Authentification forte par opérateur de réseau mobile à l'usage des utilisateurs de services web

Jean-Jacques SCHWARTZMANN, Olivier GRUSON
Orange Labs, Networks and Carriers
FT/OLNC/RD/MAPS/STT
Caen, France
jeanjacques.schwartzmann@orange.com
olivier.gruson@orange.com

Gaël GOURMELEN
Orange Labs, Networks and Carriers
FT/OLNC/RD/MAPS/MEP
Lannion, France
gael.gourmelen@orange.com

Résumé – **La multiplicité des services web sur lesquels les internautes doivent s'identifier entraine une inflation du nombre de mots de passe à mémoriser, d'où la nécessité de recourir à une authentification unique. D'autre part, les services en ligne à forte valeur ajoutée nécessitent un degré d'authentification bien supérieur à celui procuré par le simple couple identifiant/mot de passe. Dans ce document, nous présentons les atouts des opérateurs de téléphonie mobiles et les mécanismes qu'ils sont à même de mettre en œuvre pour fournir de manière simple des services d'authentification forte aux utilisateurs des services web distants.**

Mots clés : authentification forte, smartphone, SIM, UICC, USSD, OTP, OATH, SSO, OpenID, SAML, OAuth

I. INTRODUCTION

Nous utilisons, à titre personnel ou professionnel, de multiples services en ligne tels que portails administratifs, services bancaires, magasins en ligne, sites d'enchères, webmails, réseaux sociaux, auxquels nous confions des données confidentielles. A la première connexion à chacun de ces services, nous devons nous inscrire en indiquant nos données d'identité (réelle ou fictive selon la nature du service), et choisir un mot de passe qui servira à nous authentifier lors de nos visites ultérieures. Les utilisateurs sont donc amenés à mémoriser de nombreux couples identifiant/mot de passe pour accéder à ces différents services, dont la gestion de manière simple et sécurisée devient vite problématique : afin de pouvoir les mémoriser, les mots de passe choisis sont souvent trop courts et trop simples, et les mêmes sont souvent réutilisés pour des services différents, ce qui va à l'encontre des « bonnes pratiques ». Celles-ci consistent en effet à choisir des mots de passe dit « forts », à la fois suffisamment longs et insensibles aux attaques par dictionnaires, et à les renouveler périodiquement. Il est en outre préconisé de choisir des mots de passe distincts pour chaque service, afin d'éviter que la compromission d'un mot de passe n'ouvre pas l'accès à tous les services de l'utilisateur.

Un remède à la multiplicité des mots de passe est le SSO (Single Sign On) ou authentification unique, consistant à faire en sorte que le client n'ait à s'authentifier qu'en une seule fois à l'aide d'un mot de passe unique, par l'intermédiaire d'un fournisseur d'identité que tous les autres services utilisent pour l'authentification du client. Ces solutions reposent sur des standards comme OpenID [1], SAML[2] ou OAuth[3]. Le mot de passe unique doit répondre aux règles de « bonnes pratiques » énoncées plus haut, car en cas de compromission, l'attaquant dispose des « clés du château » et c'est l'ensemble des services de l'utilisateur qui est compromis. Il est toutefois possible d'éviter ce risque en renforçant l'authentification à l'aide d'un facteur supplémentaire tel qu'un élément matériel détenu par l'utilisateur, ou bien encore une authentification biométrique.

