



批量检测SQL注入

 [muhe](#)  2018-01-11 共156403人围观，发现6个不明物体

WEB安全

0×01 前言

SQL注入，这个类型的漏洞我真的学了好久好久好久好久，即是我刚刚开始接触安全就学习的第一种漏洞，也是一直至今为止还在学习的漏洞类型，只能说，感觉自己还是有很多还是不会的。从一开始的手工一个网站一个网站去测到之后的用google hacking的方法去找可疑链接，再到后面用sqlmap批量检测。也是经历了至少半年的时间。今天写了个调用sqlmapapi的脚本，想跟大家分享一些走过的坑和思路。

0×02 SQL注入批量测试的几种方法

本文的目的在于通过看别人的代码来学习原理，同时也掌握自己造轮子的能力。这里只是列举了我自己平时用到的几种方法，当然网上能找到更多更好的工具，或者更好的思路，都是值得学习的。

下面就列举三种方法是：sqliv、sqlmap -m、sqlmapapi。

1.1 sqliv

1.1.1 下载地址：

【<https://github.com/Hadesy2k/sqliv>】

1.1.2 说明：





作者是Hadesy2k，可以看到作者大大更新还是挺勤快的，写这篇文章的日期是11月23日（文章第一版）（第二版已经12月28日），从图里可以看出作者在月初的时候更新了一波。

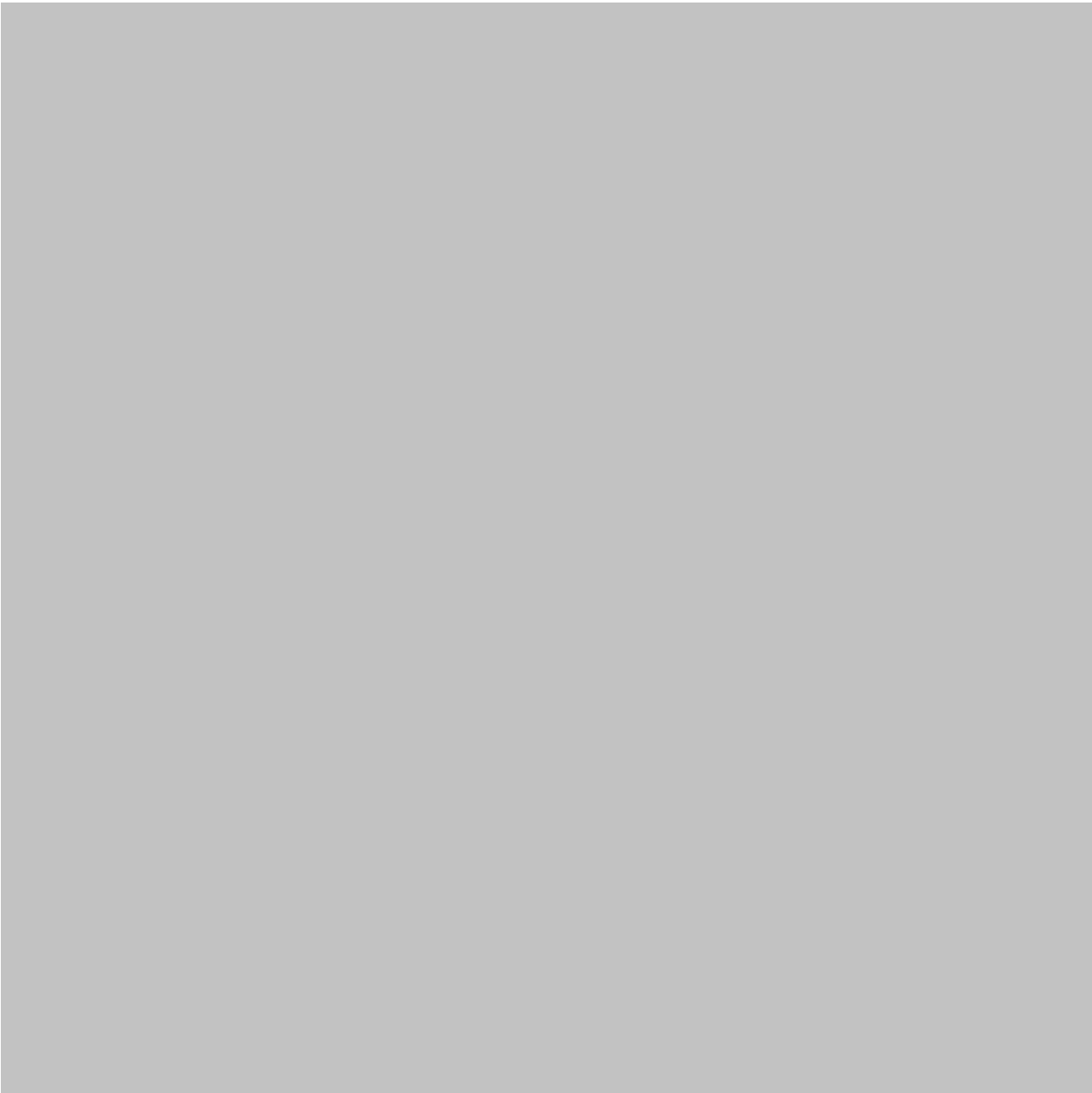
而更之前在10月份的时候我看了一遍这个代码，发现作者在检测sql注入点的时候只是在参数值后加了个单引号，来检测返回页面的报错信息。以这种方式去测试当然会遗漏很多呀。而且如果一个链接后跟多个参数，同时所有参值里加单引号。我头上直冒汗。

