(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 102033781 B
(45)授权公告日 2012.07.18

(21)申请号 201110029570.X

(22)申请日 2011.01.27

(73)专利权人 中标软件有限公司
地址 200030 上海市徐汇区番禺路 1028 号 10 楼 1006-1010 室

(72)发明人 韩乃平 徐宁 田勇

(74)专利代理机构 上海智信专利代理有限公司 31002
代理人 王洁 郑暄

(51)Int.Cl.
G06F 9/48(2006.01)
G06F 21/00(2006.01)

(56)对比文件
CN 1726462 A,2006.01.25, 全文.
US 2010/0050111 A1,2010.02.25, 全文.
CN 101493783 A,2009.07.29, 全文.

审查员 徐春

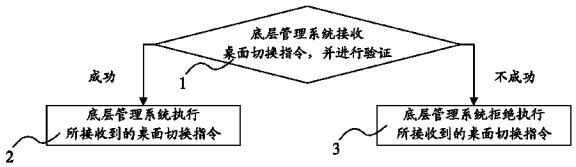
权利要求书 1 页 说明书 8 页 附图 5 页

(54)发明名称

虚拟机桌面系统切换方法

(57)摘要

本发明涉及一种虚拟机桌面系统切换方法，该桌面系统切换方法包括底层管理系统对所接收的桌面切换指令进行验证的步骤，若验证成功，则执行切换指令，若验证不成功，则拒绝执行。采用了该发明的虚拟机桌面系统切换方法，能够通过该验证过程，实现安全可靠的桌面系统切换，其该切换过程反应迅速，对虚拟机系统的性能没有明显的损耗，为用户提供了更为良好的使用体验。



CN 102033781 B

1. 一种虚拟机桌面系统切换方法，所述的虚拟机包括底层管理系统、以及运行于所述的底层管理系统上的两个桌面系统，其特征在于，所述的底层管理系统为 Xen 系统，所述的两个桌面系统分别为 Windows 系统和 Linux 系统，所述的底层管理系统与所述的桌面系统均具有相同的同步码，所述的桌面系统切换方法具体包括以下步骤：

(1) 底层管理系统根据用户所进行的虚拟桌面切换操作，接收相应的桌面切换指令，并对所接收的桌面切换指令进行验证；

其中，所述的桌面切换指令为经同步码加密的桌面切换指令消息包，所述的底层管理系统接收相应的桌面切换指令并对所接收的桌面切换指令进行验证，具体包括以下步骤：

(11) 底层管理系统收到经同步码加密的桌面切换指令消息包；所述的经同步码加密的桌面切换指令消息包具体是指：将身份信息、同步码和指令码通过 Hash 计算后获得的包括 Hash 码和指令码在内的桌面切换指令消息包；

(12) 底层管理系统通过运算获得桌面切换指令消息包的同步码；

(13) 底层管理系统将所获得的同步码与其具有的同步码比对，若一致，则验证成功，若不一致，则验证不成功；

(2) 若验证成功，则所述的底层管理系统执行所接收到的桌面切换指令；

(3) 若验证不成功，则所述底层管理系统拒绝执行所接收到的桌面切换指令。

2. 根据权利要求 1 所述的虚拟机桌面系统切换方法，其特征在于，所述的步骤 (12) 具体是指：

底层管理系统将桌面切换指令消息包解密获得身份信息、同步码和指令码。

3. 根据权利要求 1 所述的虚拟机桌面系统切换方法，其特征在于，所述的同步码为单向离散函数验证码。

4. 根据权利要求 1 所述的虚拟机桌面系统切换方法，其特征在于，所述的 Xen 系统具有输入设备功能库，所述的底层管理系统根据用户所进行的虚拟桌面切换操作接收相应的桌面切换指令，具体包括以下步骤：

(01) 用户键入桌面切换键盘信号；

(02) 所述的输入设备功能库获得所述的桌面切换键盘信号。

5. 根据权利要求 4 所述的虚拟机桌面系统切换方法，其特征在于，所述的输入设备功能库为 Xen 系统中服务层封装上的输入设备扩展功能应用层。

6. 根据权利要求 1 所述的虚拟机桌面系统切换方法，其特征在于，所述的桌面系统具有桌面切换程序，所述的底层管理系统根据用户所进行的虚拟桌面切换操作接收相应的桌面切换指令，具体包括以下步骤：

(01′) 用户运行桌面切换程序；

(02′) 桌面切换程序通过 Windows 驱动模型直接调用 Windows 系统的内核驱动程序；

(03′) Windows 系统的内核驱动程序向 Windows 系统的虚拟硬件发送桌面切换消息；

(04′) Xen 系统从 Windows 系统的虚拟硬件接收桌面切换指令。

虚拟机桌面系统切换方法

技术领域

[0001] 本发明涉及虚拟机技术领域，特别涉及多域切换技术领域，具体是指一种虚拟机桌面系统切换方法。

背景技术

[0002] 虚拟机系统体系运行三个系统：即特权管理系统、通用桌面系统和安全 Linux 桌面系统。其中，特权管理系统为虚机系统的特权域系统，负责对后两个系统进行资源分配和管理。通用系统和安全系统是用户主要使用的两个系统，虚拟机系统默认引导是进入通用系统（如 Windows 系统），进入系统后用户可正常进行日常事务处理，如需处理一些涉及私密敏感信息时（如登录网上银行办理业务、安全办公存储等），可通过切换方式直接转入安全系统进行操作。