Cette nécessité d'une authentification renforcée se fait sentir d'autant plus que les utilisateurs ont maintenant à leur disposition des services en ligne à forte valeur ajoutée : ainsi le concept du « Cloud Computing », en vogue dans les entreprises depuis la fin des années 2000, s'ouvre maintenant aux particuliers. Il consiste – même si le NIST en donne une définition plus restrictive [4] – à déléguer les traitements informatiques et le stockage des données vers des serveurs distants, accessibles via un simple navigateur web. Ainsi, de plus en plus d'utilisateurs, du simple particulier aux grandes entreprises, confient à des services web externes un grand nombre de données sensibles dont il est nécessaire de sécuriser le stockage et la transmission. A cet effet, le protocole Transport Layer Security (TLS), permettant la mise en place de sessions web sécurisées (HTTPS), est maintenant couramment déployé par les serveurs, afin de garantir leur authenticité vis-à-vis du client, ainsi que la confidentialité et l'intégrité des données de celui-ci. En plus de l'authentification du serveur, le protocole TLS permet également l'authentification du client, on parle alors d'authentification mutuelle. Pour cela, le client doit posséder un certificat électronique, ce qui nécessite une infrastructure PKI (Public Key Infrastructure) impliquant la génération des clés publiques et privées, la certification des clés publiques, la publication des certificats, leur révocation éventuelle, la gestion de listes de révocation (CRL, Certificate Revocation List), et éventuellement la distribution de supports physiques (carte à puce ou clé USB) destinés à stocker les certificats, pour gagner en mobilité et sécuriser le stockage. Bien que considérée comme très sûre, la PKI souffre de sa trop grande complexité de déploiement : en France, lors de la mise en place en 2002 de la télé-déclaration des revenus, l'administration fiscale imposait à l'utilisateur la création d'un certificat avec lequel il pouvait s'authentifier sur son espace personnel et signer électroniquement sa déclaration, mais jugé trop complexe à créer, et donc de nature à freiner l'usage de la

télé-déclaration, ce certificat client a été rendu facultatif en 2009.

Compte-tenu de la multiplicité des services web à forte valeur ajoutée, de leur audience croissante, et de la nécessité de protéger efficacement l'accès aux données qu'ils hébergent, il existe un besoin croissant de systèmes d'authentification à la fois sûrs et simples à déployer et à utiliser, reposant sur l'authentification forte et le SSO.

Dans la section 2 de cet article, nous présentons quelques solutions de SSO et d'authentification forte couramment utilisées pour l'accès aux services web. Dans la section 3 nous présentons les atouts des opérateurs de réseaux mobiles pour proposer des systèmes d'authentification forte et unique, et dans la section 4 nous décrivons les schémas d'authentification forte que nous avons étudiés et développés.

II. ETAT DES LIEUX DE L'AUTHENTIFICATION SUR LE WEB

A. *Fédération d'identitées et authentification unique*

L'authentification unique ou SSO a pour but de permettre à l'utilisateur de s'authentifier en une seule fois (généralement avec un couple identifiant/mot de passe) pour accéder à l'ensemble de ses services. Dans le cadre d'un véritable SSO, les services délèguent l'authentification de leurs utilisateurs à une entité appelée fournisseur d'identité. Ce mécanisme repose alors sur une première phase de fédération d'identités permettant à l'utilisateur de lier son identité chez son fournisseur d'identité à ses différents comptes de services. Lors de cette première étape, le fournisseur de service peut proposer à l'utilisateur de faire le lien avec un compte existant ou en créer un à la volée sur la base des informations transmises par le fournisseur d'identité. Dans le souci du respect de la vie privée de l'utilisateur, celui-ci doit pouvoir contrôler quelles informations sont transmises au fournisseur de service : c'est la première des « 7 lois de l'identité » [5] élaborées par Kim Cameron, architecte de l'identité chez Microsoft.

Les protocoles de SSO et fédération d'identités comme OpenID, SAML (ou même OAuth appliqué à ce contexte) reposent tous sur les mêmes principes, décrits à la Figure 1. Toute la logique d'authentification est prise en charge par le fournisseur d'identité ce qui peut aussi faciliter l'introduction de mécanismes d'authentification forte pour l'accès à un ensemble de services sensibles (mise en œuvre uniquement par cette entité spécialisée).

