

分布式监控系统

功能需求

1, client实时监控的实现

<http://47.93.11.51/zentao/story-view-1-3-project-12.html>

1. 在Client端，单独创建子进程对六个指标项进行实时监控，按照项目不同，设置不同的时间间隔。（MEM, CPU 5S）（Disk 1min）（SYSinfo, User 1min）（Proc 30S）
2. 监控所得数据先存放在本地，每10次检测写1次硬盘
3. 系统调用可以使用system及popen
4. 请各位务必独立完成，可组内讨论，但不能COPY代码。

2, 检测到报警信息主动告知master

<http://47.93.11.51/zentao/story-view-2-4-project-12.html>

1. Client在自动检测的同时，如果检测到有报警信息，应第一时间告知Master
2. Client检测到恶意进程时，也主动告知Master
3. Master应该在收到警告信息后，将警告信息实时写到一个文件中，或者使用Linux Mail功能向管理员告警
- 4.

3, 监控数据处理

<http://47.93.11.51/zentao/story-view-10-2-project-12.html>

Client监控数据处理 在Master未上线之前，应由Client自行监控系统运行的各项指标

监控所得数据存放到指定位置（位置在配置文件中配置）

当单个数据文件大于30M的时候，应将其压缩并备份到备份路径（备份路径在配置文件中配置）

压缩备份的文件，其文件名需要唯一，并能按照时间顺序发送给Master

Client在Master上线的第一次数据传输过程中，应将之前备份文件解压传送到Master

在文件处理过程中，需要使用文件锁保证文件的正确性

发送给Master的任何数据，将在Client端不再保留。

为什么一定要对文件进行压缩备份？

在生产中，磁盘空间是很重要的，所以我们要尽可能的减少服务对磁盘空间的占用。

在很多情况下，服务器的磁盘会分层，也就是根据不同种类磁盘的读写速度等，对每一种磁盘的使用场景进行约束，对于要经常读写的热数据，我们会存放在读写速度快，但空间小的优质磁盘上，一般是专业的企业级PCIE SSD，而对于备份数据这样的冷数据，就会存放在读写速度慢，但是空间大的机械硬盘，甚至是磁带库中。

4, [脚本]恶意进程

<http://47.93.11.51/zentao/project-story-12--byModule-9.html>

在5s内如果某一进程的CPU或者内存占用超过50%，视为恶意进程

第一次检测时，获取所有可能的恶意进程

如果第一次检测时有潜在恶意进程，等待5s继续检测

如果潜在恶意进程仍然占超过阈值的资源，视为恶意进程

5, [脚本]c p u信息获取

<http://47.93.11.51/zentao/story-view-6-2-project-12.html>

CPU信息获取： 时间 负载1（1分钟） 负载2（5分钟） 负载3（15分钟） 占用率（时间间隔0.5） 当前温度 警告（normal, note（50-70），warning（70~））

该脚本主要用以获取CPU信息，请按照上述样式输出。

注意：请使用文件获取CPU占用率，不要使用其他命令获取

6, [脚本]系统运行概况获取

<http://47.93.11.51/zentao/story-view-7-1-project-12.html>

系统运行概况 时间 主机名 OS版本 内核版本 运行时间 平均负载 磁盘总量 磁盘已用% 内存大小 内存已用% CPU温度 磁盘报警级别 内存报警级别 CPU报警级别

请按照上述样式输出。

7, [脚本]用户信息获取

<http://47.93.11.51/zentao/story-view-8-1-project-12.html>

用户统计 时间 用户总数（非系统用户） 近期活跃用户（3个） 具有ROOT权限用户 当前在线用户_登录IP_TTY
[suyelu_IP_TTY,sd_ip_tty]

[suyelu,pi2]

8, [脚本]m e m内存信息获取

<http://47.93.11.51/zentao/story-view-5-2-project-12.html>

1. MEM 在树莓派上，使用Shell编程，输出以下内存信息：

间 总量 剩余量 当前占用（%） 占用百分比动态平均值 018-01-12_15:00:28 1023M 200M 80.4% 60.8%

动态平均值=0.3动态平均值（上一次）+0.7当前占用比 在脚本编写过程中，可以参考以下内容： `awk '{printf("%s\n", $1, $2)}' echo "scale=1;0.324+0.736" | bc` `array=(a c d e f)` `array=($1 $2)`

