Jemalloc 记录文档

编辑历史

更新日期	作者	更新内容
2024.02.04	胡益华	Initialization
2024.08.11	胡益华	修改完善对 jemalloc 介绍

目录

1. 引言

1.1 概述

众所周知，C/C++编写的程序拥有极致的运行速度，并且编译后的程序所占的存储空间非常小。因此，C/C++在嵌入式中被广泛应用。

不过其安全问题也是一直被诟病。一般来说，安全问题主要出在内存上。了解 C 语言的工程师对内存泄露、内存碎片等一定不会陌生。尤其是某些嵌入式设备，可用内存本身就岌岌可危，如果程序运行时有持续的内存泄露，那这个进程过不了多久就会被操作系统杀死。

如果一块 2KB 的堆区内存只释放了 1.5KB，那么因为少释放了 0.5KB 就产生了内存泄露；有比如频繁地申请释放小块内存，即使没有发生内存泄露，因此产生的内存碎片也是不理想的。

内存泄漏的影响不仅体现在进程运行时，占用的内存会变大；甚至可能因此受到内存溢出攻击。例如对未被释放的内存块中的数据或代码进行覆盖、替换等，后果都是不堪设想的。

虽然业内都在调侃：没有不安全的语言，只有不安全的工程师。但是再好的开发者也难免出错，为了对抗 C 的内存问题，开发者们也是绞尽脑汁。耳熟能详的 asan、ptmalloc、jemalloc 等本质上都是为了解决 C 的内存问题。这些都是我使用过，体验比较好的工具，本文档主要对 jemalloc 的使用做一些记录。

1.2 jemalloc

jemalloc 是一款开源的软件，它的源代码中通过 alias 对 malloc、calloc、free 等函数进行了重定义。jemalloc 的内存管理机制非常优秀，就我个人使用举例，我的程序正常运行时使用的 RSS 约为 160MB，而使用 jemalloc 后 RSS 变为约 130MB。而且据说 jemalloc 可以显著减少内存碎片的产生。jemalloc 是用 C 语言写的，我粗略浏览过它的源代码，实在难懂，只能评价精通 C 语言，精通各类操作系统，精通各类编译器。

不过因为一些开源协议的特性，jemalloc 似乎不被允许商用，可能某些条件下也可以，这方面我不太懂。如果商用的话，可以用 tcmalloc 等代替。不过就我个人测试的情况而言，jemalloc 真是表现最优秀的。

尽管 jemalloc 不可以直接商用，但它同时提供了记录内存开销，定位内存问题的功能，所以用它来查找定位内存上的漏洞也是很好的选择。

源代码 gitbub 地址如下：

https://github.com/jemalloc/jemalloc/

官方 wiki 地址如下：

https://github.com/jemalloc/jemalloc/wiki

官方环境变量说明地址如下：

http://jemalloc.net/jemalloc.3.html#opt

2. 安装

以 Linux，centos7 环境为例，下载源码后解压。

(1) ./autogen.sh

运行 autogen.sh 文件，用于生成 configure。

(2) ./configure --prefix=/$HOME/tool/jemalloc-dev/artifacts/hi3519dv500 \

--enable-prof \

--build=x86_64-unknown-linux-gnu \

--host=aarch64-linux-gnu \

CC=aarch64-linux-gnu-hi3519dv500-v2-gcc \

CXX=aarch64-linux-gnu-hi3519dv500-v2-g++

运行 configure 文件，用于生成 Makefile。假设需要在 hi3519dv500 芯片上使用 jemalloc。

1) --enable-prof 表示启用堆分析和泄漏检测功能。

2) --build 指定当前系统环境，不指定也可以。

3) --host 表示交叉编译的工具链，不使用--host 只指定编译工具链，会提示：

If you meant to cross compile, use `--host'.

所以 x86_64 平台下不需要指定--host 参数。

(3) make -j && make install

(4) 指定可执行文件环境变量等。