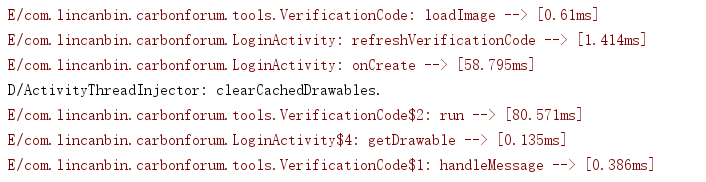
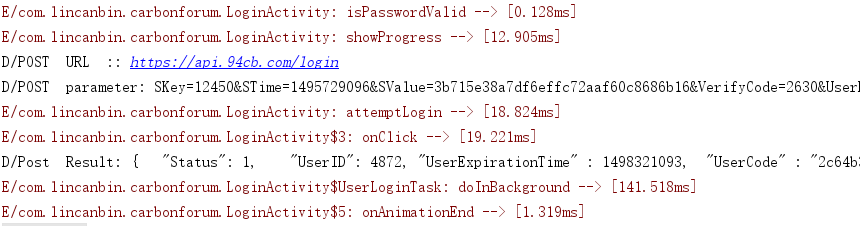
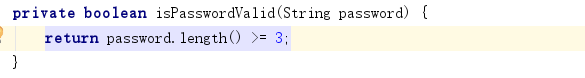
## Android-Carbon-Forum-master



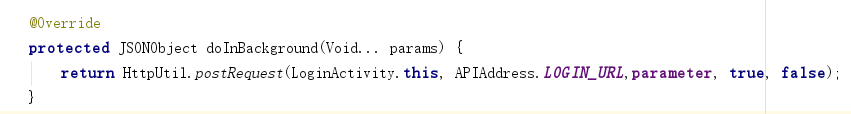
**进入登录界面后，程序首先载入验证码，进行登录准备工作。**



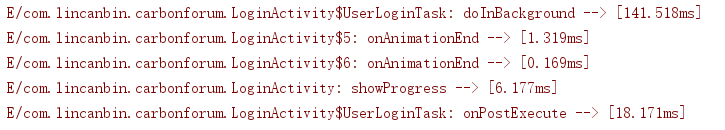
**在输入了用户名、密码和验证码后，程序的isPasswordValid等方法检测密码合法性，**



**程序源码表明密码不能少于三个字符。随后点击登录按钮，从aop返回的数据来看，在验证码的异步读取显示和通过attempLogin方法进行初步检测后，应用程序向服务器端（url::https://api.94cb.com/login）发送数据，数据信息包括UserName和Password。**



**HttpUtil.postRequest方法访问服务器，返回json对象。**



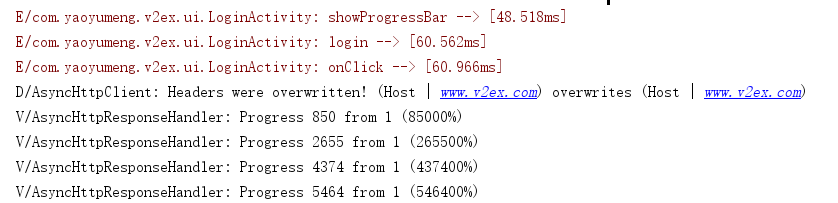
**帐户密码信息发送至服务器端后，服务器端返回相关数据，由onPostExecute方法处理服务器返回的数据。**



**分析源码可以看出用户数据均由服务器端提供，然后向应用程序各个组件广播用户信息，进行配置更新。**

**至此，整个登录过程已经完成。这类应用通过客户端向服务器端发送帐户密码校验，校验成功则由服务端数据库返回相关用户数据，客户端根据返回的数据进行配置。这种权限控制方式较为安全，常见的漏洞是利用sql注入的方式非法访问数据库的数据，然后返回越权访问的数据，客户端就能利用这些非法数据。**

