

麒麟云平台辅助监控软件设计与实现

魏江涛, 刘 潇, 杨 力, 张 峰

(中国卫星海上测控部, 江苏 江阴 214431)

摘要: 对麒麟云平台核心组件的监测、维护和故障恢复等方法进行了研究, 采用 Qt 及 Shell 技术研发了麒麟云平台辅助监控软件, 该监控软件采用三层架构监控系统模型, 即业务应用、数据采集和数据存储, 实现了对麒麟云平台核心组件状态监测和远程运维, 经测试, 该监控软件图形化直观显示麒麟云平台关键核心组件的工作状态, 满足麒麟云平台监测和运维需求, 提高了麒麟云平台运行可靠性和运维效率, 有效降低了人力维护成本。

关键词: 麒麟云平台; 可靠性; 故障检测; 故障恢复; 运维工具

Design and Implementation of Auxiliary Monitoring Software for KylinCloud Platform

WEI Jiangtao, LIU Xiao, YANG Li, ZHANG Feng

(China Satellite Maritime Tracking and Control Department, Jiangyin 214431, China)

Abstract: Research on the monitoring, maintenance, and fault recovery methods based on the core components of the KylinCloud Platform is conducted. The Qt and Shell technologies are used to develop the KylinCloud Platform auxiliary monitoring software. This software adopts a three-layer architecture monitoring system model: business application, data collection, and data storage, and achieves the statue detection and remote maintenance for the core components of the KylinCloud platform. Through testing, the results show that the monitoring software displays the working status for the core components of the KylinCloud platform in a graphical and intuitive manner, meeting the requirements of detection and maintenance. It improves the operation reliability and maintenance efficiency of the cloud platform, and effectively reduces the maintenance cost.

Keywords: KylinCloud platform; reliability; fault detection; fault recovery

0 引言

随着信息技术发展, 云计算从虚拟化、虚拟化+云管理, 发展到融合云平台, 再到如今的融合 IaaS、PaaS 和 SaaS, 云化的技术架构帮助用户快速完成基础设施和架构建设, 实现业务云化^[1-2]。

中国卫星海上测控部计算机系统是基于 OpenStack 构建的云服务产品, 在飞腾 CPU 的银河麒麟云平台上部署运行测控数据处理软件、远程实时数据交换软件、数据存储软件、数据仿真软件等多项重要软件。该平台由于建设较早, 缺乏有效的运维监控工具, 日常运行维护主要是通过人工查看云平台各个节点和组件参数, 围绕关键特征进行监控、分析^[3]; 这种维护模式的方式相对单一, 需要耗费较大的时间和人力资源; 同时还存在信息滞后, 无法实时监控, 缺乏异常告警等问题^[4],

轻微故障或者一般故障得不到及时处置, 在故障发生后, 问题分析排查定位困难, 容易演变为严重故障。目前, 以 Ganglia、Nagios、Collectd 等技术为代表的云平台运维工具无法适配国产飞腾硬件, 且这些开源工具可定制性和扩展灵活性不足, 无法满足用户多样性监控需求^[5-7]。

因此, 为提升麒麟云平台运维效率和可靠性, 本文研究设计和实现了一款自动化运维工具可以实时监测麒麟云平台的各项指标, 并进行组件维护和故障恢复操作, 实现对麒麟云平台监测、组件维护和故障恢复功能, 从而有效降低维护人力成本, 有效提升麒麟云平台的运维效率和可靠性^[8-9]。

自动化运维工具可以通过收集和分析麒麟云平台的数据, 及时发现问题和异常情况, 并自动进行相应的维

收稿日期:2024-04-25; 修回日期:2024-06-11。

作者简介:魏江涛(1978-),男,硕士,高级工程师。

引用格式:魏江涛,刘 潇,杨 力,等.麒麟云平台辅助监控软件设计与实现[J].计算机测量与控制,2025,33(6):60-66.

护和修复操作。这样可以大大减少故障的处理时间,提高故障恢复的速度和效率^[10-11]。

1 麒麟云平台辅助监控软件设计

1.1 麒麟云平台辅助监控软件功能设计

中国卫星海上测控部采用基于飞腾 CPU 的银河麒麟云平台,该云平台是基于 OpenStack 构建的云服务产品。OpenStack 是开源的云计算管理平台项目,是一系列软件开源项目的组合。OpenStack 提供的可扩展云计算服务可以分为私有云和公有云,其云计算管理平台具有实施简单、大规模扩展性好、丰富、标准统一^[12]。

