

基于 Ganglia 和 Nagios 的云计算平台智能监控系统

张小银 陈国胜

(安徽工业大学现代教育技术与网络管理中心,安徽 马鞍山 243032)

摘 要: 随着现代数据中心云计算规模日益增长,云计算平台的智能运维管理面临较大挑战,尤其在实时监控领域方面。首先对云计算监控技术进行了深入分析和研究,然后在开源云计算平台 Hadoop 环境下,将 Ganglia 和 Nagios 两种开源监控软件进行整合,并利用移动飞信来实现对云计算平台的实时监控。实验结果表明,该系统可对云计算平台内主机和服务以及运行环境的各项性能指标进行全方位监控,实现对故障的实时预警和报警,使得管理人员能准确定位、实时处理云平台异常情况,从而提高了云平台的服务质量,有较好的应用价值。

关键词: 云计算; Hadoop; 性能指标; 监控系统

中图分类号: TP393 **文献标志码:** A **文章编号:** 1672-1098(2016)04-0069-06

Intelligent Monitoring System on Cloud Computing Platform Based on Ganglia and Nagios

ZHANG Xiao-yin, CHEN Guo-sheng

(Modern Education Technology and Network Management Center, Anhui University of Technology, Maanshan Anhui 243032, China)

Abstract: With the growing scale of cloud computing in modern data centers, the intelligent operation and maintenance management faces a great challenge, especially in real-time monitoring. After a thorough analysis and research of cloud computing monitoring technologies, this paper integrates two open-source monitoring software Ganglia and Nagios in a Hadoop open-source cloud computing platform, and uses a mobile message software FeiXin to achieve real-time monitoring of the cloud computing platform. Experimental results show that the proposed system realizes an all-round monitoring of performance indicators for hosts and service of operating environment in cloud computing platform and a real-time warning of faults, which help management personnel accurately locate and real-time process abnormal situations. Therefore the system improves the quality of service of cloud computing platform and has a good practical value.

Key words: cloud computing; hadoop; performance indicators; monitoring system

随着云计算技术的不断成熟发展,云计算平台的规模以及资源也不断增加。现代数据中心的运维管理面临着重大挑战,传统的管理方法和管理模式已经无法满足要求。

为提高云计算平台的可靠性,保证服务质量,有必要在云计算平台中引入监控机制^[1-3],以便能准确定位性能异常或故障的节点,及时做出恢复和

调整;掌握整个系统的运行状况,分析系统瓶颈,为整个系统负载均衡提供数据支持,在系统出现异常时能起到预警的作用。

Ganglia 是加州大学伯克利分校发起的一个开源监控项目,主要用来监控大规模分布式系统的性能^[4]。Nagios 也是功能强大的开源监控系统,能监控所指定(本地和远程)的主机以及服务,可利用

收稿日期: 2014-12-02

作者简介: 张小银(1962-),男,安徽当涂人,实验师,本科,研究方向: 计算机网络。

故障状态实现故障报警^[5]。Ganglia 侧重于数据采集,没有内置网络服务的监控和故障状态级别,Nagios 更侧重于告警功能,配置文件较多,配置步骤繁琐。因此,本文采用两款软件结合,协同工作。

经过几年的迅速发展,Hadoop 已经成为开源云计算平台的佼佼者,目前具有广泛的用户群体。因此,研究利用 Ganglia 和 Nagios 整合来监控 Hadoop 系统具有广阔的应用前景。

1 相关技术原理

1.1 Hadoop 技术

云计算是并行计算、分布式计算和网格计算的发展。具有超大规模、虚拟化、按需服务、高可靠性和高扩展性等特点。

Hadoop 是一个开源的分布式计算平台,由 Apache 软件基金会支持发布。整个 Hadoop 的体系结构主要是通过 HDFS 来实现对分布式存储的底层支持,通过 MapReduce 来实现对分布式并行任务处理的程序支持的。

