

金航数码服务器虚拟化系统

版本 V5

2021 年 2 月

产品技术白皮书



金航数码科技有限责任公司出品

版权所有 © 金航数码科技有限责任公司 2021

文档由航空工业信息技术中心(金航数码科技有限责任公司,简称金航数码)版权所有。未经本公司同意,严禁复制或者传播本档的任何内容。最终用户仅能够在内部复制和传播。

联系信息

地 址	北京市朝阳区曙光西里甲 5 号院航空工业大厦 E 座
邮 编	100028
网 址	www.avic-digital.com
电 话	010-58355588
传 真	010-58355100

文档版本历史

版本	日期	备注
1.0	2021 年 2 月 5 日	

免责声明

本文档不得视为或解释为金航数码科技有限责任公司(下文简称为“金航数码公司”)和客户之间有效而且有约束力的协议。金航数码公司对本文档内容的准确性、完整性或充分性或文档的使用不做任何担保,而且特别明确表示,对于本文档的实效性和对特定目的的适用性,不做任何明示或暗示的担保。此外,金航数码公司保留修订本文档及其内容的权利。

目 录

1	产品简介.....	1
1.1	技术背景	1
1.1.1	虚拟化技术发展历程	1
1.1.2	虚拟化技术工作机理	3
1.1.3	X86 虚拟化技术简介	7
1.2	虚拟化在涉密行业应用面临的问题.....	10
1.3	金航数码服务器虚拟化系统.....	11
1.4	参考标准	11
2	产品体系架构	12
2.1	总体架构	12
2.2	功能架构	13
2.3	物理架构	13
2.3.1	主管理服务器	14
2.3.2	虚拟化资源服务器	14
2.3.3	SAN 存储	14
2.3.4	交换网络	15
3	产品功能特性	16
3.1	总体技术特点	16
3.2	高可控性（HIGH CONTROLLABILITY）	16
3.2.1	全部源代码自主可控	16
3.2.2	自主分布式调度任务执行引擎	16
3.3	高安全性（HIGH SECURITY）	17
3.3.1	三员分权管理	17
3.3.2	密级管理	18
3.3.3	虚拟机安全隔离管理	20
3.3.4	虚拟网络安全隔离管理	20
3.3.5	虚拟防火墙管理	21

3.3.6	虚拟机数据加密保护管理	21
3.3.7	虚拟机剩余信息保护管理	22
3.3.8	虚拟机逃逸监控管理	22
3.3.9	虚拟机操作日志审计管理	23
3.3.10	第三方安全审计支持	24
3.3.11	第三方防病毒系统支持	24
3.4	高性能（HIGH PERFORMANCE）	24
3.4.1	虚拟化性能	24
3.4.2	虚拟 CPU 性能优化	25
3.4.3	虚拟内存性能优化	26
3.4.4	虚拟机 GPU 直通	27
3.4.5	存储裸设备挂载	27
3.5	高可用性（HIGH AVAILABILITY）	28
3.5.1	虚拟机在线迁移	28
3.5.2	虚拟机高可用调度	29
3.5.3	虚拟机 FT 热备高可用	29
3.5.4	虚拟机操作系统传统高可用	30
3.6	高可管理性（HIGH MAINTAINABILITY）	30
3.6.1	虚拟机资源热扩展管理	30
3.6.2	虚拟化调度任务跟踪分析	31
3.6.3	资源运行状态高效监控	31
3.6.4	系统告警管理	32
3.7	高可定制性（HIGH CUSTOMIZATION）	32
3.7.1	资源监控可视化定制	32
3.7.2	审批流程定制	33
3.7.3	调度任务定制	33
3.7.4	系统界面皮肤定制	33
4	典型硬件配置	34
4.1	服务器	34

4.2	SAN 存储.....	35
5	典型应用场景	36
5.1	涉密网非标虚拟化系统升级替代.....	36
5.2	新建涉密虚拟化系统	36
5.3	新建非密虚拟化系统	37
6	缩略语	38
6.1	缩略语	38

1 产品简介

1.1 技术背景

1.1.1 虚拟化技术发展历程

在计算机科学中，虚拟化（Virtualization）是一个表现逻辑群组或计算机资源的子集的进程，用户可以用与原本的组织管理不同的方式来存取这些进程，影响最广泛的虚拟化例子是Java虚拟机（JVM）。

早在上世纪60-70年代，IBM就已经在360/67、370等硬件体系实现了虚拟化。IBM的虚拟化通过虚拟机监视器VMM（Virtual Machine Monitor）把一个硬件虚拟成多个硬件（VM，Virtual Machine），各VM之间可以认为是完全隔离的。这个隔离不同于各进程之间的地址空间隔离。无论是内存，设备，还是处理器等对各VM来讲，都被认为是自己单独一套的。在VM上可以运行不同的操作系统（称为Guest OS）而不会对其它的VM产生影响。

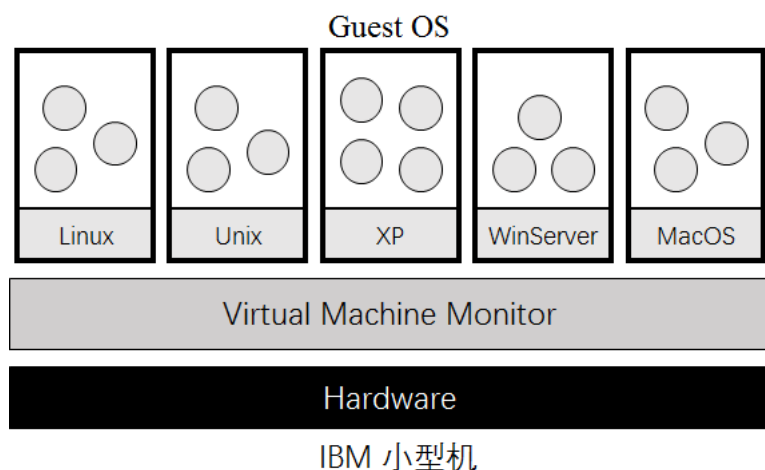


图 1 IBM 小型机虚拟化架构示意图

上世纪90年代，以VMware为代表的部分虚拟化软件厂商推出一种基于“全虚拟化”模式的X86服务器平台虚拟化解决方案，但是这种完全依赖软件进行模拟仿真的“全虚拟化”模式，每个虚拟机获得的虚拟计算资源都要由底层的

Hypervisor控制和分配，同时需要进行虚拟CPU指令到物理CPU指令的二次转换，而二次转换带来的开销使得“全虚拟化”的性能较差，无法真正达到物理机的运行效果。为解决性能问题，出现了一种称为“半虚拟化”的过渡虚拟化技术，通过对虚拟机Guest OS进行定制修改，实现部分开销较大的虚拟CPU指令不需要二次转换，使定制的虚拟机Guest OS获得接近物理机的性能，但是修改Guest OS也带来了系统指令级的冲突及运行效率问题，需要投入大量复杂的定制化工作，对于非操作系统原厂的定制化版本（如Windows），也将失去操作系统原厂商的持续技术支持。

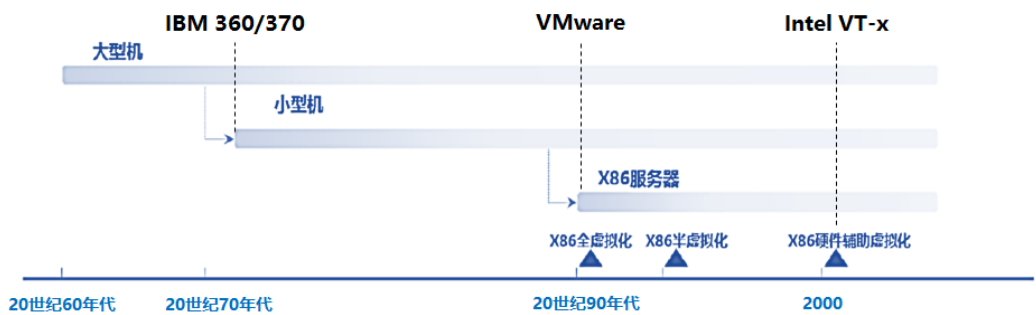


图 2 虚拟化技术发展历史

目前，虚拟化技术已经发展到了硬件支持的阶段，“硬件虚拟化”技术就是把纯软件虚拟化技术的各项功能用硬件电路来实现，可减少VMM运行的系统开销，可同时满足CPU半虚拟化和二进制转换技术的需求，X86处理器一共有4个不同优先级，术语称为Ring，从Ring 0-Ring3。Ring 0的优先级最高，Ring 3最低。Ring 0用于操作系统内核，Ring 1和Ring 2用于操作系统服务，Ring 3用于应用程序。

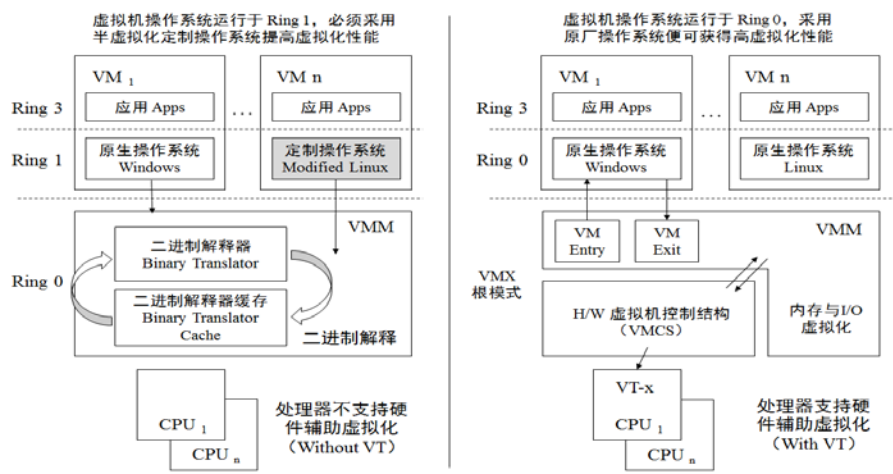


图 3 Intel VT 硬件辅助虚拟化架构示意图

2005年，Intel公司推出了VT虚拟化技术，包括对指令集虚拟化的支持VT-x

和VT-i技术，还包括对I/O设备虚拟化支持的VT-d技术，在虚拟化过程中利用VT虚拟化技术所提供的VMX模式运行虚拟机管理器VMM，通过硬件底层指令集来支持虚拟机管理器VMM与已安装Guest OS(虚拟机上的操作系统)之间的切换，使得虚拟化技术更加简单、高效、可靠。

1.1.2 虚拟化技术工作机理

虚拟化技术的主要思想是提高硬件资源的利用效率，将一个独立的硬件服务器虚拟成多个逻辑服务器（称为虚拟机），就是在一个物理服务器上运行多个虚拟机， 这些虚拟机共享底层硬件， 从虚拟机中的应用的角度看，虚拟机就象是一个物理服务器，有完整的操作系统、CPU、内存、I/O设备（网卡、磁盘等）。

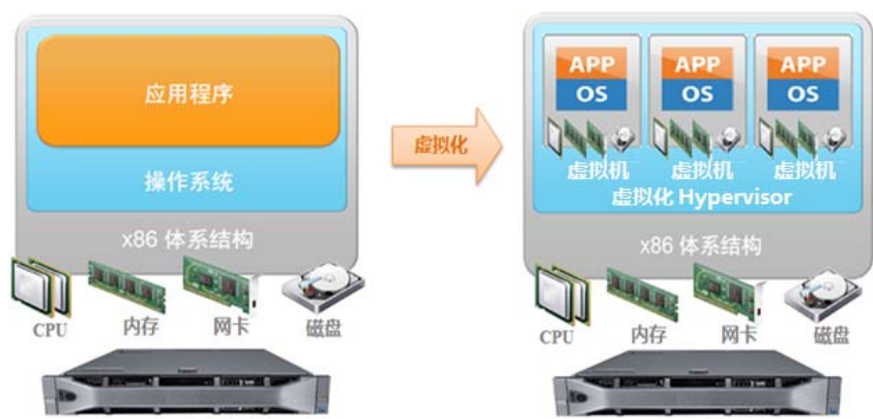


图 4 服务器虚拟化示意图

当前，虚拟化技术可分为服务器虚拟化、存储虚拟化、网络虚拟化、桌面虚拟化等类型，本文档主要涉及服务器虚拟化技术，其它虚拟化技术不做详述。

(1) 服务器虚拟化

如上图所示，服务器虚拟化就是使软件和硬件相互分离，把软件从主要安装硬件中分离出来。它可以在服务器架构中的多个位置实施虚拟化，包括应用程序与操作系统之间或操作系统与硬件之间，后者指位于下层的虚拟化软件通过空间上的分割、时间上的分时以及模拟，抽象出一个虚拟的硬件接口，向上层操作系统提供一个与它原先期待一致的服务器硬件环境，使得上层操作系统可以直接运行在虚拟环境上，可允许多个操作系统同时运行在单个物理服务器上。