OpenStack 由 4 大部分组成,包括控制节点、计算节点、网络节点和存储节点,各个节点有不同的分工。控制节点,顾名思义,它负责对其余节点进行控制,比如建立虚拟机、迁移、网络和存储分配等;计算节点承载了虚拟机的运行任务;外网络与内网络之间的通信由网络节点管理;虚拟机的额外存储管理则由存储节点负责^[13-17]。

在 OpenStack 云实例的生命周期中,为了处理计算、网络、存储、身份和镜像,OpenStack 提供了 6 个稳定可靠的核心服务,具体如下:

1) Nova (计算服务): Nova 是 OpenStack 中的计算服务组件,主要负责管理和调度计算资源,包括虚拟机实例的创建、销毁和监控。它是 OpenStack 云平台的弹性控制器,提供了弹性的计算能力,支持自动扩展和负载均衡。Nova 负责各种计算资源、网络、授权和测度等处理支持操作,对整个 OpenStack 云实例生命期进行管理^[18-19]。

Nova 可以创建、启动、停止、重启和销毁虚拟机实例,它提供了对虚拟机实例的完整生命周期管理;它也负责将虚拟机实例调度到合适的物理主机上运行,通过调度算法和资源监控来实现负载均衡和资源优化。Nova 也支持自动扩展并监控虚拟机实例的运行状态和性能指标,根据实际需求如负载情况等动态调整计算资源的数量,提供相应的测度数据,以使用户进行性能优化和故障排查。

2) Neutron (网络服务): Neutron 提供了网络服务,包括虚拟网络的创建、管理和连接。Neutron 是 OpenStack 中的网络服务组件,主要负责虚拟网络的创建、管理和连接。Neutron 允许用户创建和管理虚拟网络,包括子网、路由器、网络拓扑等。用户可以根据自己的需求,灵活地定义和配置虚拟网络。同时 Neutron 负责将虚拟机实例连接到合适的虚拟网络中,提供网络接口和端口的管理,以及虚拟机实例和虚拟网络之间的连接。Neutron 支持软件定义网络技术,允许用户通过

网络控制器来定义和管理网络拓扑。这样用户可以根据实际需求,动态调整网络拓扑和配置。Neutron 提供了安全组和防火墙等网络安全功能,用于保护虚拟网络和虚拟机实例的安全。用户可以定义安全规则,限制网络流量和访问权限。

通过 Neutron 的功能,OpenStack 云平台可以实现灵活的虚拟网络管理和连接,支持用户自定义网络拓扑和安全策略。这为云平台的网络部署和管理提供了更大的灵活性和可扩展性。

3) Cinder (块存储服务): Cinder 是 OpenStack 中的块存储服务组件,主要用于提供持久化的块存储设备,用于虚拟机实例的数据存储和持久化。Cinder 允许用户创建和管理块存储设备,这些设备可以被虚拟机实例挂载和使用。用户可以根据需要,动态地创建和删除块存储设备。Cinder 负责将块存储设备挂载到虚拟机实例上,以供虚拟机实例进行数据存储和访问,支持块存储设备的卸载和解除挂载。Cinder 也支持存储卷的快照和备份功能,用户可以对存储卷进行快照和备份操作,以保护数据的安全性和可靠性。Cinder 允许用户对存储卷进行扩展和缩减操作,以适应不同的存储需求。用户可以根据实际需求,动态调整存储卷的大小,支持多种存储后端,包括本地存储、网络存储和云存储等。用户可以根据自己的需求,选择合适的存储后端。通过 Cinder 的功能,OpenStack 云平台可以提供持久化的块存储服务,为虚拟机实例提供可靠的数据存储和持久化能力。

4) Ceph/Swift (对象存储服务): Swift 是一个分布式对象存储系统,可用于存储大规模的非结构化数据。它提供了高可用性和可扩展性,并支持数据冗余和数据备份。Ceph 也是一个分布式存储系统,负责支持虚拟机镜像的后端数据存储,它能够支持上千个存储节点的规模,支持 TB 到 PB 级的数据。Ceph 支持的 Object 接口也兼容 Swift。麒麟云平台采用 Ceph 分布式对象存储系统,它通过使用一个对象存储设备 (OSD) 运行在每个存储节点上,Ceph 可以通过 OSD 获取关于存储的拓扑信息^[20]。

5) Keystone (身份认证服务): Keystone 提供了身份认证和授权服务,用于管理用户、角色和权限。它提供了单一登录 (SSO) 和身份验证服务,用于保护和管理其他 OpenStack 服务。

