

OSPP2023结项报告

赵昊

项目基本信息

- 名称：libevhttp使用libhv替代libevent
- 代码：23b970534

项目背景

- **描述**：libhv是一个类似于libevent、libev、libuv的跨平台网络库，提供了更易用的接口和更丰富的协议。libevent是一个轻量级的高性能网络库，旨在替换事件驱动网络服务器中的事件循环，提供了三类事件的统一反应机制，支持在发生IO事件、信号事件、计时器超时事件后执行回调函数。
- **目标**：将libevhttp中使用到libevent中至少80%接口替代为使用libhv的接口，并且保持原有软件功能正常。

方案描述

- **总体思路**：使用libhv实现libevent的API，从而在无需修改libevhttp源码的情况下将底层的libevent替换为libhv。
- **具体做法**：
 - 新增一个hv2ev.h文件。
 - 在该文件中，基于libhv的API实现libevent的API，例如基于hloop_new实现event_base_new。

```
HV_INLINE struct event_base *event_base_new(void) {  
    struct event_base *base = NULL;  
    HV_ALLOC(base, sizeof(struct event_base));  
    base->loop = hloop_new(HLOOP_FLAG_QUIT_WHEN_NO_ACTIVE_EVENTS);  
    base->timer = NULL;  
    return base;  
}
```

- hv2ev.h可以直接作为libevhttp的头文件之一，也可以尝试合入libhv的仓库。
- libevhttp只需include合适的头文件、链接到libhv库即可，最终目标如下图所示。

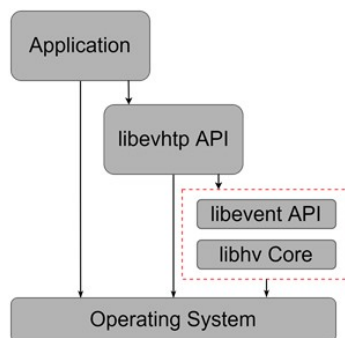


Figure 1: 项目成果逻辑示意图

- **注意事项**：
 - 以libevent-2.1.12-stable的API为标准。

- libevhttp自行设计了线程池模型，没有采用libevent的多线程接口和锁接口，因此基于libhv实现libevent API时，无需考虑接口的线程安全问题。
- openEuler操作系统的内核是Linux，因此当遇到无法解决的跨平台问题时，以Linux为准。
- libevent某些接口的功能复杂、参数众多，本项目优先实现libevhttp所涉及的部分。

进度概述

- 已完成方案设计。
- 已完成47(/49)个接口的实现和替换。
- 对已实现的接口进行了单元测试，13/13个集成测试程序可正常运行。

代码完成情况

下方是所有需要替换的接口及完成情况

待替换接口	接口描述	完成情况
event_base_new()	创建并返回一个event_base结构体	已完成
event_base_free()	销毁event_base结构体	已完成
event_base_loop()	开启事件循环	已完成
event_base_loopbreak()	退出事件循环	已完成
event_base_loopexit()	在超时退出事件循环，若没有指定超时时间，即刻退出循环	已完成
event_new()	创建并返回一个event结构体	已完成
event_free()	销毁event结构体	已完成
event_add()	添加event到event_base中，并开启监听相应事件	已完成
event_del()	取消监听某个事件	已完成
event_active()	激活event	已完成
evconnlistener_new()	监听指定文件描述符的tcp连接，创建并返回一个evconnlistener	已完成
evutil_make_socket_nonblocking()	设置某文件描述符为non-blocking模式	已完成
evutil_make_socket_closeonexec()	设置某文件描述符为close-on-exe模式	已完成
evutil_closesocket()	关闭套接字	已完成