(2) 裸金属架构

裸金属架构（Bare Metal）最早由Robert Goldberg于1973年在其关于虚拟机

体系架构的研究报告中提出。Goldberg将Hypervisor分为两类(Type 1和Type 2)，Type 1型的Hypervisor指虚拟资源到物理资源仅需翻译转换一次；Type 2型的Hypervisor指虚拟资源到物理资源需翻译转换两次。随着虚拟化技术的发展，业界将Type 1型的Hypervisor称为裸金属（Bare Metal）架构，通俗的理解为Hypervisor直接运行在硬件上，将Type 2型的Hypervisor称为“Hosted”，通俗的理解为运行在OS上。

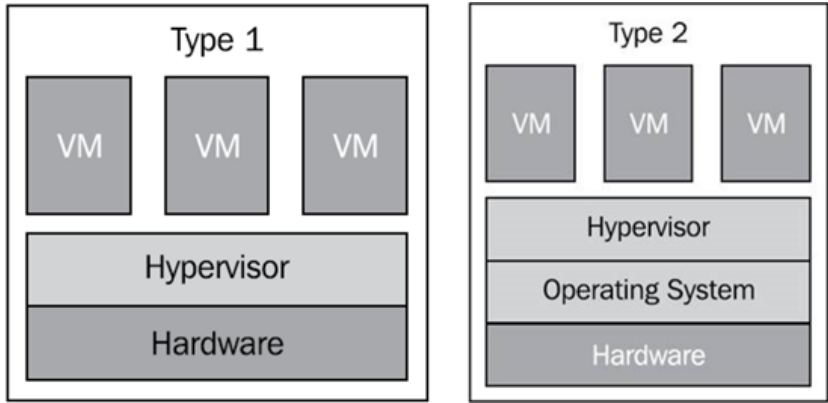


图 5 Type 1 型与 Type 2 型 Hypervisor 架构比较

Type 1（裸金属架构型）—— Hypervisor直接运行在硬件上，使用和管理底层的硬件资源，虚拟机Guest OS对真实硬件资源的访问都要通过Hypervisor来完成，作为底层硬件的直接操作者，Hypervisor拥有硬件的驱动程序。

Type 2（宿主型）—— Hypervisor之下还有一层宿主操作系统，由于虚拟机Guest OS对硬件的访问必须经过宿主操作系统，因而带来了额外的性能开销。

（3）虚拟化资源

服务器虚拟化将物理资源主要虚拟为三种类型的虚拟资源：虚拟CPU、虚拟内存和虚拟 I/O 设备。

■ 虚拟CPU

虚拟CPU是由CPU虚拟化技术实现。X86处理器一共有4个不同优先级，从Ring 0—Ring 3用来分隔操作系统软件和应用软件。Ring 0的优先级最高，Ring 3最低。Ring 0用于操作系统内核，Ring 1和Ring 2用于操作系统服务，Ring 3用于应用程序。那些只能在处理器的最高特权级（Ring 0）执行的指令称之为特权指令，一般可读写系统关键资源的指令决大多数都是特权指令。如果执行特权指令时处理器的优先级不在Ring 0，通常会引发一个异常，操作系统会捕获这种异常。

常并通过陷入来处理这个非法指令。

在Intel推出VT-x硬件辅助虚拟化技术之前，CPU虚拟化方法就是将虚拟机Guest OS运行在非特权级(Ring 3)，而将Hypervisor运行于最高特权级(Ring 0)。虚拟机Guest OS的大部分非特权指令仍可以在物理CPU上直接运行，当执行到特权指令时，就会陷入到Hypervisor模拟执行。由于X86处理器指令集中有十多条读写系统关键资源的指令不是特权指令，长期以来针对X86的CPU虚拟化实现大致分为两类，即以VMWare为代表的全虚拟化和以Xen为代表的半虚拟化。两者区别主要在对非特权敏感指令的处理上，全虚拟化采用的是动态监测捕获方法，即运行时监测，捕捉后在Hypervisor中模拟；而半虚拟化则主将所有涉及的非特权敏感指令全部替换，这样就避免了不能捕获的大量陷入过程。

以Intel VT-x硬件辅助虚拟化技术推出后，该技术增加了两种处理器工作模式：根(Root)操作模式和非根(Non-root)操作模式。Hypervisor运行在Root操作模式下，而Guest OS运行在Non-root操作模式下。这两个操作模式分别拥有自己的特权级环，Hypervisor和虚拟机Guest OS分别运行在这两个操作模式的Ring 0。这样，既能使Hypervisor运行在Ring 0，也能使Guest OS运行在Ring 0。Root操作模式和Non-root操作模式的切换是通过新增的CPU指令(VMXON,VMXOFF等)来完成。

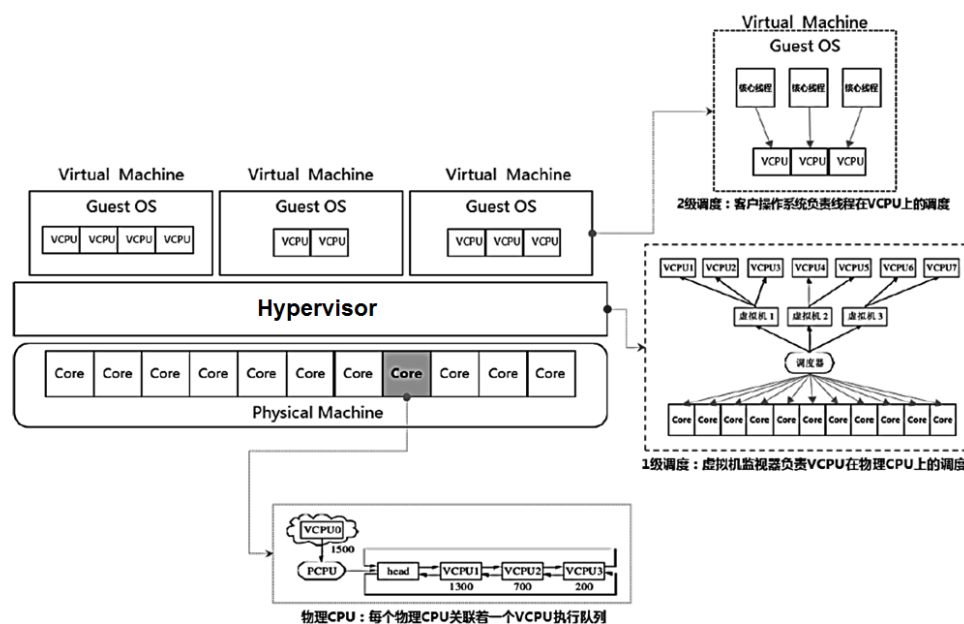


图 6 虚拟 CPU 两级调度示意图

如上图所示，Hypervisor和虚拟机Guest OS构成了虚拟CPU的两级调度框架。

虚拟机Guest OS负责第2级调度，即线程或进程在虚拟CPU上的调度，Guest OS中的核心线程映射到相应的虚拟CPU上。Hypervisor负责第1级调度，即虚拟CPU在物理CPU上的调度。两级调度的调度策略和机制不存在依赖关系。Hypervisor中的虚拟CPU调度器负责物理CPU在各个虚拟机之间的分配与调度。虚拟CPU可以采用分时复用或空间复用策略调度在一个或多个物理CPU执行，也可以与物理CPU建立一对一固定的映射关系。

服务器虚拟化系统可以通过设置物理CPU核与虚拟CPU的配比值，限制物理服务器可供应的虚拟CPU数量，每台物理服务器可以提供的最大虚拟CPU数量可参考以下计算公式：

$$\text{Max}_{\text{VCPU}} = (\text{CPU核数} \times \text{CPU数}) \times \text{配比值} (\geq 2)$$

■ 虚拟内存

虚拟内存是由内存虚拟化技术实现。服务器虚拟化系统的Hypervisor控制整个物理内存资源，通过页式内存管理，维护虚拟地址（VA）到物理地址（PA）、物理地址到机器地址（MA）的映射关系。

虚拟地址（VA）是指虚拟机Guest OS提供给应用程序使用的线性地址空间。

物理地址（PA）是指Hypervisor抽象的虚拟机看到的伪物理地址。

机器地址（MA）是指真实的机器地址，即物理服务器的内存地址。

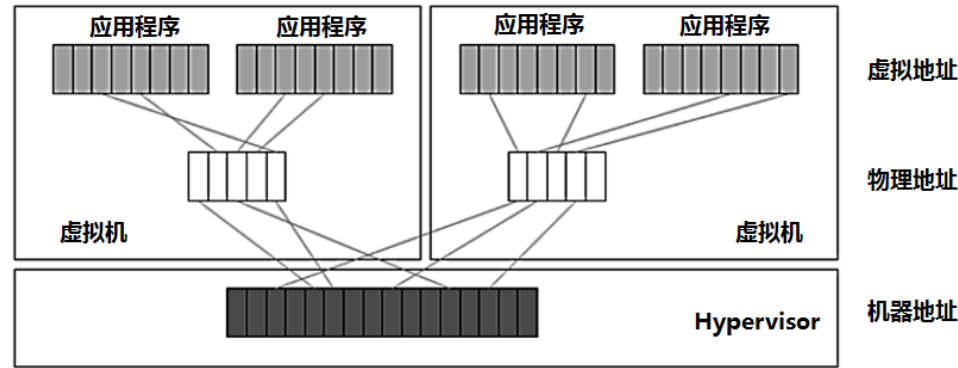


图 7 虚拟内存映射模型示意图

Hypervisor维护一套页表，负责PA 到MA 的映射。虚拟机Guest OS 维护一套页表，负责VA 到PA 的映射。实际运行时，应用程序访问VA，经虚拟机Guest OS 的页表转换得到PA，再由Hypervisor使用Hypervisor的页表将PA 转换为MA。

■ 虚拟I/O设备

虚拟I/O设备是由I/O设备虚拟化技术实现。服务器虚拟化系统的Hypervisor

通过I/O虚拟化技术来复用物理I/O设备资源，目前虚拟I/O设备主要有两种：设备模拟和设备直通。

设备模拟是通过软件模拟物理设备完全一样的接口，虚拟机Guest OS驱动无须修改就能驱动这个虚拟设备，虚拟机Guest OS为完成一次I/O操作要涉及到多个寄存器的操作，Hypervisor要拦截每个寄存器访问并进行相应的模拟，性能较低。

设备直通是直接将物理设备分配给某个虚拟机Guest OS，由Guest OS直接访问I/O设备，建立高效的I/O虚拟化直通道。

1.1.3 X86 虚拟化技术简介

当前，维基百科列举的虚拟化技术有超过60种，基于X86（CISC）体系的超过50种，也有基于RISC体系的，其中有4种虚拟化技术是当前最为成熟而且应用最为广泛的，分别是：商业的VMware和Hyper-V、开源的XEN和KVM。

(1) VMware

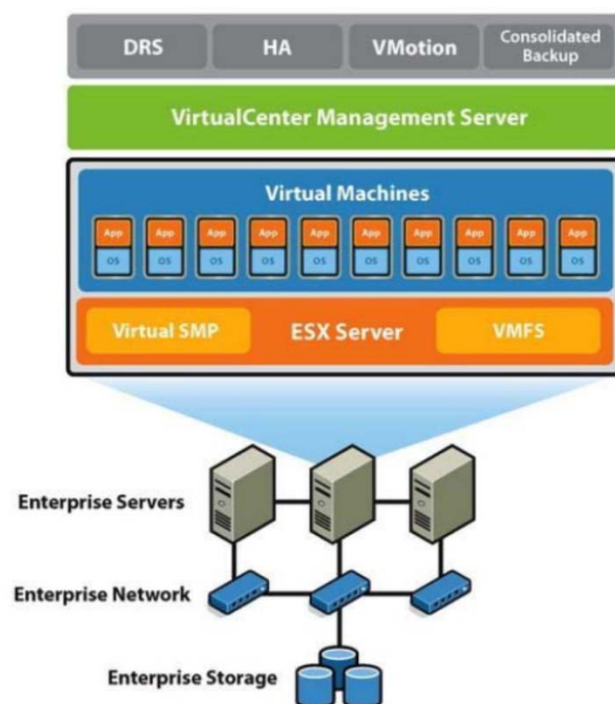


图 8 VMware 虚拟化技术架构示意图

VMware成立于1998年，1999年发布桌面版VMware Workstation产品，2001年发布企业级X86虚拟化产品ESX 1.0，2003年推出了vCenter，提供VMotion、DRS、HA等虚拟化调度功能，后又推出ESX的简化版ESXi，2011年推出管理