[0003] 基于多核 CPU 的硬件虚拟化技术，可有效改善虚拟系统的运行效率和性能问题，且为双系统并行运行提供有力的保障。基于多核 CPU 的虚拟机系统可为每个虚拟系统分配一个单独的核心，并在多核共享多个核心、共同拥有一条前端总线的条件下，进一步优化多域的访问流程。但现有的虚拟机系统缺乏一种安全有效，且反应迅速的桌面系统切换机制，使虚拟机系统的应用受到制约。

[0004] 基于可信计算技术实现可信虚拟机监视器 TVMM，并在可信虚拟机监视器 TVMM 内部实施对敏感资源的安全访问控制策略，保证整个虚拟机系统底层核心的安全可信。

[0005] 特权管理系统实现对并行运行的双系统进行底层虚拟资源的安全管理与安全监控。此外，该系统也是整个虚拟机系统的管理控制中心，一般只能由特权管理员用户才能进入操作。特权管理系统采用 Xen 系统，Xen 系统是一个开放源代码虚拟机监视器，它可以在单个计算机上运行多达 100 个满特征的操作系统。操作系统可以通过进行显式地修改（移植）以在 Xen 上运行（但是提供对用户应用的兼容性）。这使得 Xen 无需特殊硬件支持，就能达到高性能的虚拟化。

发明内容

[0006] 本发明的目的是克服了上述现有技术中的缺点，提供一种安全有效，反应迅速，且应用简便的虚拟机桌面系统切换方法。

[0007] 虚拟机包括底层管理系统、以及运行于所述的底层管理系统上的两个桌面系统。为了实现上述的目的，本发明的虚拟机桌面系统切换方法包括以下步骤：

[0008] (1) 底层管理系统根据用户所进行的虚拟桌面切换操作，接收相应的桌面切换指令，并对所接收的桌面切换指令进行验证；

[0009] (2) 若验证成功，则所述的底层管理系统执行所接收到的桌面切换指令；

[0010] (3) 若验证不成功，则所述底层管理系统拒绝执行所接收到的桌面切换指令。

[0011] 该虚拟机桌面系统切换方法中，所述的底层管理系统与所述的桌面系统均具有相同的同步码，所述的桌面切换指令为经同步码加密的消息包，所述的底层管理系统接收相

应的桌面切换指令，并对所接收的桌面切换指令进行验证，具体包括以下步骤：

[0012] (11) 底层管理系统收到经同步码加密的桌面切换指令消息包；

[0013] (12) 底层管理系统通过运算获得消息包的同步码；

[0014] (13) 底层管理系统将所获得的同步码与其具有的同步码比对，若一致，则验证成功，若不一致，则验证不成功。

[0015] 该虚拟机桌面系统切换方法中，所述的经同步码加密具体是指：将身份信息、同步码和指令码通过 Hash 计算后获得的包括 Hash 码和指令码在内的消息包。

[0016] 该虚拟机桌面系统切换方法中，所述的步骤 (12) 具体是指：底层管理系统将消息包解密获得身份信息、同步码和指令码。

[0017] 该虚拟机桌面系统切换方法中，所述的同步码为单向离散函数验证码。

[0018] 该虚拟机桌面系统切换方法中，所述的底层管理系统为 Xen 系统，所述的两个桌面系统分别为 Windows 系统和 Linux 系统。

[0019] 该虚拟机桌面系统切换方法中，所述的 Xen 系统具有输入设备功能库，所述的用户所进行的虚拟桌面切换操作，具体包括以下步骤：

[0020] (01) 用户键入桌面切换键盘信号；

[0021] (02) 所述的输入设备功能库获得所述的桌面切换键盘信号。

[0022] 该虚拟机桌面系统切换方法中，所述的输入设备功能库为 Xen 系统中服务层封装上的输入设备扩展功能应用层。

[0023] 该虚拟机桌面系统切换方法中，所述的桌面系统具有桌面切换程序，所述的用户所进行的虚拟桌面切换操作，具体包括以下步骤：

[0024] (01′) 用户运行桌面切换程序；

[0025] (02′) 桌面切换程序通过 Windows 驱动模型直接调用 Windows 系统的内核驱动程序；

[0026] (03′)Windows 系统的内核驱动程序向 Windows 系统的虚拟硬件发送桌面切换消息；

[0027] (04′)Windows 系统的虚拟硬件向 Xen 系统发送桌面切换指令。

[0028] 该虚拟机桌面系统切换方法中，所述的步骤 (3) 包括以下步骤：

[0029] (31)Xen 系统通过守护进程 Switch Daemon 通知窗口管理器；

[0030] (32) 窗口管理器通过 VNC 协议切换所显示的桌面窗口。

[0031] 采用了该发明的虚拟机桌面系统切换方法，由于虚拟机的底层管理系统在接收到相应的桌面切换指令后，对指令进行验证，验证成功后，才会执行所接收到的桌面切换指令，若验证不成功，则拒绝执行指令。使得该虚拟机桌面系统切换方法能够通过该验证过程，实现安全可靠的桌面系统切换，其该切换过程反应迅速，对虚拟机系统的性能没有明显的损耗，为用户提供了更为良好的使用体验。

