

中文图书分类号: TP391.1

密 级: 公开

UDC: 004

学 校 代 码: 10005



硕士专业学位论文

PROFESSIONAL MASTER DISSERTATION

论 文 题 目: 基于 UEFI 的硬盘文件安全加载策略研究与实现

论 文 作 者: 唐治中

专业类别/领域: 计算机技术

指 导 教 师: 张建标 教授

论文提交日期: 2020 年 12 月

UDC: 004
中文图书分类号: TP391.1

学校代码: 10005
学 号: S201861807
密 级: 公开

北京工业大学硕士专业学位论文

题 目: 基于 UEFI 的硬盘文件安全加载策略研究
与实现

英文题目: RESEARCH ON MULTI-VIEW COMPLE-
MENTARITY BASED SLAM

论 文 作 者: 唐治中

专业类别/领域: 计算机技术

研 究 方 向: 信息安全

申 请 学 位: 工程硕士专业学位

指 导 教 师: 张建标 教授

所 在 单 位: 信息学部计算机学院

答 辩 日 期: 2020 年 12 月

授予学位单位: 北京工业大学

独 创 性 声 明

本人声明所呈交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京工业大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

签 名：_____

日 期： 年 月 日

关于论文使用授权的说明

本人完全了解北京工业大学有关保留、使用学位论文的规定，即：学校有权保留送交论文的复印件，允许论文被查阅和借阅；学校可以公布论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存论文。

（保密的论文在解密后应遵守此规定）

签 名：_____

日 期： 年 月 日

导师签名：_____

日 期： 年 月 日

摘 要

中文摘要内容

关键词：同步定位与地图构建；多视角互补；捆绑优化；线条优化；复合特征优化

Abstract

English abstract

Key words: simultaneous localization and mapping; multi-view complementarity; bundle adjustment; line sequence optimization; complex feature optimization

目 录

摘 要	I
Abstract	III
第 1 章 绪论	1
1.1 研究背景及意义.....	1
1.2 国内外研究现状.....	1
1.3 主要研究内容	2
1.4 本文组织结构	3
第 2 章 相关知识及技术介绍	5
2.1 UEFI 概述.....	5
2.1.1 UEFI 系统结构.....	5
2.1.2 UEFI 协议运作方式	7
2.2 UEFI 固件文件系统数据存储方式介绍	8
2.3 UEFI 文件系统协议栈.....	10
2.3.1 总体介绍.....	10
2.3.2 相关驱动介绍	10
2.4 BMC 技术介绍	10
2.4.1 BMC 在系统中的位置	10
2.4.2 BMC 与 BIOS 通讯方式	10
2.5 可信计算技术	10
2.5.1 可信计算信任链	10
2.5.2 可信平台模块	10
第 3 章 基于 UEFI 的硬盘文件安全加载策略研究	11
3.1 安全漏洞分析	11
3.1.1 UEFI 环境文件加载过程分析	11
3.1.2 从硬盘攻击关键文件.....	11
3.1.3 从固件芯片攻击关键驱动	11
3.2 信任链的传递	11
3.3 总体架构设计	11
3.4 模块设计	11
3.5 本章小结	11

第 4 章 基于 UEFI 的硬盘文件可信加载详细设计	12
4.1 度量计算模块设计	12
4.2 硬盘文件度量模块设计	12
4.3 驱动文件度量模块	12
4.4 固件和硬盘访问模块设计	12
4.4.1 固件访问设计	12
4.4.2 硬盘访问设计	12
4.5 BMC 通信模块设计	12
4.5.1 BMC 驱动设计	12
4.5.2 BMC 驱动度量方式	12
4.6 本章小结	12
第 5 章 硬盘文件可信加载方案实现及测试	13
5.1 BMC 驱动程序实现	13
5.2 UEFI 启动过程中驱动文件度量实现	13
5.3 UEFI 环境中硬盘文件度量实现	13
5.4 可信度量模块实现	13
5.5 测试与分析	13
5.6 本章小结	13
第 6 章 实例章节	14
6.1 二级标题	14
6.1.1 三级标题	14
6.2 公式	14
6.2.1 引用	15
6.2.2 插图	15
6.2.3 表格	15
6.2.4 列表	17
结 论	19
攻读硕士学位期间发表的学术论文	21
参考文献	23
致 谢	25

