

# Spécification technique de besoins

<b>Version</b>	1.4
<b>Date</b>	11 février 2013
<b>Rédigé par</b>	Giovanni HUET, Romain PIGNARD
<b>Relu par</b>	Florian GUILBERT, Emmanuel MOCQUET

## MISES À JOUR

Version	Date	Modifications réalisées
0.1	26/11/2013	Création
0.2	02/01/2013	Ajout des cas d'utilisation
0.3	30/01/2013	Modifications mineures
1.0	31/01/2013	Relecture
1.1	06/02/2013	Corrections après rdv client
1.2	06/02/2013	Corrections
1.3	07/02/2013	Corrections (modifications C10)
1.4	11/02/2013	Modifications (priorité F-FI-30)

## Table des matières

<b>1</b>	<b>Objet</b>	<b>4</b>
<b>2</b>	<b>Documents applicables et de référence</b>	<b>4</b>
<b>3</b>	<b>Terminologie et sigles utilisés</b>	<b>4</b>
<b>4</b>	<b>Exigences fonctionnelles</b>	<b>6</b>
4.1	Présentation de la mission du produit logiciel . . . . .	6
4.2	Communications sécurisées . . . . .	6
4.3	Génération de nombres aléatoire . . . . .	6
4.4	Déblocage de la carte . . . . .	7
4.5	Transmission de données . . . . .	8
4.6	Chiffrement/Déchiffrement . . . . .	9
4.7	Signature/Vérification de données . . . . .	11
4.8	Administration des cartes . . . . .	12
<b>5</b>	<b>Exigences opérationnelles</b>	<b>13</b>
<b>6</b>	<b>Exigences d'interface</b>	<b>13</b>
<b>7</b>	<b>Exigences de qualité</b>	<b>13</b>
<b>8</b>	<b>Exigences de réalisation</b>	<b>13</b>

## 1 Objet

Ce projet propose la mise en place de solutions cryptographiques pour sécuriser les données qu'un utilisateur place sur un réseau social au moyen d'authentifications fortes.

Il s'agirait donc ici de développer une extension pour le logiciel Mozilla Firefox permettant à l'utilisateur de gérer le chiffrement de ses données sur le réseau social Facebook. Cette extension utilisera une application Java pour assurer les traitements lourds. Pour gérer l'authentification forte, cette application dialoguera avec une carte à puce qui contiendra les données sensibles de l'utilisateur (login/mot de passe), clef privée, ...

Le dialogue avec cette carte à puce se fera par l'intermédiaire d'un client Java.

Ce projet est une composition de deux sous-projets :

- Étude et mise en œuvre de solutions d'authentifications et de signatures par cartes à puce, proposé par Magali BARDET ;
- Solutions cryptographiques pour les réseaux sociaux, proposé par Ayoub OTMANI ;

Dans ce document, nous présentons le sous-projet SC (pour SmartCards) utilisé par l'entité FaceCrypt du sous-projet SSN (Secure Social Network), cette utilisation est adaptable à d'autres situations.

## 2 Documents applicables et de référence

- Manuel d'utilisation ;
- Tutoriel d'installation ;
- cartes-a-puce.pdf, le sujet du projet.

## 3 Terminologie et sigles utilisés

**CdR** : Cahier de Recettes ;

**AdR** : Analyse des Risques ;

**DAL** : Document d'Architecture Logicielle ;

**PdD** : Plan de développement ;

**STB** : Spécification Technique de Besoins ;

**SC** : *SmartCard*, relatif au sous-projet sur les cartes à puce ;

**SSN** : *Secure Social Network*, relatif au sous-projet sur Facebook ;

**FaceCrypt** : Application Java gérant les traitements lourds (chiffrement/déchiffrement) de l'extension et étant en relation avec la carte à puce ;

**IHM** : Interface Homme-Machine, (interface graphique) ;

**Utilisateur** : entité (humain ou programme) interagissant avec ce sous-projet ;

**Système** : ce sous-projet ;

**Sécurisé** : ce terme sous-entend un chiffrement des données et une vérification de leur intégrité ;

**SoftCard** : application effectuant le relais entre la carte et FaceCrypt ;

**Extension** : programme incorporé dans le navigateur ;

**Aléatoire** : les résultats suivent une distribution de probabilité uniforme ;

**Pseudo-aléatoire** : les résultats sont indistingables en temps polynomial d'une distribution de probabilité uniforme ;

