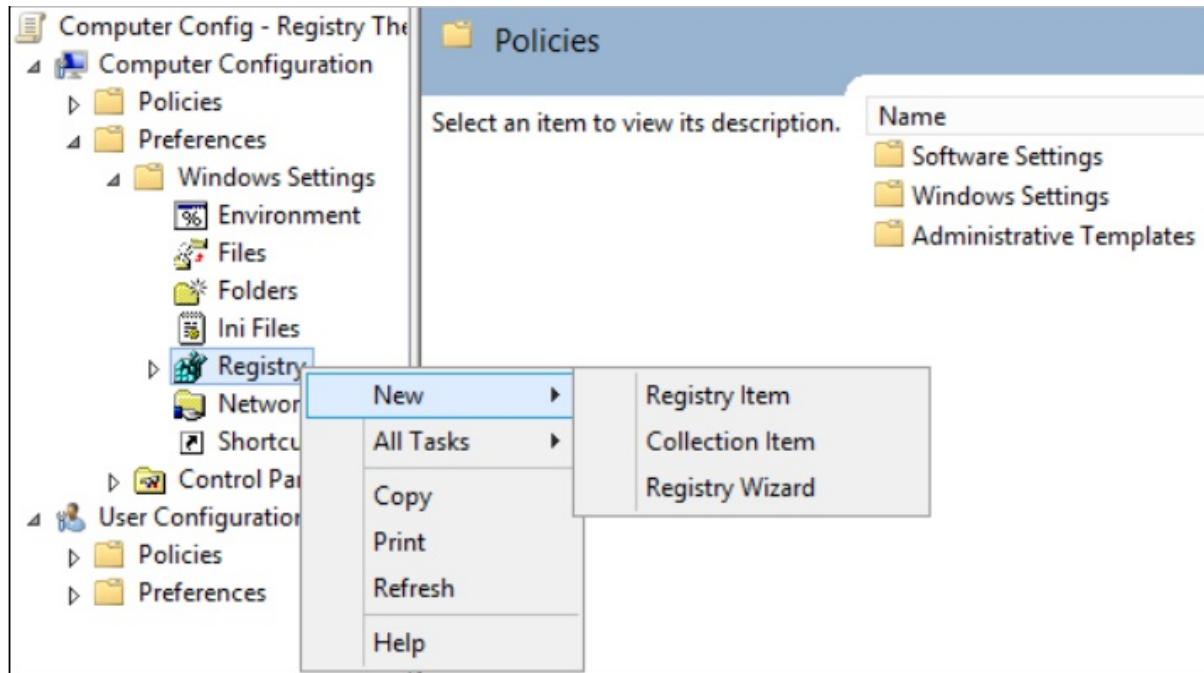
注意：默认包含的 MRxSMB10 现已作为依赖项删除。

## 使用组策略配置流程

1. 打开组策略管理控制台。右键单击应包含新首选项的组策略对象 (GPO)，然后单击 编辑。

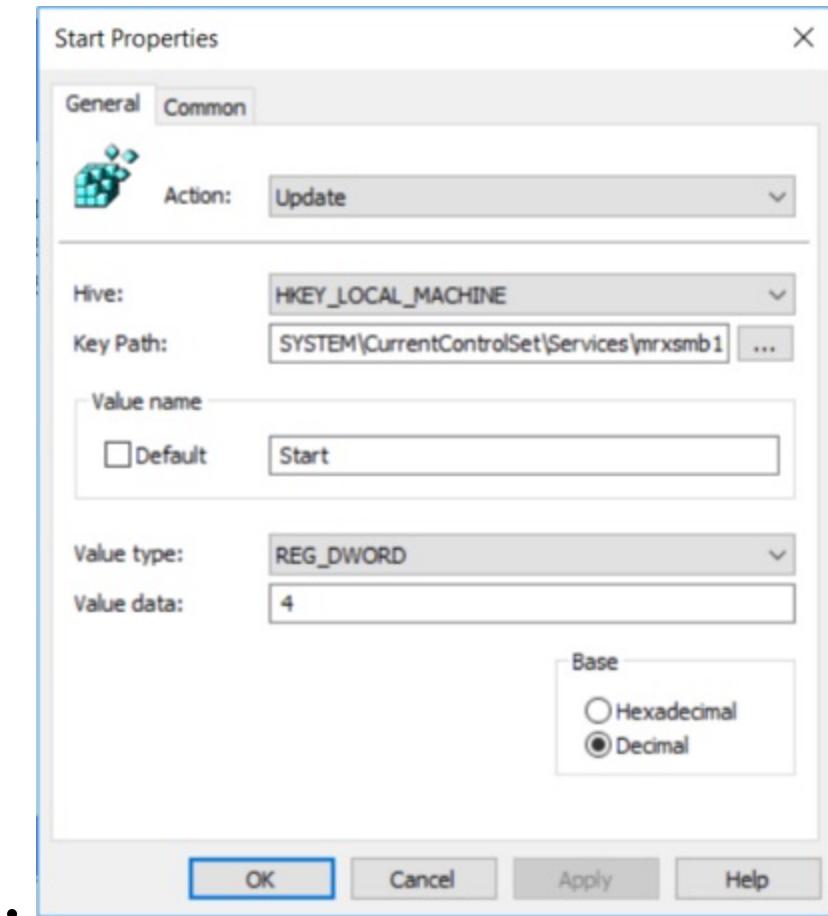
2. 在 计算机配置 下的控制台树中，展开 首选项 文件夹，然后展开 Windows 设置 文件夹。

3. 右键单击 注册表 节点，指向 新建，然后选择 注册表项。



4. 在新建注册表属性 对话框中，选择以下内容：

- 操作：更新
- Hive: HKEY\_LOCAL\_MACHINE
- 注册表项路径： SYSTEM\CurrentControlSet\services\mrxsmb10
- 值名称： Start
- 值类型： REG\_DWORD
- 值数据： 4

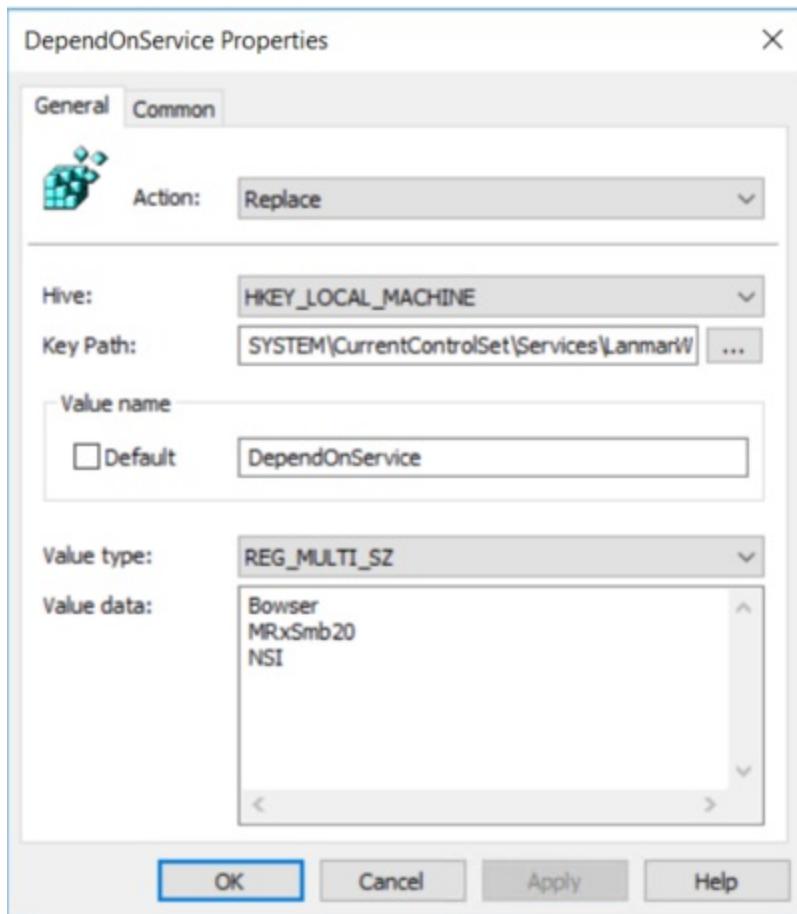


然后删除刚刚禁用的 MRxSMB10 的依赖项

5. 在新建注册表属性 对话框中，选择以下内容：

- 操作： 替换
- Hive: HKEY\_LOCAL\_MACHINE
- 注册表项路径: SYSTEM\CurrentControlSet\Services\LanmanWorkstation
- 值名称: DependOnService
- 值类型 REG\_MULTI\_SZ
- 值数据:
  - Bowser
  - MRxSmb20
  - NSI

注意： 这 3 个字符串不带项目符号（具体如下）



在 Windows 的多个版本中，默认值包括 MRxSMB10，通过将其替换为此多值字符串，实际上就删除了作为 LanmanServer 依赖项的 MRxSMB10，结果是从四个默认值减少为上述这三个值。

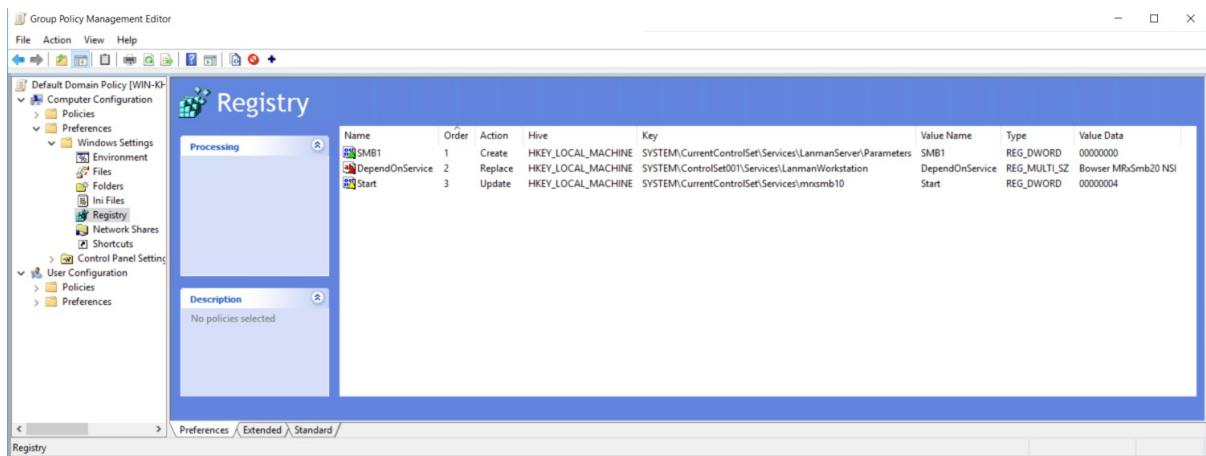
注意：使用组策略管理控制台时，无需使用引号或逗号。只需在各行键入每个项，如上面所示。

## 需要重新启动

应用策略且正确设置注册表后，必须重新启动目标系统，然后才能禁用 SMB v1。

## 摘要

如果所有设置均在同一组策略对象 (GPO) 中，组策略管理将显示以下设置。



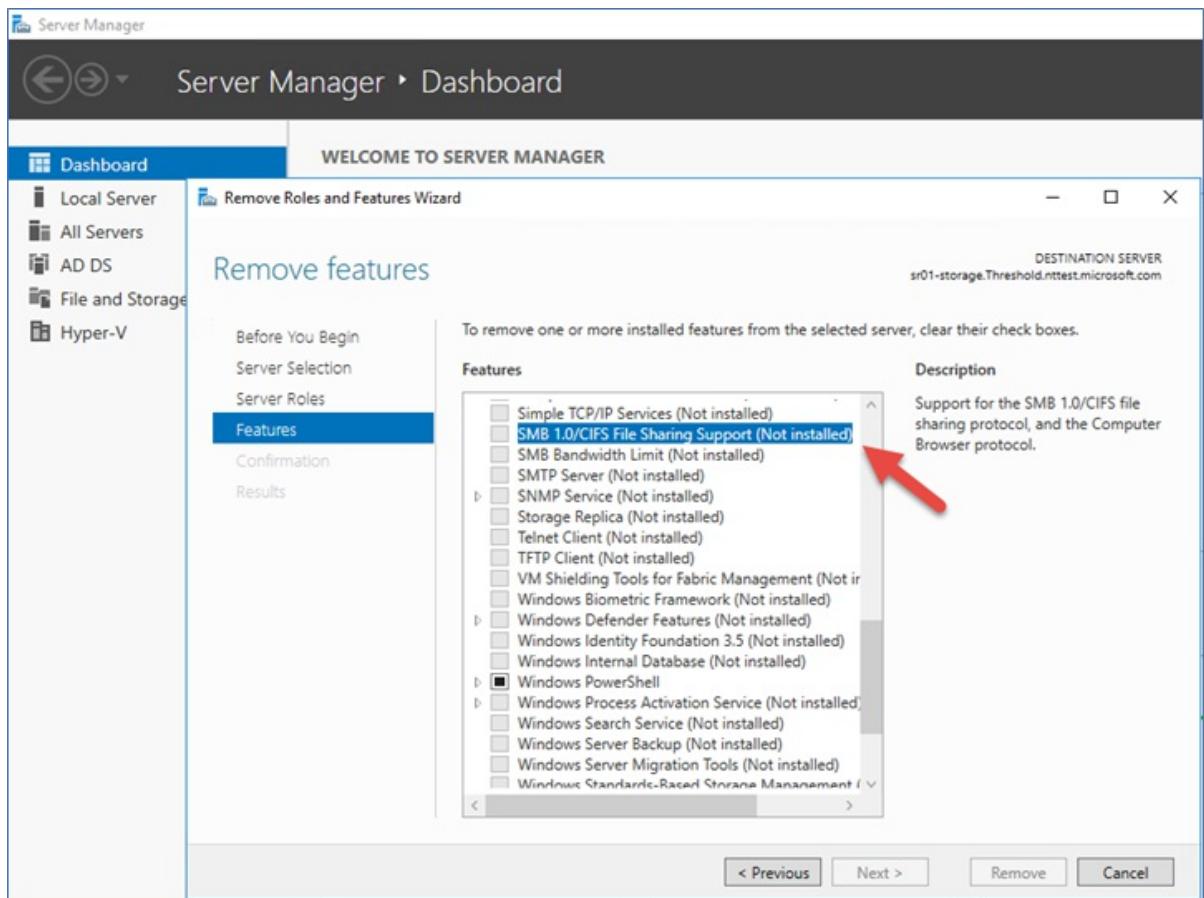
## 测试和验证

配置完成后即允许策略进行复制和更新。作为测试的必要步骤，请从 CMD.EXE 提示符处运行 `gpupdate/force`，然后查看目标计算机，以确保注册表设置得以正确应用。确保 SMBv2 和 SMBv3 在环境中的所有其他系统中正常运行。

注意：请务必重新启动目标系统。

## 如何在 Windows 8.1、Windows 10、Windows 2012 R2 和 Windows Server 2016 中轻松删除 SMBv1

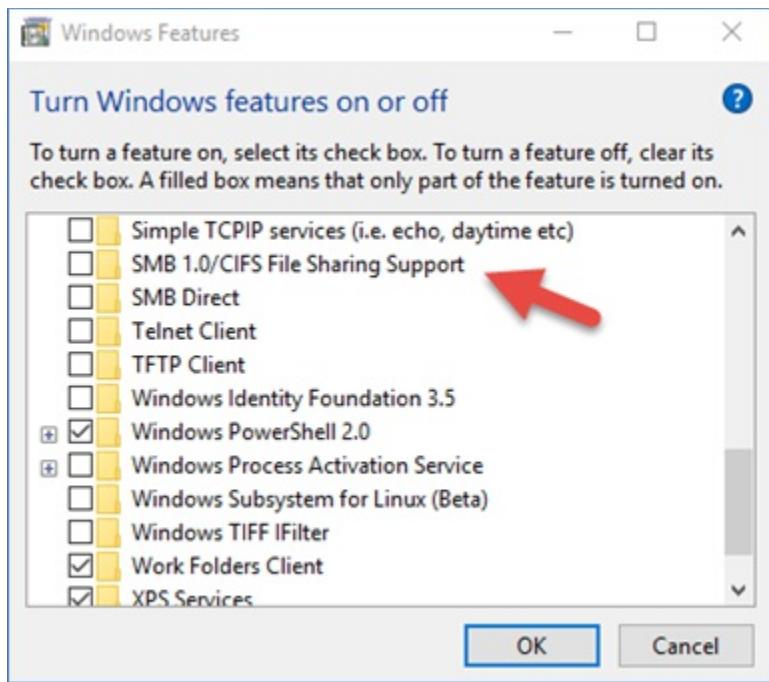
**Windows Server:** 使用“服务器管理器”



## Windows Server: 使用 PowerShell (Remove- WindowsFeature FS-SMB1)

```
Administrator: Windows PowerShell
PS C:\> PS C:\> Remove-WindowsFeature -Name FS-SMB1
Success Restart Needed Exit Code      Feature Result
----- -----No          NoChangeNeeded {}
PS C:\> -
```

Windows 客户端：使用“添加或删除程序”



## Windows 客户端：使用 PowerShell (Disable-WindowsOptionalFeature -Online -FeatureName smb1protocol)

```
Administrator: Windows PowerShell
PS C:\> Disable-WindowsOptionalFeature -Online -FeatureName SMB1Protocol
Path          :
Online        : True
RestartNeeded : False

PS C:\>
```

## 参考与适用性

本文来自微软官方技术文档：[如何在 Windows 和 Windows Server 中启用和禁用 SMBv1、SMBv2 和 SMBv3](#)。

如有变化，以微软官方为准。

这篇文章中的信息适用于：

- Windows 10 Pro released in July 2015,
- Windows 10 Enterprise released in July 2015
- Windows Vista Enterprise
- Windows Vista Business
- Windows Vista Home Basic
- Windows Vista Home Premium

- Windows Vista Ultimate
- Windows 7 Enterprise
- Windows 7 Home Basic
- Windows 7 Home Premium
- Windows 7 Professional
- Windows 7 Ultimate
- Windows Server 2008 Datacenter
- Windows Server 2008 Enterprise
- Windows Server 2008 Standard
- Windows Server 2008 R2 Datacenter
- Windows Server 2008 R2 Enterprise
- Windows Server 2008 R2 Standard
- Windows 8
- Windows 8 Enterprise
- Windows 8 Pro
- Windows Server 2012 Datacenter
- Windows Server 2012 Datacenter
- Windows Server 2012 Essentials
- Windows Server 2012 Foundation
- Windows Server 2012 Foundation
- Windows Server 2012 Standard
- Windows Server 2012 Standard
- Windows Server 2016

- **FTP**匿名登录或弱口令漏洞及服务加固
- **Docker**服务安全加固
- **Kubernetes**服务安全加固
- **Hadoop**环境安全加固
- **FileZilla FTP Server** 安全加固
- **Elasticsearch**服务安全加固
- **phpMyadmin** 服务安全加固

## 漏洞描述

FTP 弱口令或匿名登录漏洞，一般指使用 **FTP** 的用户启用了匿名登录功能，或系统口令的长度太短、复杂度不够、仅包含数字、或仅包含字母等，容易被黑客攻击，发生恶意文件上传或更严重的入侵行为。

## 漏洞危害

黑客利用弱口令或匿名登录漏洞直接登录 **FTP** 服务，上传恶意文件，从而获取系统权限，并可能造成数据泄露。

## 加固方案

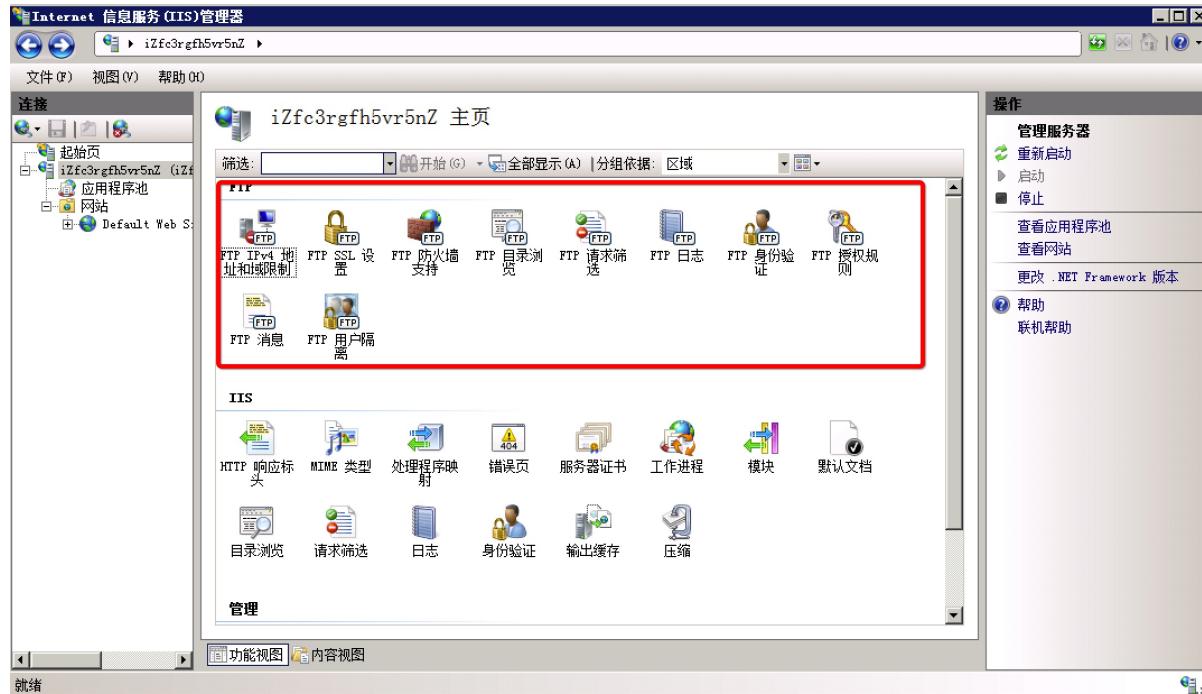
不同 **FTP** 服务软件可能有不同的防护程序，本修复方案以 **Windows server 2008** 中自带的 **FTP** 服务和 **Linux** 中的 **vsftpd** 服务为例，您可参考以下方案对您的 **FTP** 服务进行安全加固。

重要提示：

- 请确保您的 **FTP** 服务软件为官方最新版本。同时，建议您不定期关注官方发布的补丁，并及时进行更新。
- 强烈建议不要将此类型的服务在互联网开放，您可以使用 **VPN** 等安全接入手段连接到 **FTP** 服务器端，同时使用 [安全组](#) 来控制访问源IP。

## Windows 系统 **FTP** 服务安全加固\*\*

打开 IIS 信息服务管理器，查看所有 **FTP** 服务相关的安全加固功能。



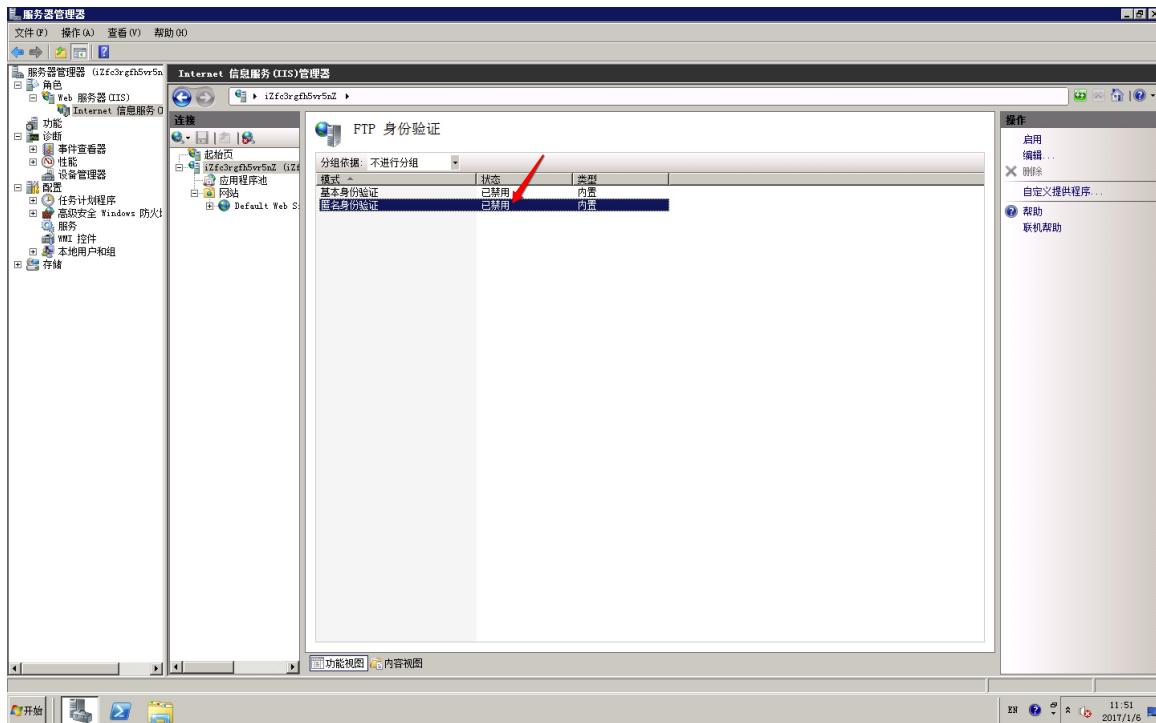
### 1. 禁用匿名登录

## 1>. 创建 FTP 帐户。

在开始 > 管理工具 > 计算机管理 > 本地用户和组 中，创建用户，设置强密码（密码建议八位以上，包括大小写字母、特殊字符、数字等混合体，不要使用生日、姓名拼音等常见字符串），并设置该用户属于 GUESTS 用户组。

