

Projet Android MBDS 2018/2019

Objet : Création d'une Application de messagerie chiffrée de bout à bout

Environnement utilisé pour nos tests

Nous avons testé l'application sur des mobiles Android avec un API supérieur ou égale à 23. Ainsi que sur des émulateurs sous API 28.

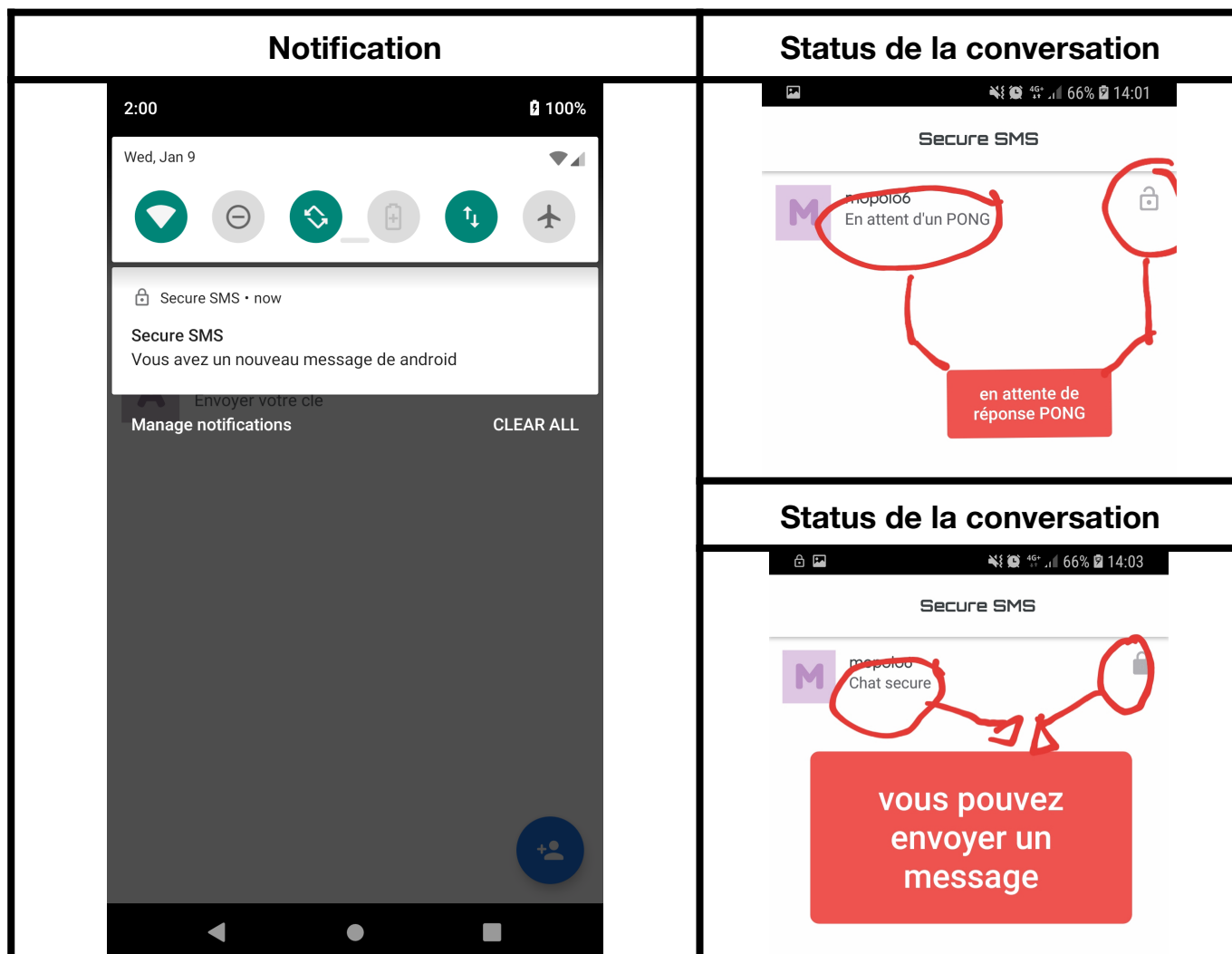
Scénarios

Au démarrage de l'application nous avons une première activité qui se lance (SplashScreen) dans cette activité on vérifie si il existe une session c'est-à-dire si nous avons un access_token (on nous sauvegarde notre access_token dans un SharedPreferences) est valide (URL : <http://baobab.tokidev.fr/api/validate>), si il y a une session alors on démarre notre service (MyServiceFetchMessage.java) ensuite on est redirigé vers la page principale (MainActivity). Si il y a pas de session on est redirigé vers la page de Login.

Pour l'envoi de messages (ALICE) il faut ajouter un nouveau contact (un contact qui possède aussi un compte). Lors de l'ajout d'un nouveau contact on génère une paire de clef (RSA) et un PING lui est automatiquement envoyé. De l'autre côté le nouveau contact (BOB) reçoit un message PING et il lui suffit de cliquer sur le BUTTON pour générer une clef secrète (AES) et renvoyer un PONG. Une fois le PING et le PONG envoyés les deux utilisateurs (ALICE et BOB) peuvent communiquer.

Le service tournant en arrière plan nous notifie de l'arrivée d'un nouveau message.

Les Ecrans



SplashScreen

Splash screen
Activity



L'interface Login

Secure
SMS

Username

Password

LOGIN

No account yet ? Create one

L'interface Register

Secure SMS

Username

Password

Confirm password

CREATE ACCOUNT

Already a member? Login

L'interface ContactList

Secure SMS

- nana3
Chat secure
- nana4
Chat secure
- android
Envoyer votre cle

Le service est toujours en cours d'exécution

L'interface Messagerie

android

Salut