**PRNG** : (Pseudo Random Number Generator) générateur de nombres pseudo-aléatoires ;

**RNG** : (Random Number Generator) générateur de nombres aléatoires ;

**PIN** : (Personal Identification Number) code servant à authentifier l'utilisateur ;

**PUK** : (Personal Unlock Key) code servant à débloquer la carte quand trop de codes PIN erronés ont été entrés.

## 4 Exigences fonctionnelles

### 4.1 Présentation de la mission du produit logiciel

Référence	Fonctionnalité Globale	Acteur	Priorité
F-Gl-10	Génération de nombres aléatoires	SmartCard, SoftCard, Utilisateur	Indispensable
F-Gl-20	Déblocage de la carte ( <i>via</i> authentification par code PIN et PUK)	SmartCard, SoftCard, Utilisateur	Indispensable
F-Gl-30	Transmission de données	SmartCard, SoftCard, Utilisateur	Indispensable
F-Gl-40	Chiffrement/déchiffrement de données	SmartCard, SoftCard, Utilisateur	Indispensable
F-Gl-50	Signature/Vérification de données	SmartCard, SoftCard, Utilisateur	Indispensable
F-Gl-60	Administration des cartes	Administrateur	Secondaire

### 4.2 Communications sécurisées

La communication sécurisée est une précondition pour tous les prochains cas d'utilisation. Les composants établissent des secrets cryptographiques pour des tunnels sécurisés après s'être mutuellement authentifiés.

### 4.3 Génération de nombres aléatoire

SoftCard doit être capable d'utiliser le générateur de nombres aléatoire de SmartCard.

Nom : C1	Génération de nombres aléatoire	
Acteurs concernés	SmartCard, SoftCard	
Description	SmartCard génère un nombre aléatoire de la taille demandée	
Préconditions		
Evénements déclenchants	Demande de SoftCard	
Conditions d'arrêt	SmartCard renvoie un nombre aléatoire à SoftCard	
Description du flot d'événements principal :		
Acteur(s)		Système
1. SoftCard demande à SmartCard un nombre aléatoire de longueur donnée. 3. SoftCard récupère le nombre.		2. SmartCard génère un nombre grâce au RNG intégré et le renvoie à SoftCard.
Flots secondaires :		
Flots d'exceptions :		

#### 4.4 Déblocage de la carte

Pour utiliser une SmartCard, l'utilisateur devra entrer son code PIN afin de « débloquer » celle-ci.

Nom : C2		Authentification d'un utilisateur/Déblocage de la carte	
Acteurs concernés		Utilisateur, SmartCard, SoftCard	
Description		Le système authentifie l'utilisateur grâce au code PIN contenu sur SmartCard.	
Préconditions		L'utilisateur a une carte valide et connaît le code PIN, SoftCard et SmartCard sont authentifiés.	
Evénements déclenchants		SoftCard a besoin d'utiliser SmartCard.	
Conditions d'arrêt		L'authentification a réussi ou a échoué.	
Description du flot d'événements principal :			
Acteur(s)		Système	
1. L'utilisateur insère la carte et demande à SoftCard d'utiliser SmartCard 3. L'utilisateur tape le code PIN		2. SoftCard demande le code PIN à l'utilisateur 4. SoftCard envoie le code PIN à SmartCard 5. SmartCard vérifie le code PIN et passe en état « débloqué » pendant 30 minutes s'il est correct 6. SoftCard informe l'utilisateur du résultat	
Flots secondaires :		7. SoftCard redemande le code PIN à l'utilisateur.	
Flots d'exceptions :		7 bis. L'utilisateur a tapé 3 mauvais codes. La carte se verrouille.	

Si l'utilisateur a entré plusieurs mauvais codes PIN, la carte se bloque et il peut la débloquent avec le code PUK.

Nom : C3		Déblocage de la carte par code PUK	
Acteurs concernés		Utilisateur, SmartCard, SoftCard	
Description		L'utilisateur débloque sa carte avec le code PUK	
Préconditions		L'utilisateur a une carte valide mais bloquée, il connaît le code PUK, SoftCard et SmartCard sont authentifiés	
Evénements déclenchants		SoftCard a besoin d'utiliser SmartCard	
Conditions d'arrêt		La carte est déverrouillée ou inutilisable	
Description du flot d'événements principal :			
Acteur(s)		Système	
1. L'utilisateur insère la carte et demande à SoftCard d'utiliser SmartCard 3. L'utilisateur tape le code PUK		2. SoftCard indique que le code PIN est verrouillé et que la carte doit être débloquée par le code PUK 4. SoftCard envoie le code PUK à SmartCard 5. SmartCard vérifie le code PUK, génère un nouveau code PIN aléatoire et le renvoie à SoftCard. 6. SoftCard informe l'utilisateur de son nouveau code PIN	
Flots secondaires :		7. SoftCard redemande le code PUK à l'utilisateur.	
Flots d'exceptions :		7 bis. L'utilisateur a tapé 3 mauvais codes PUK. La carte se verrouille définitivement et doit être remplacée.	