## 2>. 禁用匿名登录。

### • Windows 2008 系统 FTP 禁用匿名登录服务

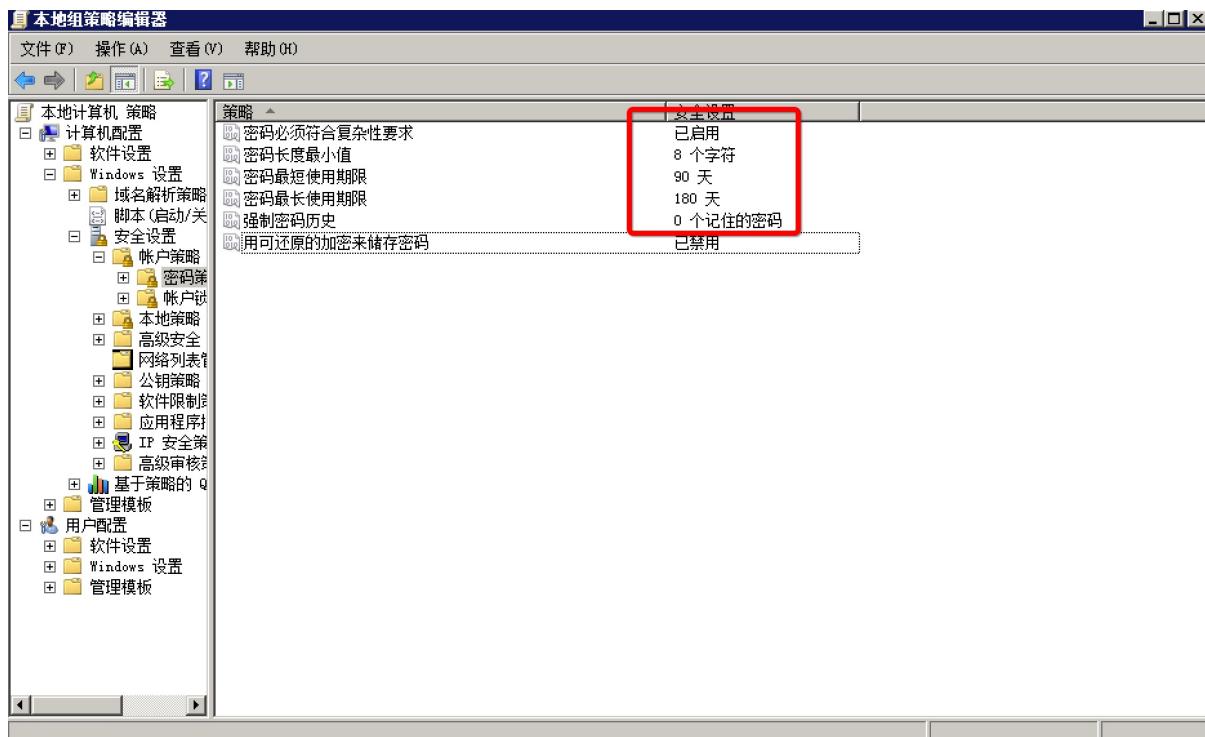


### • Windows 2012 系统 FTP 禁用匿名登录服务



## 2. 启用强密码安全策略

在 Windows 系统中，强密码策略是通过组策略控制的。您可以打开本地组策略编辑器（gpedit.msc），计算机配置 > Windows 设置 > 安全设置 > 账户策略 > 密码策略，启用密码复杂策略。



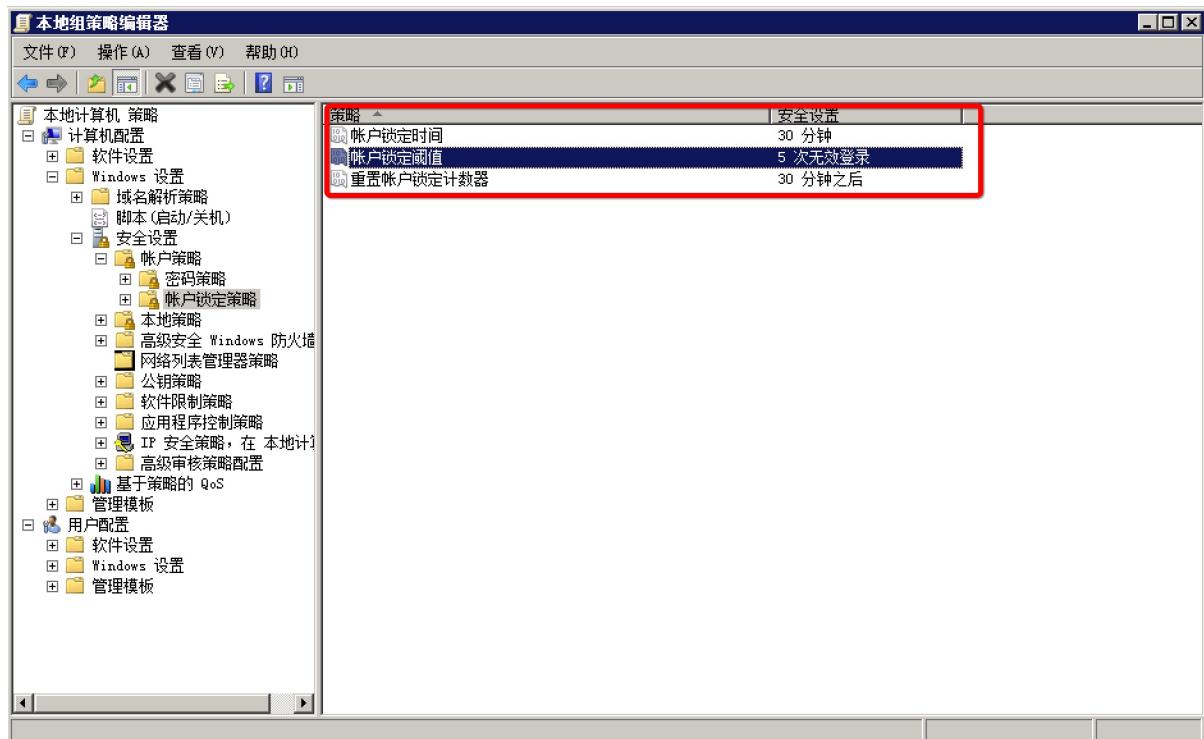
启用 密码必须符合复杂性要求 策略后，在更改或创建用户密码时会执行复杂性策略检测，密码必须符合以下最低要求：

- 密码不能包含账户名
- 密码不能包含用户名中超过两个连续字符的部分
- 密码至少有六个字符长度
- 密码必须包含以下四类字符中的至少三类字符类型：英文大写字母(A-Z)、英文小写字母(a-z)、10个基本数字(0-9)、特殊字符（例如：!、¥、#、%）

注意： 推荐 Windows 所有需要进行用户认证的服务都采用上述复杂密码策略。

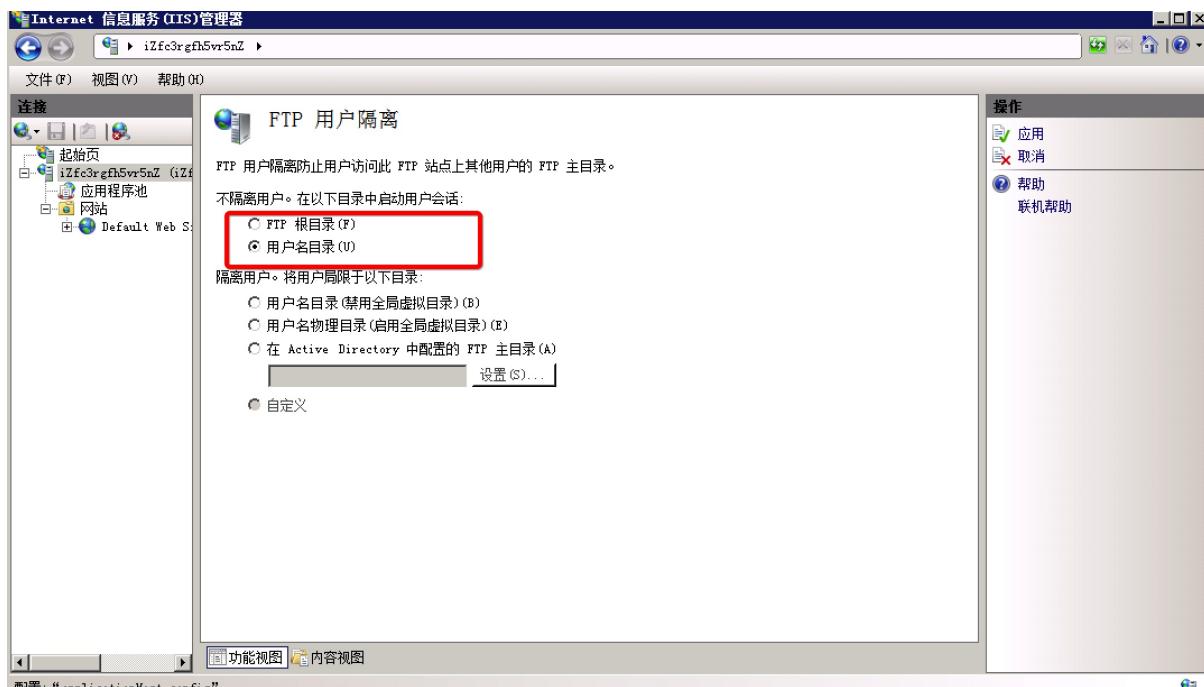
## 3. 启用账户登录失败处理机制

该机制对登录失败的账户实施强处理，可有效防止暴力破解攻击事件。



#### 4. 启用 FTP 目录隔离机制

FTP 目录隔离功能可以防止用户查看其它用户目录的文件，防止数据泄露。

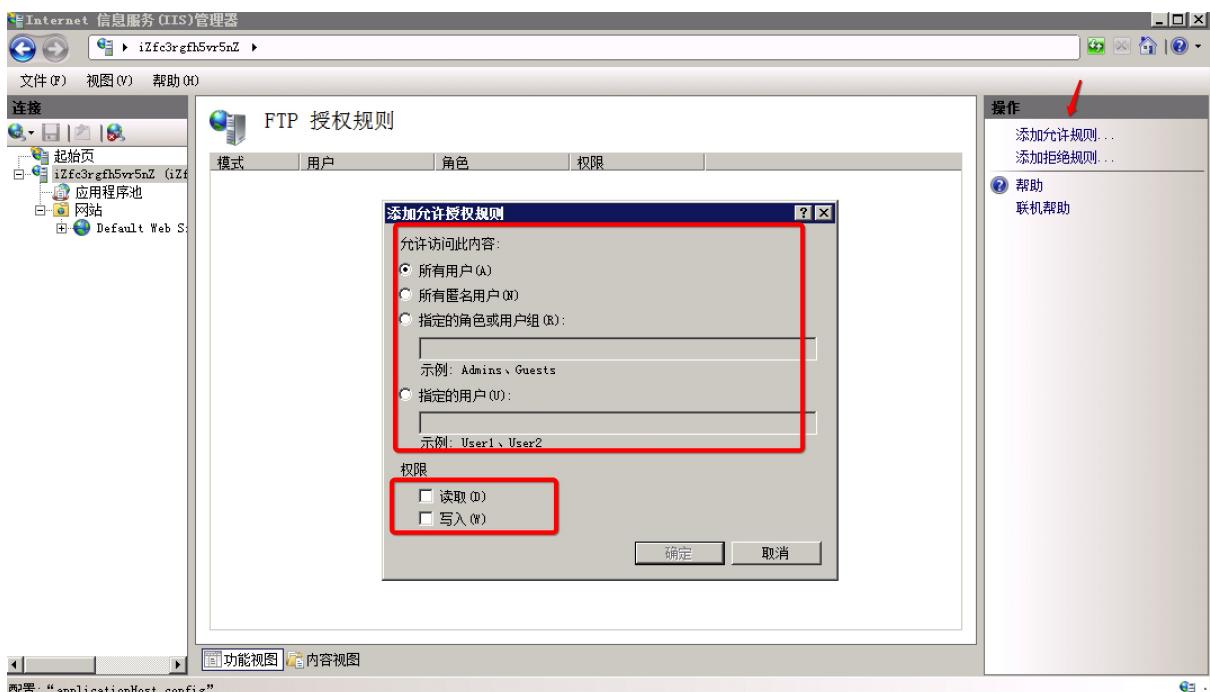


#### 5. 指定访问源 IP



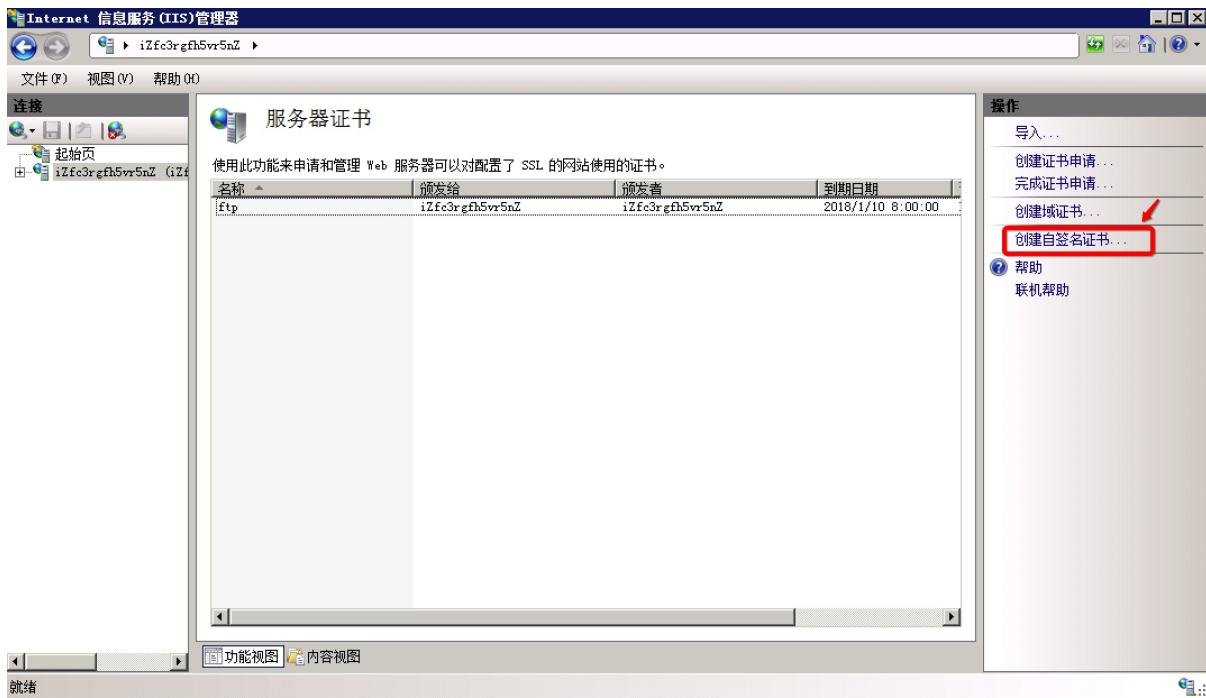
## 6. 启用授权机制

您可以根据业务需求配置授权规则，限制用户访问的权限。

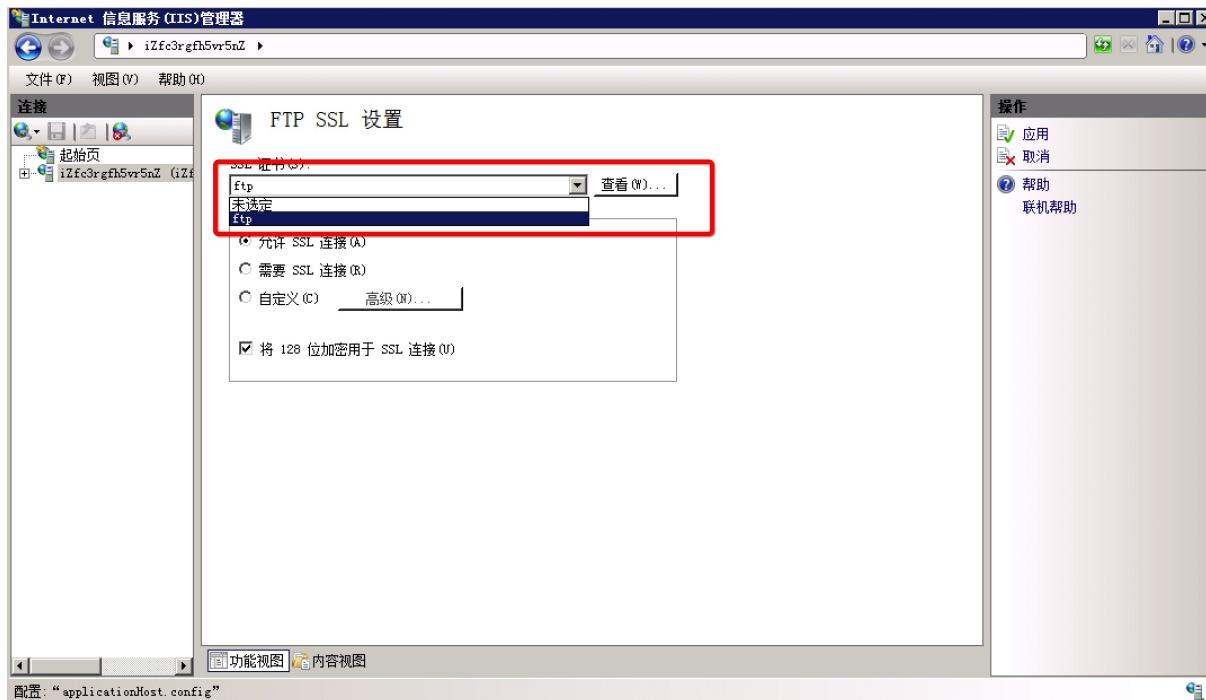


## 7. 启用 SSL 加密传输功能

启用 SSL 加密传输功能，需要先创建服务器证书：

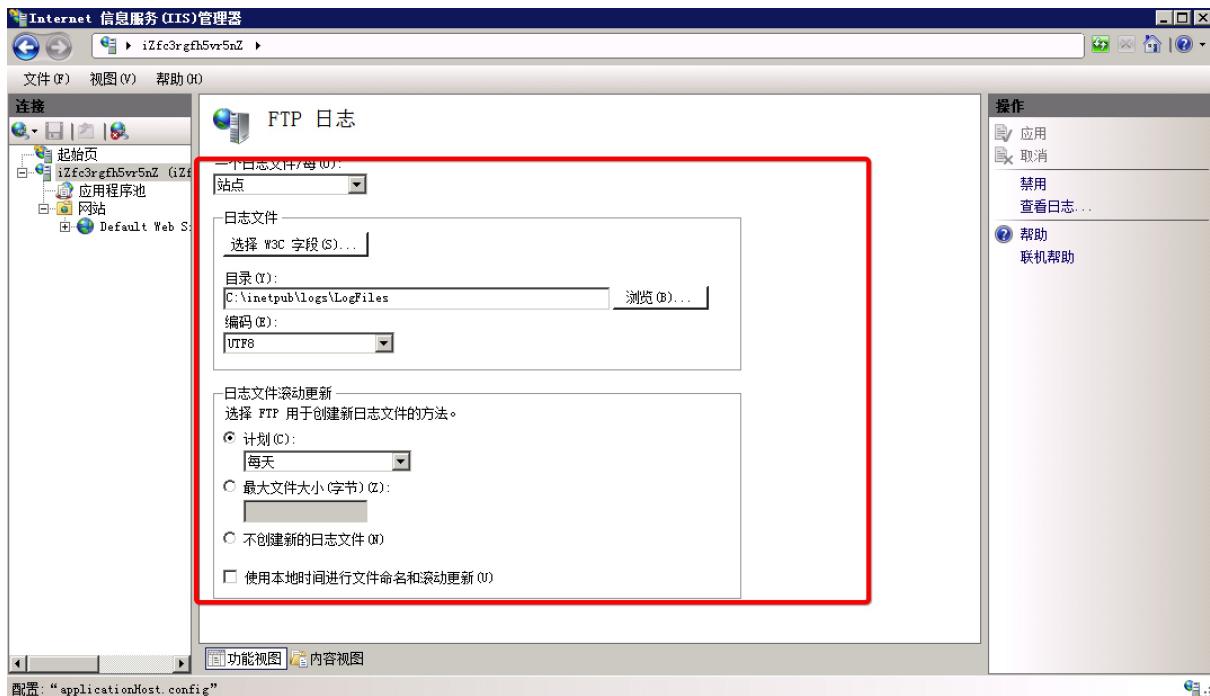


在 FTP SSL 设置中，选定已创建的服务器证书即可。



## 8. 启用日志功能

IIS 中的 FTP 日志是默认启用的，您可以根据磁盘空间情况配置日志空间大小和其他策略。



## FileZilla FTP Server 安全加固

FileZilla FTP Server 是一个非常流行的开源的、免费的 FTP 客户端、服务器端软件，如果您使用该搭建 FTP 服务，FileZilla FTP Server 提供了相关的安全功能，您可以参考 [FileZilla FTP Server 安全加固](#) 方案加固您的 [FileZilla FTP Server](#) 的安全。

## Linux 系统 vsftpd 服务安全加固

### 1. 及时安装更新补丁

在安装更新补丁前，备份您的 vsftpd 应用配置。从 [VSFTPD官方网站](#) 获取最新版本的 vsftpd 软件安装包，完成升级安装。或者，您可以下载最新版 vsftpd 源码包，自行编译后安装更新。您也可以执行 `yum update vsftpd` 命令通过 yum 源进行更新。

### 2. 禁用匿名登录服务

添加一个新用户（test），并配置强密码。例如，执行 `useradd -d /home -s /sbin/nologin test` 命令。

- 其中，`/sbin/nologin` 参数表示该用户不能登录 Linux shell 环境。
- `test` 为用户名。
- 通过 `passwd test` 命令，为该用户配置强密码。密码长度建议八位以上，且密码应包括大小写字母、特殊字符、数字混合体，且不要使用生日、姓名拼音等常见字符串作为密码。

修改配置文件 `vsftpd.conf`，执行 `#vim /etc/vsftpd/vsftpd.conf` 命令。

`anonymous_enable=NO`，将该参数配置为 NO 表示禁止匿名登录，必须要创建用户认证后才能登录 FTP 服务。

### 3. 禁止显示 banner 信息

修改 VSFTP 配置文件 `vsftpd.conf`，设置 `ftpd_banner=Welcome`。重启 vsftpd 服务后，即不显示 banner 信息。

```
>ftp 192.168.10.200
Connected to 192.168.10.200.
220 Welcome
User (192.168.10.200:(none)):
```

## 4. 限制 FTP 登录用户

在 `ftpusers` 和 `user_list` 文件中列举的用户都是不允许访问 FTP 服务的用户（例如 `root`、`bin`、`daemon` 等用户）。除了需要登录 FTP 的用户外，其余用户都应该添加至此拒绝列表中。

## 5. 限制 FTP 用户目录

修改 VSFTP 配置文件 `vsftpd.conf`。

```
chroot_list_enable=YES
chroot_list_file=/etc/vsftpd/chroot_list
```

新建 `/etc/vsftpd/chroot_list` 文件，并添加用户名。例如，将 `user1` 添加至该文件，则 `user1` 登录 FTP 服务后，只允许在 `user1` 用户的 `home` 目录中活动。

## 6. 修改监听地址和默认端口

例如，修改 VSFTP 配置文件 `vsftpd.conf`，设置监听 `1.1.1.1` 地址的 `8888` 端口。

```
listen_address=1.1.1.1
listen_port=8888
```

## 7. 启用日志记录

修改 VSFTP 配置文件 `vsftpd.conf`，启用日志记录。

```
xferlog_enable=YES
xferlog_std_format=YES
```

如果您需要自定义日志存放位置，可以修改 `xferlog_file=/var/log/ftplog`。

## 8. 其他安全配置

修改 VSFTP 配置文件 `vsftpd.conf`。

```
//限制连接数
max_clients=100
max_per_ip=5
//限制传输速度
anon_max_rate=81920
local_max_rate=81920
```

注意： 如果您不需要使用 FTP 服务，建议您关闭该服务。



Docker的快速增长得益于它不仅是一款简单、易用的轻量虚拟环境的工具，而且它还有自己特有的概念，并且越来越多的特性的添加。有时不是很容易获取有关于它们的正确信息，从而造成误解。尤其是，安全隐患常常随着情绪被高估或低估而非基于正确信息。但是为了让Docker成为方便、安全的工具，了解准确的信息去使用Docker是至关重要的。

## 1. 加固主机操作系统

在部署前需要对服务器操作系统进行安全加固，例如：更新所有软件补丁、配置强密码、关闭不必要的服务端口等，具体参考以下手册：

Windows操作系统安全加固

Linux操作系统加固

## 2. 使用强制访问控制策略

使用强制访问控制（mandatory access control (MAC)）对Docker中使用的各种资源根据业务场景的具体分析进行资源的访问的控制。

启用AppArmor或SELinux功能：

```
docker run --interactive --tty --security-opt="apparmor:PROFILENAME" centos /bin/bash  
docker daemon --selinux-enabled
```

## 3. 配置严格的网络访问控制策略

根据实际应用会被外网访问的端口（例如：管理界面、API 2375端口等重要端口）、应用会与外网的交互网络地址、端口、协议等进行梳理，使用iptables或使用ECS安全组策略对网络的出入进行严格的访问控制。

## 4. 不要使用root用户运行docker应用程序

在实际应用程序使用中，有一些必须要使用root用户才能够进行的操作，那么从安全的角度，需要将这一部分与仅使用普通用户权限执行的部分分离解耦。那么如何在docker中使用普通用户权限对不需要root权限执行的部分进行实施呢？

在编写dockerfile时，使用类似如下的命令进行创建一个普通权限的用户，并设置创建的UID为以后运行程序的用户，如下：RUN useradd noroot -u 1000 -s /bin/bash --no-create-homeUSER norootRUN Application\_name

docker命令参考：<https://docs.docker.com/reference/builder/#user><https://docs.docker.com/reference/builder/#run>

## 5. 禁止使用特权

默认情况下，Docker容器是没有特权的，默认一个容器是不允许访问任何设备的；当使用—privileged选项时，则此窗口将能访问所有设备。例如：打开此选项后，即可以进行对Host中的/dev/下有的所有设备进行操作。若非要在host上的某些设备进行访问的话，可以使用—device来进行设备的添加，而不是全部的设备。

## 6.Docker容器资源配额控制

### CPU资源配额控制

#### CPU份额控制

Docker提供了`-cpu-shares`参数，在创建容器时指定容器所使用的CPU份额值。使用示例：

使用命令`docker run -tid -cpu-shares 100 ubuntu:stress`，创建容器，则最终生成的cgroup的cpu份额配置可以下面的文件中找到：

Docker提供了`-cpu-period`、`-cpu-quota`两个参数控制容器可以分配到的CPU时钟周期。`-cpu-period`是用来指定容器对CPU的使用要在多长时间内做一次重新分配，而`-cpu-quota`是用来指定在这个周期内，最多可以有多少时间用来跑这个容器。跟`-cpu-shares`不同的是这种配置是指定一个绝对值，而且没有弹性在里面，容器对CPU资源的使用绝对不会超过配置的值。

`cpu-period`和`cpu-quota`的单位为微秒（ $\mu\text{s}$ ）。`cpu-period`的最小值为1000微秒，最大值为1秒（ $10^6 \mu\text{s}$ ），默认值为0.1秒（100000  $\mu\text{s}$ ）。`cpu-quota`的值默认为-1，表示不做控制。

举个例子，如果容器进程需要每1秒使用单个CPU的0.2秒时间，可以将`cpu-period`设置为1000000（即1秒），`cpu-quota`设置为200000（0.2秒）。当然，在多核情况下，如果允许容器进程需要完全占用两个CPU，则可以将`cpu-period`设置为100000（即0.1秒），`cpu-quota`设置为200000（0.2秒）。

使用示例：

使用命令`docker run -tid -cpu-period 100000 -cpu-quota 200000 ubuntu`，创建容器

#### CPU core控制

对多核CPU的服务器，Docker还可以控制容器运行限定使用哪些CPU内核和内存节点，即使用`-cpuset-cpus`和`-cpuset-mems`参数。对具有NUMA拓扑（具有多CPU、多内存节点）的服务器尤其有用，可以对需要高性能计算的容器进行性能最优的配置。如果服务器只有一个内存节点，则`-cpuset-mems`的配置基本上不会有明显效果。

使用示例：命令`docker run -tid -name cpu1 -cpuset-cpus 0-2 ubuntu`，表示创建的容器只能用0、1、2这三个内核。

