雷达图

中度可信度描述已自动生成

文档版本 v1.2.9

发布日期 2022-8-7

**QingCloud企业云**

**平台巡检手册**

1. 文档描述

|  |  |
| --- | --- |
| 标题 | QingCloud企业云平台巡检手册\_v1.2.9 |
| 项目名称 |  |
| 类别 | 规范文档 □ 安装文档 □ 配置文档 □  设计方案 □ 实施文档 □ 测试文档 □  需求文档 □ 行政管理 □ 其他 ■ |
| 摘要 |  |
| 创建日期 | 2020年2月5日 |
| 文档所属部门 |  |
| 项目执行部门 |  |
| 文档作者 | 李晓博 |
| 文件名称 |  |

1. 文档发布

|  |  |
| --- | --- |
| 存档名称 | QingCloud企业云平台巡检手册\_v1.2.9 |
| 版本号 |  |
| 发布时间 |  |
| 文档状态 |  |

1. 修改记录

|  |  |  |  |
| --- | --- | --- | --- |
| 版本号 | 修改日期 | 作者/修改者 | 修改内容 |
| v1.0 | 2020年2月5日 | 李晓博 |  |
| v1.1 | 2020年2月26日 | 李晓博 | 增加青云架构、适用对象等 |
| v1.2 | 2020年2月28日 | 李晓博 | 加入BM、Border、h3cvg等项 |
| V1.2.1 | 2020年3月15日 | 刘凯 | 添加SSD卡OP空间检查，IO性能检查 |
| V1.2.2 | 2020年5月28日 | 刘凯 | 将巡检项按巡检周期分类。 |
| V1.2.4 | 2020年10月22日 | 刘凯 | 文档更新 |
| V1.2.5 | 2021年07月07日 | 冯佩 | 增加NOVG巡检，面向客户细化操作步骤 |
| V1.2.6 | 2022年06月08日 | 冯佩 | 8.1.1 8.1.2章节增加老版本postgressql和pgpool相关命令和输出 |
| V1.2.7 | 2022年08月07日 | 刘凯 | 增加pgserver和pgpool连接数检查。  增加网络和IO性能检查  增加LB检查。  增加磁盘Media Error检查 |
| V1.2.8 | 2022年09月21日 | 冯佩 | 优化了操作命令，增加了ip地址余量检查等 |
| V1.2.9 | 2022年10月28日 | 冯佩 | 增加了账户类型巡检，修改了rsserver巡检，优化了巡检命令 |

1. 文档审批

|  |  |  |
| --- | --- | --- |
| 项目职位 | 批准人 | 日期 |
|  |  |  |
|  |  |  |
|  |  |  |

**目录**

[1. 文档说明 7](#_Toc8847)

[1.1. 适用对象 7](#_Toc12091)

[1.2. 关键术语 7](#_Toc19022)

[1.3. 参考文档 8](#_Toc26947)

[1.4. 巡检目的 8](#_Toc24971)

[1.5. 巡检范围 8](#_Toc18476)

[2. 青云架构概述 9](#_Toc30984)

[2.1. 环境基本信息 9](#_Toc32424)

[2.2. 基本组件 10](#_Toc4573)

[3. 巡检要求 11](#_Toc26093)

[3.1. 无驻场的项目 11](#_Toc4834)

[3.2. 有驻场的项目 12](#_Toc23047)

[4. 巡检项目-全部节点 12](#_Toc32314)

[4.1. 每天巡检 12](#_Toc14768)

[4.1.1. qingcloud进程状态 12](#_Toc4177)

[4.1.2. 时间同步状态 14](#_Toc11728)

[4.1.3. 文件系统空间 14](#_Toc31706)

[4.1.4. 剩余内存 15](#_Toc15895)

[4.1.5. 系统负载 16](#_Toc21122)

[4.1.6. 系统进程 16](#_Toc9038)

[4.2. 每周或按月检查 16](#_Toc26740)

[4.2.1. NTP状态 16](#_Toc3412)

[4.2.2. NTP源 17](#_Toc12525)

[4.2.3. 系统kern日志 18](#_Toc484)

[4.2.4. Netconsole日志 19](#_Toc17513)

[4.2.5. Inode使用率 19](#_Toc25231)

[4.2.6. Crontab状态 20](#_Toc1325)

[4.2.7. supervisord状态 20](#_Toc23446)

[4.2.8. vxlan port状态 21](#_Toc15680)

[4.2.9. 预付费账户检查 21](#_Toc19079)

[5. 巡检项目-物理机节点 22](#_Toc10511)

[5.1. 每天巡检 22](#_Toc25290)

[5.1.1. 物理机网卡bond状态 22](#_Toc15442)

[5.2. 每周或按月检查 23](#_Toc18301)

[5.2.1. 物理机网卡状态 23](#_Toc6889)

[5.2.2. Memblaze Flash卡OP空间检查(可选) 23](#_Toc19792)

[5.2.3. 磁盘Media Error检查 24](#_Toc19910)

[6. 巡检项目-zoocassa 26](#_Toc27726)

[6.1. 每天检查 26](#_Toc15405)

[6.1.1. Zookeeper状态 26](#_Toc26785)

[6.1.2. cassandra状态 26](#_Toc29632)

[6.1.3. memcached状态 27](#_Toc13023)

[6.1.4. redis状态（仅在global环境执行） 28](#_Toc24125)

[7. 巡检项目- webservice 29](#_Toc17252)

[7.1. 每周或按月检查 29](#_Toc14806)

[7.1.1. apache状态 29](#_Toc4541)

[7.1.2. nginx状态 30](#_Toc4481)

[8. 巡检项目-pgserver和pgpool 31](#_Toc4151)

[8.1. 每周或按月检查 31](#_Toc7364)

[8.1.1. pgserver状态 31](#_Toc5381)

[8.1.2. pgserver连接数 32](#_Toc27320)

[8.1.3. pgpool状态 33](#_Toc9643)

[8.1.4. pgpool连接数 35](#_Toc6752)

[9. 巡检项目-Proxy和DNS 36](#_Toc32386)

[9.1. 每周或按月检查 36](#_Toc12939)

[9.1.1. Haproxy状态 36](#_Toc16272)

[9.1.2. Nginx状态 37](#_Toc32208)

[9.1.3. Bind9状态 38](#_Toc12566)

[10. 巡检项目-计算节点 39](#_Toc13160)

[10.1. 每天检查 39](#_Toc25972)

[10.1.1. DRBD状态 39](#_Toc15354)

[10.1.2. repl状态 39](#_Toc27483)

[10.1.3. zpool状态 40](#_Toc6541)

[10.2. 每周或按月检查 41](#_Toc6758)

[10.2.1. swap状态 41](#_Toc29756)

[10.2.2. IO性能检查 42](#_Toc5513)

[10.2.3. 单节点VM数量 43](#_Toc6499)

[10.2.4. IP forward状态 43](#_Toc14848)

[10.2.5. 透明大页 44](#_Toc26973)

[10.2.6. 维护工具 46](#_Toc9883)

[11. 巡检项目-IAAS平台管理 47](#_Toc20494)

[11.1. 每天巡检 47](#_Toc9892)

[11.1.1. console控制台 47](#_Toc1732)

[11.2. 每周或按月检查 48](#_Toc25773)

[11.2.1. 计算节点状态 48](#_Toc4674)

[11.2.2. 资源使用率 48](#_Toc11525)

[11.2.3. 操作日志 48](#_Toc17896)

[11.2.4. 评估计算资源用量 48](#_Toc1193)

[11.2.5. 查看基础网络地址余量 49](#_Toc7512)

[12. 巡检项目-X86 VG节点 49](#_Toc16617)

[12.1. 每周或按月检查 49](#_Toc2056)

[12.1.1. EIP数量 49](#_Toc27938)

[12.1.2. CPU负载状态 51](#_Toc24225)

[12.1.3. IP forward状态 52](#_Toc30510)

[12.1.4. EIP流量排序 53](#_Toc14364)

[13. 巡检项目-物理主机（BM） 53](#_Toc8320)

[13.1. 每周或按月检查 54](#_Toc17658)

[13.1.1. BM服务 54](#_Toc24843)

[13.1.2. TFTP服务 54](#_Toc18699)

[13.1.3. boot\_image检查 55](#_Toc4702)

[13.1.4. gobgp邻居关系检查 55](#_Toc20640)

[13.1.5. BM日志 56](#_Toc11417)

[14. 巡检项目-Border 56](#_Toc25630)

[14.1. 每周或按月检查 57](#_Toc27696)

[14.1.1. Border版本 57](#_Toc21830)

[14.1.2. ssh 服务检查 58](#_Toc11922)

[14.1.3. 检查用户信息 59](#_Toc23477)

[14.1.4. BGP邻居关系 59](#_Toc1321)

[15. 巡检项目-H3C 6800 VG（VG3.0） 60](#_Toc8805)

[15.1. 每周或按月检查 60](#_Toc14680)

[15.1.1. Eipctl控制器服务 60](#_Toc13881)

[15.1.2. BGP邻居关系 61](#_Toc4444)

[15.1.3. H3C与eipctl链接 61](#_Toc1492)

[15.1.4. H3C交换机eip\_network信息 62](#_Toc25913)

[16. 巡检项目-rsserver 64](#_Toc25937)

[16.1. 每周或按月检查 64](#_Toc19489)

[16.1.1. rsserver服务 64](#_Toc23704)

[16.1.2. rsserver的BGP邻居关系 65](#_Toc11761)

[16.1.3. 交换机的BGP邻居关系 66](#_Toc32089)

[16.1.4. 查看交换机的路由信息 66](#_Toc8634)

[17. 性能分析 67](#_Toc9725)

[17.1. 网络性能分析 67](#_Toc2416)

[17.2. IO性能分析 68](#_Toc272)

[18. LB负载均衡器 68](#_Toc7222)

[19. IAAS巡检总结和建议 69](#_Toc19686)

[19.1. 巡检总结 69](#_Toc23518)

[19.2. 巡检建议 69](#_Toc5520)

# 文档说明

## 适用对象

文档适用于对青云平台有一定了解的运维人员。

运维人员需要掌握青云平台基础架构和青云平台使用服务名称以及功能，还需要掌握linux系统基础命令的使用。

本文档针对青云IAAS平台巡检，主要包含VG、KS和Hyper等节点服务。

## 关键术语

|  |  |
| --- | --- |
| **服务名** | **功能说明** |
| API Server | 对外提供API服务，是青云核心系统的接入层，用户的所有请求包括控制台的请求以及API的请求都会经过其向后端服务发送 |
| ws | global的消息入口 |
| billing | 计费模块 |
| distributed | 分布式锁服务 |
| nf | 通知模块，包括和用户相关的通知和告警 |
| boss | boss系统服务模块 |
| io | IO和zone里的PUSH server结合起来完成web push功能，即资源的动态变化会推送到客户端 |
| zk | 名字服务，所有qingcloud的服务都会向zk注册自己的信息 |
| pg | 关系型数据库，存放用户信息和global的资源信息 |
| cass | 存放监控数据，允许丢失并最终一致 |
| mc | 存放缓存数据，允许丢失 |
| fg | zone的消息入口，zone和global消息交互的唯一通道，主要功能为用户界面和API服务的提供接口 |
| scheduler | 实现控制台上“定时器”的功能，可用来定期执行一系列任务 |
| cb | web terminal的proxy server，控制台上“vnc”的功能，使用户可以通过浏览器访问到主机实例 |
| push | 配合global的IO server完成web push的功能 |
| distributed | 分布式锁服务 |
| nf | 通知模块，和用户无关的通知和告警 |
| alarm | 告警模块，负责收集报警类信息，包括错误，报警信息 |
| autoscalling | 实现控制台上“自动伸缩”的功能，可以帮助用户基于资源的监控告警规则动态调节配置或集群规模 |
| billing\_delegator | 实现流量计费的功能 |
| zk | 名字服务，所有qingcloud的服务都会向zk注册自己的信息 |
| pg | 关系型数据库，存放zone资源信息 |
| cass | 存放监控数据，允许丢失并最终一致 |
| mc | 存放缓存数据，允许丢失 |

