**课题编号： 密级：**

**国家重点研发计划**

**课题实施方案**

|  |  |
| --- | --- |
| 课题名称： |  |
| 课题承担形式： | 联合承担 |
|  |  |
| 课题单位： |  |
| 课题参与单位： |  |
|  |  |
|  |  |
| 课题起止年限： |  |

**目 录**

[1. 文档介绍 1](#_Toc364957426)

[1.1 文档目的 1](#_Toc364957427)

[1.2 文档范围 1](#_Toc364957428)

[1.3 读者对象 2](#_Toc364957429)

[1.4 术语与解释 3](#_Toc364957430)

[1.5 参考资料 3](#_Toc364957431)

[1.6 文档概览 4](#_Toc364957432)

[2. 总体设计 6](#_Toc364957433)

[2.1 需求规定 6](#_Toc364957434)

**[2.1.1 功能要求 6](#_Toc364957435)**

**[2.1.2 性能要求 10](#_Toc364957436)**

**[2.1.3 输入输出要求 12](#_Toc364957437)**

**[2.1.4 其它专门要求 13](#_Toc364957438)**

[2.2 运行环境 15](#_Toc364957439)

**[2.2.1 硬件环境 15](#_Toc364957440)**

**[2.2.2 软件环境 16](#_Toc364957441)**

[2.3 基本设计概念 20](#_Toc364957442)

**[2.3.1 技术架构 20](#_Toc364957443)**

**[2.3.2 业务架构及流程 22](#_Toc364957444)**

**[2.3.3 设计约束 36](#_Toc364957445)**

[2.4 详细系统架构 37](#_Toc364957446)

[2.5 功能需求与系统模块的关系 46](#_Toc364957447)

[2.6 尚未解决的问题 53](#_Toc364957448)

[2.7 开发工具 53](#_Toc364957449)

[2.8 遵循标准的设计 54](#_Toc364957450)

[2.9 本系统与其他课题交互 55](#_Toc364957451)

[3. 关键技术方案 57](#_Toc364957452)

[3.1 <关键技术1-XXXX> 57](#_Toc364957453)

**[3.1.1 问题描述 57](#_Toc364957454)**

**[3.1.2 解决方案 57](#_Toc364957455)**

[3.2 <关键技术2- XXXX > 59](#_Toc364957456)

**[3.2.1 问题描述 59](#_Toc364957457)**

**[3.2.2 解决方案 60](#_Toc364957458)**

[4. 子系统设计 73](#_Toc364957474)

[4.1 <子系统1-XXX> 73](#_Toc364957475)

**[4.1.1 技术及系统架构 73](#_Toc364957476)**

**[4.1.2 实现流程 78](#_Toc364957477)**

[4.2 <子系统2-XXX> 82](#_Toc364957478)

**[4.2.1 技术及系统架构 82](#_Toc364957479)**

**[4.2.2 实现流程 85](#_Toc364957480)**

[5. 界面设计 109](#_Toc364957499)

[5.1 <界面1-XXX> 109](#_Toc364957500)

[5.2 <界面2-XXX> 109](#_Toc364957501)

**[5.2.1 可信计数器管理界面 109](#_Toc364957502)**

**[5.2.2 可信计数器新增界面 110](#_Toc364957503)**

[6. 接口设计 120](#_Toc364957515)

[6.1 外部接口 120](#_Toc364957516)

**[6.1.1 提供XXX使用的外部接口 121](#_Toc364957517)**

**[6.1.2 提供XXX使用的外部接口 124](#_Toc364957518)**

[6.2 内部接口 127](#_Toc364957520)

**[6.2.1 内部接口1-XXX系统提供的接口 128](#_Toc364957521)**

**[6.2.2 内部接口2-XXX提供的接口 131](#_Toc364957522)**

[6.3 其它接口 141](#_Toc364957526)

[7.运行设计 142](#_Toc364957527)

[7.1 运行模块组合 142](#_Toc364957528)

[7.2 存储设计 146](#_Toc364957529)

[7.3 运行控制 146](#_Toc364957530)

[7.4 运行时间 146](#_Toc364957531)

[8. 系统数据结构设计 148](#_Toc364957532)

[8.1 主要数据结构设计 148](#_Toc364957533)

**[8.1.1 XXX 148](#_Toc364957534)**

**[8.1.2 XXX 148](#_Toc364957535)**

[8.2 数据结构与系统模块的关系 153](#_Toc364957543)

[8.3 数据库设计 156](#_Toc364957544)

**[8.3.1 数据库表结构 156](#_Toc364957545)**

**[8.3.2 数据库实体关系图 176](#_Toc364957546)**

**[8.3.3 其它数据库设计 181](#_Toc364957547)**

[8.4 其它数据相关设计 181](#_Toc364957548)

[9. 系统错误处理设计 183](#_Toc364957549)

[9.1 出错信息 183](#_Toc364957550)

[9.2 补救措施 185](#_Toc364957551)

[9.3 系统维护设计 187](#_Toc364957552)

[10. 安全性设计 189](#_Toc364957553)

[10.1 概述 190](#_Toc364957554)

**[10.1.1 设计原则 190](#_Toc364957555)**

**[10.1.2 设计思路 191](#_Toc364957556)**

[10.2 XXX安全设计 191](#_Toc364957557)

**[10.2.1 XXX的安全设计 191](#_Toc364957558)**

**[10.2.2 XXX的安全设计 194](#_Toc364957559)**

**[11. 性能设计](#_Toc364957568)** [212](#_Toc364957568)

**[11.1 性能指标描述](#_Toc364957569)** [212](#_Toc364957569)

**[11.2 性能指标分析](#_Toc364957570)** [213](#_Toc364957570)

**[11.3 性能指标运行环境](#_Toc364957571)** [215](#_Toc364957571)

[12. 附录 219](#_Toc364957572)

版本信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **版本/状态** | **作者** | **参与者** | **编制/修订历史时间** | **备注(修改单号)** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**1. 文档介绍**

本章应提供整个概要设计说明书的概述。它应包括此概要设计说明书的目的、范围、定义、首字母缩写词、缩略语、参考资料和概述。

**1.1 文档目的**

此文档以需求分析阶段得到的软件需求为依据，为目标系统的逻辑架构的设计和实现而编写的《概要设计说明书》。编制此文档的目的是为了对该包的整体需求、总体框架、关键技术方案、各子系统设计、界面设计、接口设计、运行设计、数据结构设计以及系统错误处理设计等进行定义，对开发环境、测试和运行环境配置进行约定，明确系统功能范围，使之能够为项目后期的系统详细设计提供指导，也为产品实施（开发、测试）、集成优化以及与各相关包和后期运营提供参考和依据。

**1.2 文档范围**

[简要说明此概要设计说明书适用的软件应用程序、特性或其他子系统分组、与其相关的用例模型，以及受到此文档影响的任何其他事物。]

**1.3 读者对象**

该文档将需求分析阶段得到的软件需求，转变为目标系统的逻辑架构。从而确定能实现软件的功能，性能要求集合的最合理的软件系统结构，在较高层次，实现对该安全虚拟集群系统管理软件系统的模块划分。因此，该文档的读者主要是需求分析人员、详细设计人员、系统开发人员、系统测试人员以及集成测试人员。

**1.4 术语与解释**

Hypervsior: 虚拟机监控器，用于提供虚拟资源和管理虚拟机。

KVM: Kernel-based Virtual Machine 基于内核的虚拟机，是使用硬件虚拟化技术的开源Hypervisor，自Linux 2.6.20之后集成到Linux的各个主要发行版本中。

VM: Virtual Machine 虚拟机，通过软件模拟的具有完整硬件系统功能的、运行在一个完全隔离环境中的完整计算机系统。

GOS: Guest Operating System 客户机操作系统，运行在虚拟化环境中的操作系统。

HOS: Host Operating System 宿主机操作系统，用于承载虚拟化环境。

GVA: Guest Virtual Address 客户虚拟地址。

GPA: Guest Physical Address 客户物理地址。

HVA: Host Virtual Address 宿主机虚拟地址。

HPA: Host Physical Address 宿主机物理地址。

EPT: Extended Page Table 扩展页表，用于虚拟化环境中将GOS的物理地址GPA转化为HOS的物理地址HPA。

PUT: Page Usage Table 页表使用情况表，用于记录物理页的真实使用情况。

**1.5 参考资料**

[本小节应完整列出此概要设计说明书中其他部分所引用的任何文档。]

《计算机软件开发规范（GB8566-88）》

《系统虚拟化》：英特尔开源软件技术中心. 系统虚拟化[M]. 清华大学出版社, 2009.

《软件工程导论》：摩尔, R) 著, 马振晗,等. 软件工程导论: 第2版[M]. 清华大学出版社, 2008.

《软件工程技术概论》：朱三元, 钱乐秋, 宿为民. 软件工程技术概论[J]. 2002.

