



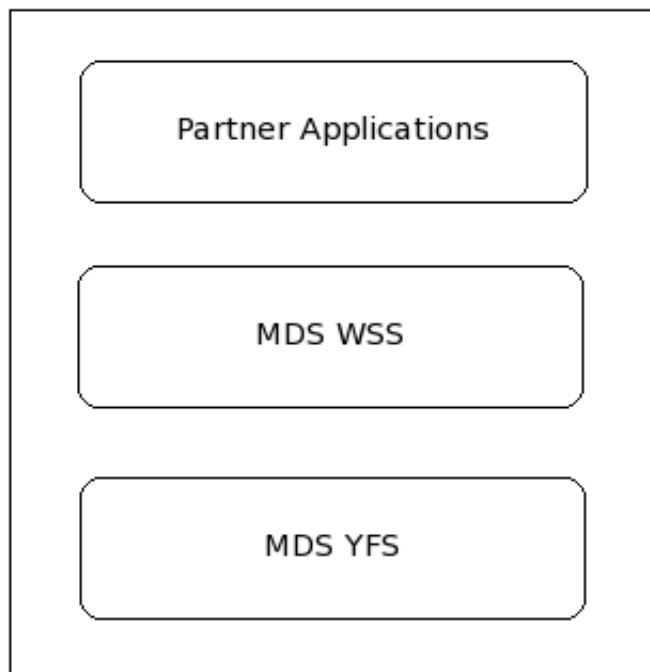
美地森 Web 存储服务
MDS Web Storage Service (WSS)
开发者手册
API 版本 2008-08 beta1

Table of Contents

MDS USS 概述.....	3
MDS USS 平台:.....	3
有关命名空间.....	4
帐户设置.....	5
主帐户.....	5
App 帐户.....	5
子帐户.....	5
认证.....	5
认证操作命名空间(auth).....	6
login.....	6
logout.....	7
passwd.....	7
文件操作命名空间(usfs).....	8
mkdir.....	8
rm.....	9
rmdir.....	9
ls.....	10
mv.....	11
down.....	12
up.....	13
传输操作命名空间(trans).....	14
Upload.....	14
Http Download.....	15

MDS USS 概述

MDS USS 平台：



美地森通用存储服务是一种允许用户通过调用 web 服务接口对文件进行上传，下载和组织在互联网存储服务。它建立在美地森自主研发的分布式文件系统 YFS 之上，其目的是为用户提供大容量，高可用，高安全的存储服务。

伙伴的应用程序可以通过调用 MDS WSS REST 接口使用 MDS Web 存储服务，关于 REST 的详细信息可以 [REST](#) 访问。REST 接口可通过 HTTPS 调用。

REST 接口可通过 HTTP / HTTPS POST 或者 GET 请求相应的 web 页面进行调用，例如：

URL 格式为：

<http://wss.googlefs.com/ws/{NameSpace}/MethodName?name=value&name2=value2>

下面为认证的一个实例：

<http://wss.googlefs.com/ws/auth/login/?app=8da051b0-a60f-4c22-a8e0-d9380edafa6f&username=testuser&password=abc123>

REST 请求路径：

有些接口需要路径参数，代表一个文件或者目录。路径为相对路径。以子用户的根目录作为当前目录。

下面为两种路径为同一文件：

相对路径：web/index.html

相对路径：./web/index.html

REST 回复格式

“0” 代表成功，大于 0 表示错误

```
<Response>
```

```
    <ResponseCode>0</ResponseCode>
```

```
    <ResponseMessage>Operation success</ResponseMessage>
```

```
</Response>
```

```
<Response>
```

```
    <ResponseCode>1</ResponseCode>
```

```
    <ErrorMessage>Username or password error</ErrorMessage>
```

