



链滴

用 Python 实现 windows 下 ping 程序

作者: [cuijianzhe](#)

原文链接: <https://ld246.com/article/1587195363747>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

背景

公司直播间直播课程，有时会出现发题、金币刷新等卡顿问题，为了解释不是网络而是接口或者服务题，此程序诞生。

因为不带多线程版本ping程序挂钩多个ip的话，容易出现时间缺失（如下），所以加上多线程，同时序简化了很多.....

```
2020-04-19 18:20:41 来自 192.168.100.1 的回复: 字节=32 时间<1ms TTL=64
2020-04-19 18:20:42 来自 192.168.100.1 的回复: 字节=32 时间<1ms TTL=64
2020-04-19 18:20:44 来自 192.168.100.1 的回复: 字节=32 时间<1ms TTL=64
2020-04-19 18:20:45 来自 192.168.100.1 的回复: 字节=32 时间<1ms TTL=64
2020-04-19 18:20:46 来自 192.168.100.1 的回复: 字节=32 时间<1ms TTL=64
2020-04-19 18:20:47 来自 192.168.100.1 的回复: 字节=32 时间<1ms TTL=64
2020-04-19 18:20:48 来自 192.168.100.1 的回复: 字节=32 时间<1ms TTL=64
2020-04-19 18:20:49 来自 192.168.100.1 的回复: 字节=32 时间<1ms TTL=64
2020-04-19 18:20:51 来自 192.168.100.1 的回复: 字节=32 时间<1ms TTL=64
2020-04-19 18:20:52 来自 192.168.100.1 的回复: 字节=32 时间<1ms TTL=64
2020-04-19 18:20:53 来自 192.168.100.1 的回复: 字节=32 时间<1ms TTL=64
2020-04-19 18:20:54 来自 192.168.100.1 的回复: 字节=32 时间<1ms TTL=64
```

ping参数自选

用法: ping [-t] [-a] [-n count] [-l size] [-f] [-i TTL] [-v TOS]
[-r count] [-s count] [[-j host-list] | [-k host-list]]
[-w timeout] [-R] [-S srcaddr] [-c compartment] [-p]
[-4] [-6] target_name

选项:

- t Ping 指定的主机，直到停止。
若要查看统计信息并继续操作，请键入 Ctrl+Break；
若要停止，请键入 Ctrl+C。
- a 将地址解析为主机名。
- n count 要发送的回显请求数。
- l size 发送缓冲区大小。
- f 在数据包中设置“不分段”标记(仅适用于 IPv4)。
- i TTL 生存时间。
- v TOS 服务类型(仅适用于 IPv4。该设置已被弃用，
对 IP 标头中的服务类型字段没有任何影响)。
- r count 记录计数跃点的路由(仅适用于 IPv4)。
- s count 计数跃点的时间戳(仅适用于 IPv4)。
- j host-list 与主机列表一起使用的松散源路由(仅适用于 IPv4)。
- k host-list 与主机列表一起使用的严格源路由(仅适用于 IPv4)。
- w timeout 等待每次回复的超时时间(毫秒)。
- R 同样使用路由标头测试反向路由(仅适用于 IPv6)。
根据 RFC 5095，已弃用此路由标头。
如果使用此标头，某些系统可能丢弃回显请求。
- S srcaddr 要使用的源地址。
- c compartment 路由隔离舱标识符。
- p Ping Hyper-V 网络虚拟化提供程序地址。
- 4 强制使用 IPv4。
- 6 强制使用 IPv6。

带多线程ping指定主机

```
#!/bin/python3
```

```
#####
```

```

# This program is to check the ping in the Live Room #
# Date: 2020-4-21 #
# Author: cuijianzhe #
# Email: 598941324@qq.com #
#####
import threading
import os
import subprocess
import re
import time
import datetime as dt
def ping_host(ip):
    nowdate = dt.datetime.now().strftime('%Y%m%d') # 获取文件日期后缀
    p = subprocess.Popen(["ping.exe", '-n', '1', ip],
                          stdin=subprocess.PIPE,
                          stdout=subprocess.PIPE,
                          stderr=subprocess.PIPE,
                          shell=True)
    out = p.stdout.read().decode('gbk')
    ping = re.search(r'来自.+|请求..+', out).group()
    now_time = dt.datetime.now().strftime('%F %T')
    file_name = 'ping_%s_%s.txt' % (ip, nowdate)
    with open('C:\limi_ppt_log\ping\%s' % file_name, 'a') as f:
        f.write(str(now_time) + ' ' + str(ping))

if __name__ == '__main__':
    time.sleep(10)
    with open('./hosts.txt', 'r') as f:
        hosts = f.read().split("\n")

    path = "C:\limi_ppt_log\ping"
    if os.path.isdir(path):
        while True:
            time.sleep(1)
            for IP in hosts:
                #多线程同时执行
                thread = threading.Thread(target=ping_host, args=(IP,))
                thread.start()
    else:
        os.makedirs(path) #可创建多级目录
        while True:
            time.sleep(1)
            for IP in hosts:
                thread = threading.Thread(target=ping_host, args=(IP,))
                thread.start()

```

ping程序（不带线程） subprocess:

- 自动判断目录存在否则生成存放日志目录
- 请求返回值只要超过1s则判定超时
- ping请求返回值加上时间戳

```
# -*- coding:utf8 -*-
```

```
#!/usr/bin/python3
import subprocess
import re
import datetime as dt
import time
nowdate = dt.datetime.now().strftime('%Y%m%d')
class LinkState(object):
    def __init__(self,ip):
        self.ip = ip
        self.getLinkState(self.ip)

    # 获取链路状态
    def getLinkState(self,ip):
        #运行ping程序
        p = subprocess.Popen(["ping.exe", '-n', '1', ip],
            stdin = subprocess.PIPE,
            stdout = subprocess.PIPE,
            stderr = subprocess.PIPE,
            shell = True)
        #得到ping的结果
        out = p.stdout.read().decode('gbk')
        # print(out.decode('gbk'))
        regex = re.compile(r'来自..+', re.M).findall(out)
        now_time = dt.datetime.now().strftime('%F %T')
        file_name = 'ping_ip_%s.txt' % nowdate
        with open('C:\limi_ppt_log\%s' % file_name, 'a') as f:
            f.write(str(now_time) + ' ' + str(regex) + '\n')

if __name__ == '__main__':
    ip = 'baidu.com' #要ping的主机
    while True:
        time.sleep(1)
        LinkState(ip)
```

调用os.popen

注：此代码生成exe无法运行.....原因待查找

```
#!/bin/python3
#####
# This program is to check the ping in the Live Room #
# Date: 2020-4-18                                     #
# Author: cuijianzhe                                  #
# Email: 598941324@qq.com                             #
#####
import os
import re
import datetime as dt
import time
nowdate = dt.datetime.now().strftime('%Y%m%d')
def baidu():
    ping = os.popen('ping www.baidu.com -n 1').read()
    ping = re.compile(r'来自..+', re.M).findall(ping)
    now_time = dt.datetime.now().strftime('%F %T')
```

```
file_name = 'ping_baidu_%s.txt' % nowdate
with open('C:\\limi_ppt_log\\%s'%file_name, 'a') as f:
    f.write(str(now_time) + ' ' + str(ping) + '\n')

if __name__ == "__main__":
    while True:
        time.sleep(1)
        baidu()
```

生成exe文件

D:\python_cuijianzhe\cuijianzhe\limi>pyinstaller -F -w test.py

-w: 关闭调试窗口

切记, 不可将该程序命名为ping.exe