**1.6 文档概览**

本文档对“虚拟机安全套件核心模块”的整体定位、业务流程、功能描述（包括变更功能及扩充功能）、系统接口进行定义，也对系统的设计约束、数据字典、质量特性、适用标准等进行定义。各章内容具体说明如下：

第1章对文档的编写目的、文档范围、预期读者、相关术语解释、参考资料和文档编写的组织方式进行介绍；

第2章对项目进行总体设计描述，包括本项目的需求规定、软硬件运行环境、技术架构、业务架构、系统架构、功能需求与系统模块的关系以及尚未解决的问题等，并对开发工具加以确认。

第3章对项目设计的各个关键技术进行阐述，包括对问题的描述以及相应的解决方案。

第4章对项目涉及的所有子系统，分别进行系统架构和实现流程的阐述。

第5章阐述项目的界面设计，并对主要界面进行展示和说明。

第6章阐述项目的接口设计，并对项目涉及的多个内、外部接口和其它接口分别进行描述。

第7章对项目的运行进行涉及，包括运行模块组合、运行控制和运行时间三方面的内容。

第8章对系统的数据结构进行详细的设计说明。包括主要的数据结构设计、数据结构与系统模块的关系以及数据库设计等。

第9章对系统中的错误处理进行了规范，包括出错信息规范、补救措施说明以及系统维护的设计。

第10章是系统安全设计，包括系统安全设计机制，框架。

第11章附录。

**2. 总体设计**

[本章应提供本包交付物的总体设计情况，其中的关键技术及子系统设计等具体内容在后续相应章节中进行细化。]

**2.1 需求规定**

[本节说明对本包交付物主要的功能、性能、输入输出等要求。]

**2.1.1 功能要求**

[列出合同及需求规格说明书中对本包交付物的功能要求。]

虚拟机安全套件三大功能：

支持内存分区域存储，物理的隔离敏感数据不被篡改；

支持内存动态分配标记、追踪和内存数据释放保护；

支持内存受控共享，兼容现有的内存动态变更机制。

**2.1.2 性能要求**

[列出合同及需求规格说明书中对本包交付物的性能要求。]  
虚拟机安全套件系统性能损耗<=10%

**2.1.3 输入输出要求**

[列出合同及需求规格说明书中对本包交付物的输入输出支持要求，例如：能够输入进行处理的数字内容格式、输出的许可证格等。]

无

**2.1.4 其它专门要求**

[列出合同及需求规格说明书中对本包交付物的其它专门要求，例如：可扩展性、可集成性等。]

无

**2.2 运行环境**

[本节对本包交付物的运行环境进行规定。]

**2.2.1 硬件环境**

[列出运行本包交付物所需要的硬件设备，说明其用途及配置要求等情况。]

虚拟机安全套件的运行依赖于国产硬件基础环境，主要针对两个硬件平台64位ARMv8架构和x86架构，用于构建涉密云的计算、控制节点。计算节点的服务器平台的CPU硬件需要支持虚拟化功能，支持一台服务器上运行多台虚拟机。控制节点用于运行涉密云管理软件，全局管理和调度涉密云平台的资源池。

**2.2.2 软件环境**

[列出运行本包交付物所需要的支持软件，包括要用到的操作系统、数据库管理软件、中间件软件等，说明其用途及版本要求等情况。]

虚拟化软件套件依赖于虚拟化系统，因此其软件环境主要包括Hypervisor软件、HOS和GOS系统和云平台管理软件。

KVM Hypervisor：KVM Hypervisor用于对物理资源进行虚拟，为VM的运行提供CPU、内存和I/O设备支持。针对ARM平台，包含KVM的Linux版本要求在3.9以上，而对于x86平台，包含KVM的Linux版本要求在3.18以上。

Libvirt：虚拟化API，在涉密云平台中结合云平台管理软件提供虚拟资源管理、VM调度和管理等，版本要求在3.0.0以上。

Qemu：Qemu用于和KVM结合提供用户层的接口，同时提供I/O设备的虚拟化功能，模拟VM的I/O操作，版本要求在2.6以上。

GOS：运行于虚拟化平台中的客户虚拟机，用于运行客户的应用负载，GOS要求是Linux发行版，Linux内核版本在2.6以上。

虚拟机管理软件：虚拟机管理软件OpenStack用于全局管理云平台资源池，处理用户请求和负责负载均衡调度，OpenStack版本要求在～～

**2.3 基本设计概念**

[本节说明本包交付物的基本设计概念和处理流程，尽量使用图表的形式。]

**2.3.1 技术架构**

[清晰描述实现本包交付物的整体技术架构，建议在文字描述的同时使用图表。]

**2.3.2 业务架构及流程**

[清晰描述本包交付物的基本业务架构，及其中的业务处理流程，建议在文字描述的同时使用图表。]

**2.3.3 设计约束**

**2.4 详细系统架构**

[用一览表及框图的形式说明本包交付物的系统模块划分，扼要说明每个系统模块的功能，分层次地给出模块之间的控制与被控制关系。]

[系统划分单元的名称建议按照自上而下的下列层次：本包交付物——系统/平台/工具等交付物——子系统——功能模块……。例如对于包8，可能就对应于：包8交付物——互联网出版版权保护系统——集成分发系统——授权模块……。其中，子系统、功能模块等划分单位在本概要设计中统称为“系统模块”。]



|  |  |  |  |
| --- | --- | --- | --- |
| 系统 | 子系统 | 模块 | 功能 |
| 虚拟机  安全套件系统 | 虚拟机地址空间隔离 | 地址隔离空间创建和空间安全切换 | 实现安全可信执行环境 |
| 虚拟机与宿主机安全切换 | 虚拟机上下文切换模块和虚拟机退出重定向模块 | 保证虚拟机运行时安全，阻止控制流攻击，阻止恶意篡改系统关键数据 |
| 虚拟机地址映射监控 | 多重映射监控模块和双映射监控模块 | 阻止多重映射和双映射攻击，阻止用户敏感信息泄露 |

**2.5 功能需求与系统模块的关系**

[本节说明需求规格说明书中各项功能需求的实现同各个系统模块的分配关系，建议使用下表描述。]

|  |  |  |
| --- | --- | --- |
| 功能需求 | 子系统 | 系统模块 |
| 支持内存分区域存储，物理的隔离敏感数据不被篡改； | 虚拟机地址空间隔离、虚拟机与宿主机安全切换模块 | 隔离空间创建和空间安全切换门 |
| 支持内存动态分配标记、追踪和内存数据释放保护； | 虚拟机地址映射监控模块 | 内存标记与多重映射攻击防护模块 |
| 支持内存受控共享，兼容现有的内存动态变更机制。 | 虚拟机地址映射监控模块 | 内存共享接口设定 |

**2.6 尚未解决的问题**

[本节说明在概要设计过程中尚未解决而设计者认为在系统完成之前必须解决的各个问题。]

目前为止，基本解决虚拟机之间的隔离、虚拟机内系统安全防护和云平台的虚拟机入侵检测监控。

**2.7 开发工具**

[本节列出开发本包交付物所需的工具。]

虚拟机安全增强套件的开发依赖于Linux环境，需要的开发工具如下：

ARM模拟器、Qemu模拟器、DS-5调试器和GCC编译器。

**2.8 遵循标准的设计**

**2.9 本课题与项目其他课题交互**

**3. 关键技术方案**

[在本概要设计说明书的这一章中，应对实现本包交付物所需突破的关键技术，给出较为完整的技术解决方案。]

**3.1 <关键技术1-虚拟机安全套件>**

[第1项关键技术的技术方案，包括问题描述和解决方案。除文字说明外，请使用**图表**说明该技术方案，保证清晰、完整、正确、可行。]

针对涉密云虚拟机安全套件的研究，主要解决虚拟机的安全隔离和虚拟机内部的系统和应用的安全，并结合云虚拟机保护系统实现虚拟机从启动到结束的全生命周期的保护。对此，虚拟机安全套件主要突破的关键技术是虚拟机地址空间的隔离、虚拟机内存映射的监控、虚拟机与宿主机上下文之间切换时的寄存器信息保护和控制数据正确性验证、虚拟机自身系统完整性的验证和rootkit等安全威胁的入侵检测等技术。

对于虚拟机地址空间的隔离，我们采取物理内存页标记、跟踪技术。在云虚拟化平台中，对物理内存的分配、回收以及映射信息进行记录，防止虚拟机的内存被重映射和多重映射。基于标记、追踪实现虚拟机的安全空间隔离。

对于虚拟机的上下文保护，则采取虚拟机退出重定向机制，将虚拟机退出事件的处理交由安全部件处理，然后在进行上下文保护之后，再对HOS的信息进行恢复，从而保证VM的数据不被泄漏，并且能验证VM的控制数据结构的正确性，比如VM的EPT页表基地址、VM的页表基地址和VM退出事件的监控等。

对于虚拟机内部的系统和软件的防护，主要采取免代理的监测和监控机制。对于监测我们采取云平台与入侵检测系统结合，在全数据中心提供一套全局的入侵检测软件，能够对虚拟化平台以及其上运行的VM进行检测，并对异常事件进行处理。而对于虚拟机的内部软件防护，主要解决VM内部系统的完整性实施监控和动态控制，保证GOS的完整性和应用程序之间的隔离。