附图说明

[0032] 图 1 为本发明的虚拟机桌面系统切换方法的步骤流程图。

[0033] 图 2 为本发明的虚拟机桌面系统切换方法中所采用的消息通信验证机制原理示意图。

[0034] 图 3 为本发明的虚拟机桌面系统切换方法中所采用的硬切换方法的原理示意图。

[0035] 图 4 为本发明的虚拟机桌面系统切换方法中所采用的软切换方法的原理示意图。

[0036] 图 5 为本发明的虚拟机桌面系统切换方法中的基层 Xen 系统和桌面 Windows 系统的结构示意图。

[0037] 图 6 为本发明的虚拟机桌面系统切换方法中的基层 Xen 系统和桌面 Linux 系统的结构示意图。

具体实施方式

[0038] 为了能够更清楚地理解本发明的技术内容，特举以下实施例详细说明。

[0039] 请参阅图 1 所示，为本发明的虚拟机桌面系统切换方法的步骤流程图。所述的虚拟机包括底层管理系统、以及运行于所述的底层管理系统上的两个桌面系统。

[0040] 在一种实施方式中，所述的虚拟机桌面系统切换方法具体包括以下步骤：

[0041] (1) 底层管理系统根据用户所进行的虚拟桌面切换操作，接收相应的桌面切换指令，并对所接收的桌面切换指令进行验证；

[0042] (2) 若验证成功，则所述的底层管理系统执行所接收到的桌面切换指令；

[0043] (3) 若验证不成功，则所述底层管理系统拒绝执行所接收到的桌面切换指令。

[0044] 其中，所述的底层管理系统为 Xen 系统，所述的两个桌面系统分别为 Windows 系统和 Linux 系统。所述的步骤 (3) 具体包括以下步骤：

[0045] (31)Xen 系统通过守护进程 Switch Daemon 通知窗口管理器；

[0046] (32) 窗口管理器通过 VNC 协议切换所显示的桌面窗口。

[0047] 如图 2 所示，为本发明的虚拟机桌面系统切换方法中所采用的消息通信验证机制原理示意图。在一种较优选的实施方式中，所述的底层管理系统与所述的桌面系统均具有相同的同步码，所述的桌面切换指令为经同步码加密的消息包，所述的底层管理系统接收相应的桌面切换指令，并对所接收的桌面切换指令进行验证，具体包括以下步骤：

[0048] (11) 底层管理系统收到经同步码加密的桌面切换指令消息包；

[0049] (12) 底层管理系统通过运算获得消息包的同步码；

[0050] (13) 底层管理系统将所获得的同步码与其具有的同步码比对，若一致，则验证成功，若不一致，则验证不成功。

[0051] 在进一步优选的实施方式中，所述的经同步码加密具体是指：将身份信息、同步码和指令码通过 Hash 计算后获得的包括 Hash 码和指令码在内的消息包。所述的步骤 (12) 具体是指：底层管理系统将消息包解密获得身份信息、同步码和指令码。所述的同步码为单向离散函数验证码。

[0052] 如图 3 所示，为本发明的虚拟机桌面系统切换方法中所采用的硬切换方法的原理示意图。在更优选的实施方式中，所述的 Xen 系统具有输入设备功能库，所述的用户所进行的虚拟桌面切换操作，具体包括以下步骤：

[0053] (01) 用户键入桌面切换键盘信号；

[0054] (02) 所述的输入设备功能库获得所述的桌面切换键盘信号。

[0055] 其中，所述的输入设备功能库为 Xen 系统中服务层封装上的输入设备扩展功能应用层。

[0056] 如图 4 所示，为本发明的虚拟机桌面系统切换方法中所采用的软切换方法的原理示意图。在另一种更优选的实施方式中，所述的桌面系统具有桌面切换程序，所述的用户所进行的虚拟桌面切换操作，具体包括以下步骤：

[0057] (01′) 用户运行桌面切换程序；

[0058] (02′) 桌面切换程序通过 Windows 驱动模型直接调用 Windows 系统的内核驱动程序；

[0059] (03′)Windows 系统的内核驱动程序向 Windows 系统的虚拟硬件发送桌面切换消息；

[0060] (04′)Windows 系统的虚拟硬件向 Xen 系统发送桌面切换指令。

[0061] 在本发明的应用中，为了实现发明目的，保证虚拟机系统的易用性、安全性，本系统设计了软、硬两种系统切换机制。并借助验证机制保证发送切换信号的域间通信安全。VMM 系统由 Xen 虚拟机来实现，称为 host 系统 (Dom0)；运行在 host 之上的两个虚拟桌面系统，称为 guest 系统，一个安装 Windows 系统 (VM-Windows)，另一个安装中标麒麟 Linux 桌面系统 (VM-Linux)。

[0062] 其具体方案如图 5 及图 6 所示，软件系统主要分为三部分，即 Host 主机上的守护进程 (Switch Daemon)、VM-Windows 上的 VM-Windows-tools(图 5) 和 VM-Linux 上的 VM-Linux-tools(图 6)。

[0063] Host 主机上的守护进程主要完成消息的监听及收到消息后，对消息的处理。处理内容主要包括消息认证、域间切换和 Host 上共享目录的同步。

[0064] VM-Windows 上的 VM-Windows-tools 与 VM-Linux 上的 VM-Linux-tools 主要完成将用户的操作信号通过虚拟设置驱动程序所控制的虚拟设备传递给 Host 主机上的守护进程。信号种类有三种：键盘硬切换信号、桌面图标软切换信号和数据传输信号。具体的说，VM-Windows-tools 通过虚拟设备驱动程序将信号传递出去，而 VM-Linux-tools 通过 Xen 的事件通道机制将信号传递出去。

[0065] 一、发明要实现 Guest 系统中的 Windows 和 Linux 桌面切换。有两种实现方案：

[0066] 1、“硬中断”方式，通过直接按键盘快捷键 (如 F12) 的方式实现桌面切换。

[0067] 在 Xen 系统上，所有键盘信号是被 Xen 截获的。但对于 I/O 设备的数据处理则是由 Dom0 上的 Host 系统来负责的。对于图形应用程序，所有 I/O 设备数据均由 XServer 负责向上层应用程序分发，即 XServer 会首先收到键盘事件。如图 3 所示，XLib 和 XInput2 均为第三方库，供 XClient 端调用。具体实现方法是利用 XLib 的扩展功能 XInput2 功能库，实现对键盘事件的提取。当提取到 F12 键的键盘事件后，守护进程 (Switch Daemon) 通知窗口管理器，切换通过 VNC 协议显示的另一桌面窗口，即达到桌面切换的目的。