6) Glance (镜像服务): Glance 提供了虚拟机镜像的存储和管理功能。它允许用户上传、注册和检索镜像,用于虚拟机实例的创建和部署。

OpenStack 的 6 个核心服务主要担当系统的基础架构,其余项目则负责管理控制面板、编排、裸机部署、

信息传递、容器及统筹管理等。

在运行过程中，为了实现共同计算与共享资源，麒麟云平台依赖虚拟化技术，把多台物理服务器实体虚拟化，从而构成一个资源池。基于 Docker 容器技术可以为主机上的各种应用创建一个轻量级的、可移植的、自给自足的容器，是一款轻量级的“虚拟机”。因此，开发者可以便捷地把应用和依赖包统一打包到一个可移植的容器中，再发布到任何安装了 Docker 引擎的服务器上（Linux 机器、Windows 机器均可），实现虚拟化。

中国卫星海上测控部麒麟云平台总共配置 3 个控制节点、8 个计算节点。针对麒麟云平台的核心关键业务进行监控软件功能设计，主要包含两部分：一是核心组件服务监测功能，二是关键服务组件维护及故障恢复功能，如图 1 所示。

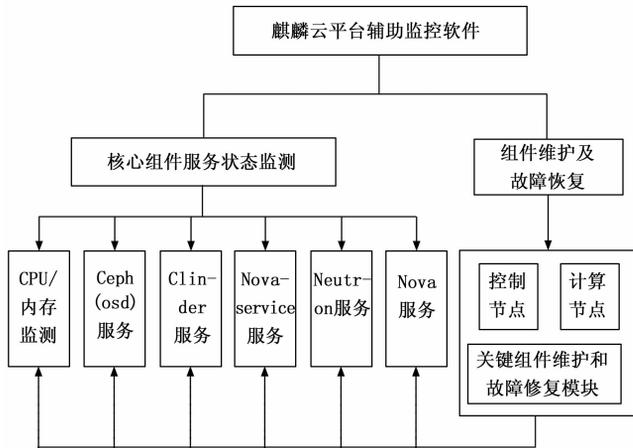


图 1 麒麟云计算平台监控软件功能设计

麒麟云平台核心组件监测功能主要是对支撑麒麟云平台运行的核心组件服务 Ceph（包括 OSD 状态监测）服务、Nova 组件服务、Neutron 组件服务、Clinder 组件服务、Docker 组件服务进行监测，同时辅助监控软件对系统资源进行监测管理，包括 CPU 使用率、内存使用率、硬盘使用率等。在监测功能设计时，对组件服务状态进行监测，可以设置不同等级的监控模式，分为手动模式和自动监控模式。自动监控模式下，辅助监控软件定时对各个组件状态信息进行采集和监测分析，实时显示各个组件的工作情况，异常时及时告警提示。手动监控模式下用户可以根据需要手动采集各个组件的工作状态，手动模式下可以减少信息采集的频率，降低云平台负荷。

麒麟云平台关键服务组件维护及故障恢复功能主要是在核心组件发生异常时，通过组件维护模块对异常的组件服务进行维护和故障修复。由于不同的云平台运维命令和修复指令不同，该监控软件在组件维护和故障修复时针对麒麟云平台的具体情况，设计国产化云平台组件维护和故障修复集，建立常用组件维护和典型故障自动修复集，用户在组件维护和修复时可以进行方便选择使用，并能实时查看维护及故障修复结果。

1.2 麒麟云平台辅助监控软件架构设计

麒麟云平台辅助监控软件由业务功能层、服务层和数据层组成，如图 2 所示。

业务功能层负责提供监控软件的各种功能，如实时监控、告警管理、日志分析等。在实践中主要表现为组件监测显示、组件维护管理和配置管理。组件监测显示是以 Tab 页面的方式直观显示各个核心组件当前的工

