针对云计算 IaaS 隔离性的测试系统①

赵 炎, 程绍银, 蒋 凡

(中国科学技术大学 计算机科学与技术学院, 合肥 230027)

摘 要: 为解决 IaaS 云计算隔离性的测试问题, 通过分析和总结已有的隔离性的测试方法, 设计并实现了针对云计算 IaaS 基于消息中间件的分布式测试系统. 实现时, 采用基于消息中间件的分布式架构, 并将控制端与测试端独立, 降低了耦合性, 增强了可扩展性. 在 OpenStack 云计算环境中进行测试, 验证了系统设计的可行性. 该测试系统适用于云服务安全的其他能力测试.

关键词: 云计算; IaaS; 分布式测试; 隔离性; OpenStack

Testing System for Cloud Computing IaaS Isolation Properties

ZHAO Yan, CHENG Shao-Yin, JIANG Fan

(School of Computer Science and Technology, University of Science and Technology of China, Hefei 230027, China)

Abstract: By analyzing and summarizing the existing test methods for isolation of cloud IaaS, a large-scale distributed testing system based on message middleware is designed and implemented for validating the isolation of a cloud IaaS. The system is a message-based distributed architecture; the control nodes and the testing nodes are separated, which helps to reduce the coupling of the system and enhance scalability. By testing the isolation of cloud IaaS in operation, the feasibility of the testing system is verified on the OpenStack platform. The testing system is suitable for testing other security abilities of cloud services.

Key words: cloud; IaaS; distributed testing; isolation; OpenStack

IDC 发布的《中国公有云服务追踪研究》[1]表明, 国内公有云服务市场将在 2015 年至 2018 年持续高速增长, 其中 IaaS(Infrastructure as a Service)增长最为明显, 将由 2013 年的 44%上升至 2018 年的 61.3%. 2015 年 4 月 1 日开始实施的中国国家标准《信息安全技术 云计算服务安全能力要求》[2], 明确对系统虚拟化、网络虚拟化和存储虚拟化的安全能力提出了要求, 反复强调了隔离性的重要性, 隔离性是第三方测评机构针对云服务安全能力的测评重点.

隔离性是虚拟化的重要功能, Jeanna Neefe Matthews[3]测试了几种不同虚拟技术的隔离性, 采用的测试方法是在 1 台物理机器上创建 4 台虚拟机, 每个虚拟机上都运行 apache web server, 其中 1 台虚拟机做压力测试, 在进行压力测试时观察是否对未做压力测试的虚拟机有影响. Jeanna 等人的测试方法虽然能在一定程度上比较不同虚拟化技术的隔离性, 但是测试并没有体现出云计算的大规模性. Rally[4]是针对 Openstack 的性能测试[5]工具, 其有几种不同的测试方式: (1)在不具有 OpenStack 环境的硬件上自动部署 OpenStack 环境, 然后模拟真实用户的负载, 最后评估测试结果; (2)使用已经部署好的 OpenStack 云环境, 模拟真实用户的负载, 最后评估测试结果; (3)在特定的硬件上部署 OpenStack, 运行指定的基准测试集并保存性能数据. Rally 虽然能够自动化的对云计算进行性能测试, 但是其局限于 OpenStack, 同时也仅对性能做出了评估, 并没有体现出对隔离性的测试.

本文实现了一种针对云计算 IaaS 的大规模分布式测试系统, 重点针对处理器、网络、存储的隔离性进行测试, 弥补 Jeanna 测试方法的不足, 可扩展性强, 可针对每一台虚拟机模拟真实用户的负载, 同时能够

① 收稿时间:2016-04-22;收到修改稿时间:2016-05-26 [doi:10.15888/j.cnki.csa.005558]

模拟大规模用户在线的情况, 对云平台造成严重负载, 并检测在高负荷情况下云平台的隔离性, 同时, 该测试系统适用于云服务安全的其他能力测试.

1 系统架构设计与关键技术

1.1 系统结构设计

隔离性测试系统分为三类节点, 每一个节点为一台虚拟主机.