VMware ESX/ESXi 服务器环境的虚拟化产品集合VMware Infrastructure，后更名为VMware vSphere，并对外定义为云操作系统。2004年VMware被EMC收购，2016年EMC被DELL收购。VMware是最成功和影响最广泛的企业级X86虚拟化产品。

(2) Hyper-V

Hyper-V是微软推出的X86服务器虚拟化技术，首个版本于2008年7月发布，Hyper-V有两种发布版本：一是独立版，如Hyper-V Server 2008，以命令行界面实现操作控制；二是内嵌版，如Windows Server 2008，Hyper-V作为一个可选开启的角色。

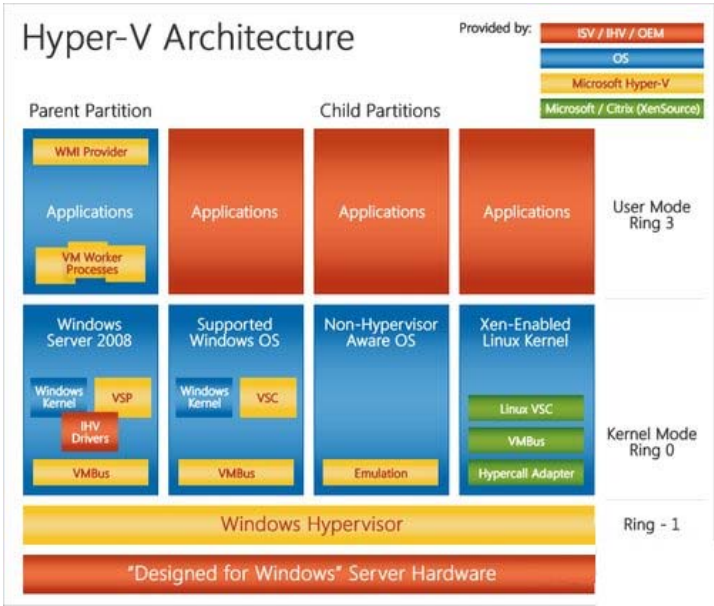
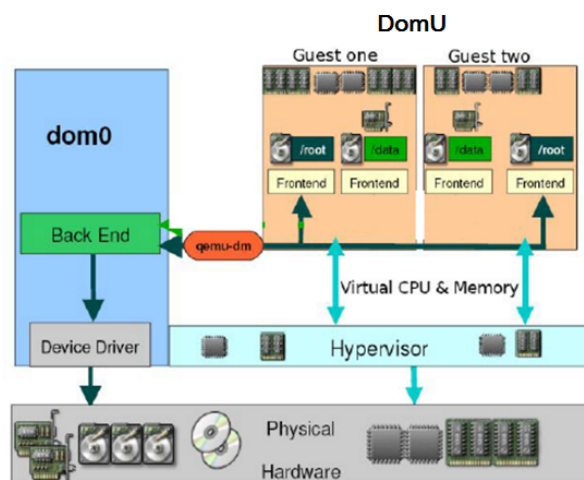


图 9 Hyper-V 虚拟化技术架构示意图

一旦在Windows Server 2008中开启了Hyper-V角色，系统会要求重新启动服务器。在这次重新启动过程中，Hyper-V的Hypervisor接管了硬件设备的控制权，先前的Windows Server 2008则成为Hyper-V的首个虚拟机，称之为父分区。Hypervisor仅实现了CPU的调度和内存的分配，而父分区控制着I/O设备，它通过物理驱动直接访问网卡、存储等。

(3) XEN

XEN最初是剑桥大学XenSource的一个开源研究项目，2003年9月发布了首个版本XEN 1.0，2007年XenSource被Citrix公司收购，开源XEN由www.xen.org组织继续推进。



XEN的Hypervisor是服务器经过BIOS启动之后载入的首个程序，然后启动一个具有特定权限的虚拟机，称之为Domain 0(简称Dom 0)。Dom 0的操作系统可以是Linux或Unix，Domain 0实现对Hypervisor控制和管理功能。在所承载的虚拟机中，Dom 0是唯一可以直接访问物理硬件(如存储和网卡)的虚拟机，它通过本身加载的物理驱动，为其它虚拟机(Domain U，简称DomU)提供访问存储和网卡的桥梁。

KVM的全称是Kernel-based Virtual Machine，字面意思是基于内核虚拟机。其最初是由Qumranet公司开发的一个开源项目，2007年1月首次被整合到Linux 2.6.20内核中，并与内核一起发布；2008年，Qumranet被RedHat所收购，但KVM本身仍是一个开源项目，由RedHat、IBM等厂商支持。

图 11 KVM 虚拟化技术架构示意图

KVM是Linux内核中的一个可装载模块，其功能是将Linux内核转换成一个裸金属的Hypervisor。通过KVM模块的加载将Linux内核转变成Hypervisor，KVM在Linux内核的用户(User)模式和内核(Kernel)模式基础上 增加了客户(Guest)模式。Linux本身运行于内核模式，主机进程运行于用户模式，虚拟机则运行于客户模式。

上述4种X86虚拟化技术部分核心指标对比如下图所示。

	KVM	VMWare ESX	Microsoft HyperV	Critix XenServer
进程隔离	✓	✓	✓	✓
强制访问控制（MAC）	✓	✗	✗	✗
基于角色的访问控制	✓	✓	✓	✓
裸金属架构	✓	✓	✓	✓
认证	✓	✓	✓	✓
审计跟踪	✓	✓	✓	✓
CC安全评估级	✓ EAL4+ *	✓ EAL4+	✓ EAL4+	✓ EAL2+
CC安全测试用例集	✓	✗	✗	✗
FIPS 140-2 认证加密	✓	✓	✓	✗
源代码开放	✓	✗	✗	✓
资源控制	✓	✓	✓	✓
磁盘加密	✓	✓	✓	✓

▲注意：金航数码服务器虚拟化系统采用是KVM虚拟化技术。

1.2 虚拟化在涉密行业应用面临的问题

虚拟化是企业级数据中心基础架构的必然发展趋势，但是由于原有的BMB17、BMB20 等计算机系统分级保护安全标准缺少针对虚拟化系统的相关安全技术要求与保密测评要求，为了保证涉密信息系统的安全性，自 2015 年起国家保密局明确要求，涉密信息系统必须使用经过国家保密局科技测评中心检测的虚拟化产品，否则不受理现场测评和风评，军工企业等涉密单位数据中心虚拟化系统建设的迫切需求一直被限制。

为了解决虚拟化技术在涉密信息系统中安全应用问题，国家保密局启动涉密

信息系统虚拟化技术标准制定工作，至 2017 年 5 月涉密信息系统虚拟化技术相关标准 BMB29、BMB30 正式发布，金航数码服务器虚拟化系统作为第一批通过测评的涉密服务器虚拟化产品，将广泛适用于军工企业等涉密虚拟化系统建设。

1.3 金航数码服务器虚拟化系统

金航数码服务器虚拟化系统 V5 产品是由金航数码科技有限责任公司（金航数码）研制的，主要定位于军工、企事业单位数据中心关键业务虚拟化平台领域，采用裸金属架构和全虚拟化的 KVM 技术，在开源基础上实现持续安全增强和优化，提供完整的服务器虚拟化全生命周期管理功能，充分发挥系统在安全性和可定制性的优势，提供数据中心关键业务对于高性能、高可靠、安全性和高可适应性上的各种虚拟化功能要求，产品满足分级保护安全管理体系及 BMB30 标准安全要求，基于产品构建的虚拟化系统符合 BMB29 标准安全要求。

1.4 参考标准

- GB/T 25069-2010 信息安全技术 术语
- BMB 17 涉及国家秘密的信息系统分级保护要求
- BMB 20 涉及国家秘密的信息系统分级保护管理规范
- BMB 29 涉密信息系统虚拟化技术应用安全保密要求
- BMB 30 涉密信息系统服务器虚拟化和桌面虚拟化产品安全保密技术要求
- GB 17859-1999 计算机信息系统安全等级保护划分准则
- GB/T 25058-2010 信息系统安全等级保护实施指南
- GB/T 22239-2008 信息系统安全等级保护基本要求

2 产品体系架构

2.1 总体架构

金航数码服务器虚拟化系统总体上包括三大类逻辑组件：虚拟资源管理、安全保密管理和安全审计管理。



图 12 金航数码服务器虚拟化系统总体架构示意图

■ 虚拟资源管理

金航数码服务器虚拟化系统虚拟资源管理逻辑组件，由虚拟化基础设施服务全生命周期管理相关的计算、网络、存储、模板&镜像、调度任务、用户、运维、服务&API等八大类资源系统管理功能模块构成，提供完整的服务器虚拟化系统管理能力。

■ 安全保密管理

金航数码服务器虚拟化系统安全保密管理逻辑组件，由三员授权管理、密级管理、计算资源安全管理、存储资源安全管理、网络资源安全管理等安全保密功能模块构成，提供符合机密级及其以下分级保护安全管理体系及BMB29、BMB30等标准安全要求的安全保密管理能力。

■ 安全审计管理

金航数码服务器虚拟化系统安全审计管理逻辑组件，由三员分权审计、系统日志审计、计算资源安全审计、存储资源安全审计、网络资源安全审计等安全审计功能模块构成，提供符合机密级及其以下分级保护安全管理体系及BMB29、BMB30等标准安全要求的安全审计管理能力。

2.2 功能架构

金航数码服务器虚拟化系统包含 200 多项系统管理功能、60 多项安全保密与审计管理功能，系统功能架构如下图所示。



图 13 金航数码服务器虚拟化系统功能结构示意图

2.3 物理架构

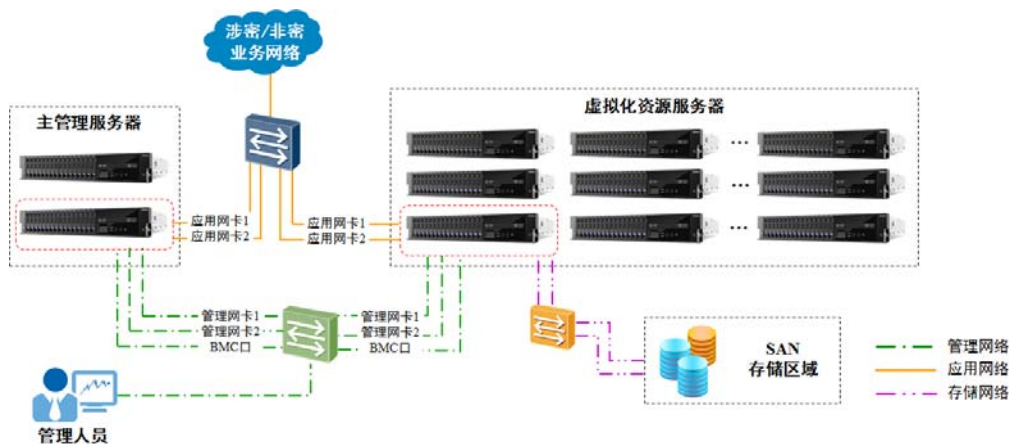


图 14 金航数码服务器虚拟化系统物理部署架构示意图

金航数码服务器虚拟化系统物理部署架构如上图所示，包括主管理服务器、

虚拟化资源池服务器、SAN 存储以及相关交换网络（应用交换网、管理交换网、存储交换网等）。

2.3.1 主管理服务器

主管理服务器实现对多个资源池所属的大量虚拟化资源服务器进行统一虚拟化管理、资源调度和运行监控。为了确保服务质量和处理性能，主管理服务器通过专门的管理网与受控资源服务器进行通信。为了避免单点故障，主管理服务器采用 HA 高可用集群部署，同时即使主管理服务器全部中断服务，也不会影响虚拟化资源服务器及其中的虚拟机正常运行。

2.3.2 虚拟化资源服务器

虚拟化资源服务器又称为宿主服务器，每台宿主服务器构成一个最小的可划分虚拟机的资源单位，多台宿主服务器可以构成一个资源池。如下图所示，虚拟化资源服务器通过应用网连接涉密或非密业务网络，使虚拟机 Guest OS 对外提供各种应用程序服务；通过管理网连接主管理服务器，接受主管理服务器的调度控制、运行监控、安全管控及基于 IMPI 的远程管理；通过存储网连接 SAN 存储区域，如果采用 FC-SAN 则为光纤存储交换网，应采用 8Gb 以上 HBA 卡，如果采用 IP-SAN 则为 10Gb 以上以太网。