Hadoop 分布式文件系统(HDFS)采用了主/从(Master/Slave)结构模型,一个 HDFS 集群由一个名称节点(Namenode)和若干个数据节点(Datanode)组成。Namenode 作为主控服务器,负责管理文件系统的元数据,Datanode 存储实际的数据。Namenode 执行文件系统的命名空间操作,如打开、关闭、重命名文件或目录,Datanode 负责处理客户的读写请求,执行数据块的创建、删除和复制工作。Namenode 使用事务日志来记录 HDFS 元数据的变化,使用映射文件存储文件系统的命名空间。采用冗余备份、副本存放、心跳检测、安全模式等策略使 HDFS 可靠性得到保证^{[2][6]}。

MapReduce 是一种并行编程框架,它将分布式运算任务分解成多份细粒度的子任务,发到由上千台机器组成的集群上,这些子任务在各处理节点之上并行处理,最终通过某些特定的规则进行合并生成最终的结果。MapReduce 任务是由一个 JobTracker 和多个 TaskTracker 节点控制完成,JobTracker 单独运行在主节点上,负责调度管理 TaskTracker,调度一个作业分解的所有子任务。主节点监控子任务的执行情况,从节点仅负责完成由主节点指派的子任务。MapReduce 将分布式运算抽象

成 Map 和 Reduce 两个步骤,最终完成 Hadoop 的并行处理任务^[6]。

1.2 云计算监控技术

1) Ganglia 监控技术 Ganglia 在结构上由 gmond、gmetad 和 gweb 三个守护进程组成,三者相互协调,具体如图 1 所示。

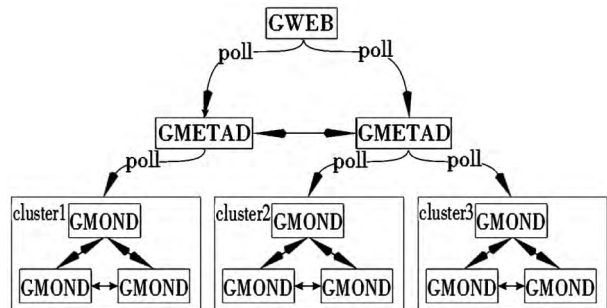


图 1 Ganglia 的结构

gmond 是数据采集器,运行在所有被监控主机上的一个守护进程。用于收集被监控主机上的基本指标,或者收集用户自定义的指标,在同一个组播或单播通道上的传递。gmond 所产生的系统负载非常小,这使得在各被监控主机上运行 gmond 时,不会影响到各主机的性能。Ganglia 的收集数据可以分为单播和组播两种工作模式。

gmetad 是数据混合收集器,运行在监控主机上的一个守护进程,gmetad 通过轮询收集各主机上 gmond 的数据,并聚合集群的各类信息,然后保存在本地 RRD 存储引擎中。

gweb 是 Web 可视化工具,采用 PHP 脚本语言实现,运行在 Ganglia 的监控主机上。可以通过浏览器从 RRDTool 数据库中抓取信息,将数据可视化,动态的生成各类图表。

Ganglia 集群是主机和度量数据的逻辑分组,一般每个集群运行一个 gmetad,可以构成层次结构,正因为有这种层次结构模式,才使得 Ganglia 可以实现良好的扩展。

2) Nagios 监控技术 Nagios 监控系统分为核心和插件两大部分。Nagios 的核心部分只提供了很少的监控功能,其它大部分监控功能需要安装相应的 Nagios 插件完成。

Nagios 可实现如下功能: 监视本地或者远程主机资源; 监视网络服务资源; 允许自定义插件来监控特定的服务; 出现异常时, 可以通过邮件、短信等方式通知管理人员; 可以事先定义事件处理程序, 当主机或者服务出现故障时自动调用指定的处理程序; 可以通过 Web 界面来监控各个主机或服务的运行状态。

Nagios 必须运行在 Linux/Unix 服务器上, 这台服务器称为监控中心, 每一台需要监视的主机或者服务都运行一个与监控中心服务器进行通信的 Nagios 软件后台程序。监控中心服务器根据读取配置文件中的指令与远程的守护程序进行通信, 并且指示远程的守护程序进行必要的检查。

