**X share – the Android App to Share Video**

Yang Xu (0447755) Zhili Guo(0398903)

**1. Introduction**

This application is for the project of course CS 9223. Its main idea is to employ Android SDK and AWS SDK for Android to realize a phone application. The project idea is illustrated in Figure 1. One person may find something funny or fancy that he would like to share with other people. By using our designed application, he can record a video and send that video to Amazon Cloud Servers. Then, once his buddies login in to this application, they could watch his shared video.

Figure 1 Project Idea

**2. Project Description**

The whole application is Android platform based. The application is built by using Android SDK. And Android Emulator is used for debugging. The application employs some Amazon cloud services. And we use AWS SDK for Android to utilize Amazon cloud services. Many services available in AWS SDK for computer are not supported in AWS SDK for Android any more (e.g. Amazon RDS). And some features are totally different (e.g. Authentication process).

The services that we used in the project include the followings: Amazon S3 to store the uploaded video contents, Amazon CloudFront to realize CDN cache for video contents, Amazon SimpleDB to store user-relationship and video contents related information, Amazon Security Token Service to request temporary, limited-privilege credentials for AWS Identity and Access Management (IAM) users or for users that you authenticate (federated users).

**3. Functionality:**

We realize a video new broadcast system in this project. There are bunch of users in the system. All the users have their buddy list. They can leave their friends video news and upload to the Amazon server. When the user log into the system, they can view their accessible video news. In order to realize this, the system should realize the following functions:

User Verification

Username and password will be required for verification. Verified users will be given a temporal Amazon AWS credential for further operation.

Add User

New user can be added into the system. Username and password is required.

Add Friend

A user can add friend relationship with other users. Multiple friends can be added for one time.

Create Video News

A user can create video news. For the video news that he created, he can set it to one of three access options: broadcast it to everyone, broadcast it to his friends, keep it to himself. The video content will be stored in Amazon S3 bucket and be distributed using Amazon Cloudfront service (CDN).

View Video News

Every time a user logs into the system, he can choose to view video news. The user will first get a list of the news and then he can play one from the list.

**4. Implementation and Test**

The development will base on Android simulator, so is the main part of test. Finally, the application will be migrated to a real Android device. Eclipse will be used for the development.

**4.1 Used Amazon Service Description**

**4.1.1 Amazon Security Token Service**

A big difference between Mobile platform and PC platform when using Amazon cloud service is the authentication process. On mobile applications, authenticating users with a token vending machine is highly recommended instead of traditional way which is fixed Amazon Credential with in the application source code.

There are two ways to achieve requesting temporary credential from Amazon with Amazon Security Token Service (Amazon STS). The difference between these two is whether you want give the token to anonymous identity or a federated user. We chose to use the one that require identity confirmation. Figure 2 shows the flow of temporary security credentials.

Our own Token Vending Machine is hosted using Amazon Elastic Beanstalk service. With correct permission configuration, federated users will not be allowed to access the particular Amazon SimpleDB database domain which contains their request history, but they are allowed to access other database domain that contains video information and friend list information and etc.