第 1 章 绪论

1.1 研究背景及意义

目前，在计算机应用领域，统一可扩展固件接口 UEFI（下同）已经成为了主流基础输入输出系统 BIOS（下同）实现方式并逐渐取代了传统 BIOS，作为一个越来越成熟的 BIOS 系统，UEFI 环境就包括了对硬盘文件加载和运行的功能，其中包括：在调试关键驱动程序时，可能需要把驱动文件放入硬盘并在 UEFI 环境中手动加载；操作系统启动需要经过 UEFI 环境，并在其中加载硬盘中的操作系统引导程序完成启动。在 UEFI 初始化完成后，可完整使用 UEFI 提供的系统功能，其中就包括了硬盘中的可扩展固件接口文件系统分区 ESP（Efi system partition）（下同）分区数据操作。这块 UEFI 可访问到的硬盘空间上会存储一些 UEFI 编译过程中重要的可扩展固件接口可执行文件.efi（下同）文件，其中就包括了操作系统启动引导文件 start_kernel.efi 这样的关键文件，因此如何在 UEFI 环境中安全的加载硬盘设备文件就成了关键问题。

因为在 UEFI 环境下访问硬盘文件，所以这些文件存在通过硬盘被篡改和通过 UEFI 文件系统被篡改的可能性。现有技术中有针对硬盘文件的硬件攻击方法，通过硬件手段修改硬盘中关键文件的内容起到注入木马程序的效果。还有一种现有技术中针对存放 BIOS 程序的闪存 flash（下同）芯片提出了一种攻击手段，其攻击原理为通过硬件手段和 flash 中数据存储格式，修改 flash 芯片中的驱动程序内容起到注入木马程序的效果。因为 UEFI 中访问硬盘文件需要经过 UEFI 文件系统协议栈，因此通过现有技术手段可修改协议栈相关驱动，从 UEFI 固件层面达到攻击效果。因此对于 BIOS 固件层面的块设备文件系统的可信验证是极其具有必要性的。

1.2 国内外研究现状

房强等在“基于固件文件系统的 UEFI 安全机制研究”一文中通过对 UEFI 安全威胁研究的分析与总结，提出了基于可信平台模块 TPM（下同）的静态度量固件文件系统中驱动程序的方法。该方案以 TPM 为可信锚点，此信任根可根据如基板管理控制器 BMC（下同）这样的底层硬件进行替代和改进；其次对于固件文件系统 FFS（下同）中的驱动文件度量为静态度量过程，及在 UEFI 初始化阶段结束后再进行度量，这样就无法保证初始化过程中的安全性。段晨辉等在“UEFI BIOS 安全增强机制及完整性度量的研究”一文中通过对 UEFI 启动阶段中信任链的设计和底层基于可信计算组织 TCG（下同）的可信链构建方案，提

出了在 UEFI 完整启动阶段过程中通过逐个阶段度量后面阶段的内容起到可信启动的效果。但该方案在整个度量过程中需要涉及到驱动加载准备阶段 PEI 阶段（下同）核心代码、所有 PEI 模块程序、驱动加载阶段 DXE（下同）核心代码、DXE 调度器代码、所有 DXE 驱动程序代码的度量工作，缺少对于特定如硬盘设备文件的文件系统的针对性安全验证，并且整个度量过程繁琐耗时，虽然做到了动态度量效果，但具有相当大的局限性，可行性不高。

专利公开号为 CN 104573499 B 专利名称为“一种基于 UEFI 的可执行程序文件保护系统和方法”的专利提出了一种通过 UEFI 固件层面的文件安全存储策略，来保证硬盘文件受到攻击和恶意篡改时可通过 BIOS 固件中的文件信息进行复原。但是该专利忽略了对于固件硬件平台的攻击可能性，无法保证固件中备份文件的安全性，即文件的硬件安全防护能力并不突出。专利公开号为 CN 110020562 A 专利名称为“基于 UEFI 的硬盘全加密方法及装置”的专利提出了一种通过检测硬盘设备是否安全可信的机制完成硬盘文件的安全加载。但是该方案忽略了通过篡改 BIOS 固件中文件系统相关驱动达到修改硬盘文件的手段，一旦固件中的文件系统被恶意篡改，在 UEFI 环境中加载硬盘文件不存在可信而言。