[0068] 2、“软中断”方式，通过点击桌面应用程序快捷方式实现桌面切换。

[0069] 如图 4 所示。软切换程序主要由 VM-Windows 上的切换程序和 Dom0 上 Host 主机中运行的切换守护进程组成。VM-Windows 上的切换程序是通过 Windows 驱动编程 WDM 实现应用层调用 Windows 系统内核以达到直接驱动虚拟串口向外发切换消息的目的。

[0070] 软切换即让系统即时响应切换的请求事件，并向 Dom0 上 Host 系统发送切换桌面信号。由于 VM-Windows 在 Windows 操作系统看来所有的硬件都是真实的，而实际上硬件是 Xen 为 VM-Windows 虚拟的设备。向外界发送信号的方式一是通过网络，二是通过虚拟硬件。

前者由于依托网络，而存在巨大风险性。后者由于是软件虚拟的硬件，这实际上提供了实现的可行性。为降低Windows软中断的开发难度，利用Windows操作系统可对虚拟硬件的自识别与自驱动的特性，在Windows上开发的切换程序，通过Windows WDM直接调用内核驱动，再通过内核驱动向虚拟硬件发送消息，即可实现VM-Windows向host发送切换消息的目的。

[0071] 二、切换信息验证技术

[0072] 消息通信机制是VM域与host切换通信的关键，域间切换是以消息通信机制为基础的。下表是系统定义的消息码。

[0073]

序号	消息类型	保留	Win/Linux	信息码					保留
1	softSwitch2Linux	0	0	0	0	0	0	1	0
2	hardSwitch2Window	0	0	0	0	0	1	0	0
3	copyFileToLinux	0	0	0	0	0	1	1	0
4	copyFileFinished(win)	0	0	0	0	1	0	0	0
5	copyClipBoard2Linux	0	0	0	0	1	0	1	0
6	copyClipBoardFinished(win)	0	0	0	0	1	1	0	0
7	Linux-Synchronous（同步msg）	0	0	0	0	1	1	1	0
8	softSwitch2Wins	0	0	1	0	0	0	1	0
9	hardSwitch2Wins	0	0	1	0	0	1	0	0
10	copyFile2Wins	0	0	1	0	0	1	1	0
11	copyFileFinished(Linux)	0	0	1	0	1	0	0	0
12	copyClipBoard2Wins	0	0	1	0	1	0	1	0
13	copyClipBoardFinished(linux)	0	0	1	0	1	1	0	0
14	Linux-synchronous（同步msg）	0	0	1	0	1	1	1	0

[0074] 表1信息码表

[0075] 为了保证消息通信不被篡改，系统设计了一套基于单向散列函数的身份验证机制。如图2所示，域间发送的消息由身份信息、同步码、消息码做hash运算得到的hash码加消息码组成，消息码明文传送。当host守护进程接收到消息后，首先将传送过来的消息码与身份信息、同步码做hash运算，得到的hash值与接收到的hash码做比对，比对成功，即通过身份验证，此后，host守护进程可对收到的消息码做进一步解码处理，否则拒绝此消息。身份信息做为系统的私钥，是由系统来维护其保密性。同步码，保证了一次一密，攻击者无法发送伪造消息。同时，VM端将定时对收发两端的同步码进行同步。

[0076] 为能更清楚地描述本发明的内容。以VNC显示和通讯加密为例，通过伪代码和注释形式，进行更进一步的说明。

[0077] VNC显示基于RFB协议实现的，主要协议实现部分和控件部分组成。这里通过控件部分的实现来说明如何进行切换。

[0078] 1、连接VNC-Server与显示初始化

[0079] 首先创建 VNC-Server 的连接，并且创建和启动显示线程。

[0080] url.setHost(VNC_Host)； // 设置主机地址

[0081] url.setPort(VNC_Port)； // 设置端口号

[0082] vncViewThread = new VNCViewThread(url)// 创建连接

[0083] vncViewThread->Start()； // 启动显示线程

[0084] connect(SIGNAL(connected()))； // 绑定连接完成信号

[0085] connect(SIGNAL(changeSize()))； // 绑定改变大小信号

[0086] connect(SIGNAL(reinit_me()))； // 绑定重新连接信号

[0087] 接下来做一些其它的处理：

[0088] fullscreenEnabled = true； // 全屏模式

[0089] view->setGrabAllKeys(true)； // 获取所有键盘事件

[0090] view->setFocus()； // 获取鼠标事件

[0091] 以上完成了 VNC 显示的初始化过程。接下来该针对本系统做实例化处理。

[0092] 2、显示实例化处理

[0093] 2.1、Windows 显示的实例化

[0094] deleteAll()； // 删除之前的控件

[0095] vncWindows Show *= new VNCShow()；// 创建 Windows 显示

[0096] vncWindowsShow->setAttribute(DeleteOnClose)；// 设置属性

[0097] vncWindowsShow->initView()； // 初始化

[0098] vncWindowsShow->setGeometry()； // 设置显示的位置和大小

[0099] vncWindowsShow->raise()； // 窗口控件重绘

[0100] vncWindowsShow->grabAllKeys()； // 捕获所有键盘事件

[0101] 2.2、Linux 显示的实例化

[0102] deleteAll()； // 删除之前的控件

[0103] vncLinuxShow *= new VNCShow()； // 创建 Windows 显示

[0104] vncLinuxShow->setAttribute(DeleteOnClose)；// 设置属性

[0105] vncLinuxShow->initView()； // 初始化

[0106] vncLinuxShow->setGeometry()；// 设置显示的位置和大小

[0107] vncLinuxShow->raise()； // 窗口控件重绘

[0108] vncLinuxShow->grabAllKeys()； // 捕获所有键盘事件

[0109] 这两部分的实例化，其实就是将 VNC 传递过来的图像显示到窗口上，并且捕获所有的键盘事件。

[0110] 3、切换显示

[0111] 切换显示的逻辑较为简单。

[0112] if(! isAllowswitch)return； // 删除之前的控件

[0113] if(isShowWindows)