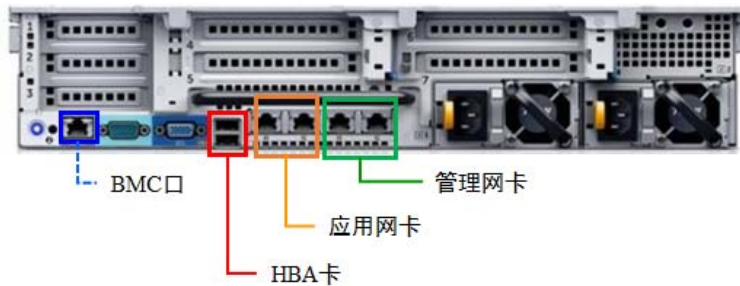


图 15 虚拟化资源服务器网络接口

2.3.3 SAN 存储

SAN 存储（FC-SAN/IP-SAN）用于集中存放虚拟化资源服务器中的虚拟资源数据。SAN 存储通过专门的存储网（FC 光纤存储交换网或 10Gb 以上以太网）连接虚拟化资源服务器。

2.3.4 交换网络

金航数码服务器虚拟化系统的物理交换网络包括应用交换网（应用网）、管理交换网（管理网）、存储交换网（存储网）等。

■ 应用网

应用网是金航数码服务器虚拟化系统连接涉密或非密业务的专用网络，使虚拟机 Guest OS 对外提供各种应用程序服务，一般采用 10Gb 以上以太网。

■ 管理网

管理网是金航数码服务器虚拟化系统对虚拟化资源服务器进行调度控制、运行监控、安全管控等的专用网络，服务器的 BMC 口也连接到应用网，一般采用 1Gb 以上以太网。

■ 存储网

存储网是金航数码服务器虚拟化系统连接 SAN 存储（FC-SAN/IP-SAN）的专用网络。

3 产品功能特性

3.1 总体技术特点

金航数码服务器虚拟化系统通过对 Linux 3.10 内核及 KVM 虚拟化组件进行持续安全加固、功能扩展、性能优化，着力打造自主可控、安全开放、高效稳定的服务器虚拟化系统，总体具有高可控性（High Controllability）、高安全性（High Security）、高性能（High Performance）、高可用性（High Availability）、高可管理性（High Maintainability）、高可定制性（High Customization）等 6H 技术特点。

3.2 高可控性（High Controllability）

3.2.1 全部源代码自主可控

金航数码服务器虚拟化系统基于 Linux 内核 3.10 版本进行深度定制，全部源代码自主可控，去除了与 KVM 虚拟化不相关的功能模块，并对已知的 Linux 内核安全漏洞进行修复。系统通过安全访问控制策略来限制宿主服务器系统用户的登录行为，系统只允许指定的非特权用户进行登录，所有非授权用户的登录操作（包括本地/远程）都将被系统拒绝。

3.2.2 自主分布式调度任务执行引擎

金航数码服务器虚拟化系统采用全自主的分布式调度任务执行引擎，以可视化的方式对资源调度任务的执行过程进行跟踪监视，可对系统中发生的故障进行快速定位和原因分析，保证大规模虚拟化资源管理中的可维性与可扩展性。

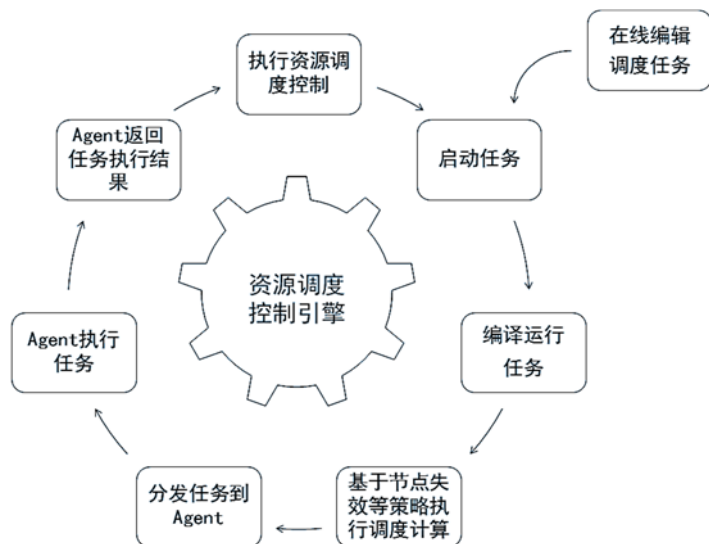


图 16 虚拟化资源调度任务执行过程

■ 调度任务执行监控

金航数码服务器虚拟化系统支持实时监控所有调度任务执行的过程，查看任务执行概览、任务分布、执行结果等实时统计结果。



图 17 调度任务执行概览示意图

3.3 高安全性（High Security）

3.3.1 三员分权管理

金航数码服务器虚拟化系统基于三员分权管理，三权分立将超级管理员权限分配给系统管理员、安全保密管理员、安全审计员，“三员”相互独立、相互制约，配合管理，可以有效的加强系统涉密信息的保密管理，减少安全风险。



图 18 三员分权管理示意图

- 系统管理员——系统日常运行管理与监控；负责资源（资源池、宿主机、虚拟机、网络、存储、模板、镜像等）创建、删除、修改；用户创建（无权限）、停用。
- 安全保密管理员——系统日常安全运行管理与监控；负责对用户资源授权管理、用户权限分配管理、安全策略的配置管理。
- 安全审计员——负责审计系统管理员和安全保密管理员的操作行为；参与关键操作的审核。

3.3.2 密级管理

金航数码服务器虚拟化系统支持虚拟化密级管理，可对资源池、物理机、虚拟机、存储、网络等资源进行密级管理。

资源池拥有相同密级的服务器、以及相同密级的存储。各个资源池中存储以及服务器物理分离，不同密级资源池不能挂载同一存储。

资源池名称	资源池类型	资源池文件类型	密级	资源池描述	资源池路径	总容量(GB)	剩余容量(GB)
1 ProRepo	高可用类型	逻辑卷	内部	对外演示环境	/dev/ProRepo	3499.99	4.99
2 lfsRepo	本地类型	文件	失控	本地存储资源池	/lfsRepo	5382.05	2910.26
3 localRepo	本地类型	文件	内部	本地资源池	/localrepo	1851.87	1756.79
4 lvmRepo	高可用类型	逻辑卷	内部	高可用存储资源池	/dev/lvmRepo	2500.04	18.04
5 mirrors	网络类型	分布式文件	开放	存储镜像的资源池	/var/opt/vsipa4s/agent/mirrors	3389.74	1324.90
6 template	网络类型	分布式文件	开放	模板资源池	/var/opt/vsipa4s/agent/template	3389.74	1324.90
7 vmbackup	网络类型	分布式文件	开放	备份资源池	/var/opt/vsipa4s/agent/vmbackup	3389.74	1324.90
8 vmbackup1	网络类型	分布式文件	内部	备份资源池(一级)	/var/opt/vsipa4s/agent/vmbackup1	3389.74	1324.90
9 vmbackup2	网络类型	分布式文件	失控	备份资源池(二级)	/var/opt/vsipa4s/agent/vmbackup2	3389.74	1324.90
10 vms	高可用类型	分布式文件	内部	分布式存储资源池	/vsipa4s/vms	3389.74	1324.90
11 vsRepo	高可用类型	逻辑卷	开放	办公桌面资源池	/dev/vsRepo	3072.02	2147.02

图 19 资源池资源密级管理

服务器要修改密级，需要退出资源池，修改密级配置，并由管理端进行重新认证，根据信息重新颁发证书。

物理机名称	物理机状态	密级	CPU核数 (个)	内存容量/已分配	本地可用磁盘 (GB)	CPU利用率	内存利用率	认证状态	机柜	位置
1 2950-2		受控	8	15.5 GB / 10.0 GB	389.09	1%	41%	通过认证	3#	5
2 2950-3		受控	8	15.5 GB / 10.0 GB	6.18	11%	79%	通过认证	4#	6
3 420-2		受控	24	62.9 GB / 63.0 GB	807.36	29%	87%	通过认证	1	1
4 420-3		受控	24	62.9 GB / 63.0 GB	61.04	1%	19%	通过认证		
5 420-4		受控	24	62.9 GB / 63.0 GB	31.95	4%	57%	通过认证		
6 420-5		受控	24	62.7 GB / 20.0 GB	149.71	8%	31%	通过认证		
7 420-6		受控	24	62.9 GB / 18.0 GB	349.77	3%	27%	通过认证		
8 6220-1		受控	24	62.9 GB / 44.0 GB		5%	50%	通过认证	5	3#
9 6220-2		受控	24	62.9 GB / 56.0 GB		3%	72%	通过认证		

图 20 宿主服务器资源密级管理

在资源池中创建的虚拟机自动设置为资源池相同密级。虚拟机密级不能进行修改，虚拟机跨资源池在线或者离线迁移时，系统要进行密级流向控制，仅允许相同密级间迁移。

编号	虚拟机名称	所属资源池	运行模式	磁盘加密	操作系统	密级	存储位置	状态	虚拟机 CPU 及利用率	内存及利用率	网络上下行
1	DEV1048	对外演示环境	预加载模式	否	Windows 7 Ultimate	内部	存储	停止	6220-3 2核	2.0 GB	0 B/S / 9 KB/S
2	DEV1049	对外演示环境	预加载模式	否	Windows 7 Ultimate	内部	存储	停止	2核	2.0 GB	
3	DEV1050	对外演示环境	预加载模式	否	Windows 7 Ultimate	内部	存储	停止	2核	2.0 GB	
4	DEV1051	对外演示环境	预加载模式	否	Windows 7 Ultimate	内部	存储	停止	2核	2.0 GB	
5	DEV_1027	王的虚拟机	预加载模式	否	Windows 7 Ultimate	受控	存储	运行中	6220-3 8核	8.0 GB	51 B/S / 8 KB/S
6	DEV_1028	DEV_1028	预加载模式	否	Windows 7 Ultimate	受控	存储	运行中	6220-2 8核	8.0 GB	46 B/S / 8 KB/S
7	DEV_1030	DEV_1030	预加载模式	否	Windows 7 Ultimate	受控	存储	运行中	6220-4 2核	4.0 GB	46 B/S / 0 B/S
8	DEV_XuBin_Arch	DEV_XuBin_Arch	预加载模式	否	Other operating system	受控	存储	运行中	6220-3 4核	8.0 GB	0 B/S / 17 KB/S
9	DEV_lifan	DEV_lifan	预加载模式	否	Windows 7 Ultimate	内部	存储	运行中	6220-3 4核	4.0 GB	15 B/S / 8 KB/S
10	DEV_liuan	DEV_liuan	预加载模式	否	Windows 7 Ultimate	内部	存储	运行中	6220-2 2核	2.0 GB	70 B/S / 8 KB/S

图 21 虚拟机资源密级管理

存储设定密级后，只能由相同密级宿主机挂载，不同密级存储在物理上隔离，宿主机只能挂载相同密级存储。

Lun UUID	Lun 名称	密级	用途	总大小 (GB)
1 36090a0b80070c4959854b57d905e0d89	裸设备2号		裸设备	100.00
2 36090a0b80070f4d28e541507905eed23	四川办公环境存储		共享存储	2048.01
3 360fff13a0e41715cc83383b6ca044074	测试环境存储2			500.01
4 360fff17aed4036aac970f504179b647e	裸设备测试1号3	内部		10.00
5 360fff17aed40463e8b3e5512159b0421	内网办公1	受控		500.01
6 360fff17aed4056478b3eb512159b046c	内网办公2	受控		500.01
7 360fff17aed4076db4f752505179b4498	内网办公3	内部		500.01
8 360fff17aed40c5a991405579179b240f	内网办公4	受控		500.01
9 360fff17aed40d64e8b3e3513159b2403	内网办公5	受控		500.01
10 360fff17aed40e6438b3e8512159b24eb	内网办公6	受控		500.01
11 360fff17aed40f6b091408579179b446d	内网办公7	受控		500.01
12 360fff17aed904c32a92ae5b729b4493	测试环境存储3	开放		500.01
13 360fff17aed809c2ca92ab5b729b440e	测试环境存储4	开放		500.01

图 22 存储资源密级管理

分布式虚拟交换机创建时需要设定密级，虚拟交换机只能与相同密级资源池建立关联。

分布式交换机管理						
刷新 新增 配置 删除 实时监控 查看属性						
编号	名称	描述	设备编号	控制器状态	实时监控开关	类型
1	switch.ProStrgSwitch	受控	对外演示环境存储网	br-strg	未开启	分布式交换机
2	switch.ProSwitch	受控	对外演示环境	br-mgmt	开启	分布式交换机
3	switch.TestSwitch	内部	测试交换机	br4	未开启	分布式交换机