① 控制节点 C: 控制节点在不同阶段, 其作用有所不同. 在隔离性测试系统部署阶段, 控制节点主要作用为调用云平台 API 创建虚拟机, 并虚拟主机指定角色, 将角色分为中继节点和测试节点; 在测试阶段, 其主要作用是向中继节点发送测试命令或测试例, 并从中继节点收集测试结果并分析得出测试结论;

② 中继节点 R: 主要作用是将控制节点的测试命令或测试例转发给测试节点, 收集测试节点的监控信息, 将测试节点的测试结果及监控信息转发给控制节点, 每一个中继节点即为消息中间的 broker;

③ 测试节点 T: 主要作用是从中继节点接收测试命令或测试例, 实时监控虚拟主机状态, 并将测试结果和监控信息发送给中继节点. 测试节点从测试例库中调用测试例, 进行计算、网络、存储等隔离性测试, 同时也能进行其他安全能力测试.

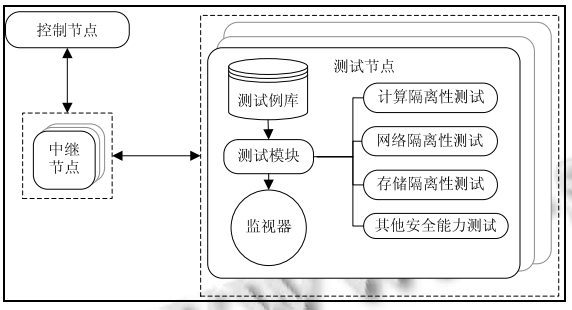

图 1 测试系统结构

图 1 所示为隔离性测试系统架构. 在这个隔离性测试系统中, 控制节点只需要一个, 中继节点和测试节点为多个. 三者之间的连接关系: 控制节点与多个中继节点通讯, 中继节点又与多个测试节点通讯, 中继节点还可与多个中继节点通讯. 将隔离性测试系统分为三类节点的主要原因是测试节点在高负荷的情况下, 可能出现无法优先为其它节点转发消息, 导致其他节点不能正常进行测试. 同时, 多个中继节点可以缓解大量测试节点给控制节点带来的压力. 一种典型的隔离性测试系统拓扑如图 2 所示. 采用图 2 所示的拓扑结构使测试系统主要优点是松散耦合, 系统扩展性强. 相比于传统的采用 RPC 的分布式架构[6], 基于消息中间件的分布式架构将控制节点与测试节点之间的通讯全部交给消息中间件, 降低了两者的耦合程度. 同时, 控制节点无需管理与测试节点的连接, 只需要增加中继节点即可增加更多的测试节点, 系统扩展性强.

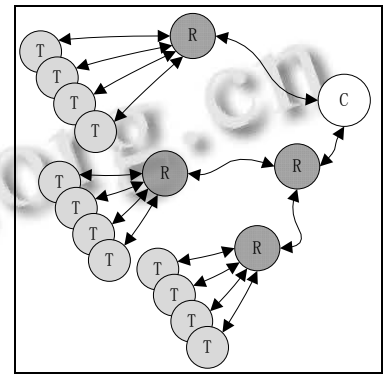

图 2 隔离性测试系统拓扑

1.2 关键技术

1.2.1 系统部署

图 2 中的每一个节点即为一台虚拟主机, 所以测试系统需要大量的虚拟主机. 在云平台创建所有虚拟主机前, 需要一台虚拟主机作为控制节点, 控制节点连接云平台创建大量虚拟主机. 在系统部署前修改云平台提供的系统镜像, 在镜像中预先安装测试程序, 以便在虚拟主机启动后自动连接控制节点. 控制节点创建虚拟主机的流程如图 3 所示, 具体描述如下:

步骤 1: 控制节点判断是否具有云平台管理员权限, 若是则跳至步骤 2, 否则跳至步骤 3;

步骤 2: 控制节点利用管理员权限创建一批用于测试的临时租户, 至步骤 4;

步骤 3: 控制节点等待测试人员提供用于测试的租户认证信息;

步骤 4: 控制节点利用用于测试的租户信息, 使用定制镜像创建大量虚拟主机;

步骤 5: 虚拟主机创建成功后即可进行隔离性测试.

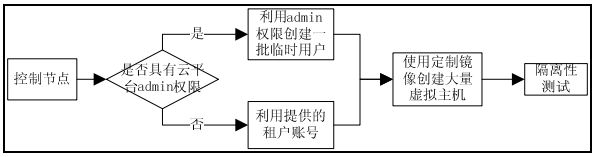

图 3 控制节点创建虚拟主机流程

1.2.2 隔离性测试方法

针对 IaaS 的计算、网络、存储资源的隔离性测试主要集中在 CPU、内存、文件 I/O 与网络的隔离性测试.