#### CPU配额控制参数的混合使用

当上面这些参数中时，`cpu-shares`控制只发生在容器竞争同一个内核的时间片时，如果通过`cpuset-cpus`指定容器A使用内核0，容器B只是用内核1，在主机上只有这两个容器使用对应内核的情况下，它们各自占用全部的内核资源，`cpu-shares`没有明显效果。

`cpu-period`、`cpu-quota`这两个参数一般联合使用，在单核情况或者通过`cpuset-cpus`强制容器使用一个CPU内核的情况下，即使`cpu-quota`超过`cpu-period`，也不会使容器使用更多的CPU资源。

`cpuset-cpus`、`cpuset-mems`只在多核、多内存节点上的服务器上有效，并且必须与实际的物理配置匹配，否则也无法达到资源控制的目的。

### 内存配额控制

和CPU控制一样，Docker也提供了若干参数来控制容器的内存使用配额，可以控制容器的swap大小、可用内存大小等各种内存方面的控制。主要有以下参数：

**memory-swappiness**: 控制进程将物理内存交换到swap分区的倾向，默认系数为60。系数越小，就越倾向于使用物理内存。值范围为0-100。当值为100时，表示尽量使用swap分区；当值为0时，表示禁用容器 swap 功能(这点不同于宿主机，宿主机 swappiness 设置为 0 也不保证 swap 不会被使用)。

**-kernel-memory**: 内核内存，不会被交换到swap上。一般情况下，不建议修改，可以直接参考docker的官方文档。

**-memory**: 设置容器使用的最大内存上限。默认单位为byte，可以使用K、G、M等带单位的字符串。

**-memory-reservation**: 启用弹性的内存共享，当宿主机资源充足时，允许容器尽量多地使用内存，当检测到内存竞争或者低内存时，强制将容器的内存降低到memory-reservation所指定的内存大小。按照官方说法，不设置此选项时，有可能出现某些容器长时间占用大量内存，导致性能上的损失。

**-memory-swap**: 等于内存和swap分区大小的总和，设置为-1时，表示swap分区的大小是无限的。默认单位为byte，可以使用K、G、M等带单位的字符串。如果–memory-swap的设置值小于–memory的值，则使用默认值，为–memory-swap值的两倍。

## 7.不要运行不可信的Docker镜像

不要运行不可信的Docker镜像作为互联网服务器，避免运行不完全理解的Docker镜像作为互联网服务器。

## 8.开启日志记录功能

Docker的日志可以分成两类，一类是stdout标准输出，另外一类是文件日志。

Dockerd支持的日志级别debug、info、warn、error、fatal，默认的日志级别为info。必要的情况下，还需要设置日志级别，这也可以通过配置文件，或者通过启动参数-l或—log-level。

方法一：配置文件/etc/docker/daemon.json

```
{  
  "log-level": "debug"  
}
```

方法二： docker run 的时候指定–log-driver=syslog —log-opt syslog-facility=daemon

## 9.定期安全扫描和更新补丁

在生产环境中使用漏洞扫描工具可以检测镜像中的已知漏洞。容器通常都不是从头开始构建的，所以一定要进行安全扫描，便于及时发现基础镜像中任何可能存在的漏洞，并及时更新补丁。在应用程序交付生命周期中加入漏洞扫描的安全质量控制，防止部署易受攻击的容器。

通过采用以上积极的防范措施，即在整个容器的生命周期中建立和实施安全策略，可以有效的保证一个集成容器环境的安全性。

参考文档：

<https://benchmarks.cisecurity.org/downloads/show-single/index.cfm?file=docker16.100> <https://linux-audit.com/docker-security-best-practices-for-your-vessel-and-containers/>  
[https://d3oypxn00j2a10.cloudfront.net/assets/img/Docker%20Security/WP\\_Intro\\_to\\_container\\_security\\_03.20.20](https://d3oypxn00j2a10.cloudfront.net/assets/img/Docker%20Security/WP_Intro_to_container_security_03.20.20)

15.pdf <https://docs.docker.com/edge/engine/reference/commandline/dockerd/>

# Jenkins服务安全加固

Jenkins早期版本的默认配置下没有安全检查。任何人都可以以匿名用户身份进入Jenkins，执行build操作。然而，对大多数Jenkins应用，尤其是暴露在互联网的应用，安全控制是非常重要的。从Jenkins 2.0开始，其默认配置中启用了许多安全选项，以确保Jenkins环境安全，除非管理员明确禁用某些保护。

本文介绍了Jenkins管理员可操作的各种安全选项，用来加固您的Jenkins服务，以防被黑客攻击。

## Jenkins访问控制

Jenkins访问控制分为：安全域（即认证）与授权策略。

- 安全域决定Jenkins在认证的过程中从哪里寻找用户，默认选项有：Jenkins专有用户数据库，LDAP，和Servlet容器代理。
- 授权策略决定用户登录后可以做什么，默认选项有：任何用户可以做任何事（没有任何限制），安全矩阵，登录用户可以做任何事情，遗留模式，项目矩阵授权策略。

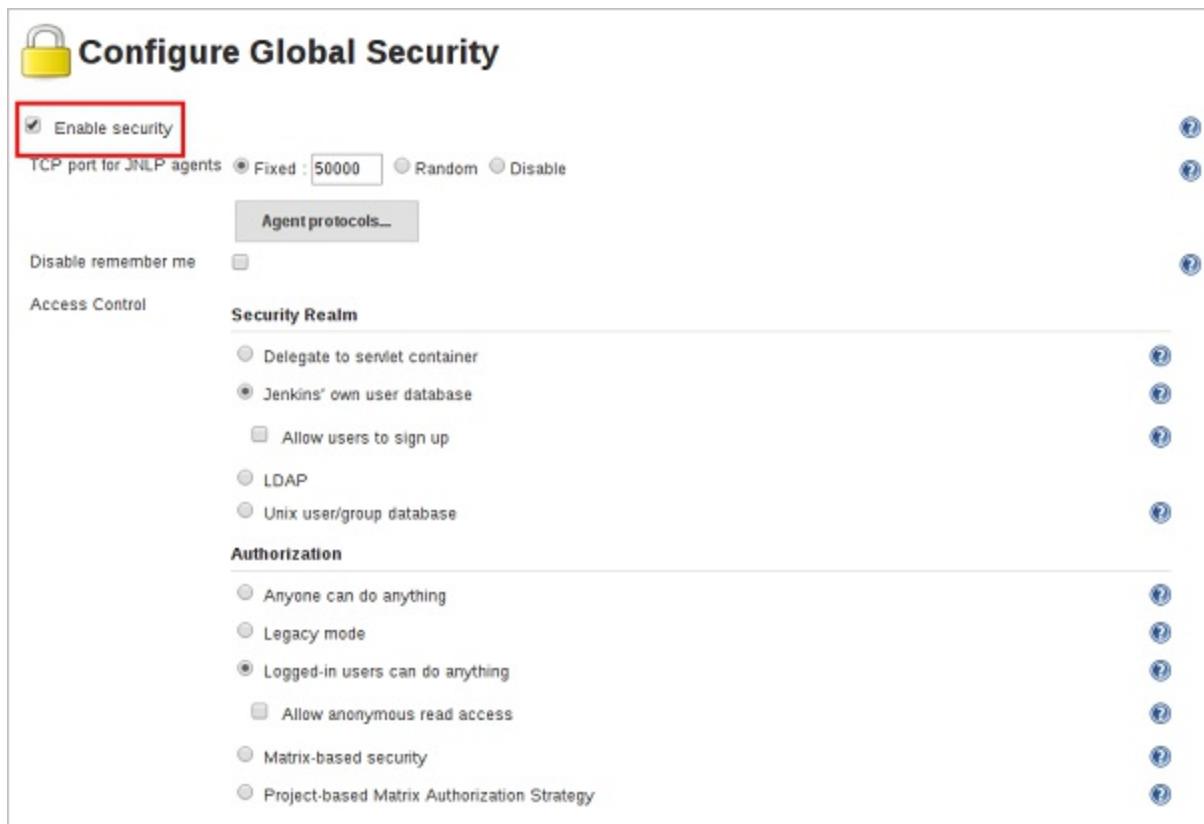
## Jenkins加固方案

### 关注安全漏洞

定期关注[Jenkins官方安全公告](#)，使用或更新到官方最新版本的Jenkins，防止部署存在安全漏洞的版本。

### 启用安全性设置

自2.0版本起，Jenkins默认勾选Enable security复选框。Jenkins管理员可以在Web UI的启用安全性部分启用，配置或禁用适用于整个Jenkins环境的关键安全功能。



默认情况下，匿名用户没有权限，而登录的用户具有完全的控制权。用户可以使用用户名和密码登录，以执行匿名用户不可用的操作。哪些操作要求用户登录取决于所选择的授权策略及其配置。对于任何非本地（测试）Jenkins环境，应始终启用此复选框。

## 配置JNLP TCP端口

Jenkins使用TCP端口与通过JNLP协议启动的代理（如基于Windows的代理）进行通信。截止Jenkins 2.0，默认情况下此端口被禁用。

对于希望使用基于JNLP代理的管理员，以下两种类型的端口可供选用：

- 随机：随机选择JNLP端口，避免Jenkins主机发生冲突。该方式的缺点是在Jenkins主引导期间，难以管理允许JNLP流量的防火墙规则。
- 固定：由Jenkins管理员选择JNLP端口，端口在Jenkins主控器的重新启动之间是一致的。这使得管理防火墙规则更容易，允许基于JNLP的代理连接到主服务器。

## 启用访问控制

访问控制是保护Jenkins环境免受未经授权使用的主要机制。在Jenkins中配置访问控制包括以下三个方面：

- 管理控制台。根据一般的管理方式，管理人员不需要直接在互联网上进行管理，仅需要根据业务资深需求，对管理控制台访问源IP、端口进行限制，防止被恶意人员访问后台。
- 安全域。通知Jenkins环境如何以及在哪里获取用户（或标识）的信息，也被称为“认证”。
- 授权配置。通知Jenkins环境，哪些用户和/或组在多大程度上可以访问Jenkins的哪些方面。

使用安全域和授权配置，可以在Jenkins中轻松地配置非常刚性的身份验证和授权方案。此外，一些插件（如基于角色的授权策略）可以扩展Jenkins的访问控制功能，以支持更细微的身份验证和授权方案。

## 选择合理的授权方式

安全领域或认证表明谁可以访问Jenkins环境，而授权解决的是他们可以在Jenkins环境中访问什么。

默认情况下，Jenkins支持以下的授权选项：

- 所有人都可以控制Jenkins。每个人都可以完全控制Jenkins，包括尚未登录的匿名用户。请勿将本设置用于本地测试Jenkins管理以外的任何其他设置。
- 传统模式。如果用户具有“admin”角色，他们将被授予对系统的完全控制权，否则该用户（包括匿名用户）将仅具有读访问权限。不要将本设置用于本地测试Jenkins管理以外的任何设置。
- 登录用户可以做任何事情。在这种模式下，每个登录的用户都可以完全控制Jenkins。根据高级选项，还可以允许或拒绝匿名用户读取Jenkins的访问权限。此模式有助于强制用户在执行操作之前登录，以便有用户操作的审计跟踪。
- 基于矩阵的安全性。该授权方案可以精确控制哪些用户和组能够在Jenkins环境中执行哪些操作。

基于项目的矩阵授权策略。此授权方案是基于Matrix的安全性的扩展，允许在项目配置屏幕中单独为每个项目定义附加的访问控制列表（ACL）。这允许授予特定用户或组访问指定的项目，而不是Jenkins环境中的所有项目。使用基于项目的矩阵授权定义的ACL是加法的，使得在“配置全局安全性”屏幕中定义的访问权限将与项目特定的ACL组合。

		Overall					
User/group	Administer	Configure	UpdateCenter	Read	RunScripts	Upload	Plugins
Anonymous	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
admin	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

上图表中的每一行表示用户或组（也称为“角色”）。这包括名为“匿名”和“认证”的特殊条目。“匿名”条目表示授予访问Jenkins环境的所有未认证用户的权限。而“已认证”用于向访问环境的所有经过身份验证的用户授予权限。

矩阵中授予的权限是加法的。例如，如果用户“kohsuke”在“开发人员”和“管理员”组中，则授予“kohsuke”的权限将包含授予给“kohsuke”，“开发人员”，“管理员”，“认证”和“匿名”的所有权限。

## 选用与认证和用户管理相关的插件

Jenkins提供一系列与认证和用户管理相关的插件，用户可以根据业务需求安装使用 (<https://plugins.jenkins.io/>)。例如：

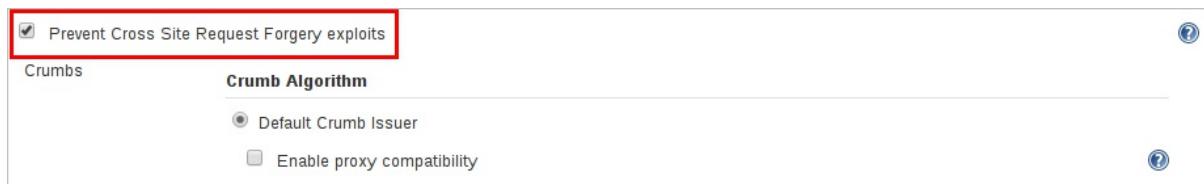
- Active Directory Plugin：允许使用Microsoft Active Directory（即Windows域账号）进行认证。
- Crowd Plugin：允许使用Atlassian Crowd进行认证。

- **Script Security Realm Plugin:** 允许使用自定义的脚本进行认证。
- **Role Strategy Plugin:** 提供了基于角色的授权策略，允许定义全局的和项目集的角色，并为用户分配相应角色。

## 启用CSRF保护

跨站点请求伪造（或CSRF/XSRF）是一种漏洞，它允许未经授权的第三方通过模仿另一个经过身份验证的用户对Web应用程序执行请求。在Jenkins环境的上下文中，CSRF攻击可能允许恶意actor删除项目，更改构建或修改Jenkins的系统配置。

为了防范此类漏洞，自2.0以来所有Jenkins版本，在默认情况下都已启用CSRF保护。



启用该选项后，Jenkins将会在可能更改Jenkins环境中的数据的任何请求上检查CSRF令牌或“crumb”。这包括任何表单提交和对远程API的调用，包括使用“基本”身份验证的表单。

但是，CSRF保护可能会对Jenkins更高级的使用带来挑战，例如：

- 某些Jenkins功能（如远程API）在启用此选项时变得更难使用。
- 通过配置不正确的反向代理访问Jenkins，可能使CSRF HTTP头被从请求中删除，导致受保护的操作失败。
- 未经过CSRF保护测试的过时插件可能无法正常工作。

## 参考资料

有关CSRF漏洞的更多信息，请参考[OWASP网站](#)。

Jenkins提供比较丰富的安全功能，您可以参考更加详细的[官方文档](#)进行配置。

**Kubernetes**提供了许多可以极大地提高应用程序安全性的选项。配置它们要求你熟悉 **Kubernetes** 以及其部署的安全要求。

以下是部署安全的**Kubernetes**应用的建议：

## 确保镜像没有安全漏洞

运行有漏洞的容器使你的环境会遭受损害的风险。许多攻击可以简单地通过将软件升级为没有漏洞的版本来避免。

在部署前，应该确保所有的操作系统软件、**Kubernetes**软件为官方最新版本，防止部署后因为漏洞而造成入侵事件。

在运维过程中，要不断实现Continuous Security Vulnerability Scanning（持续安全漏洞扫描）——容器可能包括含有已知漏洞（CVE）的过时包。新的漏洞每天都会发布，所以这不是一个“一次性”的工作，对镜像持续进行安全评估是至关重要的。

定期对环境进行安全更新，一旦发现运行中容器的漏洞，你应该及时更新镜像并重新部署容器。尽量避免直接更新（例如，‘apt-update’）到正在运行的容器，因为这样打破了镜像与容器的对应关系。

使用**Kubernetes**滚动升级功能升级容器非常简单，该功能允许通过升级镜像到最新版本来逐步更新正在运行的容器。

## 确保在你的环境中只使用授权镜像

如果无法保证只运行符合组织策略的镜像，那么组织会面临运行脆弱甚至恶意容器的危险。从未知的来源下载和运行镜像是危险的，它相当于在生产服务器上运行未知服务商的软件，所以千万别这样做！

使用私有镜像存储你的合法镜像，这样能大量减少可能进入到你的环境的镜像数量。将成安全评估（如漏洞扫描）加入持续集成（CI）中，使其成为构建流程的一部分。

持续集成应确保只使用审查通过的代码来构建镜像。当镜像构建成功后，要对它进行安全漏洞扫描，然后只有当没有发现问题时，镜像才能被推送私有镜像仓库。在安全评估中失败的镜像不应该被推送到镜像仓库中。

**Kubernetes**镜像授权插件的工作已经完成（预计随**kubernetes 1.4**发布）。该插件允许阻止未授权镜像的分发。具体请查看[详情](#)。

## 限制对**Kubernetes**节点的直接访问

应该限制SSH登陆或SSH Key免登**Kubernetes**节点，减少对主机资源未授权的访问。应该要求用户使用“`kubectl exec`”命令，此命令能够在不访问主机的情况下直接访问容器环境。

你可以使用**kubernetes**授权插件来进一步控制用户对资源的访问。它允许设置对指定命名空间、容器和操作的细粒度访问控制规则。

## 修改默认端口

**Kubernets API Server**进程提供**Kuvernetes API**。通常情况下，有一个进程运行在单一**kubernetes-master**节点上。

默认情况，**Kubernetes API Server**提供HTTP的两个端口：

## 1.本地主机端口

- HTTP服务默认端口8080，修改标识—`insecure-port`
- 默认IP是本地主机，修改标识—`insecure-bind-address`
- 在HTTP中没有认证和授权检查
- 主机访问受保护

## 2.Secure Port

- 默认端口6443，修改标识—`secure-port`
- 默认IP是首个非本地主机的网络接口，修改标识—`bind-address` HTTPS服务。
- 设置证书和秘钥的标识，`-tls-cert-file`, `-tls-private-key-file`
- 认证方式，令牌文件或者客户端证书
- 使用基于策略的授权方式

## 3.移除：只读端口

基于安全考虑，会移除只读端口，使用Service Account代替。

## API管理端口访问控制

在某些配置文件中有一个代理（nginx）作为API Server进程运行在同一台机器上。该代理是HTTPS服务，认证端口是443，访问API Server是本地主机8080端口。在这些配置文件里，Secure Port通常设置为6443。

使用[ECS安全组防火墙规则](#)，限制外部HTTPS通过443端口访问。

上面的都是默认配置，每个云提供商可能会有所不同，您可以根据不同的业务场景灵活配置和调整。

## 创建资源间的管理界限

限制用户权限的范围可以减少错误或恶意活动的影响。Kubernetes 命名空间允许将资源划分为逻辑命名组。在一个命名空间中创建的资源对其他命名空间是隐藏的。

默认情况下，用户在Kubernetes 集群中创建的每个资源运行在名称为“default”的默认空间内。你也可以创建额外的命名空间并附加资源和用户给它们。你可以使用Kubernetes 授权插件来创建策略，以便将不同用户的访问请求隔离到不同的命名空间中。

例如：以下策略将允许 ‘alice’ 从命名空间 ‘fronto’ 读取pods。

```
{  
  "apiVersion": "abac.authorization.kubernetes.io/v1beta1",  
  "kind": "Policy",  
  "spec": {  
    "user": "alice",  
    "namespace": "fronto",  
    "resource": "pods",  
    "readonly": true  
  }  
}
```

# 定义资源配置额

运行没有资源限制的容器会将你的系统置于DoS或被其他租户干扰的风险中。为了防止和最小化这些风险，你应该定义资源配置额。

默认情况下，Kubernetes 集群中的所有资源没有对CPU 和内存的使用限制。你可以创建资源配置额策略，并附加到 Kubernetes命名空间中来限制Pod对CPU和内存的使用。

下面的例子将限制命名空间中Pod 的数量为4个，CPU 使用在1和2之间，内存使用在1GB 和 2GB 之间。

compute-resources.yaml:

```
apiVersion: v1
kind: ResourceQuota
metadata:
  name: compute-resources
spec:
  hard:
    pods: "4"
    requests.cpu: "1"
    requests.memory: 1Gi
    limits.cpu: "2"
    limits.memory: 2Gi
```

分配资源配置额到命名空间：

```
kubectl create -f ./compute-resources.yaml --namespace=myspace
```

# 划分网络安全域

在相同的Kubernetes集群上运行不同的应用程序会导致恶意程序攻击其他应用程序的风险。所以网络分割对确保容器只与那些被允许的容器进行通信很重要。

Kubernetes 部署的挑战之一是创建Pod,服务和容器之间的网络分段。原因在于容器网络标识符（IP地址）动态的“天性”，以及容器可以在同一节点或节点间进行通信的事实。

谷歌云平台的用户受益于自动防火墙规则，能够防止跨集群通信。类似的实现可以使用网络防火墙或SDN解决方案部署。这方面的工作由Kubernetes 网络特别兴趣小组（Special Interest Group）完成，这将大大提高 pod到pod 的通信策略。

新的网络策略API应该解决 Pod之间创建防火墙规则的需求，限制容器化可以进行的网络访问。

下面展示了只允许前端（frontend）Pod访问后端（backend）Pod的网络策略：

```
POST /apis/net.alpha.kubernetes.io/v1alpha1/namespaces/tenant-a/networkpolicies
{
  "kind": "NetworkPolicy",
  "metadata": {
    "name": "pol1"
  },
  "spec": {
    "allowIncoming": [
      {
        "from": [
          {
            "pods": { "segment": "frontend" }
          }
        ],
        "toPorts": [{


```

```
        "port": 80,
        "protocol": "TCP"
    }]
},
"podSelector": {
    "segment": "backend"
}
}
```

点击[这里](#)阅读更多网络策略的内容。

## 将安全环境应用到你的**Pods**和容器中

当设计你的容器和**pods**时，确保为你的**pods**，容器和存储卷配置安全环境。安全环境是定义在yaml文件中的一项属性。它控制分配给 pod/容器/存储卷的安全参数。一些重要的参数是：

安全环境设置项	描述
SecurityContext->runAsNonRoot	容器应该以非root用户运行
SecurityContext->Capabilities	控制分配给容器的Linux行为
SecurityContext->readOnlyRootFilesystem	控制容器对root文件系统是否只读
PodSecurityContext->runAsNonRoot	防止root用户作为pod的一部分运行容器

以下是一个具有安全环境参数的pod 定义示例：

```
apiVersion: v1
kind: Pod
metadata:
  name: hello-world
spec:
  containers:
    # specification of the pod's containers
    # ...
    securityContext:
      readOnlyRootFilesystem: true
      runAsNonRoot: true
```

点击[查看更多策略信息。](#)

## API Server认证与授权

API Server权限控制分为三种：Authentication（身份认证）、Authorization(授权)、AdmissionControl(准入控制)。

### 1. 身份认证：

当客户端向Kubernetes非只读端口发起API请求时，Kubernetes通过三种方式来认证用户的合法性。kubernetes中，验证用户是否有权限操作api的方式有三种：证书认证，token认证，基本信息认证。

#### 证书认证：

设置apiserver的启动参数: `--client_ca_file=SOMEFILE`

这个被引用的文件中包含的验证client的证书，如果被验证通过，那么这个验证记录中的主体对象将会作为请求的username。

## Token认证:

设置apiserver的启动参数: `--token_auth_file=SOMEFILE`

token file的格式包含三列: token, username, userid。当使用token作为验证方式时，在对apiserver的http请求中，增加一个Header字段: Authorization，将它的值设置为: Bearer SOMETOKEN。

## 基本信息认证:

设置apiserver的启动参数: `--basic_auth_file=SOMEFILE`

如果更改了文件中的密码，只有重新启动apiserver使其重新生效。其文件的基本格式包含三列: passwork, username, userid。当使用此作为认证方式时，在对apiserver的http请求中，增加一个Header字段: Authorization，将它的值设置为: Basic BASE64ENCODEDUSER:PASSWORD。

## 2. 授权:

在Kubernetes中，认证和授权是分开的，而且授权发生在认证完成之后，认证过程是检验发起API请求的用户是不是他所声称的那个人。而授权过程则判断此用户是否有执行该API请求的权限，因此授权是以认证的结果作为基础的。Kubernetes授权模块应用于所有对APIServer的HTTP访问请求（只读端口除外），访问只读端口不需要认证和授权过程。APIServer启动时默认将authorization\_mode设置为 AlwaysAllow模式，即永远允许。

Kubernetes授权模块检查每个HTTP请求并提取请求上下文中的所需属性（例如: user, resource kind, namespace）与访问控制规则进行比较。任何一个API请求在被处理前都需要通过一个或多个访问控制规则的验证。

目前Kubernetes支持并实现了以下的授权模式（authorization\_mode），这些授权模式可以通过在apiserver启动时传入参数进行选择。

```
--authorization_mode=AlwaysDeny  
--authorization_mode=AlwaysAllow  
--authorization_mode=ABAC
```

AlwaysDeny 模式屏蔽所有的请求（一般用于测试）。AlwaysAllow模式允许所有请求，默认apiserver启动时采用的便是AlwaysAllow模式）。ABAC（Attribute-Based Access Control，即基于属性的访问控制）模式则允许用户自定义授权访问控制规则。

## ABAC模式:

一个API请求中有4个属性被用于用户授权过程:

UserName: String类型，用于标识发起请求的用户。如果不进行认证、授权操作，则该字符串为空。

ReadOnly: bool类型，标识该请求是否仅进行只读操作（GET就是只读操作）。

Kind: String类型，用于标识要访问的Kubernetes资源对象的类型。当访问例如/api/v1beta1/pods等API endpoint时，Kind属性才非空，但访问其他endpoint时，例如/version, /healthz等，Kind属性为空。

**Namespace:** String类型，用于标识要访问的Kubernetes资源对象所在的namespace。

对ABAC模式，在apiserver启动时除了需要传入—authorization\_mode=ABAC选项外，还需要指定—authorization\_policy\_file=SOME\_FILENAME。authorization\_policy\_file文件的每一行都是一个JSON对象，该JSON对象是一个没有嵌套的map数据结构，代表一个访问控制规则对象。一个访问控制规则对象是一个有以下字段的map：

```
user: --token_auth_file指定的user字符串。  
readonly: true或false，如果是true则表明该规则只应用于GET请求。  
kind: Kubernetes内置资源对象类型，例如pods、events等。  
namespace: 也可以缩写成ns。
```

一个简单的访问控制规则文件如下所示，每一行定义一条规则。

```
{"user":"admin"}  
{"user":"alice", "ns": "projectCaribou"}  
{"user":"kubelet", "readonly": true, "kind": "pods"}  
 {"user":"kubelet", "kind": "events"}  
 {"user":"bob", "kind": "pods", "readonly": true, "ns": "projectCaribou"}
```

注：缺省的字段与该字段类型的零值（空字符串，0，false等）等价。

规则逐行说明如下。

- 第一行表明，admin可以做任何事情，不受namespace，资源类型，请求类型的限制。
- 第二行表明，alice能够在namespace “projectCaribou”中做任何事情，不受资源类型，请求类型的限制。
- 第三行表明，kubelet有权限读任何一个pod的信息。
- 第四行表明，kubelet有权限读写任何一个event。
- 第五行表明，Bob有权限读取在namespace “projectCaribou”中所有pod的信息。