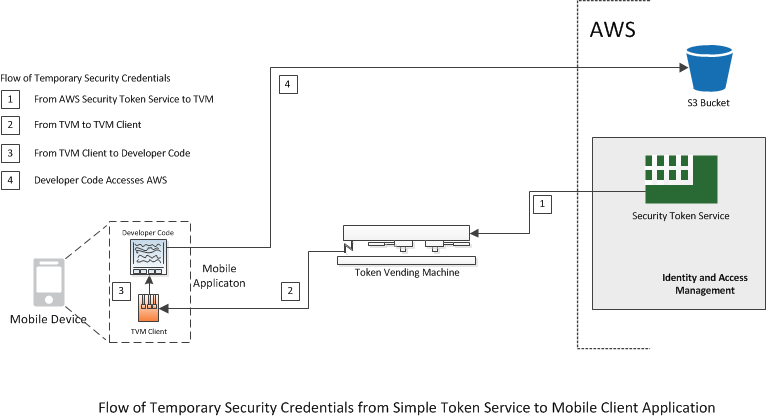


Figure 2 Flow of Temporary Security Credentials

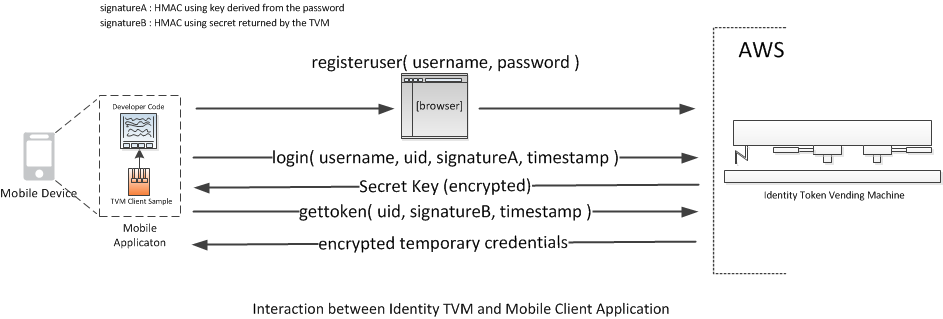


Figure 3 Interaction between identity TVM and Mobile client Application

**4.1.2 Other Amazon Service**

Besides security token services, we also use other Amazon services. The video contents will be uploaded to Amazon S3 bucket. And those contents will be set to the appropriate access control policies. To deliver video contents to client side more quickly, we also use CloudFront service to do the CDN cache. The realization of these services in Mobile platform is almost the same as the realization in computer. Since Mobile platform doesn’t support Amazon RDS service any more, we have to turn to use Amazon SimpleDB service, which is very similar to Google Datastore.

**4.2 Mobile Client Layouts:**

Figure 4 Mobile Client Layouts

Mobile client layouts are illustrated in Figure 4. We mainly have 9 pages: Main, Terminal, Buddy List, Register, News List, Record News, Add Friend, View News and Upload News. The functions for these layouts are listed below:

A. Main: This is the first layout when you open that software. You can choose to login in or register your account

B. Terminal: Here, the main functions provided by the system are listed. You can choose to add new friend, upload video news or view video news.

C. Buddy List: You can view the list of your friends here.

D. Register: This is the layout for your register. After that, you can login in to the system by using username and password.

E. News List: Here you can see the list of all video news. You can choose one of them to view. When you choose one, you enter into layout View News.

F. Record News: In this layout, you record the video news that you want to share. After recording that, you can view the recorded video.

G. Add Friend: In this layout, you can add your new friend name here.

H. View News: Here, you can view your video news.

I. Upload News: You need to add the news title before you upload it to the server. And you have three options to do the video sharing: broadcast it to everyone, broadcast it to his friends, keep it to himself.

**4.3 Test**

Because our system has video related function, this makes our development kind of complicated. We need to do our test in the real phone machine.

After all the functions have been developed, the final software runs on a Google Nexus 4 android phone. Eight users are added into the system and relative friend relationship. Also, video messages are generated by all the users to do sharing.

**5. Final Result**

We have two video demos “register.avi” and “record video.avi” for this part.

In the video “register.avi”, we register a new account “review1”. When the new account first login in, he has no friend at all. He could select to view all users in the system. And he adds several people as his buddies. Then, some users appear in his friend list.

In the video “record video.avi”, we login in to the system using account name “guotest”. Then we choose to record one video about 14s duration. We set its name to be “phone and other stuff” and choose to broadcast this recorded video to every user. Then I use another phone to login in as account “review1”. Video News “phone and other stuff” would show in my news list. And I could see that video when I touch the video. Also, other users would also see that video news, which is not show in our demo video. This shows that we realize the video sharing function.

**6. Future Work**

In this project, we realize a video sharing system by using Amazon cloud services. In that design, we consider user would only record video for a very short time. It is kind of like twitter, where the message is very short. In the future, we could add more functions to our app, like the “unfriend” function, selecting existed video to share, deleting uploaded video, blocking particular user’s sharing, etc. We also want to optimize database function, like batch put attributes to the database. And we want to develop UI to make it fancier.