|  |  |  |  |
| --- | --- | --- | --- |
| **文件编码** | EY-PM-TS002 |  |  |
| **最新发布日期** | 2017年 | **当前版本** | 1.0 |

**易云捷讯科技（北京）有限公司**

**EayunCloud总体设计**

**郑重声明：**易云捷讯科技（北京）有限公司版权所有。本文档中任何部分未经易云捷讯科技（北京）有限公司书面授权，不得将材料泄露给第三方，不得以任何手段、任何形式进行复制与传播。

**变更履历**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **版本** | **日期** | **变更位置** | **变更理由/变更内容** | **变更人** | **备注** |
| 1.0 | 2017-01-10 |  | 新建 | 段彬彬  曹翔宇  王建凯  郑海龙  高翔  刘壮壮 |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# 目录

[目录 3](#_Toc471912025)

[1. OBS文件上传 5](#_Toc471912026)

[背景 5](#_Toc471912027)

[遇到的难点 5](#_Toc471912028)

[1.1 跨域访问 5](#_Toc471912029)

[1.2 鉴权 5](#_Toc471912030)

[设计实现 5](#_Toc471912031)

[1.1 UploadStorageCtrl 5](#_Toc471912032)

[1.2 obsService 6](#_Toc471912033)

[1.3 ObsHttp 8](#_Toc471912034)

[2. 各后付费资源定期扣费 9](#_Toc471912035)

[业务介绍 9](#_Toc471912036)

[业务流程 9](#_Toc471912037)

[3. 对象存储定期扣费 10](#_Toc471912038)

[业务介绍 10](#_Toc471912039)

[业务流程 11](#_Toc471912040)

[4. 欠费处理 13](#_Toc471912041)

[业务介绍 13](#_Toc471912042)

[业务流程 14](#_Toc471912043)

[5. 云资源监控数据采集和报警 17](#_Toc471912044)

[业务分析 17](#_Toc471912045)

[业务流程图 17](#_Toc471912046)

[采集业务 17](#_Toc471912047)

[报警监控项业务 18](#_Toc471912048)

[6. 计划任务漏跑检测 20](#_Toc471912049)

[业务描述 20](#_Toc471912050)

[业务流程图 22](#_Toc471912051)

[7. API总体设计 23](#_Toc471912052)

[业务描述 23](#_Toc471912053)

[业务流程图 24](#_Toc471912054)

[8. API监控报警 25](#_Toc471912055)

[业务描述 25](#_Toc471912056)

[业务流程图 26](#_Toc471912057)

[9. 底层数据同步 29](#_Toc471912058)

[业务描述 29](#_Toc471912059)

[流程图 30](#_Toc471912060)

[10. ECSC预付费资源创建 31](#_Toc471912061)

[概述 31](#_Toc471912062)

[1、生成订单阶段 31](#_Toc471912063)

[2、创建资源阶段 33](#_Toc471912064)

[3、资源状态同步处理 35](#_Toc471912065)

[工具接口说明 37](#_Toc471912066)

[4.1、创建成功后的处理接口 37](#_Toc471912067)

[4.2、创建失败后的处理接口 37](#_Toc471912068)

[4.3、第三方（支付宝）支付接口 39](#_Toc471912069)

[11. 资源变配 42](#_Toc471912070)

[业务描述 42](#_Toc471912071)

[流程图 44](#_Toc471912072)

# OBS文件上传

## 背景

原本的文件上传需要先将文件上传到服务器，再上传至OBS存储，导致文件上传速度较慢，经讨论后决定改为使用js在客户端直接上传文件到OBS存储，减少一次文件传输。

## 遇到的难点

* 1. 跨域访问

解决办法：新建bucket时设置为允许跨域访问

* 1. 鉴权

解决办法：每次调用OBS的api之前，先调用后台接口进行加密

## 设计实现

* 1. UploadStorageCtrl

UploadStorageCtrl中封装了前台校验，校验通过后调用obsService中提供的接口上传文件



* 1. obsService

obsService中封装了调用OBS文件上传API的方法（通过ObsHttp调用），并对外提供文件上传（uploadFile）和取消上传（cancelUpload）两个方法。





* 1. ObsHttp

ObsHttp封装了调用后台加密的方法，使用ObsHttp发起的请求，都会先请求一次后台进行加密。

# 各后付费资源定期扣费

## 业务介绍

在后付费计费中,所有的后付费云资源(云主机，云硬盘，云硬盘快照，私有网络，负载均衡，公网ip，vpn)的按时扣费，扣费时间为每小时扣一次费。

## 业务流程



每小时扣费时,先查询出所有的客户,再在去查询属于各个客户的所有有效的后付费资源,以客户为粒度的去扣费,在扣费过程中,所有关于金额相关的操作,一律放在分布式锁中进行,保证无论在单线程/多线程情况下,账户金额都能保证正确。