大部分基于 UEFI 的文件加载方案，没有对固件层面的 UEFI 文件系统驱动可信度量的过程，无法保证固件层面针对硬盘文件内容的攻击；而且提出的固件驱动的量度局限于静态度量，无法做到真实运行环境中的动态度量。不能有效保证硬盘文件的可信性，不能保证 UEFI 环境中硬盘文件的安全加载。

1.3 主要研究内容

鉴于 UEFI 是 BIOS 系统的一种统一可扩展标准和方案，拥有着模块化的系统部件添加结构，使他成为 BIOS 功能开发的首选方案，目前市场上的 BIOS 实现也越来越多的统一使用 UEFI。与此同时，UEFI BIOS 作为计算机启动到加载操作系统的一个中间阶段，需要在系统启动时加载硬盘中 ESP 分区中的操作系统引导程序；并且随着 UEFI BIOS 的功能的不断发展，UEFI BIOS 环境中甚至能够播放视频音频，而硬盘设备又是计算机中数据的主要存储介质，这就增加了 BIOS 与硬盘设备之间交互频率，BIOS 环境中加载硬盘设备文件的安全性就显得更加重要。因此研究基于 UEFI 的硬盘设备文件加载的安全机制就具有重要意义。

本文主要研究内容如下：

(1) 针对 UEFI BIOS 系统加载硬盘文件过程中存在的安全威胁，通过研究硬盘设备用于存放 UEFI BIOS 环境中可访问数据的 ESP 分区的组织结构，研究

UEFI BIOS 环境中读取硬盘文件的方式，以及研究以 UEFI 型驱动程序作为基础的可信度量机制，以此确保针对硬盘设备攻击的文件安全性。结合可信计算理论提出基于 UEFI 的硬盘设备文件可信加载的总体框架，研究解决 UEFI BIOS 环境加载硬盘文件过程中的安全可信。

(2) 针对 UEFI BIOS 系统加载 UEFI BIOS 存储闪存设备中的驱动文件过程中存在的安全问题，研究作为 UEFI BIOS 中最为关键的系统组件 UEFI 驱动程序在闪存介质中的数据存储格式，研究 PEI 和 DXE 两个加载 UEFI 驱动程序的主要启动阶段的具体驱动加载流程和原理，研究可信度量功能在这两个主要阶段中的构建位置。以此确保针对 BIOS 闪存攻击硬盘文件系统程序（以达到在 BIOS 加载硬盘文件时篡改文件内容的目的）过程的安全性。

(3) 针对 UEFI BIOS 系统在运行阶段加载硬盘设备文件过程中，文件数据经过 UEFI 文件系统协议栈逐层向上传递数据过程中的安全问题，研究 UEFI 文件系统协议栈中每层对应的 UEFI 协议，以及协议对应的 DXE 阶段的具体驱动实现。以此确定在 DXE 阶段度量 BIOS 闪存中的文件系统程序时，具体应操作的驱动对象。

(4) 根据本系统安全方案中使用的底层可信平台 BMC 及基板管理控制器，研究 UEFI BIOS 中 BMC 驱动程序的实现原理及具体过程，研究 UEFI BIOS 和 BMC 如何通过基础驱动程序构建安全方案中取得基准值及度量报告的写入方法，以此确保 BMC 作为本系统安全方案的可信平台及可信根。

1.4 本文组织结构

全文结构一共分为六大部分，各部分内容如下：

- 第一章 绪论

首先对本文的研究背景和研究意义进行了介绍，然后阐述了现有的 UEFI 环境下硬盘文件保护方法的现状及本系统在这个基础上做的改进。介绍了本文的主要研究内容，最后介绍了本文的组织结构。

