十二、使用和滥用服务

Linux黑客基础–72–服务管理概述

服务是在后台运行的应用程序，等待您使用它。,linux中称为Daemon服务。

Apache 设置web服务器

Openssh服务–––远程访问

MySQL访问数据–––数据库服务

PostgreSQL（数据库服务）存储黑客信息，如在msf中会使用

服务控制工具

systemctl 命令 服务名称

动作命令：

启动 start

停止 stop

重启 restart

状态 status

把服务设为开机启动 enable

把服务设为开机不启动 disable

查询服务是否开机启动 is-enabled

Linux黑客基础-73-apache2服务架设网站-（1）配置文件解析

使用 APACHE Web 服务器创建 HTTP Web 服务器

apt-get install apache2　　安装

systemctl start apache2　启动

lsof -i :80　查看是否启动

1、配置文件解析

/usr/share/doc/apache2/README.Debian.gz　帮助文档

/etc/apache2 配置文件目录

/etc/apache2

├── apache2.conf //主配置文件（ main configuration file）

├── conf-available

├── conf-enabled //全局指令配置

　　*.conf

├── envvars

├── magic

├── mods-available

├── mods-enabled //模块的管理

*.load

*.conf

├── ports.conf //定义监听的地址和端口

├── sites-available

└── sites-enabled //定义了虚拟主机的配置

/etc/apache2/apache2.conf

命令语句解析

Include ports.conf //包含ports.conf中的配置

ports.conf

Listen 192.168.195.76:8088 //定义监听的地址和端口

- apache2.conf 是主要配置 文件。它通过包含所有剩余的配置将各个部分放在一起 文件，以启动 Web 服务器。
- ports.conf 始终包含在 主配置文件。它用于确定侦听端口 传入连接，并且可以随时自定义此文件。
- 启用 mods/、conf-enabled/ 和启用站点/目录中的配置文件包含 管理模块、全局配置的特定配置片段 片段或虚拟主机配置。
- 它们通过可用的符号链接激活 来自各自配置文件的配置文件 *-可用/对应。这些应该得到管理 通过使用我们的助手 A2enmod， a2dismod， a2ensite，a2dissite和 a2enconf， A2不信任 .有关详细信息，请参阅它们各自的手册页。
- 二进制文件称为 apache2。由于使用了 环境变量，在默认配置中，Apache2需要 使用 /etc/init.d/apache2 或 apache2ctl 启动/停止。**直接调用**

/usr/bin/apache2 将不适用于 默认配置。

2. apache2的启动脚本和服务程序

（1）服务脚本控制程序

/etc/init.d/apache2

apache2ctl

（2）服务程序

/usr/bin/apache2

3. 网站的主目录

默认是/var/www/html

4. 默认页面

index.html

Linux黑客基础-74-apache2服务架设网站-（2）配置虚拟主机

https://news.netcraft.com/archives/2023/03/23/march-2023-web-server-survey.html web服务器调差

流行的web服务器

nginx

Nginx (engine x) 是一个高性能的HTTP和反向代理web服务器 [13] ，同时也提供了IMAP/POP3/SMTP服务。Nginx是由伊戈尔·赛索耶夫为俄罗斯访问量第二的Rambler.ru站点（俄文：Рамблер）开发的，公开版本1.19.6发布于2020年12月15日。 [11]

其将源代码以类BSD许可证的形式发布，因它的稳定性、丰富的功能集、简单的配置文件和低系统资源的消耗而闻名。2022年01月25日，nginx 1.21.6发布。 [12]

Nginx是一款轻量级的Web 服务器/反向代理服务器及电子邮件（IMAP/POP3）代理服务器，在BSD-like 协议下发行。其特点是占有内存少，并发能力强，事实上nginx的并发能力在同类型的网页服务器中表现较好。