虚拟机安全套件的整体解决方案如下图所示：



**4. 子系统设计**

[本章应提供本包交付物各个子系统的设计情况。]

**4.1 高强度隔离网关（信工所耿魁）**

[第1项子系统，从技术架构和实现流程等方面描述该子系统的设计。除文字说明外，请使用**图表**说明该子系统，保证清晰、完整、正确、可行。]

**4.1.1 技术及系统架构**

[描述本项子系统的技术架构，包括其功能模块划分连接、各模块功能任务的分配等内容，建议除文字说明外，请使用**图表**。]

**4.1.2界面设计**

[较为详尽地对本包交付物的用户使用界面设计进行描述。]

**<界面1-xxxx>**

[描述第1套操作界面，包括其样式、包含的要素、实现的功能等内容。如果需要，可继续分节，设多级目录详细描述。]

**4.1.3接口设计**

[对本包交付物的外部、内容及其它接口的设计情况进行描述。相对于需求规格说明书中的接口需求，这里应是设计后的具体实现。]

**外部接口**

[说明本包交付物同外界的所有接口的安排包括软件与硬件之间的接口、本系统与各支持系统之间的接口关系。如果需要，可分节设多级目录详细描述。]

**内部接口**

[说明本包交付物之内的各个系统模块之间的接口的安排。如果需要，可分节设多级目录详细描述。]

**4.1.4运行设计**

[对本包交付物实际运行过程的设计进行描述，包括运行模块组合、对运行的控制，以及运行时占用资源的安排等。]

**1 运行模块组合**

[说明对系统施加不同的外界运行控制时所引起的各种不同的运行模块组合，说明每种运行所历经的内部模块的支持软件。]

**2 存储设计**

**3 运行控制**

[说明每一种外界的运行控制的方式方法和操作步骤。]

**4 运行时间**

[说明每种运行模块组合将占用各种资源的时间。]

**4.1.5系统数据结构设计**

[本章对本包交付物中涉及的各式数据结构、数据库，以及其它数据的设计进行清晰、完整的描述。]

**1 主要数据结构设计**

[本节对本包交付物中所需主要数据结构的设计进行描述。本节的数据结构设计可能只是一个初步设计，随着研发的进一步深入，相应的数据结构会不断的调整。注意，对每个数据结构，除了给出中文表意的名称外，还应给出程序中的命名]

**2 数据结构与系统模块的关系**

[本节说明上述数据结构同各个系统模块的关系，建议使用下表描述。]

**3 数据库设计**

[本节对本包交付物所用到的数据库设计进行描述，包括数据库的表结构、实体关系图等内容。]

**4.1.6系统错误处理设计**

[本章用于描述系统中对于各种错误出现进行处理的有关设计。]

**1 出错信息**

[用一览表的方式说明每种可能的出错或故障情况出现时，系统输出信息的形式、含意及处理方法。]

**2 补救措施**

[说明故障出现后可能采取的变通措施。包括：

a.后备技术（说明准备采用的后备技术，当原始系统数据万一丢失时启用的副本的建立和启动的技术，例如周期性地把磁盘信息记录到磁带上去就是对于磁盘媒体的一种后备技术。）

b.降效技术（说明准备采用的后备技术，使用另一个效率稍低的系统或方法来求得所需结果的某些部分，例如一个自动系统的降效技术可以是手工操作和数据的人工记录。）

c.恢复及再启动技术（说明将使用的恢复再启动技术，使软件从故障点恢复执行或使软件从头开始重新运行的方法。）]

**3 系统维护设计**

[本节说明为了系统维护的方便而在本包交付物内部设计中作出的安排，包括在程序中专门安排用于系统的检查与维护的检测点和专用模块。]

**4.1.7安全性设计**

**1 概述**

**1-1 设计原则**

**1-2 设计思路**

**2 xxxx的安全设计**

【具体系统在设计】

**4.1.8性能设计**

**1 性能指标描述**

**2 性能指标分析**

**3 性能指标运行环境**

**4.1.9附录**

[本章提供与系统开发有关的详细、专门的资料。如果包含附录，应明确指出是否将附录当作概要设计的一部分。]

**4.2 云存储安全访问中间件（信工所游瑞邦）**

[第1项子系统，从技术架构和实现流程等方面描述该子系统的设计。除文字说明外，请使用**图表**说明该子系统，保证清晰、完整、正确、可行。]

**4.2.1 技术及系统架构**

[描述本项子系统的技术架构，包括其功能模块划分连接、各模块功能任务的分配等内容，建议除文字说明外，请使用**图表**。]

**4.2.2界面设计**

[较为详尽地对本包交付物的用户使用界面设计进行描述。]

**<界面1-xxxx>**

[描述第1套操作界面，包括其样式、包含的要素、实现的功能等内容。如果需要，可继续分节，设多级目录详细描述。]

**4.2.3接口设计**

[对本包交付物的外部、内容及其它接口的设计情况进行描述。相对于需求规格说明书中的接口需求，这里应是设计后的具体实现。]

**外部接口**

[说明本包交付物同外界的所有接口的安排包括软件与硬件之间的接口、本系统与各支持系统之间的接口关系。如果需要，可分节设多级目录详细描述。]

**内部接口**

[说明本包交付物之内的各个系统模块之间的接口的安排。如果需要，可分节设多级目录详细描述。]

**4.2.4运行设计**

[对本包交付物实际运行过程的设计进行描述，包括运行模块组合、对运行的控制，以及运行时占用资源的安排等。]

**1 运行模块组合**

[说明对系统施加不同的外界运行控制时所引起的各种不同的运行模块组合，说明每种运行所历经的内部模块的支持软件。]

**2 存储设计**

**3 运行控制**

[说明每一种外界的运行控制的方式方法和操作步骤。]

**4 运行时间**

[说明每种运行模块组合将占用各种资源的时间。]

**4.2.5系统数据结构设计**

[本章对本包交付物中涉及的各式数据结构、数据库，以及其它数据的设计进行清晰、完整的描述。]

**1 主要数据结构设计**

[本节对本包交付物中所需主要数据结构的设计进行描述。本节的数据结构设计可能只是一个初步设计，随着研发的进一步深入，相应的数据结构会不断的调整。注意，对每个数据结构，除了给出中文表意的名称外，还应给出程序中的命名]

**2 数据结构与系统模块的关系**

[本节说明上述数据结构同各个系统模块的关系，建议使用下表描述。]

**3 数据库设计**

[本节对本包交付物所用到的数据库设计进行描述，包括数据库的表结构、实体关系图等内容。]

**4.2.6系统错误处理设计**

[本章用于描述系统中对于各种错误出现进行处理的有关设计。]

**1 出错信息**

[用一览表的方式说明每种可能的出错或故障情况出现时，系统输出信息的形式、含意及处理方法。]

**2 补救措施**

[说明故障出现后可能采取的变通措施。包括：

a.后备技术（说明准备采用的后备技术，当原始系统数据万一丢失时启用的副本的建立和启动的技术，例如周期性地把磁盘信息记录到磁带上去就是对于磁盘媒体的一种后备技术。）

b.降效技术（说明准备采用的后备技术，使用另一个效率稍低的系统或方法来求得所需结果的某些部分，例如一个自动系统的降效技术可以是手工操作和数据的人工记录。）

c.恢复及再启动技术（说明将使用的恢复再启动技术，使软件从故障点恢复执行或使软件从头开始重新运行的方法。）]

**3 系统维护设计**

[本节说明为了系统维护的方便而在本包交付物内部设计中作出的安排，包括在程序中专门安排用于系统的检查与维护的检测点和专用模块。]

**4.2.7安全性设计**

**1 概述**

**1-1 设计原则**

**1-2 设计思路**

**2 xxxx的安全设计**

【具体系统在设计】

**4.2.8性能设计**

**1 性能指标描述**

**2 性能指标分析**

**3 性能指标运行环境**

**4.2.9附录**

[本章提供与系统开发有关的详细、专门的资料。如果包含附录，应明确指出是否将附录当作概要设计的一部分。]

**4.3 电子文件MB检测设备（信工所刘宗斌）**

[第1项子系统，从技术架构和实现流程等方面描述该子系统的设计。除文字说明外，请使用**图表**说明该子系统，保证清晰、完整、正确、可行。]

**4.3.1 技术及系统架构**

[描述本项子系统的技术架构，包括其功能模块划分连接、各模块功能任务的分配等内容，建议除文字说明外，请使用**图表**。]

**4.3.2界面设计**

[较为详尽地对本包交付物的用户使用界面设计进行描述。]

**<界面1-xxxx>**

[描述第1套操作界面，包括其样式、包含的要素、实现的功能等内容。如果需要，可继续分节，设多级目录详细描述。]

**4.3.3接口设计**

[对本包交付物的外部、内容及其它接口的设计情况进行描述。相对于需求规格说明书中的接口需求，这里应是设计后的具体实现。]

**外部接口**

[说明本包交付物同外界的所有接口的安排包括软件与硬件之间的接口、本系统与各支持系统之间的接口关系。如果需要，可分节设多级目录详细描述。]