图 23 虚拟交换机资源密级管理

3.3.3 虚拟机安全隔离管理

金航数码服务器虚拟化系统基于虚拟机安全隔离管理机制，在 CPU 虚拟化、内存虚拟化、I/O 设备虚拟化等技术（参考 1.1.2 节 虚拟化技术工作原理内容）的隔离机制基础上,进一步通过 Linux 内核的 SE Linux 实现强制访问控制(MAC)和多类别安全（MCS）标记功能。虚拟化系统在分配虚拟机资源时，将会为虚拟机进程、虚拟机所拥有的虚拟内存、虚拟磁盘、虚拟 I/O 设备等资源分配一个随机的强制访问控制和多类别安全标记，在强制访问控制策略的约束下，虚拟机仅能访问自身所拥有到的虚拟内存、虚拟磁盘、虚拟 I/O 设备等资源，无法跳出限制且对其他资源不具有任何读写权限，确保虚拟机之间虚拟 CPU、虚拟内存、虚拟 I/O 设备的强安全隔离。

3.3.4 虚拟网络安全隔离管理

金航数码服务器虚拟化系统支持虚拟交换网络安全隔离管理，系统提供分布式虚拟交换机，支持 VLAN、端口镜像、IP/MAC 绑定等功能。

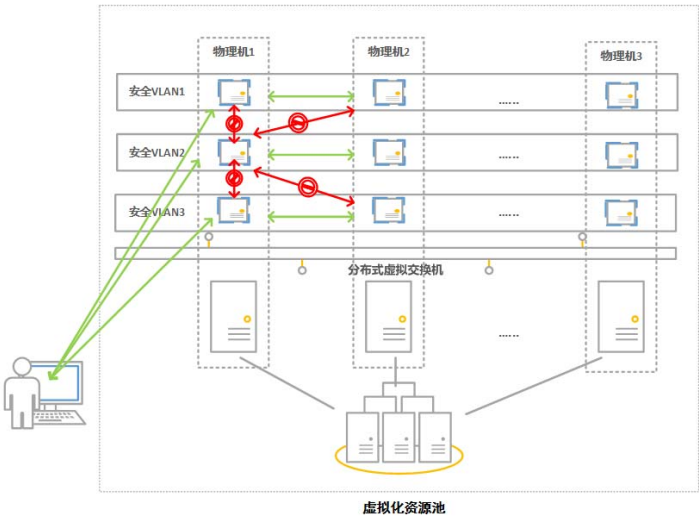


图 24 虚拟网络安全隔离管理示意图

如上图所示，启用虚拟网络隔离机制，可将同一 VLAN 中的虚拟机调度到

资源池中不同的物理服务器上运行，确保同一物理服务器上运行的全部为不同 VLAN 中的虚拟机。

虚拟交换机默认禁止了网卡的混杂模式，但可以通过端口镜像（PortMirror）监听指定的虚拟网卡，实现虚拟网络的病毒防护和异常侦测。

3.3.5 虚拟防火墙管理

金航数码服务器虚拟化系统提供虚拟防火墙管理功能，通过虚拟防火墙对虚拟机网络按照 IP、端口、协议等设置网络控制策略。

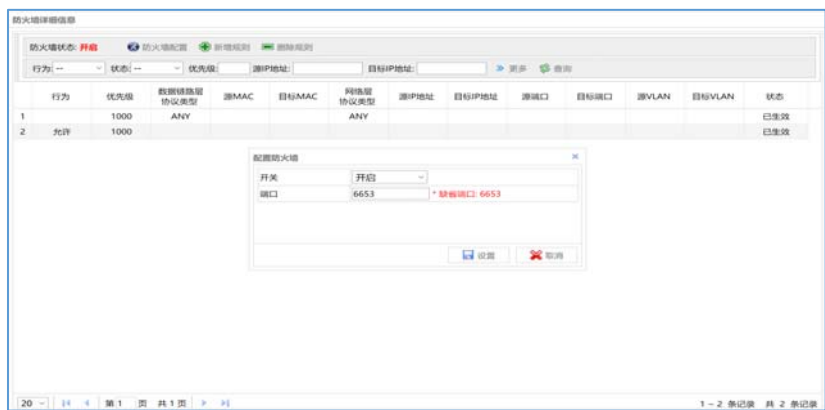


图 25 虚拟防火墙管理

3.3.6 虚拟机数据加密保护管理

金航数码服务器虚拟化系统可使用多种加密算法（SM 国密算法、128 bit 的 AES 加密算法等）对虚拟机进行闭环加密保护，包括：虚拟机创建（在线安装）、虚拟机运行、虚拟机克隆、虚拟制作模板、虚拟机备份恢复（默认加密）、虚拟机导入导出、虚拟机迁移。

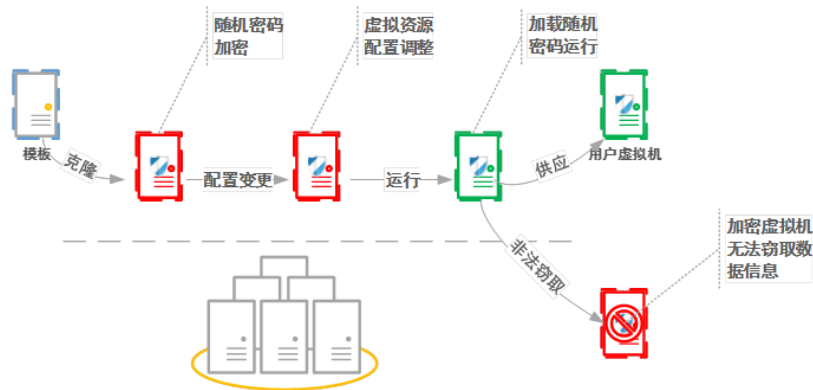


图 26 虚拟机加密保护闭环示意图

▲注意：加密虚拟机的克隆时间开销会比非加密虚拟机增加 30%左右，但对虚拟机运行时性能开销影响不超过 5%。

3.3.7 虚拟机剩余信息保护管理

金航数码服务器虚拟化系统基于虚拟机虚拟内存、虚拟磁盘的剩余信息保护管理机制，杜绝通过内存、存储等剩余信息攻击安全威胁。

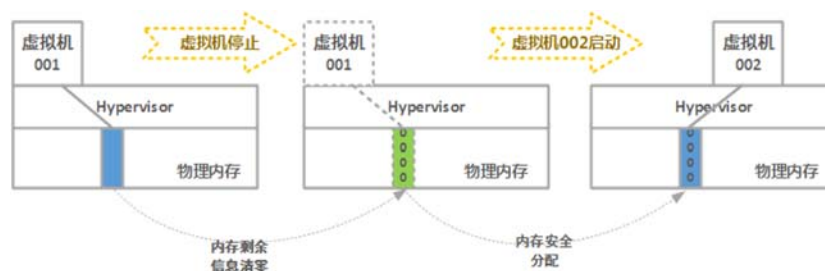


图 27 虚拟机内存剩余信息保护机制示意图

如上图所示，当关闭虚拟机时，系统将对释放的内存做强制写“0”处理，从而保障在新启动的虚拟机中无法检测到内存中的剩余有用信息。

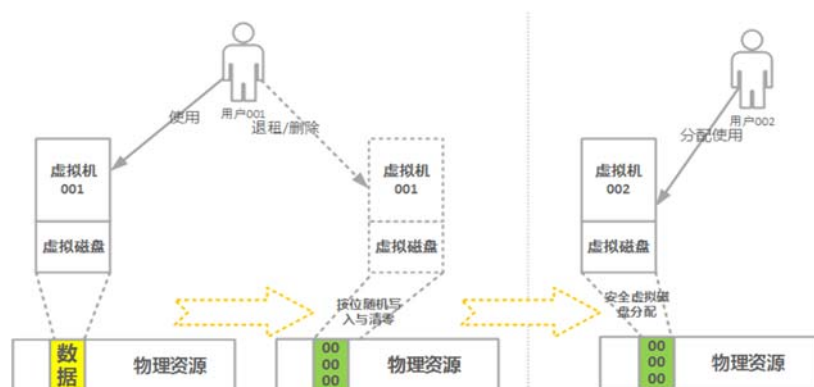


图 28 虚拟机磁盘剩余信息保护机制示意图

如上图所示，当删除虚拟机时，系统将首先确认虚拟磁盘对应的物理磁盘卷信息，并针对该虚拟机涉及到的每块磁盘先后进行按位随机数据写入与按位全“0”数据写入操作，通过随机数据与全“0”数据的三轮强制操作，完成虚拟机磁盘的“低级格式化”操作，实现对虚拟机磁盘剩余数据的保护。

3.3.8 虚拟机逃逸监控管理

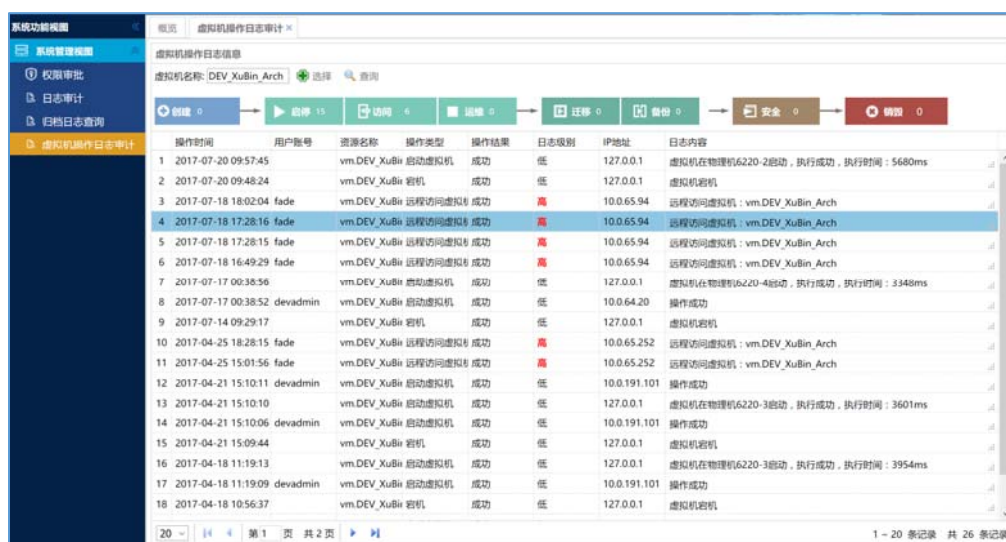
虚拟机逃逸是指利用虚拟机软件或者虚拟机中运行的软件的漏洞进行攻击，达到攻击或控制虚拟机宿主操作系统的目的。金航数码服务器虚拟化系统对已知

针对 KVM、QEMU 的逃逸攻击漏洞进行修复和安全加固，并持续从 NVD 等官方漏洞库跟踪和修复最新的虚拟机逃逸攻击漏洞。

金航数码服务器虚拟化系统提供虚拟机逃逸监控功能，通过 Linux 内核的 SE Linux 实现强制访问控制(MAC)，实现了对虚拟机访问未授权资源(如进程、文件、存储等)以及执行特权指令(如 SYSCALL、SYSRET 等)的行为监控。

3.3.9 虚拟机操作日志审计管理

金航数码服务器虚拟化系统基于三员分权安全审计机制，提供详细的虚拟机操作日志审计功能，并且审计日志具有防篡改能力，在审计日志内容被篡改或删除后，系统将日志数据恢复到篡改前的状态，分析记录篡改的类型。