远程被监控的机器可以是任何能够与其进行通信的主机。根据远程主机返回的应答, Nagios 将根据配置以合适的行动进行回应, 通过一种或者多种方式报警。

NRPE 是 Nagios 的一个功能扩展, 它可在远程 Linux/Unix 主机上执行插件程序, 通过在远程主机上安装 NRPE 构件以及 Nagios 插件程序, 向监控中心提供该主机的一些本地的情况。

2 云计算平台智能监控体系

本文在开源云计算平台 Hadoop 环境下, 将 Ganglia 和 Nagios 两种开源监控软件进行整合, 配合移动飞信来实现对云计算平台的实时监控。形成了如图 2 所示的一整套云计算平台的智能监控体系。

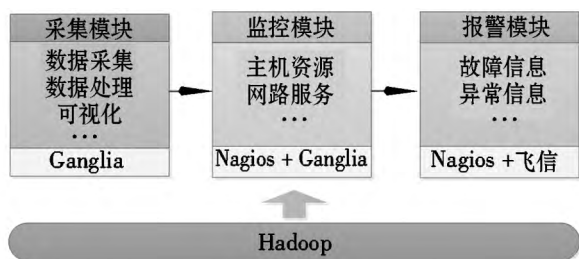


图2 云计算平台智能监控体系

2.1 采集模块

该模块通过整合 Ganglia 和 Hadoop 平台来采集监控主机的基本指标或者用户自定义的指标。然后进行数据处理, 包括信息聚合、分类、可视化, 生成报表等。具体过程如下:

Ganglia 的监控进程 (gmond) 发送的指标格式是有明确定义的。用户可配置 Hadoop 指标子系

统, 按照 Ganglia 的要求, 直接向 Ganglia 发送指标数据。用户可以根据需要, 用 Ganglia 对 Hadoop 的一个或全部上下文进行监控, 需要监控的 Hadoop 上下文包括 Java 虚拟机 (JVM) 上下文, 远程调用 (RPC) 上下文, 分布式文件系统 (DFS) 上下文, Mapreduce (mapred) 上下文等配置项。每个上下文对应一个 Hadoop 指标子系统, 每个子系统包括多项 Hadoop 指标。

Hadoop 的配置见文件 `hadoop - metrics2. properties`, 配置项的前缀是上下文名称, 每个上下文配置项都有如下三个属性。

Class, 指标的输出格式。

Period, 指标更新的时间间隔。

Servers, 发布指标的单播或多播地址列表。

例如, 配置 Mapreduce (mapred) 上下文格式如下

```
{
    mapred.class = org.apache.hadoop.metrics.
ganglia.GangliaContext31
    mapred.period = 10
    mapred.servers = host1
}
```

在本系统中, 由于只有少数主机需要处理, 为简化启用和配置, Ganglia 只使用单个集群。

在组播模式下, 当节点规模过大, 组播会对系统性能产生一定的影响。在本系统中, 尽管只有三个节点, 仍采用单播传输模式。

2.2 监控模块

该模块通过整合 Nagios 与 Ganglia 来完成相关资源的监控, 包括主机资源、网络资源等。

在后台, Nagios 实际上只是单一进行调度和通告的引擎。Nagios 本身并不能监控任何内容, 只能调度插件程序的执行, 并处理输出结果。

本系统采用 Nagios 来监控 Ganglia 指标。Ganglia 项目在 gweb 模块中包含了一系列官方 Nagios 插件。这些插件使得 Nagios 用户可以创建一些服务, 将存储在 Ganglia 中的指标和 Nagios 中定义的告警阈值进行比较。

在实际应用中, 使用 Ganglia 插件来监控系统, 如: Ganglia 内部使用心跳计数器来确定某台主机是否在运转; 将给定主机的单个指标与预定义的 Nagios 的门限值进行比较来检查特定主机的某种指标; 检查特定主机上多种指标; 检查使用正则表达式所定义范围内主机的多种指标; 验证一套主机