**内部接口**

[说明本包交付物之内的各个系统模块之间的接口的安排。如果需要，可分节设多级目录详细描述。]

**4.3.4运行设计**

[对本包交付物实际运行过程的设计进行描述，包括运行模块组合、对运行的控制，以及运行时占用资源的安排等。]

**1 运行模块组合**

[说明对系统施加不同的外界运行控制时所引起的各种不同的运行模块组合，说明每种运行所历经的内部模块的支持软件。]

**2 存储设计**

**3 运行控制**

[说明每一种外界的运行控制的方式方法和操作步骤。]

**4 运行时间**

[说明每种运行模块组合将占用各种资源的时间。]

**4.3.5系统数据结构设计**

[本章对本包交付物中涉及的各式数据结构、数据库，以及其它数据的设计进行清晰、完整的描述。]

**1 主要数据结构设计**

[本节对本包交付物中所需主要数据结构的设计进行描述。本节的数据结构设计可能只是一个初步设计，随着研发的进一步深入，相应的数据结构会不断的调整。注意，对每个数据结构，除了给出中文表意的名称外，还应给出程序中的命名]

**2 数据结构与系统模块的关系**

[本节说明上述数据结构同各个系统模块的关系，建议使用下表描述。]

**3 数据库设计**

[本节对本包交付物所用到的数据库设计进行描述，包括数据库的表结构、实体关系图等内容。]

**4.3.6系统错误处理设计**

[本章用于描述系统中对于各种错误出现进行处理的有关设计。]

**1 出错信息**

[用一览表的方式说明每种可能的出错或故障情况出现时，系统输出信息的形式、含意及处理方法。]

**2 补救措施**

[说明故障出现后可能采取的变通措施。包括：

a.后备技术（说明准备采用的后备技术，当原始系统数据万一丢失时启用的副本的建立和启动的技术，例如周期性地把磁盘信息记录到磁带上去就是对于磁盘媒体的一种后备技术。）

b.降效技术（说明准备采用的后备技术，使用另一个效率稍低的系统或方法来求得所需结果的某些部分，例如一个自动系统的降效技术可以是手工操作和数据的人工记录。）

c.恢复及再启动技术（说明将使用的恢复再启动技术，使软件从故障点恢复执行或使软件从头开始重新运行的方法。）]

**3 系统维护设计**

[本节说明为了系统维护的方便而在本包交付物内部设计中作出的安排，包括在程序中专门安排用于系统的检查与维护的检测点和专用模块。]

**4.3.7安全性设计**

**1 概述**

**1-1 设计原则**

**1-2 设计思路**

**2 xxxx的安全设计**

【具体系统在设计】

**4.3.8性能设计**

**1 性能指标描述**

**2 性能指标分析**

**3 性能指标运行环境**

**4.3.9附录**

[本章提供与系统开发有关的详细、专门的资料。如果包含附录，应明确指出是否将附录当作概要设计的一部分。]

**4.4 云虚拟边界防护设备（数据所）**

[第1项子系统，从技术架构和实现流程等方面描述该子系统的设计。除文字说明外，请使用**图表**说明该子系统，保证清晰、完整、正确、可行。]

**4.4.1 技术及系统架构**

[描述本项子系统的技术架构，包括其功能模块划分连接、各模块功能任务的分配等内容，建议除文字说明外，请使用**图表**。]

**4.4.2界面设计**

[较为详尽地对本包交付物的用户使用界面设计进行描述。]

**<界面1-xxxx>**

[描述第1套操作界面，包括其样式、包含的要素、实现的功能等内容。如果需要，可继续分节，设多级目录详细描述。]

**4.4.3接口设计**

[对本包交付物的外部、内容及其它接口的设计情况进行描述。相对于需求规格说明书中的接口需求，这里应是设计后的具体实现。]

**外部接口**

[说明本包交付物同外界的所有接口的安排包括软件与硬件之间的接口、本系统与各支持系统之间的接口关系。如果需要，可分节设多级目录详细描述。]

**内部接口**

[说明本包交付物之内的各个系统模块之间的接口的安排。如果需要，可分节设多级目录详细描述。]

**4.4.4运行设计**

[对本包交付物实际运行过程的设计进行描述，包括运行模块组合、对运行的控制，以及运行时占用资源的安排等。]

**1 运行模块组合**

[说明对系统施加不同的外界运行控制时所引起的各种不同的运行模块组合，说明每种运行所历经的内部模块的支持软件。]

**2 存储设计**

**3 运行控制**

[说明每一种外界的运行控制的方式方法和操作步骤。]

**4 运行时间**

[说明每种运行模块组合将占用各种资源的时间。]

**4.4.5系统数据结构设计**

[本章对本包交付物中涉及的各式数据结构、数据库，以及其它数据的设计进行清晰、完整的描述。]

**1 主要数据结构设计**

[本节对本包交付物中所需主要数据结构的设计进行描述。本节的数据结构设计可能只是一个初步设计，随着研发的进一步深入，相应的数据结构会不断的调整。注意，对每个数据结构，除了给出中文表意的名称外，还应给出程序中的命名]

**2 数据结构与系统模块的关系**

[本节说明上述数据结构同各个系统模块的关系，建议使用下表描述。]

**3 数据库设计**

[本节对本包交付物所用到的数据库设计进行描述，包括数据库的表结构、实体关系图等内容。]

**4.4.6系统错误处理设计**

[本章用于描述系统中对于各种错误出现进行处理的有关设计。]

**1 出错信息**

[用一览表的方式说明每种可能的出错或故障情况出现时，系统输出信息的形式、含意及处理方法。]

**2 补救措施**

[说明故障出现后可能采取的变通措施。包括：

a.后备技术（说明准备采用的后备技术，当原始系统数据万一丢失时启用的副本的建立和启动的技术，例如周期性地把磁盘信息记录到磁带上去就是对于磁盘媒体的一种后备技术。）

b.降效技术（说明准备采用的后备技术，使用另一个效率稍低的系统或方法来求得所需结果的某些部分，例如一个自动系统的降效技术可以是手工操作和数据的人工记录。）

c.恢复及再启动技术（说明将使用的恢复再启动技术，使软件从故障点恢复执行或使软件从头开始重新运行的方法。）]

**3 系统维护设计**

[本节说明为了系统维护的方便而在本包交付物内部设计中作出的安排，包括在程序中专门安排用于系统的检查与维护的检测点和专用模块。]

**4.4.7安全性设计**

**1 概述**

**1-1 设计原则**

**1-2 设计思路**

**2 xxxx的安全设计**

【具体系统在设计】

**4.4.8性能设计**

**1 性能指标描述**

**2 性能指标分析**

**3 性能指标运行环境**

**4.4.9附录**

[本章提供与系统开发有关的详细、专门的资料。如果包含附录，应明确指出是否将附录当作概要设计的一部分。]

**4.5 SM云虚拟化桌面应用系统（信工所杨露露）**

[第1项子系统，从技术架构和实现流程等方面描述该子系统的设计。除文字说明外，请使用**图表**说明该子系统，保证清晰、完整、正确、可行。]

**4.5.1 技术及系统架构**

[描述本项子系统的技术架构，包括其功能模块划分连接、各模块功能任务的分配等内容，建议除文字说明外，请使用**图表**。]

**4.5.2界面设计**

[较为详尽地对本包交付物的用户使用界面设计进行描述。]

**<界面1-xxxx>**

[描述第1套操作界面，包括其样式、包含的要素、实现的功能等内容。如果需要，可继续分节，设多级目录详细描述。]

**4.5.3接口设计**

[对本包交付物的外部、内容及其它接口的设计情况进行描述。相对于需求规格说明书中的接口需求，这里应是设计后的具体实现。]

**外部接口**

[说明本包交付物同外界的所有接口的安排包括软件与硬件之间的接口、本系统与各支持系统之间的接口关系。如果需要，可分节设多级目录详细描述。]

**内部接口**

[说明本包交付物之内的各个系统模块之间的接口的安排。如果需要，可分节设多级目录详细描述。]

**4.5.4运行设计**

[对本包交付物实际运行过程的设计进行描述，包括运行模块组合、对运行的控制，以及运行时占用资源的安排等。]

**1 运行模块组合**

[说明对系统施加不同的外界运行控制时所引起的各种不同的运行模块组合，说明每种运行所历经的内部模块的支持软件。]

**2 存储设计**

**3 运行控制**

[说明每一种外界的运行控制的方式方法和操作步骤。]

**4 运行时间**

[说明每种运行模块组合将占用各种资源的时间。]

**4.5.5系统数据结构设计**

[本章对本包交付物中涉及的各式数据结构、数据库，以及其它数据的设计进行清晰、完整的描述。]

**1 主要数据结构设计**

[本节对本包交付物中所需主要数据结构的设计进行描述。本节的数据结构设计可能只是一个初步设计，随着研发的进一步深入，相应的数据结构会不断的调整。注意，对每个数据结构，除了给出中文表意的名称外，还应给出程序中的命名]