#### 4.5 Transmission de données

SmartCard contient des données propres à l'utilisateur, elle doit alors permettre la transmission de ces données. Ici, c'est FaceCrypt qui souhaite récupérer les données de l'utilisateur.



Nom : C4	Transmission login/mot de passe au SocialNetwork
Acteurs concernés	SmartCard, SoftCard, FaceCrypt, Social Network
Description	SmartCard transmet le couple login/mdp à FaceCrypt
Préconditions	SmartCard est débloquée avec le bon code PIN, SoftCard et FaceCrypt sont authentifiés. SoftCard et SmartCard sont authentifiés
Evénements déclenchants	L'utilisateur veut se connecter sur SocialNetwork
Conditions d'arrêt	L'utilisateur est connecté auprès de SocialNetwork.
Description du flot d'événements principal :	
Acteur(s)	Système
1. FaceCrypt demande à SoftCard le login/mdp du Social Network 5. FaceCrypt envoie au Social Network le login/mdp.	2. SoftCard demande à SmartCard le login/mdp du Social Network. 3. SmartCard envoie le login/mdp à SoftCard 4. SoftCard envoie à FaceCrypt le login/mdp du Social Network.
Flots secondaires :	
Flots d'exceptions :	1. Authentification invalide

#### 4.6 Chiffrement/Déchiffrement

La carte procède au chiffrement et au déchiffrement de la clef de chiffrement symétrique pour chaque message avec les clés asymétriques adéquates.

Pour le chiffrement, SoftCard utilise la clef publique du destinataire. Pour le déchiffrement, SmartCard utilise la clef privée stockée en mémoire sur la carte.

Cet exemple concerne FaceCrypt mais est aisément adaptable à tout autre système.

Nom : C5	Déchiffrement de données
Acteurs concernés	SmartCard, SoftCard, FaceCrypt
Description	SmartCard déchiffre des données, envoyées par FaceCrypt avec la clé privée de chiffrement stockée sur la carte.
Préconditions	SmartCard est débloquée avec le bon code PIN, Authentification entre SoftCard et FaceCrypt. SoftCard et SmartCard sont authentifiés
Evénements déclenchants	Demande de FaceCrypt
Conditions d'arrêt	SmartCard renvoie un résultat du déchiffrement à SoftCard qui transmet à FaceCrypt
Description du flot d'événements principal :	
Acteur(s)	Système
1. FaceCrypt envoie des données chiffrées à SoftCard 5. FaceCrypt récupère les données déchiffrées	2. SoftCard transmet les données chiffrées à SmartCard 3. SmartCard déchiffre les données avec la clé privée de chiffrement stockée et renvoie le résultat du déchiffrement à SoftCard 4. SoftCard transmet le résultat à FaceCrypt.
Flots secondaires :	
Flots d'exceptions :	Une erreur de déchiffrement provoque l'envoi d'un message d'erreur, ne permettant pas de connaître sa cause

La demande de chiffrement de données vient de l'utilisateur ou de toute autre application tierce, dans le cas d'une utilisation plus générale.

Nom : C6	Chiffrement de données
Acteurs concernés	Utilisateur, SoftCard
Description	SoftCard chiffre les données avec la clef publique du destinataire
Préconditions	
Evénements déclenchants	Demande de l'utilisateur
Conditions d'arrêt	SoftCard renvoie les données chiffrées
Description du flot d'événements principal :	
Acteur(s)	Système
1. L'utilisateur transmet des données à chiffrer à SoftCard, ainsi que la clef publique du destinataire ;	2. SoftCard reçoit les données en clair et les chiffre avec la clef publique du destinataire ; 3. SoftCard renvoie les données chiffrées à l'utilisateur.
Flots secondaires :	
Flots d'exceptions :	

#### 4.7 Signature/Vérification de données

<b>Nom : C7</b>		<b>Signature de données</b>	
<b>Acteurs concernés</b>		SoftCard, SmartCard	
<b>Description</b>		SmartCard utilise la clef privée pour signer des données fournies par SoftCard	
<b>Préconditions</b>		SmartCard et SoftCard sont mutuellement authentifiées et l'utilisateur a déverrouillé la carte	
<b>Evénements déclenchants</b>		Demande de l'utilisateur	
<b>Conditions d'arrêt</b>		Les données sont signées ou l'utilisateur annule	
<b>Description du flot d'événements principal :</b>			
<b>Acteur(s)</b>		<b>Système</b>	
1. SoftCard envoie les données qu'il faut signer à SmartCard. 3. SoftCard récupère la signature.		2. SmartCard signe les données et envoie la signature a SoftCard	
<b>Flots secondaires :</b>			
<b>Flots d'exceptions :</b>			

