CPU Usage Visualization Analysis

Vincent
COSC3P99 - Project UI
Brock University
St. Catharines, Ontario
vzhang2980@hotmail.com

***Abstract*—Perform Data Visualization techniques on lttng trace events to determine CPU time distribution between packets.**
***Index Terms*—LTTng, Ubuntu, kernel, packet, trace**

I. Introduction

In this analysis, LTTng traces were used to gain insights of the Linux kernel scheduling behaviour. The primary focus was on the ”sched_switch” event which is a kernel event that occurs when the scheduler switches execution between tasks. This event provides valuable data into how the CPU time is distributed between tasks. The context.packet_seq_num field is one of the fields associated with the ”sched_switch” event. It represents the unique identifier for the packet associated with the event in the trace.

II. Background

After successfully installing LTTng (Linux Trace Toolkit Next Generation) on Ubuntu, a trace was generated using the ”https://lttng.org” website. This platform was selected due to the fact that it provides a convenient and standardized platform for using LTTng.

A line graph was selected as the display since they are particularly useful for the visualization of trends and relationships over an interval. It is easier to see the patterns of values with a line graph where the connections between points help in following the sequence of the data.

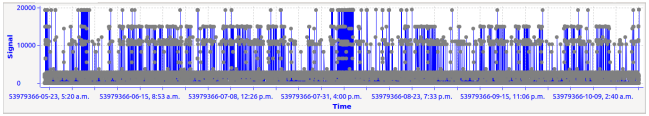

Fig. 1. lttng trace sched_switch, prev_tid

In the sched_switch event, there are multiple fields which include the previous, and next command, task id, priority, and state. These fields when visualizing large data results in frequent jumps between data values. This makes it difficult to accurately determine issues in the provided trace.

The context.packet_seq_num field was selected to be visualized as it can be used to correlate events that occur within the same packet. This field provides a method to correlate context switch events within a trace which in turn aids the analysis of the scheduling behaviour and system activity.

Through the visualization of the context.packet_seq_num field, the user is able to determine which packet is running at a time. This field allows for the grouping of related events which is useful for analyzing multiple events that occurred. It can also be used to identify gaps and irregularities through the sequence of values. This information is crucial in assessing the completeness of the trace.

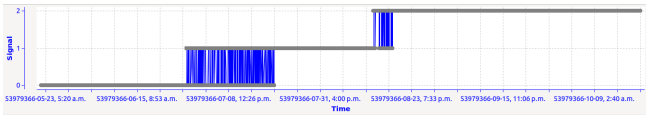

Fig. 2. lttng trace sched_switch, context.packet_seq_num

In the data sample, the values of the context.packet_seq_num are shown to repeatedly switch between values. This shows that the system involves multi-threading or parallel execution causing events from different threads or processes to be interleaved, resulting in non-sequential packet sequence numbers.

III. Conclusion

Through the selected visualization technique, the plugin user is able to quickly monitor the CPU distribution between packets to both identify irregularities and assess the completeness of the trace.