操作时间	用户账号	资源名称	操作类型	操作结果	日志级别	IP地址	日志内容
2017-07-20 09:57:45	vm.DEV_XuBin	启动虚拟机	成功	低	低	127.0.0.1	虚拟机在物理机6220-2启动, 执行成功, 执行时间: 5680ms
2017-07-20 09:48:24	vm.DEV_XuBin	宕机	成功	低	低	127.0.0.1	虚拟机宕机
2017-07-18 18:02:04	fade	vm.DEV_XuBin	远程访问虚拟机	成功	高	10.0.65.94	远程访问虚拟机: vm.DEV_XuBin_Arch
2017-07-18 17:28:16	fade	vm.DEV_XuBin	远程访问虚拟机	成功	高	10.0.65.94	远程访问虚拟机: vm.DEV_XuBin_Arch
2017-07-18 17:28:15	fade	vm.DEV_XuBin	远程访问虚拟机	成功	高	10.0.65.94	远程访问虚拟机: vm.DEV_XuBin_Arch
2017-07-18 16:49:29	fade	vm.DEV_XuBin	远程访问虚拟机	成功	高	10.0.65.94	远程访问虚拟机: vm.DEV_XuBin_Arch
2017-07-17 00:38:56	vm.DEV_XuBin	启动虚拟机	成功	低	低	127.0.0.1	虚拟机在物理机6220-4启动, 执行成功, 执行时间: 3348ms
2017-07-17 00:38:52	devadmin	vm.DEV_XuBin	启动虚拟机	成功	低	10.0.64.20	操作成功
2017-07-14 09:29:17	vm.DEV_XuBin	宕机	成功	低	低	127.0.0.1	虚拟机宕机
2017-04-25 18:28:15	fade	vm.DEV_XuBin	远程访问虚拟机	成功	高	10.0.65.252	远程访问虚拟机: vm.DEV_XuBin_Arch
2017-04-25 15:01:56	fade	vm.DEV_XuBin	远程访问虚拟机	成功	高	10.0.65.252	远程访问虚拟机: vm.DEV_XuBin_Arch
2017-04-21 15:10:11	devadmin	vm.DEV_XuBin	启动虚拟机	成功	低	10.0.191.101	操作成功
2017-04-21 15:10:10	vm.DEV_XuBin	启动虚拟机	成功	低	低	127.0.0.1	虚拟机在物理机6220-3启动, 执行成功, 执行时间: 3601ms
2017-04-21 15:10:06	devadmin	vm.DEV_XuBin	启动虚拟机	成功	低	10.0.191.101	操作成功
2017-04-21 15:09:44	vm.DEV_XuBin	宕机	成功	低	低	127.0.0.1	虚拟机宕机
2017-04-18 11:19:13	vm.DEV_XuBin	启动虚拟机	成功	低	低	127.0.0.1	虚拟机在物理机6220-3启动, 执行成功, 执行时间: 3954ms
2017-04-18 11:19:09	devadmin	vm.DEV_XuBin	启动虚拟机	成功	低	10.0.191.101	操作成功
2017-04-18 10:56:37	vm.DEV_XuBin	宕机	成功	低	低	127.0.0.1	虚拟机宕机

图 29 虚拟机操作日志审计管理

系统功能概览

系统管理概览

权限审批

日志审计

归档日志查询

虚拟机操作日志审计

概览

日志审计

系统日志信息

校验日志

日志防护

用户姓名

关键字

IP地址

全部类型

级别:全部

结果:全部

开始日期:2017-07-31

结束日期:2017-08-01

查询

操作时间

1 2017-07-31 17:5

2 2017-07-31 17:5

3 2017-07-31 17:4

4 2017-07-31 17:4

5 2017-07-31 17:4

6 2017-07-31 17:4

7 2017-07-31 17:4

8 2017-07-31 17:4

9 2017-07-31 17:4

10 2017-07-31 17:3

11 2017-07-31 17:3

12 2017-07-31 17:3

13 2017-07-31 17:3

14 2017-07-31 17:3

15 2017-07-31 17:3

16 2017-07-31 17:34:39

17 2017-07-31 17:34:32

18 2017-07-31 17:34:10

20 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:39

20 - 1 2017-07-31 17:34:32

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-31 17:34:10

20 - 1 2017-07-3

图 30 日志防篡改检查

3.3.10 第三方安全审计支持

金航数码服务器虚拟化系统提供主动上报(Syslog)和被动轮询(WebService)的方式与国家保密局认可的第三方安全审计系统无缝集成。

A screenshot of the 'SYSLOG配置' (Syslog Configuration) window. It contains several input fields: '服务器地址' (Server Address) with '10.0.12.250' and a hint '输入IP地址,格式如192.168.0.1'; '服务器端口' (Server Port) with '1468' and a hint '默认514'; '服务协议' (Service Protocol) with a dropdown set to 'TCP'; 'Facility' with '128'; '证书文件路径' (Certificate File Path) with '/opt/vsip4x/server/configuration/server.jks'; '证书文件密码' (Certificate File Password) with masked characters; '对方信任列表文件路径' (Peer Trust List File Path) with '/opt/vsip4x/server/configuration/client.jks'; and '对方列表文件密码' (Peer List File Password) with masked characters. There are also icons for help and save in the top right.

图 31 Syslog 审计服务配置

3.3.11 第三方防病毒系统支持

金航数码服务器虚拟化系统支持与奇安信、360、安天等第三方软件无缝集成,包括集中式、轻代理等网络版或单机版防病毒系统。可为宿主机提供病毒与恶意代码防护的能力。

3.4 高性能 (High Performance)

3.4.1 虚拟化性能

金航数码服务器虚拟化系统可以通过设置资源池的物理 CPU 与 vCPU 的配比值 (默认值为 2),限制宿主服务器可供应的最大 vCPU 数量,充分保证虚拟化系统性能。

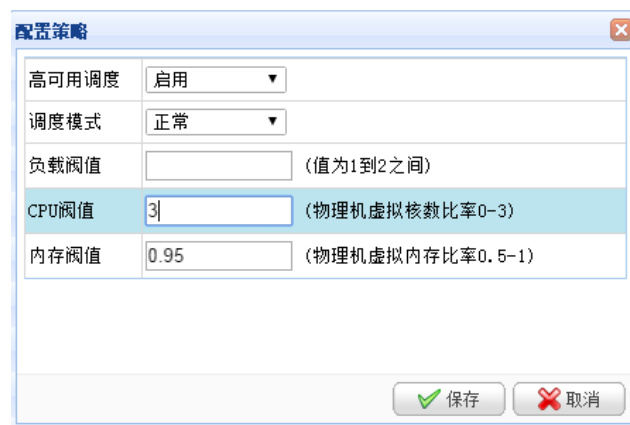
A screenshot of the '配置策略' (Configuration Strategy) window. It has a title bar with a close button. The window contains several settings: '高可用调度' (High Availability Scheduling) with a dropdown set to '启用' (Enable); '调度模式' (Scheduling Mode) with a dropdown set to '正常' (Normal); '负载阈值' (Load Threshold) with an empty input field and a hint '(值为1到2之间)'; 'CPU阈值' (CPU Threshold) with an input field containing '3' and a hint '(物理机虚拟核数比率0-3)'; and '内存阈值' (Memory Threshold) with an input field containing '0.95' and a hint '(物理机虚拟内存比率0.5-1)'. At the bottom, there are two buttons: '保存' (Save) with a green checkmark icon and '取消' (Cancel) with a red X icon.

图 32 资源池的物理 CPU 与 vCPU 的配比值设置

金航数码服务器虚拟化系统部分性能指标如下表所示。

单台虚拟机支持最大 vCPU 数量	128 个
单台虚拟机支持最大内存容量	1024GB
单台虚拟机支持最大虚拟磁盘容量	64TB
单台虚拟机支持最大虚拟磁盘数量	26 个
单台虚拟机支持最大虚拟网卡数量	8 个
单个高可用资源池支持物理机最大数量	100 台
单个高可用资源池支持虚拟机最大数量	5000 台
虚拟机（20GB）完全克隆创建时间	≦ 2 分钟
虚拟机（20GB）链接克隆创建时间	≦ 2 秒钟
虚拟机在线迁移时间	≦ 12 秒
虚拟机高可用调度恢复时间	≦ 2 分钟

3.4.2 虚拟 CPU 性能优化

金航数码服务器虚拟化系统支持基于 NUMA 架构的 CPU 性能优化技术，将一个物理 CPU 及与其绑定的内存地址称为一个 NUMA Node，系统将尽量使同一个虚拟机的 vCPU 尽量被分配到同一个 NUMA Node 中，避免虚拟机的 vCPU 跨不同的 NUMA Node，减少跨 NUMA Node 间的内存访问而导致的内存访问延迟，避免 CPU 资源分配不平衡引起的虚拟机性能瓶颈问题，特别适用于 Oracle、SQL Server 等重负载应用。

当虚拟机启动时，系统将根据实时的宿主机 CPU 和内存负载，选择一个负载较轻的 NUMA Node 运行该虚拟机，并将虚拟机的虚拟 CPU 调度范围限制在该 NUMA Node 上，如下图所示。

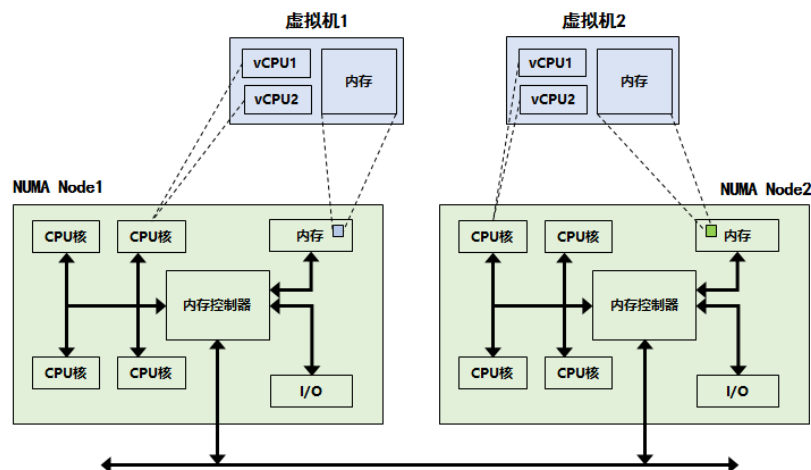


图 33 基于 NUMA 架构的 CPU 性能优化

3.4.3 虚拟内存性能优化

虚拟化系统一般会采用内存复用技术，以提高物理内存资源的利用效率，但是内存复用产生频繁的内存交换，会造成虚拟机内存及整体性能下降以及内存信息泄漏安全风险。

为了优化内存的使用，金航数码服务器虚拟化系统没有采用内存复用技术，而是基于内存气泡技术（Ballooning）对虚拟机内存地址空间进行动态调度，当虚拟机实际占用的物理内存过多时，系统调度虚拟机 Guest OS 从其闲置虚拟机物理内存链表中返回多余无用物理内存给气球；当虚拟机实际占用的物理内存资源不足时，系统调度虚拟机 Guest OS 必须回收一部分物理内存，以满足气球申请客户机物理内存的需要，提升物理内存利用率的同时，尽可能减少对虚拟机性能的影响，同时，在进行内存动态释放过程中，内存信息将被安全清除，确保内存信息的安全。

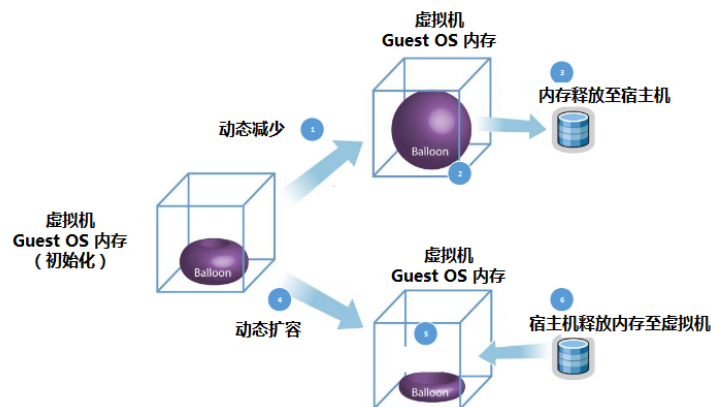


图 34 基于内存气泡技术（Ballooning）的虚拟内存性能优化

3.4.4 虚拟机 GPU 直通

金航数码服务器虚拟化系统支持 GPU 直通功能，通过 Intel VT-d 技术把物理 GPU 直接分配给虚拟机，使虚拟机能够完全拥有物理 GPU 的资源和 3D 图形处理性能。

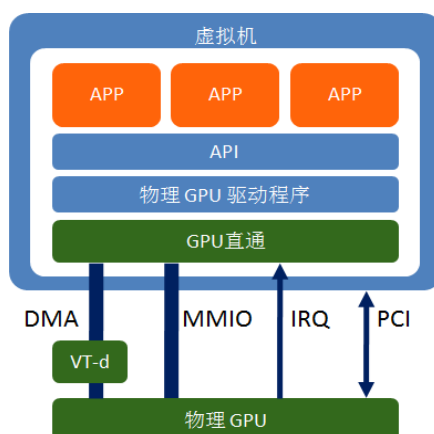


图 35 虚拟机 GPU 直通示意图

3.4.5 存储裸设备挂载

金航数码服务器虚拟化系统为虚拟机提供了一种机制来直接访问 SAN 存储系统上的 LUN，通过使用物理设备映射，可以让虚拟机识别 SCSI 磁盘，虚拟机直接通过 SCSI 命令操作裸存储设备，适用于 Oracle RAC 等需要高性能存储的应用。

