流媒体框架总体设计

2012\_12\_7 rui.sun

1. 总体设计

本系统属于云监控系统，旨在方便用户设置不同的监控端将录像上传至云服务器中。

方便用户实时用 pc、web、ios、iphone 等实时查看。

1. 需求分析

本系统一共分为 4 大块，注册管理、设备 bridge agent 组管理、中心服务器，存储服务器组成。

注册管理主要对用户的信息、设备的信息进行管理。

bridge agent 主要负责接入第三方不同的设备。

中心服务器，负责登陆管理、用户请求分发、管理调度、保持心跳、报警预警等。

存储服务器主要负责录像和回放请求。

1. 概要设计
2. 注册管理

将分为一个管理员(root 立创建一个固定的组group)，和 N 个用户组成。

其中管理员，拥有至高无尚的权限。（包括添加设备、控制计划录像时间、管理 ptz 等).

管理员，可以对用户的权限有一个设置，如，是否可以控制 ptz 等。（单独的一个表）。

对于登陆的管理员，可以选择逐一添加的设备。

1. 前端 bridge agent

此模块主要用来适配不同的用户前端。（如汉邦、大华设备等）。

主要分为以下几个需求。

1. 向服务器注册，表明以在线。（告之服务器 http 的 root 目录是多少）。

2. 启动并监听中心服务器的添加任务请求。

3. 添加特定的设备，并实时的产成码流。（如果有上传任务的，可以发送至指定的 http此功能可以随时调用）。

4. 停止设备。

5. 保持心跳及断电、出错向服务器报警及本地报警日志。

1. 存储服务器

这部分将使用 php + opendfs 来配合完成存储的需求。

主要有以下功能。

1. 接收特定用户的设计的上传流请求，并产生数据库记录。

2. 提供 api 用于设备录像记录列表的检索。（以小时为单位，因为服务器录像是以小时为单位的。）

3. 根据用户需要查看的特殊小时段，生成一个访问地址。动态生成一个列表。

1. 中心服务器

中心服务器将是工作的心脏，需要考虑诸多稳定性，以及 fail-over 等需求。

1. 负责用户注册、管理用户、设备的请求。（用户上下线，不在通知给其它的用户，设备上下线要通知给在线的用户（以报警事件）。）。

2. 负责提供 api，取得当前所有的组内用户，设备、删除用户设备功能。

3. 记录所有的事件日志，例如说，用户登陆，退出，删除、修改、管理。

4. 负责接入所有连接到设备上的设备，所有登陆到服务器的用户信息（临时信息）。

5. 负责转发用户请求到设备 agent，例如说 ptz。

6. 负责添加录像计划，并管理用户的录像计划，到点的时候，会发送到存储服务器。

1. 详细设计
2. 注册管理

注册管理模块

整个注册管理是比较重要的一环，贯穿着整个系统的配置和动作。

其中用户的数据信息分为在线部分和离线部分。

在线数据包括

1. 用户登陆的状态信息。（管理员，附属的用户）

2. 前端 agent 在线状态。

3. 中心服务器在线状态。

静态数据包括

1. 用户账户信息（管理员，附属于的用户）

2. 账户对用户的权限设计表。

3. 用户设置的可以访问的 agent 列表。

4. 用户设置的存储计划时间。

5. 中心服务器的固定登陆 ip 信息。（预设2+最好）

对于日志部分分为

1. 用户的操作日志。

类型分为 login/out, add/del user, setperssion, realplay ba, stop ba,

admin add/del record task/

2. 用户的错误日志

1. ba 启动错误，ba 断线, ba 任务启动失败，存储上传失败，user 非正常退出，

关于数据库部分

首先是 ba 表

t\_account\_ agent 字段为

uuid, ref\_admin\_uid, plugin\_name, dev\_ip, dev\_port, dev\_user, dev\_pwd, channel\_number, channel\_count, seg\_folder, seg\_file, seg\_in\_count, seg\_per\_sec, comment, created\_time

t\_account\_license\_type

license\_type, comment

其中恒字段为

0, 'this license is now must be checking by service root.'

1, 'this license have expired time limited.'

2, 'this license is for ever for use'

t\_account\_license

license\_type 0: 正在审核 1: 有时间限制 2: 永久

ref\_admin\_uid, license\_type, licensed\_time, expired\_time

// 用于以后更新的授权，直接写在一个单独日志表里面即可

t\_account\_admin 字段为

uid, pwd, nickname,comment, created\_time

t\_account\_child\_permission 字段为

ref\_admin\_uid, can\_view\_ba, can\_view\_record, can\_ptz\_ba, can\_task\_record

t\_account\_child 字段为

uid, pwd, ref\_admin\_uid, nickname,comment ,created\_time

工作原理：

一

一开始用户在客户端，创建一个管理员，接着，会录入到 t\_user\_admin 中

接着建立到 t\_user\_admin\_license 表中 license\_index = 0 licensetype = 0

接着服务器会收到一个审核，并提示出来。（可以一开始提示，或者加入到待审核列表中）

如果这个客户端root建立起来了，就可以会其配置子的用户了。

对于添加 ba 设备的选项 guid 是动态生成的，作为一个设备的唯一标识，但也不一定

（在添加判断的时候是靠，ip和channel\_number判断的）。

1. 前端 bridge agent
2. 存储服务器
3. 中心服务器
4. 后记