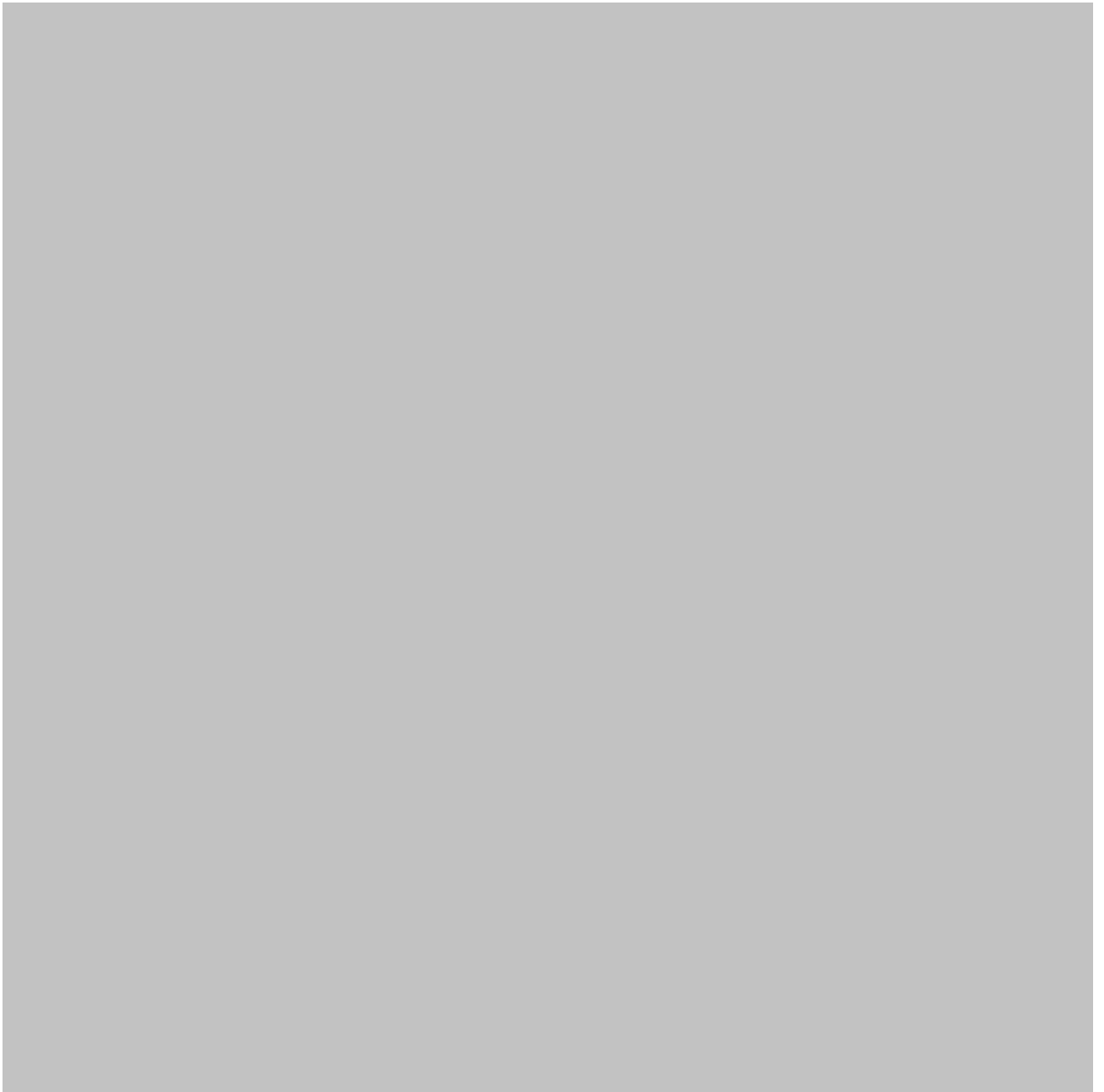
在【src/scanner.py】里可以看到这一句：

```
【website = domain + "?" + ("&".join([param + payload for param in queries]))】
```

