Тестовое задание

Написать программу, использующую eBPF для сбора информации об исполняющихся процессах, сериализует их и сохраняет их в файл в формате JSON по порядку исполнения процессов.

Пример:

По очереди исполнялись процессы: bash, ls, ncat

Тогда файл должен содержать примерно следующее:

JSON

1. { pid: 123, cmdline: "/usr/bin/bash ...", timetamp: <posix ts> }
2. { pid: 124, cmdline: "/usr/bin/ls ...", timetamp: <posix ts> + N }
3. { pid: 125, cmdline: "/usr/bin/ncat ...", timetamp: <posix ts> + N1 }

Требования к программе:

- ЯП: C++23, C
- Система сборки: CMake
- Обязательные библиотеки: libbfp, nlohmann-json
- Менеджер библиотек: vcpkg
- Канал передачи данных: perf buffer
- Программа должна обеспечивать гарантию последовательности событий (perf buffer ее не дает)
- Программа должна быть слинкована статически полностью и не содержать никаких динамических зависимостей, в том числе libc++ и libc
- eBPF код должен отсеивать kthreads

Список собираемых данных:

- command line
- comm
- environment
- pid, tgis, pgid, ppid
- comm
- timestamp (в миллисекундах)
- uid, gid

Требования по сдаче:

- Код должен содержать пояснения к неочевидным моментам

- Время на выполнение: 3-4 рабочих дня
- Если не успеваешь доделать до конца, остаются недофикшеные баги то можно скинуть не полностью, оставив пояснения по необходимым доработкам и их содержанию.