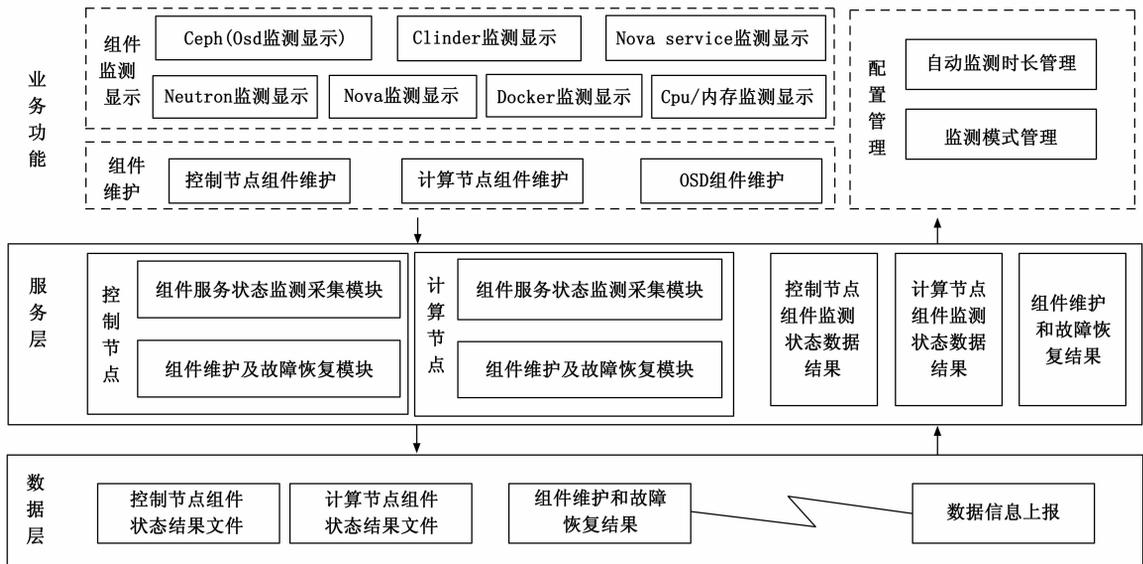


图 2 软件系统架构

作状态,方便用户操作。组件维护管理是对各个重要核心组件进行维护与故障修复,配置管理负责监测模式管理和自动监测时长管理。

服务层提供监控软件的各种服务,如数据采集、数据处理、数据存储等,是辅助监控软件的核心模块,该层负责采集和发送各类监测数据和维护反馈数据,为业务层提供对应的服务支撑。它是连接业务层和数据层的中间桥梁,处理业务层的逻辑和数据操作指令。服务层中控制节点和计算节点利用组件状态采集模块采集、发送、上报采集的数据结果,供业务层监测分析显示,组件维护及故障恢复模块对麒麟云平台的组件进行操控维护和故障修复,将执行结果发送业务功能层显示。

数据层负责接收来自各个监测采集模块的数据,包括监控指标数据、日志数据等,并将其存储到相应的文本文件中;数据层存储监控软件所采集到的数据,是存储服务层中各个监测采集模块采集的数据的存储和管理层。所有的监测数据都以文本文件的形式进行数据传输、信息上报和数据存储等操作。数据层将采集到的数据以文本文件的形式进行传输,更便于在不同的系统之间进行数据交换和共享;数据层将采集到的数据以文本文件的形式进行上报,使得上层系统或管理人员进行数据分析和决策更加便捷,该层负责将采集到的数据存储到相应的文本文件中,以便长期保存和查询。

通过以文本文件的形式进行数据传输、信息上报和数据存储,数据层可以提供灵活、可扩展和易于管理的数据存储和管理方案。同时,文本文件的形式也便于数据的导入和导出,以支持数据的迁移和备份等操作。

麒麟云平台辅助监控软件通过提供丰富的业务功能、可靠的服务和高效的数据存储,辅助管理员对监控系统的运行状态进行实时监控,使得问题能得到及时的发现和解决,系统的稳定性和可靠性也得到大幅提高。

1.3 软件设计采用的关键技术

1.3.1 麒麟云平台核心组件状态监测技术

核心组件监测信息采集涉及对系统或设备的核心组件进行实时或定期的数据收集,是状态监测分析的关键,利用 Shell 采集脚本来完成信息的采集。在软件的实现过程中,通过 Qt 远程调用 Shell 采集脚本,采集结果的输出则由管道和重定向实现,具体实现步骤如下:

1) sshpass 命令非交互式 SSH 登录:

SSH 是 Linux 远程连接的重要方式,它可以在客户端和服务端之间建立安全的连接,从而实现远程访问和管理。使用 SSH 服务可以确保通信过程中的数据传输和身份验证的安全性。SSH 通过加密通信和公钥/私钥的身份验证机制,保护了远程连接的安全性。这使得用户可以在不安全的网络环境中,安全地登录和操作远程

服务器。监测终端微机通过 SSH 服务访问控制节点或计算节点。

在监测终端微机中,通过 SSH 服务可以实现对控制节点或计算节点的访问和监控。监测终端微机可以使用 SSH 客户端软件,通过输入正确的服务器地址、用户名和密码,连接到目标节点。一旦建立了 SSH 连接,监测终端微机就可以执行各种命令、查看和修改文件等操作。

通过使用 SSH 服务,可以确保远程连接的安全性和可靠性,同时提供了对控制节点或计算节点的远程访问和监控能力。

使用 SSH 登录到远程 Linux 服务器时,默认状态下,执行 SSH、Scp 等命令时会出现交互密码输入要求,而利用 sshpass 命令可以实现非交互式 SSH 密码验证,无需再次输入密码。