每个参数值加payload后组合一起，现在依然是这种方式。

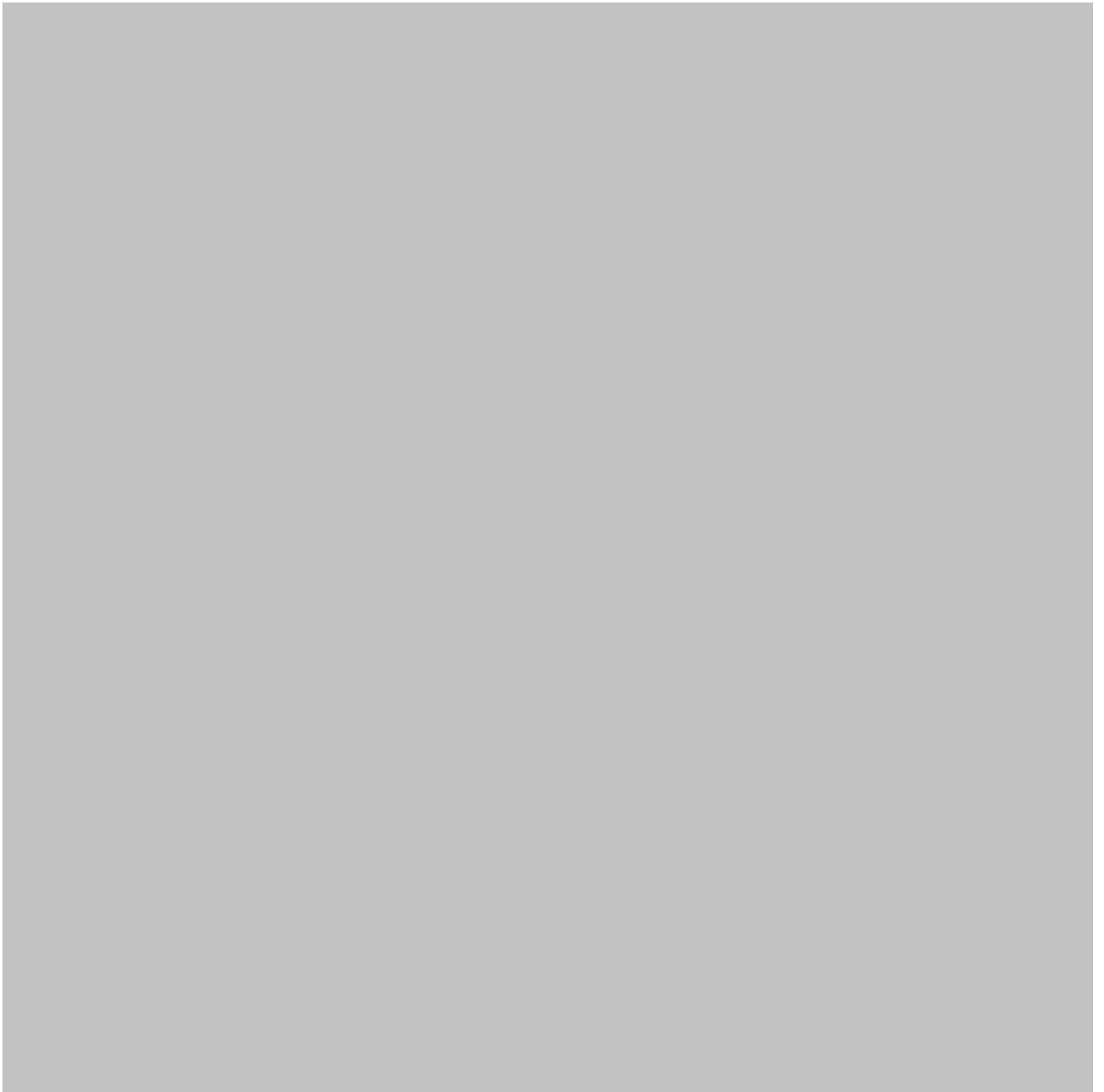






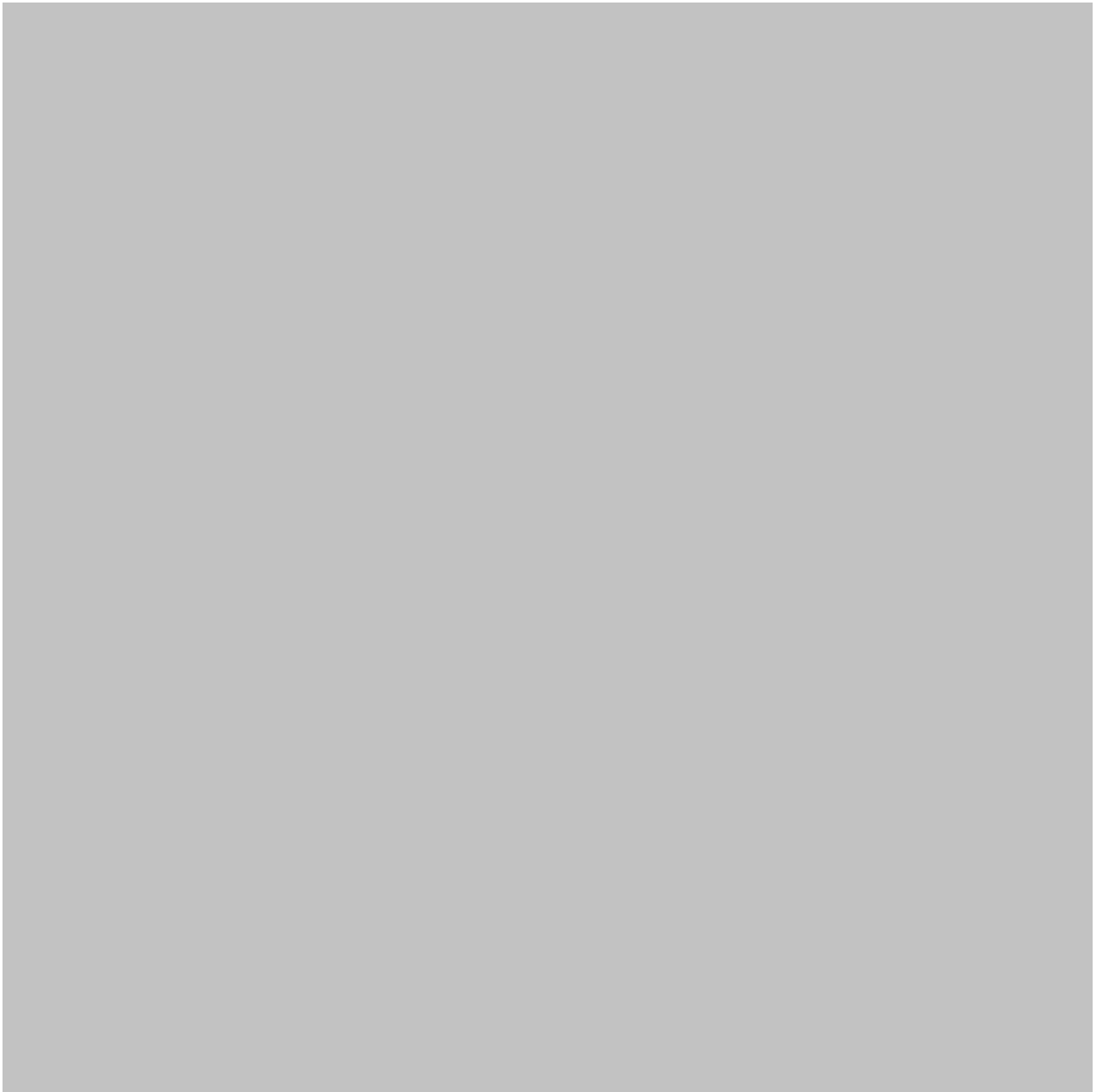
这个是lesson是没有注入的，参数内容改变的话会影响页面返回结果，所以正确的方法应该是单独给每个参数的内容payload进行测试。





我自己改了下，变成分别单独测试所有参数值，有需要的话。大家可以去我的github【<https://github.com/Martin2877/sqliv-M/blob/master/src/scanner.py>】看。