[0114] {

[0115] showLinux()；

[0116] isShowWindows = false；

[0117] }

[0118] else

[0119] {

[0120] showWindows() ;

[0121] isShowWindows = true ;

[0122] }

[0123] 4、“硬切换”和“软切换”

[0124] “硬切换”和“软切换”是切换的触发条件。

[0125] 4.1、“硬切换”是当用户按下键盘的某一个功能键（本例中为F12）后，触发桌面切换显示。

[0126] while(1)

[0127] {

[0128] XEvent ev ;

[0129] XGenericEventCookie * cookie = (XGenericEventCookie *)&ev.xcookie ;

[0130] XNextEvent(g.display,(XEvent *)&ev) ;

[0131] if(XGetEventData(g.display, cookie)&&cookie- > type = = GenericEvent)

[0132] {

[0133] if(cookie- > evtype = = XI_KeyPress)

[0134] {

[0135] checkkey(&g,cookie- > data) ;// 判断是否 F12

[0136] }

[0137] }

[0138] }

[0139] 4.2、“软切换”是用户通过点击工具栏按钮或者是双击桌面切换快捷方式触发桌面切换显示。

[0140] if(event-key() = = Key_12)

[0141] {

[0142] switchShow() ; // 切换显示

[0143] }

[0144] 5、域间通讯

[0145] 域间通讯主要是虚拟机之间的通讯。在主机上开辟了两个共享的文件夹。针对Windows，主机中启用 Samba 服务。针对 Linux，主机中启用 NFS 服务。当需要传递文件时，通过发送命令，主机首先同步两个共享的文件夹，之后，虚拟机通过访问文件夹来实现文件的传递。

[0146] 主机同步是通过命令实现的 :

[0147] rsync-r/linuxpath/windowspath

[0148] 6、认证机制的实现

[0149] 认证主要是通讯密文实现的，认证机制的主要原理后边会介绍到，这里介绍实现的过程。

[0150] 6.1、加密过程：

[0151] QString headsync(sync)； //获取同步码

[0152] QString sendrawcmd = headsync+rawCmd;//获取要加密的命令

[0153] QString md5cmd = getMd5(sendrawcmd);//对命令进行 Md5 加密

[0154] QString sendCmd =" DomU/Host *" +md5cmd+" *" +sendrawcmd；

[0155] 6.2、解密过程：

[0156] 解密的过程也与加密类似，是对接收到的命令进行截取，对 MD5 码进行比对。

[0157] 采用了该发明的虚拟机桌面系统切换方法，由于虚拟机的底层管理系统在接收到相应的桌面切换指令后，对指令进行验证，验证成功后，才会执行所接收到的桌面切换指令，若验证不成功，则拒绝执行指令。使得该虚拟机桌面系统切换方法能够通过该验证过程，实现安全可靠的桌面系统切换，其该切换过程反应迅速，对虚拟机系统的性能没有明显的损耗，为用户提供了更为良好的使用体验。

[0158] 在此说明书中，本发明已参照其特定的实施例作了描述。但是，很显然仍可以作出各种修改和变换而不背离本发明的精神和范围。因此，说明书和附图应被认为是说明性的而非限制性的。

