

1. 讲了 Linux 的一些操作

ACL (Access control list)

POSIX ACL

NFS... ACL

gcc 特权操作

bill 文件, output file, 覆盖没写密码

gcc 漏洞 (机制问题)

Linux capability

sel4 introduction

通过 security namespace

传统的和 capability 有什么区别

赋予 capability 只有内核能操作

文件系统支持 capability, 系统可以 track capability

通过 64 位/32 位

超级权限

线程 capability sets

2. cap

kernel 层

困惑的副本

capability 提供了一种面向对象访问控制形式

acl 用环境权限都确定访问权限

为什么引入 capability

以后不用考虑用户是否访问

不用考虑可能有什么恶意访问

数据包过慢, 信息流很慢

符调直

sel4 no capability

Linux no capability

(可加载模板)

限制

之前人

rust 实现 capability.

直接调用了一些 .h 库, unsafe

3. 实时操作系统基于与处理器的调度算法

FreeRTOS 是嵌入式 待调研

QNX 一个嵌入式操作系统，但没有综合所有能力 capability 的权限控制 (sandbox 可用于隔离)

可用于与处理器
与处理器调度效率不理想

实时与核调度算法

我们设计的系统在多核上
运行的能力

减少任务关联性

减少 cache miss

看 **MC 103-2**
单核运行与核

在 **亲和性**

与处理器的亲和性
负载均衡

具体算法

AMP
SMP

与处理器亲和性

基于分组的与核嵌入式

4. 版本迭代 FreeRTOS

最新 v. 10.4

可做的任务:

1. task notification 关注
2. port MPU's
3. stream buffers 关注

任务间通信

再看缓冲介绍

实时操作系统不要有 buffer

会有延迟

4. 针对 RV 更新了一些，但我们更 am，这部分可以不看

7个.C文件, 我们只改了6个
主要 stream buffer

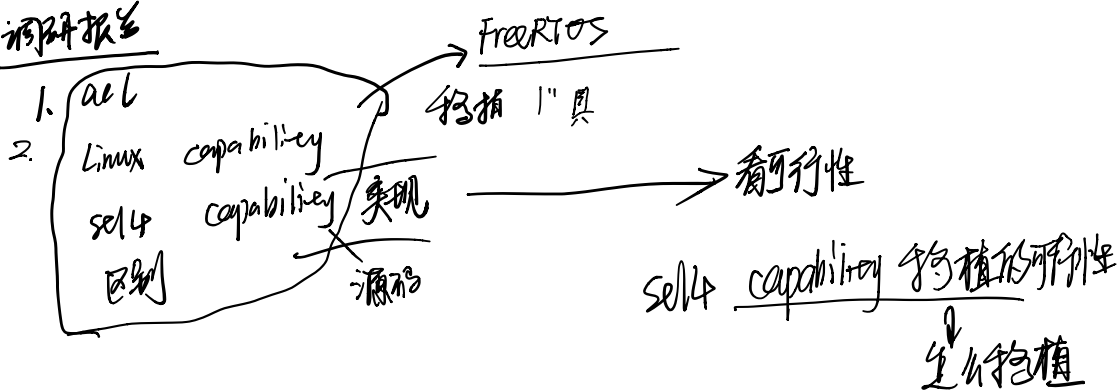
tasks.c 有五行

3': RTEMS 实时处理系统

把单核做到多核从哪些方面修改

1. 启动顺序问题
2. 信号和互斥问题
3. 任务调度算法
4. 移植

调研报告



3. 看论文

多核特性的最小表现

FreeRTOS安全性
没表现.

4. buffer.task
看源码

Thank you