值的高兴的是，作者把检测POST请求放在他的【To Do】里了，期待。





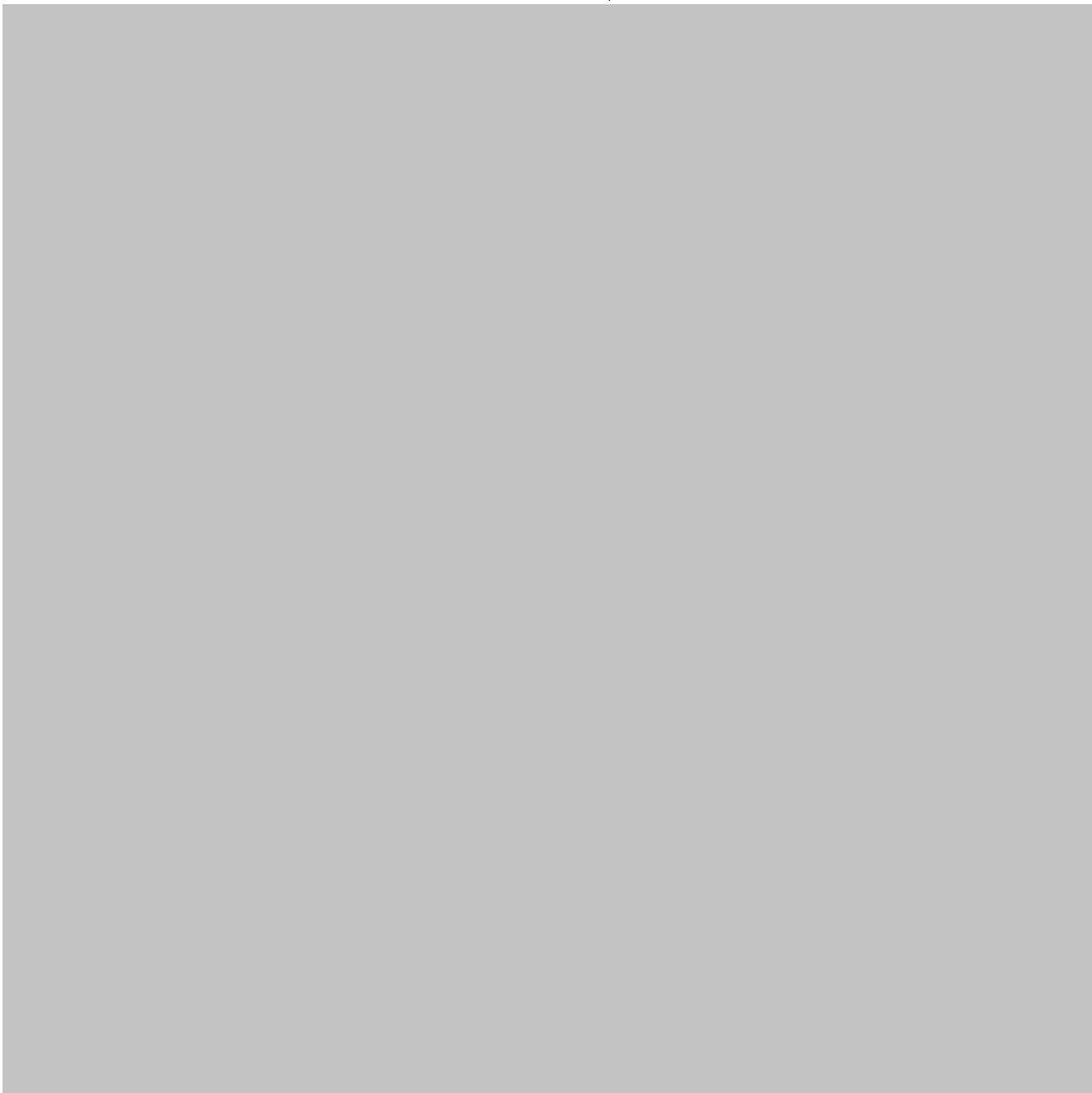
（而当我12月28号再去看的时候，作者又更新了许多，还加入了docker更方便部署。点个赞。）