运行 Shell 采集脚本命令 sshpass:

```
sshpass-p pwd ssh user@remoteserver'command'
```

其中,remoteserver 是远程服务器名,command 为具体的 Shell 命令。例如:查询某计算节点的服务日期,可输入以下语句:

```
sshpass-p1234 root@comuter1 'date'
```

2) 重定向采集结果:

默认情况下,Shell 脚本命令的执行结果是输出到标准输出(stdout,通常为本机显示器),通过 sshpass 在远程服务器上运行 Shell 命令也不例外。在 Linux 系统中,输出重定向可以将结果输出到指定目标位置,调用格式如下:

```
sshpass-p1234 root@comuter1 'date'>tmpFile
```

上述命令将执行结果重定向到了监控终端本机存储执行结果的文件 tmpFile 中,tmpFile 中内容则可辅助监控软件进行解析和显示。

3) 通过 QProcess 类调用状态监测采集脚本:

Shell 脚本直接运行解释器,不需要编译,直接将脚本作为解释器的程序参数运行。通过 QProcess 类对每一个 Shell 采集脚本命令独立创建进程,独立运行,可多任务进程并行而不会相互阻塞,执行效率较高。下面是一个 QProcess 类执行远程调用命令过程:

```
QStringprogram="sshpass-ppwdroot@remoteserver command >tmpFile"
```

```
QStringListarguments;
```

```
//创建输入参数存储列表
```

```
arguments<<str;//str为输入的参数
```

```
QProcess * process =newQProcess();
```

```
process ->start(program,arguments);
```

```
//执行远程调用命令
```

process->waitForStarted();

通过以上步骤，组件状态信息可以被采集、传输和显示，使业务层能够及时了解和监控组件的状态，从而做出相应的决策和调整。

1.3.2 组件维护与故障恢复技术

当组件服务出现异常时，需要进行组件维护和故障恢复，通过控制节点或计算节点进行相应的维护操作。维护和故障修复主要是利用组件维护和故障恢复集来完成，针对各个控制节点或计算节点组件维护和故障修复生成特定 Shell 脚本集，用户只需要选择相应的维护节点即可进行维护和故障恢复操作。

麒麟云平台核心组件维护和故障恢复功能采用应答式通信方式，即监控端向被监测端发起请求，被监测端进行响应并返回响应结果。监控端在发起维护请求时，同样需要利用 sshpass 命令实现非交互式 ssh 登录，在被监测端执行修复指令，被监测端将修复指令的执行结果实时返回到监控终端。

要实现监控端实时显示故障恢复的反馈结果，可以通过 Qt 的信号与槽机制来实现。以下是一种可能的实现方式：

1) 在监控端的代码中，创建一个槽函数来处理被监测端返回的修复结果。该槽函数将负责接收并处理修复结果；

2) 在监控端代码中，与 SSH 命令执行相关的代码中，将修复结果与槽函数进行关联。这样，当修复结果返回时，槽函数将被触发；

3) 在槽函数中，对修复结果进行解析和处理。根据实际需要，可以将修复结果显示在监控终端的界面上，或者执行其他后续操作。

从而监控端可以实时接收并显示被监测端的修复结果，及时了解故障恢复的反馈情况。槽函数关联的具体过程如下：

```

QProcess * process = new QProcess();
//新增一个 QProcess 对象用于执行当前脚本 connect
(process,&QProcess::readyReadStandardOutput,this,[=]())
{ui->textBrowser_Output->append(QString(process->readAll-
StandardOutput().constData()));});
// 将 QProcess 进程 readyReadStandardOutput 信号
//进行槽函数关联,输出到监控终端。

```

2 麒麟云平台辅助监控软件功能实现

麒麟云平台辅助监控软件采用 Qt5.9 进行研发实现，并在中国卫星海上测控部实际应用环境中进行测试，可以有效验证麒麟云平台辅助监控软件的性能和稳定性。通过在实际应用环境中的测试过程中，收集和析了软件的运行数据，包括性能指标、故障情况等，可

以更全面地评估软件在实际使用中的表现，确保麒麟云平台辅助监控软件能够在实际应用中发挥出最佳的作用，及时发现并解决可能存在的问题和缺陷，确保其在实际应用中能够发挥出最佳的作用，满足用户的需求。