上的一个和多个指标值是否相同。

某 Nagios 插件运行原理示意如图 3 所示。

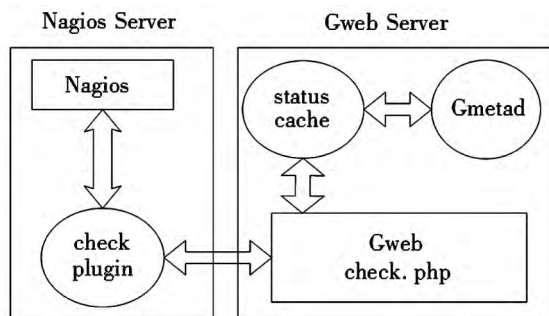


图 3 Nagios 插件运行原理示意图

在图 3 中,这些插件和一系列专门为此创建的 gweb PHP 脚本进行交互。每个 PHP 脚本从插件接收参数,解析从 gmetad 获取的有关状态缓存,提取被监控实体当前的指标值,并返回。Nagios 插件和 PHP 脚本成对出现。

实际应用中,在 hosts.cfg 中定义 hostgroup 格式如下:

```
define hostgroup{
    hostgroup_name    ganglia - metrics
    alias              ganglia - metrics
    members            host1 ,host6 ,host7
}
```

同时,自定义一个 ganglia - metrics.cfg 文件,文件中定义了很多要执行的操作命令。如定义 check_ganglia_heartbeat 命令如下

```
define command {
    command_name      check_ganglia_heartbeat
    command_line      $ USER1 $ /check_heart-
    beat.sh
    host = $ HOSTNAME $ threshold = $ ARG1 $
}
```

其中,check_heartbeat.sh 是 Nagios 插件。

此外,可以使用 Nagios 监控 Ganglia 主机的运行情况,如使用 check_nrpe 守护进程,监控 Ganglia 的所有故障,如监控汇聚主机上的 gmetad 和 rrd-cached 以及所有主机上的 gmond,监控 TCP 端口(如 gmetad 和 gmond 的监听端口)的连通性等。

2.3 报警模块

为了减轻工作负担,使管理人员能实时获取云计算平台运行异常或故障信息,本文使用整合了

Nagios 与移动飞信的报警模块将报警信息直接送到管理人员手中。

Nagios 利用插件使用 Ganglia 采集的信息,在运行指标超过阈值的情况下通知管理人员,通知方式采用移动飞信。

Nagios 下飞信的配置主要包括:

飞信命令定义

配置 commands.cfg 文件,定义一个服务故障时发送报警短信的指令,如下

```
define command{
    command_name    notify - service - by - sms
    command_line    /usr/local/fetion/fetion-mobile
    = 139xxxxxxx-pwd = xxxx-to = $ CONTACT-
    PAGER $
    -msg - utf8 = " $ HOSTADDRESS $ /HOS-
    TALIAS $ /$ SERVICEDESC $ is $ SERVICES-
    TATE $ "
}
```

定义一个主机故障时发送报警短信的指令,如下

```
define command{
    command_name    notify - host - by - sms
    command_line    /usr/local/fetion/fetion-mobile
    = 139xxxxxxx
    -pwd = xxxx-to = $ CONTACTPAGER $
    -msg - utf8 = " Host $ HOSTSTATE $ alert
    for $ HOSTNAME $ ! on $ DATETIME $ "
}
```

其中“139xxxxxxx”用来指定发送飞信的手机号码,“xxxx”用来指定此手机号飞信的密码。

Nagios 插件可以返回 OK、Warning、Critical 和 Unknown 四种状态。

联系人模板定义

配置 templates.cfg 文件,在 generic - contact 定义中,service_notification_commands 增加 notify - service - by - sms,host_notification_commands 增加 notify - host - by - sms,也就是在 commands.cfg 文件中新定义的两个指令。

```
define contact{
    name    generic - contact;
    ...
    service_notification_commands    notify - service
    - by - sms;
```

```

host_notification_commands notify - host - by -
sms;
...
}