## 参考文档

## 巡检目的

为了更好的维护和管理青云系统和虚机业务，需要定期对青云系统整体环境进行巡检。对预期风险进行及时处理。本文档对青云平台的巡检内容，操作方法，判断标准及处理建议进行说明，并提供了巡检样例，旨在为巡检人员提供操作和判断参考。

## 巡检范围

|  |  |  |
| --- | --- | --- |
| 全部节点 |  | 必选 |
| 物理机节点 |  | 必选 |
| Zoocassa |  | 必选 |
| Webservice |  | 必选 |
| Pgserver&pgpool |  | 必选 |
| Proxy&Dns |  | 必选 |
| 计算节点（hyper） |  | 必选 |
| IAAS平台管理 |  | 必选 |
| X86 VG节点 |  | 根据巡检环境 |
| 物理主机（BM） |  | 根据巡检环境 |
| Border |  | 根据巡检环境 |
| H3C 6800 VG |  | 根据巡检环境 |
| rsserver |  | 根据巡检环境 |
| 性能分析 |  | 必选 |
| LB负载均衡器 |  | 必选 |

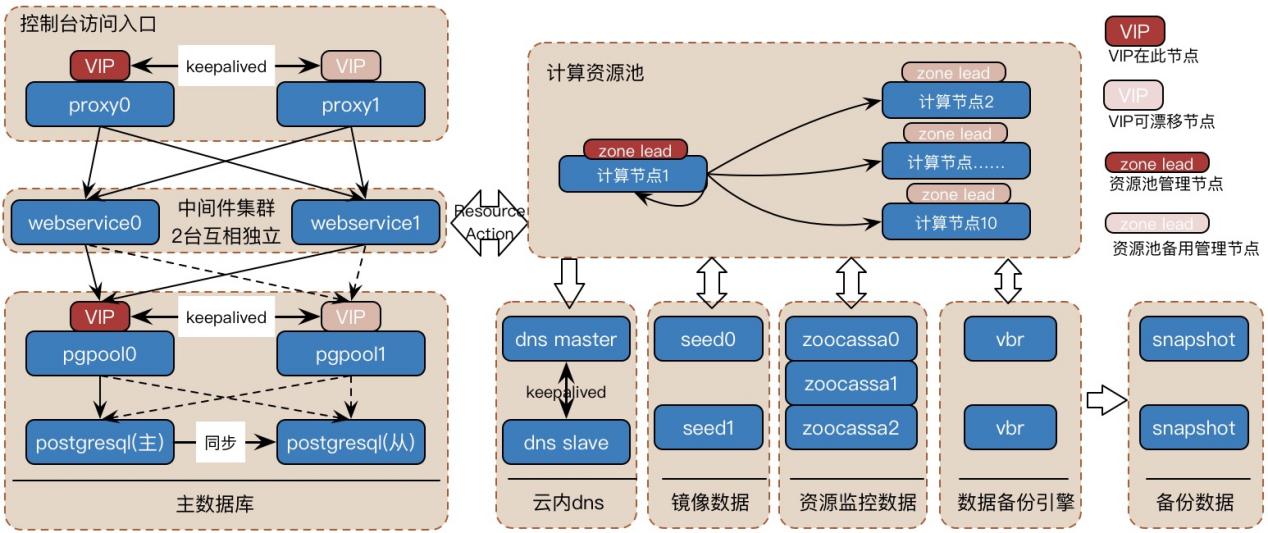
# 青云架构概述

## 环境基本信息

* 该部分内容，一般是首次巡检或者平台有变更时需要做

|  |  |  |  |
| --- | --- | --- | --- |
| 检查项 | 环境信息 | 备注 |  |
| IAAS版本 |  | root@qingcloud-firstbox:~# cat /pitrix/version.json |  |
| 物理机及管理虚拟机OS版本 |  | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh all 'lsb\_release -a|grep Description' |  |
| 网络架构 | SDN2.0/SDN3.0/VLAN | 命令：  root@qingcloud-firstbox:~#grep sdn\_driver /pitrix/conf/variables/global/server.yaml.<ZONE\_ID>  输出：  sdn\_driver: 'device'  sdn\_driver: 'nfv'  说明：  sdn\_driver: 'device' 表示vlan模式  sdn\_driver: 'nfv' 表示SDN模式  <ZONE\_ID>请替换为实际zone\_id，每个zone都要执行一次 |  |
| 现有基础网络 |  | 登陆console查看基础网络： |  |
| 存储架构 | SDS1.0/SDS2.0/SANC/NASC | 登陆BOSS查看存储架构： |  |
| 节点数量 | KS  VBR  计算节点 | root@qingcloud-firstbox:~#  cat /pitrix/conf/nodes/ks-phy|awk '{print NF-2}'  root@qingcloud-firstbox:~#  cat /pitrix/conf/nodes/vbr|awk '{print NF-2}'  root@qingcloud-firstbox:~#  cat /pitrix/conf/nodes/hyper|awk '{print NF-2}' |  |
| Globlzone |  | root@qingcloud-firstbox:~#  grep is\_global\_zone /pitrix/conf/variables/variables.yaml |  |
| 子zone |  | root@qingcloud-firstbox:~#  grep is\_sub\_zone /pitrix/conf/variables/variables.yaml |  |
| 其它环境比较特殊的内容 |  |  |  |

## 基本组件



青云个角色在私有云生产环境中的 关联与依赖（数字越小优先级越高，未标明虚拟机的都要求为物理机）

除了hyper以外 的其他角色最好每个都能有两个以达到高可用的目的。

在保证ks节点的完 整性的前提下，hyper的多少直接决定了你的集群的业务量上限以及跑复杂应用（大数据/lb集群 /redis集群…)的能力，以及优化资源分配的能力

计算节点（Hyper）承载 所有青云的计算服务，规划时这部分节点不能同任何其他角色共存，并且要尽可能多规划机器给它

pgserver ：业务数据库，需要以物理机的形式放到io性能高的节点上

pgpool ：数据库连接池，最好放到物理机上

zoocassa：业务数据库／缓存和hyper节点的leader选举，需要以物理机的形式放到io性能高 的节点上（需要奇数个，并 且至少３个）

vgateway：主要承载同客户的外网的互 联，它由于是cpu密集型角色，最好以物理机的形式部署（vg的重要性还要看2.0网络在私有云的部署 方式）

webservice：（虚拟机），承载前端页面以及 相关的api/数据库服务

dns- master/dns-slave：（虚拟机）域名访问服务，（通常两个角色各创建一个）

proxy ：代理服务（虚拟机）（用户internet网入口，子zone／global zone通讯入口，各种业务端口转发），需要能访问外网，通常放在vg上

vbr：为大容量的数据提供 压缩服务，通常为大容量 sata盘，平时较为空闲，工作时会占用大量cpu/io，所以建议在机器不够的时候，可以和其他一些负载不重的大容量节点放在 一起（比如seed)

seed：存放系统镜像和用户自定义镜像，sata盘，通常是非常冷的数据 由于是io密集型，建议最好单独部署，业务量不大的初期可以考虑和vbr一起

snapshot：存放用户的快照，sata盘，通常是较冷数据，只有用户创建快照的时候会使用，由于是io密集型，建议单独部署。

# 巡检要求

## 无驻场的项目

* 终端命令行中开启日志输出，将操作过程的屏显输出到文件中，文件和巡检报告需同时发送给青云对应项目的项目经理。

putty：

在全局配置中，点击Logging，勾选上Printable output，选定好日志的输出路径，之后登陆即可，在登录中的操作记录会记录到指定日志中

Xshell：

步骤1.依次点击文件、日志、启动即可启动xshell的屏幕记录功能

步骤2.开始执行特定的脚本，屏幕会输出一大堆的东西，这些记录会记录到步骤1指定的文件中。

步骤3.依次点击文件、日志、停止即可停止xshell的屏幕记录功能

* 巡检工作需向客户邮件申请，申请邮件抄送青云对应项目经理，巡检报告需发送给客户，需客户签字确认或邮件确认。
* 对于无驻场的环境，由合作伙伴按月对本文档中提到的内容进行巡检。
* 对有要求“这里插入输出结果的文件”的巡检项，请按要求保留相关结果文件。

## 有驻场的项目

* 标为每周或每月的检查，对有驻场的项目，是每周进行一次巡检。
* 标为每天巡检的检查项，需每天进行巡检，并在工作群内汇报巡检结果。
* 终端命令行中开启日志输出，将操作过程的屏显输出到文件中，文件和巡检报告需同时发送给青云对应项目项目经理。

putty：

在全局配置中，点击Logging，勾选上Printable output，选定好日志的输出路径，之后登陆即可，在登录中的操作记录会记录到指定日志中

Xshell：

步骤1.依次点击文件、日志、启动即可启动xshell的屏幕记录功能

步骤2.开始执行特定的脚本，屏幕会输出一大堆的东西，这些记录会记录到步骤1指定的文件中。

步骤3.依次点击文件、日志、停止即可停止xshell的屏幕记录功能

# 巡检项目-全部节点

以下巡检项目，需要在青云平台上的全部节点执行。

## 每天巡检

### qingcloud进程状态

|  |  |
| --- | --- |
| 检查项 | qingcloud进程状态 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh all "supervisorctl status"  root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh all "supervisorctl status|grep -v RUNNING" |
| 判断标准 | 正常是所有服务状态为RUNNING |
| 检查结果 | 请在此说明QingCloud服务运行是否正常。  如果有非running状态的，请在下方列出。 |
|  |

正常输出示例：

Hyper

|  |
| --- |
| supervisorctl status  boss\_daemon RUNNING pid 51391, uptime 116 days, 19:59:01  compute\_agent RUNNING pid 43349, uptime 116 days, 20:36:30  compute\_monitor RUNNING pid 42948, uptime 116 days, 20:36:34  compute\_proxy RUNNING pid 2134, uptime 325 days, 4:27:57  compute\_server RUNNING pid 43544, uptime 116 days, 20:36:26  libvirtd RUNNING pid 59865, uptime 116 days, 19:21:54  repl\_proxy RUNNING pid 27109, uptime 293 days, 18:51:04  repl\_server RUNNING pid 42868, uptime 116 days, 20:36:35  zk\_proxy RUNNING pid 2135, uptime 325 days, 4:27:57 |

webservice

|  |
| --- |
| supervisorctl status  alarm\_server RUNNING pid 20038, uptime 116 days, 20:03:00  autoscaling\_broker RUNNING pid 20043, uptime 116 days, 20:03:00  autoscaling\_broker\_proxy RUNNING pid 1097, uptime 391 days, 20:30:00  autoscaling\_server RUNNING pid 20149, uptime 116 days, 20:02:52  autoscaling\_server\_proxy RUNNING pid 1088, uptime 391 days, 20:30:00  billing\_delegator RUNNING pid 20230, uptime 116 days, 20:02:45  billing\_delegator\_broker RUNNING pid 20642, uptime 116 days, 20:02:20  billing\_delegator\_broker\_proxy RUNNING pid 1090, uptime 391 days, 20:30:00  billing\_delegator\_proxy RUNNING pid 1098, uptime 391 days, 20:30:00  boss\_collector RUNNING pid 18797, uptime 116 days, 19:16:23  distributed\_proxy RUNNING pid 1091, uptime 391 days, 20:30:00  distributed\_server RUNNING pid 21047, uptime 116 days, 20:02:05  fg\_proxy RUNNING pid 1086, uptime 391 days, 20:30:00  fg\_server RUNNING pid 21104, uptime 116 days, 20:02:03  io\_server RUNNING pid 21208, uptime 116 days, 20:01:56  libvirtd RUNNING pid 1085, uptime 391 days, 20:30:00  push\_server RUNNING pid 21243, uptime 116 days, 20:01:49  scheduler\_broker RUNNING pid 21677, uptime 116 days, 20:01:35  scheduler\_broker\_proxy RUNNING pid 1092, uptime 391 days, 20:30:00  scheduler\_server RUNNING pid 21683, uptime 116 days, 20:01:35  scheduler\_server\_proxy RUNNING pid 1095, uptime 391 days, 20:30:00  topology\_server RUNNING pid 22098, uptime 116 days, 20:01:26  topology\_server\_proxy RUNNING pid 1080, uptime 391 days, 20:30:01 |