IIS

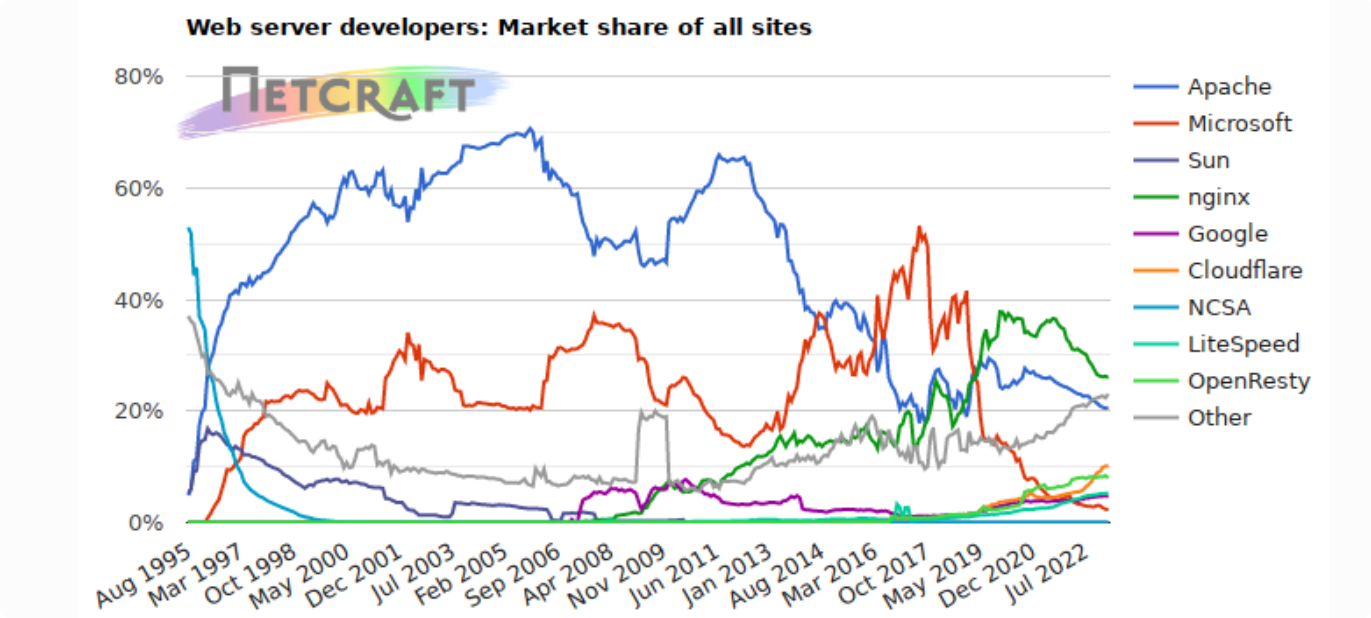
IIS是缩写，全称Internet Information Services（IIS,互联网信息服务),是由微软公司提供的基于运行Microsoft Windows的互联网基本服务。

IIS是指World Wide Web server服务，IIS是一种Web（网页）服务组件，专业的说，IIS可以赋予一部主机电脑一组以上的IP地址，而且还可以有一个以上的域名作为Web网站。．做过服务器配置的都应该知道IIS. . 制作好了网站怎么才能让别人浏览，就是通过网站服务器来实现的。IIS只是网站服务器的一种而已。

apache

Apache(音译为阿帕奇)是世界使用排名第一的Web服务器软件。它可以运行在几乎所有广泛使用的计算机平台上，由于其跨平台和安全性被广泛使用，是最流行的Web服务器端软件之一。

Apache是用C语言开发的基于模块化设计的web应用，核心代码不多。多数功能分散在各个模块中



流行的动态网站组合

LAMP

Linux+Apache+MySQL+PHP

LAMP 是指Linux（操作系统）+ Apache （HTTP 服务器）+ MySQL（数据库）和 PHP（网络编程语言），一般用来建立 web 应用平台。和 Java/J2EE 架构相

比， LAMP 具有 Web 资源丰富、轻量、快速开发等特点；与微软的 .NET 架构相比，LAMP具有通用、跨平台、高性能、低价格的优势。因此 LAMP 无论是性能、质量还是价格都是企业搭建网站的首选平台。

LNMP

Linux+Nginx+MySQL+PHP

https://www.ecshop.com/products/pc 典型的LNMP案例

LNMP：Linux系统下Nginx+MySQL+PHP这种网站服务器架构。Nginx是一个高性能的HTTP和反向代理服务器，也是一个IMAP/POP3/SMTP代理服务器。Mysql是一个小型关系型数据库管理系统。PHP是一种在服务器端执行的嵌入HTML文档的脚本语言。

WAMP

windows+Apache+MySQL+PHP

Windows下的Apache+Mysql/MariaDB+Perl/PHP/Python，一组常用来搭建动态网站或者服务器的开源软件，本身都是各自独立的程序，但是因为常被放在一起使用，拥有了越来越高的兼容度，共同组成了一个强大的Web应用程序平台。

虚拟主机的配置

/etc/apache2/sites-enabled/000-default.conf 配置文件位置

<VirtuaiHost *:80>

SeverName www.hy.com 网站的主机名

DocumentRoot /var/www/html/ 网站的根目录

DirectoryIndex index.html 网站的默认首页

ErrorLog ${APACHE_LOG_DIR}/error.log //错误日志

CustomLog ${APACHE_LOG_DIR}/access.log combined //访问日志，combined定义了日志的格式

</VirtualHost>

对目录做授权控制

<Directory /var/www/>

Options Indexes FollowSymLinks

//OPtions 选项

//Indexs 允许索引

//FollowSymLinks 允许使用软链接

AllowOverride None //不允许覆盖

Require all granted //授权所有的权限（如读）

</Directory>

定义一个自己的站点

网站的主机名 www.heyuan.com

网站的根（主）目录 /opt/www/heyuan

监听地址：端口 192.168.18.130：80

mkdir -pv /opt/www/heyuan

创建一个测试页面

第一步，对目录做授权

vi /etc/apache2/apache2.conf

<Directory /opt/www>

Options Indexes FollowSymLinks

AllowOverride None

Require all granted

</Directory>

第二步 定义虚拟主机

vi /etc/apache2/sites-enabled/001-heyuan.conf

<virtualhost 192.168.18.130:80>

ServerName www.heyuan.com

DocumentRoot /opt/www/heyuan

DirectoryIndex index.html

ErrorLog ${APACHE_LOG_DIR}/heyuan_error.log

CustomLog ${APACHE_LOG_DIR}/heyuan_access.log combined

</virtualhost>

第三步 检查语法并重启服务

apache2ctl -t 检查配置文件的语法

systemctl restart apache2

验证:

建议在/etc/hosts文件中添加

192.168.18.130 www.heyuan.com

定义第二个站点

网站的主机名 www.lisijia.com

网站的根（主）目录 /opt/www/lisijia

监听地址：端口 192.168.18.130：80

mkdir –pv /opt/www/lisijia

创建一个测试页面

定义虚拟主机

vi /etc/apache2/sites–enabled/001–heyuan.conf

```
<virtualhost 192.168.18.130:80>
        ServerName www.lisijia.com
        DocumentRoot /opt/www/lisijia
        DirectoryIndex index.html
        ErrorLog ${APACHE_LOG_DIR}/lisijia_error.log
        CustomLog ${APACHE_LOG_DIR}/lisijia_access.log combined
</virtualhost>
```

apache2ctl –t 检查配置文件的语法

systemctl restart apache2

建议在/etc/hosts文件中添加

192.168.18.130 www.lisijia.com

日志文件存放路径

grep –R 'APACHE_LOG_DIR' /etc/apache2

日志文件目录 /var/log/apache2

Linux黑客基础-75-SSH服务-介绍及针对SSH服务的口令破解

SSH 是 Secure Shell 的简写。

能够提供一个安全的远程连接

tcp/22

systemctl start ssh　开启

systemctl enable ssh　设置开机自启动

使root用户可以远程登录kali

SSH服务配置文件：/etc/ssh/sshd_config

远程连接SSH服务器

nmap -p 22 192.168.18.137　　指定扫描的端口

ssh 用户名@远程主机

针对ssh的暴力破解

1、准备一个口令字典

cat >>password_list << EOF

2、准备一个口令破解工具

hydra

https://github.com/facebookresearch/hydra 官网

Hydra是一款非常强大的暴力破解工具，它是由著名的黑客组织THC开发的一款开源暴力破解工具。Hydra是一个验证性质的工具，主要目的是：**展示安全研究人员从远程获取一个系统认证权限。**

Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak – Please do not use in military or secret service organizations,

案例

hydra –l root –P password_list ssh://192.168.18.130

–l：指定登录的用户名

–P：口令字典文件

```
[22][ssh] host: 192.168.18.130   login: root   password: 1
1 of 1 target successfully completed, 1 valid password found
```

centos 安全日志文件：/var/log/secure

Linux黑客基础-76-SSH服务-更改SSH服务端口号

如何防范针对ssh的口令暴力破解

1、更改默认的服务端口

vi /etc/ssh/sshd_config

修改端口号

netstat -tnlp | grep "sshd"　　验证

客户端连接

ssh -p 22226 root@192.168.18.130

服务端查看连接

lsof -i :22226

黑客

nmap -p 1-65535 192.168.195.76

nmap -p- 192.168.195.76

PS:端口号的范围　1-65535

nmap -sV -p- 192.168.195.76

-sV　探测服务的版本

┌──(root💀kali2)-[~]

└──# nmap -sV -p- 192.168.195.76

Starting Nmap 7.91 (https://nmap.org) at 2021-11-01 11:28 CST

Nmap scan report for www.xiaofeixia.com (192.168.195.76)

Host is up (0.0000020s latency).

Not shown: 65533 closed ports

PORT STATE SERVICE VERSION

80/tcp open http Apache httpd 2.4.48 ((Debian))

22226/tcp open ssh OpenSSH 8.4p1 Debian 6 (protocol 2.0)

Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

hydra -l root -P password.lst -s 22226 ssh://192.168.195.76

-s 指定服务端口

Linux黑客基础-77-SSH服务-使用tcp_wrappers保护SSH服务

如何防范针对ssh的口令暴力破解

1、更改默认的端口

2、设置复杂的密码且定期更改口令

3、使用密钥认证

4、定期查看安全日志文件

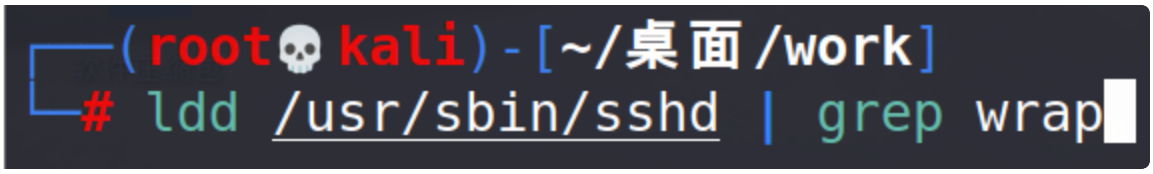
5、利用TCP Wrappers构建访问控制列表

TCP Wrappers也称为tcp_wrappers。他是一个基于主机的网络访问控制列表系统，类似于ACL

TCP Wrappers的核心是名为libwrap的库

所有调用这个库的长须都可以利用libwrap提供的网络访问控制能力

在linux系统中，我们可以使用ldd命令来判断一个程序是否调用了libwrapp的库



Debian 软件包为tcpd

Centos 软件包 tcp_wrappers

远程IP请求连接的时候， TCP Wrappers检查策略是先看/etc/hosts.allow中是否允许， 如果允许就直接放行； 如果没有， 则再看/etc/hosts.deny中是否禁止，如果禁止， 那么就禁止连接； 否则允许连接。