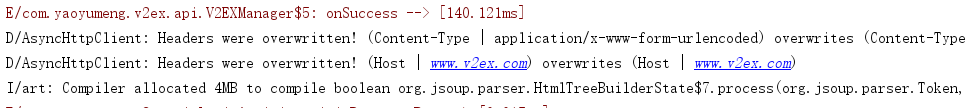
## V2ex-android



**输入用户名和密码之后，应用程序会初始化进度条，从aop截获的数据来看，加载进度条之后进行登录**



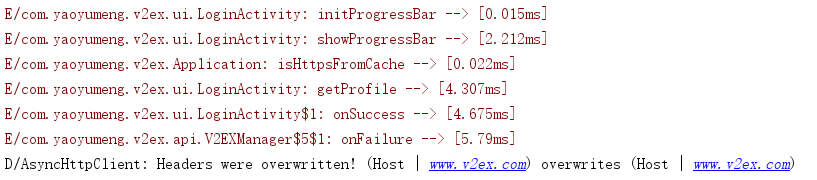
**在登录过程中调用了V2EXManager的相关方法，向服务器端发送用户登录数据，从截获的方法调用数据可以看出，服务器端url为**[**www.v2ex.com**](http://www.v2ex.com)**。**



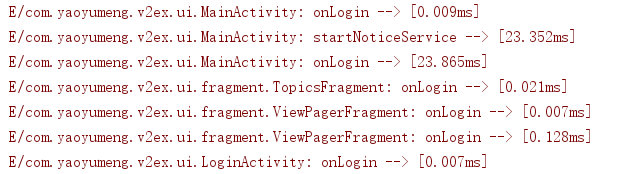
**V2EXManager中的loginWithUsername方法中封装了客户端接收服务器端验证数据以及用户信息的网络处理接口**

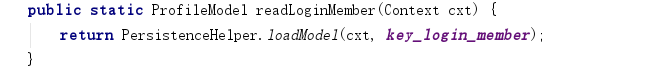


**如果服务器端返回用户名和密码匹配成功的验证消息，客户端会接收服务器端回复的字节流数据，并且解析成用户数据，保存在程序相关类的成员变量中。**

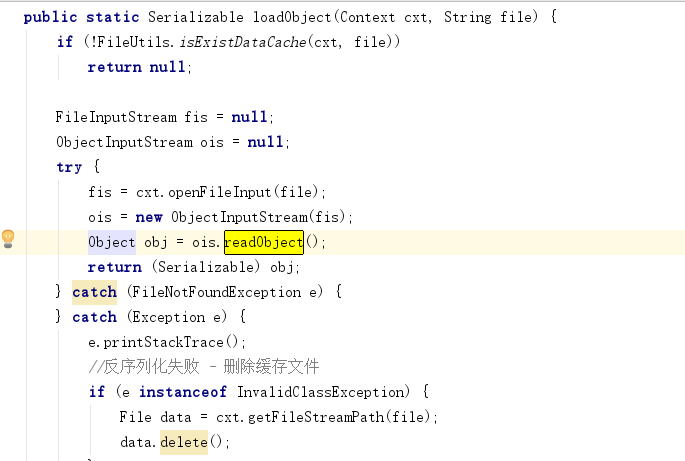


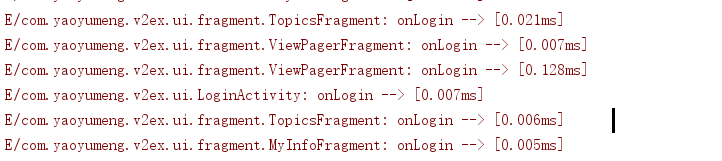
**验证成功后调用getProfile方法开始获取登录用户基本资料，ProfileModel类中封装了获取用户基本资料的url，由服务器端返回用户基本资料。**