# 对象存储定期扣费

## 业务介绍

除了后付费云资源的按时扣费外，对象存储也是需要按时扣费，在客户开通了对象存储服务后，从下一整点开始，即需要计算该客户每小时使用的对象存储所需扣除的费用，对象存储扣费项包括存储容量，下载流量，请求次数，如开通了cdn，则还包括cdn下载流量，https请求数，动态请求数。

## 业务流程



由于对象存储扣费时的计费因子有多个，并且是在不同的计划任务中去获得的，所以必须等到所有的统计都成功后，才可执行计费逻辑，否则不会去执行计费。

# 欠费处理

## 业务介绍

所有的云资源(包括预付费和后付费)，对象存储服务，在欠费或者达到到期时间后，如未充值或未续费，则需要对资源进行欠费/到期处理。

## 业务流程



在回收站的资源不参与到期超过保留时长处理，需将资源从回收站移出后才可执行到期超过保留时长处理。



后付费资源计算保留时长时间是从首次达到信用额度时算起，在回收站的资源不参与欠费处理，需将回收站的资源从回收站移出后才可参数欠费处理；后付费资源包括云主机，云硬盘，云硬盘快照，私有网络，负载均衡，公网ip，vpn，对象存储，其中对象存储欠费超过保留时长后的操作为禁止上传与下载，开启/关闭cdn，但仍可正常查看bucket信息。

无论后付费还是预付费云资源，云主机超过保留时长处理为更改资源状态，关机，解绑云硬盘与公网ip；云硬盘为更改资源状态，与云主机解绑；云硬盘快照(只有后付费)为更改资源状态；私有网络为更改资源状态，将带宽设为1；负载均衡为更改资源状态，解绑公网ip；VPN为更改资源状态；公网ip为释放。

# 云资源监控数据采集和报警

## 业务分析

系统通过计划任务每分钟从底层采集云主机的各项指标信息，包括CPU利用率、内存占有率、网卡下行速率、网卡下行速率、磁盘写吞吐和磁盘写吞吐，将采集到的信息存入mongo中，并将最新的两条信息存入redis中。

ECMC和ECSC均可针对云主机的各项指标设置监控报警，运维或用户设置了报警规则，并添加了需要监控的云主机后，系统会通过计划任务每分钟校验主机的指标信息是否满足了报警的触发条件，如果持续一段时间内均满足，则会触发报警产生报警信息，并根据设置的报警联系人信息发送相应的邮件和短信。

连续时间段内满足触发条件时，只在第一次产生报警信息和发送短信邮件，之后不再产生和发送；如果用户手动消除了报警信息，则会继续产生和发送；如果连续24h满足了触发条件，则会重新给设置的联系人发送短信和邮件，不会产生新的报警信息。

## 业务流程图

### 采集业务

每分钟0秒执行



### 报警监控项业务

每分钟0秒执行一次



# 计划任务漏跑检测

## 业务描述

我们的系统现在有若干不同业务的计划任务在后台静默致行，为不同的系统功能提供服务，大部分情况下其都是可以正常执行的，但是不排除由于某些原因而导致的计划任务出现漏跑的情况，此业务功能的主要目的便是监听每一项计划任务的执行状态，若有漏跑现象发生，则会将漏跑的时间点以及漏跑次数记录到数据库中，同时通过短信和邮件提醒有关人员，同时在ECMC管理端有一个专门的页面显示漏跑记录，相关的人员便可以通过这些信息去做一些功能性的排查。