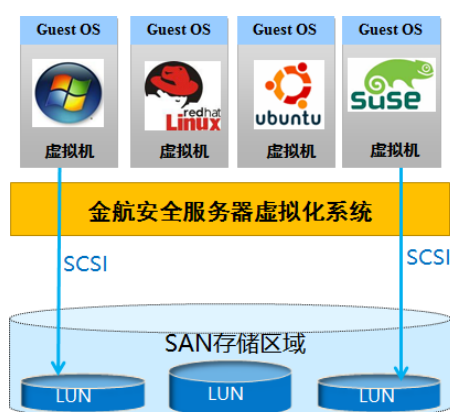


图 36 SAN 存储裸设备挂载示意图

3.5 高可用性（High Availability）

3.5.1 虚拟机在线迁移

金航数码服务器虚拟化系统提供虚拟机在线迁移（又称为热迁移）功能，虚拟机热迁移过程中，不中断虚拟机中应用服务，用户无感知，同时，系统在虚拟机迁移过程中，对源宿主服务器和目的宿主服务器之间传输的虚拟机状态数据进行加密，保证虚拟机迁移过程的安全性。虚拟机在线迁移过程如下图所示。

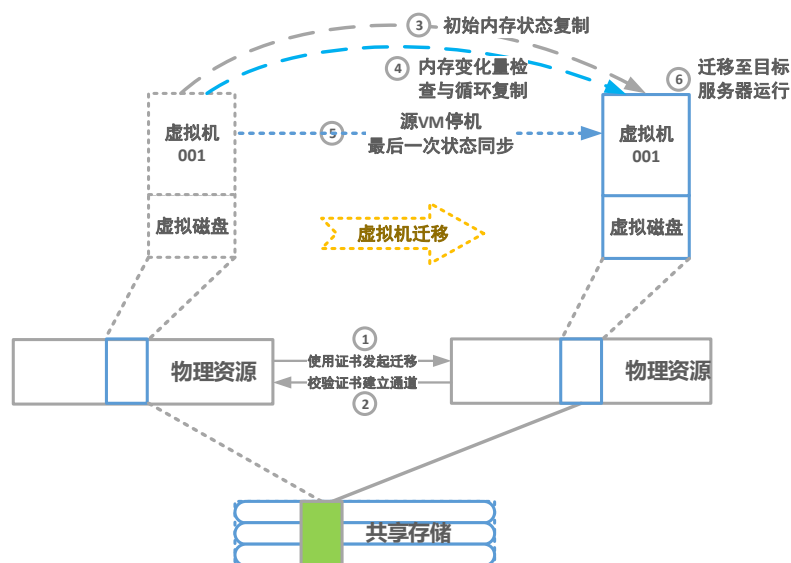


图 37 虚拟机在线迁移过程

（1）在虚拟机迁移开始时，虚拟机所在的源宿主服务器发起连接请求；

（2）目的宿主服务器在接收到源宿主服务器发来的请求后，对源宿主服务器的证书信息进行校验，如果校验通过，则源宿主服务器使用此加密通道开始执行虚拟机迁移操作；如果校验未通过，则连接被拒绝，迁移操作被中断；

（3）虚拟化系统通过宿主服务器之间建立的安全通道，将虚拟机的状态数据从宿主源服务器循环复制到目标服务器，此时虚拟机还在源宿主服务器上运行，第一个循环内将全部内存状态复制到目的宿主服务器上，在这个过程中，虚拟化系统会持续监视虚拟机内存的任何变化；

（4）后续的循环中，检查上一个循环中虚拟机内存是否发生了变化，假如发生了变化，那么 虚拟化系统会将发生变化的内存页重新复制到目的宿主服务器中，并覆盖掉先前的内存页；在这个阶段，虚拟化系统仍会持续监视虚拟机内存的变化情况；

(5) 当源宿主服务器与目的宿主服务器之间的内存差异接近零时，内存复制操作结束，暂停源虚拟机，在源虚拟机和目标虚拟机都停机的情况下，将最后一个循环的内存差异数据和源虚拟机设备的工作状态复制到目的宿主服务器；

(6) 最后将虚拟磁盘从源宿主服务器上解锁，切换到目的宿主服务器上，恢复目标虚拟机运行，完成虚拟机的在线迁移切换。

▲注意：金航数码服务器虚拟化系统支持多台虚拟机同时进行迁移。

3.5.2 虚拟机高可用调度

金航数码服务器虚拟化系统支持虚拟机高可用调度功能，提供针对宿主机故障和虚拟机内部故障的两种高可用调度机制：

(1) 当运行业务应用的虚拟机所属的宿主服务器出现故障时，避免漏判、误判故障，系统在 2 分钟之内启用高可用调度，将运行应用的虚拟机调度至其他宿主服务器上继续运行；

(2) 当运行业务应用的虚拟机内部宕机或遭遇非法关机时，系统将虚拟机重新在资源池中调度运行，确保虚拟机处于稳定运行状态。

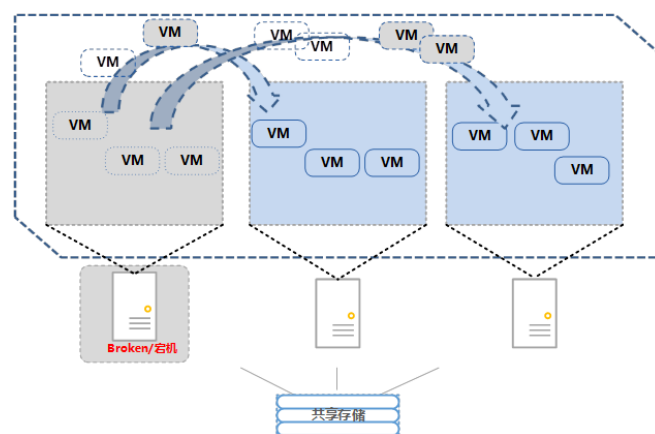


图 38 虚拟机高可用调度

3.5.3 虚拟机 FT 热备高可用

金航数码服务器虚拟化系统支持虚拟机 FT 热备高可用调度功能，可以为虚拟机（Master）创建一个备份虚拟机（Slave），在 Master 虚拟机与 Slave 虚拟机间进行虚拟机运行状态实时同步，从而保证虚拟机在运行过程中对外提供连续服务能力。在 FT 运行过程中，由 Master 虚拟机对外提供服务、接收外部输入、读

写数据磁盘，Slave 虚拟机实时同步 Master 虚拟机状态，但处于暂停模式。当 Master 虚拟机发生故障时，Slave 虚拟机可以进行毫秒级切换，业务无感知，

3.5.4 虚拟机操作系统传统高可用

金航数码服务器虚拟化系统支持传统操作系统高可用部署模式，当业务应用需要在操作系统层配置高可用，虚拟化系统提供“共享盘”的技术，为多个虚拟机提供共享数据，支持如 Windows 操作系统层的双机热备 HA 部署。

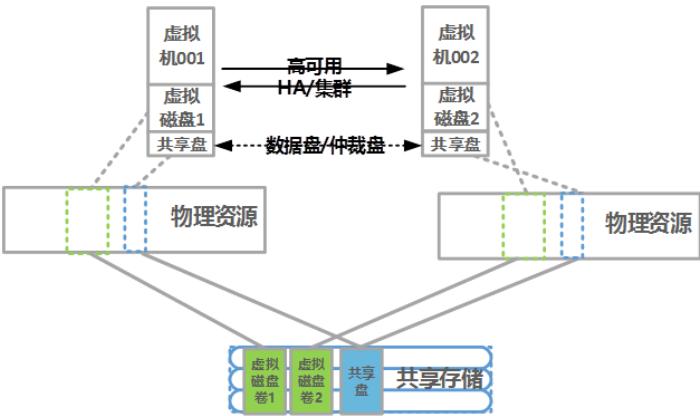


图 39 虚拟机传统操作系统高可用部署模式

3.6 高可管理性（High Maintainability）

3.6.1 虚拟机资源热扩展管理

金航数码服务器虚拟化系统支持虚拟机 vCPU、内存、虚拟网卡和虚拟磁盘的热插拔、热扩容管理。基于 VirtIO 的虚拟设备热插拔架构，动态给在线虚拟机增加或减少虚拟机资源，来实现虚拟机资源的动态可管理。

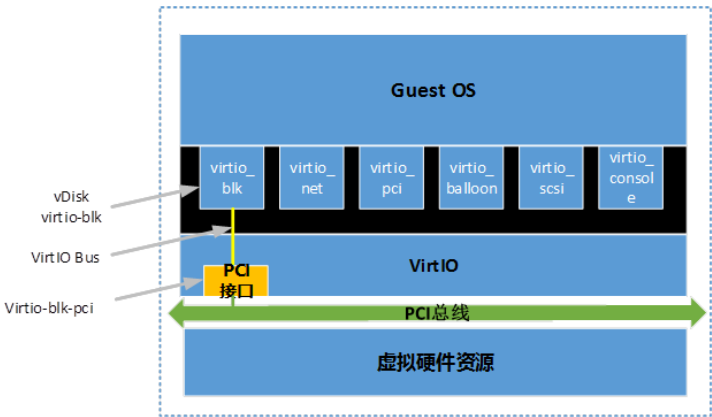


图 40 基于 VirtIO 的虚拟设备热插拔架构

▲注意：虚拟机CPU、内存的热增加功能，需要操作系统支撑，支持的操作系统包括：Windows 2008 Server、Windows 2012、SUSE Linux、RedHat、中标麒麟等；删除虚拟机的CPU、内存时，会在下次重启后生效。

3.6.2 虚拟化调度任务跟踪分析

金航数码服务器虚拟化系统支持对所有调度任务的任务实例 ID、任务名称、任务启动时间、任务所在宿主服务器、父任务、任务参数、任务执行状态进行跟踪分析。

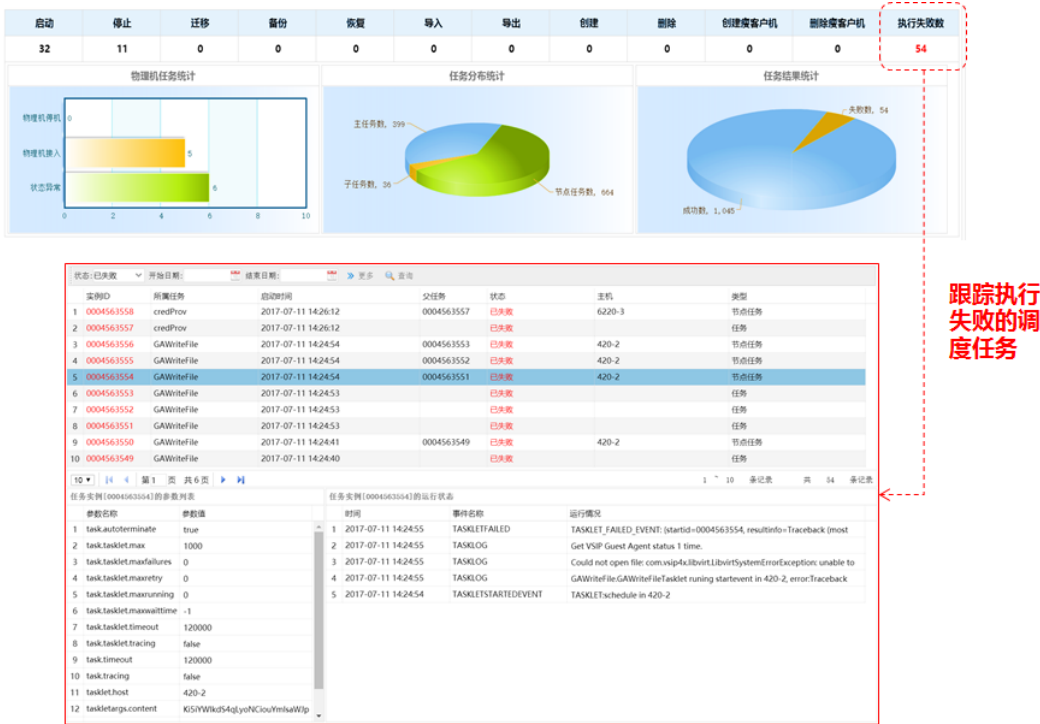


图 41 调度任务执行跟踪分析

3.6.3 资源运行状态高效监控

金航数码服务器虚拟化系统可对资源池存储可用容量、读写 I/O 最高的虚拟机进行监控，精心设计的主机健康矩阵视图，可秒级快速定位多个资源池中大规模宿主服务器、虚拟机中的状态异常对象。如下图所示，“绿色”表示运行正常的宿主服务器或虚拟机，“黄色”表示运行且告警状态的宿主服务器或虚拟机，“灰色”表示停止状态的虚拟机。

