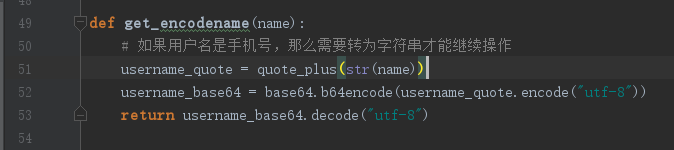
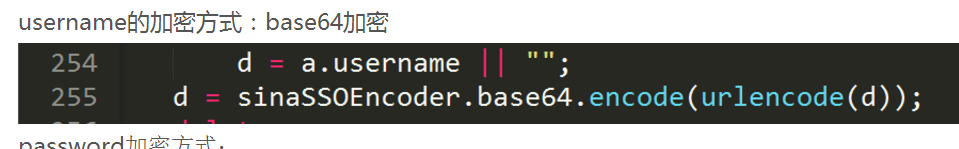
1. 在wblogin\login.py有get\_encodename函数



根据博客http://blog.csdn.net/nobody\_wang/article/details/60957893

中说明的一样。

Username的加密方式为base64加密



所以对一个不知道是否为字符串的变量name，应先对它进行str操作，然后转格式quote\_plus。现在的操作等于urlencode(d)，

The legacy interface:

base64.b64encode(*s*, *altchars=None*)

Encode a byte string using Base64.

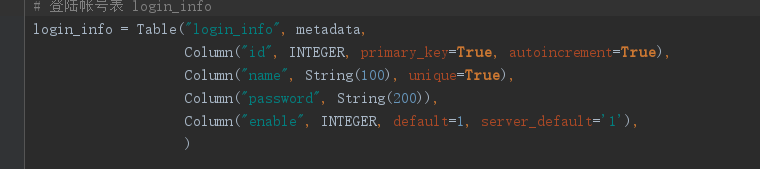
*s* is the string to encode. Optional *altchars* must be a string of at least length 2 (additional characters are ignored) which specifies an alternative alphabet for the + and / characters. This allows an application to e.g. generate URL or filesystem safe Base64 strings. The default is None, for which the standard Base64 alphabet is used.

The encoded byte string is returned.

base64.decode(*input*, *output*)

Decode the contents of the binary *input* file and write the resulting binary data to the *output* file. *input* and *output* must be [file objects](mk:@MSITStore:C:\Users\Wu\AppData\Local\Programs\Python\Python35-32\Doc\Python351.chm::/glossary.html#term-file-object). *input* will be read until input.read() returns an empty bytes object.

其中name是String变量

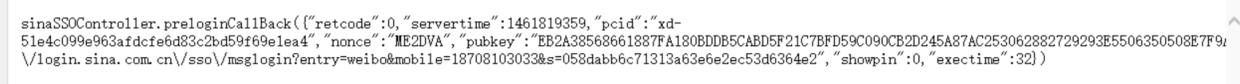


还有一种方法

首先，我们先用正常的账号登陆，具体看会有些什么请求。这里我用的是Http Analyzer抓包(Filders也是一个不错的选择)。下面是正常登陆流程的截图：



现在微博、空间等大型网站在输入用户名后基本都会做编码或者加密处理，这里在用户名输入框输入我的账号，通过抓包工具可以看到服务器会返回一段字符串：



*图2*

这一步就是预登陆过程，同学们可以自己试试。登陆的时候我们需要用到其中的servertime、nonce、pubkey等字段。当然这个不是我自己猜想的，后面的步骤会做说明。

还有一点，就是预登陆的url:

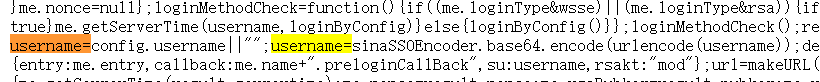
http://login.sina.com.cn/sso/prelogin.php?entry=weibo&callback=sinaSSOController.preloginCallBack&su=&rsakt=mod&checkpin=1&client=ssologin.js(v1.4.18)&\_=1461819359582

这里su的值是自己用户名经过base64编码的值。但可能你们会问我是如何知道的呢，待会儿我会讲到。经过实测，如果我们这里不给su传参数，其实也是可以的。为了最真实的模拟用户登录，我们最好还是带上它的值。

请看图一的第一条js请求http://i.sso.sina.com.cn/js/ssologin.js，同学们可以点进去看，这个就是前面提到的加密用户名和密码等一系列的加密文件了，如果有同学非要问我是怎么找到这个加密文件的，我也只有说：反复抓包，从在浏览器输入weibo.com过后就找js文件请求路径，然后再用代码格式化工具打开，挨着一个一个看，在代码中搜关键字，比如这里我们可以搜"nonce"、“servertime”等，就能找到加密文件了。

打开加密文件我们可以看到加密用户名的代码，在加密js文件中搜索'username'，可以看到有一行代码为:

