**CHAPTER- 3.2: Packet Sniffer**

**3.2.1) DEFINITION:**

Packet sniffers are applications or utilities that read data packets traversing the network within the Transmission Control Protocol/Internet Protocol (TCP/IP) layer. When in the hands of network administrators, these tools “sniff” internet traffic in real-time, monitoring the data, which can then be interpreted to evaluate and diagnose performance problems within servers, networks, hubs and applications.

When packet sniffing is used by hackers to conduct unauthorized monitoring of internet activity, network administrators can use one of several methods for detecting sniffers on the network. Armed with this early warning, they can take steps to protect data from illicit sniffers.

## **Ways To protect networks from illicit sniffers:**

* Do not use public Wi-Fi networks
* Rely on a trusted VPN connection
* Always deploy robust antivirus software
* Look for secure HTTPS protocols before surfing the web
* Don’t fall prey to social engineering tricks and traps

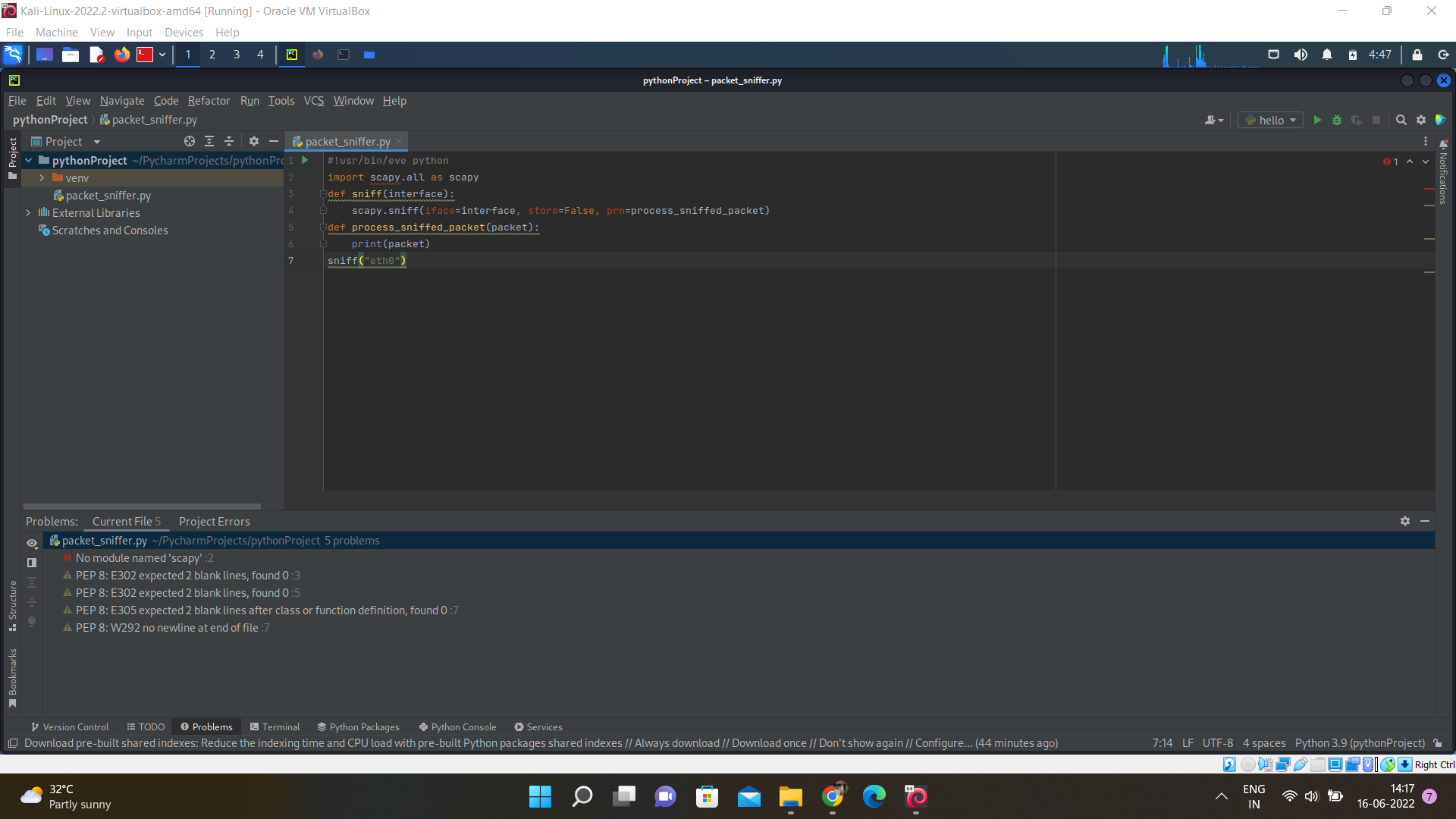
**3.2.2) TOOLS REQURIED**

* Kali Linux
* PyCharm
* Python3

### **3.2.3) PROCEDURE:**

### **Step-1: Sniffing packets using Scapy:**

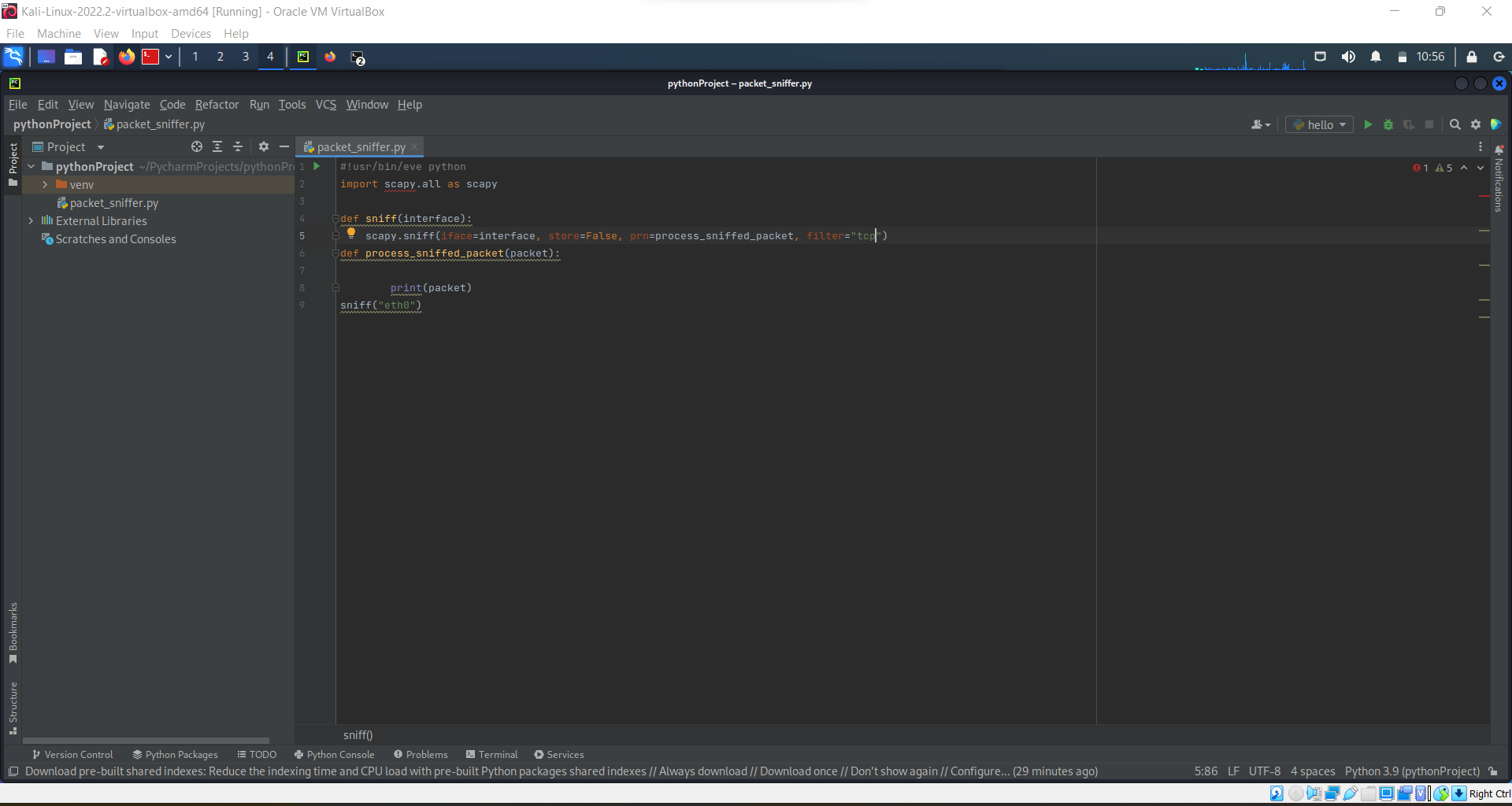
To sniff the packets, use the sniff () function. The sniff () function returns information about all the packets that has been sniffed. It can capture data sent to/ from interface.