Nom : C8		Vérification de données
Acteurs concernés		SoftCard
Description		SoftCard utilise la clef publique pour vérifier des données signées
Préconditions		
Evénements déclenchants		Demande de l'utilisateur
Conditions d'arrêt		Les données sont vérifiées, ou non
Description du flot d'événements principal :		
Acteur(s)		Système
2. SoftCard vérifie les données et renvoie "vrai" si la signature et son équivalent en clair correspondent ; faux sinon.		
Flots secondaires :		
Flots d'exceptions :		

#### 4.8 Administration des cartes

Nom : C9		Initialisation d'une carte	
Acteurs concernés		Utilisateur, SmartCard, SoftCard, FaceCrypt	
Description		L'utilisateur paramètre la carte avec ses identifiants	
Préconditions		SmartCard et SoftCard sont mutuellement authentifiées et l'utilisateur a déverrouillé la carte	
Evénements déclenchants		Demande de l'utilisateur	
Conditions d'arrêt		Les applets ont été installées sur la carte. Les clefs ont été initialisées, l'identifiant et le mot de passe ont été stockés	
Description du flot d'événements principal :			
Acteur(s)		Système	
1. L'utilisateur entre ses identifiants Facebook et choisit ou non de générer un nouveau mot de passe. 4. FaceCrypt enregistre le nouveau mot de passe et envoie la confirmation de modification à SoftCard.		2. Un utilitaire installe les applets et les clefs de chiffrement et de signature sur la carte, génère un nouveau mot de passe si besoin, stocke temporairement les identifiants Facebook (anciens et nouveaux) et crée un code PIN, qu'il transmet à l'utilisateur. 3. SoftCard envoie le nouveau mot de passe à FaceCrypt. 5. SoftCard transmet la confirmation à SmartCard qui stocke définitivement le mot de passe et efface l'ancien.	
Flots secondaires :			
Flots d'exceptions :		Si le compte n'existe pas, une erreur est générée.	

Nom : C10	Modification du mot de passe
Acteurs concernés	Utilisateur, SmartCard, SoftCard, FaceCrypt
Description	L'utilisateur souhaite changer son mot de passe Facebook
Préconditions	SmartCard et SoftCard sont mutuellement authentifiées et l'utilisateur a déverrouillé la carte
Evénements déclenchants	Demande de l'utilisateur
Conditions d'arrêt	Le mot de passe a été modifié et enregistré sur Facebook et sur la carte.
Description du flot d'événements principal :	
Acteur(s)	Système
1. L'utilisateur choisit l'option permettant de générer un nouveau mot de passe. 3. FaceCrypt enregistre la modification et envoie une confirmation du changement de mot de passe à SoftCard.	2. SoftCard transmet le mot de passe à SmartCard qui le stocke temporairement. Il notifie ensuite FaceCrypt de la modification. 4. SoftCard transmet la confirmation à SmartCard qui stocke définitivement le mot de passe et efface l'ancien.

Flots secondaires :	
Flots d'exceptions :	Si le compte n'existe pas, une erreur est générée.

## 5 Exigences opérationnelles

Référence	Fonctionnalité	Priorité
F-FO-10	Le système fonctionne	Indispensable
F-FO-20	La caractéristique aléatoire d'un nombre généré (par le générateur aléatoire) est vérifiable	Indispensable

## 6 Exigences d'interface

Référence	Fonctionnalité	Priorité
F-FI-10	SoftCard communique de manière sécurisée avec FaceCrypt	Indispensable
F-FI-20	SoftCard présente une interface pour demander le code PIN	Indispensable
F-FI-30	SoftCard communique de manière sécurisée avec SmartCard	Indispensable

## 7 Exigences de qualité

Référence	Fonctionnalité	Priorité
F-FQ-10	Le système sera livré pour le 01 mars 2013	Indispensable
F-FQ-20	Une documentation de développement est fournie	Indispensable
F-FQ-30	Le système est adaptable	Important
F-FQ-40	L'utilisation d'une fonction cryptographique ne doit pas ralentir le système	Indispensable
F-FQ-50	Authentification mutuelle entre toutes les entités (SmartCard, SoftCard et FaceCrypt)	Indispensable

## 8 Exigences de réalisation

Référence	Fonctionnalité	Priorité
F-FR-10	Un SDK et un manuel sont fournis	Indispensable