2019-01-09T14:05:29.031+01:00

bonjour

2019-01-09T13:05:41Z

test ok

2019-01-09T14:05:54.611+01:00

ok 5/5

2019-01-09T13:06:12Z

Le service est toujours en cours d'exécution

Saisir le message

SEND

android

Envoyer PONG

SEND PONG

android pub key SEND_PONG

L'interface AjouterContact

Add Contact

Username

ADD USER

Affichage mode paysage

Secure SMS

mopolo6

- mopolo6
Chat secure

ok 5/5

2019-01-09T14:06:10.933+01:00

allo

2019-01-09T14:08:17.085+01:00

azertyuiop

Le service est toujours en cours d'exécution

Saisir le message

SEND

Les Services

Le service est démarré lors de la connexion d'un utilisateur et fait des appel serveur toute les 5 secondes pour récupérer les nouveaux messages.

/!\ Voir la classe **MyServiceFetchMessage.java** dans le package **service**

Intent

Pour le partage il faut au préalable déjà avoir parlé avec une personne c'est-à-dire que l'action PING-PONG est déjà été réalisé et que la conversation est sécurisé.

Modèle de données

Ici nous avons utilisé 2 tables pour notre application.

User : pour sauvegarder les utilisateurs que nous ajoutons comme contact

Message : pour sauvegarder les messages en local

User		
uid	INT	identifiant d'un utilisateur
username	TEXT	le nom d'un utilisateur (unique)
thumbnail	TEXT	une image (pour l'instant représenté par une couleur aléatoire)
status	TEXT	pour notifier si la clé a été transmise et si la conversation est sécurisé
publicKey	TEXT	représente la clé publique généré par celui qui envoi le message en premier
privateKey	TEXT	représente la clé privée généré par celui qui envoi le message en premier
aesKey	TEXT	représente la clé Secrete généré par celui qui reçoit une demande de conversation

Message	
uid	INT
author	TEXT
authorKey	TEXT
message	TEXT
alreadyReturned	BOOLEAN
currentUser	BOOLEAN

Arborescence des Classes java



Membre :

- BOUKOU Grace
- Askia Mohamed