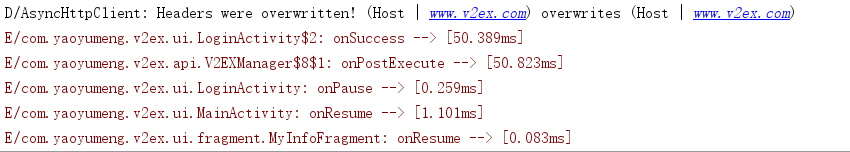


**在得到服务器端的登录确认和用户基本资料后，应用程序开始读取本地缓存中的用户数据，分析源码可知读取用户数据的方式是以用户名作为索引，读取用户数据。**





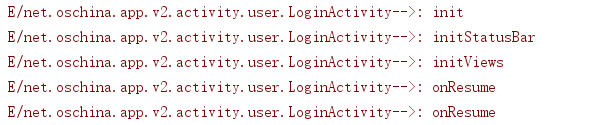
**得到本地的用户配置数据后，应用程序开始配置用户信息，广播通知各个组件用户已经登录，调用TopicsFragment和ViewPageFragment类中相关方法更新视图，加载用户私有的数据，例如主题收藏和特别关注等。**



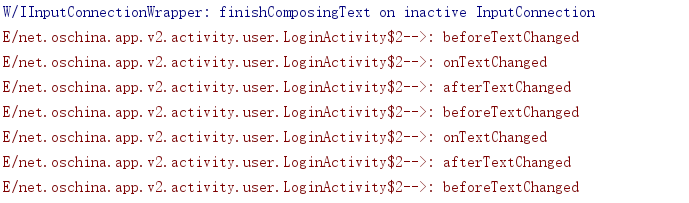
**服务器端的基本用户数据和本地数据都读取完毕后，应用程序进行视图更新，此时可看做登录成功，最后进行一些登录成功后的相关处理。**

**至此，登录完成，应用获取用户数据分为两个过程，第一步由客户端向服务器端发送用户名和密码，等待服务器端的验证。第二步，如果服务器验证成功则根据用户名读取本地缓存数据，应用程序进行配置。第一阶段为访问数据库以获取身份验证，安全防护较为完备，但是第二步读取本地数据没有较好的安全保护。如果在获得服务器端验证成功的信息之后，即将读取本地数据时，利用aop拦截到用户名信息，并对其进行修改，程序继续运行将会获取到其他用户在本地的缓存数据，配置客户端信息。这是一个权限漏洞，可以非法访问其他用户的数据。如果要提高本地用户数据的安全性，读取本地用户数据时可以使用其他方式读取，例如根据用户名建立哈希表，避免用户名直接与本地数据相关联，这种方式可以略微提高数据安全性。**

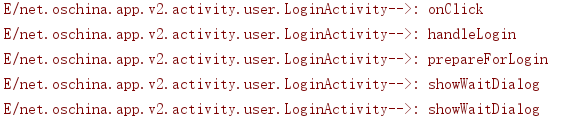
## 哎嘛APP

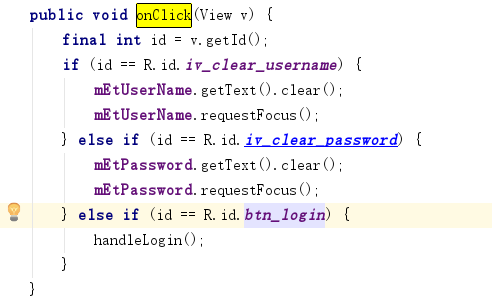


**登录前进行的初始化工作，加载登录界面。**

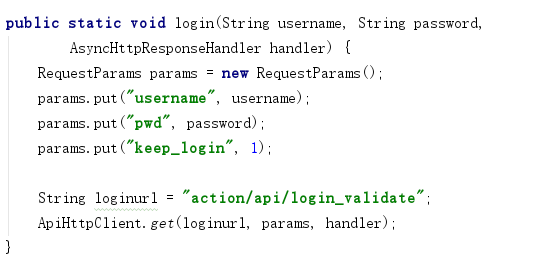


**由程序处理输入，在输入用户名和密码之后点击登录按钮。**

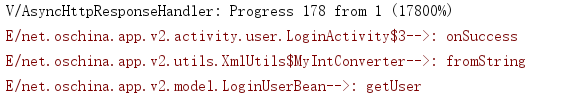




**点击登录后调用handleLogin开始登录，**



**分析源码可以看出由AsynHttpResponseHandler类处理来自服务器端的数据，登录请求的参数包括用户名和密码。**





**在得到服务器端的成功验证后，服务器会返回基本用户信息，客户端获取基本用户信息。**