1.1.3 多地址

因为sqliv本身是没有多地址批量的，所以我们自己写一个咯，还是比较简单的，见下图，代码也在我的github里





1.2 sqlmap -m

1.2.1 下载地址：

【<https://github.com/sqlmapproject/sqlmap>】

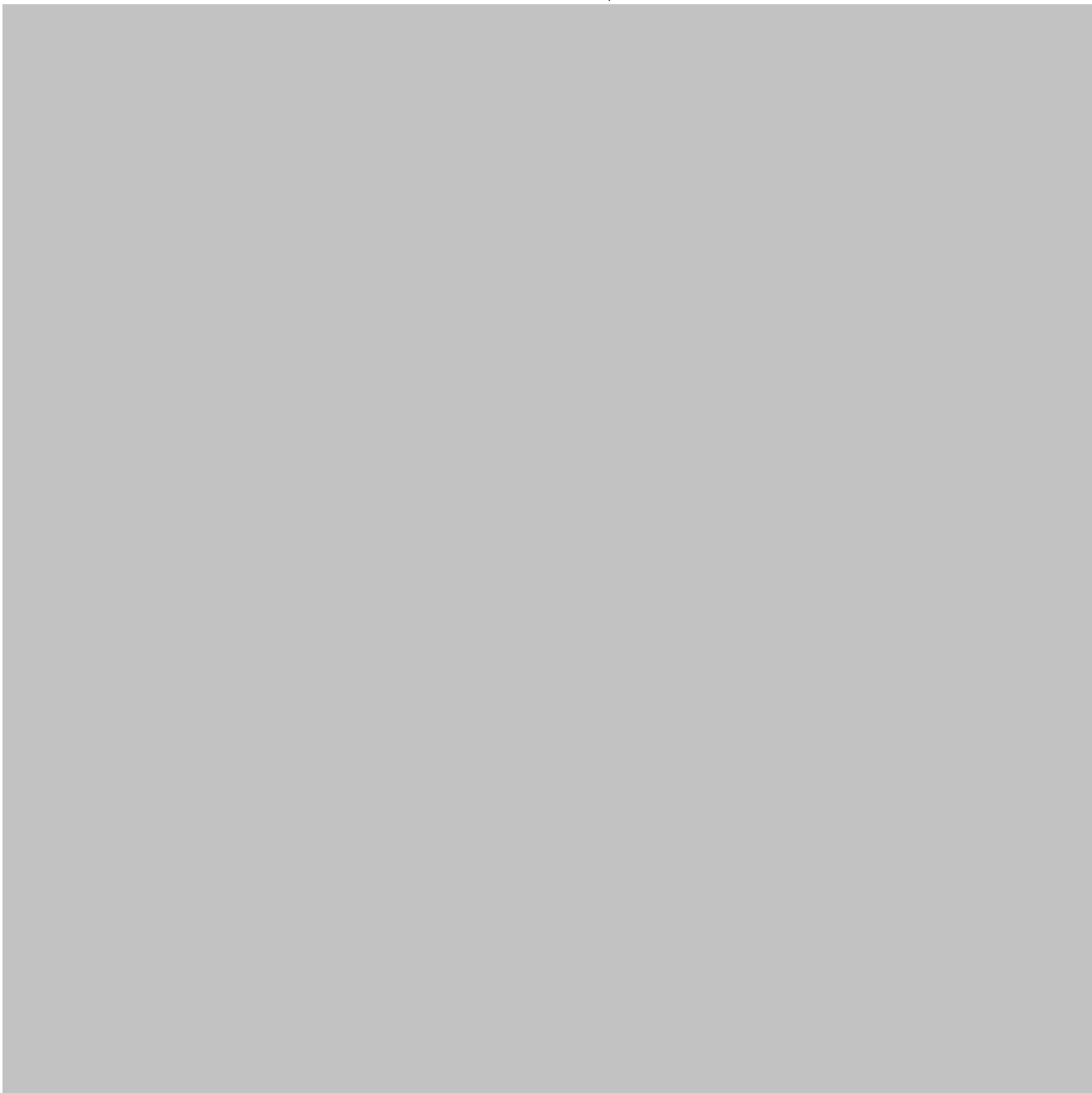
1.2.2 说明：