一个授权过程就是一个比较API请求中各属性与访问控制规则文件中对应的各字段是否匹配的一个过程。当apiserver接收到一个API请求时，该请求的各属性就已经确定了，如果有一个属性未被设置，则apiserver将其设为该类型的空值（空字符串，0，false等）。匹配规则很简单，如下所示。

如果API请求中的某个属性为空值，则规定该属性与访问控制规则文件中对应的字段匹配。

如果访问控制规则的某个字段为空值，则规定该字段与API请求的对应属性匹配。

如果API请求中的属性值非空且访问控制规则的某个字段值也非空，则将这两个值进行比较，如果相同则匹配，反之则不匹配。

API请求的属性元组（tuple）会与访问控制规则文件中的所有规则逐条匹配，只要有一条匹配则表示匹配成功，如若不然，则授权失败。

更多关于[Kubernetes API](#) 访问控制介绍请点击查看。

## 记录所有的日志

Kubernetes提供基于集群的日志，允许将容器活动日志记录到一个日志中心。当集群被创建时，每个容器的标准输出和标准错误都可以通过运行在每个节点上的Fluentd服务记录到Stackdriver或Elasticsearch中，然后使用Kibana进行查看。

## 总结

**Kubernetes**对创建安全部署提供多种选择，没有适合所有情况的万能解决方案，所以熟悉这些安全选项、了解它们如何提高应用程序安全性是很重要的。

我们推荐这篇文章中提到的安全实践，将**Kubernetes**的灵活配置能力加入到持续集成中，自动将安全性无缝融合到整个流程中。

参考信息：

**Kubernetes**官方最佳实践: <http://blog.kubernetes.io/2016/08/security-best-practices-kubernetes-deployment.html>

**Kubernetes** API文档: <http://docs.kubernetes.org.cn/31.html>

# Hadoop 介绍

Hadoop 是一个由 Apache 基金会所开发的一个开源、高可靠、可扩展的分布式计算框架。

Hadoop 的框架最核心的设计就是 HDFS 和 MapReduce 模块。HDFS 为海量的数据提供了存储，MapReduce 则为海量的数据提供了计算。

- HDFS 是 Google File System (GFS) 的开源实现。
- MapReduce 是一种编程模型，用于大规模数据集(大于1TB)的并行运算。

## Hadoop 环境安全问题

### 1. WebUI 敏感信息泄漏

Hadoop 默认开放了很多端口来提供 WebUI 服务。下表列举了相关的端口信息：

模块	节点	默认端口
HDFS	NameNode	50070
	SecondNameNode	50090
	DataNode	50075
	Backup/Checkpoint node	50105
MapReduce	JobTracker	50030
	TaskTracker	50060

通过访问 NameNode WebUI 管理界面的 50070 端口，可以下载任意文件。而且，如果 DataNode 的默认端口 50075 开放，攻击者可以通过 HDFS 提供的 restful API 对 HDFS 存储的数据进行操作。

### 2. MapReduce 代码执行漏洞

### 3. Hadoop 的第三方插件安全漏洞

- Cloudera Manager 版本 <= 5.5
  - [Cloudera Manager CVE-2016-4949 Information Disclosure Vulnerability](#)
  - [Template rename stored XSS \(CVE-2016-4948\)](#)
  - [Kerberos wizard stored XSS \(CVE-2016-4948\)](#)
  - [Host addition reflected XSS \(CVE-2016-4948\)](#)
- Cloudera HUE 版本 <= 3.9.0
  - [Enumerating users with an unprivileged account \(CVE-2016-4947\)](#)
  - [Stored XSS \(CVE-2016-4946\)](#)
  - [Open redirect](#)

- Apache Ranger 版本 <= 0.5
  - Unauthenticated policy download
  - [Authenticated SQL injection \(CVE-2016-2174\)](#)
- Apache Group Hadoop 2.6.x
  - [Apache Hadoop MapReduce信息泄露漏洞\(CVE-2015-1776\)](#)

## 4. Hive 任意命令/代码执行漏洞

Hive 是建立在 Hadoop 上的数据仓库基础构架。它提供了一系列的工具，可以用来进行数据的提取转化加载（ETL），是一种可以存储、查询和分析存储在 Hadoop 中的大规模数据的机制。Hive 定义了简单的类 SQL 查询语言，称为 HQL，它允许熟悉 SQL 的用户查询数据。同时，HQL 语言也允许熟悉 MapReduce 开发者的开发自定义的 mapper 和 reducer 来处理内建的 mapper 和 reducer 无法完成的复杂的分析工作。

HQL 语言可以通过 transform 命令自定义 Hive 使用的 Map/Reduce 脚本，从而调用 Shell、Python 等语言，导致攻击者可以通过 Hive 接口等相关操作方式直接获取服务器权限。

## 安全加固方案

根据上述 Hadoop 环境安全问题可以发现，对外暴露服务端口会存在严重的安全风险。建议您按照以下方式为 Hadoop 环境进行安全加固。

### 1. 网络访问控制

使用 [安全组防火墙](#) 或本地操作系统防火墙对访问源 IP 进行控制。如果您的 Hadoop 环境仅对内网服务器提供服务，建议不要将 Hadoop 服务所有端口发布到互联网。

### 2. 启用认证功能

启用 Kerberos 认证功能。

### 3. 更新补丁

不定期关注 Hadoop 官方发布的最新版本，并及时更新补丁。

## 更多信息

[Hadoop 所有端口信息](#)

端口	作用
9000	fs.defaultFS, 如 : hdfs://172.25.40.171:9000
9001	dfs.namenode.rpc-address, DataNode会连接这个端口
50070	dfs.namenode.http-address
50470	dfs.namenode.https-address
50100	dfs.namenode.backup.address
50105	dfs.namenode.backup.http-address
50090	dfs.namenode.secondary.http-address, 如 : 172.25.39.166:50090
50091	dfs.namenode.secondary.https-address, 如 : 172.25.39.166:50091
50020	dfs.datanode.ipc.address
50075	dfs.datanode.http.address
50475	dfs.datanode.https.address
50010	dfs.datanode.address, DataNode的数据传输端口
8480	dfs.journalnode.rpc-address
8481	dfs.journalnode.https-address
8032	yarn.resourcemanager.address
8088	yarn.resourcemanager.webapp.address, YARN的http端口
8090	yarn.resourcemanager.webapp.https.address
8030	yarn.resourcemanager.scheduler.address
8031	yarn.resourcemanager.resource-tracker.address
8033	yarn.resourcemanager.admin.address
8042	yarn.nodemanager.webapp.address
8040	yarn.nodemanager.localizer.address
8188	yarn.timeline-service.webapp.address
10020	mapreduce.jobhistory.address
19888	mapreduce.jobhistory.webapp.address
2888	ZooKeeper, 如果是Leader, 用来监听Follower的连接
3888	ZooKeeper, 用于Leader选举
2181	ZooKeeper, 用来监听客户端的连接
60010	hbase.master.info.port, HMaster的http端口
60000	hbase.master.port, HMaster的RPC端口
60030	hbase.regionserver.info.port, HRegionServer的http端口
60020	hbase.regionserver.port, HRegionServer的RPC端口
8080	hbase.rest.port, HBase REST server的端口
10000	hive.server2.thrift.port
9083	hive.metastore.uris

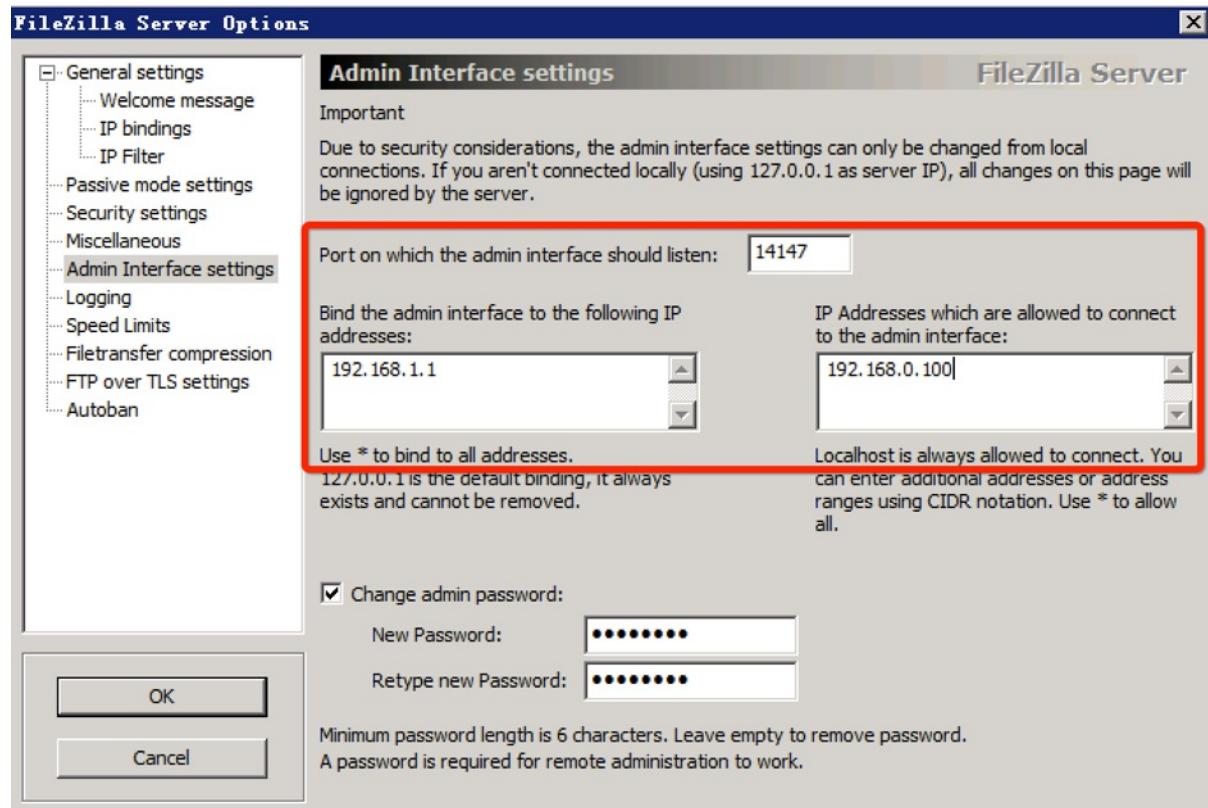
Hadoop safari : Hunting for vulnerabilities Hadoop Default Ports Quick Reference

FileZilla 是一款免费的跨平台 FTP 应用程序，由 FileZilla Client 和 FileZilla Server 组成。本文档依据 FileZilla Server 0.9.59 版本，向您提供一系列简便有效的加固方案，帮助您安全地使用 FileZilla。

注意：本文提到的大部分配置都是通过 FileZilla 服务器的 Edit > Settings > FileZilla Server Options 菜单来实现的。

## 设置管理密码

服务器的管理密码默认为空，建议您设置一个较复杂的密码。例如，应至少包含大小写字母、数字、特殊符号中的任意两种。

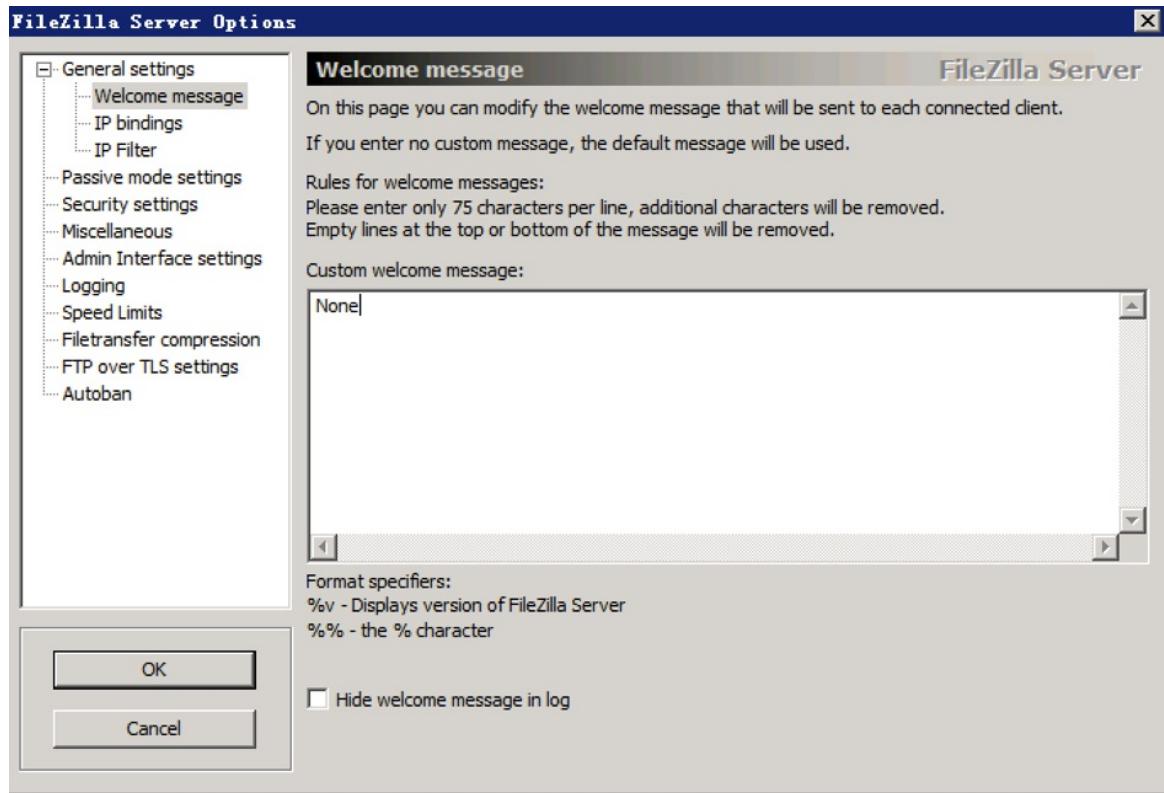


## 修改 Banner 信息

在访问 FTP 服务器时，默认会在 Banner 中显示服务器的版本信息，通过屏蔽版本信息显示，可以加大恶意攻击的时间成本。操作步骤如下：

1. 前往 General settings > Welcome message。

2. 从右侧的 Custom welcome message 输入框中删除 %v 变量，或者直接将全部文本替换为自定义的文字。



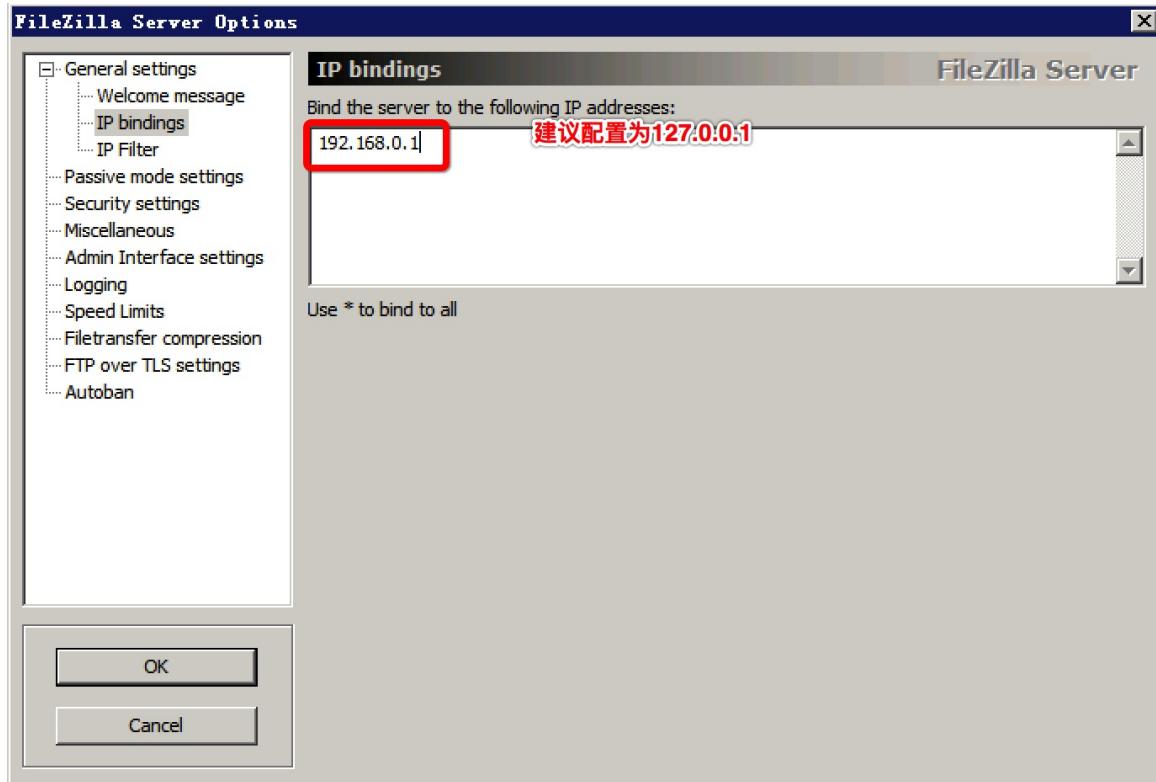
3. 建议勾选下面的 Hide welcome message in log，以减少日志中的垃圾信息。

## 设置监听地址和端口

建议只在一个地址上启用 FTP 服务。例如，若您只需要在内网使用 FTP 服务时，就不必在服务器绑定的公网地址上开启 FTP 服务。操作步骤如下：

1. 前往 General settings > IP Bindings。

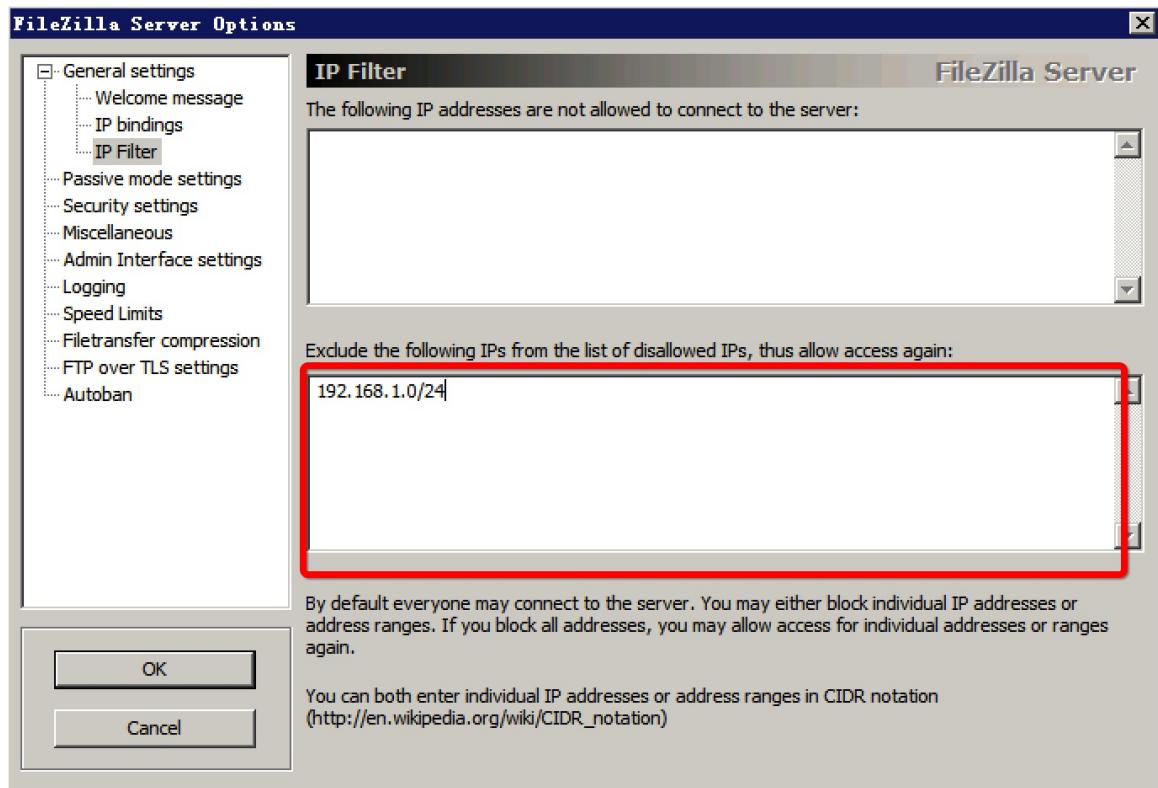
2. 在右侧窗口中将默认的 \* 号修改为指定的地址。



## 使用访问控制

设置全局 IP 过滤器，限制允许访问的 IP 地址。操作步骤如下：

1. 前往 General settings > IP Filters。
2. 在右侧上部窗口中填入要阻止访问的 IP 范围，在右侧下部窗口中填写允许访问的 IP 范围。  
注意：通常采用阻止所有 IP（填写 \*），然后仅允许部分 IP 的方式来进行有效的限制。例如，下图中仅允许 192.168.1.0/24 网段访问 FTP 服务。



另外，FileZilla 服务器也支持用户级和用户组级的 IP 过滤器。前往 **Edit > Users/Groups** 打开对应设置页，在设置页中找到 **IP Filters**，然后选择需要设置的用户，设置允许和拒绝的 IP 即可。设置方法与全局 IP 过滤器相同。

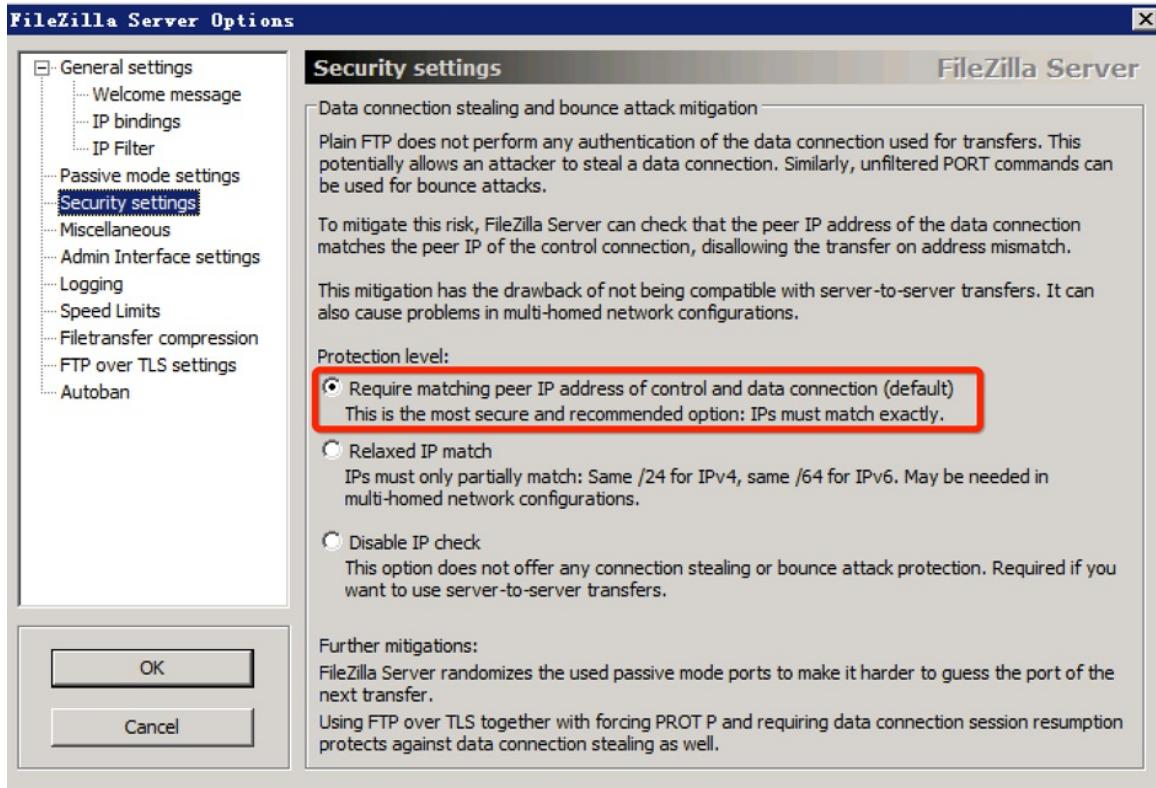
## 开启 **FTP Bounce** 攻击防护

FTP Bounce 攻击是一种利用 FXP 功能的攻击形式，默认情况下服务器未关闭相关功能，建议将相关功能设置为阻止。

如果服务器需要在与某个特定 IP 的服务器之间使用该功能，建议使用 **IPs must match exactly** 选项，然后通过 **IP Filters**（见 使用访问控制）来进行限制来访 IP。操作步骤如下：

1. 前往 **General settings > Security settings**。

2. 如下图所示，默认选项已经启用了需要精确匹配连接地址，建议不要修改。

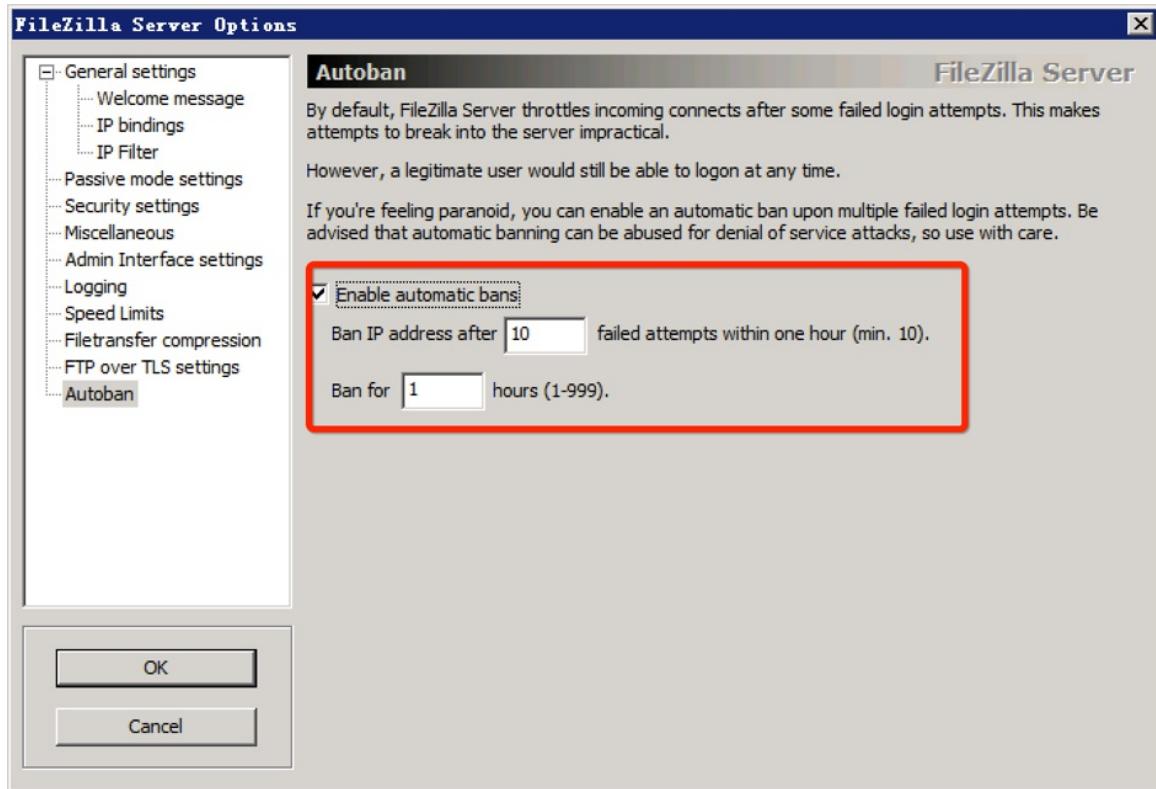


## 配置用户认证策略

默认情况下，当出现多次用户认证失败后，服务器会断开与客户端的连接，但并没有严格的限制策略。通过下面的设置，可以对连续多次尝试登录失败的客户端 IP 进行阻止，干扰其连续尝试行为。