```

添加联系人

配置 contacts. cfg 文件,修改联系人为 nagiosadmin 的定义,增加“pager 139xxxxxxx”一行,其中“pager”用来指定接收报警短信的手机号码。

```

define contact{
    contact_name nagiosadmin;
    use generic - contact;
    ...
    pager 139xxxxxxx
}

```

3 云计算平台智能监控系统的实现

3.1 系统总体架构及实现环境

本系统使用三台 VMware 虚拟机,根据需要组建 Hadoop 集群,Hadoop 集群各主机参数列表如表 1 所示。

表 1 Hadoop 集群各主机参数列表

主机名	IP 地址	CPU/个	内存/G	硬盘/G
host1	10.100.98.128	10	10	200
host6	10.100.98.124	8	8	100
host7	10.100.98.126	8	8	100

在三台主机中安装 CentOS 6.5 系统,开发环境安装 JDK1.7,安装 Hadoop 1.1.2,配置 host1 作为 NameNode、SecondaryNameNode,三台主机均配置为 DataNode,在 host1 上运行 JobTracker,在三台主机上均运行 TaskTracker。

在 host1 上安装 Ganglia - gmetad 3.1.7, Ganglia - gweb 3.4.2, Web 服务器 Apache2.2, 脚本语言 php - 5.3.3 在三台主机均安装 Ganglia - gmond 3.1.7。

在 host6 主机上安装 Nagios Core 3.4.4, nagios - plugins - 1.5, nrpe - 2.14, Web 服务器 Apache2.2, 脚本语言 php - 5.3.3, 安装移动飞信 fecton。系统总体架构如图 4 所示。

3.2 监控系统实现效果

Ganglia 和 Nagios 均具有丰富的 Web 展示功

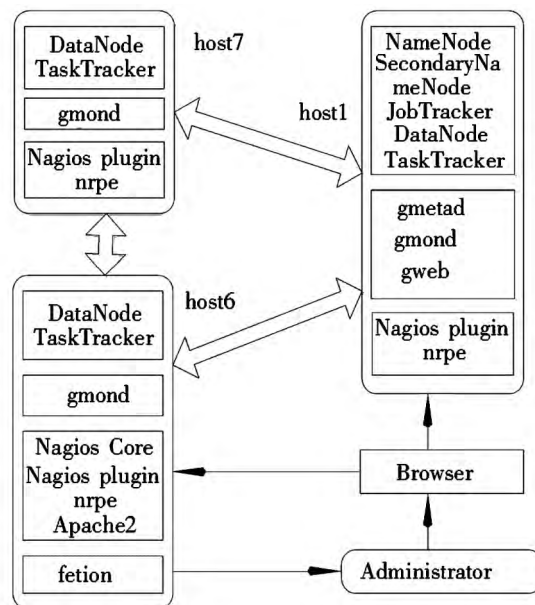


图 4 系统总体架构示意图

能。实现环境中,Ganglia 能监控 hadoop 集群及各主机性能指标众多,大约有几百个,下面只展示其中的几个,图 5 ~ 图 7 是 Ganglia 监控效果图(截图): host6 节点一月内平均负荷如图 5 所示, host1 节点一天内 jobtracker. heartbeats 指标如图 6 所示, host1 节点一天内 namenode. blockReport_num_ops 指标如图 7 所示。