Dns

|  |
| --- |
| supervisorctl status  libvirtd RUNNING pid 994, uptime 46 days, 9:30:58  metadata\_server RUNNING pid 993, uptime 46 days, 9:30:58  name\_proxy RUNNING pid 996, uptime 46 days, 9:30:58  name\_server RUNNING pid 995, uptime 46 days, 9:30:58 |

Proxy

|  |
| --- |
| supervisorctl status  cb\_server RUNNING pid 1029, uptime 33 days, 21:37:57  libvirtd RUNNING pid 1028, uptime 33 days, 21:37:57  nf\_server RUNNING pid 1027, uptime 33 days, 21:37:57 |

VG

|  |
| --- |
| supervisorctl status  boss\_daemon RUNNING pid 49207, uptime 293 days, 18:03:21  libvirtd RUNNING pid 20328, uptime 618 days, 17:15:23  vgateway\_proxy RUNNING pid 20327, uptime 618 days, 17:15:23  vgateway\_server RUNNING pid 43343, uptime 116 days, 19:42:54 |

### 时间同步状态

|  |  |
| --- | --- |
| 检查项 | 时间同步状态 |
| 检查方法 | root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh all "date -R" |
| 判断标准 | 正常是所有节点时间相差在3秒以内 |
| 检查结果 | 请在此说明时间同步是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| Thu, 15 Sep 2022 09:40:04 +0800 |

### 文件系统空间

|  |  |
| --- | --- |
| 检查项 | 文件系统空间 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh all "df -h |awk '\$5+0>80{print \$1,\$5,\$NF}'" |
| 判断标准 | 正常是文件系统使用率不超过80% |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| df -h  Filesystem Size Used Avail Use% Mounted on  /dev/sda1 46G 18G 26G 42% /  none 4.0K 0 4.0K 0% /sys/fs/cgroup  udev 189G 12K 189G 1% /dev  tmpfs 38G 1.4M 38G 1% /run  none 5.0M 0 5.0M 0% /run/lock  none 189G 0 189G 0% /run/shm  none 100M 0 100M 0% /run/user  /dev/sdb1 9.9T 171G 9.7T 2% /pitrix |

### 剩余内存

|  |  |
| --- | --- |
| 检查项 | 剩余内存 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh all "free -h;free -m|grep Mem|awk '{print \$3/\$2\*100\"%\"}'" |
| 判断标准 | 正常是内存使用率不超过70% |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| Execing [demo1r02n02] with [free -h;free -m|grep Mem|awk '{print $3/$2\*100"%"}'] ...  total used free shared buff/cache available  Mem: 124G 52G 45G 40M 26G 65G  Swap: 29G 160M 29G  42.4369% |

### 系统负载

|  |  |
| --- | --- |
| 检查项 | 系统负载 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh all "uptime" |
| 判断标准 | 正常是load average不超过40% |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@demo1r01n01:~# uptime  10:28:19 up 207 days, 19:45, 20 users, load average: 1.93, 2.25, 2.59 |

### 系统进程

|  |  |
| --- | --- |
| 检查项 | 进程总数和僵尸进程 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh all "top -b -n 1 | grep zombie" |
| 判断标准 | 正常是zombie为0 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| Tasks: 384 total, 1 running, 383 sleeping, 0 stopped, 0 zombie |

## 每周或按月检查

### NTP状态

|  |  |
| --- | --- |
| 检查项 | NTP进程状态 |
| 检查方法 | root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh all "/etc/init.d/ntp status|grep Active;ntpq -np" |
| 判断标准 | 正常是：  所有节点ntp服务状态正常为running  st列的值小于16 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| Execing [demo1r01n05] with [/etc/init.d/ntp status|grep Active;ntpq -np] ...  Active: active (running) since Mon 2020-12-14 16:35:51 CST; 1 years 10 months ago  remote refid st t when poll reach delay offset jitter  ==============================================================================  \*78.10.1.10 LOCAL(0) 11 u 361 1024 377 0.071 1.478 0.497  +78.10.1.11 78.10.1.10 12 u 248 1024 377 0.072 0.112 0.395 |

### NTP源

|  |  |
| --- | --- |
| 检查项 | NTP进程状态 |
| 检查方法 | root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh all "grep ^server /etc/ntp.conf" |
| 判断标准 | 正常是：  云平台的NTP节点指向客户自己的NTP服务器或者公网NTP服务器  云平台的非NTP节点指向云平台的NTP节点 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| 云平台非NTP节点：  root@demo1r01n03:~# grep ^server /etc/ntp.conf  server 172.31.48.10  server 172.31.48.20  云平台NTP节点：  root@demo1r01n00:~# grep ^server /etc/ntp.conf  server 0.ubuntu.pool.ntp.org  server 1.ubuntu.pool.ntp.org  server 2.ubuntu.pool.ntp.org  server 3.ubuntu.pool.ntp.org  server ntd.ubuntu.com  server 127.127.1.0 # local clock |

### 系统kern日志

|  |  |
| --- | --- |
| 检查项 | 系统kern日志 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh all " tail -n 10 /var/log/kern.log |egrep -i 'error|fail'" |
| 判断标准 | 查看有无近期相关报错 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@demo1r01n03:~# tail -n 1000 /var/log/kern.log |egrep -i 'error|fail'  Jun 30 18:27:59 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jun 30 18:58:03 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jun 30 19:28:04 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jun 30 19:58:29 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jun 30 20:28:27 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jun 30 20:58:49 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jun 30 21:28:48 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jun 30 21:59:04 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jun 30 22:29:09 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jun 30 22:59:18 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jun 30 23:29:36 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jun 30 23:59:58 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 00:29:46 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 01:00:07 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 01:30:07 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 02:00:25 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 02:30:21 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 03:01:10 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 03:30:46 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 04:00:59 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 04:31:07 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 05:01:20 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 05:31:26 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 06:01:43 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 06:32:00 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 07:01:57 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 07:32:11 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 08:02:24 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 08:32:22 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 09:02:50 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 09:32:46 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 10:02:53 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 10:33:12 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 11:03:23 demo1r01n03 kernel: block nbd10: Send disconnect failed -32  Jul 1 11:33:29 demo1r01n03 kernel: block nbd10: Send disconnect failed -32 |

### Netconsole日志

|  |  |
| --- | --- |
| 检查项 | netconsole日志 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh all "grep 'modprobe netconsole' /etc/rc.local.tail ;lsmod|grep netconsole" |
| 判断标准 | 正常是：  每个节点的netconsole模块都是开启的  每个节点的配置文件都存在，且MAC是IAAS管理网络网关的MAC或者netconsole服务器的MAC，且只有一个MAC。 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| Execing [demo1-v2v0] with [grep 'modprobe netconsole' /etc/rc.local.tail ;lsmod|grep netconsole] ...  modprobe netconsole netconsole=@172.31.48.207/eth0,6666@172.31.48.11/d8:67:d9:e9:82:c1  netconsole 20480 0 |

### Inode使用率

|  |  |
| --- | --- |
| 检查项 | Inode使用率 |
| 检查方法 | root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh all "df -ih" |
| 判断标准 | 正常是inode使用率不超过50% |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| Filesystem Inodes IUsed IFree IUse% Mounted on  udev 990K 390 990K 1% /dev  tmpfs 998K 13K 985K 2% /run  /dev/vda2 6.1M 112K 6.0M 2% /  tmpfs 998K 1 998K 1% /dev/shm  tmpfs 998K 4 998K 1% /run/lock  tmpfs 998K 17 998K 1% /sys/fs/cgroup  tmpfs 998K 4 998K 1% /run/user/0 |

### Crontab状态

|  |  |
| --- | --- |
| 检查项 | crontab状态 |
| 检查方法 | root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh all "crontab -l" |
| 判断标准 | 正常是所有节点都存在crontab配置 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| crontab -l  \* \* \* \* \* cd /pitrix/bin; ./alert-agent >> /pitrix/log/alert-agent.log 2>&1 |

### supervisord状态

|  |  |
| --- | --- |
| 检查项 | supervisord状态 |
| 检查方法 | root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh all " ls -ltr /var/log/supervisor/ | tail -n 3" |
| 判断标准 | 查看有无近期生成的supervisord.log |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| -rw------- 1 root root 3869367 Sep 15 00:30 scheduler\_server-stderr---supervisor-Nzfojw.log  -rw------- 1 root root 3685 Sep 15 00:30 billing\_delegator\_broker-stderr---supervisor-07O0iE.log  -rw------- 1 root root 4209 Sep 15 00:37 billing\_delegator-stderr---supervisor-4p0yqf.log |

### vxlan port状态

|  |  |
| --- | --- |
| 检查项 | vxlan prot状态 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh phy "cat /etc/modprobe.d/vxlan-port.conf"  root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh phy "cat /sys/module/vxlan/parameters/udp\_port" |
| 判断标准 | 正常是所有vxlan端口都是4789 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@demo1r01n01:~# cat /etc/modprobe.d/vxlan-port.conf  options vxlan udp\_port=4789 log\_ecn\_error=N  root@demo1r01n01:~# cat /sys/module/vxlan/parameters/udp\_port  4789 |

### 预付费账户检查

|  |  |
| --- | --- |
| 检查项 | 检查平台中是否有预付费账户 |
| 检查方法 | root@qingcloud-firstbox:~# /pitrix/bin/exec\_sql.py -d account -c "select user\_id,user\_name,email,paid\_mode from users where paid\_mode='prepaid';" |
| 判断标准 | 正常是环境中所有账户都为后付费账户 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@qingcloud-firstbox:~# /pitrix/bin/exec\_sql.py -d account -c "select user\_id,user\_name,email,paid\_mode from users where paid\_mode='prepaid';"  +--------------+--------------+------------------------------------+-----------+  | user\_id | user\_name | email | paid\_mode |  +--------------+--------------+------------------------------------+-----------+  | usr-Q0860HRP | 1 | yubing@scdemo.com#1 | prepaid |  | usr-U8C7fQVW | testxgh | xiaoguanghai@bjklb.com#testxgh | prepaid |  | usr-abFDUBqS | yinping | yinping@yunify.com#yinping | prepaid |  | usr-rkhOQoa7 | test1 | lixiaobo@yunify.com#test1 | prepaid |  | usr-ugLCCNpf | 1111 | jiaqianghe@yunify.com#1111 | prepaid |  | usr-yK3KH9p3 | 123 | guodongliu@yunify.com#123 | prepaid |  +--------------+--------------+------------------------------------+-----------+ |