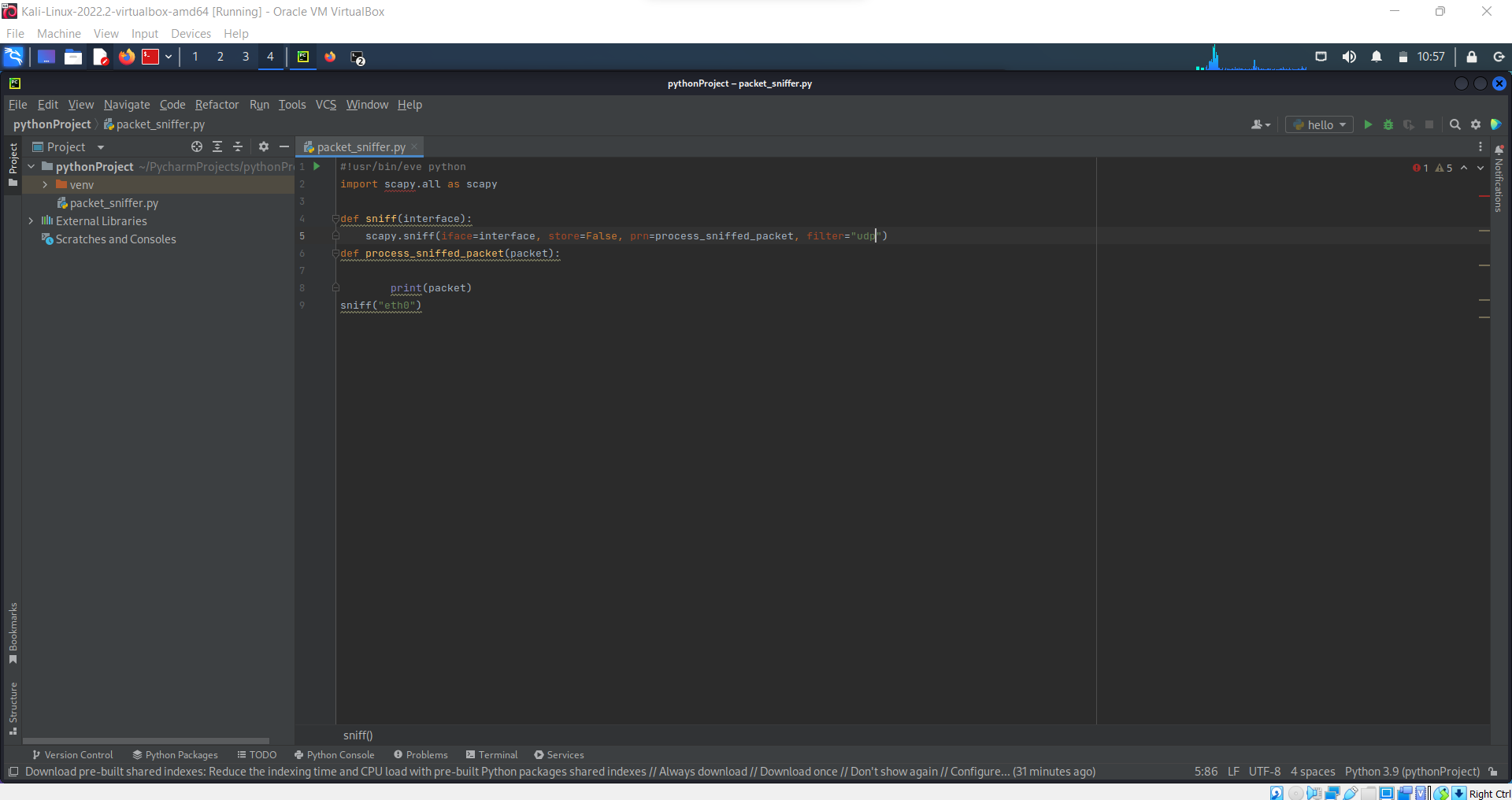
**Step-2:** **Extracting data From a Specific Layer:**

There are different layers like Tcp, Udp etc. We use filter=” layer name ” to extract data.