业务基本算法：计划任务漏跑检测这项功能本身也是作为一个计划任务而存在，利用Spring与Quartz的集成方式实现，部署在ECMC服务端，每分钟整点执行检测。

1. ECMC服务的部署方式为集群部署，所以首先我们需要控制在同一个时刻，只能有一台机器执行该计划任务，利用Redis计数的方式实现。
2. 查询出系统数据字典中所配置的所有需要监测的计划任务信息。若数据字典中没有配置该类数据，则表示监测所有的计划任务。
3. 遍历每一项需要检验的计划任务项，根据指定的判定规则判断当前计划任务是否出现了漏跑现象。检测规则：考虑到每一项计划任务的执行时间以及一些其他的额外因素，我们需要设置一个一分钟的容错时间，检测规则即为：“若该项任务当前时间大于等于下次执行时间加上这一个容错时间的值，则判定该任务在当前这个时间产生了漏跑”。
4. 根据指定的规则计算当前的任务已经连续漏跑的次数。计算规则为：“总次数=当前时间减去下一次执行时间的值除以该任务的标准执行间隔时间，结果向下取整后加1”。
5. 更新或者重新插入一条计划任务漏跑异常信息到数据库，该信息分为以下及部分内容：计划任务名称、当前时间、计划任务Cron表达式、计划任务执行标准间隔时间、开始漏跑时间、最后一次漏跑时间点、总漏跑次数。
6. 经过以上的步骤之后，已经可以得出当前时刻所有产生漏跑的计划任务信息，将这些信息保存到Redis中。
7. 最后通过邮件、短信发送报警提醒。发送规则：“不同时刻连续出现的漏跑现象只发送一次漏跑提醒，在开始的时间点；且发送时将所有的漏跑信息组合在一起一次性发送”。
8. 根据上述的规则，从Redis中取出上一个时刻的漏跑信息与本次时刻的漏跑信息作比较，若有新计划任务漏跑信息出现，则通过短信邮件的方式提醒这部分的漏跑信息。

## 业务流程图



# API总体设计

## 业务描述

API总体设计是API需求的开始总体分为以下几个部分：

扩展性设计：API架构的设计必须要满足良好的可扩展性，因为API的服务是随着时间和需求不断增加丰富的，必须要达到方便添加新API接口的能力。采用注解的形式，API系统服务会自动扫描所有被标注指定注解的服务接口，通过注解我们可以扫描得到详细的操作任务名称以及对应的方法和类名称，最后可以通过反射的方式进行业务调用。

统一异常处理：利用Spring提供的“@ControllerAdvice”机制来提供全局的异常处理，系统调用产生的所有异常最终都会从此入口进入，程序中便可以根据实际的业务需要对异常进行解析操作。

## 业务流程图



# API监控报警

## 业务描述

API的监控报警针对API的服务调用，API服务的特点是访问客户以及访问的数据量以及访问时间等指标都是不确定因素，统计各个指标的变化数据或者变化趋势有时候便显得尤为重要，比如一个客户在自己的机器上很频繁的调用不同的API服务，那我们怎么知道这个客户是正常的用户还是恶意攻击的行为呢？资源监控的作用便是实时的监测每一类访问的指标变化趋势，假如观察到某个客户对每个API服务访问的时间持续增加的话，就该引起管理员的注意了。报警管理旨在当出现某些可以预料的意外或者错误情况的时候，自动给管理员发送对应的报警提醒信息以便及时处理，例如，设定五分钟之内连续正确率低于50%，可能就说明API服务系统出现了比较大的漏洞了，那么报警管理模块会按照指定的时间间隔去检测当前的执行场景是否满足各个报警条件，若满足，则立刻发送对应的报警信息。

对应的资源监控和报警管理的算法可以参见《v1.0 API概要设计（资源监控与报警管理）.doc》文档说明。

## 业务流程图





# 底层数据同步

## 业务描述

底层数据的同步就是为了使上层数据与底层数据保持一致。

流程描述：

1. 分别获取底层数据库和上层数据库的对应资源的数据。
2. 以底层数据为标准进行对比，针对可能出现的情况分别做处理：
   1. 底层和上层都存在，根据底层数据更新对应的上层数据；
   2. 底层存在，上层不存在，将对应的数据添加到上层数据库中；
   3. 底层不存在，上层存在，删除上层对应的数据，并且给运维发送删除资源的邮件。

## 流程图



# ECSC预付费资源创建

## 概述

Ecsc中预付费资源的创建流程大体上可以分成三个阶段，分别是生成订单阶段，创建资源阶段和资源同步阶段，以下将分别展开讲述各个阶段大体的流程。

## 1、生成订单阶段

流程图如下：



流程叙述：

1. 提交订单时，记录用户将要创建的资源的配置参数到cloudorder\_资源名称数据表中。
2. 是否使用余额支付的方式，这里分两种情况讨论：
   1. 使用余额支付，并且余额足够支付产品金额。
      1. 将订单数据信息状态修改为“处理中”。
      2. 向RabbitMQ发送新购资源的消息。
      3. 向用户发送消息，提醒新订单已生成。
   2. 采用第三方（支付宝）支付或部分采用第三方支付。
      1. 将订单数据信息状态修改为“待支付”。
      2. 调用第三方（支付宝）支付接口（后面章节详述）。