推荐采用白名单机制 ★★★★★

/etc/hosts.allow

sshd:192.168.195.21,192.168.22.33

/etc/hosts.deny

sshd:ALL

以上的配置表示只允许192.168.195.21和192.168.22.33这两台主机可以使用SSH连接

（6）使用DenyHosts、Fail2ban类似的安全工具

http://denyhosts.sourceforge.net/

DenyHosts 是一个脚本，旨在由 Linux 系统管理员运行，以帮助 阻止 SSH 服务器攻击（也称为基于字典的攻击和暴力破解） 攻击）。

http://www.fail2ban.org/wiki/index.php/Main_Page

Fail2ban扫描日志文件（例如/var/log/apache/error_log）并禁止显示恶意迹象的IP——密码失败过多、寻求漏洞等。通常，Fail2Ban 用于更新防火墙规则以在指定的时间内拒绝 IP 地址，尽管也可以配置任何任意**其他操作**（例如发送电子邮件）。开箱即用的Fail2Ban带有各种服务（apache，courier，ssh等）的**过滤器**。

Linux黑客基础-78-SSH服务-端口敲门Knocked服务概述

（7）给SSH服务带个隐身的斗篷

通过Knockd（端口敲门）服务把SSH服务隐藏起来

端口敲门服务（knockd）服务通过动态的添加iptables规则来隐藏系统开启的服务，使用自定义的一系列序列号来“敲门”。

通过这种方法使系统开启需要访问的服务端口，才能对外访问。

不使用时，再使用自定义的序列号来“关门”，将端口关闭，不对外监听。

进一步提升了服务和系统的安全性。

knockd is a port-knock server. It listens to all traffic on an ethernet (or PPP) interface, looking for special "knock" sequences of port-hits. A client makes

these port-hits by sending a TCP (or UDP) packet to a port on the server. This port need not be open -- since knockd listens at the link-layer

level, it sees

all traffic even if it's destined for a closed port. When the server detects a specific sequence of port-hits, it runs a command defined in its configuration

file. This can be used to open up holes in a firewall for quick access.

Linux黑客基础-79-端口敲门-Knocked服务安装.文件介绍

Step1 安装Knockd服务

apt-get update

apt-get install knockd

Step2 Knockd服务的配置文件

updatedb

locate knockd

/etc/knockd.conf ★★★★★

/etc/default/knockd ★★★★★

/usr/sbin/knockd 服务程序

/var/log/messages Linux默认的系统日志文件★★★★★

Linux黑客基础-80-端口敲门-Knocked服务配置文件解析

1. Knockd服务的配置文件 /etc/knockd.conf

[options]

UseSyslog //使用系统日志

[openSSH]

sequence = 7000,8000,9000 //自定义的序列（端口号）

seq_timeout = 5 //端口连接的超时时间

command = /sbin/iptables –A INPUT –s %IP% –p tcp ––dport 22 –j ACCEPT

tcpflags = syn //tcp的flag位为syn，只接受一个新连接

[closeSSH]

sequence = 9000,8000,7000

seq_timeout = 5

command = /sbin/iptables –D INPUT –s %IP% –p tcp ––dport 22 –j ACCEPT

tcpflags = syn

[openHTTPS]

sequence = 12345,54321,24680,13579

seq_timeout = 5

command = /usr/local/sbin/knock_add –i –c INPUT –p tcp –d 443 –f %IP%

tcpflags = syn

2. 更改日志文件

[options]

LogFile = /var/log/portknocking.log //定义日志文件的位置

3. 重要语句解析

Firewalld

端口敲门成功之后要执行的命令

command = /sbin/iptables –A INPUT –s %IP% –p tcp ––dport 22 –j ACCEPT

iptables 是Linux内置防火墙的管理工具

–A 添（追）加一条规则，默认是最后一条规则 append

–I 插入一条规则

–D 删除一条规则，默认是第一条规则

–s 指定数据包的源

–d 指定数据包的目标

–p 指定协议

––dport 目标端口

––sport 源端口

–j 动作

常用的动作有

ACCEPT 接受

DROP　丢弃

REJECT 丢弃，但是会弹回消息

LOG　记录到日志

-L　查看规则

-n　以数字形式显示

iptables -L -n　//查看防火墙的规则

-P　去更改默认规则

iptables -P INPUT DROP //把INPUT链的默认规则更改为DROP

案例：

Kali主机可以ping通别人，但别人ping不通Kali

ping 基于ICMP协议

echo-request (ping) 8　请求

echo-reply (pong) 0　回复

iptables –p icmp ––help //查看icmp协议的具体帮助

––icmp–type 定义icmp的类型

––line–numbers 查看规则的编号

iptables –A INPUT –p icmp ––icmp–type echo–reply –j ACCEPT Kali主机可以ping通别人，但别人ping不通Kali

```
┌──(root💀kali)-[~]
└─# iptables -A INPUT -p icmp --icmp-type echo-reply -j ACCEPT
```

```
┌──(root💀kali)-[~]
└─# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0            icmptype 0
```