图 3 为 Ceph 和 OSD 监测结果图，指示灯绿色通常表示 OSD 工作状态正常，OSD 正在正常运行和提供服务。指示灯白色表示 OSD 未启用，即该 OSD 当前未被使用或处于非活动状态。指示灯红色表示 OSD 工作状态异常，OSD 发生了故障或出现了其他问题，需要进行故障排除和修复。Ceph 状态正常为 HEALTH_OK 状态，并显示为绿色，异常时显示为红色。通过对 OSD 和 Ceph 工作状态的图形化直观显示，方便系统用户实时查看。



图 3 Ceph 及 osd 监测结果图

图 4 显示了麒麟云平台 Neutron 核心组件的实时监测结果，alive 状态列显示当前组件的工作状态，正常时绿色显示，而组件出现异常时该状态列红色突出显示，说明该组件需要进行故障排除和修复。通过 alive 状态列的文字的颜色可以直观地显示组件的服务状态，让系统用户能够快速识别和解决系统异常情况，提高系统的可靠性和稳定性。

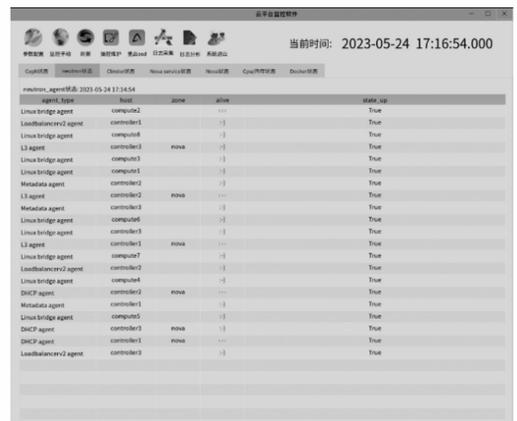


图 4 Neutron 监测结果图

图 5 为麒麟云平台 Clinder 核心组件的监测结果, 组件工作状态正常, Status 状态列会以绿色 “up” 状态表示, 异常状态时会以红色 “down” 状态表示。异常状态会被突出显示, 从图中可以看出计算节点 2 的组件服务存在异常。通过图 5 的状态列的直观显示, 用户可以快速识别和解决存储服务的问题, 直观地了解各个计算节点的组件服务状态, 帮助用户提高系统的可靠性和稳定性, 满足系统用户的运维需求。

Host	name	Status	State	Updated_at
compute1	nova	enabled	up	2023-05-24T09:15:00.000000
compute2	nova	enabled	up	2023-05-24T09:15:04.000000
compute3	nova	enabled	up	2023-05-24T09:15:09.000000
compute4	nova	enabled	up	2023-05-24T09:15:06.000000
compute5	nova	enabled	up	2023-05-24T09:15:05.000000
compute6	nova	enabled	up	2023-05-24T09:15:03.000000
compute7	nova	enabled	up	2023-05-24T09:15:07.000000
compute8	nova	enabled	up	2023-05-24T09:15:02.000000
control01	nova	enabled	up	2023-05-24T09:15:05.000000
control02	nova	enabled	up	2023-05-24T09:15:11.000000
control03	nova	enabled	up	2023-05-24T09:15:02.000000
compute010ad-1	nova	enabled	up	2023-05-24T09:15:03.000000
compute010b-1	nova	enabled	up	2023-05-24T09:15:06.000000
compute010c-1	nova	enabled	up	2023-05-24T09:15:05.000000
compute010d-1	nova	enabled	up	2023-05-24T09:15:02.000000
compute010e-1	nova	enabled	up	2023-05-24T09:15:03.000000
compute010f-1	nova	enabled	up	2023-05-24T09:15:07.000000
compute010g-1	nova	enabled	up	2023-05-24T09:15:02.000000
compute010h-1	nova	enabled	up	2023-05-24T09:15:04.000000
compute010i-1	nova	enabled	up	2023-05-24T09:15:02.000000

图 5 Clinder 监测结果图

图 6 为麒麟云平台组件恢复及故障恢复操控界面, 维护人员可以准确地选择需要维护和修复的组件, 进行快速地故障维护和故障修复。用户可以在故障集中选择需要修复的故障, 可方便快速地执行故障修复操作, 例如重新启动 mon 服务、重启 docker 服务等, 用户可以选择在某个控制节点及计算节点进行操控, 修复指令执行后, 修复指令的执行结果会在图中右侧 “状态维护结果” 进行显示, 方便系统用户可以及时了解修复的状态和结果, 满足系统用户远程运维需求。