9, [脚本]磁盘信息获取

<http://47.93.11.51/zentao/story-view-4-2-project-12.html>

DISK 磁盘总量

磁盘剩余量

占用比例

各分区占用比例

各分区总量

分区剩余情况

2018-01-12_16:48:23 标识（0为整个磁盘，1为分区） 磁盘还是分区（disk| /boot, /） 磁盘/分区总量 磁盘/分区剩余量 占用比例

10 , m a s t e r 配置文件

<http://47.93.11.51/zentao/story-view-9-2-project-12.html>

Master端配置文件的作用 告知Master需要监控的主机范围

告知Master客户端Client的监听端口

收集到的数据存放位置

Master运行日志存放的位置

Master端配置文件的形式 Master端的所有配置信息形式Option=Value的形式，每条配置信息占一行数据。

`master=pi1 client_port=8732 PIHealthLog=/var/log/pihealth/pihealthd.log`

Master端配置文件的名称及路径 Master端配置文件完整路径为/etc/pihealthd.conf

对于软件中需要的更多配置，可根据自己的实际需求，将相应的配置信息添加到自己的配置文件中。

11, l o g m a n g e r 日志管理系统的实现

<http://47.93.11.51/zentao/story-view-3-1-project-12.html>

Master需将自己的运行日志，记录到以下位置/var/log/pihealthd.log 每条日志信息，应包含时间，事件，及提示性说明(错误，警告，接收数据，连接主机等等) Master需记录以下状态信息：启动，运行出错及详细说明，连接到客户端的地址，接收数据存放位置 日志以函数的形式实现 Client的日志存放在用户家目录下~/log/pihealth.log

12, 部署前测试

<http://47.93.11.51/zentao/story-view-11-1-project-12.html>

资源消耗测试 在正式部署之前需要做资源消耗的测试

Master, Client端正常运行, 使用nmon, dstat等工具检测CPU, 磁盘等主要指标项

不管是Master还是Client, 正常运行时所占资源越少越好

Client端所占资源超过总的10%不宜上线

稳定性测试 资源消耗测试通过后, 进行稳定性测试

测试要求10台树莓派全部部署Client, 自己的电脑作为Master, 跑2小时, 程序正常, 系统资源消耗正常, 无多余僵尸进程

通过第2条测试后, 跟树莓派管理员申请, 将Master迁移到树莓派上, 1Master 对 5 Client跑12小时无异常为通过

13, 部署上线

<http://47.93.11.51/zentaop/story-view-12-2-project-12.html>

部署上线 稳定性测试最终阶段阶段, 需部署上线。所有用户将Master部署在Pi9, Pi10上

Master端服务名: pihealthd_username

Client端服务名: client_username

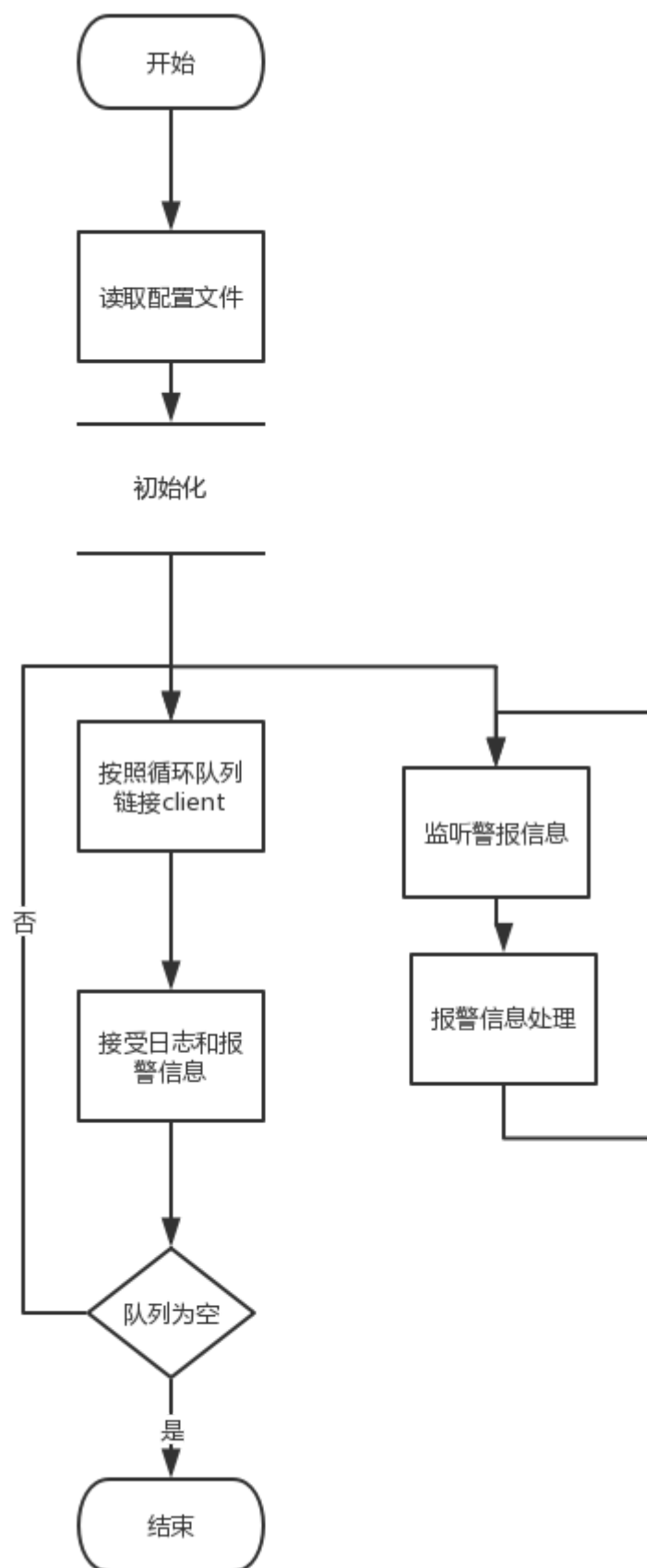
以下以用户zq为例 Master端可以使用以下命令:

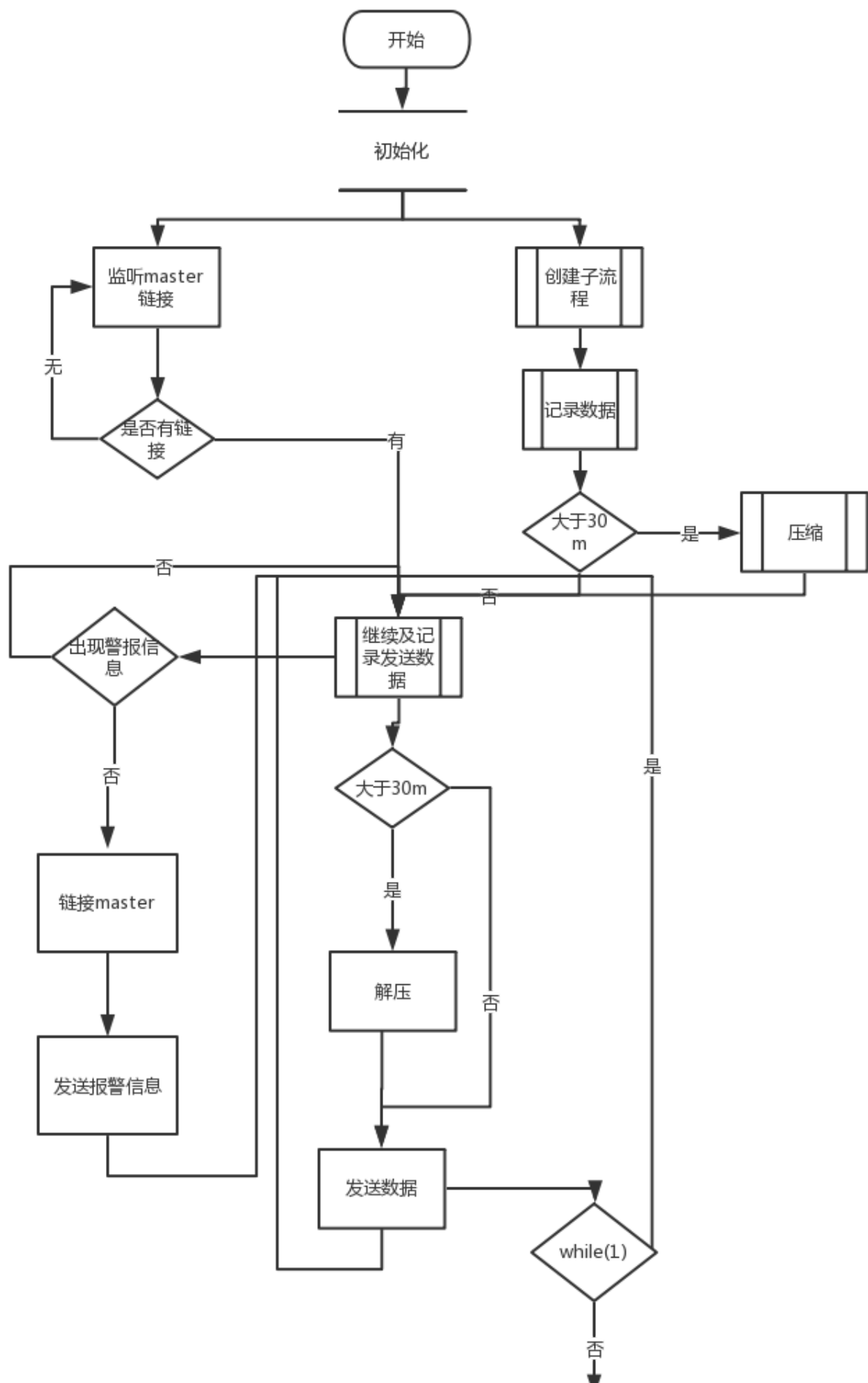
service pihealthd_zq start #启动服务 service pihealthd_zq stop #停止服务 service pihealthd_zq restart #重启服务 service pihealthd_zq status #查看状态 以Ubuntu现行命令格式:

systemctl start pihealthd_zq.service #启动服务 systemctl stop pihealthd_zq.service #停止服务 systemctl restart pihealthd_zq.service #重启服务 systemctl status pihealthd_zq.service #查看状态

Client端服务跟Master要求一致

使用命令systemctl enable pihealthd_zq.service将Master和Client端设置为开机自启







结束