给规则增加编号

```
┌──(root💀kali)-[~]
└─# iptables -L -n --line-numbers
Chain INPUT (policy ACCEPT)
num  target     prot opt source               destination
1    ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0            icmptype 0
```

Linux黑客基础–81–端口敲门–Knocked服务配置实战

Knockd服务的具体配置

Step0: 更改防火墙的INPUT的默认规则

iptables –P INPUT DROP

iptables-save

保存防火墙的规则

Step1: 更改配置文件

可以保持默认配置

Step2: vi /etc/default/knockd

START_KNOCKD=1

Step3 启动knockd服务

sytemctl start knockd

sytemctl enable knockd

Step4 客户端验证

启动

（1）使用knock客户端工具 apt-get install knockd

nmap -p22 192.168.195.76

knock -v 192.168.195.76 7000 8000 9000

nmap -p22 192.168.195.76

在服务端 tail -f /var/log/knockdport.log

关闭

knock 192.168.195.76 9000 8000 7000

（2）端口敲门脚本

for x in 7000 8000 9000;do nmap –Pn ––max–retries 0 –p $x 192.168.195.76;done

nmap –Pn ––max–retries 0 –p 7000 192.168.195.76

–Pn 禁止主机发现，假设主机是存活的

—max–retries ，表示端口扫描探测包最多被重传几次

–p 指定扫描的端口

for x in 9000 8000 7000;do nmap –Pn ––max–retries 0 –p $x 192.168.195.76;done

Linux黑客基础–82–Mysql–01–概述

MySQL 是数据库驱动 Web 应用程序背后使用最广泛的数据库。在我们现代的 Web 2.0 技术时代，几乎
每个网站都是数据库驱动的，这意味着 MySQL 拥有大部分网络的数据。

Mysql就是互联网上广为使用的数据库系统

MySQL 最初由瑞典的 MySQL AB 于 1995 年开发，然后在 2008 年被 SunMicrosystems 收购，而后者又于 2009 年被 Oracle 收购 – 因此 MySQL 现在归Oracle 所有。甲骨文是世界上最大的数据库软件发行商，因此开源社区对甲骨文保持 MySQL 开源的承诺存在重大挑战。因此，现在有一个名为“ Maria”的MySQL数据库软件的分支，致力于保持该软件及其后续版本的开源。作为 Linux 管理员或黑客，你应该关注 MariaDB。

MariaDB是Mysql的开源后续分支

数据库是黑客的金羊毛。也是黑客的首选攻击目标
很多web应用（如Discuz！--论坛、eshop--购物商城）都会把Mysql作为数据库的首选

流行的内容管理系统（ CMS），如 Joomla， Drupal 和 Ruby on Rails、帝国、织梦 也都使用 MySQL。

Linux黑客基础-83-Mysql-02-启动及基本操作

1、启动

工作端口：3306

systemctl start mysql　启动

lsof -i :3306　查看是否启动

systemctl enable mysql　开机自启动

netstat -tnlp | grep mariadbd

```
┌──(root💀kali)-[~]
└─# netstat -tnlp | grep mariadbd
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN      11919/mariadbd
```

2、连接数据库

mysql –u root –p

```
┌──(root💀kali)-[~]
└─# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 44
Server version: 10.5.12-MariaDB-1 Debian 11

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

3、使用数据库

MariaDB [(none)]> select version();　查看当前数据库的版本

```
MariaDB [(none)]> select version();
+-------------------+
| version()         |
+-------------------+
| 10.5.12-MariaDB-1 |
+-------------------+
1 row in set (0.001 sec)
```

MariaDB [(none)]> select user();　查看当前用户

```
MariaDB [(none)]> select user();
+----------------+
| user()         |
+----------------+
| root@localhost |
+----------------+
1 row in set (0.001 sec)
```

MariaDB [(none)]> select database();　查询当前操作数据库

```
MariaDB [(none)]> select database();
+------------+
| database() |
+------------+
| NULL       |
+------------+
1 row in set (0.001 sec)
```

MariaDB [(none)]> show databases;　查看当前数据库

```
MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
+--------------------+
3 rows in set (0.001 sec)
```

MariaDB [(none)]> use mysql;　使用数据库

```
MariaDB [(none)]> use mysql;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [mysql]>
```

MariaDB [mysql]> show tables;　查看数据库的表

```
MariaDB [mysql]> show tables;
+---------------------------+
| Tables_in_mysql           |
+---------------------------+
| column_stats              |
| columns_priv              |
| db                        |
| event                     |
```

MariaDB [mysql]> DESCRIBE user;　描述表的结构

```
MariaDB [mysql]> DESCRIBE user;
+------------------------+---------------------+------+-----+----------+-------+
| Field                  | Type                | Null | Key | Default  | Extra |
+------------------------+---------------------+------+-----+----------+-------+
| Host                   | char(60)            | NO   |     |          |       |
| User                   | char(80)            | NO   |     |          |       |
| Password               | longtext            | YES  |     | NULL     |       |
| Select_priv            | varchar(1)          | YES  |     | NULL     |       |
| Insert_priv            | varchar(1)          | YES  |     | NULL     |       |
```

MariaDB [mysql]> select user,host,password from mysql.user;　查询mysql系统中的用户信息