神器sqlmap，不多说，-m模式，就是用来批量测试url的，不过是单线程，比较慢，效果比sqlmapapi好。

使用命令：

```
python sqlmap.py -m urls.txt --batch
```





1.3 sqlmapapi

1.3.1 下载地址：

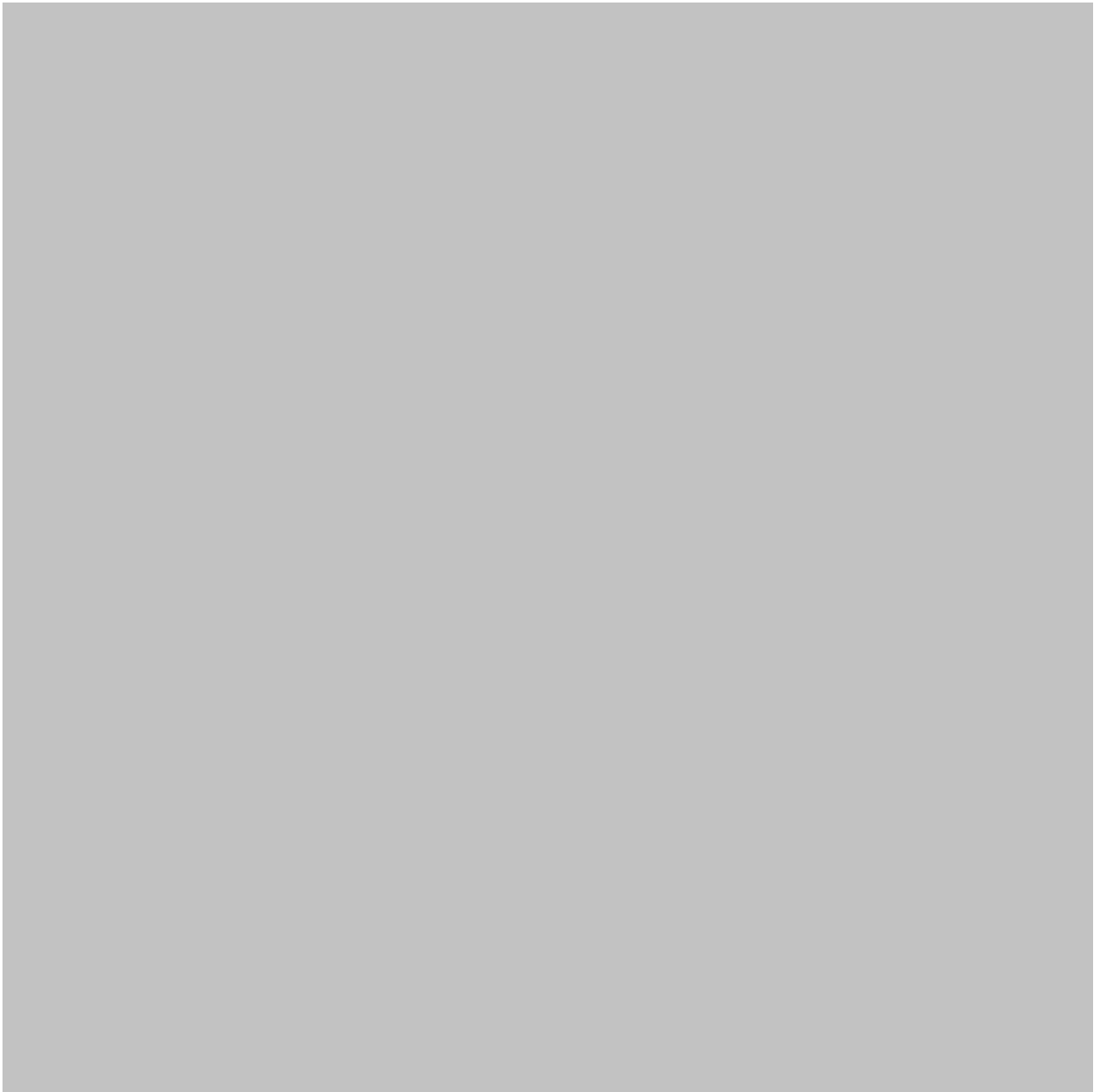
【<https://github.com/Martin2877/sqlmapapi-M>】