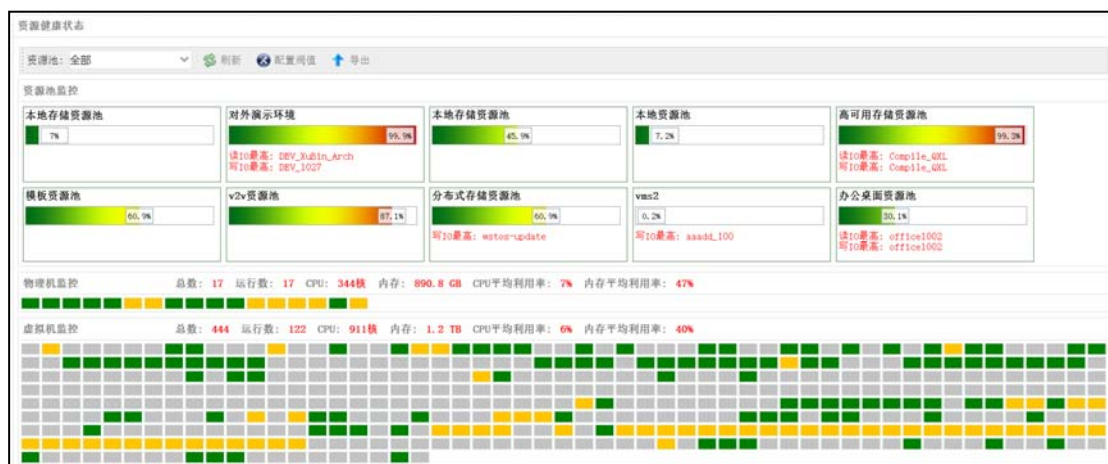


图 42 资源运行健康状态监控

3.6.4 系统告警管理

金航数码服务器虚拟化系统提供可配置阈值的自助告警管理功能，以及统一的监控、告警展现功能，也可以通过 Email 等发送告警内容，使运维人员实施监控和快速定位故障源。

系统提供的告警级别分为一般告警和严重告警，告警项包括：

- 资源类告警：包括 CPU 负载、内存负载、磁盘剩余告警
- I/O 类告警：包括磁盘 I/O、网络 I/O 告警
- 故障类告警：包括虚拟机宕机、物理机故障等告警

3.7 高可定制性（High Customization）

3.7.1 资源监控可视化定制

金航数码服务器虚拟化系统针对大屏可视化监控运维管理需求，提供可配置的资源监控可视化功能，可定制的资源监控包括：

- 资源池的整体监控：提供 CPU、内存、存储空间的总量、已分配量、已占用，资源池的物理机总数、虚拟机总数、虚拟机运行数等。
- 物理机的实时监控：提供 CPU 负载、内存负载、磁盘总的 I/O、各磁盘的占用情况、各磁盘的 I/O、网卡总 I/O、各网卡的 I/O、服务器进程信息等。
- 虚拟机的实时监控：提供 CPU 负载、内存负载、磁盘总的 I/O、各磁盘

的 I/O、网卡总 I/O、各网卡的 I/O 等。

3.7.2 审批流程定制

金航数码服务器虚拟化系统提供了资源管理流程接口，可根据用户实际的管理流程要求，与第三方 workflow 管理系统进行审批流程定制。

3.7.3 调度任务定制

金航数码服务器虚拟化系统基于自主分布式任务引擎，为用户提供了可定制的任务调度策略功能。用户可以根据时间（例如：每周对虚拟机进行一次备份）或事件（平台所支持的事件类型包括：物理机启动、物理机状态变化、物理机停止、物理机加入网格、资源池状态变化、进程终止、FT 热备开始、FT 热备结束、虚拟机启动、虚拟机加入网格、虚拟机停止、虚拟机所在宿主机发生变化、虚拟机完成备份、虚拟机完成恢复、虚拟机完成删除、虚拟机完成迁移、虚拟机状态变化）来触发定制新的个性化调度任务的执行。

3.7.4 系统界面皮肤定制

金航数码服务器虚拟化系统提供多套系统界面皮肤模板，并可根据用户的需求定制新的系统界面皮肤模板。

4 典型硬件配置

4.1 服务器

金航数码服务器虚拟化系统的服务器包括主管理服务器和虚拟化资源池服务器，推荐采用2U、4U机架式服务器，典型硬件配置如下表所示。

表 1 服务器典型硬件配置

服务器名称	硬件配置	备注
管理节点服务器	高度：2U 处理器：2×Intel Xeon E5-2623 V4 2.6 GHz, 10M 缓存, 4 核 内存：8GB RDIMM, 2133MT/s 硬盘：4×300GB 15K RPM SAS 2.5 英寸热插拔硬盘 网卡：6×1Gb Base-T 网口 Raid 卡：512MB 以上缓存，支持 RAID 0, 1, 5, 10 BMC 口：支持 IMPI 1.0 以上协议	可选用 AMD 同等配置处理器
资源节点服务器 (2U)	高度：2U 处理器：2×Intel Xeon E5-2650 v4 2.2GHz, 30M 缓存, 12 核 内存：192GB RDIMM, 2133MT/s (16GB×16) SSD 盘：2×400GB SSD 固态硬盘 SATA 6Gbps 2.5 英寸热插拔硬盘 硬盘：4×300GB 15K RPM SAS 2.5 英寸热插拔硬盘 网卡：4×1Gb Base-T 网口, 4×10Gb SFP+网口 HBA 卡：2×8Gb 光纤通道 HBA Raid 卡：1GB 以上缓存，支持 RAID 0, 1, 5, 10 BMC 口：支持 IMPI 1.0 以上协议	可选用 AMD 同等配置处理器
资源节点服务器 (4U)	高度：4U 处理器：4×Intel Xeon E7-4850 v4 2.1GHz, 40M 缓存, 16 核 内存：512GB RDIMM, 2133MT/s (16GB×32) SSD 盘：2×400GB SSD 固态硬盘 SATA 6Gbps 2.5 英寸热插拔硬盘 硬盘：4×600GB 15K RPM SAS 2.5 英寸热插拔硬盘 网卡：4×1Gb Base-T 网口, 4×10Gb SFP+网口	可选用 AMD 同等配置处理器

	HBA 卡：2×16Gb 光纤通道 HBA Raid 卡：1GB 以上缓存，支持 RAID 0, 1, 5, 10 BMC 口：支持 IMPI 1.0 以上协议	
--	--	--

4.2 SAN 存储

金航数码服务器虚拟化系统的SAN存储推荐使用中高端IP-SAN存储，存储磁盘容量根据实际需求设计，典型硬件配置如下表所示。

表 2 SAN 存储典型硬件配置

设备名称	硬件配置	备注
虚拟化系统 集中 SAN 存储	控制器：2 个并行架构控制器，最大可扩展至 16 个 缓 存：≥136GB 高速缓存，最大可扩展至 1024GB 端 口：≥8 个 10GbE SFP+端口 磁 盘：≥12 个 400G SSD 热插拔硬盘，12 个 TB NL-SAS 热插拔硬盘，最大可扩展至 7680 块硬盘 管 理：数据自动分层管理软件，实现冷热数据的自动分层管理；连续数据保护功能；性能监控管理软件，支持集中式 GUI 管理，可监控和分析多个存储组中的所有存储系统的性能，实现在单一管理界面下的性能监控、性能分析等	可选用同等配置 FC-SAN 存储

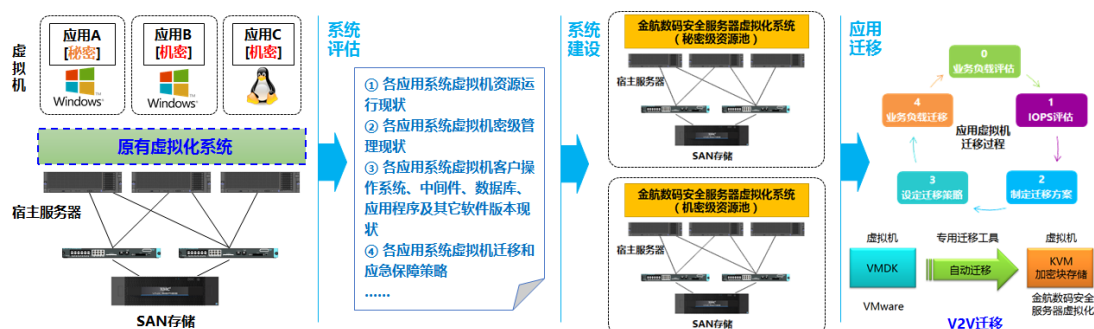
5 典型应用场景

金航数码服务器虚拟化系统重点面向涉密企业单位，提供对非标虚拟化系统升级替代和新建涉密虚拟化系统，同时也广泛适用于非密虚拟化系统建设。



5.1 涉密网非标虚拟化系统升级替代

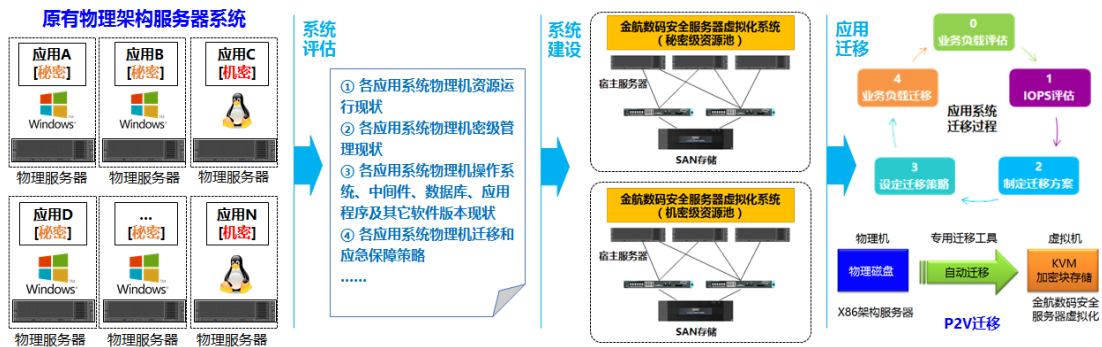
针对用户已建非标虚拟化系统，可采用金航数码服务器虚拟化系统进行升级替代，提高虚拟化系统的安全可控性，达到系统安全测评的要求。



5.2 新建涉密虚拟化系统

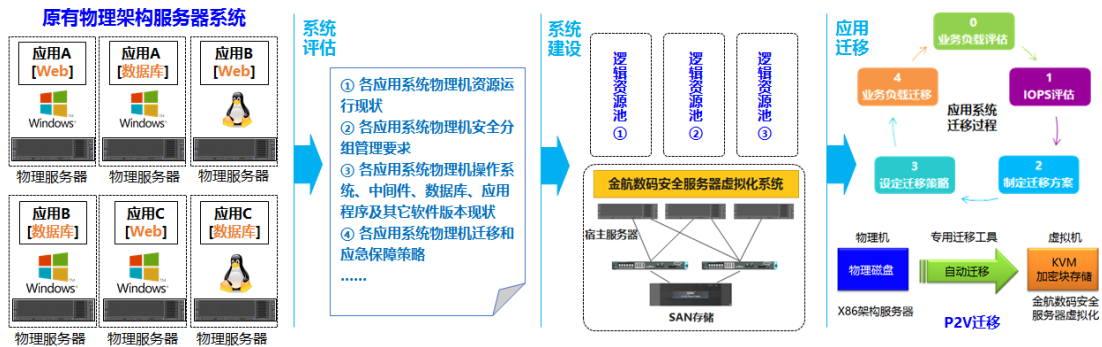
针对用户在涉密网部位新建服务器虚拟化系统，可采用金航数码服务器虚拟

化系统进行建设，提高资源利用效率，达到系统保密测评的安全要求。



5.3 新建非密虚拟化系统

针对用户在商网、内部网、等保网等非涉密部位新建服务器虚拟化系统，可采用金航数码服务器虚拟化系统进行建设，提高资源利用效率，达到系统安全测评的要求。



6 缩略语

6.1 缩略语

英文缩略语	英文全称	中文全称
VMM	Virtual Machine Monitor	虚拟机监视器
VM	Virtual Machine	虚拟机
OS	Operating System	操作系统
I/O	Input/Output	输入/输出
SAN	Storage Area Network	存储区域网络
VLAN	Virtual Local Area Network	虚拟局域网
HA	High Availability	高可用
BMC	Baseboard Management Controller	基板管理控制器
NUMA	Non-Uniform Memory Access	非一致存储访问结构