# 巡检项目-物理机节点

以下巡检项目，需要在青云平台上的物理节点执行

## 每天巡检

### 物理机网卡bond状态

|  |  |
| --- | --- |
| 检查项 | 物理机网卡bond状态 |
| 检查方法 | * 查看bond状态及其端口是否都是up状态   root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh phy "ip link |grep bond"   * 查看bond速率是否正常。   root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh phy "ethtool bond0 |grep Speed" |
| 判断标准 | 正常是：  所有节点的bond及其端口都为up状态  所有节点的bond速率都为聚合后的速率，一般为20000Mb/s或50000Mb/s  注:spine-leaf环境不做bond |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@demo1r01n00:~ # ip link |grep bond  6: ens2f0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER\_UP> mtu 1500 qdisc mq master bond0 state UP mode DEFAULT group default qlen 1000  7: ens2f1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER\_UP> mtu 1500 qdisc mq master bond0 state UP mode DEFAULT group default qlen 1000  10: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER\_UP> mtu 1500 qdisc noqueue state UP mode DEFAULT group default qlen 1000  root@demo1r01n00:~ # ethtool bond0 |grep Speed  Speed: 20000Mb/s |

## 每周或按月检查

### 物理机网卡状态

|  |  |
| --- | --- |
| 检查项 | 物理机网卡状态 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh phy "ifconfig bond0|grep error" |
| 判断标准 | 正常是每个节点的errors为0或是比较小的数字，且不能持续增长 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@demo1r01n00:~ # ifconfig bond0|grep error  RX packets:30901558366 errors:22 dropped:27 overruns:0 frame:22  TX packets:36184231593 errors:0 dropped:4 overruns:0 carrier:0 |

### Memblaze Flash卡OP空间检查(可选)

* 仅针对有Memblaze Flash卡的环境。如没有，则跳过。

|  |  |
| --- | --- |
| 检查项 | 针对Memblaze SSD卡OP空间容量 |
| 检查方法 | * 在fb上执行命令确认环境中是否有910/916系列问题固件   /pitrix/upgrade/exec\_nodes.sh phy "nvme list | egrep '21300AH0|213009B0'"   * 如果没有nvme命令，先在firstbox节点上执行以下命令安装   /pitrix/upgrade/update.sh phy nvme pblaze-nvmemgr   * 根据排查结果登录具体物理机，执行命令排查磁盘op空间   nvme smart-log /dev/xxx |grep available\_spare  注: xxx为问题磁盘名称   * 结果举例：   nvme smart-log /dev/nvme0n1|grep available\_spare  available\_spare                     : 78% |
| 判断标准 | 请记录上述百分比  若available\_spare小于80%，申请更换SSD；  若available\_spare大于80%小于98%，申请升级固件；  若available\_spare大于98%，暂不处理，定期巡检。 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

### 磁盘Media Error检查

|  |  |
| --- | --- |
| 检查项 | 磁盘的Media Error、Other Error |
| 检查方法 | 不同服务器使用的raid卡不同，以下列出几种常见raid卡的巡检方法   1. LSI raid卡检查方法  * 确认物理机使用的raid卡型号   root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh phy '/opt/MegaRAID/MegaCli/MegaCli64 -AdpAllInfo -aALL |grep LSI'   * 有如下输出，说明是LSI raid卡，可以使用Megacli命令查询   root@demo1r02n01:~#  /opt/MegaRAID/MegaCli/MegaCli64 -AdpAllInfo -aALL |grep LSI  Product Name : LSI 2208 MegaRAID   * 检查磁盘error信息   root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh phy '/opt/MegaRAID/MegaCli/MegaCli64 -PDList -aALL |grep -i error' |
| 判断标准 | 正常是所有节点error为0 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

正常输出示例：

|  |
| --- |
| Execing [demo2r02n01] with [/opt/MegaRAID/MegaCli/MegaCli64 -PDList -aALL |grep -i error] ...  Media Error Count: 0  Other Error Count: 0  Media Error Count: 0  Other Error Count: 0  Media Error Count: 0  Other Error Count: 0  Media Error Count: 0  Other Error Count: 0  Media Error Count: 0  Other Error Count: 0  Media Error Count: 0  Other Error Count: 0  Media Error Count: 0  Other Error Count: 0  Media Error Count: 0  Other Error Count: 0  Media Error Count: 0  Other Error Count: 0  Media Error Count: 0  Other Error Count: 0  Media Error Count: 0  Other Error Count: 0  Media Error Count: 0  Other Error Count: 0  Media Error Count: 0  Other Error Count: 0  Media Error Count: 0  Other Error Count: 0  Media Error Count: 0  Other Error Count: 0  Media Error Count: 0  Other Error Count: 0 |

# 巡检项目-zoocassa

## 每天检查

### Zookeeper状态

|  |  |
| --- | --- |
| 检查项 | zookeeper状态 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh zoocassa "/opt/zookeeper/bin/zkServer.sh status" |
| 判断标准 | 正常是1个节点为leader,2个节点为follower |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@qcac12u36:~# /opt/zookeeper/bin/zkServer.sh status  JMX enabled by default  Using config: /opt/zookeeper/bin/../conf/zoo.cfg  Mode: follower  ---------------------------------------------------------------  root@qcac13u36:~# /opt/zookeeper/bin/zkServer.sh status  JMX enabled by default  Using config: /opt/zookeeper/bin/../conf/zoo.cfg  Mode: follower  ---------------------------------------------------------------  root@qcbe15u36:~# /opt/zookeeper/bin/zkServer.sh status  JMX enabled by default  Using config: /opt/zookeeper/bin/../conf/zoo.cfg  Mode: leader |

### cassandra状态

|  |  |
| --- | --- |
| 检查项 | cassandra状态 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh zoocassa "/etc/init.d/cassandra status|grep Active"  root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh zoocassa "nodetool status" |
| 判断标准 | 正常是：  cassandra 是running状态  nodetool status三台节点状态均为UN |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@demo1-zoocassa0:~# /etc/init.d/cassandra status|grep Active  Active: active (running) since Tue 2021-06-29 10:18:32 CST; 1 weeks 0 days ago  root@demo1-zoocassa0:~# nodetool status  Datacenter: datacenter1  =======================  Status=Up/Down  |/ State=Normal/Leaving/Joining/Moving  -- Address Load Tokens Owns (effective) Host ID Rack  UN 172.31.48.214 5.7 GB 256 67.9% 1e5692bc-80cd-433d-ac9a-fa1020f441e6 rack1  UN 172.31.48.213 4.59 GB 256 64.2% c5d758d4-9548-4980-ae34-ab271f95bea3 rack1  UN 172.31.48.212 3.82 GB 256 67.9% f371689f-8db9-40d5-82b9-339d59083f97 rack1 |

### memcached状态

|  |  |
| --- | --- |
| 检查项 | memcached状态 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh zoocassa "/etc/init.d/memcached status|grep Active" |
| 判断标准 | 正常是memcached都是running状态 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@demo1-zoocassa0:~# /etc/init.d/memcached status  ● memcached.service - memcached daemon  Loaded: loaded (/lib/systemd/system/memcached.service; enabled; vendor preset: enabled)  Active: active (running) since Tue 2021-06-29 10:18:32 CST; 1 weeks 0 days ago  Main PID: 956 (memcached)  Tasks: 6  Memory: 7.9M  CPU: 30.872s  CGroup: /system.slice/memcached.service  └─956 /usr/bin/memcached -m 64 -p 11211 -u memcache  Jun 29 10:18:32 demo1-zoocassa0 systemd[1]: Started memcached daemon. |

### redis状态（仅在global环境执行）

|  |  |
| --- | --- |
| 检查项 | redis状态 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh zoocassa "ps -ef | grep redis |grep -v grep" |
| 判断标准 | 正常是3个节点上,redis及sentinel进程都存在 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| Execing [demo1-zoocassa0] with [ps -ef | grep redis |grep -v grep] ...  root 1333 1 0 Jul28 ? 03:13:44 /usr/local/bin/redis-server 0.0.0.0:6379  root 1343 1 0 Jul28 ? 03:59:17 /usr/local/bin/redis-server \*:26379 [sentinel] |

# 巡检项目- webservice

## 每周或按月检查

### apache状态

|  |  |
| --- | --- |
| 检查项 | apache状态 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh webservice "service apache2 status |grep Active" |
| 判断标准 | 正常是webservice的apache服务状态为running |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh webservice "service apache2 status |grep Active"  ------------------------------------------------  TARGET NODES:  demo1-webservice0,demo1-webservice1,demo2-webservice0,demo2-webservice1  ------------------------------------------------  TARGET COMMAND:  service apache2 status |grep Active  ------------------------------------------------  Are you sure to exec the command on the above nodes [y/N]?y  Execing [demo1-webservice0] with [service apache2 status |grep Active] ...  Active: active (running) since Thu 2022-09-15 12:27:01 CST; 1h 56min ago  Execing [demo1-webservice1] with [service apache2 status |grep Active] ...  Active: active (running) since Wed 2022-09-14 12:09:07 CST; 1 day 2h ago  Execing [demo2-webservice1] with [service apache2 status |grep Active] ...  Active: active (running) since Wed 2022-09-14 11:34:46 CST; 1 day 2h ago  Execing [demo2-webservice0] with [service apache2 status |grep Active] ...  Active: active (running) since Wed 2022-09-14 11:34:46 CST; 1 day 2h ago  Done. |

### nginx状态

|  |  |
| --- | --- |
| 检查项 | nginx状态 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh webservice "service nginx status |grep Active" |
| 判断标准 | 正常是webservice的nginx服务状态为running |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh webservice "service nginx status |grep Active"  ------------------------------------------------  TARGET NODES:  demo1-webservice0,demo1-webservice1,demo2-webservice0,demo2-webservice1  ------------------------------------------------  TARGET COMMAND:  service nginx status |grep Active  ------------------------------------------------  Are you sure to exec the command on the above nodes [y/N]?y  Execing [demo1-webservice1] with [service nginx status |grep Active] ...  Active: active (running) since Wed 2022-06-01 15:51:37 CST; 3 months 14 days ago  Execing [demo1-webservice0] with [service nginx status |grep Active] ...  Active: active (running) since Thu 2022-07-07 14:48:23 CST; 2 months 9 days ago  Execing [demo2-webservice0] with [service nginx status |grep Active] ...  Active: active (running) since Mon 2022-06-13 13:35:13 CST; 3 months 2 days ago  Execing [demo2-webservice1] with [service nginx status |grep Active] ...  Active: active (running) since Wed 2022-05-18 13:06:48 CST; 3 months 28 days ago  Done. |

# 巡检项目-pgserver和pgpool

以下巡检项目，需要在青云平台上的2台pgserveer和2台pgpool节点执行

## 每周或按月检查

### pgserver状态

|  |  |
| --- | --- |
| 检查项 | pgserver状态 |
| 检查方法 | 任选一种方法:  root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh pgserver '/etc/init.d/postgresql status|grep Active'  root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh pgserver 'service postgresql status|grep Active'  root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh pgserver 'systemctl status postgresql.service|grep Active' |
| 判断标准 | 正常是：  pg9的pgserver状态为online或者online,recovery  pg10的pgserver状态为running |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| PG9输出示例：  root@demo1r01n01:~#service postgresql status（主库）  9.3/main (port 5432): online  root@demo1r01n02:~#service postgresql status（备库）  9.3/main (port 5432): online,recovery  PG10输出示例：  root@demo1r01n01:~# /etc/init.d/postgresql status  ● postgresql.service - PostgreSQL database server  Loaded: loaded (/etc/systemd/system/postgresql.service; enabled; vendor preset: enabled)  Active: active (running) since Tue 2021-05-11 14:47:38 CST; 1 months 25 days ago  Process: 36462 ExecStop=/opt/postgresql-10.4/bin/pg\_ctl -s -D /opt/postgresql-10.4/data/ stop -m fast (code=exited, status=0/SUCCESS)  Process: 36471 ExecStart=/opt/postgresql-10.4/bin/pg\_ctl -s -D /opt/postgresql-10.4/data/ start -w -t 120 (code=exited, status=0/SUCCESS)  Main PID: 36474 (postgres)  CGroup: /system.slice/postgresql.service  ├─ 1524 postgres: postgres zone [local] idle  ├─36474 /pitrix/postgresql-10.4/bin/postgres -D /opt/postgresql-10.4/data  ├─36543 postgres: logger process  ├─36544 postgres: startup process recovering 0000000900000058000000F1  ├─36548 postgres: checkpointer process  ├─36549 postgres: writer process  ├─36550 postgres: stats collector process  └─36551 postgres: wal receiver process streaming 58/F1D127C8 |