- 第二章 相关知识及技术介绍

首先对 UEFI BIOS 及可扩展固件接口标准的基础输入输出系统的整体设计和层次结构进行了介绍，包括启动流程、协议加载方式、数据库句柄及固件文件系统数据存储格式。其次对 UEFI 文件系统协议栈及涉及到的具体驱动程序做了进一步说明。然后介绍了 BMC 及基板控制管理器的基本功能及与 BIOS 通信方式。最后介绍了可信计算技术的现状及发展。

- 第三章 基于 UEFI 的硬盘文件可信加载总体架构设计

首先对 UEFI 环境中加载硬盘文件的安全漏洞进行了分析，其次介绍了此系

统中信任链的涉及与传递，然后介绍了系统的总体架构设计，包括了 DXE 和运行时两个主要的度量阶段。最后介绍了此系统的功能模块划分。

- 第四章 基于 UEFI 的硬盘文件可信加载详细设计

分别对度量计算模块、硬盘文件度量模块、驱动文件度量模块、固件和硬盘访问模块、日志记录模块、BMC 通信模块给出了具体设计和详细介绍。

- 第五章 硬盘文件可信加载方案实现及测试

根据此系统的安全方案设计，介绍了如 BMC 驱动、文件系统相关驱动和硬盘中关键文件度量的代码实现过程，以及实验相关的编译运行环境。通过实验说明了安全漏洞的具体攻击方法和此系统的预防效果。

- 结论

对本文的相关研究工作进行了总结和分析，并根据功能测试结果指出本方案的不足之处。并提出了下一步的研究方向。

第 2 章 相关知识及技术介绍

2.1 UEFI 概述

本章将会介绍 UEFI 系统中关系到本文关键技术的基础内容，其中具体包括了 UEFI 基本结构、UEFI 固件存储格式、UEFI 文件系统协议栈相关内容及驱动程序介绍，还包括基板管理控制器 BMC 及可信计算的基础内容。

2.1.1 UEFI 系统结构

UEFI（Unified Extensible Firmware Interface，统一可扩展固件接口）定义了操作系统和平台固件直接的接口，UEFI 提供了一个统一可扩展的固件平台，并针对平台特性定义了一系列接口。该平台整体处于硬件与操作系统中间，平台最上层的可扩展固件接口包含了平台提供的 API 函数、启动时服务（EFI Boot Services）、运行时服务（EFI Runtime Services）和操作系统引导程序，下层则是根据 UEFI 规范实现的平台固件。UEFI 的平台架构如图 2-1 所示：