① CPU 隔离性测试是在用户指定的一段时间内循环进行高强度的浮点运算. 在进行浮点运算的过程中, 不断检查运算结果是否正确. 如果结果不正确, 会将结果信息保存在结果文件中, 并将错误信息发送给中继节点. 中继节点收集 CPU 使用情况和测试结果, 并分析测试结果, 将收集和分析的结果发送给控制节点. CPU 的隔离性测试, 我们采取 1 个测试节点不进行高强度的浮点运算, 其他测试节点进行高强度的浮点运算的方式, 通过分析测试结果, 判断浮点运算测试节点是否对没有进行浮点运算的节点产生影响. 如果出现浮点运算结果错误或没有进行浮点运算的节点 CPU 使用率波动较大, 则说明云平台的隔离性欠佳.

② 内存隔离性测试大致分为两种测试例: 负载和轻负载. 内存负载采取不断的申请、锁定和释放内存的方式, 如果申请指定大小的内存失败, 则会自动降低指定的内存大小, 然后再次申请, 以此往复, 导致测试节点内存基本被消耗尽. 轻负载则是指测试节点不进行高强度的内存消耗, 保持常态运行. 通过实时监控内存负载与轻负载的测试节点, 比较两者在同时间段内内存最大值及使用量的变化判断隔离性是否符合要求.

③ 文件 I/O 隔离性测试主要测试在大量测试节点同时读写文件时是否对文件 I/O 速度造成影响. 针对隔离性的测试主要测试两种情况下的文件 I/O 速度, 第一种是所有测试节点同一时段进行大量文件读写, 第二种是在每一簇中随机选取一个测试节点进行大量文件读写. 通过比较两种情况下文件 I/O 的速度来判断隔离性是否符合要求.

④ 网络隔离性测试不同于 CPU、内存和文件 I/O 的隔离性测试, 后三种测试时测试节点运行测试例不需要其他测试节点的协助, 而网络隔离性测试需要其他测试节点协助, 模拟网络通信, 其中一方作为服务端, 另一方作为客户端. 《信息安全技术 云计算服务安全能力要求》中明确提出了对虚拟主机的带宽进行管理, 因此测试虚拟主机的带宽需要两个测试节点间协助测试. 典型的情景是假设两个测试节点 A、B 分别位于不同簇内, 则 A、B 互为服务端和客户端. 一种简单的网络隔离性测试拓扑如图 4 所示, 图中虚线表示网络隔离性测试. 通过分析测试结果, 判断是否对虚拟主机网络带宽进行了管理, 是否影响其他虚拟主机网络. 测试节点 A、B 的选择则采取编号相差最小且在不同簇的测试节点, 如果某簇中有未匹配成功的测试节点, 则该测试节点与中继节点或簇内其他尚未匹配的测试节点间进行网络隔离性测试, 具体的描述见算法 1, 其中步骤 4 会判断是否是不同簇且未配对节点, 满足这两个条件则会配对, 步骤 10 则是在节点首次匹配失败时尝试与中继节点匹配, 如果中继节点已经与其他测试节点匹配, 则测试节点与本簇中其他测试节点匹配, 算法 2 中步骤 11-17 则是尝试与簇内其它节点匹配, 以上条件都不满足的节点则不进行网络隔离性测试. 网络隔离性测试要比较所有虚拟主机并发进行网络带宽测试得到的带宽值是否与虚拟主机单次测试的网络带宽相同, 如果两者相差较大则说明云平台没有对虚拟主机的网络带宽进行妥善的管理, 不满足网络隔离性的要求.

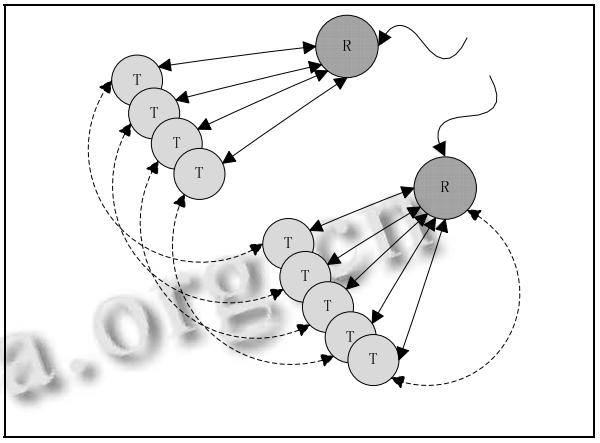

图 4　网络隔离性测试拓扑

算法 1.　网络隔离性测试节点匹配算法

输入: *testIndexs*-所有测试节点编号,

rIndex-中继节点编号.

