用户认证与存储服务

一、服务范例

服务均通过TCP协议进行通讯，以便能够灵活部署到网络节点中。

服务端口范围设定在2500~2599之间。

通讯语言使用XML：以<x>为xml数据流开始，</x>为xml数据流结束。

XML的处理使用org.xml.sax和DOM4J开源项目以保证效率。

二、用户认证服务

用户认证服务包括：

* 用户认证  
  请求：<v><u>username</u><p>password</p></v>  
  OK相应：<ok>随机会话编号</ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 用户会话认证  
  请求：<c><s>会话编号</s></c>  
  OK相应：<ok>用户名</ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 用户添加  
  请求：<a><u>username</u><p>password</p></a>  
  OK相应：<ok></ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 用户删除  
  请求：<d><si>会话编号</si><u>username</u></d>  
  OK相应：<ok></ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 用户密码修改  
  请求：<m><si>会话编号</si><p>new password</p></m>  
  OK相应：<ok></ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 用户组添加  
  请求：<ga><si>会话编号</si><g>group name</g></ga>  
  OK相应：<ok></ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 用户组删除  
  请求：<gd><si>会话编号</si><g>group name</g></gd>  
  OK相应：<ok></ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 用户组枚举  
  请求：<gl><si>会话编号</si></gl>  
  OK相应：<ok><g>group1</g><g>group2</g>…</ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 用户组中的用户枚举  
  请求：<gul><si>会话编号</si><g>group name</g></gul>  
  OK相应：<ok><u>user1</u><u>user2</u>…</ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 用户关联到用户组(此处不能关联到app组)  
  请求：<gua><si>会话编号</si><u>user</u><g>group </g></gua>  
  OK相应：<ok></ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 用户取消用户组关联(此处不能与app组取消关联)  
  请求：<gur><si>会话编号</si><u>user</u><g>group name</g></gur>  
  OK相应：<ok></ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 用户的组属性枚举  
  请求：<ug><si>会话编号</si><u>user</u></ug>  
  OK相应：<ok><g>group1</g><g>group2</g>…</ok>  
  失败相应：<err c=“错误代码”>错误原因</err>

内置管理员组：admin（管理员组），app（应用组）

内置管理员账户：admin，属于admin组

密码枚举保护：连续密码错误三次则暂停对该用户的认证5分钟。

用户会话失效：会话建立60分钟后失效，会话失效后需要重新认证。

实现Java API，封装上面的通讯。

三、存储服务

每一个应用会使用一个用户来访问存储服务，因此存储服务的管理和访问都基于用户进行

存储管理功能（仅admin组用户）：

* 枚举用户  
  请求：<mul><si>会话编号</si></mul>  
  OK相应：<ok><u>user1</u><u>user2</u>…</ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 添加用户(包括三个动作：新建表user\_group相应项，新建表app\_store相应项，新建物理空间中相应的文件夹)(暂未考虑当操作中出现连接等问题时的redo/undo问题)  
  请求：<mua><si>会话编号</si><u>user</u></mua>  
  OK相应：<ok></ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 删除用户(包括三个动作：删除表user\_group相应项，删除表app\_store相应项，删除物理空间中相应的文件夹) (暂未考虑当操作中出现连接等问题时的redo/undo问题)  
  请求：<mud><si>会话编号</si><u>user</u></mud>   
  OK相应：<ok></ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 读取用户配额  
  请求：<mur><si>会话编号</si><u>user</u></mur>  
  OK相应：<ok><q>空间大小（MB）</q><a>已用空间大小(MB)</a></ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 设置用户配额  
  请求：<mus><si>会话编号</si><u>user</u><q>空间大小（MB）</q></mus>  
  OK相应：<ok></ok>  
  失败相应：<err c=“错误代码”>错误原因</err>

存储使用功能（仅app组用户）：

* 枚举目录(结果未包括子目录名，只列出了子文件名)  
  请求：<l><si>会话编号</si><p>路径</p></l>  
  OK相应：<ok><f>文件名1</f><f>文件名2</f>…</ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 建立目录(<m>与前面某项重复，改成<md>)  
  请求：<md><si>会话编号</si><p>目录路径</p></md>  
  OK相应：<ok></ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 测试文件存在  
  请求：<e><si>会话编号</si><p>路径</p></e>  
  OK相应：<ok>true/false</ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 删除目录  
  请求：<r><si>会话编号</si><p>路径</p></r>  
  OK相应：<ok></ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 获得文件属性(<t>创建时间</t> 改成 <i>创建时间</i>,文件长度应该为文件/目录大小)  
  请求：<fa><si>会话编号</si><p>路径</p></fa>  
  OK相应：<ok><t>d目录/f文件</t><i>创建时间</i><m>最后修改时间</m><l>文件长度(大小)</l></ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 读取文件  
  请求：<fr><si>会话编号</si><p>文件路径</p><b>启示字节</b><e>终止字节</e></fr>  
  OK相应：<ok><![CDATA[数据]]></ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 写入文件(<b>起始字节</b> 表示如果没有原文件或原文件内容为空，则忽略b，直接新建文件并将数据写入；如果原文件有内容，且b<length，则从b处把新数据追加进文件；如果原文件有内容，且b>=length，则直接追加数据进文件。可以在以后改成一个交互版本，询问读者的意图。)  
  请求：<fw><si>会话编号</si><p>路径</p><b>启示字节</b><d><![CDATA[数据]]></d></fw>  
  OK相应：<ok></ok>  
  失败相应：<err c=“错误代码”>错误原因</err>
* 删除文件(<d>与前面某项重复，改成<df>)  
  请求：<df><si>会话编号</si><p>路径</p></df>  
  OK相应：<ok></ok>  
  失败相应：<err c=“错误代码”>错误原因</err>

当一个用户的文件总量超出配额时，返回配额超出错误。

实现Java API，类似于java.io.File类，封装上面的通讯。