```
MariaDB [mysql]> select user,host,password from mysql.user
    -> ;
+-------------+-----------+----------+
| User        | Host      | Password |
+-------------+-----------+----------+
| mariadb.sys | localhost |          |
| root        | localhost | invalid  |
| mysql       | localhost | invalid  |
+-------------+-----------+----------+
3 rows in set (0.001 sec)
```

MariaDB [mysql]> select user,password,host from mysql.user where user='root';　where指定条件查询

```
MariaDB [mysql]> select user,password,host from mysql.user where user='root';
+------+----------+-----------+
| User | Password | Host      |
+------+----------+-----------+
| root | invalid  | localhost |
+------+----------+-----------+
1 row in set (0.003 sec)
```

Linux黑客基础-84-Mariadb-03-更改root密码

MariaDB [mysql]> select user,host,password from mysql.user;　查看user这个表

```
MariaDB [mysql]> select user,host,password from mysql.user;
+-------------+-----------+----------+
| User        | Host      | Password |
+-------------+-----------+----------+
| mariadb.sys | localhost |          |
| root        | localhost | invalid  |
| mysql       | localhost | invalid  |
+-------------+-----------+----------+
3 rows in set (0.003 sec)
```

MariaDB [mysql]> alter user 'root'@'localhost' identified by 'heyuan';　修改roo他密码为heyuan

```
MariaDB [mysql]> alter user 'root'@'localhost' identified by 'heyuan';
Query OK, 0 rows affected (0.002 sec)
```

MariaDB [mysql]> flush privileges;　刷新权限

```
MariaDB [mysql]> flush privileges;
Query OK, 0 rows affected (0.002 sec)
```

MariaDB [mysql]> select user,host,password from mysql.user;　查看更改后的密码

