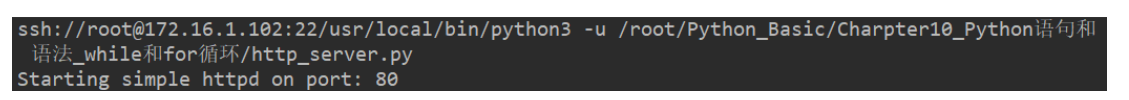
**Python基础 第八天作业2023.8.15**

1. **使用While循环监控TCP/80的HTTP服务运行状态**

一个简单HTTP服务器:

#!/usr/bin/python3.6  
# -\*- coding=utf-8 -\*-  
from http.server import HTTPServer, CGIHTTPRequestHandler  
port = 80  
httpd = HTTPServer(('',port), CGIHTTPRequestHandler)  
print('Starting simple httpd on port: ' + str(httpd.server\_port))  
httpd.serve\_forever()

运行HTTP服务器(Linux): 

使用netstat –tulnp查看开放端口号(Linux):

电脑萤幕画面

描述已自动生成

写一个Python脚本，使用While循环，每一秒查看一下TCP/80(一定要区分TCP或UDP,80或8000)是否被打开，如果打开就退出循环，并且打印告警信息，执行效果如下。

文本

描述已自动生成

import socket  
import time  
  
host = '192.168.28.130'  
port = 80  
  
while True:  
 try:  
 sock = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)  
 sock.settimeout(1) # 设置连接超时时间为1秒  
 sock.connect((host, port))  
 sock.close()  
 print(f"HTTP（TCP/80）服务已经被打开")  
 break  
 except (socket.timeout, ConnectionRefusedError):  
 print(f"等待一秒重新开始监控！")  
 time.sleep(1)

文本

描述已自动生成

1. **找到两个清单中相同的内容**

两个清单如下:

list1 = ['aaa', 111, (4, 5), 2.01]  
list2 = ['bbb', 333, 111, 3.14, (4, 5)]

运行结果如下:

文本

描述已自动生成

list1 = ['aaa', 111, (4, 5), 2.01]  
list2 = ['bbb', 333, 111, 3.14, (4, 5)]  
  
def check\_elements(list1, list2, element):  
 if element in list1 and element not in list2:  
 return str(element) + " only in List1"  
 elif element in list2 and element not in list1:  
 return str(element) + " only in List2"  
 elif element in list1 and element in list2:  
 return str(element) + " in List1 and List2"  
 else:  
 return str(element) + " doesn't belong to either list"  
  
elements\_to\_check = list1  
for element in elements\_to\_check:  
 result = check\_elements(list1, list2, element)  
 print(result)

文本

描述已自动生成

1. **创建一个简单的ping函数**

提示:如何使用kamene模块(之前的名字为Scapy)

安装:  
pip3 install -i https://pypi.douban.com/simple/ kamene  
交互界面测试:  
>>> from kamene.all import \*  
>>> import logging  
>>> logging.getLogger("kamene.runtime").setLevel(logging.ERROR) # 关闭不必要的报错  
>>> ping\_pkt = IP(dst='196.21.5.254')/ICMP() # 制造一个Ping包  
>>> ping\_result = sr1(ping\_pkt, timeout=2, verbose=False) # Ping并且把返回结果复制给ping\_result  
>>> ping\_result.show() # 查看回显结果

代码提示:

图形用户界面, 网站

描述已自动生成

import logging  
logging.getLogger("kamene.runtime").setLevel(logging.ERROR)  
from kamene.all import \*  
  
def qytang\_ping(ip):  
 ping\_pkt = IP(dst=ip) / ICMP()  
 reply = sr1(ping\_pkt, timeout=2, verbose=False)  
  
 if reply:  
 return f'通'  
 else:  
 return f'不通'  
  
host = '192.168.28.2'  
ping\_result = qytang\_ping(host)  
print(host + ' ' + ping\_result + '!')

文本

描述已自动生成