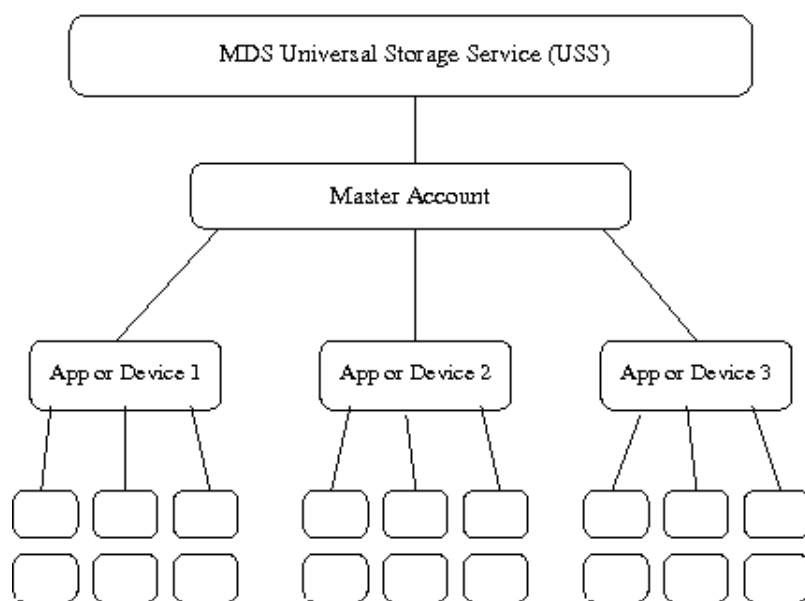
```
</Response>
```

有关命名空间

命名空间是对一个或多个实体的逻辑上下文抽象。我们的存储服务使用命名空间来管理一组相关的服务接口，在命名空间内定义的接口与该命名空间相关联，相同名字的接口可在多个命名空间中定义，即某一命名空间下的接口独立于其他命名空间的接口。

在后面的章节中将详细解释各命名空间下的接口调用。

帐户设置



MDS WSS 支持两种类型帐户，主帐户和用户帐户，主帐户为使用 WSS 的计费帐户，并管理其下的子帐户。

主帐户

主帐户支持多个子帐户。例如：如果你想用 MDS WSS 作为备份服务，则你需要为这个服务创建一个主帐户，当你创建一个主帐户的时候，至少需要创建一个应用服务，然后为使用你的备份服务的用户单独创建子帐户，你可以给每个子帐户限制不同存储空间。如果你有了新业务，如邮件服务，你可以在已有的主帐户下再创建一个 Mail 的应用服务。

创建主帐户：主帐户通过 MDS WSS 的帐户管理 web 页面来创建和管理，在这里还可以在主帐户下创建不同的应用服务。

App 帐户

App 帐户用于区分不同的子帐户群，每一个 App 帐户都有一个独一无二的 key，在子帐户进行认证的时候需要该 key。

子帐户

创建子帐户：子帐户由主帐户在管理页面创建。

认证

认证系统需要用户名 / 密码和 App 帐户标识进行认证。成功之后会返回一个 session 标识，

这个标识用于后续的服务调用，这里的用户名一般指子帐户。在没有任何操作的情况下，Session 标识在 20 分钟后失效。

认证操作命名空间(auth)

认证命名空间用于连接到 USS 系统。调用 login 接口会返回一个 session 标识，这个标识用于后续操作。

login

Login 接口用于登录系统并返回一个 Session 标识，该标识帮定于调用者的 IP。

输入参数

参数	描述	必须/可选
app	用户要登入的应用帐户。	必须
username	帐号。	必须
password	密码。	必须

返回结果

名字	描述	例子
ResponseCode	登录结果,0 为成功,非 0 为失败	0
SessionToken	当前登录用户标识	54592180-7060-4D4B-BC74-2566F4B2F943

Sample REST Request

```
/ws/auth/login/?app=8da051b0-a60f-4c22-a8e0-d9380edafa6f&username=testuser&password=Abc123
```

Sample REST XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<Response>
  <ResponseCode>0</ResponseCode>
  <SessionToken>54592180-7060-4D4B-BC74-2566F4B2F943</SessionToken>
</Response>
```

logout

登出之后系统会删除 session 标识，该 session 标识将不能再被使用。

输入参数

参数	描述	必须/可选
sessionId	要登出的用户 session 标识	必须

返回值

名字	描述	例子
ResponseCode	登出结果，0 为成功，大于 0 为失败	0

Sample REST Request

```
/ws/auth/logout/?sessionId=54592180-7060-4D4B-BC74-2566F4B2F943
```

Sample REST XML Response

```
<Response>  
  <ResponseCode>0</ResponseCode>  
</Response>
```

passwd

该接口改变当前用户密码。

输入参数

参数	描述	必须/可选
app	包含当前用户的应用服务 key	必须
username	帐号	必须
oldpassword	当前密码	必须
newpassword	新密码	必须

返回值

名字	描述	例子
----	----	----

ResponseCode	修改密码返回值，0为成功，大于0为失败	0
--------------	---------------------	---

Sample REST Request