```
MariaDB [mysql]> select user,host,password from mysql.user;
+-------------+-----------+-------------------------------------------+
| User        | Host      | Password                                  |
+-------------+-----------+-------------------------------------------+
| mariadb.sys | localhost |                                           |
| root        | localhost | *F15E41DDD916B11071884EB078ABF3E713C0D4BF |
| mysql       | localhost | invalid                                   |
+-------------+-----------+-------------------------------------------+
3 rows in set (0.002 sec)
```

MariaDB [mysql]> \q　退出

Linux黑客基础-85-Mariadb-04-配置文件解析

相关软件包

mariadb-server　服务端

mariadb-client　客户端

mariadb-common　服务端和客户端都需要的公共软件包

dpkg -l | grep mariadb　查看和数据库有关的数据包

dpkg -L mariadb-server-10.5　查看软件包详细信息

/etc/mysql/　数据库的配置目录

/etc/mysql/my.cnf　通常数据库的

/etc/mysql/conf.d/*.cnf　也包含该目录中以.conf结尾的配置文件

The MariaDB/MySQL tools read configuration files in the following order:

//MariaDB/MySQL会按照下列顺序读取配置文件

0. "/etc/mysql/my.cnf"　　软链接，真正指向/etc/mysql/mariadb.cnf

1. "/etc/mysql/mariadb.cnf" (this file) to set global defaults,　设置全局默认配置

2. "/etc/mysql/conf.d/*.cnf" to set global options.　设置全局选项

3. "/etc/mysql/mariadb.conf.d/*.cnf" 设置和MARiaDB有关的选项

4. "~/.my.cnf" 用于设置和用户相关的配置文件选项（自定义配置）

tree mariadb.conf.d　　查看目录中的配置文件

mariadb.conf.d

├── 50-client.cnf　//客户端配置文件

├── 50-mysql-clients.cnf

├── 50-mysqld_safe.cnf　//服务进程安全有关的配置文件

├── 50-server.cnf　//服务配置文件

└── 60-galera.cnf

【案例】

更改监听地址

[mysqld]

vi /etc/mysql/mariadb.conf.d/50-server.cnf　修改这个配置文件

bind-address　　=0.0.0.0

systemctl restart mysql

Linux黑客基础-86-Mariadb-05-information_schema介绍

information_schema是mysql自带的一个库，这个库里存放了大量信息，保存着关于MySQL服务器所维护的所有其他数据库的信息。如数据库名，数据库的表，表栏的数据类型与访问权限等。也称为数据库字典

一、information_schema简介
在MySQL中，把 information_schema 看作是一个数据库，确切说是信息数据库。其中保存着关于MySQL服务器所维护的所有其他数据库的信息。如数据库名，数据库的表，表栏的数据类型与访问权 限等。在INFORMATION_SCHEMA中，有数个只读表。它们实际上是视图，而不是基本表，因此，你将无法看到与之相关的任何文件。

information_schema数据库表说明:

SCHEMATA表：提供了当前mysql实例中所有数据库的信息。是show databases的结果取之此表。

TABLES表：提供了关于数据库中的表的信息（包括视图）。详细表述了某个表属于哪个schema，表类型，表引擎，创建时间等信息。是show tables from schemaname的结果取之此表。

COLUMNS表：提供了表中的列信息。详细表述了某张表的所有列以及每个列的信息。是show columns from schemaname.tablename的结果取之此表。

STATISTICS表：提供了关于表索引的信息。是show index from schemaname.tablename的结果取之此表。

USER_PRIVILEGES（用户权限）表：给出了关于全程权限的信息。该信息源自mysql.user授权表。是非标准表。

SCHEMA_PRIVILEGES（方案权限）表：给出了关于方案（数据库）权限的信息。该信息来自mysql.db授权表。是非标准表。

TABLE_PRIVILEGES（表权限）表：给出了关于表权限的信息。该信息源自mysql.tables_priv授权表。是非标准表。

COLUMN_PRIVILEGES（列权限）表：给出了关于列权限的信息。该信息源自mysql.columns_priv授权表。是非标准表。

CHARACTER_SETS（字符集）表：提供了mysql实例可用字符集的信息。是SHOW CHARACTER SET结果集取之此表。

COLLATIONS表：提供了关于各字符集的对照信息。

COLLATION_CHARACTER_SET_APPLICABILITY表：指明了可用于校对的字符集。这些列等效于SHOW COLLATION的前两个显示字段。

TABLE_CONSTRAINTS表：描述了存在约束的表。以及表的约束类型。

KEY_COLUMN_USAGE表：描述了具有约束的键列。

ROUTINES表：提供了关于存储子程序（存储程序和函数）的信息。此时，ROUTINES表不包含自定义函数（UDF）。名为“mysql.proc name”的列指明了对应于INFORMATION_SCHEMA.ROUTINES表的mysql.proc表列。

VIEWS表：给出了关于数据库中的视图的信息。需要有show views权限，否则无法查看视图信息。

TRIGGERS表：提供了关于触发程序的信息。必须有super权限才能查看该表

Linux黑客基础-86-Mariadb-06-远程连接数据库及数据库授权

mysql -h 192.168.195.76 -u root -p

-h 指定连接的数据库主机，如果没有指定，默认为localhost（本机）

MySQL 赋予用户权限命令的简单格式可概括为：

grant 权限 on 数据库对象 to 用户

PS:用户格式 '用户名'@'主机'

'root'@'localhost'

对数据库操作常用的权限有：

查询（select）、插入（insert）、更新（update、alter）、删除（delete）

all 代表 所有的权限

grant all privileges on *.* to 'root'@'%' identified by 'xiaoxiao' with grant option;

PS1: % 代表任意主机

use mysql;

grant all privileges on *.* to 'root'@'%' identified by 'xiaoxiao' ;

grant all privileges on *.* to 'root'@'192.168.195.%' identified by 'xiaoxiao' ;

案例：

grant all privileges on *.* to 'root'@'192.168.195.%' identified by 'xiaoxiao';

select user,host,password from mysql.user;

PS: privileges关键字可以省略

[案例2]

grant select,insert on testdb.* to 'sdxh'@'192.168.195.%' identified by 'xiaoxiao';

撤销权限

revoke all privileges on *.* from 'root'@'%' ;

revoke select,insert on testdb.* from 'sdxh'@'192.168.195.%';

flush privileges; 刷新权限

新用户

grant all on *.* to admin@'%' identified by 'yourpassword' with grant option;

删除用户

drop user 'john'@'192.168.13.34';

【练习】

create database blog;

show databases;

grant all privileges on *.* to 'root'@'192.168.195.%' identified by 'xiaoxiao';

flush privileges;

show grants;

show grants for 'root'@'192.168.195.%'

select user,host,password from mysql.user;

另一台主机测试

mysql –h 192.168.195.76 –uroot –pxiaoxiao

创建一个普通用户并授权

grant select,insert,update,delete on blog.* to 'xh'@'192.168.195.%' identified by 'xiaoxiao';

flush privileges;

show grants for 'xh'@'192.168.195.%';