1.3.2 说明：

api模式有多线程，先通过sqlmap根目录中的sqlmapapi.py启动服务：

【python sqlmapapi.py -s】





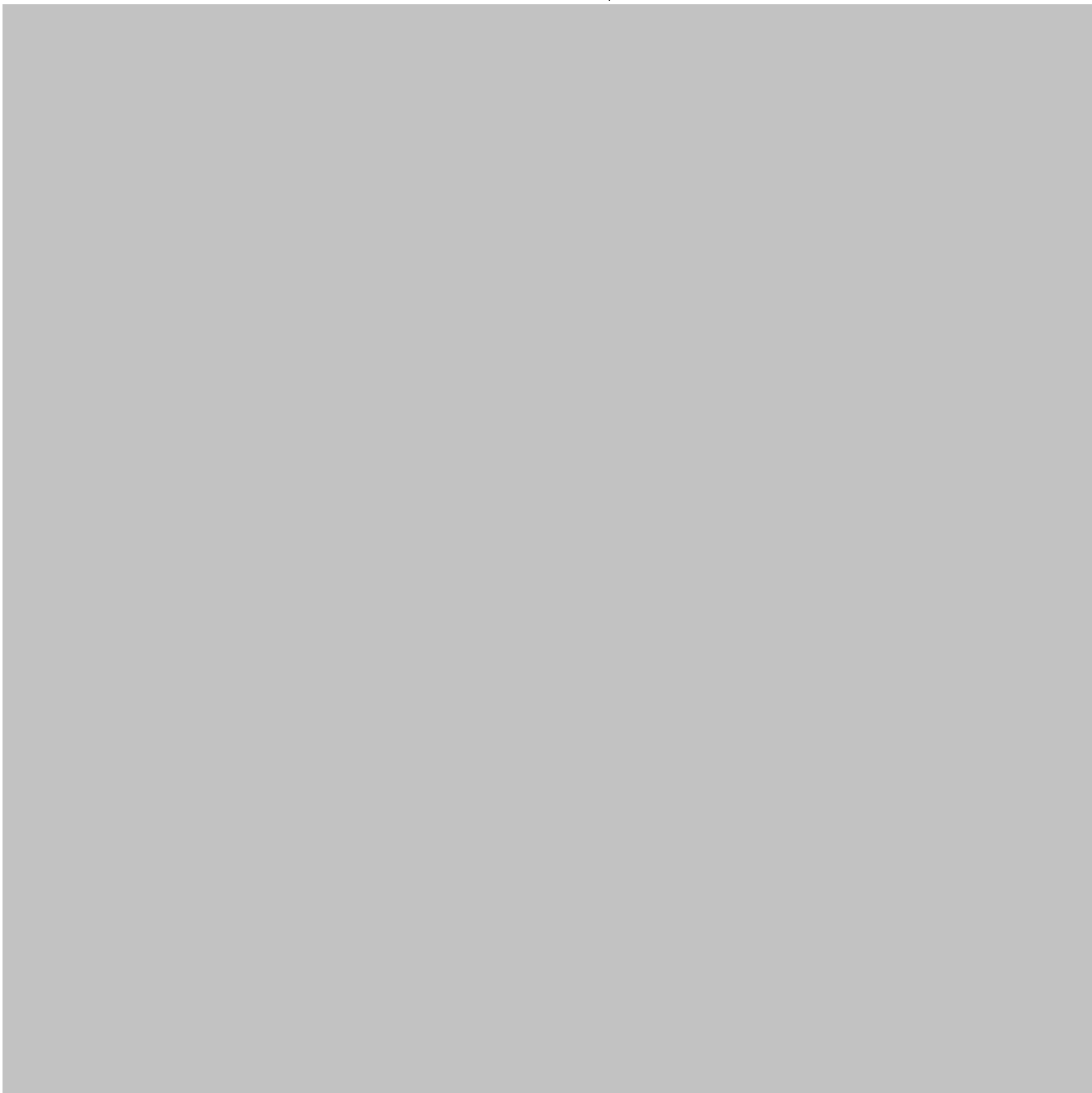
然后到sqlmapapi-M根目录下，在【url.txt】中写你要批量测试的地址：