Function NetMatch(*testIndexs*,){

步骤 1　for each *i* in *testIndexs*{

步骤 2　if *i* is matched then continue;

步骤 3　else for each *j*>*i* in *testIndexs*{

步骤 4　if *j* in another cluster and *j* is not matched {

步骤 5　*i* and *j* matches;

步骤 6 break;
步骤 7 }
步骤 8 }
步骤 9 if *i* is not matched {
步骤 10 if *rIndex* is not matched then *i* and *rIndex* matches;
步骤 11 else for each *j* in *testIndexs*{
步骤 12 if *j* and *i* is not in the same cluster then continue;
步骤 13 else if *j* is not matched {
步骤 14 *i* and *j* matches;
步骤 15 break;
步骤 16 }
步骤 17 }
步骤 18 }
步骤 19 }
}

1.2.3 消息传递

控制端与测试端间的通讯协议采用 MQTT[7]协议. MQTT 协议是基于发布/订阅的轻量级的消息协议. 既然是消息便涉及到生产者和消费者, 生产者发布消息, 消费者通过订阅获取消息. 考虑到测试系统中需要控制多个虚拟主机, 即一个生产者发送消息给多个消费者, 控制节点于测试节点通过消息中间件进行交互. 控制端与测试端间的通讯主要分为两种方式:

① 控制端发布消息: 控制端创建消息主题(topic), 测试端订阅该 topic 接受消息;

② 测试端发送消息给控制端: 控制端创建一个消息队列, 测试端向消息队列中发送发送消息.

2 系统评估与实验

2.1 系统性能评估

对于系统的性能, 由于主控节点 MC 和控制节点 C 不进行隔离性测试, 只是担当协调者的角色, 所以评估测试系统的性能主要针对监控终端. 监控终端在虚拟主机中运行前后资源占比如图 5 所示, 可知监控终端对主机影响很小.

2.2 实验

为了对云计算 IaaS 隔离性进行测试, 本文利用两台 PC 搭建了一个小型的 OpenStack 云计算系统, 两台 PC 分别作为 controller 节点和 compute 节点, 其硬件信息如下:

① controller: 4 核 cpu, 8GB 内存, 500GB 硬盘, 双千兆网卡;

② compute: 4 核 cpu, 16GB 内存, 1TB 硬盘, 双千兆网卡.

测试工具创建 10 台虚拟机实例, 其中一台作为测试系统的控制节点 C, 一台作为中继节点 R, 剩余 8 台作为测试节点.

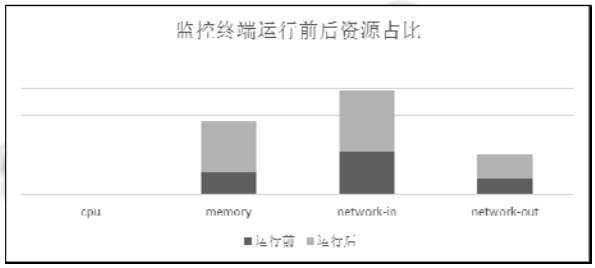

图 5 监控终端运行前后资源占比

2.2.1 CPU 隔离性测试

实验时从 8 个节点中随机选取了一台虚拟机作为不进行高强浮点运算度负载, 其他虚拟机负载作高强度的浮点运算. 对负载节点的 CPU 使用率取均值得到如图 6 所示结果. 图中负载即为进行高强度浮点运算, 未负载则没有进行浮点运算. 从图中可以看出, 负载节点 CPU 使用率从 08:45 逐步增加, 10:15 左右使用率达到 100%, 而后逐步降低. 在此期间未负载节点 CPU 使用率基本保持不变. 同时通过分析, 浮点运算结果均正确. 从而可以得知, OpenStack 实验环境满足基本的 CPU 隔离性要求.