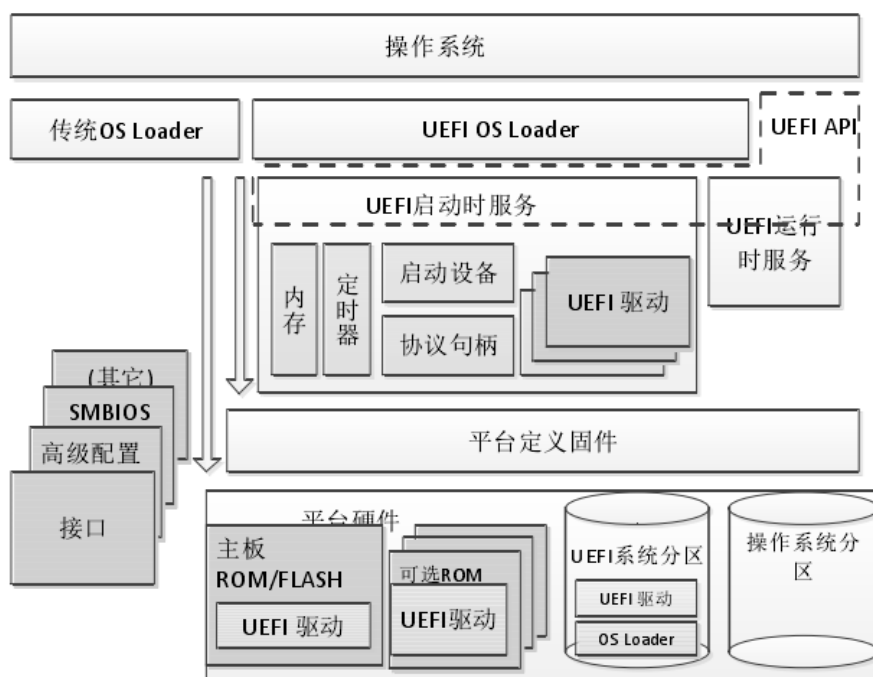


图 2-1 UEFI 系统框架图

Figure 2-1 Infrastructure of UEFI

图 2-1 描绘了 UEFI 框架中各模块的关系，UEFI 固件作为承上启下的模块，对底层硬件进行了抽象处理，又不断地对上层的操作系统提供服务，并在不同的服务层的连接中采用了标准的接口。图中 UEFI 保留了对传统 BIOS 引导操作系

统的兼容性，所以在可扩展固件接口的实现中有两种操作系统引导方式，分别为 Legacy OS Loader 和 UEFI OS Loader。

UEFI 允许操作系统预处理，实现了操作系统的引导和一些系统软件执行所需要的其它应用程序，如诊断程序、UEFI Shell、系统调试软件等，这些程序统称为 UEFI 实体（UEFI Image）。根据 UEFI 规范，UEFI Image 包含三种：UEFI 应用程序、UEFI 驱动和 OS Loaders，这些实体都是在 UEFI API 调用的基础上实现的。

从图中可以看出，UEFI 的启动时服务 Boot Service 中包含了 UEFI 在 PEI 和 DXE 两个主要加载驱动完成系统初始化过程中所需要的驱动程序、内存管理、设备管理、协议贮存等信息。而 UEFI 的启动时服务也有他自己的生命周期，从下图就可以看出启动时服务在 UEFI 启动过程中的位置。

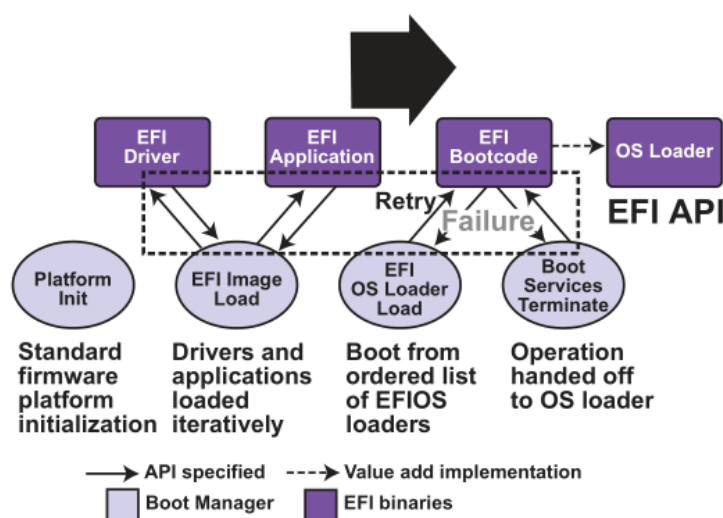


图 2-2 UEFI 启动流程图

Figure 2-2 Booting Sequence of UEFI

从图 2-2 中可以看出，Platform Init 平台初始化过程中建立 Boot Service 系统服务和 RunTime Service 系统服务，EFI Image Load 阶段包括了 PEI 和 DXE 两个主要驱动加载过程，负责加载 BIOS 固件中的 UEFI 驱动和 UEFI 应用程序的 EFI 可执行文件，之后在启动设备选择也就是 BDS 阶段中，通过用户的选择，UEFI BIOS 选择适当的操作系统引导程序进行加载，并同时退出 Boot Service 服务，并继续向上层操作系统提供 RunTime Service 服务。因而在操作系统运行过程中，可以继续使用底层 UEFI BIOS 提供的运行时服务。结合图 2-1 也可以发现，UEFI 的一些关键驱动程序，和 OS Loader 也会存放在如硬盘块设备的 ESP（EFI System Partition）系统分区中。由于本文主要对于 UEFI 启动过程中的 DXE、BDS 阶段进行安全方案设计，因此，启动时服务其中包含的如协议加载函数等为本文的主要研究对象。