图 6 组件维护及故障恢复界面

通过收集性能指标和故障情况等数据, 可以深入了解软件的运行状态, 包括响应时间、资源消耗、错误和

异常等。通过对这些数据的分析, 可以评估软件的性能、可靠性和稳定性, 发现潜在的问题, 并进行相应的优化和改进。

3 结束语

本文基于 Qt 及 Shell 等技术构建了麒麟云平台辅助监控软件架构, 该软件能够实时收集分析麒麟云平台的各项数据, 监控麒麟云平台的各个组件的运行状态, 及时发现问题和异常情况, 保证平台的正常运行; 同时该软件能够自动识别麒麟云平台的故障, 进行恢复操作, 减少平台的停机时间。实践证明, 软件具有较高的监测效率, 能够实时监测麒麟云平台的各项指标, 及时发现潜在问题, 可以提升麒麟云平台的稳定性和性能。

由于人力、时间和实验环境等因素的限制, 本系统还有许多不足的地方可改进完善。对于未来的工作, 主要有以下几个方面值得重点研究、改进、完善和补充, 比如智能化数据收集分析, 为发现更多的隐藏问题和对现行方案进行优化, 可对收集到的数据进行更深入的分析 and 挖掘; 其次是监控效率方面的提升, 进一步完善系统的监控指标和算法, 对各个指标进行综合分析, 以更准确地判断平台的运行状态和问题的严重程度。当然后续实践中, 需要不断收集数据, 根据历史数据和模型预测, 提前预防潜在的故障和问题, 进一步改进故障检测算法, 提高故障检测准确率, 减少误报和漏报的情况以完善系统的准确性和可靠性。

参考文献:

- [1] 丁秀雄. 云服务事件异常检测与分析系统的设计与实现 [D]. 西安: 西安电子科技大学, 2020.
- [2] 张亚丹. 云计算平台故障检测关键技术研究 [D]. 北京: 北京交通大学, 2016.
- [3] 姚杰. 基于日志的分布式系统异常诊断方法研究 [D]. 南京: 南京邮电大学, 2021.
- [4] 祝蓓. 基于深度学习的云平台日志异常检测方法研究 [D]. 北京: 南京航空航天大学, 2021.
- [5] 张靖雯. 基于深度学习的工控异常检测及攻击分类方法研究 [D]. 北京: 北京工业大学, 2019.
- [6] 刘潇, 张斌, 姜励锋, 等. 一种国产麒麟私有云故障分析处理方法 [J]. 价值工程, 2023, 42 (23): 146 - 149.
- [7] 张之宜. 云计算环境下实时日志分析系统的设计与实现 [D]. 杭州: 浙江大学, 2016.
- [8] 雷惊鹏, 唐雅文, 颜世波, 等. 基于 Docker 和 Kubernetes 的 ELK 日志分析系统的研究与实现 [J]. 太原学院学报 (自然科学版), 2021, 39 (2): 61 - 67.
- [9] 任明, 宋云奎. 基于深度学习的云计算系统异常检测方

