# 阅读报告1

1.论文题目：

Hyperkernel: Push-Button Verification of an OS Kernel

2.阅读报告

OS内核是计算机中最关键的组件之一，其错误会严重影响正确性和安全性。之前的研究中，已经可以通过构造机器可检验的证明，来消除操作系统内核中所有类型的错误，证明实现的行为符合其规范。但是，有很大的实现上开销，对于写出证明的人也有很高的要求。于是本文提出了一种新的解决思路—按钮式方法。借助Z3（一种由微软开发的SMT求解器），来构造一个系统同时通过很低的成本完成正确性证明。

同文件系统和一般的UNIX-like系统调用不同，操作系统有着更多更加复杂的函数和数据结构，以及更加复杂的层次关系。同时，路径数目更大，不便于Z3求解。论文中提出的方法，实际上是针对系统这个目标进行简化和修改。

Hyperkernel实现和正式验证操作系统内核功能正确性的方法，具有高度验证自动化和低证明负担。面临三个挑战：界面设计、内核代码中的虚拟内存管理、用c编写，会使形式化推理变复杂。它引入三个关键思想来实现证明自动化：限定了内核接口以避免无限循环或递归；分离内核和用户地址空间以简化虚拟内存的推理；在LLVM中间表示级别执行验证以避免建模复杂的C语言。

最后论文使用Z3 SMT解算器验证了Hyperkernel的实施，共检查了50个系统调用和其他陷阱处理程序。表明可以避免类似于xv6中发现的错误，以及该验证可以在低证明负担下实现，以较小的代价得到内核正确性的证明，也就是所谓的push-button。论文提供了：一个低开销的方法来构造一个验证过的操作系统内核、一种便于SMT求解的接口设计方法、一个有合适的性能的操作系统（Hyperkernel）的实现。

论文通过两种规范来描述“正确的”内核接口：状态机规范提出了一个内核调用和中断处理函数的状态机，以及这个规范的“正确的”C实现；声明式规范用具体的语言描述规范，提炼这两种规范。第一种包含更多的细节，而第二种在更高的层面上描述大概的作用，便于人类理解。

论文构建经过验证的操作系统内核的按钮式方法以及适用于SMT解决方案的内核接口设计。论文的主要工作主要集中在如何缩小系统并修改系统的接口上，可能有很多的时候都是在证明LLVM中间代码和接口给出的规范时，遇到了求解器不可解或者实现困难的问题，退回来修改系统本身，使得工作的目标从证明系统的正确性，并借此实现一个系统，变成简化的教学操作系统。