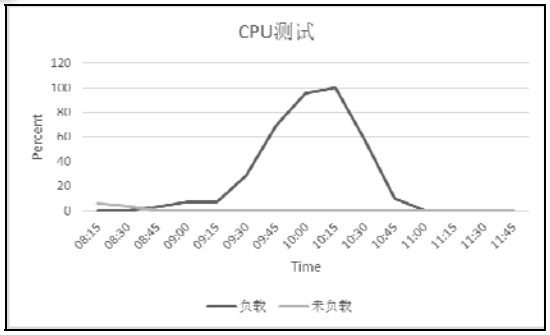

图 6 CPU 负载与未负载使用百分率

2.2.2 内存隔离性测试

实验随机选取一台虚拟机, 使其不进行内存负载, 而其余 8 台虚拟机进行内存负载, 内存负载即是尽可

能占用系统内存. 实时监控所有测试节点内存使用率, 并对负载测试节点内存使用量取平均值得到如图 7 所示的内存消耗图. 由图可知, 在 31:30 处, 负载虚拟机内存消耗迅速上升后, 在 43:30 左右后恢复正常, 期间未负载虚拟机内存消耗基本不变. 同时, 在进行内存隔离性测试的时间段内, 负载与未负载虚拟机内存大小基本保持不变, 如图 8 所示. 由两图分析可知, 实验环境基本满足内存隔离性要求.

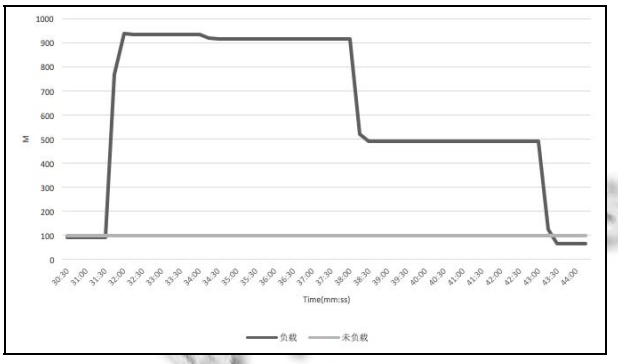

图 7　内存测试-用户内存消耗

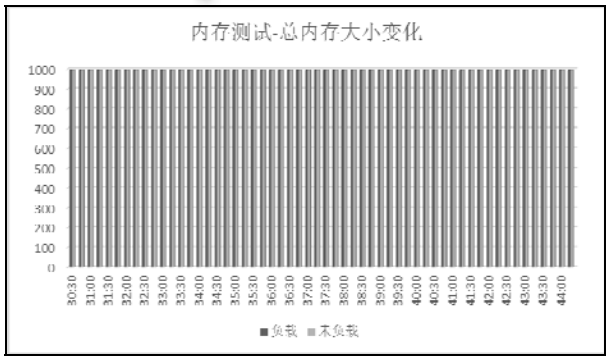

图 8　内存测试-总内存大小变化

2.2.3 网络隔离性测试

网络隔离性测试分两步进行. 第一步是随机选择两台虚拟机实例, 其中一台作为服务器端, 另外一台作为客户端, 测试两者之间的最大网络带宽, 称其为单点测试; 第二步是多个点同时开始第一步中的测试, 根据网络测试的节点匹配方法, 节点间相互匹配, 作为服务器端和客户端进行最大网络带宽测试, 并对各节点对的测试结果取均值. 两步测试得到的结果如图 9 所示.

从图 9 中数据分析可知, 单点测试时其网络带宽较大, 而多点测试时其网络带宽值几乎只有单点测试时的 1/4, 足以见得云平台没有对每台虚拟主机的带宽进行妥善管理, 不符合《信息安全技术 云计算服务安全能力要求》标准.

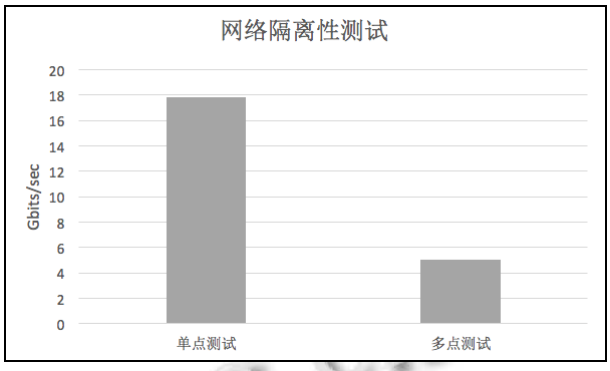

图 9　网络隔离性测试

2.2.4 磁盘 I/O 隔离性测试

磁盘 I/O 隔离性测试采取两步进行: 一台虚拟机单独进行磁盘 I/O 隔离性测试和多台虚拟机同时进行磁盘 I/O 隔离性测试. 磁盘 I/O 隔离性测试分别测试文件的读写速度、重复读写速度, 以及随机读写速度, 将两步测试分别称为 async 和 sync. async 是随机选取一台虚拟主机进行磁盘 I/O 速度测试, sync 是 8 台虚拟主机同时进行磁盘 I/O 速度测试, 得到如图 10 所示的测试结果.