**UDP:**

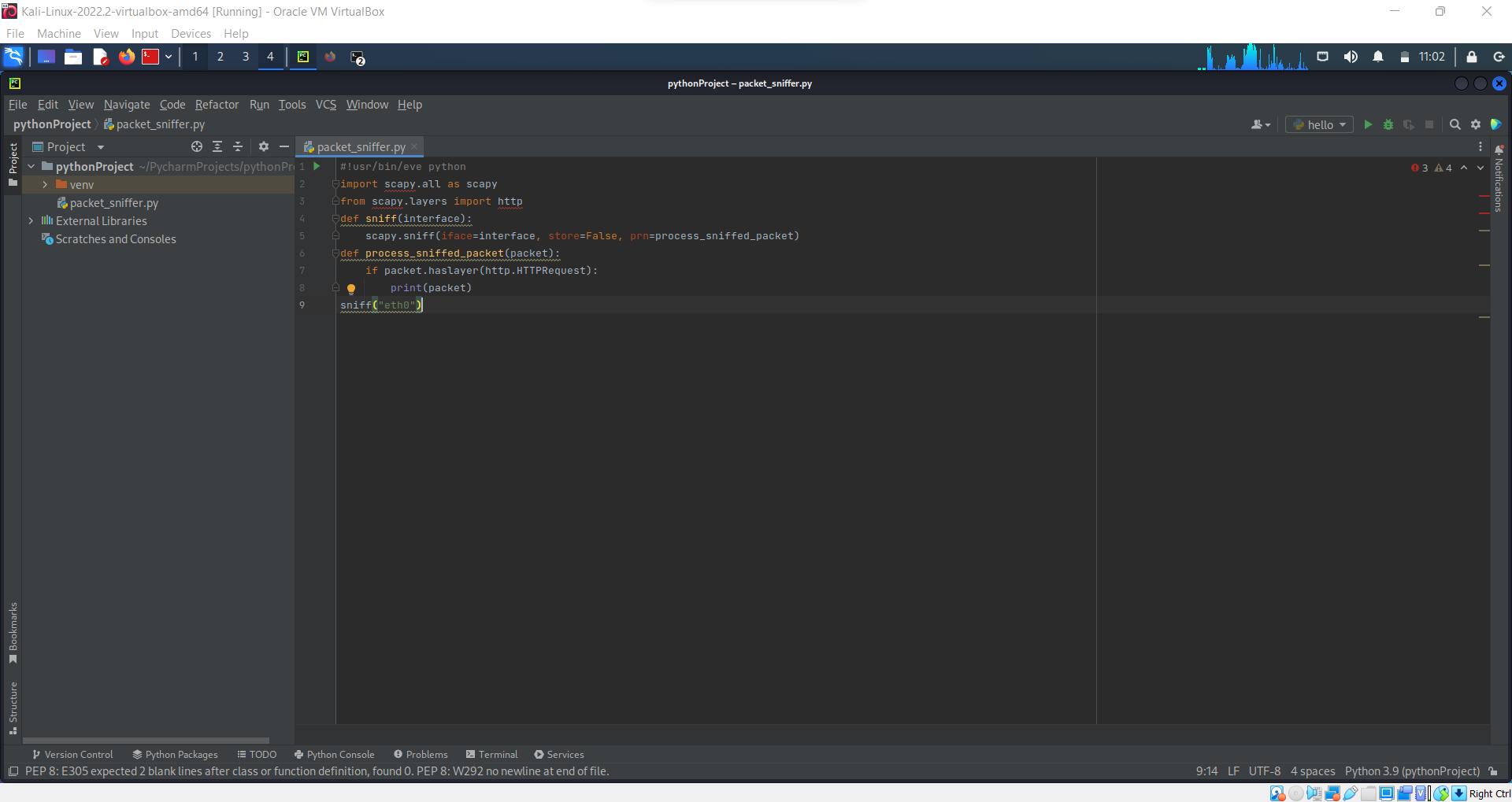


**TCP:**

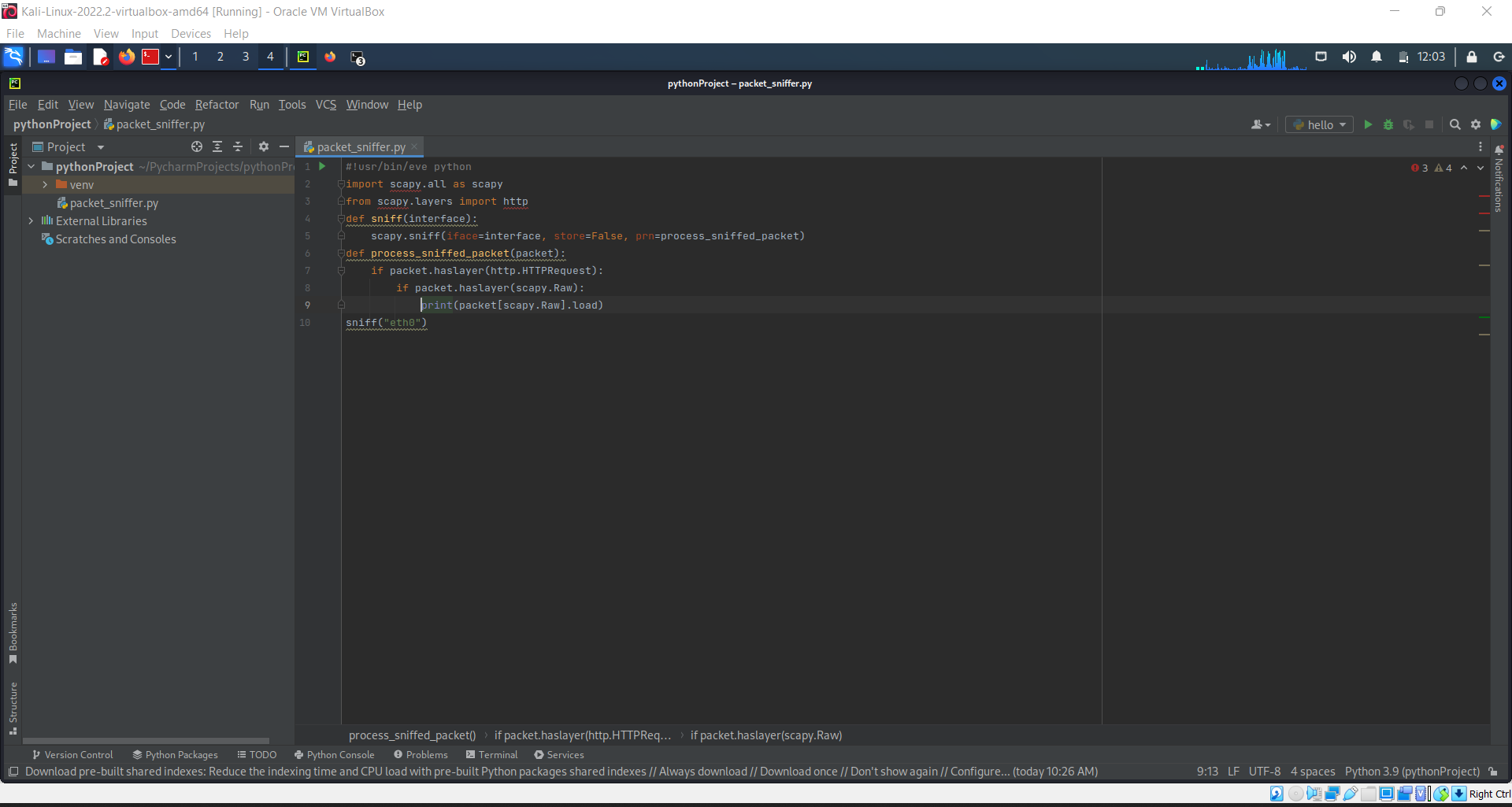


**HTTP:**

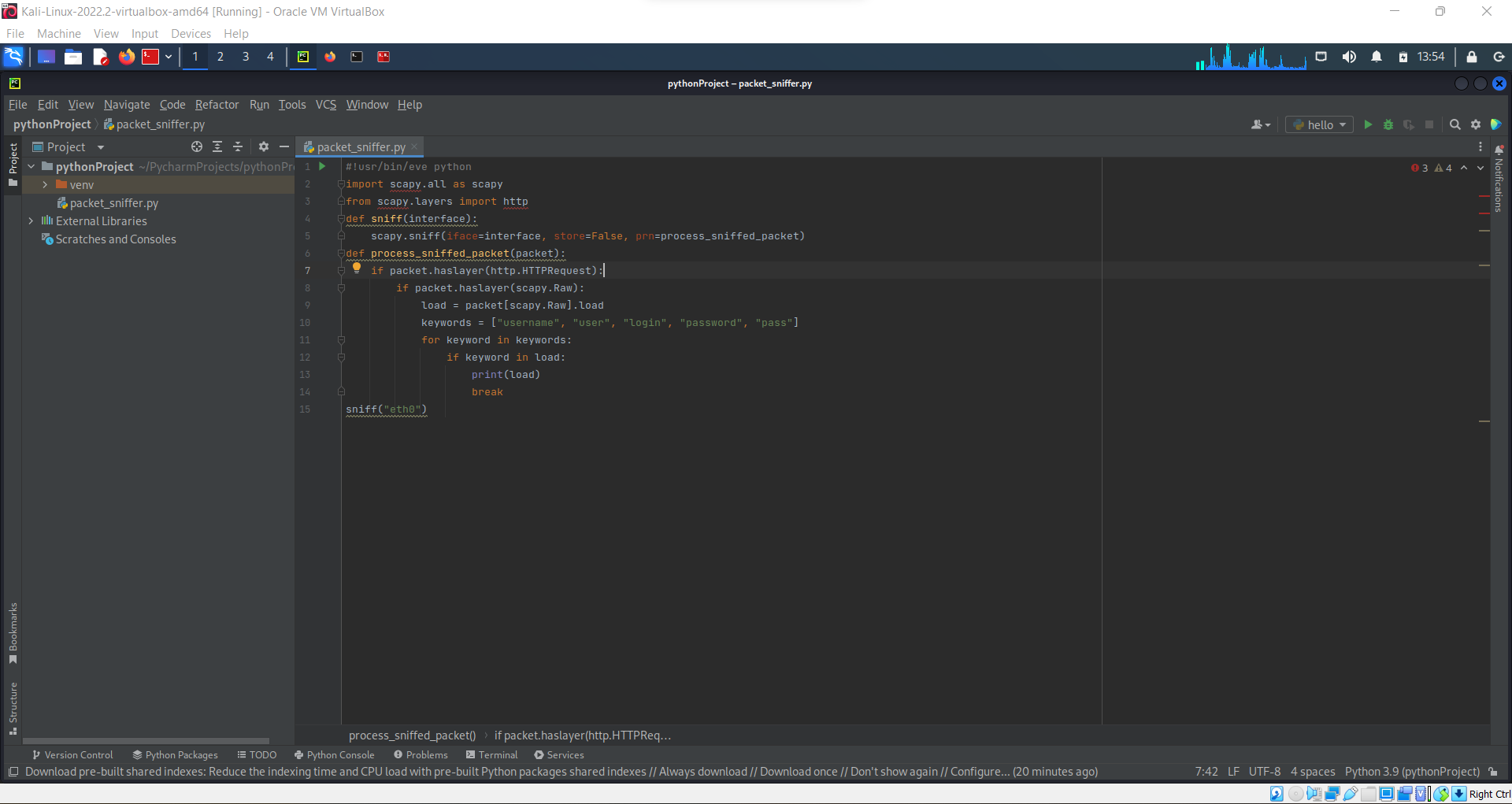