```
/ws/auth/passwd/?app=8da051b0-a60f-4c22-a8e0-d9380edafa6f
&username=testuser&oldpassword=Abc123&newpassword=Def456
```

Sample REST XML Response

```
<Response>
  <ResponseCode>0</ResponseCode>
</Response>
```

文件操作命名空间(usfs)

mkdir

该接口用于在指定位置创建一个或多个空目录。

输入参数

参数	描述	必须/可选
sessionid	有效用户 session 标识	必须
dir	创建文件夹的目录，可传多个参数创建多个文件夹	必须

返回值

名字	描述	例子
ResponseCode	创建文件夹返回值，0为成功	0

Sample REST Request

```
/ws/usfs/mkdir/?sessionid=22b16933-2cd5-40ab-981e-dc9b6cfba06d
&dir=Folders/subfolder/NewFolder/1| Folders/subfolder/NewFolder2
```

Sample REST XML Response

```
<Response>
  <ResponseCode>0</ResponseCode>
</Response>
```


rm

该接口删除一个或多个文件。

输入参数

参数	描述	必须/可选
sessionId	有效用户 session 标识	必须
file	需要删除的文件，可传多个参数以删除多个文件	必须

返回值

名字	描述	例子
ResponseCode	删除文件返回值，0 为成功	0

Sample REST Request

```
/ws/ufsf/rm/?sessionId=22b16933-2cd5-40ab-981e-dc9b6cfba06d
&file=Folders/subfolder/myfile.txt
```

Sample REST XML Response

```
<Response>
  <ResponseCode>0</ResponseCode>
</Response>
```

rmdir

该接口用户删除一个或多个文件夹，如果文件夹里还有文件也会被一起删除。

输入参数

参数	描述	必须/可选
sessionId	有效用户 session 标识	必须
dir	要删除的文件夹，传多个参数删除多个文件夹	必须

返回值

名字	描述	例子
ResponseCode	删除文件夹返回值，0 为成功	0

Sample REST Request

```
/ws/usfs/rmdir/?sessionid=22b16933-2cd5-40ab-981e-dc9b6cfba06d
&dir=Folders/subfolder
```

Sample REST XML Response

```
<Response>
  <ResponseCode>0</ResponseCode>
</Response>
```

ls

该接口用于列出指定文件夹下的文件，它会列出指定文件夹下的文件和目录的一些基本属性，如文件大小，创建时间等。

输入参数

参数	描述	必须/可选
sessionid	有效用户 session 标识	必须
dir	需要列出文件的文件夹路径，如果未指定，将列出当前登录用户的根文件夹	必须

返回值

名字	描述
ResponseCode	列出文件夹的返回值，0 为成功，如果有错误发生则返回下面的错误码
Files	如果返回值为 0，此处为列出的文件列表
ResponseMessage	如果返回值不为 0，此处为错误信息

Sample REST Request

```
/ws/usfs/ls/?sessionid=8da051b0-a60f-4c22-a8e0-d9380edafa6f
&dir=DirTest/F8
```

Sample REST XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<Response>
  <ResponseCode>0</ResponseCode>
  <Files>
    <file>post.htm</file>
    <file>bounce.c</file>
    <dir>nice</dir>
    <dir>folder</dir>
    <file>post.py</file>
    <file>wss-demo.py</file>
  </Files>
</Response>
```

mv

该接口可将文件（文件夹）从一个位置移动到另一个位置，或更改文件名称。

输入参数

参数	描述	必须/可选
sessionid	有效用户 session 标识。	必须
old_path	需要移动的文件地址	必须
new_path	目的文件夹，包含文件本身名字则更改文件名	必须

返回值

名字	描述	例子
ResponseCode	移动文件返回值，0 为成功	0

Sample REST Request

```
/ws/usfs/mv/?sessionid=22b16933-2cd5-40ab-981e-dc9b6cfba06d
&old_path=Folder/subfolder/samplefile.txt&new_path=Folder/subfolder/Dest-
Folder
```

Sample REST XML Response

```
<Response>
  <ResponseCode>0</ResponseCode>
</Response>
```

down

该接口用于获得一个当前的下载链接地址，如果没有 IP 绑定，任何人都可以用这些链接地址在一定时间内下载这些文件。

输入参数

参数	描述	必须/可选
sessionid	有效用户 session 标识。	必须
file	需要下载的文件路径，传多个参数则返回多个下载地址	必须
timeout	返回的链接在多少秒内过期	必须
ip	是否帮定用户 ip	可选