### pgserver连接数

|  |  |
| --- | --- |
| 检查项 | pgserver连接数检查 |
| 检查方法 | 1）获取pgserver密码  root@demo1-webservice0:~#  cat /pitrix/ks/webservice-base/conf/.pgserver\_passwd  输出如下：  1IOgcNsp  2）在webservice节点执行，查看pgpool与pgserver的连接数  root@demo1-webservice0:~# PGPASSWORD=1IOgcNsp（前面的pgserver密码） psql -h demo1-pgpool0（替换为实际的pgpool主机名） -d zone -U yunify -p 5432 -c "select count(1) from pg\_stat\_activity;"  输出如下：  count  -------  474  3）查看pgserver设置的最大连接数  root@demo1-webservice0:~# PGPASSWORD=1IOgcNsp psql -h demo1-pgpool0 -d zone -U yunify -p 5432 -c "show max\_connections;"  输出如下：  max\_connections  -----------------  2048 |
| 判断标准 | 正常是第2步的数值不能接近第3步的数值 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

### pgpool状态

|  |  |
| --- | --- |
| 检查项 | pgpool进程状态 |
| 检查方法 | 1）老版本pgpool检查pgpool状态  root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh pgpool 'ps -ef |grep pgpool'  2）新版本pgpool以下方法任选一种  root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh pgpool '/etc/init.d/pgpool2 status |grep Active'  root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh pgpool 'service pgpool2 status|grep Active'  root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh pgpool 'systemctl status pgpool2.service|grep Active'  3）验证是否可以登录  登陆webservice节点操作  root@demo1-webservice0:~# psql -h demo1r01n01 -d zone -U yunify  ## demo1r01n01是pgpool节点。出现输入密码提示，说明可以正常登录 |
| 判断标准 | 正常是：  老版本pgpool出现pgpool: wait for connection request进程  新版本pgpool状态为running  且第三步登陆出现密码提示 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| 老版本pgpool输出示例：  root@pgpool0:~# ps -ef |grep pgpool  postgres 490 1461 0 09:45 ? 00:00:00 pgpool: wait for connection request  postgres 576 1461 0 08:17 ? 00:00:00 pgpool: wait for connection request  postgres 621 1461 0 11:13 ? 00:00:00 pgpool: wait for connection request  postgres 815 1461 0 06:50 ? 00:00:00 pgpool: wait for connection request  postgres 900 1461 0 09:46 ? 00:00:00 pgpool: wait for connection request  postgres 1077 1461 0 09:46 ? 00:00:00 pgpool: wait for connection request  postgres 1184 1461 0 11:15 ? 00:00:00 pgpool: wait for connection request  postgres 1185 1461 0 11:15 ? 00:00:00 pgpool: wait for connection request  postgres 1186 1461 0 11:15 ? 00:00:00 pgpool: wait for connection request  postgres 1187 1461 0 11:15 ? 00:00:00 pgpool: wait for connection request  postgres 1188 1461 0 11:15 ? 00:00:00 pgpool: wait for connection request  postgres 1189 1461 0 11:15 ? 00:00:00 pgpool: wait for connection request  postgres 1190 1461 0 11:15 ? 00:00:00 pgpool: wait for connection request  postgres 1191 1461 0 11:15 ? 00:00:00 pgpool: wait for connection request  postgres 1192 1461 0 11:15 ? 00:00:00 pgpool: wait for connection request  postgres 1193 1461 0 11:15 ? 00:00:00 pgpool: wait for connection request  postgres 1194 1461 0 11:15 ? 00:00:00 pgpool: wait for connection request  postgres 1195 1461 0 11:15 ? 00:00:00 pgpool: wait for connection request  postgres 1196 1461 0 11:15 ? 00:00:00 pgpool: wait for connection request  postgres 1314 1461 0 11:15 ? 00:00:00 pgpool: wait for connection request  postgres 1337 1461 0 11:15 ? 00:00:00 pgpool: wait for connection request  postgres 1341 1461 0 11:15 ? 00:00:00 pgpool: wait for connection request  postgres 1352 1461 0 03:55 ? 00:00:00 pgpool: wait for connection request  新版本pgpool输出示例：  root@demo1r01n01:~# service pgpool2 status  ● pgpool2.service - pgpool-II  Loaded: loaded (/lib/systemd/system/pgpool2.service; enabled; vendor preset: enabled)  Active: active (running) since Thu 2021-06-24 17:12:16 CST; 1 weeks 4 days ago  Docs: man:pgpool(8)  Main PID: 6631 (pgpool)  CGroup: /system.slice/pgpool2.service  ├─6631 /usr/sbin/pgpool -n  ├─6633 pgpool: wait for connection reques  ├─6634 pgpool: wait for connection reques  ├─6635 pgpool: wait for connection reques  ├─6636 pgpool: wait for connection reques  ├─6637 pgpool: wait for connection reques  ├─6638 pgpool: wait for connection reques  ├─6639 pgpool: wait for connection reques  ├─6640 pgpool: wait for connection reques  ├─6641 pgpool: wait for connection reques  ├─6642 pgpool: wait for connection reques  ├─6643 pgpool: wait for connection reques  ├─6644 pgpool: wait for connection reques  ├─6645 pgpool: wait for connection reques  ├─6646 pgpool: wait for connection reques  ├─6647 pgpool: wait for connection reques  ├─6648 pgpool: wait for connection reques  ├─6649 pgpool: wait for connection reques  ├─6650 pgpool: wait for connection reques  ├─6651 pgpool: wait for connection reques  pgpool登录输出示例：  root@demo1-webservice0:~# psql -h demo1r01n01 -d zone -U yunify  Password for user yunify: |

### pgpool连接数

|  |  |
| --- | --- |
| 检查项 | pgpool连接数 |
| 检查方法 | 1）在webservice节点执行，获取当前pgpool的连接数  root@demo1-webservice0:~# PGPASSWORD=1IOgcNsp psql -h demo1-pgpool0 -d zone -U yunify -p 5432 -c "SHOW pool\_pools;" |awk -F '|' '{print $12}'|grep 1 |wc -l  输出如下:  18  ## 解释如下  18是pgpool当前的连接数据  1IOgcNsp是pg密码，按照8.1.2方法获取密码  demo1-pgpool0请替换成实际环境种的pgpool主机名  2）在webservice节点执行，获取pgpool允许的最大并发连接数  root@demo1-webservice0:~# PGPASSWORD=1IOgcNsp psql -h demo1-pgpool0 -d zone -U yunify -p 5432 -c "SHOW pool\_status;" |grep num\_init\_children |awk -F '|' '{print $2}'  输出如下：  400 |
| 判断标准 | 正常是第1步的数值不能接近第2步的数值 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

# 巡检项目-Proxy和DNS

以下巡检项目，需要在青云平台上的proxy和dns节点执行

## 每周或按月检查

### Haproxy状态

|  |  |
| --- | --- |
| 检查项 | haproxy状态 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh proxy 'service haproxy status|grep Active' |
| 判断标准 | 正常是haproxy都是running状态 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh proxy 'service haproxy status|grep Active'  ------------------------------------------------  TARGET NODES:  demo1-proxy0,demo1-proxy1,demo2-proxy0,demo2-proxy1  ------------------------------------------------  TARGET COMMAND:  service haproxy status|grep Active  ------------------------------------------------  Are you sure to exec the command on the above nodes [y/N]?y  Execing [demo2-proxy1] with [service haproxy status|grep Active] ...  Active: active (running) since Wed 2022-05-18 13:06:45 CST; 3 months 29 days ago  Execing [demo1-proxy1] with [service haproxy status|grep Active] ...  Active: active (running) since Wed 2022-08-17 23:13:12 CST; 4 weeks 1 days ago  Execing [demo2-proxy0] with [service haproxy status|grep Active] ...  Active: active (running) since Thu 2022-09-08 23:51:41 CST; 1 weeks 0 days ago  Execing [demo1-proxy0] with [service haproxy status|grep Active] ...  Active: active (running) since Wed 2022-07-27 14:29:29 CST; 1 months 20 days ago  Done. |

### Nginx状态

|  |  |
| --- | --- |
| 检查项 | nginx状态 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh proxy 'service nginx status|grep Active' |
| 判断标准 | 正常是nginx都是running状态 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh proxy 'service nginx status|grep Active'  ------------------------------------------------  TARGET NODES:  demo1-proxy0,demo1-proxy1,demo2-proxy0,demo2-proxy1  ------------------------------------------------  TARGET COMMAND:  service nginx status|grep Active  ------------------------------------------------  Are you sure to exec the command on the above nodes [y/N]?y  Execing [demo2-proxy1] with [service nginx status|grep Active] ...  Active: active (running) since Tue 2022-05-31 11:48:36 CST; 3 months 16 days ago  Execing [demo1-proxy1] with [service nginx status|grep Active] ...  Active: active (running) since Wed 2022-08-17 23:13:13 CST; 4 weeks 1 days ago  Execing [demo1-proxy0] with [service nginx status|grep Active] ...  Active: active (running) since Wed 2022-07-27 14:29:30 CST; 1 months 20 days ago  Execing [demo2-proxy0] with [service nginx status|grep Active] ...  Active: active (running) since Thu 2022-09-08 23:51:41 CST; 1 weeks 0 days ago  Done. |

### Bind9状态

|  |  |
| --- | --- |
| 检查项 | bind9状态 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh dns 'service bind9 status|grep Active' |
| 判断标准 | 正常是bind9都是running状态 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh dns 'service bind9 status|grep Active'  ------------------------------------------------  TARGET NODES:  demo1-dnsmaster0,demo1-dnsmaster1,demo2-dnsmaster0,demo2-dnsmaster1  ------------------------------------------------  TARGET COMMAND:  service bind9 status|grep Active  ------------------------------------------------  Are you sure to exec the command on the above nodes [y/N]?y  Execing [demo1-dnsmaster0] with [service bind9 status|grep Active] ...  Active: active (running) since Tue 2022-06-21 14:03:02 CST; 2 months 25 days ago  Execing [demo2-dnsmaster0] with [service bind9 status|grep Active] ...  Active: active (running) since Tue 2022-02-22 13:51:26 CST; 6 months 23 days ago  Execing [demo2-dnsmaster1] with [service bind9 status|grep Active] ...  Active: active (running) since Wed 2022-05-18 13:06:43 CST; 3 months 29 days ago  Execing [demo1-dnsmaster1] with [service bind9 status|grep Active] ...  Active: active (running) since Sat 2022-08-27 16:29:44 CST; 2 weeks 5 days ago  Done. |

# 巡检项目-计算节点

以下巡检项目，需要在青云平台上的计算节点执行

## 每天检查

### DRBD状态

* 针对sds1.0存储类型的计算节点。如果没有sds1.0，则跳过

|  |  |
| --- | --- |
| 检查项 | drbd状态 |
| 检查方法 | * 确定环境中是否存在sds1.0存储   root@qingcloud-firstbox:~# cat /pitrix/conf/nodes/hyper-pair  如有输出，则存在sds1.0   * 查看drbd状态   root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh hyper-pair "drbd-overview" |
| 判断标准 | 正常是所有hyper的drbd都为Connected UpToDate/UpToDate状态。如下表所示。 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

