LTTng + OpenTelemetry benchmarks

All configurations

- LTTng configuration: No LTTng session running (NLS), LTTng session without recording (LSWR), LTTng session recording UST telemetry data (LSRU), LTTng remote session recording UST telemetry data (LRSRU)
- Type of instrumentation: No instrumentation (NI), Ciena-like Instrumentation (CI), OpenTelemetry (OTel)
- Type of exporter: LTTng Exporter (LE), Local OTel collector (LOC), Remote OTel collector (ROC)
- OTel Traces Processor (apply only for traces benchmarks): Simple (SP), Batching processor (BP)

With Ciena Instrumentation, we log directly a message to LTTng having a structure close to the one that can be built with OTel. For now, we log a simple string. A more realistic benchmark should use Ciena internal tracing library.

The remote collector is in the same network with the benchmarks host. Network packets need to go through 2 switches only.

Traces benchmarks

Pattern: Start a span, do minimal operations and end it right away. Here are All the following scenarios have been benchmarked:

1) simple: Start a span and end it right away. Measure the time to do the operation.
2) context extraction: Start a span, extract the span context that can be send to a remote process and end the span.
3) event: Start a span, add one span event and end the span.
4) attribute: Start a span and set a span attribute and end the span. Measure the time to do the operation.
5) span context: Start a span from a span context created by a remote process. End the span. Measure the time to do the operation.
6) nested span: Start a span. Start a nested span inside of that span. End both. Measure the time to create and end the inner span.

We record the time required to do mentioned operations multiple times. And then compute mean, median, average, std statistics. All results for all possibles configuration were reported in this excel file.

For comparison purposes, here are the results for the scenario 1 only:

Test cases	NLS-CI	LSWR-CI	LSRU-CI	LRSRU-CI
n spans	20,000	20,000	20,000	20,000

min (ns)	112	138	60	134
mean (ns)	311	403	393	338
max (ns)	12,250	11,493	21,935	18,509
median (ns)	283	418	391	308
std (ns)	146	161	312	187
real (ms)	202,551	202,589	202,666	202,672
user (ms)	128	113	131	145
sys (ms)	456	482	481	470

Test cases	NLS-OTel-LOC-SP	NLS-OTel-LOC-BP	NLS-OTel-ROC-SP	NLS-OTel-ROC-BP	NLS-OTel-LE-SP	NLS-OTel-LE-BP	LSWR-OTel-LE-SP	LSWR-OTel-LE-BP	LSRU-OTel-LE-SP	LRSRU-OTel-LE-SP	LSRU-OTel-LE-BP	LRSRU-OTel-LE-BP
n spans	5,000	20,000	5,000	20,000	20,000	20,000	20,000	20,000	20,000	20,000	20,000	20,000
min (ns)	1,014,362	19,074	1,931,562	21,101	97,795	12,697	105,963	23,366	94,947	61,689	23,063	43,641
mean (ns)	1,975,102	115,143	2,945,936	116,657	291,046	115,924	285,806	116,900	288,689	287,596	117,143	116,836
max (ns)	16,427,715	647,056	15,251,232	455,129	909,617	406,575	950,945	783,607	957,472	1,512,586	536,921	396,297
median (ns)	2,168,330	114,738	2,796,951	117,134	308,049	117,818	291,683	118,180	305,975	283,274	113,691	131,189
std (ns)	376,562	12,709	478,621	9,668	22,677	9,219	21,489	10,328	22,681	23,003	9,394	9,482
real (ms)	60,475	204,822	65,391	204,911	208,426	204,947	208,354	204,978	208,483	208,473	205,077	205,048
user (ms)	7,192	3,513	8,079	3,663	6,033	3,208	6,079	3,245	6,029	5,969	3,259	3,268
sys (ms)	458	358	369	330	440	411	300	379	407	461	405	379

"n spans" is number of spans used in the experiment.
"real", "user" and "sys" statistics are total time spent respectively in script, user space (CPU time) and kernel (CPU time).

