AWD二进制运维工具-AutoAWD交流分享

Badrer / 2018-12-14 14:01:00 / 浏览数 2811 安全工具 工具 顶(0) 踩(0)

前言

这次PCB-AWD打的真是憋屈，话说第二天pwn题都全解了，只因为手速太慢（辛苦队友手交flag），导致最后离奖金还差一点，所以这次必须要开发一套二进制AWD自动化

工具简介

主要是两个主文件：

■■■■■■■■■■`ssh`■■■■■■■■ AutoBinary.py
■■■pwn■■■■■■■■■■■■ batch_submit.py

0x1 AutoBinary

设计思路

主要利用paramiko模块建立ssh和sftp连接，然后通过命令执行达到一键dump的目的。

ssh有两种登录方式：

1. 用户名+密码
2. 用户名+私钥

此工具已将两种方式集成

```
usage: AutoBinary.py [-h] [--dump] [--find] [--get] [--put] [-c COMMAND]
                     [-ip HOSTNAME] [-P PORT] [-u USERNAME] [-p PASSWD]
                     [-k KEYFILE] [-r REMOTEPATH] [-l LOCALPATH] [-v]

optional arguments:
 -h, --help            show this help message and exit
 --dump                To dump binary source
 --find                To get the pathslist of flag
 --get                 To dump choose file from remote host,but if you want
                       get without losting privilege you prefect using [scp
                       -P user@hostname:remote_file local_file] to get file
 --put                 To put choose file to remote host■but if you want
                       put without losting privilege you prefect using [scp
                       -P port local_file user@hostname:remote_file] to put
                       file
 -c COMMAND, --command COMMAND
                       To exec command by ssh
 -ip HOSTNAME, --hostname HOSTNAME
                       Input remote hostname[ip]
 -P PORT, --port PORT  Input remote ssh or sftp port
 -u USERNAME, --username USERNAME
                       Input remote ssh username
 -p PASSWD, --passwd PASSWD
                       Input remote ssh passwd
 -k KEYFILE, --keyfile KEYFILE
                       Input ssh key file
 -r REMOTEPATH, --remotepath REMOTEPATH
                       Input remotepath file name to dump or overwrite it
 -l LOCALPATH, --localpath LOCALPATH
                       Input localpath file name
 -v, --version         Edit by BadRer V1.0
```

主要功能参数：

--dump　　■■`dump` pwn■■■■■■■■■`dump`■■■■■■■■■■■■■■

--find　　■■■■■■■■■■`flag`■■■■■■

--get　　■■■■■■■■■■■■ ■■`scp -P port user@ip:remote_file local_file`

--put   ■■■■■■■■■■■■ ■■`scp -P port local_file user@ip:remote_file` ,■■■■■■■■■■■■■■■■■■■■■■■■■■■■`scp`

--command   ■■`ssh`■■■■■■■■■■■■■■■■■■■■

常用的命令：

```
python AutoBinary.py -ip 192.168.43.252 -P 22 -u pwn -p 123 --dump
python AutoBinary.py -ip 192.168.43.252 -P 22 -u pwn -p 123 --find
python AutoBinary.py -ip 192.168.43.252 -P 22 -u pwn -p 123 --command 'ls'
python AutoBinary.py -ip 192.168.43.252 -P 22 -u pwn -p 123 --get -r '/home/pwn/pwn1' -l pwn1
python AutoBinary.py -ip 192.168.43.252 -P 22 -u pwn -p 123 --put -l pwn1 -r '/home/pwn/pwn1'
```

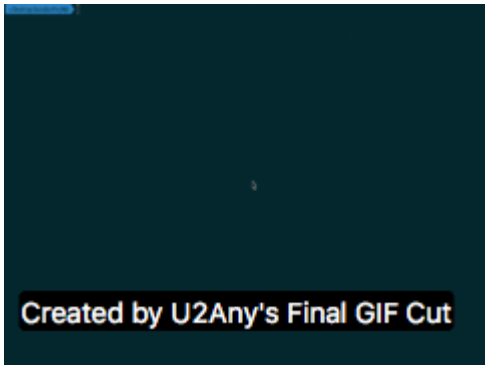
测试

经过本地测试通过用户名密码登录的方式可以成功运行

通过用户名和私钥登录的方式还有待测试。

0x2 batch_submit

批量自动化提交工具，等到队友的exp便可以打遍全场！



设计思路

通常pwn的批量自动化分为三步 ：

1. 写自动化提交（这得看主办方是否提供相应的接口，否则只能手动交flag了，哈哈）
2. 写出exp自动获取flag
3. 批量轮询所有ip，自动获取flag并提交

我实现了多线程的方式，并在本地测试了两个ip的情况，对于线上赛20多个ip的情况还有待实战！

使用说明：

1. 首先完善autoUtil/auto_submit.py中的submit_flag函数实现自动交flag
2. 完善autoUtil/auto_getflag.py中的 auto_get_submit 函数 实现自动cat flag 并且提交，此时需要使用有效的exp
3. 运行 batch_submit.py 修改对应的iplist和port，以及采用何种批量方式

批量化实现思路：

多线程

利用threading模块，对于两个主机的批量测试通过，针对大量线程不知运行效果如何。

多进程

下一步可以尝试的方向。

注意事项

此工具所在路径不能包含中文字符

此工具仅在mac以及ubuntu下测试通过，windows尚未测试。

后期的一些设想

可以再增加一些自动探测漏洞，自动种马的功能，并针对此工具的bug进行完善。

写在最后

由于某些特殊原因，不宜将此上传至github，基于交流分享的精神，特将思路和部分代码分享出来。

> 如果各位师傅对此感兴趣可以联系我一起合作，欢迎大家一起交流。QQ：1057947291
> 链接:https://pan.baidu.com/s/1VgrStl6-qiki7LAajSVFDw 密码:1lpo

上一篇：利用域名碰撞实现从任何地方发起中间人攻击 下一篇：ADIDNS 安全研究：绕过 GQ...

1. 0 条回复
   - 动动手指，沙发就是你的了！