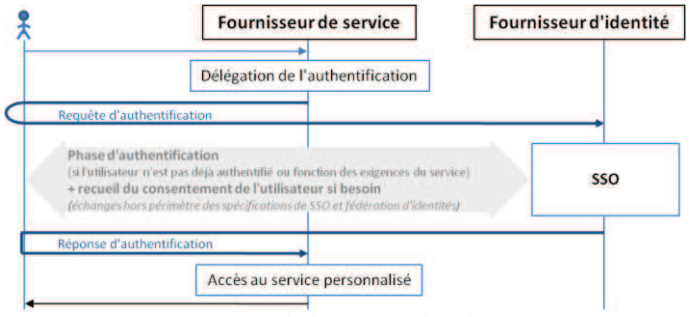

Figure 1. Fédération d'identités et authentification unique

B. *Authentification forte*

1) Authentification multi-facteurs : L'authentification multi-facteurs consiste à utiliser plus de facteurs qu'un simple identifiant et mot de passe pour s'authentifier. Les facteurs sont en général répartis dans les quatre classes suivantes :

- Ce que je sais : mot de passe, code PIN,
- Ce que je possède : carte à puce, téléphone mobile, clé USB,
- Ce que je suis (biométrie morphologique) : empreinte digitale, iris de l'œil, reconnaissance faciale, forme de la main, voix,
- Ce que je sais faire (biométrie comportementale) : dynamique de frappe au clavier, signature manuscrite dynamique, démarche, voix.

L'appartenance à une classe peut être discutée : par exemple, la voix peut-être considérée comme un facteur appartenant à la fois aux deux classes biométrie morphologique et comportementale.

L'authentification à deux facteurs est un cas particulier de l'authentification multi-facteurs. On considère qu'une authentification forte est réalisée si elle utilise au moins deux facteurs provenant de deux classes différentes. Il faut bien entendu qu'elle soit également non-rejouable – ce qui impose de sécuriser les canaux de transmission, à l'aide par exemple du protocole TLS – et que chacun des facteurs considéré individuellement soit suffisamment robuste.

La combinaison des deux premières classes de facteurs – la connaissance d'un mot de passe ou d'un code PIN et la possession d'un élément matériel – est la plus utilisée actuellement pour réaliser une authentification forte, pour les raisons suivantes :

- Le couple identifiant/mot de passe est le facteur d'authentification le plus largement répandu, il est bien accepté par les utilisateurs et reste le plus simple à mettre en place,
- Les supports physiques tels que cartes à puces, clés USB, sont également répandus depuis longtemps auprès des utilisateurs pour réaliser une authentification,
- Les technologies biométriques morphologiques comme l'empreinte digitale, la reconnaissance faciale ou iridienne sont parfois vues comme intrusives et nécessitent souvent des capteurs spécifiques, d'où un surcoût pour l'utilisateur,
- Les technologies biométriques comportementales comme la dynamique de frappe au clavier sont mieux acceptées mais sont relativement nouvelles et donc souvent considérées comme pas assez matures pour une authentification fiable.

2) One Time Password : Certains supports physiques générateurs de jetons d'authentification – et couramment désignés par abus de langage « tokens » – comme SecurID de la société RSA Security fonctionnent sur le principe du mot de passe à usage unique (One Time Password ou OTP), sur la base d'un secret partagé. L'OTP permet à l'utilisateur de prouver qu'il est bien en possession du token et évite les attaques par rejeu. Il existe deux types d'OTP : les OTP synchrones qui nécessitent une synchronisation avec le serveur et les OTP asynchrones. Les OTP synchrones sont en général basés sur le temps ou sur un compteur voire les deux. Les OTP asynchrones répondent quant à eux à un challenge envoyé par le serveur.

Figure 2. SecurID de RSA Security affichant un OTP à 6 digits

Le flux d'authentification de SecurID par OTP synchrone est représenté à la Figure 3 :

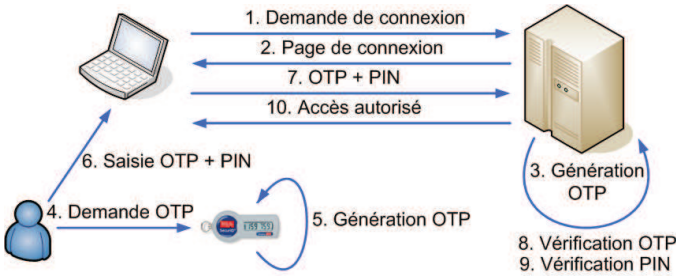

Figure 3. Authentification par OTP synchrone

Afin d'obtenir une authentification forte, l'OTP généré par le token est complété par un code PIN personnel de 4 à 8 caractères, qui constitue le second facteur d'authentification.