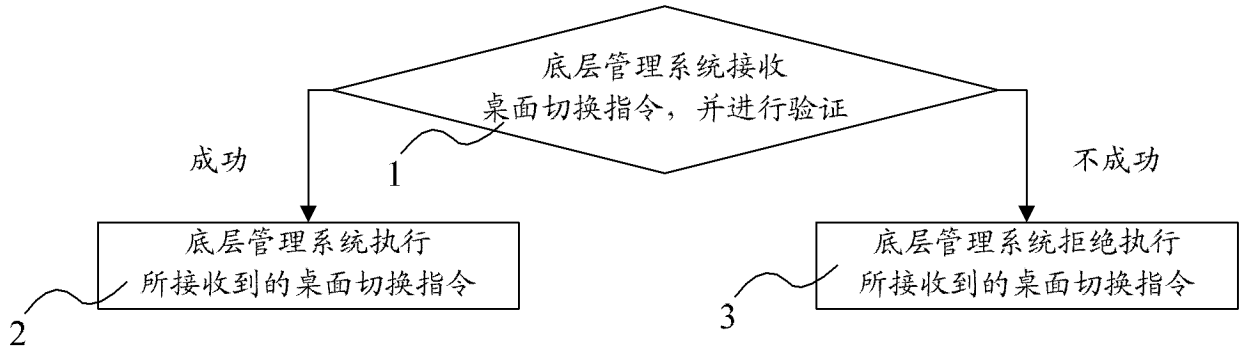

底层管理系统接收
桌面切换指令，并进行验证
1
成功
不成功
底层管理系统执行
所接收到的桌面切换指令
2
底层管理系统拒绝执行
所接收到的桌面切换指令
3

图 1

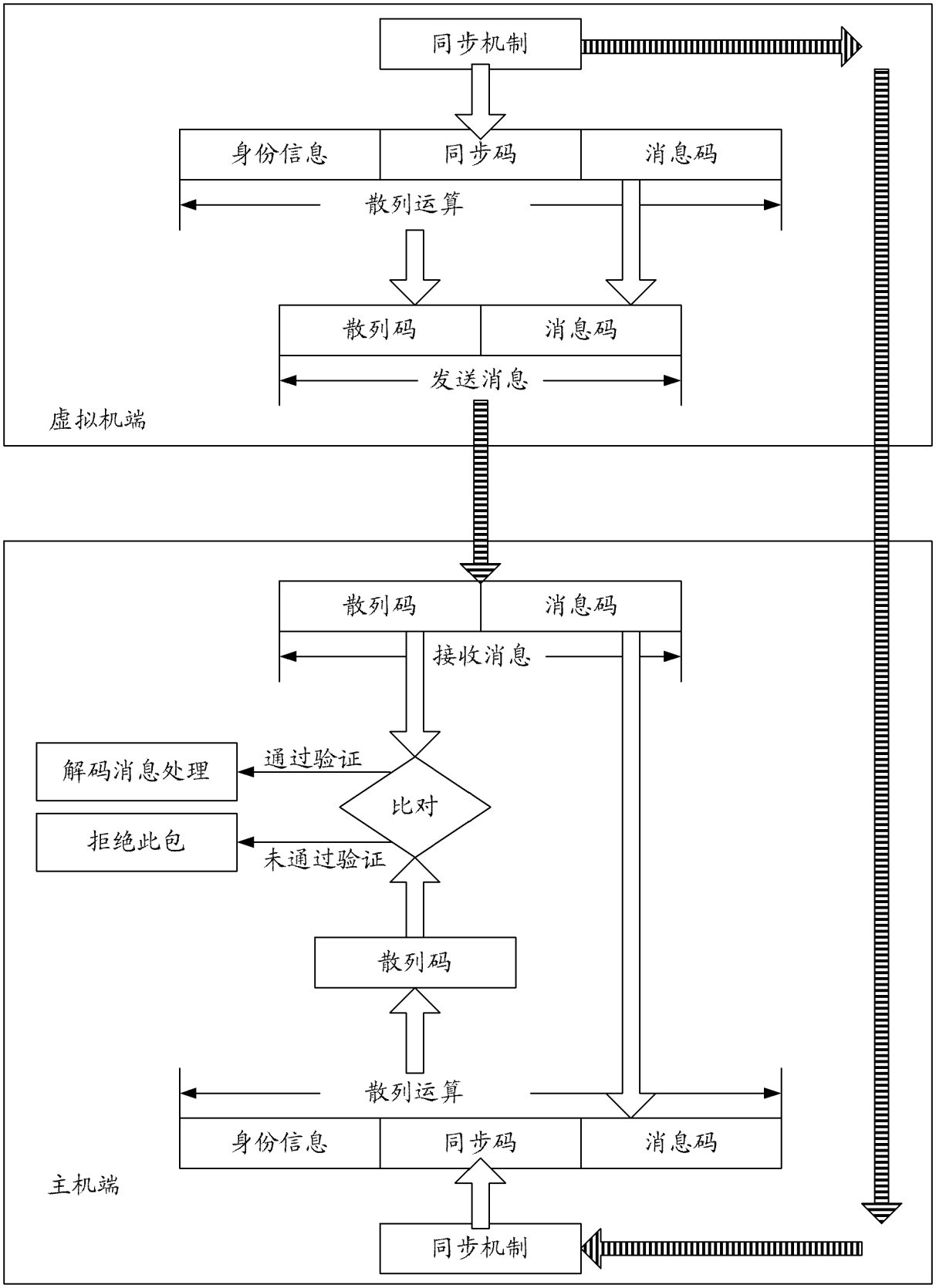

同步机制
身份信息
同步码
消息码
散列运算
散列码
消息码
发送消息
虚拟机端
散列码
消息码
接收消息
解码消息处理
通过验证
比对
拒绝此包
未通过验证
散列码
散列运算
身份信息
同步码
消息码
主机端
同步机制