- 法 [J]. 计算机技术与发展, 2019, 29 (5): 54-57.
- [10] 王力群, 黄必栋. 基于日志分析平台的监控系统的设计与实现 [J]. 计算机应用与软件, 2017, 34 (12): 158-162.
- [11] 朱宝金. 面向云计算系统的日志过滤系统的设计与实现 [D]. 杭州: 杭州电子科技大学, 2014.
- [12] 夏敏纳, 龚德良, 肖 娟. 一种面向可靠云计算的自适应故障检测方法 [J]. 计算机应用研究, 2014, 31 (2): 426-430.
- [13] 李晨光. UNIX/Linux 网络日志分析与流量监控 [M]. 北京: 机械工业出版社, 2015.
- [14] 饶 翔, 王怀民, 陈振邦, 等. 云计算系统中基于伴随状态追踪的故障检测机制 [J]. 计算机学报, 2012, 32 (5): 856-870.
- [15] 王冬阳. 面向云计算的异常检测技术的研究与实现 [D]. 上海: 上海交通大学, 2012.
- [16] 张建勋, 古志民, 郑 超. 云计算研究进展综述 [J]. 计算机应用研究, 2010, 27 (2): 429-433.
- [17] 李 鹏. 麒麟操作系统实时性能测试方法研究 [J]. 渤海大学学报 (自然科学版), 2016, 37 (3): 278-283.
- [18] 马 民, 孙国忠, 张丽艳. 云计算平台事件处理机制研究 [J]. 系统仿真技术, 2014, 10 (4): 313-320.
- [19] 程 娟, 覃 燕. 基于 LabVIEW7.0 的某实时监控软件设计与实时性分析 [J]. 现代电子技术, 2011, 34 (12): 186-188.
- [20] 胡家铭, 马 辰, 宋 达, 等. 基于国产化云平台的大数据平台性能分析 [J]. 数字化用户, 2023, 29 (24): 4-6.
- 液压, 2021, 49 (16): 185-190.
- [10] 万里瑞, 王康康, 王 辉. 利用增强多尺度模糊熵的齿轮故障诊断方法 [J]. 机械设计与研究, 2021, 37 (5): 73-77.
- [11] 张海龙, 李 博, 贾娜娟. 基于 Dijkstra 算法的电力光通信网络智能运维设备脱网故障区段定位方法 [J]. 电信科学, 2023, 39 (1): 92-99.
- [12] 王晓康, 牛 勃, 马飞越, 等. 电气设备局部放电融合诊断与智能预警系统研究 [J]. 高压电器, 2021, 57 (11): 93-100.
- [13] 商敬安, 韩 磊, 吴 彬, 等. 基于自注意力机制的多尺度电力机房二次设备巡检算法 [J]. 计算机应用, 2022, 42 (s1): 366-370.
- [14] 刘皓璐, 邵建伟, 王 雪, 等. 基于数字孪生的配电自动化终端设备状态评价与故障预判 [J]. 电网技术, 2022, 46 (4): 1605-1613.
- [15] 王桥梅, 吴 浩, 胡潇涛, 等. 基于 VMD 多尺度模糊熵的 HVDC 输电线路故障识别方法 [J]. 电力系统及其自动化学报, 2021, 33 (5): 134-144.
- [16] 车守全, 江 伟, 包从望, 等. EWT-多尺度模糊熵-VPMCD 融合算法的轴承故障识别分类应用 [J]. 机械科学与技术, 2021, 40 (9): 1397-1403.
- [17] 余孟阳, 杨 帆. 信息链接技术下自动化设备现场故障运维仿真 [J]. 计算机仿真, 2021, 38 (12): 475-479.
- [18] 杨 杰, 吴 浩, 胡潇涛, 等. 基于多尺度行波功率的 T 接线路故障识别方法 [J]. 电力系统及其自动化学报, 2021, 33 (4): 115-126.
- [19] 赵家浩, 廖晓娟, 唐锡雷. 基于改进多元多尺度加权排列熵的齿轮箱故障诊断 [J]. 组合机床与自动化加工技术, 2022 (12): 48-52.
- [20] 张 锴, 曹张保, 姜静飞. 基于中压开关柜相电流异常点数据的故障预警方法 [J]. 城市轨道交通研究, 2022, 25 (3): 207-212.
- (上接第 59 页)
- 干扰 3 种环境下均可以有效预警系统故障, 且预警响应速度较快, 可以实现实时预警, 具有一定的实用价值.

参考文献:

- [1] 许志浩, 郑诗泉, 康 兵, 等. 基于三相自搜寻比较法的电气设备过热故障识别方法 [J]. 红外技术, 2021, 43 (11): 1112-1118.
- [2] 李 岩, 刘玉娇, 李国亮, 等. 基于改进 BEMD 与 Res-LSTM 电气设备故障辨识方法 [J]. 高压电器, 2022, 58 (11): 82-87.
- [3] 姚聪伟, 庞小峰, 孙 帅, 等. 基于二次时间信息的断路器操动机构状态预警方法研究 [J]. 高压电器, 2022, 58 (11): 114-120.
- [4] BAI H, YU B. Position estimation of fault-tolerant permanent magnet motor in electric power propulsion ship system [J]. IEEJ Transactions on Electrical and Electronic Engineering, 2022, 17 (6): 890-898.
- [5] SHARMA M, RAJPUROHIT B S, AGNIHOTRI S, et al. Data analytics based power quality investigations in emerging electric power system using sparse decomposition [J]. IEEE Transactions on Power Delivery, 2022, 37 (6): 4838-4847.
- [6] 王 彪, 贾彦斌, 李茨兴, 等. 基于 SC-BP 的设备故障预警研究 [J]. 火力与指挥控制, 2023, 48 (3): 113-117.
- [7] 韩慧苗, 许 昕, 潘宏侠, 等. 基于模糊熵与 CS-ELM 的供输弹系统早期故障识别 [J]. 机床与液压, 2022, 50 (7): 164-169.
- [8] 汤占军, 孙润发. 基于多尺度模糊熵和 STOA-SVM 的风机轴承故障诊断 [J]. 电机与控制应用, 2021, 48 (12): 66-70.
- [9] 王金东, 陈 新, 赵海洋, 等. 基于精细复合多尺度模糊熵的往复压缩机轴承间隙故障特征分析方法 [J]. 机床与