Scapy does not contain Http as default in its layer. So, we import http to scapy layer.



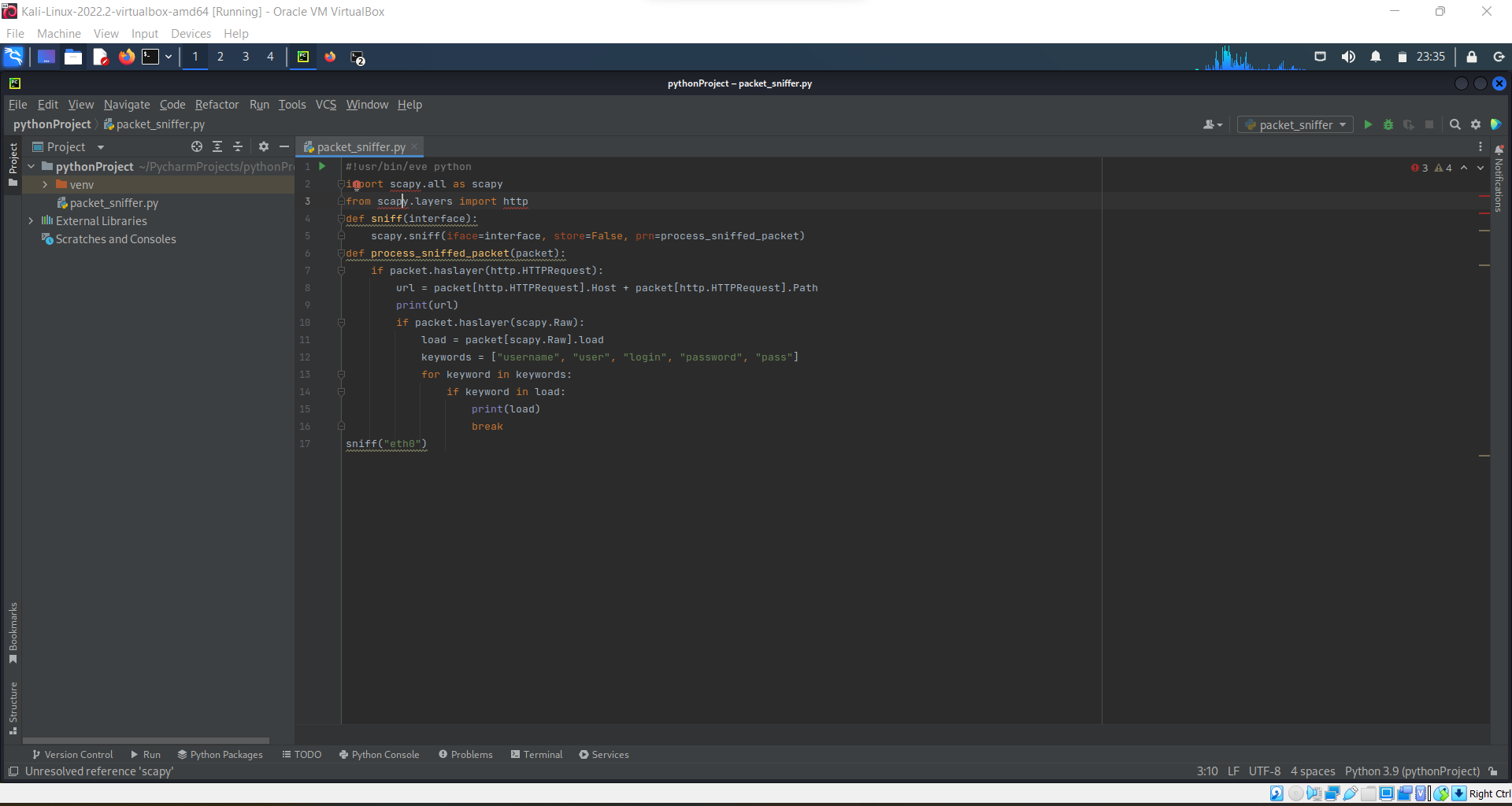
**Step-3:** **Analyzing Sniffed Packets & Extracting Fields from Layers:**