然后使用命令执行批量扫描：【python sqlmapbatch.py】





效果如图，而其关键代码如下：





这里设置了超时时间是4秒，然后测下一个，大家也可以设置更久一点，效果也会更好。我这里没有做队列什么的，大家可以自己改哈。

0×03 爬取链接

当然上面几种方法都需要用到url.txt，针对这个我另外写了一个爬虫工具，见

【<https://github.com/Martin2877/FindLinks>】，是爬取一个网站下链接的工具；网上还有很多能通过搜索找可能有sql注入的网站，这里就不多描述啦。

0×04 总结

总结一下，上面所列的几个工具或方法效果最好的当然是【sqlmap】，因为没有“超时”这个地址会一直测到底



多，不过效果没有那么好。以后加了队列应该就能顾及两个方面的问题。而sqliv相对来说比较快，因为只是加payload看返回是否有数据库报错（也是其缺点），当然可以改代码变成加载payload字典的方式去测，也算是不错的一个工具。

以上，就是这么多啦，祝愿大家能更快更多地挖出sql注入哦~

*本文作者：[muhe](#)，转载请注明来自FreeBuf.COM

上一篇：[经验分享 | 基于代理IP的挖掘与分析](#)

下一篇：[挖洞经验 | 看我如何发现价值\\$10000美金的雅虎Cookie窃取漏洞](#)

这些评论亮了



plane636

用啊D啊，5分钟就解决了（笑

[回复](#)

[亮了](#) (8)

已有 6 条评论

plane636 2018-01-11

1楼 [回](#)

用啊D啊，5分钟就解决了（笑

[亮了](#) (8)

[死宅10086](#) (7级) 这家伙太懒了，还未填写个人描述！ 2018-01-11

@ plane636 哈哈 🤔

[亮了](#) (8)

做题大牛 2018-01-11

2楼 [回](#)

sql注入你才经历半年。。。我已经研究sql注入快7年了，加油！

[亮了](#) (8)

Onls2018-01-12

3楼回

你好，那个爬链接的脚本，我刚试了下执行无效。

亮了

muhe(1级)这家伙太帅了，还未填写个人描述!2018-01-12

@ Onls 是不是缺少模块了呢？还是什么报错呢

亮了

巅峰邪恶2018-01-12

4楼回

还是觉得啊D好用啊

亮了

选择文件

未选择任何文件

昵称

请输入昵称

必须 您当前尚未登录。[登陆?](#) [注册](#)

邮箱

请输入邮箱地址

必须 (保密)

表情

插图

提交评论(Ctrl+Enter)

[取消](#)

☒ 有人回复时邮件通知我



[muhe](#)

这家伙太帅了，还未填写个人描述!

1

文章数

2

评论数

● [批量检测SQL注入](#)

2018.01.11

[浏览更多](#)

相关阅读

[\[更新\]Gui-for-sqlmap Version17612](#)

[一个有趣的实例让NoSQL注入不再神秘](#)

[CVE-2015-7857 Joomla!注入漏洞利用工具](#)

[在SQL注入中利用MySQL隐形的类型...](#)

[用Mitmproxy辅助Sqlmap自动化利用特...](#)

特别推荐



[【FB TV】一周「BUF本事件」：
WPA2惊现高危漏洞，你的WiFi还](#)

[技术剖析：海莲花（OceanLotus）
根本不是APT，它只是一个普通木](#)

[【已结束】第三届中国\(北京\)军民融合技术装备博览会现场实况](#)

[Elaine_z](#)

2017-07-03


[Windows 10新变化：系统自动更新将“强制化”，用户不再可选](#)

[dawner](#)

2015-07-20



Copyright © 2018 WWW.FREEBUF.COM All Rights Reserved [沪ICP备13033796号](#)

 阿里云 提供计算与安全防护服务