2.1.2 UEFI 协议运作方式

UEFI 中协议设计的思想为，由于 UEFI 的官方提供实现的版本为 C 语言实现，而 C 语言是一种面向过程的语言，而完全使用面向过程的思想来管理和使用众多 UEFI 协议将会使程序变得非常复杂。Protocol 作为一种对象来设计管理会更加直观。因而 UEFI 中的 Protocol 引入了面向对象的思想，其中包括：

- 用 struct 来模拟 class。
- 用函数指针（Protocol 的成员变量）模拟成员函数，此种函数的第一参数必须是指向 Protocol 的指针，用来模拟 this 指针。

从图 2-1 中可以看出，UEFI 中的协议包含于 UEFI 启动时服务中（Boot Services），由启动时服务提供的功能进行协议的加载、保存和调用等操作。其中 UEFI 启动时服务提供的协议相关的功能函数如表 2-1 所示。

表 2-1 中列出了 Boot Service 中包含的所有 UEFI 协议相关的功能函数，其中最为常见的如 InstallMultipleProtocolInterfaces 这样的加载协议函数，在众多 UEFI 应用以及底层驱动程序中都十分常见，因为他可以同时提供出需要加载的多个协议 GUID。

同时，由于启动时服务提供了追踪最新安装了的新协议内容以及他们的使用情况，因此对于 UEFI 的启动时服务来说，它可以安全地卸载并重新安装由 UEFI 驱动程序使用的协议接口。

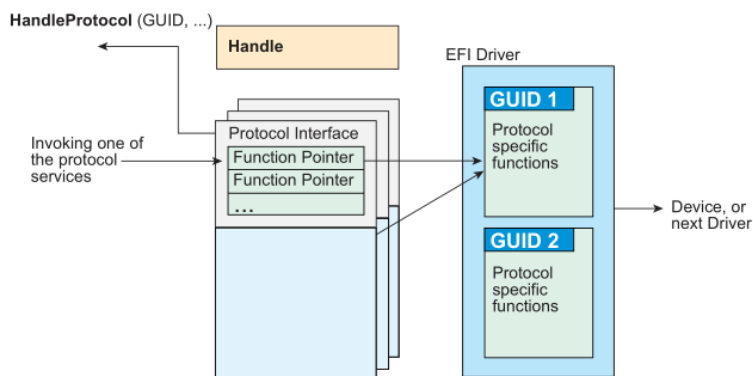


图 2-3 UEFI 协议加载方式图

Figure 2-3 Locating Protocol of UEFI

协议的加载过程可通过图 2-3 分析得知。在图 2-3 中可以看出，协议由 `HandleProtocol` 等表 2-1 中列出的装载协议用的功能函数加载到 `Handle` 句柄上，而所有的 `Handle` 则由 UEFI 内核统一存储于句柄数据库，句柄数据库也是一个链表结构用于存储记录所有的 `Handle` 句柄，这些句柄可由任意的 UEFI Image（镜像）访问，从而达到函数调用的效果。而这些协议中包含的是指向具体函数的 C 语言中的函数指针，这些具体函数则是在 DXE 阶段由 UEFI 系统表提供的加载驱动镜像函数加载并驻留在内存中的。

表 2-1 启动时服务协议功能表
Table 2-1 Boot Service Protocol Interface Functions

名称	类型	描述
InstallProtocolInterface	Boot	在设备句柄上安装一个协议接口
UninstallProtocolInterface	Boot	从设备句柄上移除一个协议接口
ReinstallProtocolInterface	Boot	在设备句柄上重新安装协议接口
RegisterProtocolNotify	Boot	注册一个事件，只要接口有信号为指定的协议安装
LocateHandle	Boot	返回支持指定协议的句柄数组
HandleProtocol	Boot	查询句柄以确定它是否支持指定的协议
LocateDevicePath	Boot	找到支持指定路径的设备路径上的所有设备协议并将句柄返回到最接近的设备路径
OpenProtocol	Boot	将元素添加到使用协议的代理列表中接口
CloseProtocol	Boot	从代理列表中移除一个元素，也就是消耗一个协议接口
OpenProtocolInformation	Boot	检索当前正在使用的代理列表协议接口
ConnectController	Boot	使用一组优先规则来找到最佳的驱动程序集管理一个控制器
DisconnectController	Boot	通知一组驱动程序以停止管理控制器
ProtocolsPerHandle	Boot	检索安装在句柄上的协议列表，函数返回的缓冲区是自动分配的
LocateHandleBuffer	Boot	从句柄数据库中检索句柄列表，该列表符合搜索条件，返回缓冲区自动已分配
LocateProtocol	Boot	在句柄数据库中找到第一个支持所需协议的句柄
InstallMultipleProtocolInterfaces	Boot	将一个或多个协议接口安装到指定句柄上
UninstallMultipleProtocolInterfaces	Boot	从指定句柄中卸载一个或多个协议接口