图 2

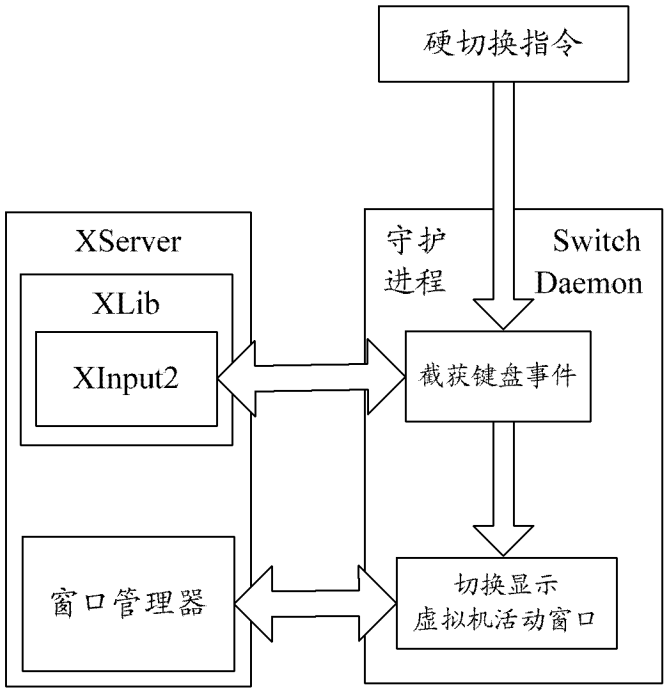

硬切换指令
XServer
XLib
XInput2
守护
进程
Switch
Daemon
截获键盘事件
窗口管理器
切换显示
虚拟机活动窗口

图 3

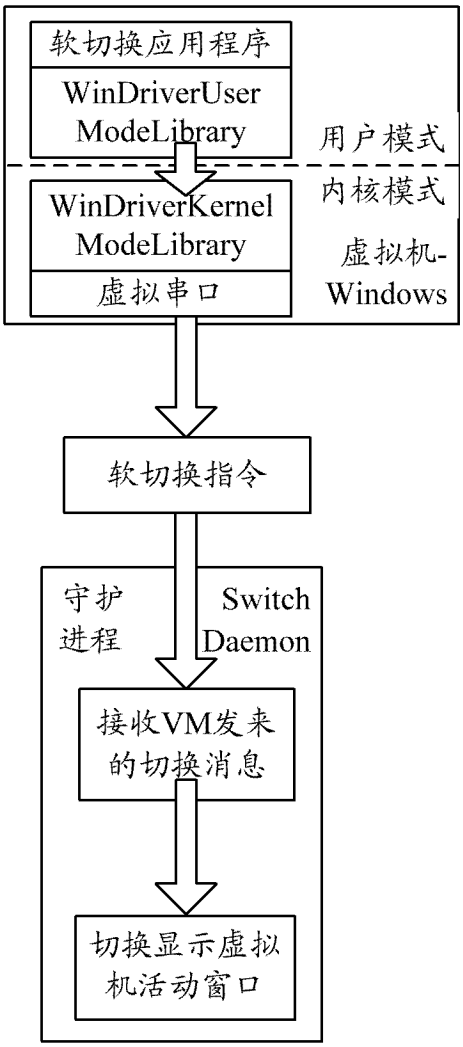

软切换应用程序
WinDriverUser
ModeLibrary
用户模式
内核模式
WinDriverKernel
ModeLibrary
虚拟串口
虚拟机-
Windows
软切换指令
守护
进程
Switch
Daemon
接收VM发来
的切换消息
切换显示虚拟
机活动窗口