正常状态示例：

|  |
| --- |
| Execing [demo1r02n02] with [drbd-overview] ...  0:demo1r02n02/0 Connected Primary/Secondary UpToDate/UpToDate /pitrix/data/container ext4 4.1T 320G 3.8T 8%  1:demo1r01n02/0 Connected Secondary/Primary UpToDate/UpToDate |

### repl状态

* 针对sds2.0存储类型的计算节点。如果没有sds2.0，则跳过

|  |  |
| --- | --- |
| 检查项 | repl状态 |
| 检查方法 | * 确定环境中是否存在sds2.0存储   root@qingcloud-firstbox:~# cat /pitrix/conf/nodes/hyper-repl   * 查看repl状态   root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh hyper-repl "repl-overview" |
| 判断标准 | 正常是所有hyper的repl-overview状态都为下表所示 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

正常状态示例:

|  |
| --- |
| root@demo2r01n03:~# repl-overview  i-71eq8m2t V /pitrix/data/container/i-71eq8m2t 0.04G 2.6T 0.0%  demo2r01n04 local 0 UpToDate  i-8u7foe8v VR /pitrix/data/container/i-8u7foe8v 4.39G 2.6T 0.16%  demo2r01n04 local 0 UpToDate  i-9l35f8k0 VR /pitrix/data/container/i-9l35f8k0 0.04G 2.6T 0.0%  demo2r02n03 local 0 UpToDate  i-alybiatw VR /pitrix/data/container/i-alybiatw 3.84G 2.6T 0.14%  demo2r02n03 local 0 UpToDate  i-bl1chdpt VR /pitrix/data/container/i-bl1chdpt 0.07G 2.6T 0.0%  demo2r02n03 local 0 UpToDate  i-gxqyxt8v VR /pitrix/data/container/i-gxqyxt8v 5.04G 2.6T 0.19%  demo2r01n04 local 0 UpToDate  i-p3bzrpc1 VR /pitrix/data/container/i-p3bzrpc1 84.29G 2.6T 3.06%  demo2r02n03 local 0 UpToDate |

### zpool状态

* 针对sds2.0存储类型的计算节点,如果没有sds2.0，请跳过

|  |  |
| --- | --- |
| 检查项 | zpool状态 |
| 检查方法 | * 确定环境中是否存在sds2.0存储   root@qingcloud-firstbox:~# cat /pitrix/conf/nodes/hyper-repl   * 查看zppol状态   root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh hyper-repl "zpool status" |
| 判断标准 | 正常是zpool state都为ONLINE，且errors值为No known data errors |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@demo2r01n03:~# zpool status  pool: vpool  state: ONLINE  scan: none requested  config:  NAME STATE READ WRITE CKSUM  vpool ONLINE 0 0 0  dfa ONLINE 0 0 0  errors: No known data errors |

## 每周或按月检查

### swap状态

* 仅针对sds2.0节点，如果没有sds2.0存储，则跳过

|  |  |
| --- | --- |
| 检查项 | swap状态 |
| 检查方法 | * 确认环境中是否存在sds2.0存储   root@qingcloud-firstbox:~# cat /pitrix/conf/nodes/hyper-repl   * 查看swap状态   root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh hyper-repl "swapon -s" |
| 判断标准 | 正常是所有sds2.0节点的swap都没有占用vpool空间 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

正常状态示例：

|  |
| --- |
| root@demo2r01n03:~# swapon -s  Filename Type Size Used Priority  /dev/sda2 partition 31250428 0 -2 |

### IO性能检查

|  |  |
| --- | --- |
| 检查项 | 检查计算节点IO监控数据是否正常 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh hyper "iostat -x 2 -c 3|grep -v nb" |
| 判断标准 | 查看每台节点是否存在读写量很低，但io util使用率非常高的情况，如果存在需要具体分析，需要联系硬件厂商分析。  比如io读写在10MB左右，但util使用率90%以上。 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@demo2r01n03:~# iostat -x 2 -c 3 |grep -v nb  Linux 4.15.0-58-generic (demo2r01n03) 07/05/2021 \_x86\_64\_ (8 CPU)  avg-cpu: %user %nice %system %iowait %steal %idle  7.57 0.00 4.03 0.03 0.00 88.37  Device: rrqm/s wrqm/s r/s w/s rkB/s wkB/s avgrq-sz avgqu-sz await r\_await w\_await svctm %util  sda 0.00 0.66 0.00 2.51 0.02 23.20 18.50 0.09 34.69 4.50 34.71 2.88 0.72  dfa 0.00 0.00 0.85 101.95 12.24 898.53 17.72 0.00 0.04 0.17 0.04 0.02 0.17  qbd0 0.00 0.00 0.00 0.00 0.00 0.00 9.02 0.00 3.41 3.41 0.00 3.41 0.00  avg-cpu: %user %nice %system %iowait %steal %idle  40.24 0.00 20.99 0.06 0.00 38.70  Device: rrqm/s wrqm/s r/s w/s rkB/s wkB/s avgrq-sz avgqu-sz await r\_await w\_await svctm %util  sda 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00  dfa 0.00 0.00 0.00 37.50 0.00 458.00 24.43 0.00 0.00 0.00 0.00 0.00 0.00  qbd0 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00  avg-cpu: %user %nice %system %iowait %steal %idle  36.36 0.00 16.34 0.00 0.00 47.30  Device: rrqm/s wrqm/s r/s w/s rkB/s wkB/s avgrq-sz avgqu-sz await r\_await w\_await svctm %util  sda 0.00 1.00 0.00 4.50 0.00 66.25 29.44 0.05 11.56 0.00 11.56 4.00 1.80  dfa 0.00 0.00 1.50 412.50 14.00 3174.00 15.40 0.02 0.05 0.00 0.05 0.01 0.40  qbd0 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 |

### 单节点VM数量

|  |  |
| --- | --- |
| 检查项 | 单节点vm数量 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh hyper "virsh list|grep i-|wc -l" |
| 判断标准 | Log记录每台hyper的VM数量 |
| 检查结果 | 描述单节点VM数量分布情况。如最多一台hyper的VM数量。 |
|  |

### IP forward状态

|  |  |
| --- | --- |
| 检查项 | ip forward状态 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh phy "cat /proc/sys/net/ipv4/ip\_forward" |
| 判断标准 | 正常是ip\_forward的配置都为1 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

正常状态示例：

|  |
| --- |
| root@demo2r01n03:~# cat /proc/sys/net/ipv4/ip\_forward  1 |

### 透明大页

|  |  |
| --- | --- |
| 检查项 | 透明大页使用情况 |
| 检查方法 | root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh all "cat /sys/kernel/mm/transparent\_hugepage/enabled" |
| 判断标准 | 正常是都为always [madvise] never |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

正常状态示例：

|  |
| --- |
| Execing [demo1a-eipctl0] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-boss0] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-boss1] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-dnsmaster0] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-dnsmaster1] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-eipctl1] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-eipctl2] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-eipctl3] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-influxdb0] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-pgpool0] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-pgpool1] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1ar01n00] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-proxy1] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1ar01n02] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1ar01n01] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-proxy0] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1ar01n03] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1ar01n04] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1ar01n06] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1ar01n05] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1ar01n07] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1ar01n08] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1ar02n00] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1ar01n09] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1ar02n01] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-warehouse0] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-webservice0] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-warehouse1] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-zoocassa1] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-zoocassa2] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-v2v0] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-webservice1] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Execing [demo1a-zoocassa0] with [cat /sys/kernel/mm/transparent\_hugepage/enabled] ...  always [madvise] never  Done. |

### 维护工具

|  |  |
| --- | --- |
| 检查项 | 是否安装维护工具 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh phy "ls /usr/bin/ |grep guest" |
| 判断标准 | 正常是：  hyper节点有  guest-ftp guest-sh safe-guest-ftp safe-guest-sh  非hyper节点有  safe-guest-ftp safe-guest-sh |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

正常状态示例：

|  |
| --- |
| Execing [demo1ar01n02] with [ls /usr/bin/ |grep guest] ...  guest-ftp  guest-sh  safe-guest-ftp  safe-guest-sh  Execing [demo1ar01n00] with [ls /usr/bin/ |grep guest] ...  safe-guest-ftp  safe-guest-sh |

# 巡检项目-IAAS平台管理

以下巡检项目，需要在青云平台上的console页面和BOSS 上查看。

## 每天巡检

### console控制台

|  |  |
| --- | --- |
| 检查项 | console控制台 |
| 检查方法 | 点击console页面，看是否有响应缓慢的情况。  打开vnc等，验证前端功能是否正常，页面是否有报错情况。 |
| 判断标准 | 正常是console页面响应及时 VNC打开正常 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

## 每周或按月检查

### 计算节点状态

|  |  |
| --- | --- |
| 检查项 | 计算节点状态 |
| 检查方法 | 登录BOSS查看计算节点状态。 |
| 判断标准 | 正常是所有计算节点应为活跃状态，或少量节点为standby状态（计划内）。 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

### 资源使用率

|  |  |
| --- | --- |
| 检查项 | 资源使用率 |
| 检查方法 | 查看BOSS中计算节点页面中的数据，关注虚拟、物理资源使用率，以及性能状态。  资源使用率判断方法请参考https://ins.qingcloud.com/d/128-iaas/3 |
| 判断标准 | 如果资源使用率普遍较高，即提醒客户扩容。 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

### 操作日志

|  |  |
| --- | --- |
| 检查项 | 操作日志 |
| 检查方法 | 查看BOSS导航栏里“操作日志”查看近期失败的任务 |
| 判断标准 | 正常应该是没有近期失败的任务，或者是计划内的任务失败 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

### 评估计算资源用量

|  |  |
| --- | --- |
| 检查项 | 安置组容量预测 |
| 检查方法 | 查看BOSS导航栏🡪总览，安置组容量预测。 |
| 判断标准 | 预估容量低于6个月，即提醒客户扩容。 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

### 查看基础网络地址余量

|  |  |
| --- | --- |
| 检查项 | 安置组容量预测 |
| 检查方法 | 查看BOSS导航栏🡪网络🡪基础网络，查看每个基础网络地址剩余量。 |
| 判断标准 | 正常是每个段ip地址空闲数不低于20个 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

# 巡检项目-X86 VG节点

* 本章巡检项目，适用有X86VG的环境。若没有X86VG，则跳过。

## 每周或按月检查

### EIP数量

|  |  |
| --- | --- |
| 检查项 | EIP数量 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh vg "/pitrix/bin/vgateway-show -L|grep sh|awk -F' |:' '{print $3}'|uniq|wc -l"  root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh vg "/pitrix/bin/vgateway-show -e|grep dst|wc -l" |
| 判断标准 | 记录每台vg的eip数量 |
| 检查结果 | 记录每台vg的eip数量 |
|  |

实际输出示例：

|  |
| --- |
| root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh vg "/pitrix/bin/vgateway-show -L|grep sh|awk -F' |:' '{print $3}'|uniq|wc -l"  ------------------------------------------------  TARGET NODES:  demo1r01n00,demo1r02n00  ------------------------------------------------  TARGET COMMAND:  /pitrix/bin/vgateway-show -L|grep sh|awk -F' |:' '{print }'|uniq|wc -l  ------------------------------------------------  Are you sure to exec the command on the above nodes [y/N]?y  Execing [demo1r01n00] with [/pitrix/bin/vgateway-show -L|grep sh|awk -F' |:' '{print }'|uniq|wc -l] ...  0  Execing [demo1r02n00] with [/pitrix/bin/vgateway-show -L|grep sh|awk -F' |:' '{print }'|uniq|wc -l] ...  12  Done.  root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh vg "/pitrix/bin/vgateway-show -e|grep dst|wc -l"  ------------------------------------------------  TARGET NODES:  demo1r01n00,demo1r02n00  ------------------------------------------------  TARGET COMMAND:  /pitrix/bin/vgateway-show -e|grep dst|wc -l  ------------------------------------------------  Are you sure to exec the command on the above nodes [y/N]?y  Execing [demo1r01n00] with [/pitrix/bin/vgateway-show -e|grep dst|wc -l] ...  3  Execing [demo1r02n00] with [/pitrix/bin/vgateway-show -e|grep dst|wc -l] ...  13  Done. |