2.2 UEFI 固件文件系统数据存储方式介绍

UEFI 固件文件系统指的是 BIOS 闪存芯片中的数据存储格式，它通过统一的固件文件系统标准用来统一闪存芯片中的文件内容和 UEFI 启动阶段内存中文件的内容。具体的固件中 UEFI 可执行程序文件存储格式如图 2-4 所示。

在图 2-4 中，左边为通过结构图的方式说明固件文件系统的数据存储格式，右边为通过数据结构中的树形结构来阐述 FFS 中的文件存储。右图中，蓝色方框代表了一个完整的 FFS 中的文件映像也就是 UEFI 中可执行文件的二进制数

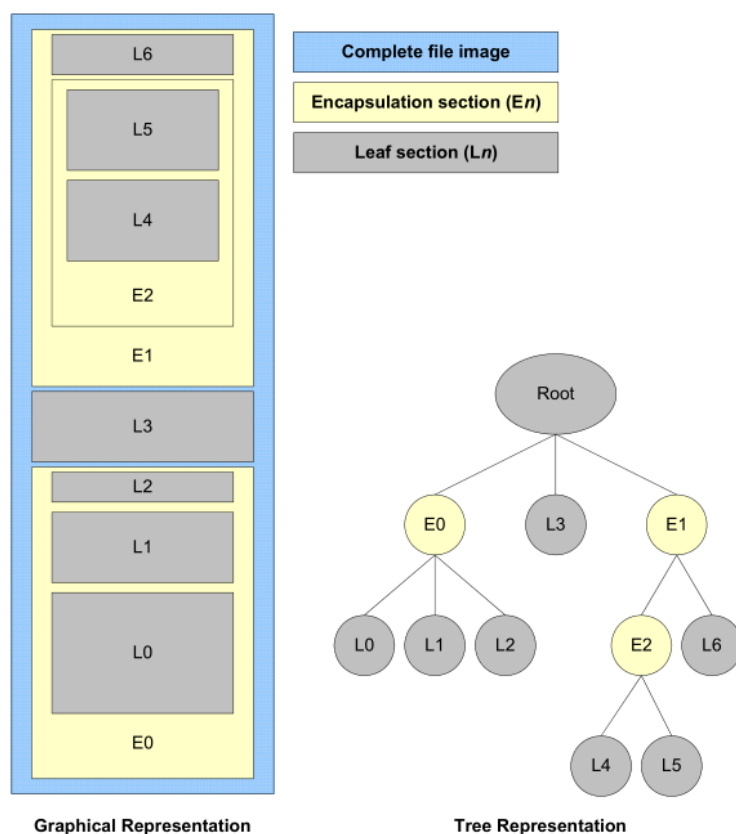


图 2-4 固件文件系统文件存储格式

Figure 2-4 Firmware file system file storage format

据，黄色方框代表了一个父目录结构，黄色方框中包含的灰色方框代表了父目录中的子目录，也就是文件影响中的最小单位。对应到右图的树结构中，黄色节点就是树中有子节点的父节点，而灰色节点则代表了叶子节点。了解固件文件系统文件的数据存储格式有助于理解 UEFI 内核在系统初始化过程中调用相关解析 FFS 中文件函数的运行过程，也有助于帮助分析本文安全方案中可信度量的具体数据内容。