1. 前往 General settings > Autoban。

2. 下图中的设置会对一小时内连续 10 次登录失败的 IP 进行阻止，阻止时长为 1 个小时。



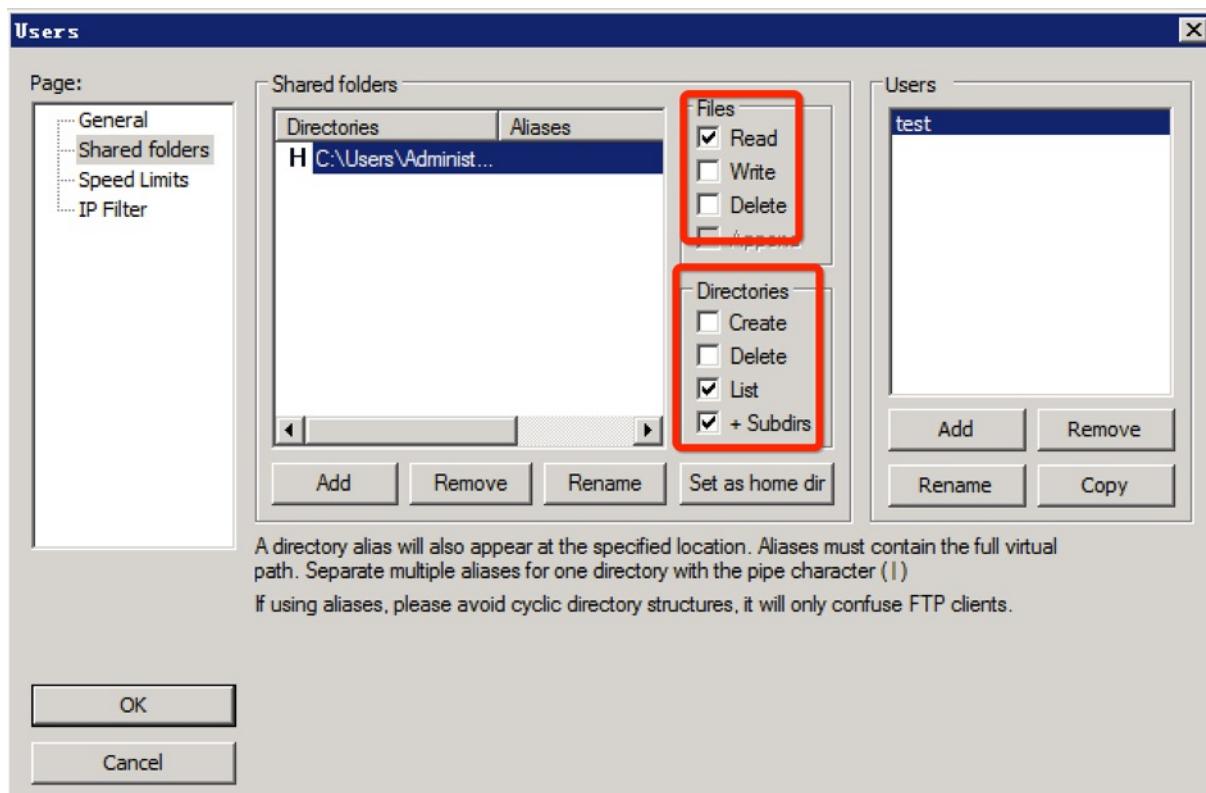
## 提高用户密码复杂度

FileZilla 服务器未提供限制密码复杂度的选项，且服务器用户是由管理员通过管理接口来添加的，用户也无法通过 FTP 命令来修改密码。因此，建议管理员在添加用户时为用户配置复杂的密码。

## 最小化访问授权

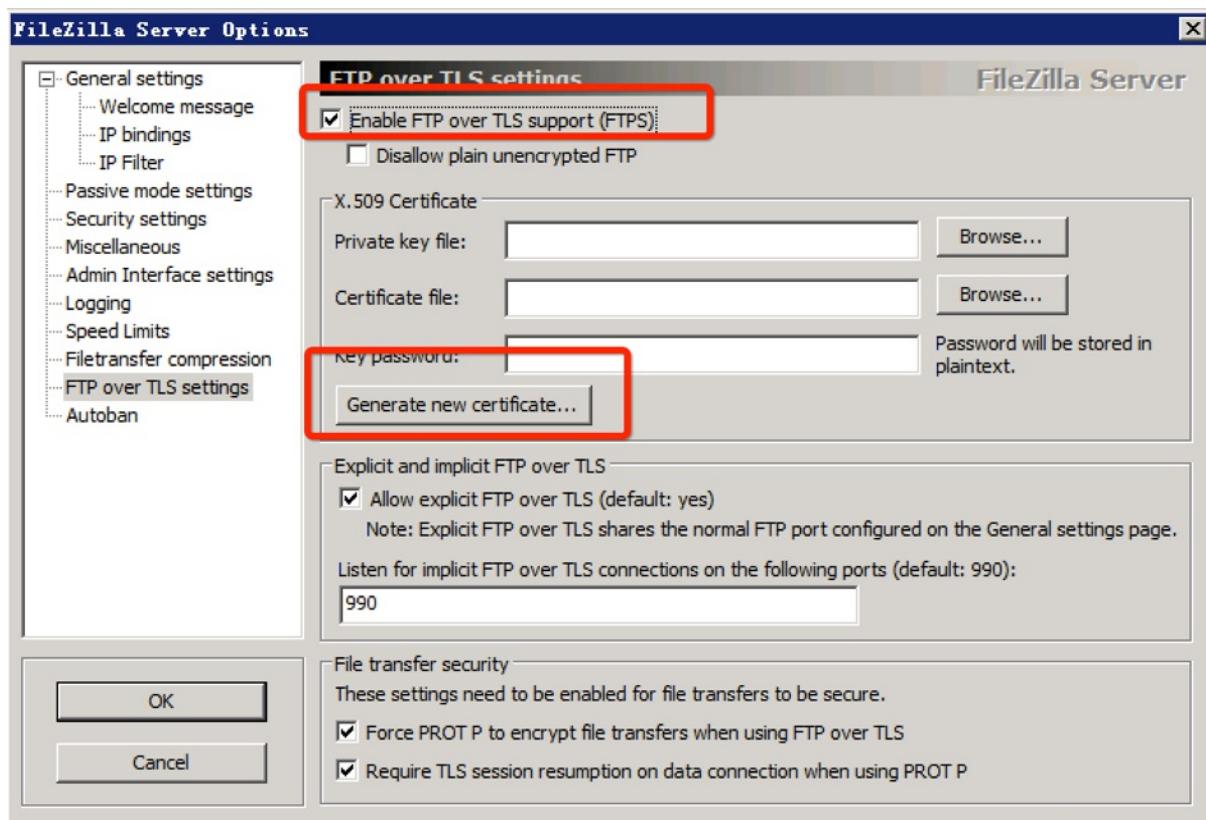
FileZilla 支持目录级别的访问权限设置，可对某个目录设置文件读、写、删除、添加、目录创建、删除、列举等权限。建议根据实际应用需要，结合用户权限最小化原则来分配文件夹的权限。

注意：该操作需要提前添加账号和组后才能配置。

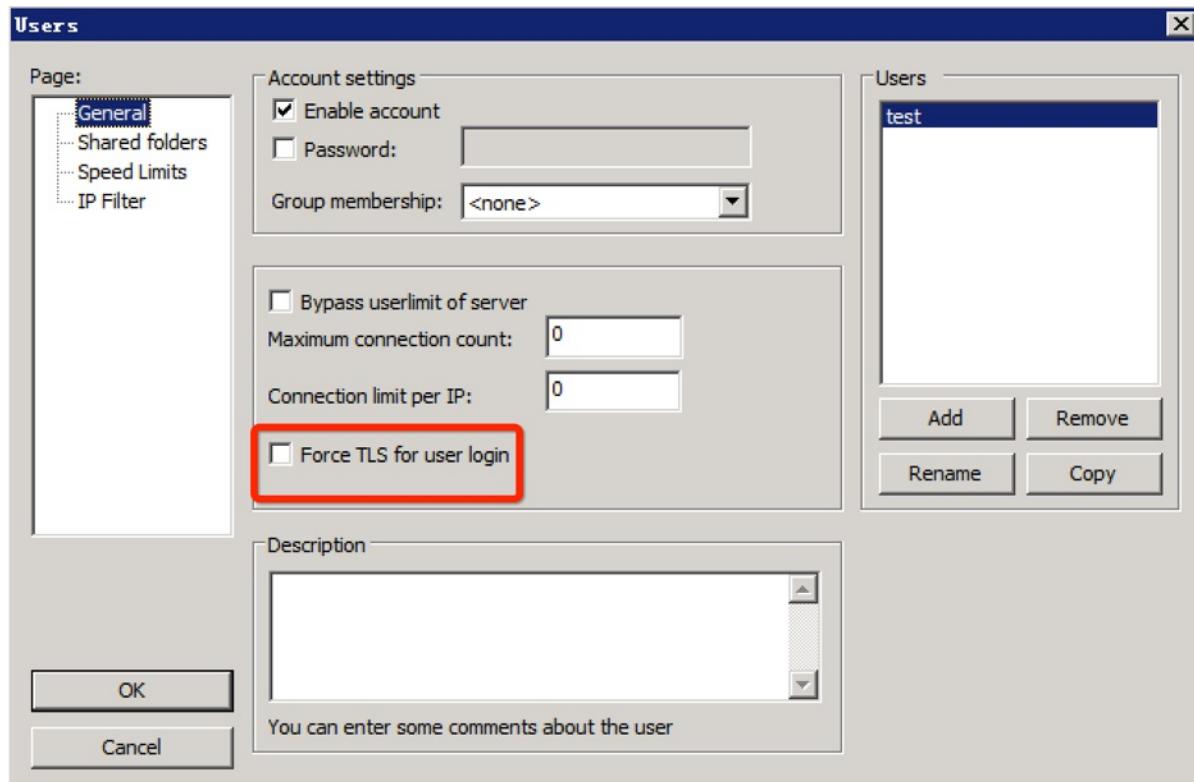


## 启用 TLS 加密认证

FileZilla 服务器支持 TLS 加密功能，用户如果没有证书可以使用自带功能来创建。

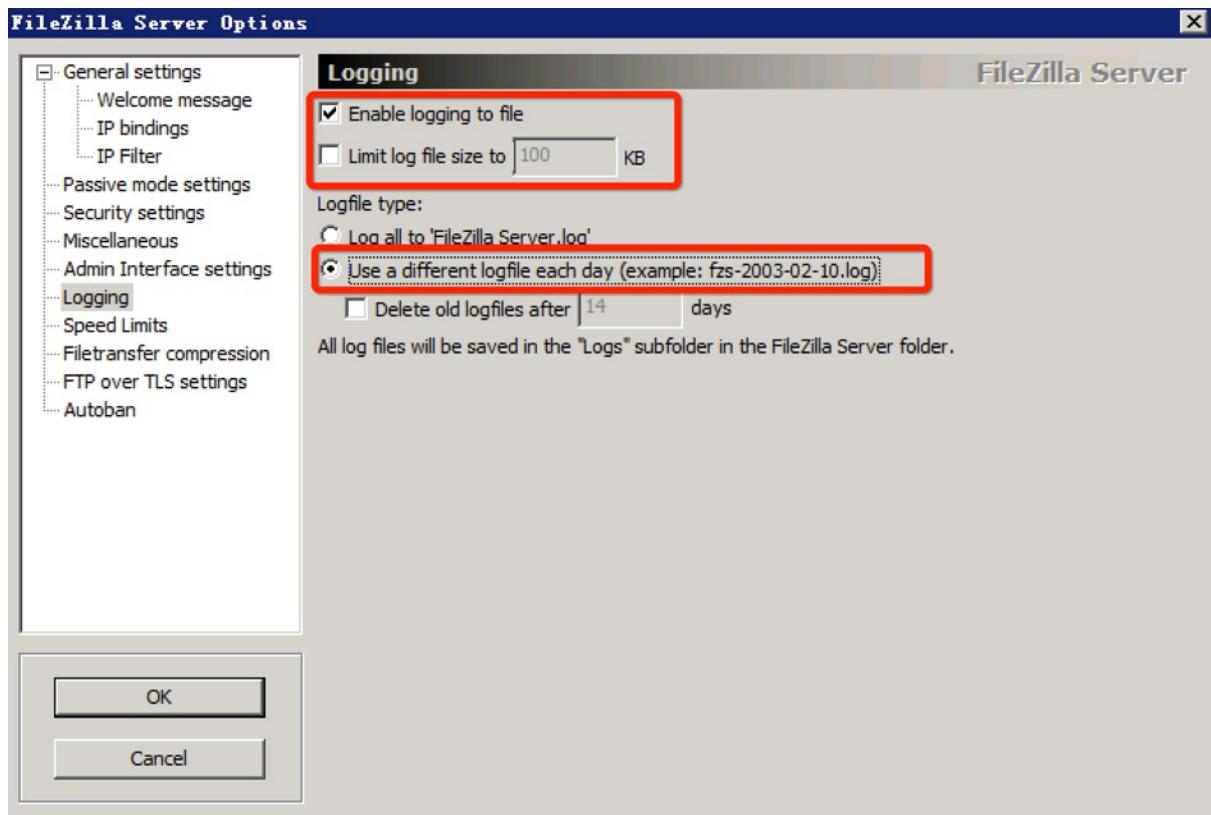


也支持针对单个用户强制启用 TLS 加密访问。

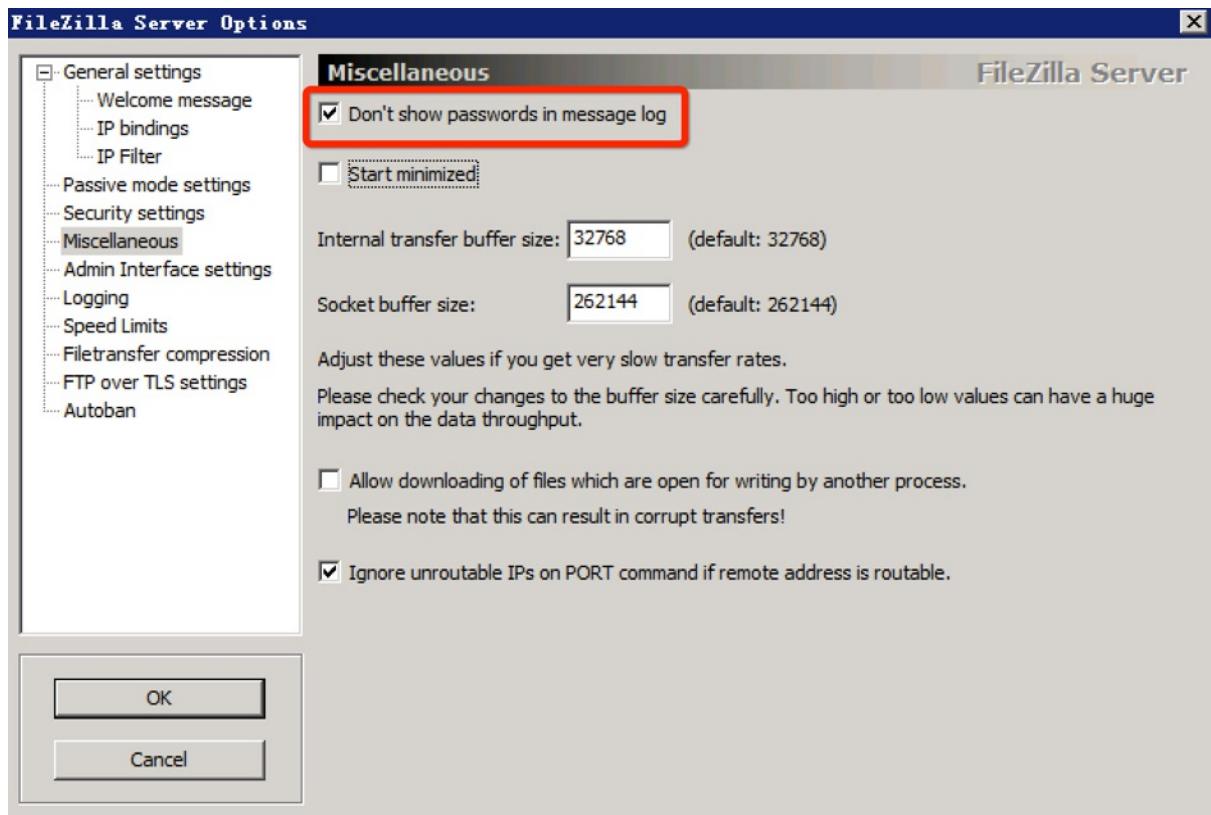


## 启动日志记录

FileZilla 服务器默认未开启日志记录，为了方便对各种事件的追查，建议开启日志记录功能，并将日志设置为每天一个日志文件，避免单文件过大。



默认情况下，日志已经设置不记录用户密码；但在加固的时候应检查此选项，确保其已启用，避免密码泄露。





**Elasticsearch** 是一个基于 **Lucene** 的搜索服务，它提供了 RESTful web 接口的分布式、多用户全文搜索引擎。Elasticsearch 是用 Java 开发的，并作为 Apache 许可条款下的开放源码发布，是第二大最流行的企业搜索引擎。

**Elasticsearch** 应用于云计算中，具有实时搜索、稳定、可靠、快速、安装使用方便等优势；但也存在一些安全隐患：默认安装完成后，**Elasticsearch** 可以使用 9200 端口通告 web 的方式访问查看数据信息。

## 漏洞详情

**Elasticsearch** 中存在以下高危漏洞。

类型	CVE	受影响版本	描述
远程命令执行	CVE-2014-3120	-	<b>Elasticsearch</b> 的脚本执行 (scripting) 功能，可以很方便地对查询出来的数据进行再加工处理。但是，其使用的 MVEL 脚本引擎没有做过任何防护（或者沙盒包装），可以直接执行任意代码。
远程代码执行	-	1.3.0-1.3.7, 1.4.0-1.4	<b>Elasticsearch</b> 使用 Groovy 作为脚本语言，虽然加入了沙盒进行控制，危险的代码会被拦截。但是由于沙盒限制不严格，仅通过黑白名单来判断，导致攻击者可以绕过沙盒，执行远程代码。
未授权访问	-	-	<b>Elasticsearch</b> 在安装了 River 机制之后可以同步多种数据库数据（包括关系型的 MySQL、MongoDB 等）。如果 <code>http://localhost:9200/cat/indices</code> 中 <code>indices</code> 包含了 <code>_river</code> ，则代表 <b>Elasticsearch</b> 已安装 River 机制。而通过泄露的 <code>http://localhost:9200/_rvier/_search</code> URL 地址，攻击者可以获取到敏感信息。

## 漏洞成因与危害

由于 **Elasticsearch** 的 HTTP 连接没有提供任何的权限控制措施，一旦部署在公共网络就容易有数据泄露的风险。

## 安全加固方案

### 使用最新的 **Elasticsearch** 版本

通过正规渠道（如 [Elastic 官网](#)）下载 **Elasticsearch** 的最新版本。

- 下载完成后，将下载文件的 sha1 值和下载时官网页面提供的 sha1 值进行对比，避免下载过程中被恶意攻击者拦截破坏文件，甚至注入恶意代码。
- 不要随便安装第三方的插件，插件有可能引入安全漏洞甚至本身自带后门，需谨慎使用。
- 关注 Elastic 网站，及时更新 Elasticsearch 至最新版本。Elasticsearch 每次版本发布都会优化和改进一部分功能，尤其是安全漏洞的补丁。同时，仔细阅读 Elasticsearch 的版本更新记录。

注意：更新升级前，建议您先进行快照备份，及本地测试。

## 网络访问控制（推荐）

**Elasticsearch** 默认端口是 9200。

- 不要把 Elasticsearch 的 9200 端口服务发布到互联网上。
- 使用 [阿里云安全组防火墙](#) 或本地操作系统防火墙对访问源 IP 进行隔离控制。

## 绑定访问源 IP

进入 config 目录，修改 `elasticsearch.yml` 配置文件中以下参数：

```
network.bind_host: 192.168.0.1
# 设置绑定的 IP 地址，可以是 IPv4 或 IPv6 地址，默认为 0.0.0.0。
network.publish_host: 192.168.0.1
# 设置其它节点和该节点交互的 IP 地址，如果不设置它会自动判断，值必须是个真实的 IP 地址。
network.host: 192.168.0.1
# 同时设置上述两个参数：bind_host 和 publish_host.
```

## 修改默认端口

进入 config 目录，修改 `elasticsearch.yml` 配置文件中以下参数：

```
transport.tcp.port: 9300
# 设置节点间交互的 TCP 端口，默认是 9300。
transport.tcp.compress: true
# 设置是否压缩 TCP 传输时的数据，默认为 false，即不压缩。
http.port: 9200
# 设置对外服务的 HTTP 端口，默认为 9200.
```

## 关闭 HTTP 访问

进入 config 目录，修改 `elasticsearch.yml` 配置文件中以下参数：

```
http.enabled: false
# 是否使用 HTTP 协议对外提供服务，默认为 true，即开启。
```

## 使用 Shield 安全插件

Shield 是 Elastic 公司为 Elasticsearch 开发的一个安全插件。在安装此插件后，Shield 会拦截所有对 Elasticsearch 的请求，并进行认证与加密，保障 Elasticsearch 及相关系统的安全性。Shield 是商业插件，需要 Elasticsearch 的商业许可。第一次安装许可的时候，会提供 30 天的免费试用权限。30 天后，Shield 将会屏蔽 `clusterhealth`, `cluster stats`, `index stats` 等 API，其余功能不受影响。

### 用户认证

使用 Shield 可以定义一系列已知的用户，并用其认证用户请求。这些用户存在于抽象的“域”中。一个“域”可以是下面几种类型：

- LDAP 服务
- ActiveDirectory 服务
- 本地 `esusers` 配置文件（类似 `/etc/passwd`）

### 权限控制

**Shield** 的权限控制包含下面几种元素：

- 被保护的资源 **SecuredResource**: 权限所应用到的对象，比如某个 `index`, `cluster` 等。
- 特权 **Privilege**: 角色对对象可以执行的一种或多种操作，比如 `read`, `write` 等。还可以是 `indices:/data/read/readonly` 等对某种对象特有的操作。
- 许可 **Permissions**: 对被保护的资源拥有的一个或多个特权，如 `read on the "products" index`。
- 角色 **Role**: 一组许可的集成，具有独立的名称。
- 用户 **Users**: 用户实体，可以被赋予多种角色，他们可以对被保护的资源执行相应角色所拥有的各种特权。

## 安装 Shield

执行安装步骤前，请确保满足以下安装环境条件：

- 您安装了 Java7 或更新版本。
- 您将 Elasticsearch 1.5.0+ 解压安装到了本机上。如果您使用 APT 或 YUM 安装，默认的安装目录可能在 `/usr/share/elasticsearch`。

参照以下步骤完成安装：

1. 进入 Elasticsearch 安装目录：

```
cd /usr/share/elasticsearch
```

2. 安装 Elasticsearch 许可插件：

```
bin/plugin -i elasticsearch/license/latest
```

3. 安装 Shield 插件：

```
bin/plugin -i elasticsearch/shield/latest
```

4. 将 Shield 配置文件移动或链接至 `/etc/elasticsearch/shield` 目录中：

```
ln -s /usr/share/elasticsearch/config/shield /etc/elasticsearch/shield
```

说明：Elasticsearch 服务在启动时会在 `/etc/elasticsearch/shield` 目录下寻找 Shield 配置文件，而这些配置文件在安装 Shield 时会出现在 `/usr/share/elasticsearch/config/shield` 中，因此需要将配置文件移动或链接至该目录。

5. 重启 Elasticsearch 服务：

```
service elasticsearch restart
```

6. 新建一个 Elasticsearch 管理员账户，填写新密码：

```
bin/shield/esusers
useradd es_admin -r admin
```

7. 直接使用 RESTFUL API 访问 Elasticsearch 的请求都会被拒绝：

```
curl -XGET 'http://localhost:9200/'
```

需要在请求中添加用户名和密码:

```
curl -u es_admin -XGET 'http://localhost:9200/'
```

更多信息, 请参考:

- [Shield 官方安装指南](#)
- [Shield 官方使用配置指南](#)

## 修改默认的 Elasticsearch 集群名称

Elasticsearch 默认的集群名称是 `elasticsearch`, 请在您的生产环境中将其修改成其他名称。确保在不同的环境和不同的集群下使用不同的名称; 并且在监控集群节点时, 如果有未知节点加入, 一定要及时预警。

## 不要以 root 身份运行 Elasticsearch

不要以 `root` 身份来运行 Elasticsearch, 不要和其他服务共用相同的用户, 并把用户的权限最小化。

应用示例:

```
sudo -u es-user ES_JAVA_OPTS="-Xms1024m -Xmx1024m"  
/opt/elasticsearch/bin/elasticsearch  
正确设置 Elasticsearch 的数据目录
```

请确保为 Elasticsearch 的目录分配了合理的读写权限, 避免使用共享文件系统。确保只有 Elasticsearch 的启动用户才有权访问目录。日志目录也需要正确配置, 避免泄露敏感信息。

## 定期对 Elasticsearch 进行备份

使用 Elasticsearch 提供的备份还原机制, 定期对 Elasticsearch 的数据进行快照备份。

## 禁用批量删除索引

Elasticsearch 支持使用全部 (`_all`) 和通配符 (\*) 来批量删除索引。在生产环境, 该操作存在一定风险, 你可以通过设置 `action.destructive_requires_name: true` 参数来禁用它。

## 启用日志记录功能

Elasticsearch 的 config 文件夹里面有两个配置文件:

- `elasticsearch.yml`: 基本配置文件。
- `logging.yml`: 日志配置文件。由于 Elasticsearch 使用 log4j 来记录日志的, `logging.yml` 中的设置请按普通 log4j 配置文件进行设置。

启用日志功能需要修改 `elasticsearch.yml` 配置文件:

```
path.logs: /path/to/logs  
# 设置日志文件的存储路径, 默认是 Elasticsearch 根目录下的 logs 文件夹
```

---

更多信息

[Elasticsearch 安全方案](#) [Elasticsearch 安全加固](#)

# 漏洞描述

phpMyadmin 是一款流行的数据库管理系统，如果口令设置过于简单，攻击者可以登录到系统，对数据库进行任意增、删、改等高风险恶意操作，从而导致数据泄露或其他入侵事件发生，安全风险高。

