

C Programming I

HW0106

Answer

作者: 吳文元 (jw910731)

日期: 2021/03/15

1 Bonus: perror

perror的功能就是可以將errno對應的錯誤訊息打印到stderr去，使用時可以傳一個字串作為輸出時的前綴，方便編寫者追蹤錯誤的原因！

其實說穿了，perror可以被視為strerror的包裝，透過封裝strerror並將他的輸出與給予的參數的一併輸出到stderr的輸出流上！兩者的差異就是，我如果使用strerror，我可以自己選擇要輸出到哪個輸出流，甚至是儲存錯誤資訊，提供更大的彈性，而perror則是較為方便，將常用的錯誤資訊處理方式封裝起來，使除錯更方便！

1.1 perror範例

這裡提供我之前寫的簡易沙盒作為perror的範例，完整的專案請見[simple-sandbox](#)

```
136 void Sandbox::child(const std::vector<std::string> &args) {
137     // prepare args and env
138     char *const *prepared_args = prepare_helper(args);
139     // preserve for env passing
140     char *const *prepared_envs = prepare_helper({});
141     // setup fd for redirection
142     setupFd();
143     // setup rlimit
144     setupLimit();
145     // execute real program
146     int ret = execve(filePath.c_str(), prepared_args, prepared_envs);
147     if(ret < 0){
148         perror("execve()");
149         exit(-1);
150     }
151 }
```

Listing 1: perror Example in src/sandbox.cpp

可以看到裡面使用了perror來捕捉execve所遇到的錯誤，供使用者與程式設計師除錯，在呼叫時，傳入了參數execve():使得在印出錯誤訊息時，可以知道是哪一個函數呼叫失敗並可以得知失敗的原因！這樣如果使用者所給定的執行檔不存在或不具有可執行權限，便可在錯誤訊息中得知！