**Step-4: Analyzing Fields and Extracting Passwords:**

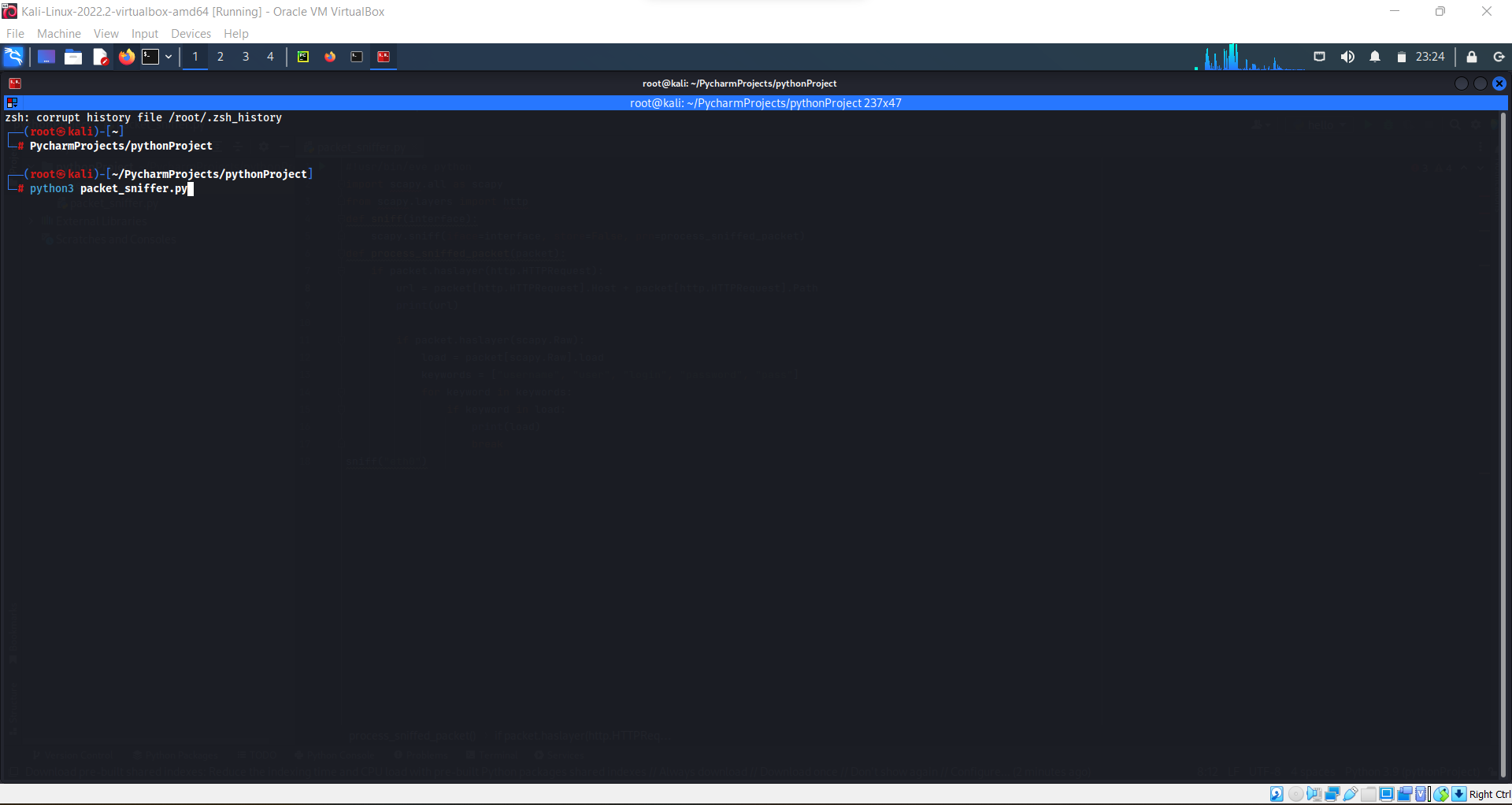


**Step-5: Extracting URL:**



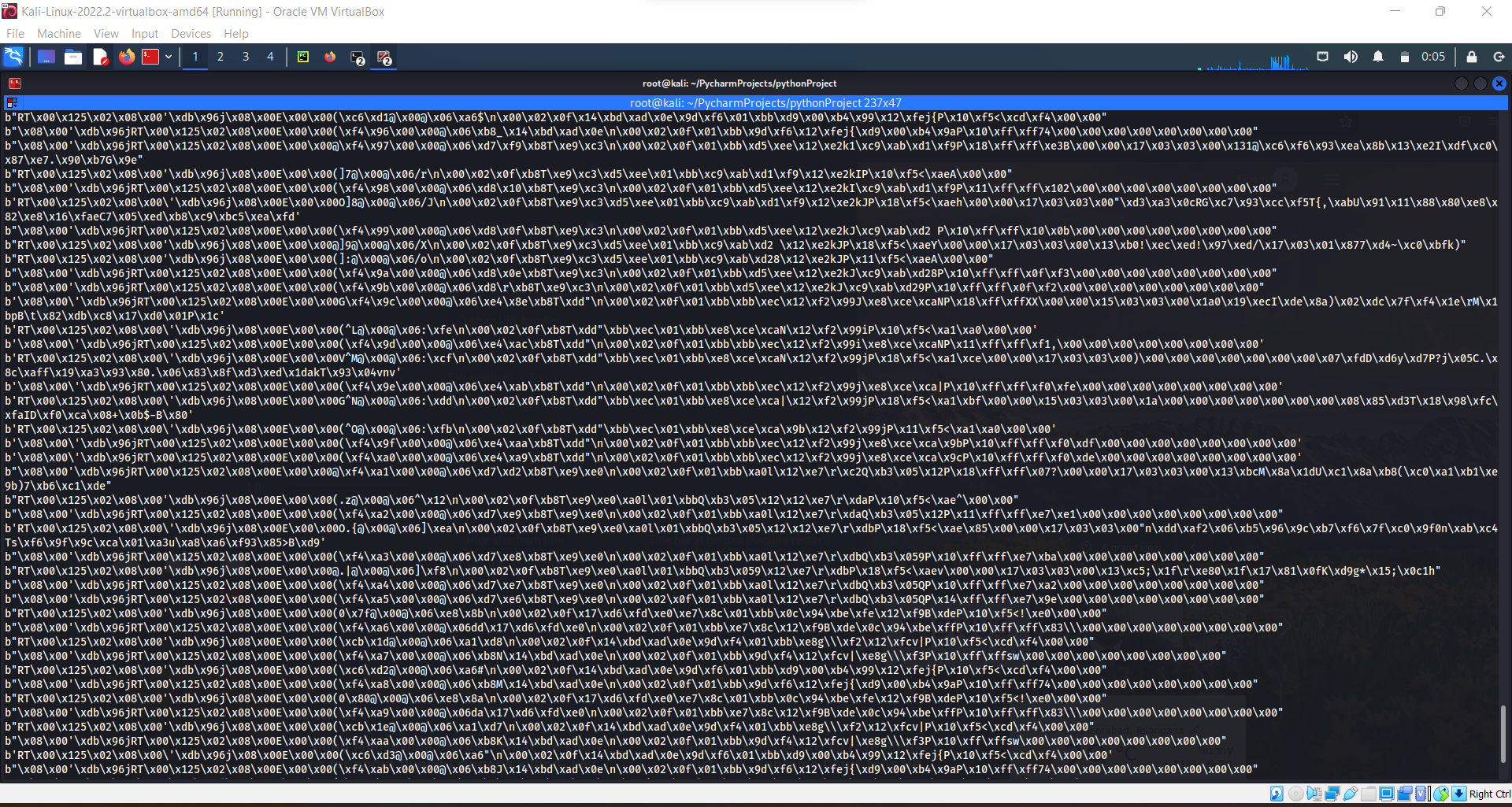