另一台主机测试

mysql –h 192.168.195.76 –uxh –pxiaoxiao

show databases;

撤销权限

revoke select,insert,update,delete on blog.* from 'xh'@'192.168.195.%';

flush privileges;

另一台主机测试

mysql –h 192.168.195.76 –uxh –pxiaoxiao

show databases;

删除用户

drop user 'xh'@'192.168.195.%';

select user,host from mysql.user;

mysql –h 192.168.195.76 –uxh –pxiaoxiao

一、grant 普通数据用户，查询、插入、更新、删除 数据库中所有表数据的权利。

grant select on testdb.* to common_user@'%'

grant insert on testdb.* to common_user@'%'

grant update on testdb.* to common_user@'%'

grant delete on testdb.* to common_user@'%'

或者，用一条 MySQL 命令来替代：

grant select, insert, update, delete on testdb.* to common_user@'%'

二、grant 数据库开发人员，创建表、索引、视图、存储过程、函数。。。等权限。

grant 创建、修改、删除 MySQL 数据表结构权限。

grant create on testdb.* to developer@'192.168.0.%';

grant alter on testdb.* to developer@'192.168.0.%';

grant drop on testdb.* to developer@'192.168.0.%';

grant 操作 MySQL 外键权限。

grant references on testdb.* to developer@'192.168.0.%';

grant 操作 MySQL 临时表权限。

grant create temporary tables on testdb.* to developer@'192.168.0.%';

grant 操作 MySQL 索引权限。

grant index on testdb.* to developer@'192.168.0.%';

grant 操作 MySQL 视图、查看视图源代码 权限。

grant create view on testdb.* to developer@'192.168.0.%';

grant show view on testdb.* to developer@'192.168.0.%';

grant 操作 MySQL 存储过程、函数 权限。

grant create routine on testdb.* to developer@'192.168.0.%'; -- now, can show procedure status

grant alter routine on testdb.* to developer@'192.168.0.%'; -- now, you can drop a procedure

grant execute on testdb.* to developer@'192.168.0.%';

三、grant 普通 DBA 管理某个 MySQL 数据库的权限。

grant all privileges on testdb to dba@'localhost'

其中，关键字“privileges”可以省略。

四、grant 高级 DBA 管理 MySQL 中所有数据库的权限。

grant all on *.* to dba@'localhost'

五、MySQL grant 权限，分别可以作用在多个层次上。

1. grant 作用在整个 MySQL 服务器上：

grant select on *.* to dba@localhost; -- dba 可以查询 MySQL 中所有数据库中的表。

grant all on *.* to dba@localhost; -- dba 可以管理 MySQL 中的所有数据库

2. grant 作用在单个数据库上：

grant select on testdb.* to dba@localhost; -- dba 可以查询 testdb 中的表。

3. grant 作用在单个数据表上：

grant select, insert, update, delete on testdb.orders to dba@localhost;

这里在给一个用户授权多张表时，可以多次执行以上语句。例如：

grant select(user_id,username) on smp.users to mo_user@'%' identified by '123345';

grant select on smp.mo_sms to mo_user@'%' identified by '123345';

4. grant 作用在表中的列上：

grant select(id, se, rank) on testdb.apache_log to dba@localhost;

5. grant 作用在存储过程、函数上：

grant execute on procedure testdb.pr_add to 'dba'@'localhost'

grant execute on function testdb.fn_add to 'dba'@'localhost'

六、查看 MySQL 用户权限

查看当前用户（自己）权限：

show grants;

查看其他 MySQL 用户权限：

show grants for dba@localhost;

七、撤销已经赋予给 MySQL 用户权限的权限。

revoke 跟 grant 的语法差不多，只需要把关键字“to”换成“from”即可：

grant all on *.* to dba@localhost;

revoke all on *.* from dba@localhost;

八、MySQL grant、revoke 用户权限注意事项

1. grant, revoke 用户权限后，该用户只有重新连接 MySQL 数据库，权限才能生效。

2. 如果想让授权的用户，也可以将这些权限 grant 给其他用户，需要选项“grant option“

grant select on testdb.* to dba@localhost with grant option;

这个特性一般用不到。实际中，数据库权限最好由 DBA 来统一管理。

Linux黑客基础-87-Mariadb-07-数据库管理员口令忘记了自么办

Step01:用以下方式重新启动mysql（以不检查权限的方式启动）

方法1

mysqld_safe --skip-grant-tables &

PS1: & 表示把程序放在后台执行

PS2: --skip-grant-tables //跳过权限检查

方法2：修改配置文件（/etc/mysql/mariadb.conf.d/50-server.cnf）

在[mysqld]下添加 skip-grant-tables , 再启动mysql

Step02:

重新连接数据库

mysql -uroot -p //此时不需要密码

Step03: 更改root用户口令

flush privileges;

use mysql;

ALTER USER 'root'@'localhost' identified by 'xiaoguang';

flush privileges; //刷新权限

//早先版本

update mysql.user set Password=password('xiaoguang') where User='root';

Step04:改完之后结束Mysql进程，并重启即可

方法1

ps aux | grep mysqld

killall -9 mysql相关进程

killall -9 mariadbd

方法2：修改配置文件

如果是修改配置文件，把添加的配置删除或注释掉即可

systemctl stop mysql

systemctl start mysql

Linux黑客基础-88-PostgreSQL与 Metasploit

PostgreSQL，或 Postgres，是另一个开源关系数据库，由于其能够轻松扩展并处理繁重的工作负载，

因此常用于非常大的互联网应用程序。是Metasploit的默认数据库。

监听在tcp/5432端口

Metasploit 是一个漏洞利用框架

msfdb init && msfconsole

PS1: && 只有前一个程序正确运行了，才会运行后面的程序

|| 只有前一个程序运行失败了，才会运行后面的程序

&

PS2: msfdb init # start and initialize the database //启动和初始化数据库

启动数据库（systemctl start postgresql）

完成数据库的初始化操作

创建一个名为msf3的PostgreSQL数据库用户

创建数据库msf、msf_test

PS3: msfconsole （Metasploit命令行接口）

msfdb reinit # delete and reinitialize the database

msfdb delete # delete database and stop using it

msfdb start # start the database

msfdb stop # stop the database

msfdb status # check service status //检查数据库服务的状态

msfdb run # start the database and run msfconsole //启动数据库并运行 msfconsole

PS4：

msf6 > db_status //查看数据库的状态

PS5:metasploit的数据库配置文件/usr/share/metasploit-framework/config/database.yml