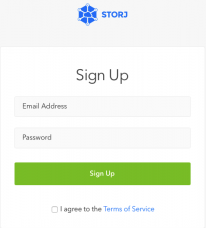
手把手教你玩区块链云存储Storj

[**franklili**](http://www.8btc.com/author/14954)**2016-06-09 10:48 发布在**[**竞争币**](http://www.8btc.com/digital-currency)[**2**](http://www.8btc.com/blockchain-cloud-storj#comment)**2096**

1. 登录[主页](http://www.8btc.com/www.storj.io)：

****

2. **创建账户**



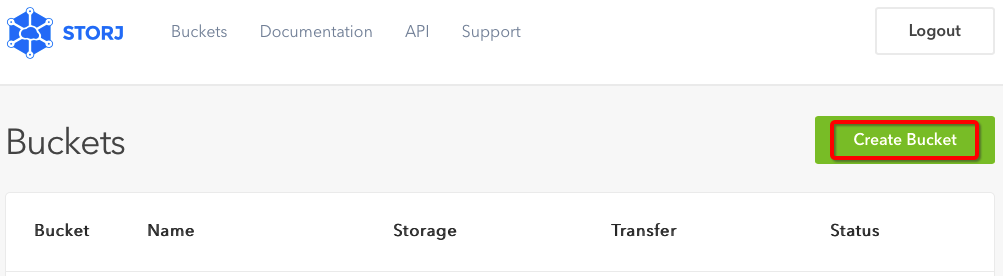
1. 点击[Sign Up](https://app.storj.io/#/signup?_k=iegv1g) 创建一个Storj 账户.
2. 在注册邮箱中的激活邮件中点击链接激活账户。

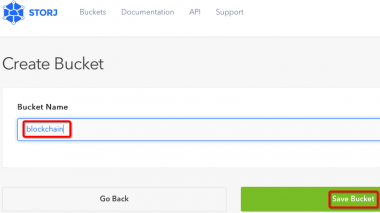
3.登录账户。在主页点击[login](https://app.storj.io/#/?_k=g8j3ms)。

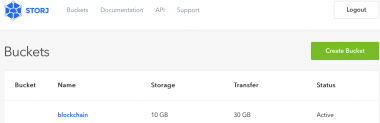
Snip20160607_7

4. 创建文件桶

我们有许多文件要存储在云端，所以我们应该创建一个文件桶，它是文件的逻辑分组，我们能对此分配许可和限制。

1)点击绿色的创建文件桶**Create Bucket**按钮。  
  
填写文件桶名，如 “blockchain”，以后还可以重命名，点击save bucket保存。





文件桶空间有10G，还有2个备份，共计30G。

5. 安装一些工具软件

我们现在有一个 Javascript 和Node.js 客户端与网络和API接口对接.你必须安装 [Node.js](https://nodejs.org/)和 [npm](https://www.npmjs.com/)。我们高度推荐为此安装 [nvm](https://github.com/creationix/nvm) (包含了node和npm)。这个教程使用了OS X EI capitan x64 电脑。

1)安装nvm.打开终端。





 复制下面这句，到终端粘贴，运行。

touch ~/.bash\_profile

再复制下面这句，到终端粘贴，运行。

curl -o- https://raw.githubusercontent.com/creationix/nvm/v0.31.1/install.sh | bash

为了检验安装是否成功，复制下面这句，到终端粘贴，运行。

command -v nvm

如果显示nvm就安装成功了。

2)安装node v4.4.3和npm v2.15.1。复制下面这句，到终端粘贴，运行。

nvm install 4.4.3

复制下面这句，到终端粘贴，运行。

npm install -g storj-bridge-client

复制下面这句storjcli 程序，到终端粘贴，运行，检查能否使用。

storjcli –help

6. 生成**椭圆曲线数字签名算法密钥**

我们能够避免密码闹心，在这个软件背后我们使用超级安全的椭圆曲线数字签名算法（ECDSA）,它将使你的密钥更安全。

你能仅使用用户名和密码访问Storj网络，但是这是不安全的。如果你每次登录到Storj网络用明文传输你的密码，在你的传输通道上偷听的任何人能够拦截你的密码。这种偷听已经很普遍很容易做。

Storj提供了用椭圆曲线数字签名算法密钥替代的选项。用椭圆曲线数字签名算法密钥，你能够为向Storj发送的每个命令生成一个唯一的签名。它验证你的身份，但不能被用于其它操作。

这个教程将展示如何生成椭圆曲线数字签名算法私钥和公钥，并把它们注册到你的Storj账户。

1)生成密钥

为了生成密钥，打开终端，按照顺序运行下面2个命令：

openssl ecparam -genkey -name secp256k1 -noout -outform DER -out private.key  
openssl ec -inform DER -in private.key -noout -text

你将看到像这样的输出：

read EC key  
Private-Key: (256 bit)  
priv:  
8f:42:d3:b4:8b:9c:41:5c:de:fc:c3:cf:07:90:18:  
8e:7b:25:bd:b9:52:d6:93:db:59:30:76:6d:g0:9a:  
fe:0b  
pub:  
04:e5:d5:b8:33:da:56:dd:54:ba:c7:46:fc:1f:35:  
6f:9c:f5:8f:dc:74:a0:11:2d:e7:c7:49:19:eb:d1:  
69:a4:55:9d:fd:ee:ea:62:j2:27:ef:31:f1:0b:d1:  
22:1b:46:62:n4:6b:ad:df:96:69:cf:36:38:4x:60:  
e3:85:10:8c:23  
ASN1 OID: secp256k1