## 2、创建资源阶段

流程图如下：



流程叙述：

1. 预付费资源的MQConsumer接口监听RabbitMQ消息队列当中资源信息。
2. 发现相应资源信息，从MQ当中取出。如果没发现，返回第一步。
3. 根据获取消息中的订单编号，在“cloudorder\_资源名称”表中获取对应订单编号的资源配置信息。
4. 以这些配置信息为参数，调用对应的openstack的创建API接口，创建资源：
   1. 创建失败，调用资源创建失败的处理接口（后面会详述）。
   2. 创建成功，将配置参数存入数据库资源表，并设置为不展示给用户的状态（用户无法通过ecsc直接看到该资源）。
5. API接口返回创建成功的资源业务状态：
   1. 如果为创建完成（一般为ACITVE），将数据资源表中对应资源，设置为用户可见。并调用资源创建成功的处理接口（后面详述）。
   2. 如果业务状态不是创建完成（例如PEDDING\_CREATE创建中），将该资源的业务状态等信息发送至Redis消息队列中。

## 3、资源状态同步处理

流程图如下：



流程叙述：

1. 每隔5秒监听Redis消息队列的资源状态消息（创建资源阶段发送的）；
2. 获取消息中资源的状态，并从底层获取资源实际的业务状态，对比两者：
   1. 前者为创建中，后者为创建完成。直接将数据库中对应资源状态更新为创建完成，并设置为用户可见状态。调用资源创建成功后的处理接口（后面详述）。
   2. 非，获取该条消息的次数：
      1. 少于101次。将该资源的状态信息重新发送回Redis消息队列。
      2. 多于100次。调用资源创建失败后的处理接口（后面详述）。

## 工具接口说明

## 4.1、创建成功后的处理接口

流程图如下：



功能叙述：

如图所示，直接修改订单数据状态，从“处理中”变成“已完成”。

### 4.2、创建失败后的处理接口

流程图如下：



流程叙述：

1. 修改订单数据状态，从“处理中”变为“处理失败已取消”。
2. 向用户发送资源创建失败的邮件。
3. 如果有部分资源创建成功，调用资源删除接口，删除创建出来的“垃圾资源”。
4. 如果删除失败，继续向用户发送资源删除失败的邮件。

### 4.3、第三方（支付宝）支付接口

流程图如下：



流程叙述：

1. 调用支付宝支付接口。
2. 判断支付结果是否已经成功。

3、支付结果已成功。

* 1. 修改订单数据状态为“处理中”。
  2. 向消息队列发送创建该资源的信息。

1. 支付结果尚未成功。
   1. 循环定时查询订单支付状态。
   2. 查询结果为已成功。返回步骤3继续执行。
   3. 查询结果仍为尚未成功。判断支付时间是否超时：
      1. 超时。则修改订单状态信息为支付失败。
      2. 未超时。终止流程，等待下一次定时查询。

# 资源变配

## 业务描述

流程描述：

1. 首先需要判断是否满足升级/降级的条件，主要有以下几个条件：
   1. 变配资源的配额是否充足；
   2. 账户余额是否充足；
   3. 是否有未完成的订单。
2. 如果满足配额、余额充足，并且没有未完成的订单，则生成变配资源的订单，订单的状态为处理中；
3. 更改资源的配置，主要有两步：
   1. 首先，调用底层的接口修改对应资源的配置；
   2. 其次，修改上层数据库中对应资源的配置。

注：如果①处理失败，不会在进行②操作。

1. 判断是否修改成功，有两种可能：
   1. 修改成功（底层和上层全部修改成功），更新订单的状态为已完成，并且发送资源变配成功的消息（用于调整计费），此次变配成功；
   2. 修改失败（底层和上层任意一个失败），更新订单的状态为处理失败已取消，此次变配失败。

注：云主机的变配是比较特殊的，主要体现在第3、4两步，原因是由于云主机存在中间状态（升级中）。不同点：第三步中，需要修改上层数据库中云主机的状态置为升级中；第四步中，需要依据底层数据状态同步来更新上层数据库中云主机的状态，并且判断是否成功。

如何确保计划任务漏跑之后计费仍然准确？

答：在设计计费清单表中，有一个字段用于标识计费资源开始/上次计费时间，如果计划任务漏跑，下次计费时，靠时间间隔也可以补扣漏跑应扣未扣的费用。例如：10:30云主机进行升级，计费清单表中会针对此次升级生成新的记录，且会标示资源开始计费时间（当前时间），如果在11:00时计划任务漏跑（计划任务漏跑会给运维发送任务漏跑的邮件），在12:00时计划任务执行的时候，会根据当前时间和上次计费时间计算出需要计费的时长（10:30~12:00，90分钟），然后扣除相应的费用。

## 流程图