返回值

名字	描述
ResponseCode	获取下载地址返回值，0 为成功，失败则列出的错误码
Host	下载服务器地址
DownloadToken	下载标识，用于文件下载，当过期时间到后该标识失效
DownloadFile	真正的文件地址
ResponseMessage	如果有错误发生，这里则是错误信息

Sample REST Request

```
/ws/usfs/down/?sessionid=8da051b0-a60f-4c22-a8e0-d9380edafa6f
&file=post.html&expiration=300&ip=123.123.123.123
```

Sample REST XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<Response>
```

```

<ResponseCode>0</ResponseCode>
<Host>wss.googlefs.com</Host>
<DownloadToken>72a36c49-ff09-4c37-8ee1-33611d38d563</DownloadToken>
<DownloadFile>/var/wss/yada/1/mds/post.htm</DownloadFile>
</Response>

```

up

该方法用于获得上传文件的上传节点，可指定用户的 IP 并绑定，上传目录地址指定之后远端上传的文件只能上传到该目录，最后可以设定过期时间，一旦过期，上传标识则无效。

输入参数

参数	描述	必须/可选
sessionid	有效用户 session 标识.	必须
ip	允许上传文件的机器 IP，默认为发起请求的机器 IP	可选
dest_dir	上传的目的文件夹	必须
timeout	过期时间	必须

返回值

名字	描述
ResponseCode	上传文件接口返回值，0 为成功
Host	上传的目的主机地址.
UploadToken	上传文件时需要的上传标识
UploadUrl	上传地址
ResponseMessage	如果返回值不为 0，这里为错误信息

Sample REST Request

```

/ws/usfs/up/?sessionid=47c05c1f-b014-4933-
b61f-76ce74a798da&dist_dir=folder&timeout=60

```

Sample REST XML Response

```

<?xml version="1.0" encoding="utf-8"?>

```

```
<Response>
  <ResponseCode>0</ResponseCode>
  <Host>wss.googlefs.com</Host>
  <UploadToken>4be428e9-3fd8-4a52-b56a-bc7a6fc4ed75</UploadToken>
  <UploadUrl>ws/trans/upload/</UploadUrl>
</Response>
```

传输操作命名空间(trans)

Upload

浏览器上传

可通过 “ws/trans/upload/” 将多个文件从浏览器上传。用户可直接上传文件到 wss。

输入参数

参数	描述	必须/可选
uploadtoken	有效上传标识	必须

Html 表单例子

```
<html>

  <body>

    <form ENCTYPE="multipart/form-data"
action="http://wss.googlefs/ws/trans/upload/" method="post">

      Upload Token: <input type="text" name="uploadtoken" /><br />

      File 1: <input type="file" name="uploadFile1"/><br />

      File 2: <input type="file" name="uploadFile2"/><br />

      <input type="submit" value="Upload"/>

    </form>
```

```
</body>
</html>
```

自定义上传

除了用上面得方法外，还可以用其他能直接处理 http 头的库如 Curl 直接对 http 头进行封装然后上传到 USS 服务器。

如：

```
Content-Type: multipart/form-data;\r\n
boundary=-----11332032131482589875648188392\r\n

Content-Length: 12852\r\n

-----11332032131482589875648188392\r\n

Content-Disposition: form-data; name="uploadtoken"\r\n

dce90857-f084-4093-bf41-544bb4ba6cf9\r\n

-----11332032131482589875648188392\r\n

Content-Disposition: form-data; name="uploadFile1"; filename="aa"\r\n

Content-Type: application/octet-stream\r\n

0xf0 0xf1 0xf2 0xf3 0xf4\r\n
```

Content-Disposition 定义了上传必须的参数名。

Content-Type 定义了上传文件的类型

Http Download

下载文件一共分两步，首先需要取得需要下载的文件在 wSS 系统中的真实下载地址然后再去下载该文件。

首先需要取得文件真实地址，这可以用在前面介绍 usfs 命名空间中的 down 方法来取得真实地址，如我们取得的地址为：

```
http://wss.googlefs.com/ws/trans/down/41e4cd3c-6ffc-4424-
bcf3-100417726438/My%20Folder/My%20File%201.txt
```

如果对该下载地址有 IP 绑定，则只有拥有该 IP 的客户端能下载该文件，否则任何客户端都能下载该文件，该下载地址在指定时间内失效。