删除公钥的冒号和回车符，它看起来像这样：

04e5d5b833da56dd54bac746fc1f356f9cf58fdc74a0112de7c74919ebd169a4559dfdeeea62j227ef31f10bd1221b4662n46baddf9669cf36384x60e385108c23  
2）生成你的密码的SHA-256哈希值

你需要关联公钥和你的Storj账户。但是第一步，你需要生成你的密码的SHA-256哈希值。DuckDuckGo网站将给你的密码的SHA-256哈希值。例如，如果你的密码是shhdonttell，我能在DuckDuckGo网站查询shhdonttell的[SHA-256哈希值](https://duckduckgo.com/?q=sha-256+shhdonttell)（点击此链接，把地址栏的shhdonttell替换成你的密码，回车）

3) 注册公钥到Storj

你现在有了需要的三样数据：一个在Storj注册的电子邮箱、你的密码的SHA-256哈希值和椭圆曲线数字签名算法公钥。输入这些值到以下的命令中注册到Storj。

curl -u 电子邮箱:你的密码的SHA-256哈希值 -X POST –header ‘Content-Type: application/json’ –header ‘Accept: application/json’ -d ‘{“key”:”公钥”}’ ‘https://api.storj.io/keys’

7.配对账号和设备

为了使未来用密钥登录的过程简单，我们创建了一个配对过程。它将创建一个配对密钥在你的电脑中，并发送公钥给我们。它将用于未来代替用户名和密码的验证。

复制下面这句到终端，粘贴并运行。

storjcli login

系统提示你输入电子邮件和密码。

[...]  > Enter your email address  >  [cats@storj.io](mailto:cats@storj.io)

[...]  > Enter your password  >  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

如果输入正确，系统提示该账户已与本电脑配对。

[info]   This device has been successfully paired.

[原文地址](https://storj.readme.io/)

8.上传和下载文件

现在我们有了一个与服务配对的客户端，我们能用这个工具做些有趣的事情。

1)获得一个猫的图片集

我们喜欢使用自由软件获得免费的猫图片。



2)查询我们的文件桶编号

为了上传我们的猫图片，我们第一步必须获得一个文件桶编号。我们以前创建了一个文件桶，这样让我们查询它的文件桶编号并使用它。请在你的终端输入以下命令：

storjcli listbuckets

系统显示：

[info]   ID: 573b4ce25da55fc8715b4c5a, Name: Test Bucket, Storage: 10, Transfer: 30

为了下一步操作，复制文件桶编号ID。在我这个案例中它是573b4ce25da55fc8715b4c5a。

3)上传一个猫图片

密钥环的密码短语

在第一次你想上传一个文件，系统将提示你输入一个密码短语。它将被用于你本地的文件，并且不会被传送给任何第三方服务或服务器。

为了上传一个猫图片，我们使用命令storjcli uploadfile 文件桶编号 文件路径。本案例使用以下命令（cat.jpg在目前的文件夹下）：

storjcli uploadfile 573b4ce25da55fc8715b4c5a cat.jpg

系统提示输入密码短语解锁你的密钥环：[...]  > Enter your passphrase to unlock your keyring  >  \*\*\*\*\*\*

系统提示正在生成加密密钥：[info]   Generating encryption key…

系统提示正在加密文件cat.jpg： [info]   Encrypting file “cat.jpg”

系统提示加密完成：[info]   Encryption complete!

系统提示正在创建存储令牌： [info]   Creating storage token…

系统提示正在存储文件，挂紧： [info]   Storing file, hang tight!

系统提示加密密钥已保存到密钥环：[info]   Encryption key saved to keyring.

系统提示文件已成功保存到文件桶中： [info]   File successfully stored in bucket.

系统提示文件名、类型、大小、文件编号： [info]   Name: cat.jpg, Type: image/jpeg, Size: 8388624 bytes, ID: 574733fb705cbc353c48eef7

4)下载一个猫图片

为了下载一个猫图片，我们使用命令storjcli downloadfile 文件桶编号 文件编号 文件路径。注意下载文件需要的文件编号列在上传的输出中。本案例使用以下命令：

storjcli downloadfile 573b4ce25da55fc8715b4c5a 574733fb705cbc353c48eef7 cat2.jpg

系统提示输入你的密码短语解锁你的密钥环：[...]  > Enter your passphrase to unlock your keyring  >  \*\*\*\*\*\*

系统提示正在创建收回令牌：[info]   Creating retrieval token…

系统提示重新定位文件指针：[info]   Resolving file pointer…

系统提示从2个通道正在下载文件：[info]   Downloading file from 2 channels…

系统提示收到数据（每个文件片段为64K大小）： [info]   Received 65536 bytes of data

系统提示收到数据：[info]   Received 65536 bytes of data

系统提示收到数据：[info]   Received 65536 bytes of data

系统提示收到数据： [info]   Received 65536 bytes of data

…

系统提示收到数据： [info]   Received 65536 bytes of data

系统提示收到数据：[info]   Received 65536 bytes of data

系统提示收到数据：[info]   Received 65536 bytes of data

系统提示收到数据： [info]   Received 65536 bytes of data

系统提示收到数据： [info]   Received 16 bytes of data

系统提示文件已下载并写到目前文件夹的cat2.jpg：[info]   File downloaded and written to cat2.jpg.

我们已经上传并下载了我们的猫图片到云端！现在如果你以前已经上传了猫图片，并且不知道文件编号，我们能用这个命令查询storjcli listfiles 文件桶编号，以下是本案例的命令：

storjcli listfiles 573b4ce25da55fc8715b4c5a

系统提示文件名、类型、大小、文件编号：[info]   Name: cat.jpg, Type: image/jpeg, Size: 8388624 bytes, ID: 574733fb705cbc353c48eef7