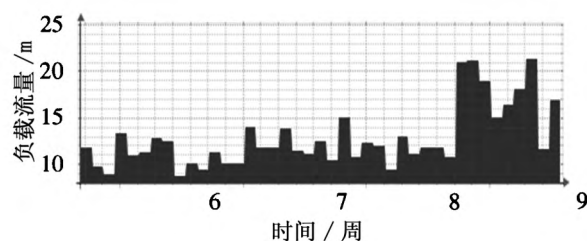


图 5 host6 节点一月内平均负荷

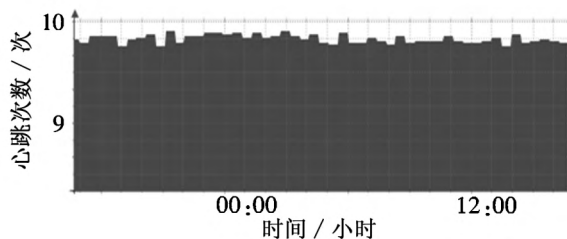


图 6 host1 节点一天内 jobtracker. heartbeats 指标

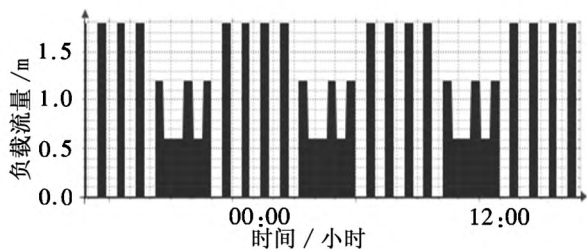


图 7 host1 节点一天内 namenode.blockReport_num_ops 指标 Nagios 监控效果只选取一张图 ,Nagios 所有主

Host	Service	Status	Last Check	Duration	Attempt	Status Information
host1	GMOND	OK	02-18-2014 11:22:04	27d 17h 6m 32s	1/3	PROCS OK: 1 process with command name 'gmond'
	check_ganglia_heartbeat	OK	02-18-2014 11:25:25	27d 17h 4m 38s	1/3	OK Last beacon received 0 days, 0:00:07
	check_ganglia_metric_disk_free	OK	02-18-2014 11:25:53	27d 17h 2m 43s	1/3	OK disk_free = 191.217 GB
	check_ganglia_metric_load_one	OK	02-18-2014 11:26:50	27d 17h 1m 46s	1/3	OK load_one = 0.00
	check_value_same_everywhere	OK	02-18-2014 11:19:47	27d 16h 59m 51s	1/3	OK, svn_revision same => 0 nodes, num_config_files same => 0 nodes
host6	GMOND	OK	02-18-2014 11:22:19	27d 17h 6m 21s	1/3	PROCS OK: 1 process with command name 'gmond'
	check_ganglia_heartbeat	OK	02-18-2014 11:25:10	27d 16h 50m 26s	1/3	OK Last beacon received 0 days, 0:00:00
	check_ganglia_metric_disk_free	OK	02-18-2014 11:27:04	27d 17h 2m 32s	1/3	OK disk_free = 87.967 GB
	check_ganglia_metric_load_one	OK	02-18-2014 11:27:02	27d 17h 1m 34s	1/3	OK load_one = 0.00
	check_value_same_everywhere	OK	02-18-2014 11:19:12	27d 16h 59m 40s	1/3	OK, svn_revision same => 0 nodes, num_config_files same => 0 nodes
host7	GMOND	OK	02-18-2014 11:22:27	27d 17h 6m 9s	1/3	PROCS OK: 1 process with command name 'gmond'
	check_ganglia_heartbeat	OK	02-18-2014 11:28:21	27d 16h 50m 15s	1/3	OK Last beacon received 0 days, 0:00:00
	check_ganglia_metric_disk_free	OK	02-18-2014 11:28:16	27d 17h 2m 20s	1/3	OK disk_free = 41.826 GB
	check_ganglia_metric_load_one	OK	02-18-2014 11:27:13	27d 17h 1m 23s	1/3	OK load_one = 0.00
	check_value_same_everywhere	OK	02-18-2014 11:19:12	27d 16h 59m 28s	1/3	OK, svn_revision same => 0 nodes, num_config_files same => 0 nodes
localhost	Current Load	OK	02-18-2014 11:27:41	29d 20h 14m 51s	1/4	OK - load average: 0.00, 0.00, 0.00
	Current Users	OK	02-18-2014 11:23:33	29d 20h 14m 13s	1/4	USERS OK - 0 users currently logged in
	HTTP	OK	02-18-2014 11:25:56	29d 18h 0m 36s	1/4	HTTP OK: HTTP/1.1 200 OK - 265 bytes in 0.001 second response time
	PING	OK	02-18-2014 11:28:19	29d 20h 12m 58s	1/4	PING OK - Packet loss = 0%, RTT = 0.06 ms
	Root Partition	OK	02-18-2014 11:28:08	29d 20h 12m 21s	1/4	DISK OK - free space: / 44003 MB (91% inode=96%):
	SSH	OK	02-18-2014 11:24:12	29d 20h 11m 43s	1/4	SSH OK - OpenSSH_5.3 (protocol 2.0)
	Swap Usage	OK	02-18-2014 11:26:04	29d 20h 11m 6s	1/4	SWAP OK - 100% free (8015 MB out of 8015 MB)
	Total Processes	OK	02-18-2014 11:28:23	29d 20h 10m 28s	1/4	PROCS OK: 169 processes with STATE = RSZOT