**3.2.4) Execution:**

**Command:** > python packet\_sniffer.py

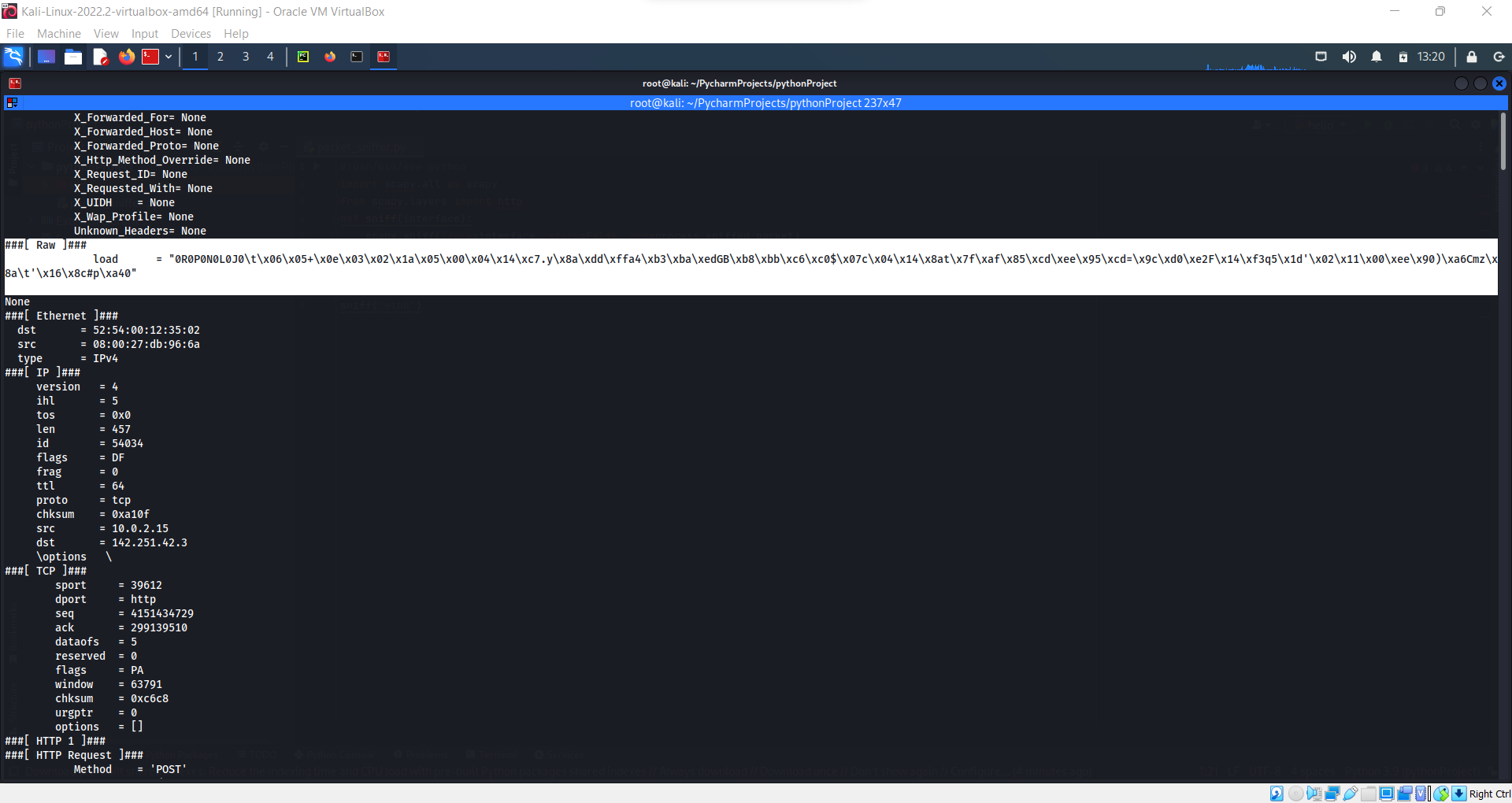


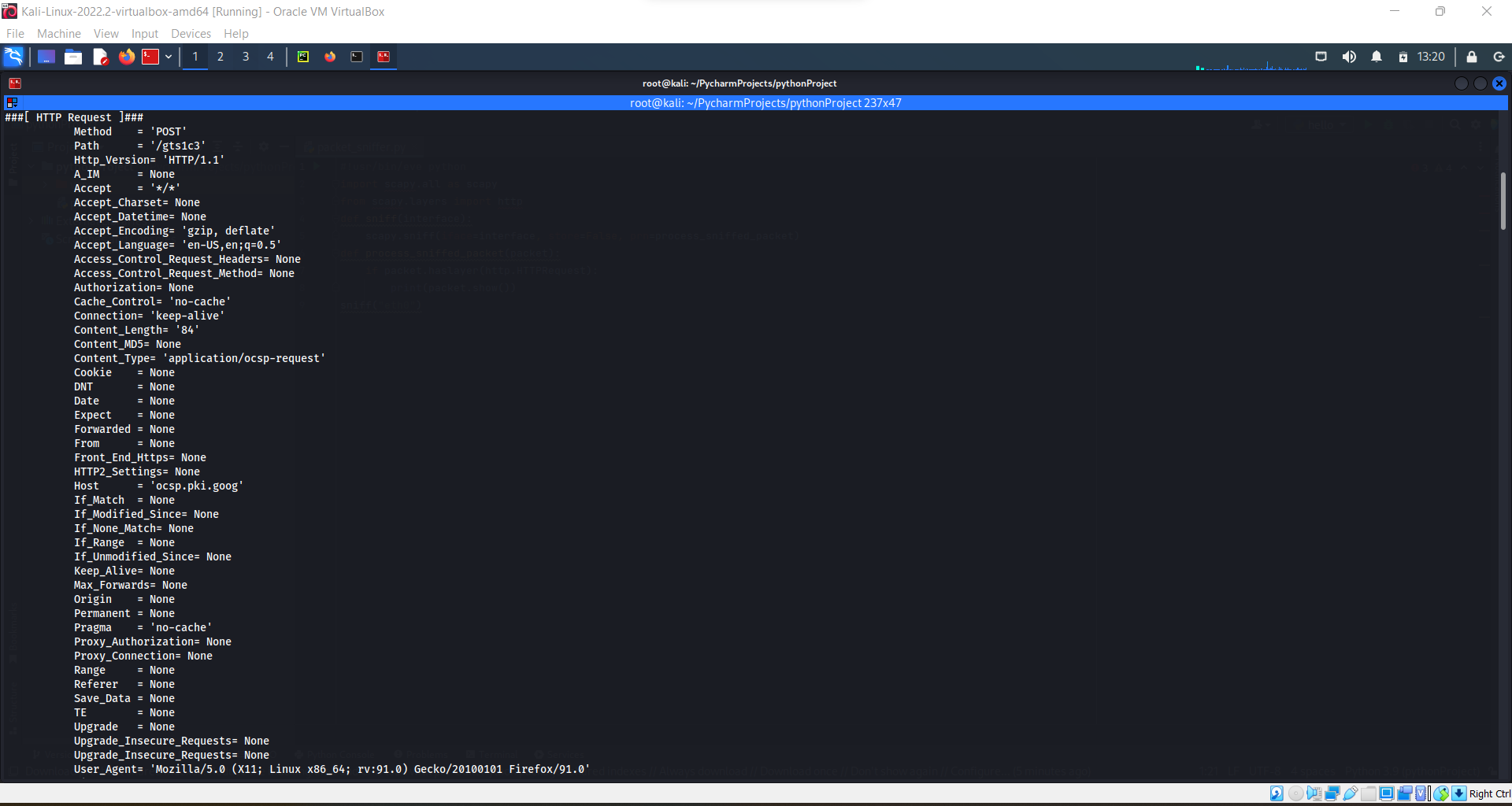