# 加固方案

根据通常的业务需求，数据库管理后台主要方便地为数据库管理员、开发人员服务，使用人员范围小，一旦对外网全部开放，将可能会造成严重的数据泄露事件发生。所以在部署安装完毕后，我们建议您对 phpMyadmin 管理控制台进行安全加固，具体如下：

## 1. 网络访问控制策略

限制访问人员 IP 配置 phpMyadmin。

- 您可以使用云服务器提供的 [安全组防火墙策略](#) 对访问源 IP 地址进行限制，避免不必要的人员访问数据库管理后台。

- phpMyadmin 也默认提供了访问控制功能，具体配置如下：

进入 phpMyAdmin 目录，找到 config.inc.php，如果没有，可以将根目录下的 config.sample.inc.php 复制为 config.inc.php。

编辑 config.inc.php，添加下面两行代码，其中 192.168.0.1 是允许访问 phpMyAdmin 的 IP，Access denied 是未经授权访问时的提示信息：

```
?$ip_prefix = '192.168.0.1';
if (substr($_SERVER['REMOTE_ADDR'], 0, strlen($ip_prefix)) != $ip_prefix) die('Access denied');
```

```
<?php
/* vim: set expandtab sw=4 ts=4 sts=4: */
/**
 * phpMyAdmin sample configuration, you can use it as base for
 * manual configuration. For easier setup you can use setup/
 *
 * All directives are explained in documentation in the doc/ folder
 * or at <http://docs.phpmyadmin.net/>.
 *
 * @package PhpMyAdmin
 */
$ip_prefix = '192.168.0.1';
if (substr($_SERVER['REMOTE_ADDR'], 0, strlen($ip_prefix)) != $ip_prefix) die(
'Access denied');

/*
 * This is needed for cookie based authentication to encrypt password in
 * cookie
 */
$cgi['blowfish_secret'] = ''; /* YOU MUST FILL IN THIS FOR COOKIE AUTH! */

/*
```

## 2.账号与口令安全策略

- 设置强度复杂的口令，可以有效避免被攻击者轻易猜解成功，设置完毕后无需重启服务，及时生效；
- 根据使用人员角色对数据库账号进行精细化授权，防止运维风险；

具体可以参见 [MySQL服务加固文档-“授权”部分](#)。

## 3.安全检测和监控

- 阿里云云盾态势感知支持该漏洞的检测和监控告警功能，建议您到控制台开通基础版态势感知，可以实时检测和告警，帮助您及时了解安全漏洞，并指导您快速修复该问题。

如果问题还未能解决，请联系售后技术支持。

- MongoDB服务安全加固
- Memcached服务安全加固
- Redis服务安全加固
- MySQL服务安全加固
- 数据库勒索事件频发，应该如何确保不被入侵勒索？

经检测发现部分阿里云用户存在**MongoDB**数据库未授权访问漏洞，漏洞危害严重，可以导致数据库数据泄露或被删除勒索，从而造成严重的生产事故。为保证您的业务和应用的安全，请根据以下修复漏洞指导方案加固**MongoDB**服务安全。

## 漏洞详情

### 漏洞危害

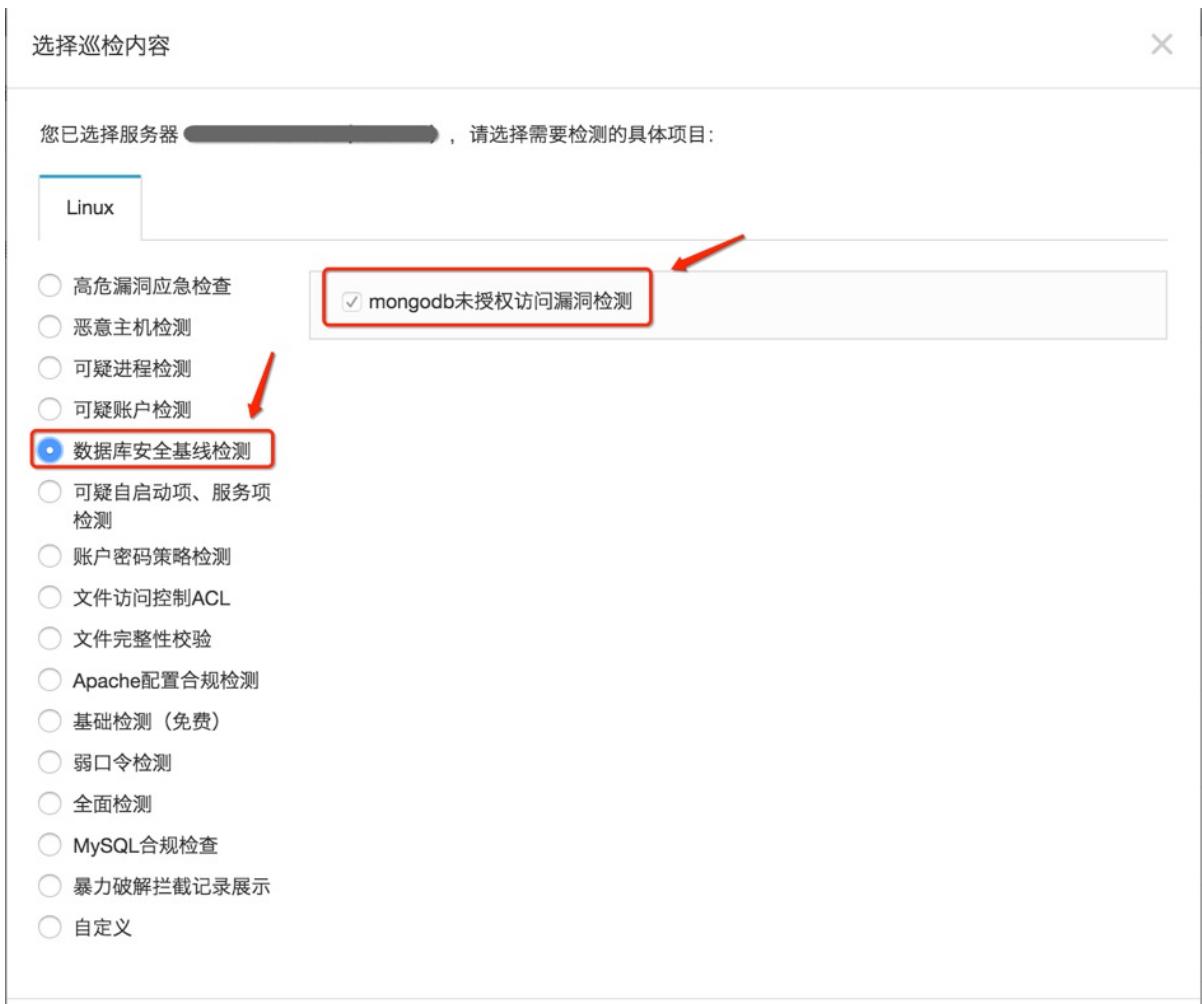
开启**MongoDB**服务后，如不添加任何参数，默认是没有权限验证的。登录的用户可以通过默认端口无需密码对数据库进行任意操作（包括增、删、改、查等高危动作），而且可以远程访问数据库。

### 漏洞成因

安装完**MongoDB**服务后默认有一个**admin**数据库，此时**admin**数据库是空的，没有记录任何权限相关的信息。当**admin.system.users**一个用户都没有时，即使**MongoDB**启动时添加了—**auth**参数，如果没有在**admin**数据库中添加用户，此时不进行任何认证还是可以做任何操作(不管是否以—**auth** 参数启动)，直到在**admin.system.users**中添加一个用户。加固的核心方案是实现只有在**admin.system.users**中添加用户之后，**MongoDB**的认证、授权服务才能生效。

### 漏洞自查

您可以登录到阿里云云盾控制台，使用云盾安骑士**MongoDB**检测是否存在此安全问题。



如果您是MongoDB管理员，也可以使用以下方式检查是否有进一步的入侵行为：

1. 查看MongoDB的日志是否完整，并确认执行删除数据库的源IP地址和时间、行为。
2. 检查MongoDB帐户，查看是否没有添加admin用户的密码（使用`db.system.users.find()`命令）。
3. 检查GridFS，查看是否存储任何文件（使用`db.fs.files.find()`命令）。
4. 检查日志文件，查看有哪些用户访问了MongoDB（使用`show log global`命令）。

## MongoDB未授权漏洞加固方案

重要提示： 如果您需要自己搭建MongoDB数据库，强烈推荐您使用[yum rpm](#)方式安装MongoDB Server 服务。

### 1. 修改默认端口。

修改默认的MongoDB 端口（默认为：TCP 27017）为其他端口。

### 2. 不要把MongoDB服务器直接部署在互联网或者DMZ上。

使用安全组防火墙或本地操作系统防火墙对访问源IP进行控制，如果仅对内网服务器提供服务，建议禁止将MongoDB服务发布到互联网。安全组相当于防火墙功能，默认公网入安全组策略为允许在互联网上访问所有端口。

安全组内实例列表	内网入方向	内网出方向	公网入方向	公网出方向			
安全组规则	授权策略	协议类型	端口范围	授权类型	授权对象	优先级	操作
	允许	全部	-1/-1	地址段访问	0.0.0.0/0	1	克隆   删除

将默认安全组删除，添加拒绝所有规则，即可屏蔽服务。

安全组内实例列表	内网入方向	内网出方向	公网入方向	公网出方向			
安全组规则	授权策略	协议类型	端口范围	授权类型	授权对象	优先级	操作
	拒绝	全部	-1/-1	地址段访问	0.0.0.0/0	1	克隆   删除

根据您的业务情况添加服务的允许规则。

### 添加安全组规则

网卡类型: 公网

规则方向: 入方向

授权策略: 允许

协议类型: TCP

\* 端口范围: 80/80 指定发布的服务端口  
取值范围为1~65535；例如“1/200”、“80/80”。

授权类型: 地址段访问

授权对象: 0.0.0.0/0 指定访问源IP  
请谨慎设置授权对象，根据授权策略的不同，0.0.0.0/0代表允许或拒绝所有IP的访问。[教我设置](#)

优先级: 1  
优先级可选范围为1-100，默认值为1，即最高优先级。

确定 取消

### 3. 使用**--bind\_ip**选项。

该选项可以限制监听接口IP。当在启动MongoDB的时候，使用**--bind\_ip 192.168.0.1**表示启动IP地址绑定，数据库实例将只监听192.168.0.1的请求。

### 4. 启动基于角色的登录认证功能。

在**admin**数据库中创建用户，如用户名**supper**，密码**supWDxsf67%H**（此处为举例说明，请勿使用此账号密码）。在未开启认证的环境下，登录到数据库。

```
[mongodb@rac3 bin]$ ./mongo 127.0.0.1:27028 (此处修改了默认端口)
MongoDB shell version: 2.0.1
connecting to: 127.0.0.1:27028/test
```

切换到**admin**数据库。

```
> use admin
switched to db admin
>
```

创建管理员账号。

```
> db.addUser("supper", "supWDxsf67%H")或 >db.createUser({user:"supper",pwd:"supWDxsf67%H",roles:["root"]})
```

```
{ "n" : 0, "connectionId" : 4, "err" : null, "ok" : 1 }
{
  "user" : "supper",
  "readOnly" : false,
  "pwd" : "51a481f72b8b8218df9fee50b3737c44",
  "_id" : ObjectId("4f2bc0d357a309043c6947a4")
}
```

管理员账号将在**system.users**中。

```
> db.getCollectionNames()
[ "system.indexes", "system.users", "system.version" ]
```

说明

- MongoDB从V3版本开始取消使用**addUser**方法，采用**db.createUser**方法创建用户。
- 账号不要设置为常见账号，密码需要满足一定的复杂度，长度至少八位以上，并包括大小写字母、数字、特殊字符混合体，不要使用生日、姓名、身份证编号等常见密码。

验证用户是否创建成功。

```
> db.auth("supper","supWDxsf67%H")
> exit
bye
```

结束进程，重启MongoDB服务。

```
./mongod --dbpath=/path/mongodb --bind_ip=192.168.0.1 --port=27028 --fork=true logpath=/path/mongod.log &
```

说明

- `admin.system.users`中将会保存比在其它数据库中设置的用户权限更大的用户信息，拥有超级权限，也就是说在`admin`中创建的用户可以对`mongodb`中的其他数据库数据进行操作。
- MongoDB系统中，数据库是由超级用户来创建的，一个数据库可以包含多个用户，一个用户只能在一个数据库下，不同数据库中的用户可以同名。
- 特定数据库（比如DB1）的用户User1，不能够访问其他数据库DB2，但是可以访问本数据库下其他用户创建的数据。
- 不同数据库中同名的用户不能够登录其他数据库，比如DB1、DB2都有user1，以user1登录DB1后，不能够登录到DB2进行数据库操作。
- 在`admin`数据库创建的用户具有超级权限，可以对`mongodb`系统内的任何数据库的数据对象进行操作。
- 使用`db.auth()`可以对数据库中的用户进行验证，如果验证成功则返回1，否则返回0。`db.auth()`只能针对登录用户所属的数据库的用户信息进行验证，不能验证其他数据库的用户信息。更多选项说明请参见[MongoDB – Add Users and Authenticate](#)。

## 5. 禁用HTTP和REST端口。

MongoDB自身带有一个HTTP服务和并支持REST接口（在V2.6以后这些接口默认是关闭的）。MongoDB默认使用默认端口监听Web服务，一般不需要通过Web方式进行远程管理，建议禁用。修改配置文件或在启动的时候选择`-nohttpinterface`参数即可。

```
nohttpinterface = false
```

## 6. 开启日志审计功能。

审计功能可以用来记录用户对数据库的所有相关操作。这些记录可以让系统管理员在需要的时候分析数据库在什么时段发生了什么事情。具体请参见[Mongodb 审计功能](#)。

## 7. 使用SSL加密功能。

MongoDB集群之间以及从客户端连接到MongoDB实例的连接应该使用SSL。使用SSL对性能没有影响并且可以防范类似于man-in-the-middle的攻击。注意MongoDB社区版默认并不支持SSL。您可以选用MongoDB企业版（支持SSL），或者从源码重新编译MongoDB并使用`--ssl`选项来获得SSL功能。具体请参见[Configure mongod and mongos for TLS/SSL](#)。以上所有配置，推荐以配置文件形式保存配置。

```
[mongodb@rac3 bin]$ vim /path/mongod.conf
port=27028-----端口。默认为27017端口，MongoDB的默认服务TCP端口，监听客户端连接。要是端口设置小于1024，比如1021，则需要root权限启动，不能用mongodb帐号启动，（普通帐号即使是27017也起不来）否则报错：[mongo --port=1021 连接]
bind_ip=192.168.0.1-----绑定地址。默认127.0.0.1，只能通过本地连接。进程绑定和监听来自这个地址上的应用连接。要是需要给其他服务器连接，则需要注释掉这个或则把IP改成本机地址，如192.168.200.201[其他服务器用 mongo --host=192.168.200.201 连接]，可以用一个逗号分隔的列表绑定多个IP地址。
logpath=/path/mongod.log-----开启日志审计功能，此项为日志文件路径，可以自定义指定。
pidfilepath=/path/mongod.pid-----进程ID，没有指定则启动时候就没有PID文件。
auth=true-----用户认证，默认false。不需要认证。当设置为true时候，进入数据库需要auth验证，当数据库里没有用户，则不需要验证也可以操作。直到创建了第一个用户，之后操作都需要验证。
logappend=true-----写日志的模式：设置为true为追加。默认是覆盖。如果未指定此设置，启动时MongoDB的将覆盖现有的日志文件。
fork=true-----是否后台运行，设置为true 启动 进程在后台运行的守护进程模式。默认false。
nohttpinterface = false-----是否禁止http接口，即28017 端口开启的服务。默认false，支持。
```

然后，启动MongoDB服务时加载配置文件。

```
[mongodb@rac3 bin]$ ./mongod -f /path/mongod.conf
```

## 8. 对业务关键敏感数据进行加密存储。

建议您梳理业务数据，对关键的敏感数据加密后入库，例如：账号、密码、邮箱地址、手机号码、身份ID等其他数据。加密算法推荐选择国际通用加密算法和多次加盐组合自定义算法，防止加密算法被破解。即使黑客获取数据后，也查看不了数据，通过“看不懂”的数据加密方式将损失降到最低。

## 9. 对数据进行本地异地备份。

完善的备份策略是保证数据安全的最后一根救命稻草。 推荐：可靠的本地备份+远程备份存储方案

- 本地备份 MongoDB 备份方式

```
>mongodump -h dbhost -d dbname -o dbdirectory
-h:
MongoDB所在服务器地址，例如：127.0.0.1，当然也可以指定端口号：127.0.0.1:27017
-d:
需要备份的数据库实例，例如：test
-o:
备份的数据存放位置，例如：c:\data\dump，该目录需要提前建立，在备份完成后，系统自动在dump目录下建立一个test目录，这个目录里面存放该数据库实例的备份数据。
```

### MongoDB 数据恢复

```
mongodb 使用 mongorestore 命令来恢复备份的数据。
语法
mongorestore 命令脚本语法如下：
>mongorestore -h dbhost -d dbname --directoryperdb dbdirectory
-h:
MongoDB所在服务器地址
-d:
需要恢复的数据库实例，例如：test，这个名称也可以和备份时候的不一样，比如test2。
--directoryperdb:
备份数据所在位置，例如：c:\data\dump\test。
--drop:
恢复的时候，先删除当前数据，然后恢复备份的数据。就是说，恢复后，备份后添加修改的数据都会被删除，慎用！
```

### Mongodump 命令可选参数列表如下所示。

语法	描述	实例
mongodump --host HOST_NAME --port PORT_NUMBER	该命令将备份所有MongoDB数据	mongodump --host w3cschool.cc --port 27017
mongodump --dbpath DB_PATH --out BACKUP_DIRECTORY		mongodump --dbpath /data/db/ --out /data/backup/
mongodump --collection COLLECTION --db DB_NAME	该命令将备份指定数据库的集合。	mongodump --collection mycol --db test

### 备份策略

- 全量备份：可以最快的时间快速恢复所有数据，缺点是备份成本大，时间长。
- 全量备份+增量备份：可以较快的恢复所有数据，缺点是恢复时间长，如果增量数据有问题，无法恢复所有数据。
- 搭建从库：直接切换到从库，前提是库的数据安全可靠。

## 10. 使用阿里云MongoDB云服务。

您可以使用更低的成本解决MongoDB的安全问题，阿里云MongoDB云数据库服务从设计之初就重点考虑了安全问题，比如完全不受针对删除数据库的勒索事件的影响。

[阿里云MongoDB云数据库介绍](#)

更多关于MongoDB加固内容请参见[MongoDB security checklist](#)。

## 漏洞描述

**Memcached**是一套常用的key-value缓存系统，由于它本身没有权限控制模块，所以对公网开放的**Memcache**服务很容易被攻击者扫描发现，攻击者通过命令交互可直接读取**Memcached**中的敏感信息。

## 修复方案

因为**Memcached**没有权限控制功能，所以需要对访问来源进行限制。

### **Memcached**服务加固方案

#### 1. 配置访问控制。

建议用户不要将服务发布到互联网上而被黑客利用，可以通过ECS安全组规则或IPtables配置访问控制规则。例如，在Linux环境中运行命令iptables -A INPUT -p tcp -s 192.168.0.2 —dport 11211 -j ACCEPT，在IPtables中添加此规则只允许192.168.0.2这个IP对11211端口进行访问。

#### 2. 绑定监听IP。

如果**Memcached**没有在公网开放的必要，可在**Memcached**启动时指定绑定的IP地址为 127.0.0.1。例如，在Linux环境中运行以下命令： memcached -d -m 1024 -u memcached -l 127.0.0.1 -p 11211 -c 1024 -P /tmp/memcached.pid

#### 3. 最小化权限运行。

使用普通权限账号运行，指定**Memcached**用户。例如，在Linux环境中运行以下命令来运行**Memcached**： memcached -d -m 1024 -u memcached -l 127.0.0.1 -p 11211 -c 1024 -P /tmp/memcached.pid

#### 4. 修改默认端口。

修改默认11211监听端口为11222端口。在Linux环境中运行以下命令：

```
memcached -d -m 1024 -u memcached -l 127.0.0.1 -p 11222 -c 1024 -P /tmp/memcached.pid
```

#### Memcached命令参数说明

- -d 是指启动一个守护进程。
- -m 是指分配给**Memcached**使用的内存数量，单位是MB，以上为1024MB。
- -u 是指运行**Memcached**的用户，推荐使用单独普通权限用户**memcached**，而不要使用root权限账户。
- -l 是指监听的服务器IP地址，例如指定服务器的IP地址为127.0.0.1。
- -p 是用来设置**Memcached**的监听端口，默认端口为11211。建议设置1024以上的端口。
- -c 是指最大运行的并发连接数，默认是1024。可按照您服务器的负载量来设定。
- -P 是指设置保存**Memcached**的pid文件，例如保存在 /tmp/memcached.pid 位置。

#### 5. 备份数据。

为避免数据丢失，升级前请做好备份，或者建立ECS硬盘快照。

## 6. 云盾检测及防护。

云盾态势感知已经支持该漏洞的检测和防护，您可以到[云盾管理控制台](#)开通并使用。

## 一.背景描述

### 1.漏洞描述

Redis 因配置不当存在未授权访问漏洞，可以被攻击者恶意利用。

在特定条件下，如果 Redis 以 root 身份运行，黑客可以给 root 账号写入 SSH 公钥文件，直接通过 SSH 登录受害服务器，从而获取服务器权限和数据。一旦入侵成功，攻击者可直接添加账号用于 SSH 远程登录控制服务器，给用户的 Redis 运行环境以及 Linux 主机带来安全风险，如删除、泄露或加密重要数据，引发勒索事件等。

### 2.受影响范围

在 Redis 客户端，尝试无账号登录 Redis：

```
root@kali:~# redis-cli -h 10.16.10.2
redis 10.16.10.2:6379> keys *
1) "1"
```

从登录结果可以看出，该 Redis 服务对公网开放，且未启用认证。

## 二.修复方案

### 1.网络层加固

指定 Redis 服务使用的网卡

默认情况下，Redis 监听 127.0.0.1。如果仅仅是本地通信，请确保监听在本地。

这种方式可以在一定程度上缓解 Redis 未授权访问的风险（例外情况下，如果 Redis 以 root 用户运行，攻击者借助已有的 webshell，就可以利用该 Redis 来反弹 shell 以实现提权）。

在 redis.conf 文件中找到 # bind 127.0.0.1，将前面的 # 去掉，然后保存。

注意：

该操作需要重启 Redis 才能生效。修改后只有本机才能访问 Redis，也可以指定访问源 IP 来访问 Redis。

```
bind 192.168.1.100 10.0.0.1
```

### 2.设置防火墙策略

如果正常业务中 Redis 服务需要被其他服务器来访问，可以通过 iptables 策略，仅允许指定的 IP 来访问 Redis 服务。

```
iptables -A INPUT -s x.x.x.x -p tcp --dport 6379 -j ACCEPT
```

## 3.账号与认证

设置访问密码

在 redis.conf 中找到 requirepass 字段，去掉其注释，并在后面填上需要的密码。Redis 客户端也需要使用此密码来访问 Redis 服务。

打开 /etc/redis/redis.conf 配置文件：

```
requirepass !QE%^E3323BDWEwwwwe1839
```

确保密码的复杂度，配置完毕后重启服务即可生效。

## 4.服务运行权限最小化

修改 Redis 服务运行账号

请以较低权限账号运行 Redis 服务，并禁用该账号的登录权限。以下操作创建了一个无 home 目录权限，且无法登录的普通账号：

```
useradd -M -s /sbin/nologin [username]
```

注意：该操作需要重启 Redis 才能生效。

## 5.服务精细化授权

隐藏重要命令

Redis 无权限分离，其管理员账号和普通账号无明显区分。攻击者登录后可执行任意操作，因此需要隐藏以下重要命令：FLUSHDB, FLUSHALL, KEYS,PEXPIRE, DEL, CONFIG, SHUTDOWN, BGREWRITEAOF, BGSAVE, SAVE, SPOP, SREM, RENAME,DEBUG, EVAL。

另外，在 Redis 2.8.1 及 Redis 3.x（低于 3.0.2）版本下存在 EVAL 沙箱逃逸漏洞，攻击者可通过该漏洞执行任意 Lua 代码。

下述配置将 config/flushdb/flushall 设置为空，即禁用该命令；也可设置为一些复杂的、难以猜测的名字。

```
rename-command CONFIG ""
rename-command flushall ""
rename-command flushdb ""
rename-command shutdown shutdown_test
```

保存后，执行 /etc/init.d/redis-server restart 重启生效。

## 6.安全补丁

定期关注最新软件版本，并及时升级 Redis 到最新版，防止新漏洞被恶意利用。



数据库管理人员可以参考本文档进行 MySQL 数据库系统的安全配置加固，提高数据库的安全性，确保数据库服务稳定、安全、可靠地运行。

## 漏洞发现

您可以使用安骑士企业版自动检测您的服务器上是否存在 MySQL 漏洞问题，或者您也可以自己排查您服务器上的 MySQL 服务是否存在安全问题。

## 安全加固

### 1. 帐号安全

#### 禁止 **Mysql** 以管理员帐号权限运行

以普通帐户安全运行 mysqld，禁止以管理员帐号权限运行 MySQL 服务。在 /etc/my.cnf 配置文件中进行以下设置。

```
[mysql.server]
user=mysql
```

#### 避免不同用户间共享帐号

参考以下步骤。

a. 创建用户。

```
mysql> mysql> insert into
mysql.user(Host,User,Password,ssl_cipher,x509_issuer,x509_sub
ject) values("localhost","pppadmin",password("passwd"),'',','');

```

执行以上命令可以创建一个 phplamp 用户。

b. 使用该用户登录 MySQL 服务。

```
mysql>exit;
@>mysql -u phplamp -p
@>输入密码
mysql>登录成功
```

#### 删除无关帐号

DROP USER 语句可用于删除一个或多个 MySQL 账户。使用 DROP USER 命令时，必须确保当前账号拥有 MySQL 数据库的全局 CREATE USER 权限或 DELETE 权限。账户名称的用户和主机部分分别与用户表记录的 User 和 Host 列值相对应。

执行DROP USER user;语句，您可以取消一个账户和其权限，并删除来自所有授权表的帐户权限记录。

### 2. 口令

检查账户默认密码和弱密码。口令长度需要至少八位，并包括数字、小写字母、大写字母和特殊符号四类中的至少两种类型，且五次以内不得设置相同的口令。密码应至少每 90 天进行一次更换。

您可以通过执行以下命令修改密码。

```
mysql> update user set password=password('test!p3') where user='root';
mysql> flush privileges;
```

## 3. 授权

在数据库权限配置能力范围内，根据用户的业务需要，配置其所需的最小权限。

查看数据库授权情况。

```
mysql> use mysql;
mysql> select * from user;
mysql>select * from db;
mysql>select * from host;
mysql>select * from tables_priv;
mysql>select * from columns_priv;
```