evutil_socketpair()	创建一对关联套接字	已完成
evutil_inet_pton()	解析一个ip地址	已完成
evutil_inet_ntop()	网络地址转换为可读的字符串表示形式	已完成
evbuffer_new()	创建并返回一个evbuffer结构体	已完成
evbuffer_free()	销毁evbuffer结构体	已完成
evbuffer_get_length()	返回evbuffer中存储的总字节数	已完成
evbuffer_add()	将数据附加到evbuffer的末尾	已完成
evbuffer_add_printf()	将格式化的数据附加到evbuffer的末尾	已完成
evbuffer_add_buffer()	将所有数据从一个evbuffer移至另一个evbuffer	已完成
evbuffer_add_iovec()	一次性将多个iovec所包含的数据添加到evbuffer末尾	已完成
evbuffer_drain()	从evbuffer的开头删除指定数量的字节数据	已完成
evbuffer_pullup()	使evbuffer从头开始指定长度的内存空间是连续的	已完成
evbuffer_expand()	扩展evbuffer的可用空间	已完成
evbuffer_prepend()	将数据添加到evbuffer的开头	已完成
evbuffer_add_reference()	引用一块内存到evbuffer中, 且没有复制操作	已完成
bufferevent_socket_new()	在现有套接字上创建并返回一个新的bufferevent	已完成
bufferevent_free()	释放与bufferevent结构体关联的内存空间	已完成
bufferevent_get_input()	返回bufferevent中的输入缓冲区	已完成
bufferevent_get_output()	返回bufferevent中的输出缓冲区	已完成
bufferevent_enable()	激活bufferevent的某个事件监听	已完成

bufferevent_disable()	关闭bufferevent的某个事件监听	已完成
bufferevent_get_enabled()	返回给定bufferevent上启用监听的事件	已完成
bufferevent_setcb()	设置bufferevent的各种回调函数	已完成
bufferevent_write_buffer()	将某个evbuffer的数据写入bufferevent的输出缓冲区，且evbuffer中的数据会被清除	已完成
bufferevent_set_timeouts()	设置bufferevent所监听的读写事件的超时时间	已完成
bufferevent_socket_connect()	如果bufferevent尚未设置套接字，则为其分配一个新的流套接字，并使其为非阻塞模式	已完成
event_set_mem_functions()	用于设置自定义的内存管理策略，默认使用malloc、free、realloc等glibc的接口	已完成
bufferevent_flush()	强制bufferevent从底层IO读取或写入尽可能多的字节，忽略其他可能阻止写入的限制	已完成
bufferevent_socket_connect_hostname()	比bufferevent_socket_connect()多了一个地址解析的动作	已完成
bufferevent_socket_connect_hostname()	该接口在libevent-2.1.12-stable中已被废弃	已完成
evsignal_new()	新建一个信号监听事件 libevent本身并未使用该接口，但在其自带的示例程序中用到	已完成
evsignal_add()	将信号监听事件添加到event_base中，开启监听 libevent本身并未使用该接口，但在其自带的示例程序中用到	已完成
event_base_gettimeofday_cached()	获取当前时间，如果可能的话，从event_base中获取缓存的时间，以减少系统调用	已完成
bufferevent_set_rate_limit()	控制bufferevent读写速率	未实现

	libevhttp本身并未使用该接口，但在其自带的示例程序中用到，且在示例程序中，也不是必要的关键接口，因此未实现	
bufferevent_get_openssl_error()和buffer_event_openssl_socket_new()	ssl相关接口 因为没有实现，因此编译libevhttp时需要打开DEVHTP_DISABLE_SSL宏	未实现

测试情况

- **单元测试**：从libevent仓库移植了20个测试用例，全部测试通过。
- **集成测试**：使用libevhttp自带的示例程序进行集成测试。将替换后的libevhttp和原始的libevhttp的运行结果进行对比，测试结果如下表所示。

测试程序	测试结果
example_basic	运行正常
example_request_fini	运行正常
example_chunked	运行正常
example_pause	运行正常
example_vhost	运行正常
test_perf	运行正常
test_query	运行正常
test_client	运行正常
test_vhost	运行正常
test_basic	运行正常
example_locality	运行正常
test_proxy	运行正常
test_extensive	运行正常