**2 数据结构与系统模块的关系**

[本节说明上述数据结构同各个系统模块的关系，建议使用下表描述。]

**3 数据库设计**

[本节对本包交付物所用到的数据库设计进行描述，包括数据库的表结构、实体关系图等内容。]

**4.5.6系统错误处理设计**

[本章用于描述系统中对于各种错误出现进行处理的有关设计。]

**1 出错信息**

[用一览表的方式说明每种可能的出错或故障情况出现时，系统输出信息的形式、含意及处理方法。]

**2 补救措施**

[说明故障出现后可能采取的变通措施。包括：

a.后备技术（说明准备采用的后备技术，当原始系统数据万一丢失时启用的副本的建立和启动的技术，例如周期性地把磁盘信息记录到磁带上去就是对于磁盘媒体的一种后备技术。）

b.降效技术（说明准备采用的后备技术，使用另一个效率稍低的系统或方法来求得所需结果的某些部分，例如一个自动系统的降效技术可以是手工操作和数据的人工记录。）

c.恢复及再启动技术（说明将使用的恢复再启动技术，使软件从故障点恢复执行或使软件从头开始重新运行的方法。）]

**3 系统维护设计**

[本节说明为了系统维护的方便而在本包交付物内部设计中作出的安排，包括在程序中专门安排用于系统的检查与维护的检测点和专用模块。]

**4.5.7安全性设计**

**1 概述**

**1-1 设计原则**

**1-2 设计思路**

**2 xxxx的安全设计**

【具体系统在设计】

**4.5.8性能设计**

**1 性能指标描述**

**2 性能指标分析**

**3 性能指标运行环境**

**4.5.9附录**

[本章提供与系统开发有关的详细、专门的资料。如果包含附录，应明确指出是否将附录当作概要设计的一部分。]

**4.6 云虚拟机保护系统（30所）**

[第1项子系统，从技术架构和实现流程等方面描述该子系统的设计。除文字说明外，请使用**图表**说明该子系统，保证清晰、完整、正确、可行。]

**4.6.1 技术及系统架构**

[描述本项子系统的技术架构，包括其功能模块划分连接、各模块功能任务的分配等内容，建议除文字说明外，请使用**图表**。]

**4.6.2界面设计**

[较为详尽地对本包交付物的用户使用界面设计进行描述。]

**<界面1-xxxx>**

[描述第1套操作界面，包括其样式、包含的要素、实现的功能等内容。如果需要，可继续分节，设多级目录详细描述。]

**4.6.3接口设计**

[对本包交付物的外部、内容及其它接口的设计情况进行描述。相对于需求规格说明书中的接口需求，这里应是设计后的具体实现。]

**外部接口**

[说明本包交付物同外界的所有接口的安排包括软件与硬件之间的接口、本系统与各支持系统之间的接口关系。如果需要，可分节设多级目录详细描述。]

**内部接口**

[说明本包交付物之内的各个系统模块之间的接口的安排。如果需要，可分节设多级目录详细描述。]

**4.6.4运行设计**

[对本包交付物实际运行过程的设计进行描述，包括运行模块组合、对运行的控制，以及运行时占用资源的安排等。]

**1 运行模块组合**

[说明对系统施加不同的外界运行控制时所引起的各种不同的运行模块组合，说明每种运行所历经的内部模块的支持软件。]

**2 存储设计**

**3 运行控制**

[说明每一种外界的运行控制的方式方法和操作步骤。]

**4 运行时间**

[说明每种运行模块组合将占用各种资源的时间。]

**4.6.5系统数据结构设计**

[本章对本包交付物中涉及的各式数据结构、数据库，以及其它数据的设计进行清晰、完整的描述。]

**1 主要数据结构设计**

[本节对本包交付物中所需主要数据结构的设计进行描述。本节的数据结构设计可能只是一个初步设计，随着研发的进一步深入，相应的数据结构会不断的调整。注意，对每个数据结构，除了给出中文表意的名称外，还应给出程序中的命名]

**2 数据结构与系统模块的关系**

[本节说明上述数据结构同各个系统模块的关系，建议使用下表描述。]

**3 数据库设计**

[本节对本包交付物所用到的数据库设计进行描述，包括数据库的表结构、实体关系图等内容。]

**4.6.6系统错误处理设计**

[本章用于描述系统中对于各种错误出现进行处理的有关设计。]

**1 出错信息**

[用一览表的方式说明每种可能的出错或故障情况出现时，系统输出信息的形式、含意及处理方法。]

**2 补救措施**

[说明故障出现后可能采取的变通措施。包括：

a.后备技术（说明准备采用的后备技术，当原始系统数据万一丢失时启用的副本的建立和启动的技术，例如周期性地把磁盘信息记录到磁带上去就是对于磁盘媒体的一种后备技术。）

b.降效技术（说明准备采用的后备技术，使用另一个效率稍低的系统或方法来求得所需结果的某些部分，例如一个自动系统的降效技术可以是手工操作和数据的人工记录。）

c.恢复及再启动技术（说明将使用的恢复再启动技术，使软件从故障点恢复执行或使软件从头开始重新运行的方法。）]

**3 系统维护设计**

[本节说明为了系统维护的方便而在本包交付物内部设计中作出的安排，包括在程序中专门安排用于系统的检查与维护的检测点和专用模块。]

**4.6.7安全性设计**

**1 概述**

**1-1 设计原则**

**1-2 设计思路**

**2 xxxx的安全设计**

【具体系统在设计】

**4.6.8性能设计**

**1 性能指标描述**

**2 性能指标分析**

**3 性能指标运行环境**

**4.6.9附录**

**4.7 虚拟机安全套件（信工所朱民）**

**4.7.1 技术及系统架构**

[描述本项子系统的技术架构，包括其功能模块划分连接、各模块功能任务的分配等内容，建议除文字说明外，请使用**图表**。]

虚拟机安全套件负责保护和监测整个涉密云平台内所有虚拟机的安全，对内负责保护虚拟机的隔离性和敏感数据的安全性；对外向云平台管理人员提供监测信息和异常事件视图，使得管理员可以方便的发现和处理异常虚拟机。虚拟机安全套件部署在计算节点和入侵检测系统节点上，是一个与Hypervisor和主机入侵检测系统兼容的模块化部件。

虚拟机安全套件包括的模块，如下图所示：



虚拟机安全套件共分为四个子模块：虚拟机地址空间隔离、VM与Hypervisor之间上下文安全切换、虚拟机I/O数据保护、虚拟机内部敏感应用隔离以及免代理的全数据中心基于虚拟机的入侵检测系统。

虚拟机地址空间隔离：虚拟机之间的空间隔离是首要前提，只有虚拟机之间被隔离开才能为其它模块提供安全基础。地址空间隔离采用虚拟机页标记、追踪的技术。在设计框架中，我们将Hypervisor对虚拟机的物理内存管理功能给移到安全套件中，这样只有Hypervisor请求安全套件才能完成虚拟机的内存管理操作。在设计中的主要难题是如何避免内存管理机制被绕过。为解决这个问题，我们将虚拟机的EPT表放置在隔离空间中，并利用权限机制使得Hypervisor对其只有读访问权限。当Hypervisor修改EPT时就会触发异常，陷入到安全套件中，然后安全套件模拟该操作，完成虚拟机的内存分配，并对内存进行标记。此技术可以保证虚拟机之间不会出现多重映射和跨域访问。此外，当Hypervisor攻击虚拟机时，该技术结合日志功能还能为用户提供验证信息。



虚拟机与虚拟机监控器安全切换：在虚拟机退出时，虚拟机的控制信息和寄存器信息仍然保留在其上下文中，并且由Hypervisor辅助完整环境切换。Hypervisor通过自身的高特权可以获取虚拟机的信息，或导致虚拟机的控制流被篡改。因此，在虚拟机与Hypervisor上下文切换时，需提供安全切换，保证虚拟机的信息和控制信息的泄漏和篡改。我们采取的措施是，在虚拟机退出时，将退出异常处理的接口定向到虚拟机安全套件中，然后由虚拟机安全套件对虚拟机的信息进行保护，然后恢复Hypervisor的上下文信息。需要注意的是，Hypervisor在处理虚拟机退出事件是需要用到一些虚拟机的寄存器信息，因此我们根据不同的虚拟机退出原因，将不同的辅助信息暴露给Hypervisor，让其能够模拟虚拟机的退出。



虚拟机I/O数据防护：虚拟机I/O数据保护的方式是对计算节点所运行的所有虚拟机的I/O操作进行截获，然后分析其参数，最终根据参数信息对虚拟机的I/O缓存数据进行哈希完整性验证和加/解密机密性保护。I/O数据防护技术依赖于加密机制，我们采用的是数据块独立的种子加密机制，该机制利用虚拟机自身的密钥对种子进行加密产生Pad，然后将Pad与真实数据进行异或运算，从而实现数据加密。该机制的优点是并发能力强，性能损耗小。而对于哈希树，我们不采用全盘验证，而是需要巧妙的设计哈希树，只对密钥和哈希值进行保护。这样则能够减少保护的对象，降低性能损耗。