通过 `REVOKE` 命令回收不必要的或危险的授权。

```
mysql> help REVOKE
Name: 'REVOKE'
Description:
Syntax:
REVOKE
priv_type [(column_list)]
[, priv_type [(column_list)]] ...
ON [object_type]
{
    *
    | *.*
    | db_name.*
    | db_name.tbl_name
    | tbl_name
    | db_name.routine_name
}
FROM user [, user] ...
```

## 4. 开启日志审计功能

数据库应配置日志功能，便于记录运行状况和操作行为。

MySQL服务有以下几种日志类型：

- 错误日志： `-log-error`
- 查询日志： `-log`（可选）
- 慢查询日志： `-log-slow-queries`（可选）
- 更新日志： `-log-update`
- 二进制日志： `-log-bin`

找到 MySQL 的安装目录，在 `my.ini` 配置文件中增加上述所需的日志类型参数，保存配置文件后，重启 MySQL 服务即可启用日志功能。例如，

```
#Enter a name for the binary log. Otherwise a default name will be used.  
#log-bin=  
#Enter a name for the query log file. Otherwise a default name will be used.  
#log=  
#Enter a name for the error log file. Otherwise a default name will be used.  
log-error=  
#Enter a name for the update log file. Otherwise a default name will be used.  
#log-update=
```

该参数中启用错误日志。如果您需要启用其他的日志，只需把对应参数前面的“#”删除即可。

#### 日志查询操作说明

```
执行show variables like 'log_%';命令可查看所有的 log。  
执行show variables like 'log_bin';命令可查看具体的 log。
```

## 5. 安装最新补丁

确保系统安装了最新的安全补丁。

注意：在保证业务及网络安全的前提下，并经过兼容性测试后，安装更新补丁。

## 6. 如果不需要，应禁止远程访问

禁止网络连接，防止猜解密码攻击、溢出攻击、和嗅探攻击。

注意：仅限于应用和数据库在同一台主机的情况。

如果数据库不需要远程访问，可以禁止远程 TCP/IP 连接，通过在 MySQL 服务器的启动参数中添加--skip-networking参数使 MySQL 服务不监听任何 TCP/IP 连接，增加安全性。

您可以使用 安全组 进行内外网访问控制，建议不要将数据库高危服务对互联网开放。

## 7. 设置可信 IP 访问控制

通过数据库所在操作系统的防火墙限制，实现只有信任的 IP 才能通过监听器访问数据库。

```
mysql> GRANT ALL PRIVILEGES ON db.*  
      --> TO 用户名@'IP子网/掩码';
```

## 8. 连接数设置

根据您的机器性能和业务需求，设置最大、最小连接数。

在 MySQL 配置文件（my.conf 或 my.ini）的 [mysqld] 配置段中添加max\_connections = 1000，保存配置文件，重启 MySQL 服务后即可生效。



从2016年12月份到2017年，互联网用户陆续遭受到不同类型数据的勒索事件，笔者统计了一下，大概至少有5中类型的针对数据的勒索事件：

- ElasticSearch勒索事件
- MongoDB勒索事件
- MySQL勒索事件
- Redis勒索事件
- PostgreSQL勒索事件
- 甚至还有针对Oracle的勒索事件

看起来只要是“裸奔”在互联网上的数据库，都未能幸免，成百上千个开放在公网的 MySQL 数据库被劫持，攻击者删除了数据库中的存储数据，并留下勒索信息，要求支付比特币以赎回数据。这对于企业用户，如果发生这样的事情，可能面临一场“灾难”。

## 基线安全问题

从MongoDB和Elasticsearch，以及现在的MySQL数据库勒索案例中，可以发现受害数据库都是因为基线安全问题，才被黑客劫持而勒索。

这些被勒索的自建数据库服务都开放在公网上，并且存在空密码或者弱口令，使得攻击者可以轻易暴力破解出密码，连接数据库，下载并清空数据。再加上不正确的安全组配置，甚至没有配置任何网络访问控制策略，导致问题被放大。

基线安全问题已经成了Web漏洞之外入侵服务器的主要途径，特别是无网络访问控制、默认账号和空口令、默认账号弱口令、后台暴露、后台无口令、未授权访问等情况。错误的配置可以导致相关服务暴露在公网上，成为黑客攻击的目标；加上采用空密码和弱口令，更方便了黑客以极低的攻击成本入侵这些服务。

## 安全隐患自查

找到了原因，就可以“对症下药”，您可以通过自动检测攻击或人工两种方式对您的数据库进行排查。

## 自动检查

阿里云安骑士提供默认的检测策略。您可以登录到控制台，查看安骑士检测的结果，并根据结果，来整改封堵漏洞。

## 工具排查

您也可以使用类似NMap这样的端口扫描工具，直接针对被检查的服务器IP，在服务器外网执行扫描，以检查业务服务器开放在外网的端口和服务。

注意：需要在授权情况下对自身业务进行扫描，不要对其他不相关的业务进行非法扫描，以避免法律风险。

## 安全建议及修复方案

配置严格网络访问控制策略 当您安装完服务，或在运维过程中发现对外开放了服务后，您可以使用Windows自带防火墙功能、Linux系统的iptables防火墙功能来配置必要的访问控制策略。

推荐您使用ECS安全组策略，控制内外网出入的流量，防止暴露更多不安全的服务。

## 对操作系统和服务进行安全加固

必要的安全加固是确保业务在云上安全可靠运行的条件，您可以使用安骑士的自动化检测和人工加固指南对业务服务器进行安全加固。

参考更多的[安全加固指南](#)。

- PHP环境安全加固

随着使用 PHP 环境的用户越来越多，相关的安全问题也变得越来越重要。PHP 环境提供的安全模式是一个非常重要的内嵌安全机制，PHP 安全模式能有效控制一些 PHP 环境中的函数（例如`system()`函数），对大部分的文件操作函数进行权限控制，同时不允许对某些关键文件进行修改（例如`/etc/passwd`）。但是，默认的 `php.ini` 配置文件并没有启用安全模式。

本文档将介绍如何使用 PHP 的安全模式功能来保护您网站的安全性。

## 一、启用 PHP 的安全模式

PHP 环境提供的安全模式是一个非常重要的内嵌安全机制，PHP 安全模式能有效控制一些 PHP 环境中的函数（例如`system()`函数），对大部分的文件操作函数进行权限控制，同时不允许对某些关键文件进行修改（例如`/etc/passwd`）。但是，默认的 `php.ini` 配置文件并没有启用安全模式。

您可以通过修改 `php.ini` 配置文件启用 PHP 安全模式：

```
safe_mode = on
```

## 二、用户组安全

当您启用安全模式后，如果`safe_mode_gid`选项被关闭，PHP 脚本能够对文件进行访问，且相同用户组的用户也能够对该文件进行访问。

因此，建议您将该选项设置为关闭状态：

```
safe_mode_gid = off
```

注意： 该选项参数仅适用于 Linux 操作系统。

如果不进行该设置，您可能无法对服务器网站目录下的文件进行操作。

## 三、安全模式下执行程序主目录

如果启用了安全模式后，想要执行某些程序的时候，可以指定需要执行程序的主目录，例如：

```
safe_mode_exec_dir = /usr/bin
```

一般情况下，如果不需要执行什么程序，建议您不要指定执行系统程序的目录。您可以指定一个目录，然后把需要执行的程序拷贝到这个目录即可，例如：

```
safe_mode_exec_dir = /temp/cmd
```

但是，更推荐您不要执行任何程序。这种情况下，只需要将执行目录指向网页目录即可：

```
safe_mode_exec_dir = /usr/www
```

注意： 执行目录的路径以您实际操作系统目录路径为准。

## 四、安全模式下包含文件

如果您需要在安全模式下包含某些公共文件，您只需要修改以下选项即可：

```
safe_mode_include_dir = /usr/www/include/
```

一般情况下，PHP 脚本中包含的文件都是在程序已经写好的，可以根据您的具体需要进行设置。

## 五、控制 PHP 脚本能访问的目录

使用**open\_basedir**选项能够控制 PHP 脚本只能访问指定的目录，这样能够避免 PHP 脚本访问不应该访问的文件，一定程度上降低了 **phpshell** 的危害。一般情况下，可以设置为只能访问网站目录：

```
open_basedir = /usr/www
```

## 六、关闭危险函数

如果您启用了安全模式，那么可以不需要设置函数禁止，但为了安全考虑，还是建议您进行相关设置。例如，您不希望执行包括**system()**等在内的执行命令的 PHP 函数，以及能够查看 PHP 信息的**phpinfo()**等函数，那么您可以通过以下设置禁止这些函数：

```
disable_functions = system, passthru, exec, shell_exec, popen, phpinfo, escapeshellarg, escapeshellcmd, proc_close, proc_open, dl
```

如果您想要禁止对于任何文件和目录的操作，那么您可以关闭以下文件相关操作。

```
disable_functions = chdir, chroot, dir, getcwd, opendir, readdir, scandir, fopen, unlink, delete, copy, mkdir, rmdir, rename, file, file_get_contents, fputs, fwrite, chgrp, chmod, chown
```

注意：以上设置中只列举了部分比较常用的文件处理函数，您也可以将上面的执行命令函数和这些文件处理函数相结合，就能抵制大部分的 **phpshell** 威胁。

## 七、关闭 PHP 版本信息在 HTTP 头中的泄露

为了防止黑客获取服务器中 PHP 版本的信息，您可以禁止该信息在 HTTP 头部内容中泄露：

```
expose_php = off
```

这样设置之后，黑客在执行telnet 80尝试连接您的服务器的时候，将无法看到 PHP 的版本信息。

## 八、关闭注册全局变量

在 PHP 环境中提交的变量，包括使用 POST 或者 GET 命令提交的变量，都将自动注册为全局变量，能够被直接访问。这对您的服务器是非常不安全的，因此建议您将注册全局变量的选项关闭，禁止将所提交的变量注册为全局变量。

```
register_globals = off
```

注意：该选项参数在 PHP 5.3 以后的版本中已被移除。

当然，如果这样设置之后，获取对应变量的时候就需要采取合理方式。例如，获取 GET 命令提交的变量 var，就需要使用\$\_GET['var']命令来进行获取，在进行 PHP 程序设计时需要注意。

## 九、SQL 注入防护

SQL 注入是一个非常危险的问题，小则造成网站后台被入侵，重则导致整个服务器沦陷。

magic\_quotes\_gpc 选项默认是关闭的。如果打开该选项，PHP 将自动把用户提交对 SQL 查询的请求进行转换（例如，把'转换为\'等），这对于防止 SQL 注入攻击有很大作用，因此建议您将该选项设置为：

```
magic_quotes_gpc = on
```

注意：该选项参数在 PHP 5.4.0 以后的版本中已被移除。

## 十、错误信息控制

一般 PHP 环境在没有连接到数据库或者其他情况下会有错误提示信息，错误信息中可能包含 PHP 脚本当前的路径信息或者查询的 SQL 语句等信息，这类信息如果暴露给黑客是不安全的，因此建议您禁止该错误提示：

```
display_errors = Off
```

如果您确实要显示错误信息，一定要设置显示错误信息的级别。例如，只显示警告以上的错误信息：

```
error_reporting = E_WARNING & E_ERROR
```

注意：强烈建议您关闭错误提示信息。

## 十一、错误日志

建议您在关闭错误提示信息后，对于错误信息进行记录，便于排查服务器运行异常的原因：

```
log_errors = On
```

同时，需要设置错误日志存放的目录，建议您将 PHP 错误日志与 Apache 的日志存放在同一目录下：

```
error_log = /usr/local/apache2/logs/php_error.log
```

注意：该文件必须设置允许 Apache 用户或用户组具有写的权限。



- Apache服务安全加固
- Tomcat服务安全加固
- 网站被植入WebShell的解决方案

## 一、账号设置

以专门的用户帐号和用户组运行 **Apache** 服务。

1.根据需要，为 **Apache** 服务创建用户及用户组。如果没有设置用户和组，则新建用户，并在 **Apache** 配置文件中进行指定。

创建 **Apache** 用户组。

```
groupadd apache
```

创建 **Apache** 用户并加入 **Apache** 用户组。

```
useradd apache -g apache
```

将下面两行设置参数加入 **Apache** 配置文件 **httpd.conf** 中：

```
User apache  
Group apache
```

2.检查 **httpd.conf** 配置文件中是否允许使用非专用账户（如 **root** 用户）运行 **Apache** 服务。

默认设置一般即符合要求。Linux 系统中默认使用 **apache** 或者 **nobody** 用户，Unix 系统默认使用 **daemon** 用户。

## 二、授权设置

严格控制 **Apache** 主目录的访问权限，非超级用户不能修改该目录中的内容。

1.**Apache** 的主目录对应于 **Apache Server** 配置文件 **httpd.conf** 中的 **Server Root** 控制项，应设置为：

```
Server Root /usr/local/apache
```

- 判定条件：非超级用户不能修改该目录中的内容。
- 检测操作：尝试进行修改，看是否能修改该目录中的内容。

该目录一般设置为 **/etc/httpd** 目录，默认情况下属主为 **root** 用户，其它用户不能修改该目录中的文件。默认设置一般即符合要求。

2.严格设置配置文件和日志文件的权限，防止未授权访问。

- 执行 **chmod 600 /etc/httpd/conf/httpd.conf** 命令设置配置文件为属主可读写，其他用户无读写权限。
- 执行 **chmod 644 /var/log/httpd/\*.log** 命令设置日志文件为属主可读写，其他用户拥有只读权限。

注意：

- **/etc/httpd/conf/httpd.conf** 配置文件的默认权限是**644**，可根据需要修改权限为**600**。
- **/var/log/httpd/\*.log** 日志文件的默认权限为**644**，默认设置即符合要求。

## 三、日志设置

**Apache** 设备应配置日志功能，对运行错误、用户访问等事件进行记录，记录内容包括时间，用户使用的 IP 地址等内容。

修改 **httpd.conf** 配置文件，设置日志记录文件、记录内容、记录格式。

错误日志：

```
LogLevel notice #日志的级别  
ErrorLog /.../logs/error_log #日志的保存位置(错误日志)
```

访问日志：

```
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Accept}i\" \"%{Referer}i\" \"%{User-Agent}i\""  
combined  
CustomLog /.../logs/access_log combined (访问日志)
```

注意：

- **ErrorLog** 指令设置错误日志文件名和位置。错误日志是最重要的日志文件。**Apache httpd** 程序将在这个文件中存放诊断信息和处理请求中出现的错误。若要将错误日志传送到 **Syslog**，则执行 **ErrorLog syslog** 命令。
- **CustomLog** 指令指定了保存日志文件的具体位置以及日志的格式。访问日志中会记录服务器所处理的所有请求。
- **LogFormat** 命令用于设置日志格式，建议设置为 **combined** 格式。
- **LogLevel** 命令用于调整记录在错误日志中的信息的详细程度，建议设置为 **notice**。日志的级别，默认是 **warn** 级别，**notice** 级别比较详细，但在实际中由于日志会占用大量硬盘空间。

## 四、禁止访问外部文件

1. 禁止 Apache 访问 Web 目录之外的任何文件。

修改 **httpd.conf** 配置文件。

```
Order Deny,Allow  
Deny from all
```

2. 设置可访问的目录。

```
Order Allow,Deny  
Allow from /web
```

说明： 其中 **/web** 为网站根目录。

3. 默认配置如下，可根据您的业务需要进行设置。

```
Options FollowSymLinks  
AllowOverride None
```

## 五、禁止目录列出

目录列出会导致明显信息泄露或下载，建议禁止 Apache 列表显示文件。在 /etc/httpd/httpd.conf 配置文件中删除 Options 的 Indexes 设置即可。

1.修改 httpd.conf 配置文件：

```
#Options Indexes FollowSymLinks #删掉Indexes
Options FollowSymLinks
AllowOverride None
Order allow,deny
Allow from all
```

将 Options Indexes FollowSymLinks 中的 Indexes 去掉，就可以禁止 Apache 显示该目录结构。Indexes 的作用就是当该目录下没有 index.html 文件时，自动显示目录结构。

2.重新启动 Apache 服务。

## 六、错误页面重定向

Apache 错误页面重定向功能可以防止敏感信息泄露。

1.修改 httpd.conf 配置文件：

```
ErrorDocument 400 /custom400.html
ErrorDocument 401 /custom401.html
ErrorDocument 403 /custom403.html
ErrorDocument 404 /custom404.html
ErrorDocument 405 /custom405.html
ErrorDocument 500 /custom500.html
```

注意：Customxxx.html 为要设置的错误页面。

2.重新启动 Apache 服务。

注意：此项配置需要应用系统设有错误页面，或者不在 httpd 中设置，而完全由业务逻辑实现。

## 七、拒绝服务防范

根据业务需要，合理设置 session 时间，防止拒绝服务攻击。

1.修改 httpd.conf 配置文件：

```
Timeout 10 #客户端与服务器端建立连接前的时间间隔
KeepAlive On
KeepAliveTimeout 15 #限制每个 session 的保持时间是 15 秒 注：此处为一建议值，具体的设定需要根据现实情况。
```

2.重新启动 Apache 服务。

注意：默认值为 Timeout 120，KeepAlive Off，KeepAliveTimeout 15，该项设置涉及性能调整。

## 八、隐藏 Apache 的版本号

隐藏 Apache 的版本号及其它敏感信息。

修改 httpd.conf 配置文件:

```
ServerSignature Off ServerTokens Prod
```

## 九、关闭 TRACE 功能

关闭 TRACE 功能，防止 TRACE 方法被访问者恶意利用。

在 /etc/httpd/conf/httpd.conf 配置文件中添加以下设置参数:

```
TraceEnable Off
```

注意：该参数适用于 Apache 2.0 以上版本。

## 十、禁用 CGI

如果服务器上不需要运行 CGI 程序，建议禁用 CGI。

如果没有CGI程序，可以修改 /etc/httpd/conf/httpd.conf 配置文件，把 cgi-bin 目录的配置和模块都进行注释。

```
#LoadModule cgi_module modules/mod_cgi.so
#ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"
#
#AllowOverride None
# Options None
#Order allow,deny
#Allow from all
#
```

## 十一、绑定监听地址

服务器有多个 IP 地址时，只监听提供服务的 IP 地址。

1. 执行以下命令查看是否绑定 IP 地址。

```
cat /etc/httpd/conf/httpd.conf|grep Listen
```

2. 修改 /etc/httpd/conf/httpd.conf 配置文件。

```
Listen x.x.x.x:80
```

监听功能默认监听所有地址，如果服务器只有一个 IP 地址可不修改该项设置，如果有多个 IP 可根据需要进行设置。

## 十二、删除缺省安装的无用文件

删除缺省安装的无用文件。

- 删除缺省 HTML 文件:

```
# rm -rf /usr/local/apache2/htdocs/*
```

- 删除缺省的 CGI 脚本:

```
# rm -rf /usr/local/apache2/cgi-bin/*
```

- 删除 Apache 说明文件:

```
# rm -rf /usr/local/apache2/manual
```

- 删除源代码文件:

```
# rm -rf /path/to/httpd-2.2.4*
```

- 删除 CGI。

可根据实际情况删除，一般情况下 /var/www/html /var/www/cgi-bin 默认就是空的。

注意：根据安装步骤不同和版本不同，某些目录或文件可能不存在或位置不同。

## 十三、禁用非法 **HTTP** 方法

禁用 PUT、DELETE 等危险的 HTTP 方法。

修改 httpd.conf 配置文件，只允许 get、post 方法。

```
<Location />
<LimitExcept GET POST CONNECT OPTIONS>
    Order Allow,Deny
    Deny from all
</LimitExcept>
</Location>
```

您可根据需要进行设置，如果需要用到 PUT 或 Delete 等 HTTP 方法的话，在 /etc/httpd/conf/httpd.conf 配置文件中相应添加即可。

Tomcat服务默认启用了管理后台功能，使用该后台可直接上传 war 文件包对站点进行部署和管理。由于运维人员的疏忽，可能导致管理后台存在空口令或者弱口令的漏洞，使得黑客或者不法分子可以利用该漏洞直接上传 Webshell 脚本导致服务器沦陷。

通常 Tomcat 后台管理的 URL 地址为 <http://IP:8080/manager/html/>，如下图所示：



黑客通过猜解到的口令登录 Tomcat 管理后台后，可以上传 Webshell 脚本导致服务器被入侵。

## 安全加固方案

由于此类型漏洞可能对业务系统造成比较严重的危害，建议您针对 Tomcat 管理后台进行以下安全加固配置。

### 1. 网络访问控制

- 如果您的业务不需要使用 Tomcat 管理后台管理业务代码，建议您使用安全组防火墙功能对管理后台 URL 地址进行拦截，或直接将 Tomcat 部署目录中 webapps 文件夹中的 manager、host-manager 文件夹全部删除，并注释 Tomcat 目录中 conf 文件夹中的 tomcat-users.xml 文件中的所有代码。
- 如果您的业务系统确实需要使用 Tomcat 管理后台进行业务代码的发布和管理，建议为 Tomcat 管理后台配置强口令，并修改默认 admin 用户，且密码长度不低于10位，必须包含大写字母、特殊符号、数字组合。

### 2. 开启 Tomcat 的访问日志

修改 conf/server.xml 文件，将下列代码取消注释：

```
<Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs"
prefix="localhost_access_log." suffix=".txt" pattern="common" resolveHosts="false"/>
```

启用访问日志功能，重启 Tomcat 服务后，在 tomcat\_home/logs 文件夹中就可以看到访问日志。

### 3. Tomcat 默认帐号安全

修改 Tomcat 安装目录 conf 下的 tomcat-user.xml 文件，重新设置复杂口令并保存文件。重启 Tomcat 服务后，新口令即生效。

## 4. 修改默认访问端口

修改 conf/server.xml 文件把默认的 8080 访问端口改成其它端口。

## 5. 重定向错误页面

修改访问 Tomcat 错误页面的返回信息，在 webapps\manger 目录中创建相应的401.html、404.htm、500.htm 文件，然后在 conf/web.xml 文件的最后一行之前添加下列代码：

```
<error-page>
    <error-code>401</error-code>
    <location>/401.htm</location>
</error-page>
<error-page>
    <error-code>404</error-code>
    <location>/404.htm</location>
</error-page>
<error-page>
    <error-code>500</error-code>
    <location>/500.htm</location>
</error-page>
```

## 6. 禁止列出目录

防止直接访问目录时由于找不到默认页面，而列出目录下的文件的情况。

在 web.xml 文件中，将 `<param-name>listings</param-name>` 改成 `<param-name>false</param-name>`。

## 7. 删除文档和示例程序

删除 webapps 目录下的 docs、examples、manager、ROOT、host-manager 文件夹。

很多站长饱受自己的小站被上传了webshell导致数据泄露或其他安全事件的困扰，下面将为大家详细介绍webshell及处置、防御方式。

## 一.什么是WebShell？

“web”的含义是显然需要服务器开放web服务，“shell”的含义是取得对服务器某种程度上操作权限。webshell常常被称为匿名用户（入侵者）通过网站端口对网站服务器的某种程度上操作的权限。

简单理解： WebShell通常是以asp、php、jsp、asa或者cgi等网页文件形式存在的一种命令执行环境，也可以称为一种网页后门。黑客在入侵网站后，通常会将WebShell后门文件与网站服务器WEB目录下正常的网页文件混在一起，然后就可以使用浏览器来访问这些后门，得到命令执行环境，以达到控制网站或者WEB系统服务器的目的。

webshell中由于需要完成一些特殊的功能就不可避免的用到一些特殊的函数，我们也可以对着特征值做检查来定位webshell，同样的webshell本身也会进行加密来躲避这种检测。

## 二.webshell长什么样子

以下是asp webshell的样例，从界面看，它的功能还是比较全的，可以对服务器的文件目录进行读写操作，如果你是网站管理员的话肯定是不希望普通用户获得下面的权限的。

The screenshot shows a web browser window with the following details:

- Address Bar:** http://tapesales.net/admin/DatabaseBackup/web.asp
- Title Bar:** 提权目录列表
- Left Sidebar (Menu):**
  - 查看硬盘
  - 站点根目录
  - 本程序目录
  - 新建目录
  - 新建文本
  - 上传文件
  - 文件夹打包-解包
  - 服务器信息
  - 查看可写目录
  - 系统服务-用户账号
  - 主机信息-组件支持
  - 管理组帐号
  - 服务器探测
  - 挂马相关
  - 批量挂马
  - 批量清马
  - 批量替换
  - 部分挂马
  - 查找木马
  - 提权相关
  - 执行Cmd命令
  - 端口扫描器
  - 注册表操作
- Table (Server Component Information):**

服务器组件信息	
服务器名	tapesales.net
服务器IP	
服务器时间	2009-8-20 12:33:14
服务器CPU数量	
服务器操作系统	
WEB服务器版本	Microsoft-IIS/6.0
Scripting.FileSystemObject	✓ 文件操作组件
wscript.shell	✗ 命令行执行组件
ADOX.Catalog	✓ ACCESS建库组件
JRO.JetEngine	✓ ACCESS压缩组件
Scripting.Dictionary	✓ 数据流上传辅助组件
Adodb.connection	✓ 数据库连接组件
Adodb.Stream	✓ 数据流上传组件
SoftArtisans.FileUp	✗ SA-FileUp 文件上传组件
LyfUpload.UploadFile	✗ 刘云峰文件上传组件
Persists.Upload.1	✓ ASPUpload 文件上传组件
JMail.SmtpMail	✓ JMail 邮件收发组件
CDONTS.NewMail	✗ 虚拟SMTP发信组件
SmtpMail.SmtpMail.1	✗ SmtpMail发信组件
Microsoft.XMLHTTP	✓ 数据传输组件

## 三.WebShell是如何入侵系统的？

### 1) 利用站点上传漏洞实现上传webshell

利用系统前台的上传业务，上传WebShell脚本，上传的目录往往具有可执行的权限。在web中有上传图像、上传资料文件的地方，上传完后通常会向客户端返回上传的文件的完整URL信息，有时候不反馈，我们也可以猜到常见的image、upload等目录下面，如果Web对网站存取权限或者文件夹目录权限控制不严，就可能被利用进行webshell攻击，攻击者可以利用上传功能上传一个脚本文件，然后在通过url访问这个脚本，脚本就被执行。然后就会导致黑客可以上传webshell到网站的任意目录中，从而拿到网站的管理员控制权限。

2) 黑客获取管理员的后台密码，登陆到后台系统，利用后台的管理工具向配置文件写入WebShell木马，或者黑客私自添加上传类型，允许脚本程序类似asp、php的格式的文件上传。

3) 利用数据库备份与恢复功能获取webshell。如备份时候把备份文件的后缀改成asp。或者后台有mysql数据查询功能，黑客可以通过执行select..in To outfile 查询输出php文件，然后通过把代码插入到mysql，从而导致生成了webshell的木马。

4) 系统其他站点被攻击，或者服务器上还搭载了ftp服务器，ftp服务器被攻击了，然后被注入了webshell的木马，从而导致网站系统也被感染。

5) 黑客直接攻击Web服务器系统漏洞入侵Web服务器在系统层面也可能存在漏洞，如果黑客利用其漏洞攻击了服务器系统，那么黑客获取了其权限，则可以在web服务器目录里上传webshell文件。