2.3 UEFI 文件系统协议栈

2.3.1 总体介绍

2.3.2 相关驱动介绍

2.4 BMC 技术介绍

2.4.1 BMC 在系统中的位置

2.4.2 BMC 与 BIOS 通讯方式

2.5 可信计算技术

2.5.1 可信计算信任链

2.5.2 可信平台模块

第 3 章 基于 UEFI 的硬盘文件安全加载策略研究

3.1 安全漏洞分析

3.1.1 UEFI 环境文件加载过程分析

3.1.2 从硬盘攻击关键文件

3.1.3 从固件芯片攻击关键驱动

3.2 信任链的传递

3.3 总体架构设计

3.4 模块设计

3.5 本章小结

第 4 章 基于 UEFI 的硬盘文件可信加载详细设计

- 4.1 度量计算模块设计
- 4.2 硬盘文件度量模块设计
- 4.3 驱动文件度量模块
- 4.4 固件和硬盘访问模块设计
 - 4.4.1 固件访问设计
 - 4.4.2 硬盘访问设计
- 4.5 BMC 通信模块设计
 - 4.5.1 BMC 驱动设计
 - 4.5.2 BMC 驱动度量方式
- 4.6 本章小结

第 5 章 硬盘文件可信加载方案实现及测试

- 5.1 BMC 驱动程序实现
- 5.2 UEFI 启动过程中驱动文件度量实现
- 5.3 UEFI 环境中硬盘文件度量实现
- 5.4 可信度量模块实现
- 5.5 测试与分析
- 5.6 本章小结

第 6 章 实例章节

以下是简单的示例代码。

加粗 [title= 代码段题目]

```
EFI_STATUS InfoRead(char *FileName,char *Buf)
{
    EFI_STATUS Status=0; //4个空格
    EFI_FILE_PROTOCOL *FileHandle=0; //4个空格
        EFI_FILE_PROTOCOL *RootHandle; //tab
    UNINT BufSize=10240; //2个空格
}
```

插入文本格式代码：

```
void main() {
    return 0;
    ASDFSJOGASJDIGOS;
}
```

6.1 二级标题

6.1.1 三级标题

6.1.1.1 四级标题

段落标题 这是一段文字。

6.2 公式

$$E_p = \sum_i \rho_h(e_{p,i}^T \Omega_i^{-1} e_{p,i}) \quad (6-1)$$

其中， ρ_h 为 Huber 鲁棒代价函数，增加了系统对于噪声点的鲁棒性……

数据集	ORB-SLAM	PL-SLA
TUMaaaaaaaaa	20.6aaaaaaaaa	45.9cccccccc
7-Scenesaaaaaa	20.0bbbbbbbbbb	45.4cccccccc

p_t	21	22
-------	----	----

		13caaaaaa
c_t aaaaaaaaaaaa	EFI_SIMPLE_TEXT_INPUT_PROTOCOL*	aaaaaaaa
		cccc
h_t aaaaaaaaaaaa	10bbbbbbbbbb	5cccccccc

6.2.4 列表

enumerate 列表带编号，默认样式 1.

- #### 6.2.4.2 itemize

-17-

- 第 1 章 绪论
- 第 2 章 相关知识及数据集
- 第 3 章 基于多视角互补的 SLAM 框架
- 第 4 章 基于多视角互补的捆绑优化算法
- 第 5 章 基于多视角互补的线条优化算法
- 第 6 章 基于多视角互补的复合特征优化算法
- 结论

结 论

这是 tzz 的论文结论

攻读硕士学位期间发表的学术论文

1.

参考文献

- [1] Rublee E, Rabaud V, Konolige K, et al. ORB: An efficient alternative to SIFT or SURF[C]. International Conference on Computer Vision, 2011. 2564–2571.
- [2] Gioi R G V, Jakubowicz J, Morel J M, et al. LSD: A fast line segment detector with a false detection control[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2008, 32(4):722–732.
- [3] Mur-Artal R, Tardós J D. ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras[J]. IEEE Transactions on Robotics, 2017, 33(5):1255–1262.
- [4] 唐治中. 基于多视角互补的 slam 算法研究学位论文. 北京: 北京工业大学大学信息学部计算机学院, 2020.

致 谢

时光荏苒，三年的研究生生活即将结束……