基于虚拟机的免代理入侵检测系统：上述技术只能保证虚拟机之间的隔离和安全，但不能保证虚拟机内部的异常行为和入侵事件。因此，我们提供基于虚拟机的免代理入侵检测系统。入侵检测系统是防火墙的补充方案，在虚拟机安全套件中该技术利用虚拟机自省技术收集虚拟机内部的若干关键点或数据结构信息，并对其进行分析，从而发现虚拟机系统中是否有违反安全策略的行为和被攻击的迹象。由于我们采用的是免代理方式，并且提供了虚拟机地址空间隔离，这对虚拟机自省技术提出了新的挑战。我们首要解决的问题是如何将虚拟机的数据结构得到。针对此问题，我们的解决方法是在虚拟机安全套件空间中，将虚拟机的内核空间进行影射（overshadow），然后在Hypervisor层则可实现。将得到的虚拟机重要数据结构信息、日志信息进行分析，则能够检测到攻击者的行为，并作出响应。



**4.7.2界面设计**

[较为详尽地对本包交付物的用户使用界面设计进行描述。]

[描述第1套操作界面，包括其样式、包含的要素、实现的功能等内容。如果需要，可继续分节，设多级目录详细描述。]

**<界面1-功能模块的配置>**

功能配置界面是对所支持的功能进行配置，在虚拟机安全套件中我们使用配置文件的方式对各个功能模块的使能进行使能配置。配置页面包含六个配置选项，在虚拟化系统启动阶段根据配置信息对功能模块进行开启。

**<界面2-异常处理>**

异常处理界面是对虚拟机防护的异常行为进行分析，并交给入侵检测系统进行处理，最后反馈给系统管理人员。

**4.7.3接口设计**

[对本包交付物的外部、内容及其它接口的设计情况进行描述。相对于需求规格说明书中的接口需求，这里应是设计后的具体实现。]

**外部接口**

[说明本包交付物同外界的所有接口的安排包括软件与硬件之间的接口、本系统与各支持系统之间的接口关系。如果需要，可分节设多级目录详细描述。]

虚拟机安全套件依赖于云平台管理软件和基于主机的入侵检测系统提供虚拟机的运行时防护，并协同云虚拟机保护系统提供虚拟机全生命周期保护。对此，需要提供的接口如下：

* Create\_VM-创建虚拟机：根据用户请求，云平台管理软件将虚拟机的配置参数传递给计算节点的Hypervisor，Hypervisor根据参数信息，给VM分配资源，然后为用户生成秘钥等元数据，并初始化VM。
* Start\_VM-启动虚拟机：根据用户请求，云平台管理软件将虚拟机的配置参数传递给计算节点的Hypervisor，Hypervisor根据参数信息，获取VM所需的资源，获取其对应的秘钥信息用于虚拟机的I/O数据保护，在虚拟机安全启动后，返回验证信息给用户，用于用户远程认证。
* Close\_VM-关闭虚拟机：根据用户的请求，云平台管理软件将虚拟机的资源进行回收，并对用户的数据进行销毁，以免信息泄露。
* Get\_VM\_Info-获取虚拟机的信息：根据入侵检测系统的需求，获取虚拟机的运行状态信息、重要接口信息、文件属性等。
* Restart\_VM-重启虚拟机：此功能相当于电脑主机中的重启键功能。
* Suspen\_VM-挂起虚拟机：当虚拟机在运行状态时，根据用户的请求将虚拟机挂起，让虚拟机滞留在当前的状态。这类似于睡眠功能，对VM不再进行调度，保留现在的运行态。当请求恢复功能时则可以重新使用虚拟机。
* Resume\_VM-恢复虚拟机：在虚拟机处于挂起状态时，根据用户的请求将虚拟机恢复到运行状态，开始调度虚拟机的工作负载。
* Delete\_VM-删除虚拟机：根据用户请求，对用户不再使用的虚拟机进行删除，删除虚拟机的磁盘数据，秘钥信息，虚拟机的元数据和配置信息。
* Remote\_ Attest-远程证明：在虚拟机启动完成后或者虚拟机运行过程中，用户发送验证请求信息，并携带一个nonce，计算节点的可信认证模块将VM的哈希值、异常事件以及nonce进行哈希计算，然后将计算结果发送给用户。
* Find\_Hidden\_Modules-发现隐藏模块：利用虚拟机自省技术扫描虚拟机的内部的模块结构，然后根据结构中的数据与虚拟机中展现出的模块对比，发现隐藏模块。
* Discover\_ARP\_Attacks-发现ARP欺骗：根据虚拟机自省得到的数据发现ARP包中的信息，发现ARP攻击。
* Detect\_NIC\_Promiscuous\_Mode-发现网卡混合模式：根据虚拟机自省得到的网络结果体，发现网卡是否隐藏开启混合模式。

**内部接口**

[说明本包交付物之内的各个系统模块之间的接口的安排。如果需要，可分节设多级目录详细描述。]

* Create\_VM\_Metadata-虚拟机元数据创建：当虚拟机进行创建时，调用该接口为虚拟机创建私有信息。
* Delete\_VM\_Metadata-虚拟机元数据删除：当虚拟机删除时，调用该接口为虚拟机销毁其私有信息。
* Alloc\_Memory\_to\_VM-内存分配：在地址空间隔离时，Hypervisor调用该接口让地址空间隔离模块为虚拟机分配内存，并完成EPT页表项的添加操作。
* Free\_Memory\_from\_VM-内存回收：在地址空间隔离时，Hypervisor调用该接口让地址空间隔离模块为虚拟机回收内存，并完成EPT页表项的删减操作。
* Share\_Memory\_among\_VMs-内存共享：当不同虚拟机需共享同一物理内存时，通过该接口，为不同虚拟机提供内存共享功能。
* Switch\_Host\_Guest- Hypervisor和VM之间上下文切换：
* Swap\_Page\_out -页面换出到磁盘：在地址空间隔离启用时，Hypervisor需要将虚拟机的数据置换到磁盘时，调用此操作，保证数据的安全性和正确性，并记录相关信息。
* Swap\_Page\_in -页面换入到内存：与Swap\_Page\_out相反，当虚拟机再次访问该页面时，Hypervisor将数据重新写入内存中，并对其进行安全和正确性检查。
* Parse\_VM\_IO\_Operation-虚拟机I/O操作解析：在虚拟机与Hypervisor间上下文切换时，通过该接口可以分析是否是虚拟机I/O操作，然后采取I/O数据保护措施。
* Protect\_VM\_IO\_Data-虚拟机I/O数据保护：如果经分析虚拟机退出原因是因为I/O操作，那对虚拟机的I/O缓存进行机密性和完整性保护。

**4.7.4运行设计**

[对本包交付物实际运行过程的设计进行描述，包括运行模块组合、对运行的控制，以及运行时占用资源的安排等。]

**1 运行模块组合**

[说明对系统施加不同的外界运行控制时所引起的各种不同的运行模块组合，说明每种运行所历经的内部模块的支持软件。]

虚拟机安全套件包含的四个模块中模拟和模块必须配套组合，而剩下的两个模块可以与它们两个模块进行任意组合。这是因为只有前两个模块的组合才能为虚拟机安全提供全面的基础保障。不同的组合方式，对系统的防护强度不同，主要有三种组合方式：、、。不同组合运行所需要的支持软件需求是：含有模块的组合除了需要云平台管理系统虚拟化Hypervisor之外，还需要基于宿主机的入侵检测系统。

**2 存储设计**

**无（本地或远程）**

**3 运行控制**

[说明每一种外界的运行控制的方式方法和操作步骤。]

由于虚拟机安全套件是在系统编译阶段进行配置，系统启动后自动生效。因此，虚拟机安全套件手外界的控制有两个阶段：系统生产时通过配置编译选项、入侵检测系统检测到异常而不能自动处理时需要管理员处理。

**4 运行时间**

[说明每种运行模块组合将占用各种资源的时间。]

虚拟机安全套件是Hypervisor和入侵检测系统的组成部分，运行时间与资源的需求和工作负载有关。虚拟机安全套件与常规的Hypervisor运行无明显区别，只是在关键操作时进行了数据记录和验证。因此，虚拟机安全套件的运行时间是可以忽略不计的。

**4.7.5系统数据结构设计**

本章对虚拟机安全套件设计中所需要的各式数据结构的设计及其之间的关系进行描述。该部分内容包括结构设计、数据结构与模块之间的关系以及数据库设计。

**1 主要数据结构设计**

[本节对本包交付物中所需主要数据结构的设计进行描述。本节的数据结构设计可能只是一个初步设计，随着研发的进一步深入，相应的数据结构会不断的调整。注意，对每个数据结构，除了给出中文表意的名称外，还应给出程序中的命名]