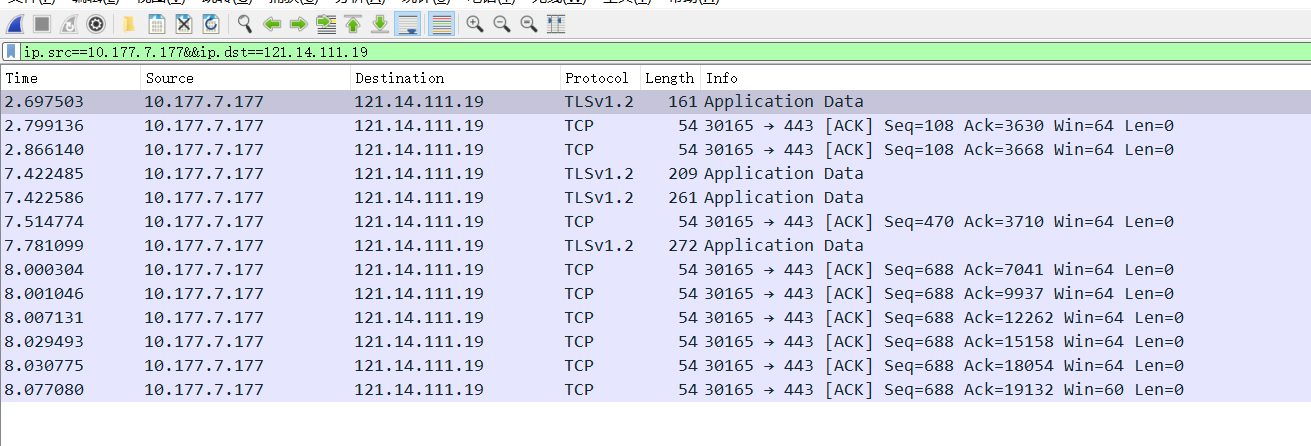
username = sinaSSOEncoder.base64.encode(urlencode(username));



同理password 加密方式和pincodeurl都可以从上边的网址中找到

1. D

现在许多网站不使用http协议，而是使用https。这点通过winshark抓包证实了

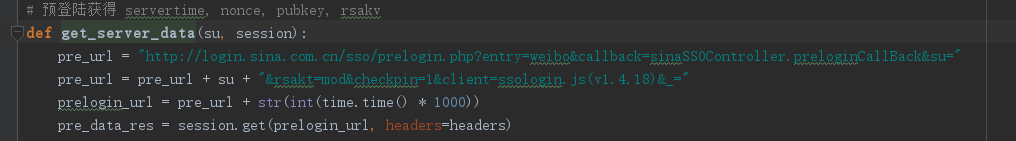
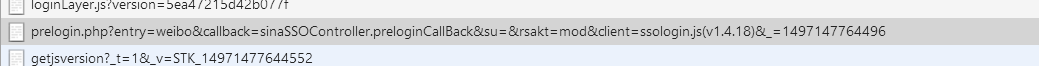