- → NLS-OTel-ROC-SP vs LSRU-OTel-LE-SP vs LRSRU-OTel-LE-SP (Exporting spans one by one as they are created using remote OpenTelemetry collector vs using Local Lttng exporter vs Exporting one by one to remote LTTng)
  - When using simple processor, spans are processed synchronously after they are created. In this situation, using LTTng to log spans should be preferred over sending traces overs the network.
- → NLS-OTel-ROC-BP vs LSRU-OTel-LE-BP vs LRSRU-OTEL-LE-BP (Same comparison but we export traces every 5s in batch in a background thread)
  - In all scenarios, the export is handled by a thread in the background. We made sure that the spans production rate is smaller than the spans synchronous export rate. The mean time to create and end a span is similar in all scenarios. Using LTTng in local or remote setup use slightly less CPU time.

- In production, the remote collector could be in a different network, which could make these results slightly vary.
- The preferred solution should be logging all traces locally to LTTng. This avoids running an OTel collector and dealing with all the network communications troubles it could add.

Metrics benchmarks

Pattern: We measure the time to do an operation without collecting any kind of metrics. And we repeat the same operation while exporting metrics every 500/1000 ms. To simulate a Ciena instrumentation, we start a thread which log a value every 500/1000 ms to LTTng. Here are all benchmarks' results:

Scenarios	NI		NLS-CI		LSWR-CI		LSRU-CI		LRSRU-CI	
Export delay (ms)	500	1000	500	1000	500	1000	500	1000	500	1000
duration (ms)	114,541	114,539	114,509	114,495	114,480	114,436	114,481	114,520	114,545	114,466
overhead (%)			-0.028	-0.026	-0.053	-0.051	-0.052	-0.05	0.004	0.006
cpu time (ms)	114,537	114,535	114,512	114,496	114,480	114,436	114,485	114,522	114,544	114,472
cpu time overhead (%)			-0.022	-0.034	-0.05	-0.086	-0.046	-0.012	0.006	-0.055

Scenarios	NLS-OTel-LOC		NLS-OTel-ROC		NLS-OTel-LE		LSWR-OTel-LE		LSRU-OTel-LE		LRSRU-OTel-LE	
Export delay (ms)	500	1000	500	1000	500	1000	500	1000	500	1000	500	1000
duration (ms)	115,264	114,960	115,290	115,030	114,681	114,584	114,663	114,588	114,712	114,681	114,649	114,572
overhead (%)	0.631	0.633	0.654	0.656	0.122	0.125	0.106	0.109	0.149	0.151	0.094	0.096
cpu time (ms)	115,806	115,287	115,816	115,348	114,800	114,651	114,786	114,653	114,836	114,749	114,776	114,650
cpu time overhead (%)	1.107	0.657	1.116	0.71	0.229	0.101	0.217	0.103	0.261	0.187	0.208	0.1

The CPU time is the total CPU time spent in user space.

For all configurations, the execution time overhead is less than 1.2% and the more the export interval is larger, the more the overhead is lower.

LTTng Metrics exporter is approximatively 50% faster than the remote exporter but the CPU time spent in user space is similar for the two configurations.

Logs benchmarks

In every configuration, we logged a 5-byte string 5000 times, complete with severity and service resource definitions.

Configuration	min (ns)	mean (ns)	max (ns)	median (ns)	std (ns)	real (ms)	user (ms)	sys (ms)
NLS-CI	79	338	19,982	358	354	50,579	55	85
LSWR-CI	91	286	15,084	316	215	50,623	49	98
LSRU-CI	60	311	914	271	57	50,624	16	132
LRSRU-CI	14	302	31,328	381	566	50,632	23	127
NLS-OTel-LOC	162,597	1,942,427	14,075,368	1,942,680	366,043	60,257	7,179	481
NLS-OTel-ROC	109,952	1,933,579	12,132,325	1,856,451	341,200	60,216	7,054	581
NLS-OTel-LE	2,454	224,581	727,890	227,603	22,160	51,732	1,191	99
LSWR-OTel-LE	99,687	225,346	655,659	247,345	25,447	51,757	1,209	87
LSRU-OTel-LE	9,975	232,732	713,687	292,836	23,224	51,791	1,255	84
LRSRU-OTel-LE	90,243	230,088	716,526	242,816	27,159	51,780	1,234	89

The process of exporting a log to an OTel collector via OTLP takes roughly 1900 microseconds, compared to approximately 230 microseconds when exporting them to LTTng. This represents an improvement factor of at least 8 times when using LTTng over OTLP.

Source code

https://github.com/dorsal-lab/opentelemetry-c-performance

https://github.com/dorsal-lab/opentelemetry-c