* 用户个人信息——用于记录用户的个人信息，是用户在云计算平台中的凭证：User\_Priv\_Info（u\_name, u\_pwd, u\_email, global\_vmid, vm\_disk）
* 虚拟机信息表——用于记录虚拟机的运行时信息，在虚拟机创建时对其进行初始化：VM\_Table（host\_vmid, vm\_image\_hash, vm\_encrypt\_key, guest\_context\_addr, ept\_base\_addr）；
* 物理页使用情况表——用于记录每个物理页的使用情况，便于对虚拟机的内存进行标记和追踪：Page\_Usage\_Table（pfn, owner\_type, owner\_id, be\_shared, permissions）；
* 虚拟机上下文信息——在虚拟机进行上下文切换时，将虚拟机的上下文信息保存在指定的Hypervisor不能访问的位置：VM\_Context\_Info：（general\_registers, system\_registers, vm\_exit\_control）
* 共享物理页属主信息链表——当物理页是共享时，此数据机构用于记录所有共用这个物理页的属主信息，此时Page\_Usage\_Table中的owner\_id项纪录的是此数据结构的基地址：Page\_Ownership\_Entry：（owner\_type, owner\_id, next\_entry）
* 虚拟机的系统信息——基于虚拟机自省技术获取虚拟机系统的内核重要结构体，根据结构体种类翻译出其含义：VM\_Struct\_Type（os\_type, struct\_type, struct\_virtual\_addr, struct\_size）
* 哈希树——通过树状结构对虚拟机的虚拟磁盘信息的数据进行完整性验证：Hash\_Tree\_Node（node\_key, node\_child[], node\_parent, node\_hash）
* 种子缓存——种子缓存是在Hypervisor中申请一块区域，通过该区域对加密的种子进行预读，减少加密损耗：Seed\_Cache（c\_vmid, c\_size）

**2 数据结构与系统模块的关系**

[本节说明上述数据结构同各个系统模块的关系，建议使用下表描述。]

|  |  |
| --- | --- |
| **系统模块** | **数据结构** |
| 虚拟机地址空间隔离 | User\_Priv\_Info  VM\_Table  Page\_Usage\_Table |
| 虚拟机与虚拟机监控器之间安全切换 | VM\_Context\_Info |
| 虚拟机I/O数据保护 | Hash Tree Node  Seed\_Cache |
| 免代理入侵检测系统 | VM\_Struct\_Type |

**3 数据库设计**

[本节对本包交付物所用到的数据库设计进行描述，包括数据库的表结构、实体关系图等内容。]

由于虚拟机安全套件运行于内存中，重要数据结构都保存在内存中，因此只有与虚拟机相关的元数据结构存储在数据库中，重要数据结构如下所示：

**虚拟机信息表**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 字段含义 | 类型 | 长度 | 默认值 | 允许空 | 主键 | 说明 |
| 1 | vmID | 虚拟机编号 | varchar | 30 |  | 否 | 是 |  |
| 2 | vmName | 虚拟机名称 | varchar | 255 |  |  |  |  |
| 3 | virtIP | IP地址 | varchar | 12 |  | 否 |  |  |
| 4 | virtMAC | MAC地址 | varchar | 12 |  | 否 |  |  |
| 5 | virtStatus | 虚拟机状态 | varchar | 10 |  | 否 |  |  |
| 6 | virtImage | 镜像编号 | int |  |  |  |  |  |
| 7 | virtConfig | 虚拟机配置信息 | int |  |  |  |  |  |

**虚拟机镜像信息表表**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 字段含义 | 类型 | 长度 | 默认值 | 允许空 | 主键 | 说明 |
| 1 | imgID | 镜像编号 | int |  |  | 否 | 是 |  |
| 2 | imgName | 镜像名称 | varchar | 255 |  | 否 |  |  |
| 3 | format | 镜像格式 | varchar | 20 |  | 否 |  |  |
| 4 | imgPath | 镜像绝对路径 | varchar | 255 |  |  |  |  |
| 5 | createTime | 镜像的创建时间 | datetime |  |  |  |  |  |
| 6 | updateDate | 镜像的最近更新时间 | datetime |  |  |  |  |  |
| 7 | imgSize | 镜像空间大小 | int |  |  | 否 |  |  |
| 8 | vmID | 虚拟机编号 | varchar | 30 |  |  |  |  |

**虚拟机配置信息**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 字段含义 | 类型 | 长度 | 默认值 | 允许空 | 主键 | 说明 |
| 1 | vmid | 虚拟机ID | varchar | 30 |  | 否 | 是 |  |
| 2 | vcpu | CPU核数 | int |  | 1 |  |  |  |
| 3 | memory | 内存大小 | varchar | 37 | 128 | 否 |  |  |
| 4 | os\_type | 系统类型 | varchar | 8 |  |  |  |  |
| 5 | os\_version | 系统版本 | varchar | 12 |  |  |  |  |

**4.7.6系统错误处理设计**

[本章用于描述系统中对于各种错误出现进行处理的有关设计。]

**1 出错信息**

[用一览表的方式说明每种可能的出错或故障情况出现时，系统输出信息的形式、含意及处理方法。]

|  |  |  |
| --- | --- | --- |
| **出现的故障** | **输出的信息形式和含义** | **处理方法** |
| 系统崩溃 | Pannic信息，系统死机 | 重启机器 |
| 产生死锁 | 无输出信息，系统停止不动 | 关闭相关任务 |
| 出现非预知异常 | 抛出异常信息 | 人为分析，然后分类处理 |

**2 补救措施**

[说明故障出现后可能采取的变通措施。包括：

对于虚拟机镜像和磁盘镜像的存储采取数据多套备份恢复技术，而对于数据库存储的数据表则在另一个安全位置存储备份副本可防止可能的灾难性数据丢失，可从多种故障中恢复数据，例如：介质故障、用户错误（误删了某个表）、硬件故障（磁盘驱动器损坏）、和自然灾难等。

**3 系统维护设计**

[本节说明为了系统维护的方便而在本包交付物内部设计中作出的安排，包括在程序中专门安排用于系统的检查与维护的检测点和专用模块。]

由于虚拟机安全套件是嵌入到虚拟化系统内部，所以维护方式是当系统出现崩溃时，对系统进行重启。而对于入侵检测系统的维护与常规的系统维护无区别。

**4.7.7安全性设计**

**1 概述**

* 1. **设计原则**

虚拟机安全套件设计原则有两点。第一，安全套件不能被绕过，Hypervisor要想实现某些功能必须要请求安全套件，由安全套件去完成，然后交还给Hypervisor做后续处理。在安全套件处理过程中不会被中断和控制流截获，安全套件的操作具有原子性。第二，安全套件能够抵御攻击，不会被来自Hypervisor或VM的攻击。在Hypervisor调用安全套件中的服务时，需要对传输的参数和调用关系进行检查，防止非法调用。第三、与IDS之间交互的安全。。。。

* 1. **设计思路**

现在涉密网互联互通、规模庞大，机器维护复杂，而云虚拟化技术能够将数据和用户隔离，非常适用于保密环境下的安全防护。但由于虚拟化技术对虚拟机的隔离不彻底，并且个虚拟机的内部防护是各自为政，增加了系统损耗。基于此，我们提供虚拟机安全套件，从外到内提供虚拟机的整体防护。通过突破虚拟机地址空间隔离保证虚拟机间的完全独立，就如运行在独立的硬件资源上；通过突破免代理的虚拟机数据抓取技术，结合现有的入侵检测系统，能够为虚拟机的内部安全威胁提供警示和异常处理机制。在一定程度对现有入侵检测系统进行扩展，从而能够保证虚拟机的内部安全。最后，通过验证平台的技术验证，实现涉密云环境下的虚拟机独立、受控、安全运行。

**2 虚拟机安全套件的安全设计**

【具体系统在设计】

虚拟机安全套件是为了在不可信的云平台中确保虚拟机之间的隔离，然后对虚拟机内部数据进行保护，从而能够全面的对虚拟机实施保护。虚拟机安全套件与Hypervisor处于同一层，但是利用了隔离机制，在同一层实现了两套页表。这样保证了Hypervisor不能够对虚拟机安全套件进行篡改和恶意调用，但是虚拟机安全套件能够对Hypervisor的行为进行监控，从而实施虚拟机隔离，虚拟机上下文保护以及虚拟机内部信息保护等功能。

此外，虚拟机安全套件利用虚拟机自省机制，透明地动态获取虚拟机的内部信息，然后结合入侵检测系统，提供全数据中心的虚拟机系统层的异常事件探测和异常处理。此设计中的虚拟机自省机制运行在独立的地址空间，因此可以安全的抓取虚拟机内的数据，然后利用安全通道将数据传递给入侵检测系统，实现全局免代理虚拟机防御。

**4.7.8性能设计**

**1 性能指标描述**

在虚拟机安全套件支持内存分区域存储，物理的隔离敏感数据不被篡改、内存动态分配标记、追踪和内存数据释放保护以及内存受控共享功能的前提下，系统的各项性能损耗不高于10%，主要包括内存管理、虚拟机切换、文件操作和网络延迟。