图 8 Nagios 所有主机服务状态详述(局部)

本系统一般只需设定 Nagios 插件返回 Critical 和 Unknown 二种状态发出报警,且一小时间隔循环发送即可。在系统主机和服务出现异常情况时,管理员接收飞信的手机会收到报警短信,格式如下:发信人为“12520139xxxxxxx”,短信内容为“XXX: 10.100.98.126' host7/GMOND is CRITICAL”,最后是接收短信日期时间,“XXX”为接收飞信手机机主的姓名。此时,管理员可在本地或远程实时维护 host7 主机上的 GMOND 服务。

在实际应用中,可以根据具体需求调整要监控的服务项。利用 Hadoop、Ganglia 和 Nagios 良好的可扩展性,动态增加节点,以便加入更多的 Hadoop 应用。通过改变 Hadoop 集群的负载,或通过调整 VMware 虚拟主机的部分参数,使系统负载达到均衡。由于系统中使用的端口众多,因此应特别注意 iptables 防火墙的设置。

4 结 语

在开源云计算平台 Hadoop 环境下,利用 Hadoop 系统提供的监控接口,将 Ganglia 和 Nagios 整合,通过 Web 可视化工具,强大的图表展示功能,直观地了解每个节点以及整个 Hadoop 系统的工作

机服务状态详述(局部)如图 8 所示,上面十五项是 hadoop 集群中 host1、host6、host7 三台主机上五个服务项的状态信息,这五个服务项分别是:GMOND、check_ganglia_heartbeat、check_ganglia_metric_disk_free、check_ganglia_metric_load_one、check_value_same_everywhere,下面八项是 Nagios 主机上服务的状态信息,本系统中 Nagios 主机即为 host6。从这些效果图可以看出,Ganglia 和 Nagios 协调工作,实现了对 Hadoop 系统性能的监控。

状态,并利用移动飞信进行故障报警,对调整 Hadoop 系统的运行参数、提高系统整体资源效率起到重要作用。

参考文献:

- [1] 袁凯. 云计算环境下的监控系统设计与实现[D]. 武汉: 华中科技大学 2012.
- [2] 张仲妹. 云计算环境下的资源监控应用研究[D]. 北京: 北方工业大学 2013.
- [3] 沈青, 董波, 肖德宝. 基于服务器集群的云监控系统设计与实现[J]. 计算机工程与科学 2012, 34(10): 73-77.
- [4] MATT M, BERNARD L, BRAD N. 陈学鑫, 张诚诚译. Ganglia 系统监控[M]. 北京: 机械工业出版社 2013: 12-15.
- [5] 陶利军. 掌控: 构建 Linux 系统 Nagios 监控服务器[M]. 北京: 清华大学出版社 2013: 8-200.
- [6] 刘鹏, 黄宜华, 陈卫卫. 实战 Hadoop: 开启通向云计算的捷径[M]. 北京: 电子工业出版社 2011: 37-38.
- [7] 李超, 梁阿磊, 管海兵. 海量存储系统的性能管理与监测方法研究[J]. 计算机应用与软件 2012, 29(7): 78-80.

(责任编辑: 李 丽 吴晓红)