图 4

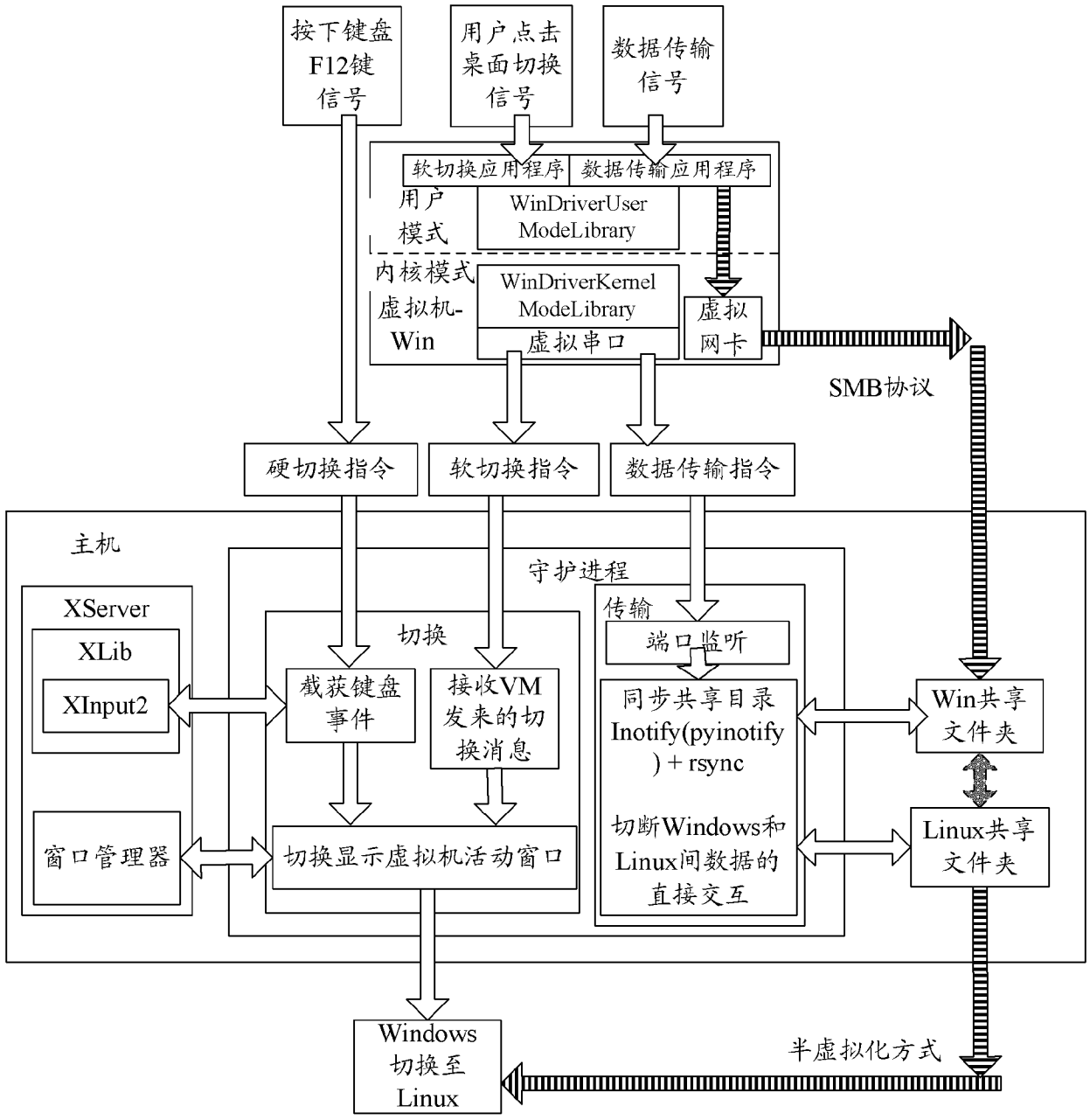

按下键盘F12键信号
用户点击桌面切换信号
数据传输信号
软切换应用程序
数据传输应用程序
用户模式
WinDriverUser ModeLibrary
内核模式
虚拟机-Win
WinDriverKernel ModeLibrary
虚拟串口
虚拟网卡
SMB协议
硬切换指令
软切换指令
数据传输指令
主机
守护进程
XServer
XLib
XInput2
切换
截获键盘事件
接收VM发来的切换消息
传输
端口监听
同步共享目录 Inotify(pyinotify) + rsync
切断Windows和Linux间数据的直接交互
Win共享文件夹
Linux共享文件夹
窗口管理器
切换显示虚拟机活动窗口
Windows切换至Linux
半虚拟化方式

图 5

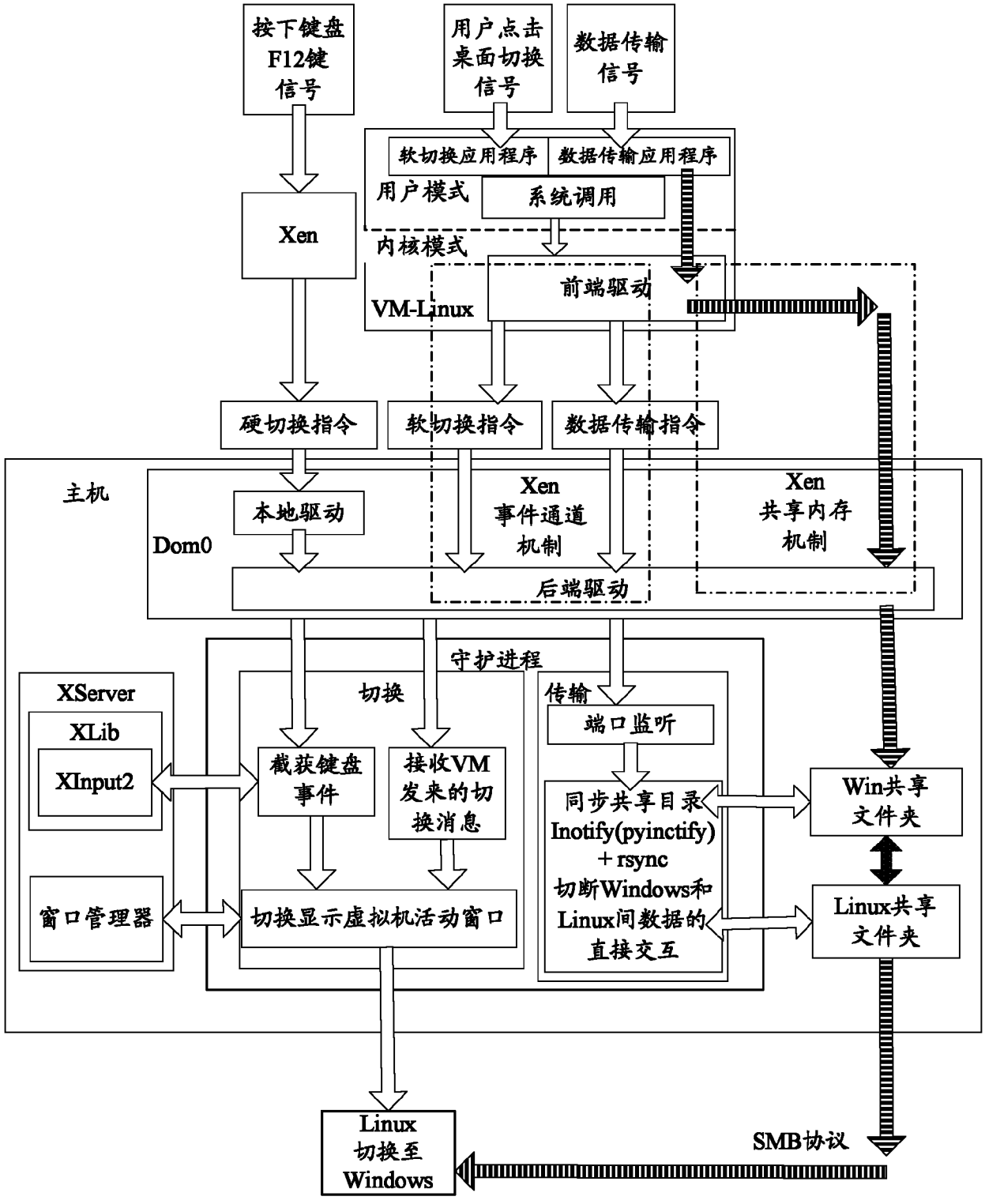

按下键盘
F12键
信号
用户点击
桌面切换
信号
数据传输
信号
软切换应用程序
数据传输应用程序
用户模式
系统调用
Xen
内核模式
VM-Linux
前端驱动
硬切换指令
软切换指令
数据传输指令
主机
本地驱动
Dom0
Xen
事件通道
机制
Xen
共享内存
机制
后端驱动
守护进程
XServer
XLib
XInput2
切换
传输
端口监听
截获键盘
事件
接收VM
发来的切
换消息
同步共享目录
Inotify(pyinctify)
+rsync
切断Windows和
Linux间数据的
直接交互
Win共享
文件夹
Linux共享
文件夹
窗口管理器
切换显示虚拟机活动窗口
Linux
切换至
Windows
SMB协议

图 6