## 四.WebShell能够肆虐的重要原因是什么？

### 1) 通过web站点漏洞上传webshell

WebShell能够被注入很大程度是由于服务器或中间件的安全漏洞。例如：老版本的IIS目录解析漏洞、文件名解析漏洞、应用后台暴露和弱口令、fast-CGI解析漏洞、apache文件解析漏洞、截断上传、后台数据库备份功能上传、利用数据库语句上传等漏洞实现。

### 2) 站点部署时混入了webshell文件

我们发现有大量的客户在使用从网上下载的第三方开源代码时，混入了WebShell的恶意脚本，造成二次入侵或多次入侵，所以在部署前期，如果不是新开发的代码，需要对代码进行恶意文件扫描查杀，防止上线后被入侵。

## 五.如何防止系统被植入WebShell?

- 配置必要的防火墙开启防火墙策略,防止暴露不必要的服务，为黑客提供利用条件。 -
- 对服务器进行[安全加固](#)，例如:关闭远程桌面这些功能、定期更换密码、禁止使用最高权限用户运行程序、使用[https](#)加密协议。
- 加强权限管理，对敏感目录进行权限设置，限制上传目录的脚本执行权限，不允许配置执行权限。
- 安装[webshell检测工具](#)，发现检测结果后，立即隔离查杀，并排查漏洞。
- 排查程序存在的漏洞，并及时修补漏洞，如果没有安全能力，可以通过[应急响应服务](#)人工介入协助排查漏洞及入侵原因，同时可以选用[阿里云商业web应用防火墙](#)防御，降低入侵机率。

# Web应用安全漏洞

- Web漏洞含义解释
- 挂马攻击和防御
- URL跳转漏洞
- CRLF HTTP头部注入漏洞
- 任意文件下载漏洞
- 域名未设置SPF解析记录
- 系统弱口令
- 后门文件漏洞
- 文件包含漏洞
- SQL注入漏洞
- 网络钓鱼攻击和防御
- 越权漏洞
- Crossdomain.xml配置不当
- 文件上传漏洞
- 应用越权漏洞
- SEO暗链
- 目录遍历攻击
- 网站备份文件泄露
- 代码执行漏洞
- 跨站攻击
- DNS区域传送漏洞
- 信息泄露漏洞

# Web漏洞含义解释

## 跨站攻击

### 漏洞描述

跨站脚本攻击（Cross-site scripting，通常简称为XSS）发生在客户端，可被用于进行窃取隐私、钓鱼欺骗、偷取密码、传播恶意代码等攻击行为。恶意的攻击者将对客户端有危害的代码放到服务器上作为一个网页内容，使得其他网站用户在观看此网页时，这些代码注入到了用户的浏览器中执行，使用户受到攻击。一般而言，利用跨站脚本攻击，攻击者可窃会话COOKIE从而窃取网站用户的隐私，包括密码。

XSS攻击使用到的技术主要为HTML和Javascript，也包括VBScript和ActionScript等。XSS攻击对WEB服务器虽无直接危害，但是它借助网站进行传播，使网站的使用用户受到攻击，导致网站用户帐号被窃取，从而对网站也产生了较严重的危害。

### 漏洞危害

1) 钓鱼欺骗：最典型的就是利用目标网站的反射型跨站脚本漏洞将目标网站重定向到钓鱼网站，或者注入钓鱼JavaScript以监控目标网站的表单输入，甚至发起基于DHTML更高级的钓鱼攻击方式。2) 网站挂马：跨站时利用IFrame嵌入隐藏的恶意网站或者将被攻击者定向到恶意网站上，或者弹出恶意网站窗口等方式都可以进行挂马攻击。3) 身份盗用：Cookie是用户对于特定网站的身份验证标志，XSS可以盗取到用户的Cookie，从而利用该Cookie盗取用户对该网站的操作权限。如果一个网站管理员用户Cookie被窃取，将会对网站引发巨大的危害。4) 盗取网站用户信息：当能够窃取到用户Cookie从而获取到用户身份使，攻击者可以获取到用户对网站的操作权限，从而查看用户隐私信息。5) 垃圾信息发送：比如在SNS社区中，利用XSS漏洞借用被攻击者的身份发送大量的垃圾信息给特定的目标群。6) 劫持用户Web行为：一些高级的XSS攻击甚至可以劫持用户的Web行为，监视用户的浏览历史，发送与接收的数据等等。7) XSS蠕虫：XSS 蠕虫可以用来打广告、刷流量、挂马、恶作剧、破坏网上数据、实施DDoS攻击等。

## CRLF攻击

### 漏洞描述

HTTP响应拆分漏洞，也叫CRLF注入攻击。CR、LF分别对应回车、换行字符。HTTP头由很多被CRLF组合分离的行构成，每行的结构都是“键：值”。如果用户输入的值部分注入了CRLF字符，它有可能改变的HTTP报头结构。

### 漏洞危害

攻击者可能注入自定义HTTP头。例如，攻击者可以注入会话cookie或HTML代码。这可能会进行类似的XSS（跨站点脚本）或会话固定漏洞。

## SQL注入攻击

### 漏洞描述

**SQL注入攻击（SQL Injection）**，简称注入攻击、**SQL注入**，被广泛用于非法获取网站控制权，是发生在应用程序的数据库层上的安全漏洞。在设计不良的程序当中，忽略了对输入字符串中夹带的**SQL指令**的检查，那么这些夹带进去的指令就会被数据库误认为是正常的**SQL指令**而运行，从而使数据库受到攻击，可能导致数据被窃取、更改、删除，以及进一步导致网站被嵌入恶意代码、被植入后门程序等危害。

## 漏洞危害

1) 机密数据被窃取 2) 核心业务数据被篡改 3) 网页被篡改 4) 数据库所在服务器被攻击变为傀儡主机，甚至企业网被入侵。

# 写入**webshell**攻击

## 漏洞描述

写入**webshell**攻击，是指WAF检测到攻击者正在往用户网站写入网页木马，企图控制服务器。

## 漏洞危害

攻击者可以在用户网站上写入一个**web**木马后门，用于操作用户网站上的文件，执行命令等等。

# 本地文件包含

## 漏洞描述

本地文件包含是指程序代码在处理包含文件的时候没有严格控制。利用这个漏洞，攻击者可以先把上传的静态文件，或网站日志文件作为代码执行，进而获取到服务器权限，造成网站被恶意删除，用户和交易数据被篡改等一系列恶性后果。

## 漏洞危害

攻击者可以利用该漏洞，在服务器上执行命令。

# 远程文件包含

## 漏洞描述

远程文件包含是指程序代码在处理包含文件的时候没有严格控制。导致用户可以构造参数包含远程代码在服务器上执行，进而获取到服务器权限，造成网站被恶意删除，用户和交易数据被篡改等一系列恶性后果。

## 漏洞危害

攻击者可以利用该漏洞，在服务器上执行命令。

# 远程代码执行

## 漏洞描述

代码注入是指由于服务端代码漏洞导致恶意用户输入在服务端被执行的一种高危安全漏洞。

## 漏洞危害

利用该漏洞，可以在服务器上执行攻击者拼装的代码。

# FastCGI攻击

## 漏洞描述

Nginx中存在一个较为严重安全问题，FastCGI模块默认情况下可能导致服务器错误的将任何类型的文件以PHP的方式进行解析。

## 漏洞危害

这将导致严重的安全问题，使得恶意的攻击者可能攻陷支持php的Nginx服务器。

# 挂马攻击和防御

## 什么是挂马攻击

挂马攻击指，攻击者在已经获得控制权的网站的网页中嵌入恶意代码（一般通过 **Iframe**、**Script** 引用）。

当用户访问该网页时，嵌入的恶意代码利用浏览器本身的漏洞、第三方 **ActiveX** 漏洞，或者其它插件（如 **Flash**、**PDF** 插件等）漏洞，在用户不知情的情况下下载并执行恶意木马。

## 挂马攻击有什么危害

网站被挂马后，表示该站点已经被黑客成功入侵。黑客可以获取用户账号密码、业务数据等其他敏感数据。

## 如何防御挂马攻击

- 使用云盾 先知 在业务代码上线前，进行代码安全测试、白盒代码审计等。
- 日常运维过程中，定期检测并修补网站本身以及网站所在服务端环境的各类漏洞，及时更新操作系统、应用服务软件补丁等。
- 使用云盾 Web 应用防火墙（WAF）进行安全防护。

# **URL** 跳转漏洞

## 漏洞描述

URL 跳转漏洞指 Web 程序直接跳转到参数中的 URL，或在页面中引入了任意的开发者 URL。

## 修复方案

在控制页面转向的地方校验传入的 URL 是否为可信域名。

# CRLF HTTP 头部注入漏洞

## 漏洞描述

CRLF 是“回车 + 换行”（\r\n）的简称。在 HTTP 协议中，HTTP Header 与 HTTP Body 是用两个 CRLF 符号进行分隔的，浏览器根据这两个 CRLF 符号来获取 HTTP 内容并显示。因此，一旦攻击者能够控制 HTTP 消息头中的字符，注入一些恶意的换行，就能注入一些会话 Cookie 或者 HTML 代码。

## 修复方案

1. 云盾 Web 应用防火墙服务可以有效拦截该漏洞的攻击代码。关于 Web 应用防火墙的更多介绍，请查看 [Web 应用防火墙产品详情页](#)。
2. 过滤 \r、\n 之类的换行符，避免输入的数据污染到其他 HTTP 消息头。

# 任意文件下载漏洞

## 漏洞描述

一些网站由于业务需求，可能提供文件查看或下载功能。如果对用户查看或下载的文件不做限制，则恶意用户能够查看或下载任意文件，可以是源代码文件、敏感文件等。

攻击者可构造恶意请求下载服务器上的敏感文件，进而植入网站后门控制网站服务器主机。

## 修复方案

- 升级您正在使用的 CMS 或插件至最新版本。
- 如果漏洞文件不再使用，请删除文件。
- 注意：删除前请做好备份。
- 如果问题还未能解决，建议您使用云盾 安全管家 服务。

# 域名未设置 **SPF** 解析记录

## 漏洞描述

SPF 记录是一种域名服务（DNS）记录，用于标识哪些邮件服务器可以代表您的域名发送电子邮件。SPF 记录的目的是为了防止垃圾邮件发送者在您的域名上，使用伪造的发件人地址发送邮件。

若您未对您的域名添加 SPF 解析记录，则黑客可以仿冒以该域名为后缀的邮箱，来发送垃圾邮件。

## 修复方案

在您的 DNS 服务提供商处，为您的域名添加一条 TXT 记录：

- 将主机字段（Host）设置为您子域名的名称。（例如，如果您的电子邮件地址是 contact@mail.example.com，则为 mail。）如果不使用子域名，则将其设为@。
- 用您的 SPF 记录填写 TXT 值字段。例如 v = spf1 a mx include: secureserver.net~all。

# 系统弱口令

## 漏洞描述

弱口令漏洞指系统口令的长度太短或者复杂度不够，如仅包含数字或字母等。

弱口令容易被破解，一旦被攻击者获取，可用来直接登录系统，读取甚至修改网站代码。

## 修复方案

修改口令，增加口令复杂度，如包含大小写字母、数字和特殊字符等。

# 后门文件漏洞

## 漏洞描述

该文件被插入某段后门的代码，黑客可直接访问此文件，并在网站上生成 Webshell。

## 修复方案

从官方获取您网站应用程序的最新版本。

# 文件包含漏洞

## 漏洞描述

文件包含漏洞是一种针对依赖于脚本运行时间的 **Web** 应用程序的漏洞。当应用程序使用攻击者控制的变量构建可执行代码的路径时，一旦其允许攻击者控制运行时执行哪个文件，则会引发该漏洞。文件包含漏洞会破坏应用程序加载代码的执行方式。

远程文件包含（**RFI**）在 **Web** 应用程序下载并执行远程文件时发生。这些远程文件通常以 **HTTP** 或 **FTP URI** 的形式，作为用户向 **Web** 应用程序提供的参数而获取。

本地文件包含（**LFI**）类似于远程文件包含，除了不包含远程文件外，只有本地文件（当前服务器上的文件）可以被包含用于执行。通过包含一个带有攻击者控制数据（如 **Web** 服务器的访问日志）的文件，仍然可以导致远程执行代码。

## 漏洞危害

该漏洞可被利用在服务器上远程执行命令。攻击者可以把上传的静态文件或网站日志文件作为代码执行，获取服务器权限，并进一步篡改用户和交易数据，恶意删除网站等。

## 修复方案

- 严格检查变量是否已经初始化。
- 建议您假定所有输入都是可疑的，尝试对所有提交的输入中可能包含的文件地址（包括服务器本地文件及远程文件）进行严格的检查，参数中不允许出现 `..` 之类的目录跳转符。
- 严格检查 `include` 类的文件包含函数中的参数是否外界可控。
- 不要仅仅在客户端做数据的验证与过滤，将关键的过滤步骤放在服务端执行。
- 在发布应用程序前，测试所有已知的威胁。

# SQL 注入攻击

## 漏洞描述

SQL 注入攻击指攻击者通过欺骗数据库服务器，来执行未授权的任意查询。

SQL 注入攻击借助 SQL 语法，针对应用程序开发者在编程过程中的缺陷或不严谨代码，当攻击者能够操作数据，向应用程序中插入一些 SQL 语句时，SQL 注入攻击就发生了。通常情况下，攻击者会在应用程序中预先定义好的查询语句结尾加上额外的 SQL 语句元素，来实现 SQL 注入攻击。

SQL 注入漏洞是目前互联网最常见也是影响非常广泛的漏洞。

## 漏洞危害

- 网页被篡改。
- 数据被篡改。
- 核心数据被窃取。
- 数据库所在服务器被攻击，变成傀儡主机。

## 修复方案

- 若您使用的是第三方 CMS 程序（如：Discuz!, DedeCMS, ECshop等），请将程序升级至最新版本。
- 过滤用户输入的数据。默认情况下，应当认为用户的所有输入都是不安全的。
  - 在网页代码中需要对用户输入的数据进行严格过滤。
  - 部署 Web 应用防火墙。
  - 对数据库操作进行监控。

# 网络钓鱼攻击和防御

## 什么是网络钓鱼攻击

网络钓鱼攻击（**Phishing**，与钓鱼的英语 **fishing** 发音相近，又名钓鱼法或钓鱼式攻击）是通过大量发送声称来自于银行或其他知名机构的欺骗性垃圾邮件，意图引诱收信人给出敏感信息（如用户名、口令、账号 ID、ATM PIN 码，或信用卡详细信息）的一种攻击方式。

## 常见攻击方式

钓鱼攻击会使用多种技术，使一封电子邮件信息或网页的显示同其运行表现出欺骗性差异。下面列出了一些较为常见攻击技术。

### 使用相似的域名，复制页面内容，假冒网页设计

由于大多数浏览器以无衬线字体显示URL，为了达到欺骗目的，攻击者会注册与要假冒的网站域名相似的域名（有时，攻击者还会改变大小写或使用特殊字符）。例如，“[paypal.com](http://paypal.com)”可用来假冒“[paypal.com](http://paypal.com)”，“[barclays.com](http://barclays.com)”可用来假冒“[barclays.com](http://barclays.com)”。

假域名也可能只是简单地将真域名的一部分插入其中。例如，用“[ebay-members-security.com](http://ebay-members-security.com)”假冒“[ebay.com](http://ebay.com)”，用“[users-paypal.com](http://users-paypal.com)”假冒“[paypal.com](http://paypal.com)”。而大多数用户缺少判断一个假域名是否为域名持有者所拥有的工具和知识，

### 隐藏URL

假冒URL会利用URL语法中一种不常见的特性来隐藏其URL。在URL语法中，用户名和密码可包含在域名前，语法结构为<http://username:password@domain/>。攻击者可能将一个看起来合理的域名放在用户名位置，而把真实的域名隐藏起来或放在地址栏的最后。例如，<http://earthlink.net%6C%6C...%6C@211.112.228.2>。

最新版本的网页浏览器已关闭这个漏洞，其方法是在地址栏显示前将URL中的用户名和密码去掉，或者完全禁用含用户名/密码的URL语法（Internet Explorer使用这种办法）。

### 使用IP地址显示

隐藏一台服务器身份的最简单办法就是使它以IP地址的形式显示，如<http://210.93.131.250>。由于许多合法URL也包含一些不透明且不易理解的数字，因此，只有懂得解析URL且足够警觉的用户才有可能对这种地址产生怀疑。

### 欺骗性的超链接

一个超链接的标题完全独立于它实际指向的URL。攻击者利用这种显示和运行间的差异，在链接标题中显示一个URL，而在背后使用了一个完全不同的URL。即便是一个有着丰富知识的用户，在看到消息中显而易见的URL后，也可能不会去检查其真实的URL。

检查超链接目的地址的方法是将鼠标放在超链接上，检查在状态栏中显示出来的URL（但所显示内容也可能被攻击者利用JavaScript或URL隐藏技术更改）。

### 隐藏提示

隐藏提示通过完全替换地址栏或状态栏达到使其提供欺骗性提示信息的目的。攻击者用JavaScript在Internet Explorer的地址栏上创建的一个简单的小窗口，用来显示一个完全无关的URL。

## 弹出窗口

在浏览器中显示的是真实的Citibank网页，但在页面上弹出了一个简单的窗口，要求用户输入个人信息。

## 社会工程

社会工程使用的是非技术手段使用户坠入陷阱。

- 一个常见的策略是急迫性，使用户急于采取行动，而较少花时间去核实消息的真实性。
- 另一个策略是威胁用户，如果不按照所要求的去做就会造成可怕的后果，如终止服务或关闭账户；少数攻击还许诺将获得巨额回报(如“你中了一个大奖!”)，但威胁攻击更为常见。

## 网络钓鱼攻击有什么危害

网络钓鱼（Phishing）攻击者利用欺骗性的电子邮件和伪造的Web站点来进行网络诈骗活动。诈骗者通常会将自己伪装成网络银行、在线零售商和信用卡公司等可信的品牌，骗取用户的私人信息。受骗者往往会泄露自己的私人资料，如信用卡号、银行卡账户、身份证号等内容。

## 如何防御网络钓鱼攻击

- 不要随意点击不明链接，打开陌生人的电子邮件。
- 对服务器或终端电脑进行安全加固，安装杀毒软件并及时升级病毒知识库和操作系统（如Windows操作系统、应用服务软件）补丁。

# 越权漏洞

## 漏洞描述

越权漏洞指在网站中某个页面上，能看到不属于当前用户身份的信息，如以用户 A 的身份能看到用户 B 的信息。

## 修复方案

- 如果您使用的是第三方 CMS，建议您将 CMS 升级到官方最新版本。
- 如果您使用自己编写的网站程序，建议您限制该页面可访问的对象，如添加权限认证、或指定 IP 才能访问。
- 如果该页面不需要使用，建议您把页面删除。

# **Crossdomain.xml 配置不当**

## **漏洞描述**

网站根目录下的 `crossdomain.xml` 文件指明了远程 Flash 是否可以加载当前网站的资源（图片、网页内容、Flash 等）。如果配置不当，可能导致遭受跨站请求伪造（CSRF）攻击。

## **修复方案**

对于不需要从外部加载资源的网站，在 `crossdomain.xml` 文件中更改 `allow-access-from` 的 `domain` 属性为域名白名单。

# 文件上传漏洞

## 漏洞描述

文件上传漏洞指网站中某个页面可以被利用来上传任意文件。

## 修复方案

- 如果您使用的是第三方 CMS，建议您及时更新系统，确保使用最新版本的系统。
- 如果您使用的是自己编写的上传功能，建议您限制可访问该页面的对象，如添加权限认证、或指定 IP 才能访问。
- 如果您不需要使用上传页面，建议您将其删除。

# 应用越权漏洞

## 漏洞描述

应用越权漏洞，指网站中某个页面因代码逻辑不严谨，导致黑客可以以普通用户身份执行管理员才能执行的操作。

## 修复方案

- 如果您使用的是第三方 CMS，建议您将其升级到官方最新版本。
- 如果您使用自己编写的网站，建议您限制可访问该页面的对象，如添加权限认证、或指定 IP 才能访问。
- 如果页面不需要使用，建议您将页面删除。

# SEO 暗链

## 漏洞描述

SEO 暗链是指网站上被植入隐藏的指向其他网站的链接，通过这些链接可以提高对方网站的搜索引擎排名。

如果服务器检测出暗链，且若非本人操作，则说明服务器已经被入侵。

## 修复方案

目前，大部分用户的网站首页被插入了包含 `jquery.min.php` 的 JS 代码，请您检查自己首页的 HTML 源码，并清除恶意链接。

此外，建议您使用云盾 [安骑士](#) 检测 `webshell` 并清理。

# 目录遍历攻击

## 漏洞描述

目录遍历是一种 HTTP 攻击，它允许攻击者访问受限制的目录，并在 Web 服务器根目录之外执行命令。该漏洞因应用程序未检查文件路径引发，可能导致服务器的任意文件或源代码泄漏。

## 修复方案

严格检查文件路径参数，将其限制在指定的范围；不要允许用户控制与文件路径相关的参数。

使用开源的漏洞修复插件，如 [云体验通用代码补丁](#)。

# 网站备份文件泄露

## 漏洞描述

网站备份文件泄露指管理员误将网站备份文件或是敏感信息文件存放在某个网站目录下。

外部黑客可通过暴力破解文件名等方法下载该备份文件，导致网站敏感信息泄露。

## 修复建议

- 不要在网站目录下存放网站备份文件或包含敏感信息的文件。
- 如需存放该类文件，请将文件名命名为难以破解的字符串。

# 代码执行漏洞

注意：若漏洞 URL 中包含/robots.txt/a.php,/favicon.ico/a.php，请参考 FastCGI 解析漏洞。

## 漏洞描述

代码执行漏洞是指应用程序对传入命令的参数过滤不严导致恶意攻击值能控制最终执行的命令，进而入侵系统，造成严重破坏的高危漏洞。

## 漏洞危害

利用这个漏洞，攻击者可以执行任意代码。

## 修复方案

### 方案 1

- 严格检查程序参数，特别是“&”，“&&”，“|”，“||”，“eval”，“execute”这类参数。
- 在代码中去除 `system` 等直接命令行执行的函数，或者禁止把通过外部传入的参数传入到该类可执行函数的参数中。
- 如果使用的是第三方建站程序，务必升级到最新版本。

### 方案 2

使用开源的漏洞修复插件，具体可参考 [云体验通用代码补丁](#)。

注意：执行此方案需要站长懂得编程并且能够修改服务器代码。

# 跨站攻击

## 漏洞描述

跨站脚本攻击（Cross-site scripting，简称XSS攻击），通常发生在客户端，可被用于进行隐私窃取、钓鱼欺骗、密码偷取、恶意代码传播等攻击行为。XSS攻击使用到的技术主要为HTML和Javascript脚本，也包括VBScript和ActionScript脚本等。恶意攻击者将对客户端有危害的代码放到服务器上作为一个网页内容，用户不经意打开此网页时，这些恶意代码会注入到用户的浏览器中并执行，从而使用户受到攻击。一般而言，利用跨站脚本攻击，攻击者可窃取会话cookie，从而获得用户的隐私信息，甚至包括密码等敏感信息。

## 漏洞危害

XSS攻击对Web服务器本身虽无直接危害，但是它借助网站进行传播，对网站用户进行攻击，窃取网站用户账号信息等，从而也会对网站产生较严重的危害。XSS攻击可导致以下危害：

- 钓鱼欺骗：最典型的就是利用目标网站的反射型跨站脚本漏洞将目标网站重定向到钓鱼网站，或者通过注入钓鱼JavaScript脚本以监控目标网站的表单输入，甚至攻击者基于DHTML技术发起更高级的钓鱼攻击。
- 网站挂马：跨站时，攻击者利用Iframe标签嵌入隐藏的恶意网站，将被攻击者定向到恶意网站上、或弹出恶意网站窗口等方式，进行挂马攻击。
- 身份盗用：Cookie是用户对于特定网站的身份验证标志，XSS攻击可以盗取用户的cookie，从而利用该cookie盗取用户对该网站的操作权限。如果一个网站管理员用户的cookie被窃取，将会对网站引发巨大的危害。
- 盗取网站用户信息：当窃取到用户cookie从而获取到用户身份时，攻击者可以盗取到用户对网站的操作权限，从而查看用户隐私信息。
- 垃圾信息发送：在社交网站社区中，利用XSS漏洞借用被攻击者的身份发送大量的垃圾信息给特定的目标群。
- 劫持用户Web行为：一些高级的XSS攻击甚至可以劫持用户的Web行为，从而监视用户的浏览历史、发送与接收的数据等等。
- XSS蠕虫：借助XSS蠕虫病毒还可以用来打广告、刷流量、挂马、恶作剧、破坏网上数据、实施DDoS攻击等。

## 修复方案

### 方案一

目前，云盾的“DDoS高防IP服务”以及“Web应用防火墙”均提供对Web应用攻击的安全防护能力。选择以上服务开通Web应用攻击防护，可以保障您的服务器安全。

### 方案二

将用户所提供的内容输入输出进行过滤。可以运用下面这些函数对出现XSS漏洞的参数进行过滤：

- PHP的htmlentities()或是htmlspecialchars()
- Python的cgi.escape()
- ASP的Server.HTMLEncode()
- ASP.NET的Server.HtmlEncode()或功能更强的Microsoft Anti-Cross Site Scripting Library
- Java的xssprotect(Open Source Library)
- Node.js的node-validator

### 方案三

使用开源的漏洞修复插件。（需要系统管理员懂得编程并且能够修改服务器代码。）

# DNS 区域传送漏洞

DNS 区域传送（DNS zone transfer）是指一台备用 DNS 服务器使用来自主 DNS 服务器的数据刷新自己的域（zone）数据库，从而避免主 DNS 服务器因意外故障影响到整个域名解析服务。

## 漏洞描述

一般情况下，DNS 区域传送只在网络里存在备用 DNS 服务器时才会使用；但许多 DNS 服务器却被错误地配置，只要有客户机发出请求，就会向对方提供一个 zone 数据库的详细信息。因此，不受信任的因特网用户也可以执行 DNS 区域传送（zone transfer）操作。

恶意用户可以通过 DNS 区域传送快速地判定出某个特定 zone 的所有主机，并收集域信息、选择攻击目标，进而找出未使用的 IP 地址，绕过基于网络的访问控制窃取信息。

## 漏洞修复

注意：建议您在修复前创建服务器快照，以免修复失败造成损失。

区域传送是 DNS 常用的功能，为保证使用安全，应严格限制允许区域传送的主机，例如一个主 DNS 服务器应该只允许它的备用 DNS 服务器执行区域传送功能。

在相应的 zone、options 中添加 allow-transfer，对执行此操作的服务器进行限制。如：

- 严格限制允许进行区域传送的客户端的 IP：

```
allow-transfer {1.1.1.1; 2.2.2.2;}
```

- 设置 TSIG key：

```
allow-transfer {key "dns1-slave1"; key "dns1-slave2";}
```

# 信息泄露漏洞

## 漏洞描述

信息泄露指在网站页面或应用中泄露了敏感信息。通过这些信息，攻击者可进一步入侵服务器。

## 修复建议

- 建议您删除探针等无用的程序，或者为其创建难以破解的名字。
- 禁用泄露敏感信息的页面或应用。
- 如果是第三方管理后台暴露在公网，建议您对目录的访问做限制。