**3.2.5) Output:**

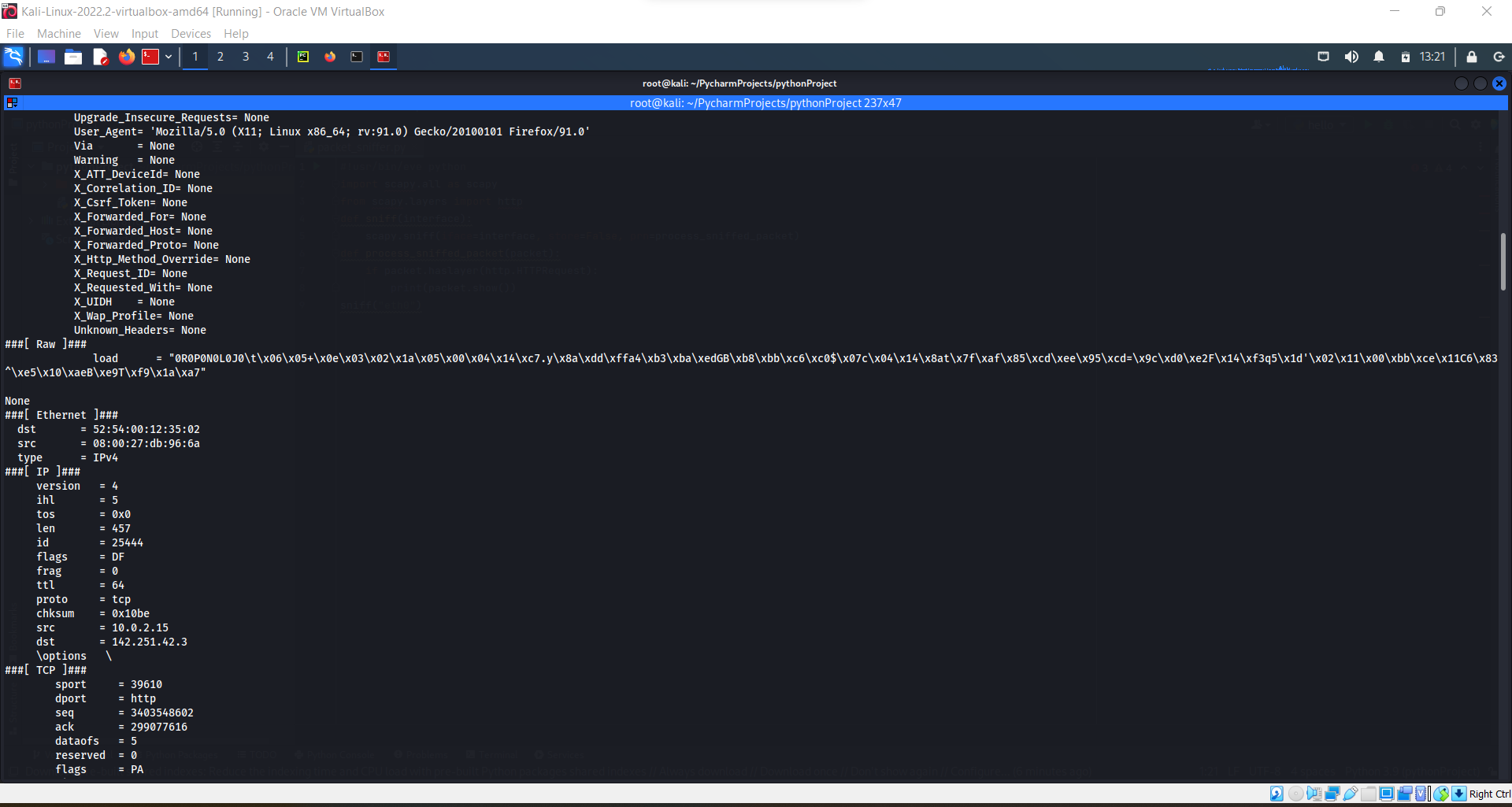
**Step1:**

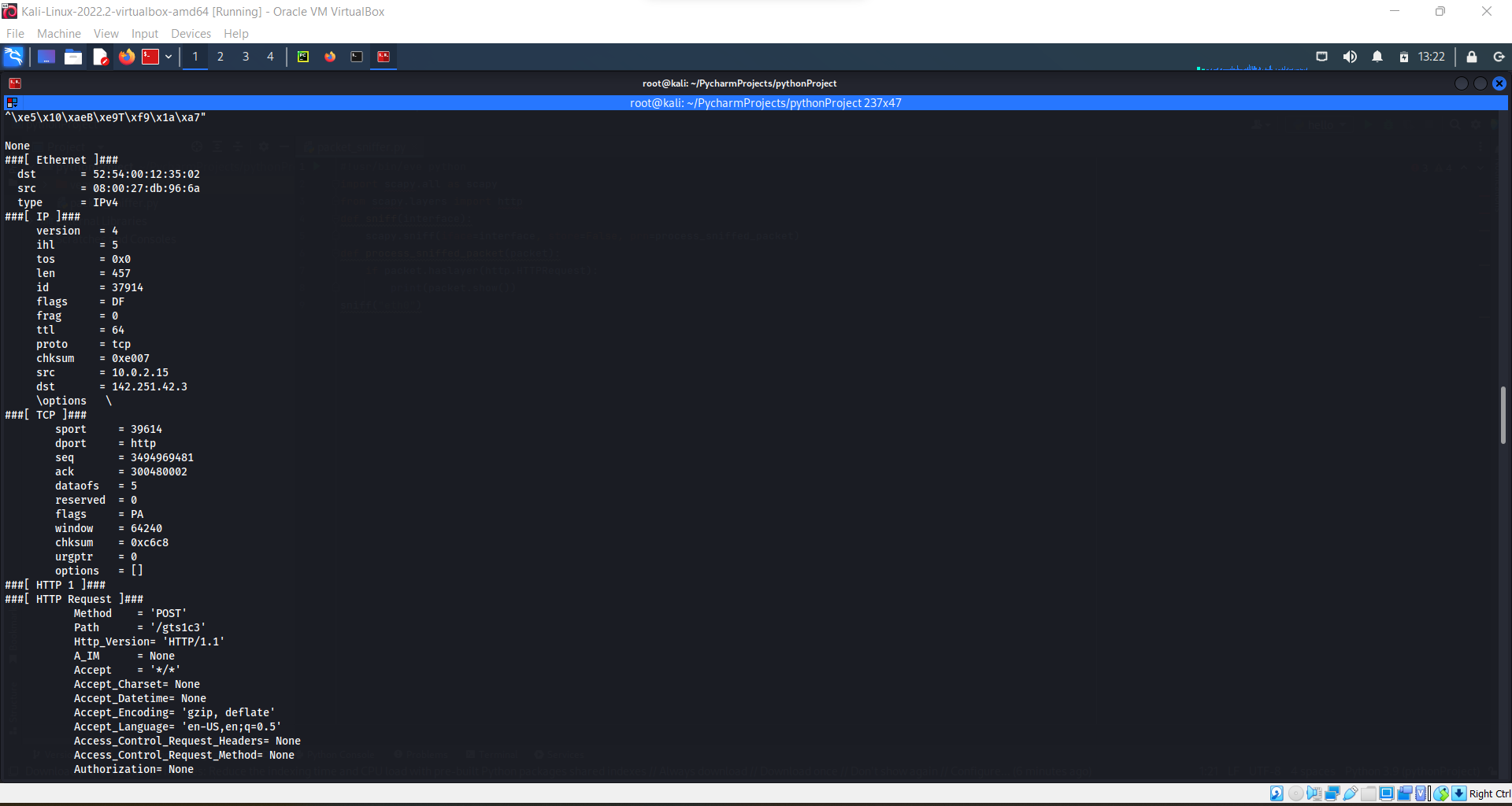


**Step 2:**

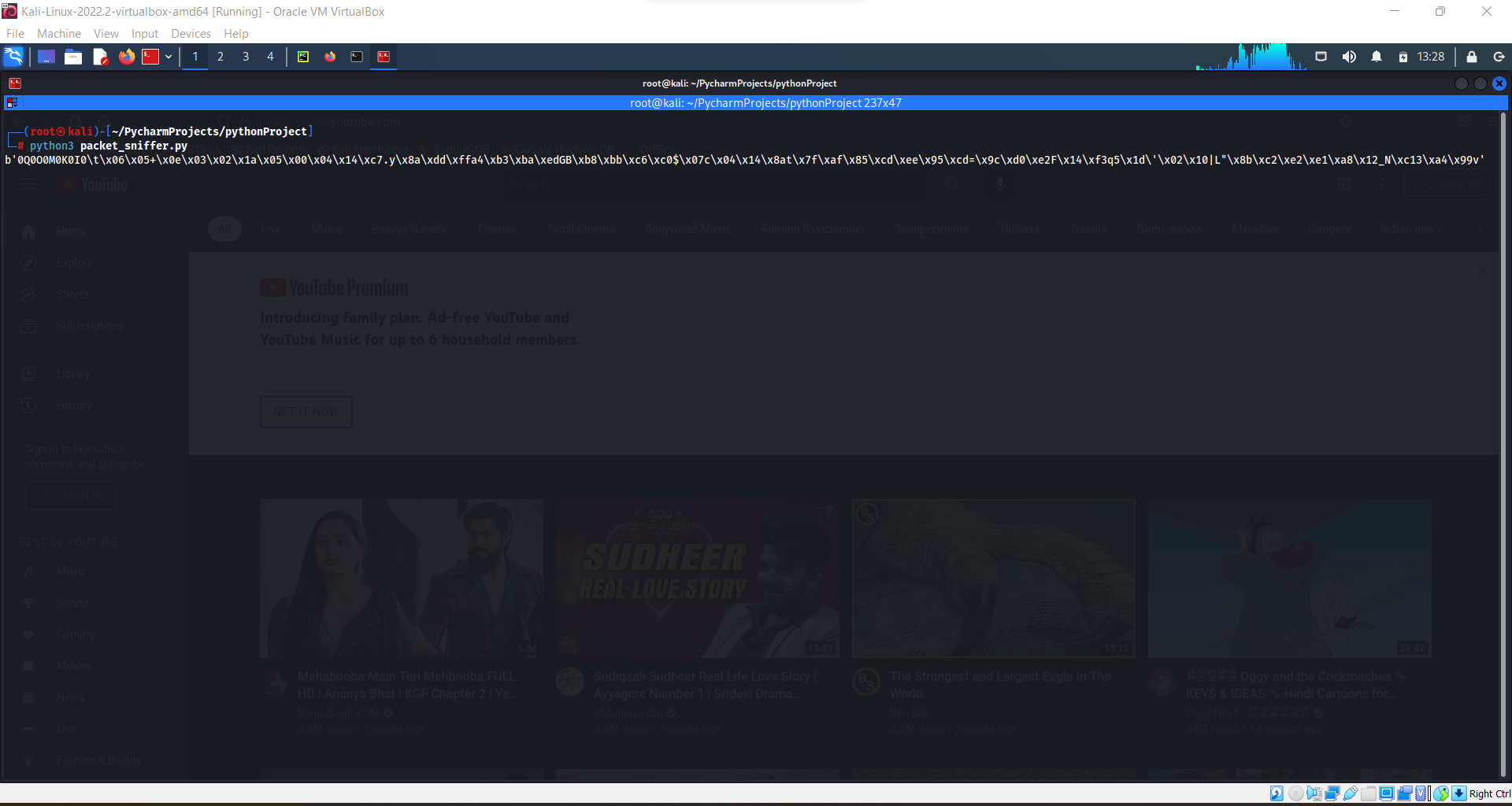