1. 爬虫的架构

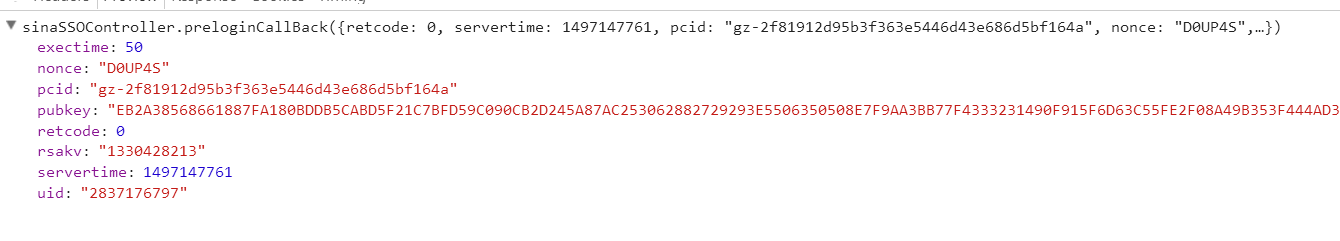
<http://cuiqingcai.com/2433.html>

如上链接所示，应该采用的Scrapy

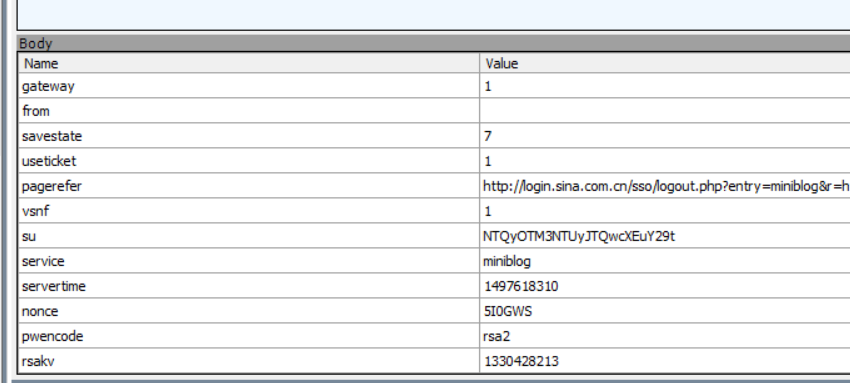
1. 预登陆

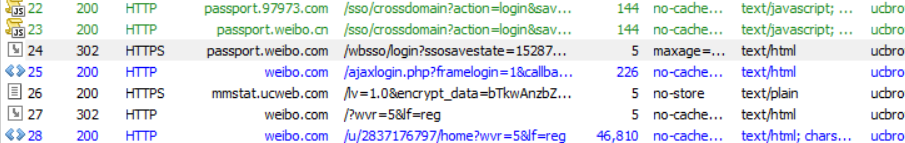


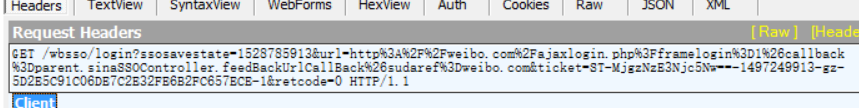
<https://login.sina.com.cn/sso/prelogin.php?entry=weibo&callback=sinaSSOController.preloginCallBack&su=&rsakt=mod&client=ssologin.js(v1.4.18)&_=1497147764496>



然后还会跳转新浪的sso







第一步



上传url的值即为script部分（这里由于不是同一次登录，值不同但结构相同），retcode=0表示登陆成功，然后打开url跳转到下一步

GET /wbsso/login?ssosavestate=1528737060&url=http%3A%2F%2Fweibo.com%2Fajaxlogin.php%3Fframelogin%3D1%26callback%3Dparent.sinaSSOController.feedBackUrlCallBack%26sudaref%3Dweibo.com&ticket=ST-MjgzNzE3Njc5Nw==-1497201060-gz-620316D7A0B98E1FAAB3C2E4B1E7F4E1-1&retcode=0 HTTP/1.1

第二步

<http://weibo.com/ajaxlogin.php?framelogin=1&callback=parent.sinaSSOController.feedBackUrlCallBack&sudaref=weibo.com>



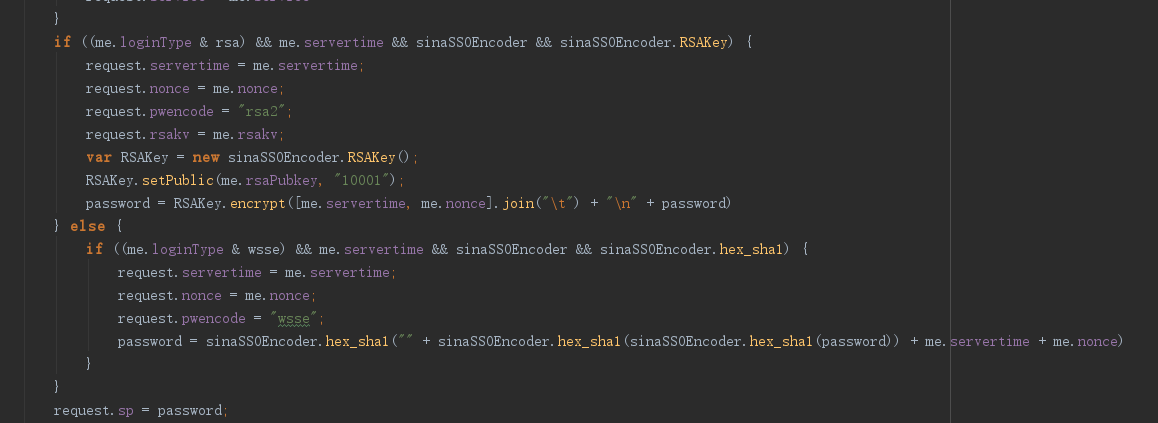
用户登陆成功，返回weibo.com的信息到客户端cookie中，记录了用户登陆信息

返回的cookie



接下来就是新浪的SSO（Single Sign-on）登陆了，就是在同一个顶级域名下，通过载入顶级域名的Cookie，来实现统一登陆。服务器会自动再次访问weibo.com（上图27），因为cookie已经写入，登陆成功服务器返回302，重定向到用户主页面。此后再访问微博中其他人的主页或使用微博的其他应用，服务器便可以通过cookie来检测用户的状态了。

1. 密码加密方式



密码有两种加密方式，一种是rsa加密，一种是进行三次hex加密，先对password进行两次hexsha1加密然后加入servertime和nonce进行第三次加密。又上图能看出