### CPU负载状态

|  |  |
| --- | --- |
| 检查项 | CPU负载状态 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh vg 'top -b -n 1 |grep %Cpu' |
| 判断标准 | 正常情况请参考实际输出示例。  关于cpu性能参数的解释:   * us （user time）列显示了用户方式下所花费 CPU 时间的百分比。us的值比较高时，说明用户进程消耗的cpu时间多，但是如果长期大于50%，需要考虑优化用户的程序。 * sy（system time） 列显示了内核进程所花费的cpu时间的百分比。这里us + sy的参考值为80%，如果us+sy 大于 80%说明可能存在CPU不足。 * ni（nice time）表示系统调整进程优先级所花费的时间 * id （idle time）列显示了cpu处在空闲状态的时间百分比。数值越大，说明CPU越空闲。 * wa （waiting time）列显示了IO等待所占用的CPU时间的百分比。这里wa的参考值为30%，如果wa超过30%，说明IO等待严重，这可能是磁盘大量随机访问造成的，也可能磁盘或者磁盘访问控制器的带宽瓶颈造成的(主要是块操作)。 * hi(hard irq time) 系统处理硬中断所花费的时间。 * si(softirq time) 系统处理软件中断所花费的时间。 * st(steal time) 被强制等待虚拟CPU的时间，此时hypervisor在为另一个虚拟处理器服务。比如st 70%表示该guest需要、但未被host满足的资源是70%。它会发生在当需要的资源分配给其它guest时。 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh vg 'top -b -n 1 |grep %Cpu'  ------------------------------------------------  TARGET NODES:  demo1r01n00,demo1r02n00  ------------------------------------------------  TARGET COMMAND:  top -b -n 1 |grep %Cpu  ------------------------------------------------  Are you sure to exec the command on the above nodes [y/N]?y  Execing [demo1r01n00] with [top -b -n 1 |grep %Cpu] ...  %Cpu(s): 0.6 us, 0.3 sy, 0.0 ni, 99.1 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st  Execing [demo1r02n00] with [top -b -n 1 |grep %Cpu] ...  %Cpu(s): 1.1 us, 0.6 sy, 0.0 ni, 98.2 id, 0.1 wa, 0.0 hi, 0.0 si, 0.0 st  Done. |

### IP forward状态

|  |  |
| --- | --- |
| 检查项 | ip forward状态 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh vg 'cat /proc/sys/net/ipv4/ip\_forward' |
| 判断标准 | 正常是每台VG的ip\_forward都为1 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh vg 'cat /proc/sys/net/ipv4/ip\_forward'  ------------------------------------------------  TARGET NODES:  demo1r01n00,demo1r02n00  ------------------------------------------------  TARGET COMMAND:  cat /proc/sys/net/ipv4/ip\_forward  ------------------------------------------------  Are you sure to exec the command on the above nodes [y/N]?y  Execing [demo1r02n00] with [cat /proc/sys/net/ipv4/ip\_forward] ...  1  Execing [demo1r01n00] with [cat /proc/sys/net/ipv4/ip\_forward] ...  1  Done. |

### EIP流量排序

|  |  |
| --- | --- |
| 检查项 | EIP流量排序 |
| 检查方法 | root@qingcloud-firstbox:~#  cat /pitrix/conf/nodes/vg | sed -e 's/nodes=(//g' -e 's/);//g'|tr " " "\n" |xargs -I {} ssh demo1-webservice0 /pitrix/lib/pitrix-scripts/eip/sort\_traffic --hosts {} -k tp -c 1 -i 10 -L 5  ##以上为一条命令，demo1-webservice0请替换为实际webservice节点名 |
| 判断标准 | 记录流量最大的5个EIP地址 |
| 检查结果 | 记录流量最大的5个EIP地址 |
|  |

实际输出示例：

|  |
| --- |
|  |

# 巡检项目-物理主机（BM）

QingCloud 物理主机 (Bare Metal)，提供高性能、资源独享、安全隔离的专属物理主机，同时具备云资源的敏捷性、灵活性与高可用。满足各类应用对高性能及稳定性的需求，同时提供完整的设备管理权限及运维服务。

* 以下巡检项目，适用存在BM的环境，若无BM，则跳过。

## 每周或按月检查

### BM服务

|  |  |
| --- | --- |
| 检查项 | BM服务状态（bm\_server和swctl\_server） |
| 检查方法 | * 确定环境中是否存在BM   root@qingcloud-firstbox:~# cat /pitrix/conf/nodes/bm   * 登陆firstbox 执行supervisorctl status命令   root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh bm 'supervisorctl status' |
| 判断标准 | 正常是每个服务都是running状态 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

正常状态示例：

|  |
| --- |
|  |

### TFTP服务

|  |  |
| --- | --- |
| 检查项 | tftp服务状态 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh bm 'service tftpd-hpa status' |
| 判断标准 | 正常是running状态 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
|  |

### boot\_image检查

|  |  |
| --- | --- |
| 检查项 | 检查boot\_image文件是否存在 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh bm 'ls -al /pitrix/data/image' |
| 判断标准 | 正常是/pitrix/data/image/下存在bootcore--bootcoree六个文件。后续可能还有增加，以ftp://bm目录下的bootcore文件为准 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
|  |

### gobgp邻居关系检查

* VXLAN模式BM才需要检查gobgp关系

|  |  |
| --- | --- |
| 检查项 | 检查bm0/1与Border是否正常建立邻居关系 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh bm 'gobgp neighbor' |
| 判断标准 | 正常状态是Establish。如下表所示。 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| root@demo1-bm0:/pitrix/data/image# gobgp n  Peer AS Up/Down State |#Received Accepted  169.254.1.5 65534 278d 21:30:31 Establ | 32 32 |

### BM日志

|  |  |
| --- | --- |
| 检查项 | 检查bm0/1管理虚拟机相关日志 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh bm " tail -n 50 /var/log/kern.log |egrep -i 'error|fail'"  root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh bm 'tail -n 50 /pitrix/log/\*.wf |egrep -i "error|CRITICAL"' |
| 判断标准 | 正常是近期无持续error报错 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
|  |

# 巡检项目-Border

* 环境中如无Border，则跳过此项检查

如果将虚拟网络和物理网络做对比，Vxnet是一个二层网络，VPC是用一个虚拟路由器将多个二层网络连起来的三层网络。Border将多个三层网络连接起来，边界路由器作为每个三层网络的边界，做路由信息的同步和数据包转发的枢纽。

Border可以连接三种网络：

1.VPC网络

2.BM网络

3.SD-WAN网关

## 每周或按月检查

### Border版本

|  |  |
| --- | --- |
| 检查项 | 检查Border版本 |
| 检查方法 | 1. 登陆border交换机   root@demo1-bm0:~# gobgp n  Peer AS Up/Down State |#Received Accepted  169.254.1.5 65534 274d 21:24:41 Establ | 32 32  root@demo1-bm0:~# ssh yop@169.254.1.5(上文border交换机的地址)  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* Copyright (c) 2004-2019 New H3C Technologies Co., Ltd. All rights reserved.\*  \* Without the owner's prior written consent, \*  \* no decompiling or reverse-engineering shall be allowed. \*  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  <DEMO-H3C 6800>   1. 在Border执行 display version命令   <DEMO-H3C 6800>display version  H3C Comware Software, Version 7.1.070, Release 2702  Copyright (c) 2004-2019 New H3C Technologies Co., Ltd. All rights reserved.  H3C S6800-54QF uptime is 39 weeks, 1 day, 21 hours, 39 minutes  Last reboot reason : User reboot |
| 判断标准 | 正常是版本等于或高于Version7.1.070 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| <demo1-H3C 6800>display version  H3C Comware Software, Version 7.1.070, Release 2702  Copyright (c) 2004-2019 New H3C Technologies Co., Ltd. All rights reserved.  H3C S6800-54QF uptime is 39 weeks, 5 days, 21 hours, 55 minutes  Last reboot reason : User reboot  Boot image: flash:/s6800-cmw710-boot-r2702.bin  Boot image version: 7.1.070, Release 2702  Compiled May 20 2019 11:00:00  System image: flash:/s6800-cmw710-system-r2702.bin  System image version: 7.1.070, Release 2702  Compiled May 20 2019 11:00:00  Patch image(s) list:  flash:/S6800-CMW710-SYSTEM-R2702H21.bin, version: Release 2702H21  Compiled May 20 2019 11:00:00 |

### ssh 服务检查

|  |  |
| --- | --- |
| 检查项 | 检查Border ssh服务正常开启 |
| 检查方法 | 在Border执行display ssh server status 命令 |
| 判断标准 | 正常是ssh与netconf服务均开启。如下表所示。 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| <demo1-H3C 6800>display ssh server status  Stelnet server: Enable  SSH version : 2.0  SSH authentication-timeout : 60 second(s)  SSH server key generating interval : 0 hour(s)  SSH authentication retries : 3 time(s)  SFTP server: Disable  SFTP Server Idle-Timeout: 10 minute(s)  NETCONF server: Enable  SCP server: Disable |

### 检查用户信息

|  |  |
| --- | --- |
| 检查项 | 检查Border用户，并赋予相应权限。 |
| 检查方法 | 在Border执行dis local-user user-name yop class manage 命令 |
| 判断标准 | 正常是yop用户有network-admin和network-operator 权限 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| <demo1-H3C 6800>dis local-user user-name yop class manage  Total 1 local users matched.  Device management user yop:  State: Active  Service type: SSH/Telnet  User group: system  Bind attributes:  Authorization attributes:  Work directory: flash:  User role list: network-admin, network-operator |

### BGP邻居关系

|  |  |
| --- | --- |
| 检查项 | 检查Border与Border ebgp邻居关系 |
| 检查方法 | display bgp peer l2vpn evpn |
| 判断标准 | 正常邻居关系是established。 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

正常状态示例：

|  |
| --- |
| <demo1-H3C 6800>display bgp peer l2vpn evpn  BGP local router ID: 169.254.1.5  Local AS number: 65534  Total number of peers: 2 Peers in established state: 2  \* - Dynamically created peer  Peer AS MsgRcvd MsgSent OutQ PrefRcv Up/Down State  10.17.62.249 65534 7775 1469 0 344 6693h45m Established  10.17.62.250 65534 809938 991950 0 344 6693h45m Established |

# 巡检项目-H3C 6800 VG（VG3.0）

* 环境中如无VG3.0，则跳过此项检查

## 每周或按月检查

VG 3.0 是使用专用网络设备替 换 x86 服务器 来实现 offloading 的 VG 方案。VG 3.0具有更强的网络处理能力及更低的成本，并且配合Region 部署，具有更高的可用性。

VG 3.0 主要实现了两个功能：公网 EIP 及公网负载均衡器。

VG 3.0 包含两个角色：硬件交换机和控制器 eipctl

### Eipctl控制器服务

|  |  |
| --- | --- |
| 检查项 | 检查eipctl控制器服务 |
| 检查方法 | * 确定环境中是否存在VG3.0   root@qingcloud-firstbox:~# cat /pitrix/conf/nodes/eipctl  如有输出，则说明是VG3.0环境   * 登录firstbox执行命令   root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh eipctl 'supervisorctl status' |
| 判断标准 | 正常服务是running状态 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