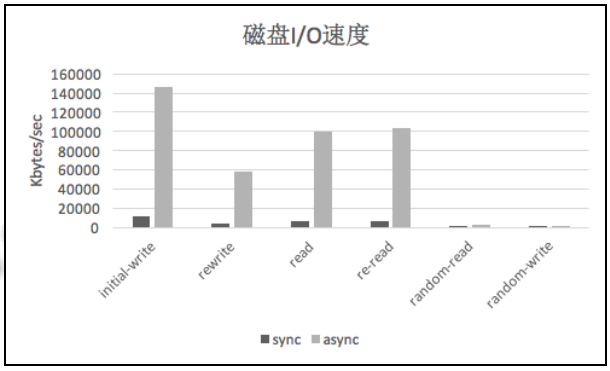

图 10　磁盘 I/O 隔离性测试结果

由图 10 分析可知, sync 测试得到的磁盘 I/O 速度明显小于 async. 由此可知云计算虚拟机的磁盘 I/O 性能受其他虚拟机影响明显, 或者说是云计算平台在负载较大时租户的虚拟机磁盘 I/O 性能会受到明显影响. 因此可知其不能满足某些对存储性能要求较高的租户, 磁盘 I/O 隔离性欠佳.

由于硬件特性, 相比于 CPU 和内存, 磁盘 I/O 速度一直阻碍着计算机的性能提升, 同时也考虑到成本问题, 对磁盘的投入也比较少. 但是, 对于云计算服

务提供商而言, 还是可以通过一些软硬件的方法来改善虚拟机磁盘 I/O 的性能, 以提高隔离性和对用户的公平. 例如, 可以参考 Dingding Li 等人[8]通过降低 Xen 硬盘协议栈上的额外开销来提升虚拟机磁盘 I/O 性能, 也可参考 Ajay Gulati 等人[9]提到的 VMware DRS 系统, 通过资源的合理调用和管理来提高系统的资源利用率和系统性能

3 结语

本文实现了一种针对 IaaS 云计算服务基于消息中间件的分布式测试系统, 重点针对处理器、网络、存储的隔离性进行测试, 可扩展性强, 可针对每一台虚拟机模拟真实用户的负载, 同时能够模拟大规模用户在线的情况, 对云平台造成严重负载, 并检测在高负荷情况下云平台的隔离性. 下一步工作是完善在不同云平台的自动部署, 并参照《信息安全技术 云计算服务安全能力要求》设计测试例.

参考文献

1 新浪科技.IDC:阿里云成国内最大 IaaS 云计算厂商, 2015-1-23.

2 左晓栋,陈兴蜀,张建军,等.GB/T 32268-2014,信息安全技术 云计算服务安全能力要求.北京:中国标准化委员会,2014.

3 Matthews JN, Hu W, Hapuarachchi M, et al. Quantifying the performance isolation properties of virtualization systems. Proc. of the 2007 workshop on Experimental computer science. ACM. 2007. 6.

4 OpenStack Foundation. http://rally.readthedocs.org.Cloud Computing.

5 Jayasinghe D, Swint G, Malkowski S, et al. Expertus: A generator approach to automate performance testing in IaaS clouds. 2012 IEEE 5th International Conference on Cloud Computing (CLOUD). IEEE. 2012. 115–122.

6 蒋雄伟,马范援.中间件与分布式计算.计算机应用,2002, 22(4):6–8.

7 贾军营,王月鹏,王少华.基于 MQTT 协议 IM 的研究和实现. 计算系统应用,2015,24(7):9–14.

8 Li D, Jin H, Liao X, et al. Improving disk I/O performance in a virtualized system. Journal of Computer and System Sciences, 2013, 79(2): 187–200.

9 Gulati A, Holler A, Ji M, et al. Vmware distributed resource management: Design, implementation, and lessons learned. VMware Technical Journal, 2012, 1(1): 45–64.