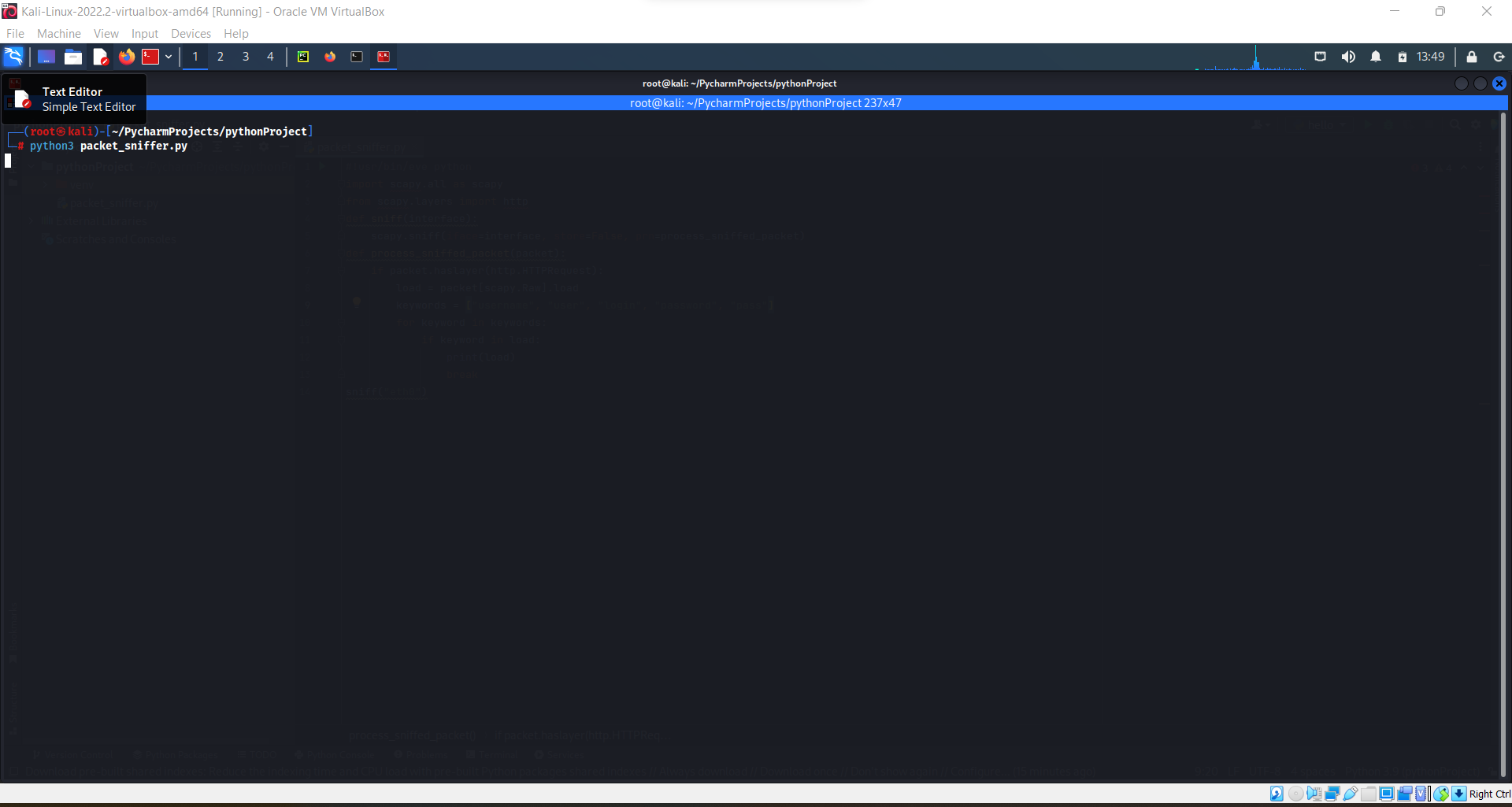


**Step 3:**



**Step- 4:**

There is no keywords in my raw so output is empty



**Step-5:**