环境实际执行结果：

|  |
| --- |
| root@eipctl0:~# supervisorctl status  eipctl\_proxy RUNNING pid 12086, uptime 57 days, 16:25:30  eipctl\_server RUNNING pid 12098, uptime 57 days, 16:25:29  telegraf RUNNING pid 25372, uptime 57 days, 17:07:57 |

### BGP邻居关系

|  |  |
| --- | --- |
| 检查项 | 检查eipctl控制器与6800vg的邻居关系 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh eipctl 'gobgp n' |
| 判断标准 | 正常状态是Establ |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

正常输出示例：

|  |
| --- |
| root@eipctl0:~# gobgp n  Peer AS Up/Down State |#Received Accepted  1.0.0.10 65533 03:24:49 Establ | 10 10 |

### H3C与eipctl链接

|  |  |
| --- | --- |
| 检查项 | h3c 6800交换机与eipctl控制器是否成功建立连接 |
| 检查方法 | 1）登陆border交换机  root@demo1-eipctl0:~# gobgp n  Peer AS Up/Down State |#Received Accepted  169.254.1.4 65534 274d 21:24:41 Establ | 32 32  root@demo1-eipctl0:~# ssh yop@169.254.1.4(上文h3cvg交换机的地址)  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* Copyright (c) 2004-2019 New H3C Technologies Co., Ltd. All rights reserved.\*  \* Without the owner's prior written consent, \*  \* no decompiling or reverse-engineering shall be allowed. \*  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  <DEMO1-H3C 6800>  2）执行display bgp peer l2vpn evpn命令  <DEMO1-H3C 6800>display bgp peer l2vpn evpn  BGP local router ID: 169.254.1.4  Local AS number: 65533  Total number of peers: 4 Peers in established state: 4  \* - Dynamically created peer  Peer AS MsgRcvd MsgSent OutQ PrefRcv Up/Down State  10.17.62.220 65533 113 160 0 21 0527h32m Established  10.17.62.223 65533 63328 79183 0 21 0527h30m Established  10.17.62.224 65533 113 156 0 21 0527h27m Established  10.17.62.227 65533 63320 66690 0 21 0527h26m Established |
| 判断标准 | 正常状态是Established |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

正常状态示例：

|  |
| --- |
| <demo1-vg>display bgp peer l2vpn evpn  BGP local router ID: 11.11.11.2  Local AS number: 65533  Total number of peers: 4 Peers in established state: 4  \* - Dynamically created peer  Peer AS MsgRcvd MsgSent OutQ PrefRcv Up/Down State  78.10.1.204 65533 174685 1298 0 75 7031h38m Established  78.10.1.205 65533 174685 1424 0 75 7031h35m Established  78.10.1.206 65533 174685 1441 0 75 7031h33m Established  78.10.1.207 65533 174685 1257 0 75 7031h31m Established |

### H3C交换机eip\_network信息

|  |  |
| --- | --- |
| 检查项 | h3c 内看到系统为每个 eip\_network 自动创建的 vsi、vsi-interface 等信息 |
| 检查方法 | 登录h3c交换机，执行display current-configuration configuration vsi命令 |
| 判断标准 | 正常状态如下表示例 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

正确输出示例：

|  |
| --- |
| <DEMO1-H3C 6800>display current-configuration configuration vsi  #  vsi 1  gateway vsi-interface 1  flooding disable all  vxlan 1  evpn encapsulation vxlan  route-distinguisher 1:1  vpn-target 169.254.1.4:1 import-extcommunity  #  vsi 1000  gateway vsi-interface 1000  flooding disable all  vxlan 1000  evpn encapsulation vxlan  route-distinguisher 1:1000  vpn-target 169.254.1.4:1000 import-extcommunity  #  vsi 1001  gateway vsi-interface 1001  flooding disable all  vxlan 1001  evpn encapsulation vxlan  route-distinguisher 1:1001  vpn-target 169.254.1.4:1001 import-extcommunity  #  vsi 1002  gateway vsi-interface 1002  flooding disable all  vxlan 1002  evpn encapsulation vxlan  route-distinguisher 1:1002  vpn-target 169.254.1.4:1002 import-extcommunity  #  return |

# 巡检项目-rsserver

* rsserver涉及的IAAS模块包括novg和vbc3.0。环境中如果没有rsserver则跳过本章

## 每周或按月检查

NOVG是复用现有的网络交换机设备，通过BGP ipv4/ipv6来进行EIP流量牵引，不需要对EIP流量进行vxlan封装，而是直接通过三层网络进行转发。

优点是：

1、方便部署实施，复用现有的网络交换机设备，不需要额外的网络设备支持

2、不需要对EIP流量进行vxlan封装/解封装，性能介于x86 VG与VG 3.0之间。

vbc3.0是vbc2.0的升级版，采用bgp，同步基础网络的路由到核心交换机。

### rsserver服务

|  |  |
| --- | --- |
| 检查项 | 检查rsserver服务 |
| 检查方法 | * 确定环境中是否存在novg   root@qingcloud-firstbox:~# cat /pitrix/conf/nodes/rsserver  如有输出，则说明存在rsserver   * 登录firstbox，执行命令   root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh rsserver 'supervisorctl status' |
| 判断标准 | 正常状态是running |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例

|  |
| --- |
| root@rsserver2:~# supervisorctl status  telegraf RUNNING pid 26708, uptime 435 days, 21:33:48 |

### rsserver的BGP邻居关系

|  |  |
| --- | --- |
| 检查项 | 检查rsserver与云平台核心交换机或者leaf交换机的邻居关系 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh rsserver 'gobgp n' |
| 判断标准 | 正常状态是Establ |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

正确输出示例：

|  |
| --- |
| root@qingcloud-firstbox:~# /pitrix/upgrade/exec\_nodes.sh rsserver 'gobgp n'  ------------------------------------------------  TARGET NODES:  demo1-rsserver0,demo1-rsserver1,demo1-rsserver2,demo1-rsserver3  ------------------------------------------------  TARGET COMMAND:  gobgp n  ------------------------------------------------  Are you sure to exec the command on the above nodes [y/N]?y  Execing [demo1-rsserver0] with [gobgp n] ...  Peer AS Up/Down State |#Received Accepted  100.127.0.1 60000 465d 19:04:37 Establ | 4 4  Execing [demo1-rsserver3] with [gobgp n] ...  Peer AS Up/Down State |#Received Accepted  100.127.0.1 60000 465d 19:04:37 Establ | 4 4  198.19.0.18 3323133970 381d 01:08:16 Establ | 4 4  198.19.0.19 3323133971 381d 01:08:13 Establ | 4 4  Execing [demo1-rsserver1] with [gobgp n] ...  Peer AS Up/Down State |#Received Accepted  100.127.0.1 60000 465d 19:04:38 Establ | 4 4  Execing [demo1-rsserver2] with [gobgp n] ...  Peer AS Up/Down State |#Received Accepted  100.127.0.1 60000 465d 19:04:38 Establ | 4 0  198.19.0.18 3323133970 381d 01:08:14 Establ | 4 4  198.19.0.19 3323133971 381d 01:08:12 Establ | 4 4  Done. |

### 交换机的BGP邻居关系

|  |  |
| --- | --- |
| 检查项 | 核心交换机交换机或者leaf交换机与rsserver是否成功建立邻居关系 |
| 检查方法 | 需要登录核心交换机或者leaf交换机，执行display bgp peer命令（需要客户配合登陆交换机） |
| 判断标准 | 正常状态是Established |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| <R2>dis bgp peer  BGP local router ID : 100.127.0.1  Local AS number : 60000  Total number of peers : 4 Peers in established state : 4  Peer V AS MsgRcvd MsgSent OutQ PrefRcv Up/Down State  100.127.0.242 4 2887717979 125 96 0 3 01:27:57 Established  100.127.0.243 4 2887717980 139 166 0 2 02:08:48 Established  100.127.0.244 4 2887717981 130 122 0 3 02:08:48 Established  100.127.0.245 4 2887717982 128 133 0 3 02:08:48 Established |

### 查看交换机的路由信息

|  |  |
| --- | --- |
| 检查项 | 核心交换机上是否能看到路由信息 |
| 检查方法 | 登录核心交换机或者leaf交换机，执行display bgp routing-table ipv4 |
| 判断标准 | 正常是：   1. 能看到eip的32位路由。（novg环境） 2. 能看到基础网络的路由。（vbc3.0环境） |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

实际输出示例：

|  |
| --- |
| <demo1-core>display bgp routing-table  BGP Local router ID is 102.42.4.67  Status codes: \* - valid, > - best, d - damped,  h - history, i - internal, s - suppressed, S - Stale  Origin : i - IGP, e - EGP, ? - incomplete  Total Number of Routes: 8  Network NextHop MED LocPrf PrefVal Path/Ogn  \*> 120.199.68.166/32 100.127.0.12 0 0 44063.5213 25727.12?  \* 100.127.0.12 0 0 44063.5214 25727.12?  \*> 120.199.68.167/32 100.127.0.23 0 0 44063.5213 25727.23?  \* 100.127.0.23 0 0 44063.5214 25727.23?  \*> 120.199.68.168/32 100.127.0.23 0 0 44063.5213 25727.23?  \* 100.127.0.23 0 0 44063.5214 25727.23?  \*> 120.199.68.169/32 100.127.0.23 0 0 44063.5213 25727.23?  \* 100.127.0.23 0 0 44063.5214 25727.23? |

# 性能分析

## 网络性能分析

|  |  |
| --- | --- |
| 检查项 | 检查网络性能 |
| 检查方法 | 1）通过boss平台[告警记录]功能，搜索近1个月的记录，关键字[nic\_status\_errors]，如有出现需要更改网卡或者光模块。  2）通过boss平台[告警记录]功能，搜索近1个月的记录，关键字[cpu\_usage]，如有出现si高的告警，需要迁移虚拟机均衡资源或者单独进行调优。  3）通过boss平台[告警记录]功能，搜索近1个月的记录，关键字[kernel\_service ]，如有出现ping to [xx] is too high 则需要观察swap的使用情况，和物理机cpu si使用情况。 |
| 判断标准 | 正常是nic\_status\_errors cpu\_usage kernel\_service搜索无记录 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

## IO性能分析

|  |  |
| --- | --- |
| 检查项 | 检查IO性能 |
| 检查方法 | 通过boss平台[告警记录]功能，搜索近1个月的记录，关键字[cpu\_usage ],如有出现container util的告警，则需要通过iostat观察物理机磁盘读写情况，网络和物理硬件问题，都会导致磁盘读写异常。 |
| 判断标准 | 正常是cpu\_usage无记录 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。 |
|  |

# LB负载均衡器

|  |  |
| --- | --- |
| 检查项 | 检查LB文件系统 |
| 检查方法 | root@qingcloud-firstbox:~#  /pitrix/upgrade/exec\_nodes.sh hyper 'grep "detected having haproxy stat problem" /pitrix/log/compute\_agent.log.wf' |
| 判断标准 | 正常是搜索无记录 |
| 检查结果 | 请在此说明状态是否正常。  如果有状态异常，请在下方列出。  如果有关键字[ detected having haproxy stat problem]，表示lb后端节点文件系统异常，需要进行处理。 |
|  |

# IAAS巡检总结和建议

## 巡检总结

|  |
| --- |
| 分类指出平台问题 |

## 巡检建议

|  |  |
| --- | --- |
| 1 | 平台版本过低，需要升级至最新版本 |
| 2 | 有性能容量问题，需要扩容hyper节点 |