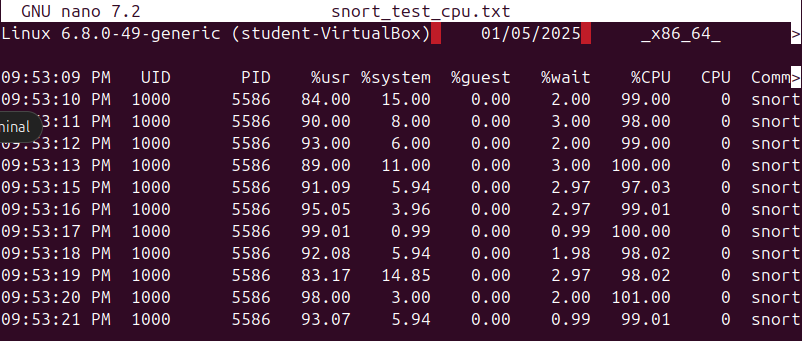
**Link to the large pcap file used**

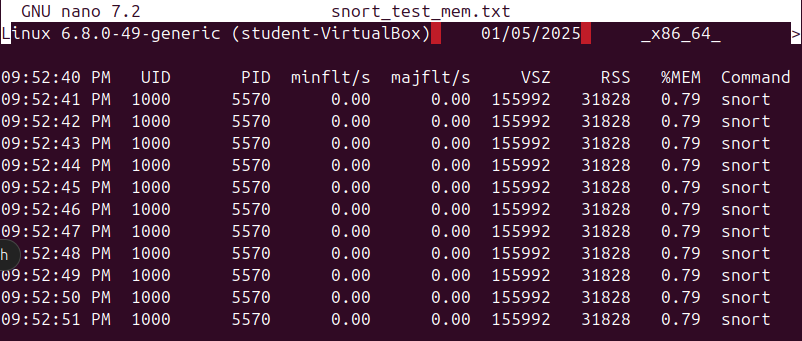
We found a zip file that contains many pcap files.

URL: <https://share.netresec.com/s/nF5zNcaXLgwdQFZ>

Then we chose randomly some of them in order to reach the 1Gb size limit and using mergecap command we merge them together and create a single pcap file.

URL: <https://drive.google.com/file/d/1EqLRYuWNOfxIgsxUKWmXfC1jcoTHMnVa/view?usp=drive_link>

**Performance metrics**

****

**Command: pidstat (-r) -p $(pgrep snort) 1 > file\_to\_be\_saved,** (-r) used for memory

As we can see it is quite CPU intensive and not so memory, because of the few rules that the default configuration has.

**Suggest optimization actions to improve snort’s performance.**

* Snort is a signature based IDS and as a result its performance depends on the number and the complexity of the rules being loaded. In this case we use default snort rules. An action to improve performance is to make custom rules focused on our use case.
* Snort uses preprocessors, modules that analyze and manipulate network traffic. Another action could be to enable only preprocessors for the specific protocols and services i need to inspect.
* Snort 3 also supports multi-threading. By enabling it, splits the incoming traffic into multiple threads, with each thread processing a portion of the traffic. Thia has a result better cpu utilization.
* If in our use case we need to examine only a part of packets, an option for optimizing performance could be to reduce the snap length to only capture the necessary portion of packets.