**2 性能指标分析**

虚拟机安全套件在系统层次实现，主要针对的是内存管理和虚拟机内部数据结构的获取，这在一定程度上会损耗虚拟机的性能。在我们的设计中，系统损耗主要在两个方面：虚拟机监控器对内存的管理和由于虚拟机自省对虚拟机调度的减少。因此，我们利用现有的系统测试集对系统的整体性能进行测试，测试集涵盖处理器、内存、磁盘I/O、数据库和网络等各个方面。通过这些测试能够对虚拟机安全套件的性能损耗进行全面展现。例如，利用Lmbench测试系统的系统调用、进程上下文切换、文件读写操作、本地通信延迟、TCP以及UDP延迟等测试项。

**3 性能指标运行环境**

利用虚拟化软件栈构建小型的涉密云平台，我们预计利用10台配有Intel E7 32核心的处理器、64G物理内存、百兆-千兆双网卡和512G本地磁盘的服务器作为计算节点，在此计算节点上部署虚拟化系统和虚拟机安全套件。同时在同样配置的2台服务器节点上部署入侵检测系统。在此环境中通过启用不同负载的虚拟机，在此虚拟机内部运行测试工具集，测试系统的功能和性能。我们主要的测试工具集是Lmbench、Bonnie、Apache和MySQL。

**4.7.9附录**

[本章提供与系统开发有关的详细、专门的资料。如果包含附录，应明确指出是否将附录当作概要设计的一部分。]

**4.8 云平台权限管理系统（信工所魏松）**

[第1项子系统，从技术架构和实现流程等方面描述该子系统的设计。除文字说明外，请使用**图表**说明该子系统，保证清晰、完整、正确、可行。]

**4.8.1 技术及系统架构**

[描述本项子系统的技术架构，包括其功能模块划分连接、各模块功能任务的分配等内容，建议除文字说明外，请使用**图表**。]

**4.8.2界面设计**

[较为详尽地对本包交付物的用户使用界面设计进行描述。]

**<界面1-xxxx>**

[描述第1套操作界面，包括其样式、包含的要素、实现的功能等内容。如果需要，可继续分节，设多级目录详细描述。]

**4.8.3接口设计**

[对本包交付物的外部、内容及其它接口的设计情况进行描述。相对于需求规格说明书中的接口需求，这里应是设计后的具体实现。]

**外部接口**

[说明本包交付物同外界的所有接口的安排包括软件与硬件之间的接口、本系统与各支持系统之间的接口关系。如果需要，可分节设多级目录详细描述。]

**内部接口**

[说明本包交付物之内的各个系统模块之间的接口的安排。如果需要，可分节设多级目录详细描述。]

**4.8.4运行设计**

[对本包交付物实际运行过程的设计进行描述，包括运行模块组合、对运行的控制，以及运行时占用资源的安排等。]

**1 运行模块组合**

[说明对系统施加不同的外界运行控制时所引起的各种不同的运行模块组合，说明每种运行所历经的内部模块的支持软件。]

**2 存储设计**

**3 运行控制**

[说明每一种外界的运行控制的方式方法和操作步骤。]

**4 运行时间**

[说明每种运行模块组合将占用各种资源的时间。]

**4.8.5系统数据结构设计**

[本章对本包交付物中涉及的各式数据结构、数据库，以及其它数据的设计进行清晰、完整的描述。]

**1 主要数据结构设计**

[本节对本包交付物中所需主要数据结构的设计进行描述。本节的数据结构设计可能只是一个初步设计，随着研发的进一步深入，相应的数据结构会不断的调整。注意，对每个数据结构，除了给出中文表意的名称外，还应给出程序中的命名]

**2 数据结构与系统模块的关系**

[本节说明上述数据结构同各个系统模块的关系，建议使用下表描述。]

**3 数据库设计**

[本节对本包交付物所用到的数据库设计进行描述，包括数据库的表结构、实体关系图等内容。]

**4.8.6系统错误处理设计**

[本章用于描述系统中对于各种错误出现进行处理的有关设计。]

**1 出错信息**

[用一览表的方式说明每种可能的出错或故障情况出现时，系统输出信息的形式、含意及处理方法。]

**2 补救措施**

[说明故障出现后可能采取的变通措施。包括：

a.后备技术（说明准备采用的后备技术，当原始系统数据万一丢失时启用的副本的建立和启动的技术，例如周期性地把磁盘信息记录到磁带上去就是对于磁盘媒体的一种后备技术。）

b.降效技术（说明准备采用的后备技术，使用另一个效率稍低的系统或方法来求得所需结果的某些部分，例如一个自动系统的降效技术可以是手工操作和数据的人工记录。）

c.恢复及再启动技术（说明将使用的恢复再启动技术，使软件从故障点恢复执行或使软件从头开始重新运行的方法。）]

**3 系统维护设计**

[本节说明为了系统维护的方便而在本包交付物内部设计中作出的安排，包括在程序中专门安排用于系统的检查与维护的检测点和专用模块。]

**4.8.7安全性设计**

**1 概述**

**1-1 设计原则**

**1-2 设计思路**

**2 xxxx的安全设计**

【具体系统在设计】

**4.8.8性能设计**

**1 性能指标描述**

**2 性能指标分析**

**3 性能指标运行环境**

**4.8.9附录**

[本章提供与系统开发有关的详细、专门的资料。如果包含附录，应明确指出是否将附录当作概要设计的一部分。]

**4.9 安全保密增强的涉密云虚拟化软件栈（信工所李青）**

[第1项子系统，从技术架构和实现流程等方面描述该子系统的设计。除文字说明外，请使用**图表**说明该子系统，保证清晰、完整、正确、可行。]

**4.9.1 技术及系统架构**

[描述本项子系统的技术架构，包括其功能模块划分连接、各模块功能任务的分配等内容，建议除文字说明外，请使用**图表**。]

**4.9.2界面设计**

[较为详尽地对本包交付物的用户使用界面设计进行描述。]

**<界面1-xxxx>**

[描述第1套操作界面，包括其样式、包含的要素、实现的功能等内容。如果需要，可继续分节，设多级目录详细描述。]

**4.9.3接口设计**

[对本包交付物的外部、内容及其它接口的设计情况进行描述。相对于需求规格说明书中的接口需求，这里应是设计后的具体实现。]

**外部接口**

[说明本包交付物同外界的所有接口的安排包括软件与硬件之间的接口、本系统与各支持系统之间的接口关系。如果需要，可分节设多级目录详细描述。]

**内部接口**

[说明本包交付物之内的各个系统模块之间的接口的安排。如果需要，可分节设多级目录详细描述。]

**4.9.4运行设计**

[对本包交付物实际运行过程的设计进行描述，包括运行模块组合、对运行的控制，以及运行时占用资源的安排等。]

**1 运行模块组合**

[说明对系统施加不同的外界运行控制时所引起的各种不同的运行模块组合，说明每种运行所历经的内部模块的支持软件。]

**2 存储设计**

**3 运行控制**

[说明每一种外界的运行控制的方式方法和操作步骤。]

**4 运行时间**

[说明每种运行模块组合将占用各种资源的时间。]

**4.9.5系统数据结构设计**

[本章对本包交付物中涉及的各式数据结构、数据库，以及其它数据的设计进行清晰、完整的描述。]

**1 主要数据结构设计**

[本节对本包交付物中所需主要数据结构的设计进行描述。本节的数据结构设计可能只是一个初步设计，随着研发的进一步深入，相应的数据结构会不断的调整。注意，对每个数据结构，除了给出中文表意的名称外，还应给出程序中的命名]

**2 数据结构与系统模块的关系**

[本节说明上述数据结构同各个系统模块的关系，建议使用下表描述。]

**3 数据库设计**

[本节对本包交付物所用到的数据库设计进行描述，包括数据库的表结构、实体关系图等内容。]

**4.9.6系统错误处理设计**

[本章用于描述系统中对于各种错误出现进行处理的有关设计。]

**1 出错信息**

[用一览表的方式说明每种可能的出错或故障情况出现时，系统输出信息的形式、含意及处理方法。]

**2 补救措施**

[说明故障出现后可能采取的变通措施。包括：

a.后备技术（说明准备采用的后备技术，当原始系统数据万一丢失时启用的副本的建立和启动的技术，例如周期性地把磁盘信息记录到磁带上去就是对于磁盘媒体的一种后备技术。）

b.降效技术（说明准备采用的后备技术，使用另一个效率稍低的系统或方法来求得所需结果的某些部分，例如一个自动系统的降效技术可以是手工操作和数据的人工记录。）

c.恢复及再启动技术（说明将使用的恢复再启动技术，使软件从故障点恢复执行或使软件从头开始重新运行的方法。）]

**3 系统维护设计**

[本节说明为了系统维护的方便而在本包交付物内部设计中作出的安排，包括在程序中专门安排用于系统的检查与维护的检测点和专用模块。]

**4.9.7安全性设计**

**1 概述**

**1-1 设计原则**

**1-2 设计思路**

**2 xxxx的安全设计**

【具体系统在设计】

**4.9.8性能设计**

**1 性能指标描述**

**2 性能指标分析**

**3 性能指标运行环境**

**4.9.9附录**

[本章提供与系统开发有关的详细、专门的资料。如果包含附录，应明确指出是否将附录当作概要设计的一部分。]