# 基于句法敏感的程序分析进行漏洞范围界定方法

1. Redebug 开源链接: https://github.com/dbrumley/redebug
2. 可参考 https://blog.csdn.net/yalecaltech/article/details/107226303
3. 运行前的准备工作

①python 版本要为 2.x

②安装对应的库：

## Dependencies

- `bitarray`, `python-magic`, and `argparse` modules: `pip install bitarray python-magic argparse`
- `libmagic` package: `apt-get install libmagic-dev` on Ubuntu/Debian, `brew install libmagic` on OSX

如果 bitarry 或者某个库安装不成功，可以参考：https://blog.csdn.net/qzzzxiaosheng/article/details/125119006 安装对应的 whl 文件

③下载 redebug

④将需要检查的程序源代码放在一个文件夹，将补丁以.diff 的格式放在另外一个文件夹下面，这两个文件夹的地址中不要有空格，地址长度不要太长，少于 259 个字符。

4. 使用方法：

本方法以漏洞软件的代码仓库、版本信息以及漏洞补丁为输入，来分析漏洞的影响版本范围。具体来说分为两大步骤。

步骤 1：获取软件各版本源代码。具体来说，结合版本信息中的版本号，可在代码仓库中切换到各版本发布对应的 commit，获取每个版本的源代码。此外，可以通过开源软件相应官网下载区下载各版本源代码。

步骤 2：判断各个版本是否受漏洞影响。识别一个给定的软件版本是否受特定漏洞影响主要参考了安全顶会论文[6]redebug 的思路。该方法以源码和漏洞补丁为输入。该方法首先分析补丁，基于规则从中提取漏洞代码特征；然后采用了句法敏感的程序分析技术，扫描所有程序代码是否存在漏洞代码特征，如果存在，就认为该程序版本受该漏洞影响。

Redebug 使用方法：

```
$ python redebug.py -h
usage: redebug.py [-h] [-n NUM] [-c NUM] [-v] patch_path source_path

positional arguments:
  patch_path            path to patch files (in unified diff format)
  source_path           path to source files

optional arguments:
  -h, --help            show this help message and exit
  -n NUM, --ngram NUM   use n-gram of NUM lines (default: 4)
  -c NUM, --context NUM
                        print NUM lines of context (default: 10)
  -v, --verbose         enable verbose mode (default: False)
```

*建议检测的时候根据 diff 文件找一下对应的文件所在位置，缩小一下文件范围，省时间而

且少出 bug

5. 示例

我们用该方法来对 CVE-2021-25640 进行漏洞的范围鉴定，通过人工检测：

该漏洞影响的组件版本：

Dubbo 2.7.0 到 2.7.9 版本

Dubbo 2.6.0 到 2.6.9 版本

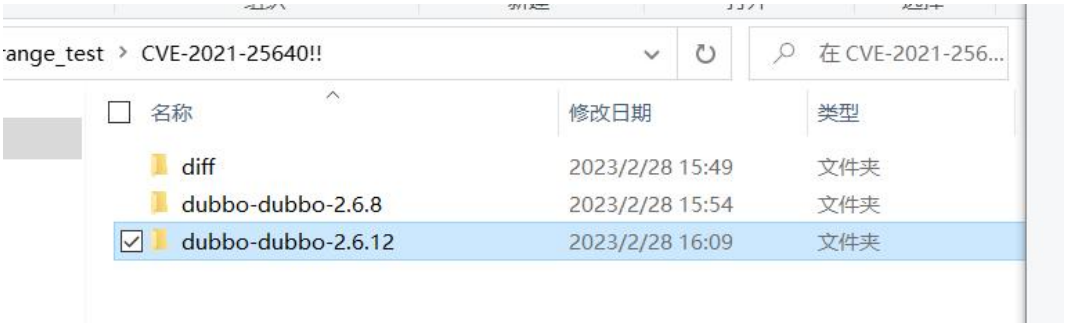
Dubbo 所有 2.5.x 版本 (不再被官方维护)

漏洞修复的组件版本:

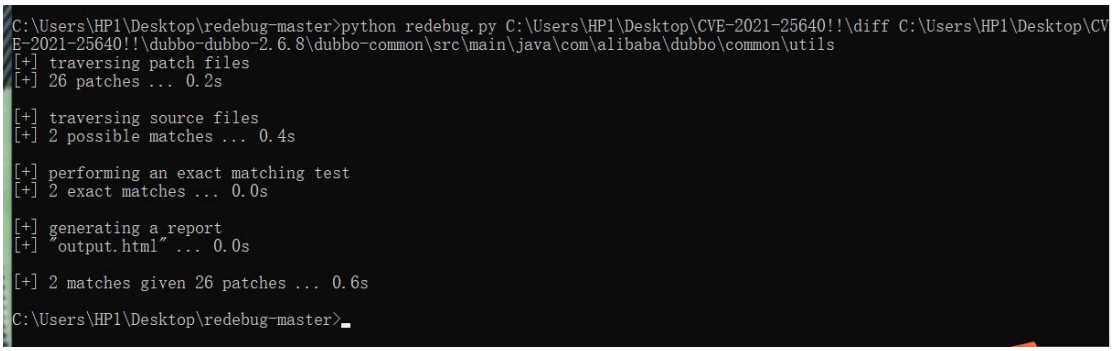
Dubbo 2.6.9 2.7.9

所以我们用 redebug 测试 dubbo-dubbo-2.6.8 和 dubbo-dubbo-2.6.12 是否含有漏洞。

将对应的.diff 和源代码放入对应的文件夹中：




分别进行测试：



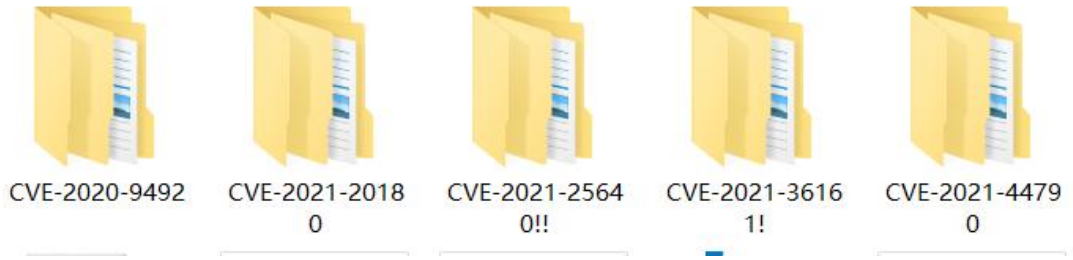

2.6.8 版本显示有 patch，对应的 output.html 文件显示了对应的位置：


2.6.12 版本显示没有 patch：

```
C:\Users\HP1\Desktop\redebug-master>python redebug.py C:\Users\HP1\Desktop\CVE-2021-25640!!\diff C:\Users\HP1\Desktop\CV
E-2021-25640!!\dubbo-dubbo-2.6.12\dubbo-common\src\main\java\com\alibaba\dubbo\common\utils
[+] traversing patch files
[+] 26 patches ... 0.2s

[+] traversing source files
[+] 0 possible matches ... 0.4s

[!] no match to be checked

C:\Users\HP1\Desktop\redebug-master>
```

6. 成功率

测试了五个 cve：




成功了一个，还有一个是都检测出了 patch，其他三个是都没有检测出 patch，检测成功率感人。