3) SMS/OTP : Le téléphone mobile, de par sa large diffusion auprès du public, est utilisé par quelques acteurs comme facteur matériel pour réaliser une authentification forte. C'est le cas de Google qui propose à ses utilisateurs la « validation en deux étapes ». Il s'agit de compléter l'authentification classique basée sur le couple identifiant/mot de passe, par la saisie d'un OTP généré par le serveur et que l'utilisateur reçoit via un SMS (Short Message Service) sur son téléphone. La Figure 4 montre la cinématique d'une telle authentification, appelée « SMS/OTP » :

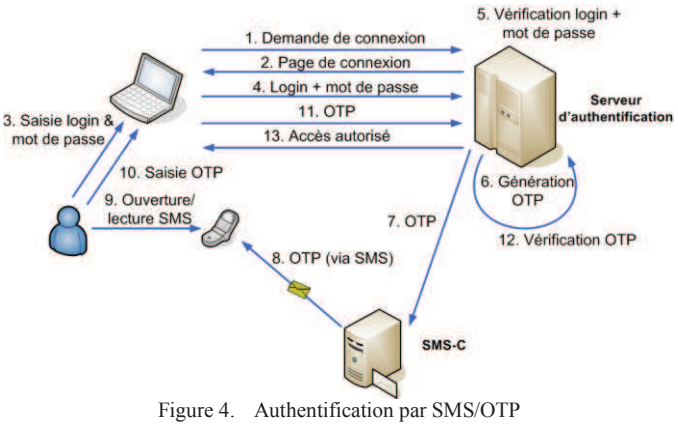

Figure 4. Authentification par SMS/OTP

6. Après vérification de l'identifiant et du mot de passe, (étape 1 à 5), le serveur génère un OTP,
7. puis l'envoie au SMS-C,
8. qui le retransmet via SMS vers le téléphone de l'utilisateur qui souhaite s'authentifier.
9. L'utilisateur ouvre le SMS et lit l'OTP,
10. et le saisit dans l'interface d'authentification.
11. L'OTP est renvoyé au serveur, prouvant ainsi que l'utilisateur est bien en possession de son téléphone.

SMS/OTP est également parfois utilisé par certaines banques comme BNP Paribas, HSBC ou La Banque Postale, pour authentifier le porteur de la carte bancaire lors d'un achat en ligne selon le protocole 3D-Secure.

Le mécanisme SMS/OTP a été la cible d'une attaque du type « homme du milieu », découverte en septembre 2010 et désignée « Man-in-the-mobile » [6]. Le préalable est la compromission de l'ordinateur de la victime par le malware ZeuS. Le scénario est le suivant :

1. Le malware détecte des actions de navigation sur les sites web utilisant SMS/OTP, ce qui permet à l'attaquant de récupérer l'identifiant et le mot de passe de la victime, ainsi que son numéro de téléphone mobile.
2. L'attaquant envoie un SMS vers le téléphone de la victime pour l'inciter à installer un malware sur celui-ci. Il faut noter que seuls les smartphones sont vulnérables.
3. L'attaquant se connecte ensuite avec l'identifiant et le mot de passe de la victime sur un site utilisant SMS/OTP, ce qui provoque l'envoi d'un OTP par le serveur via SMS vers le mobile de la victime.
4. Le malware présent sur le mobile de la victime intercepte le SMS, récupère l'OTP qu'il contient et le transmet à l'attaquant via une connexion data. Le SMS est ensuite détruit par le malware, de sorte que la victime ne se rend pas compte de l'attaque.
5. L'attaquant peut ensuite renseigner l'OTP sur la page de connexion pour être authentifié à la place de la victime.

4) Google Authenticator : Google propose également une application pour les smartphones appelée « Google Authenticator », destinée à générer le mot de passe à usage unique. Ce système, désigné sous le terme de « soft-token », permet par rapport à la solution SMS/OTP, d'éviter l'envoi du SMS par le serveur et se rapproche plus du fonctionnement d'un token tel que SecurID. Google Authenticator est conforme à l'initiative OATH [7] et génère un OTP synchrone basé sur le temps selon la méthode TOTP [8].

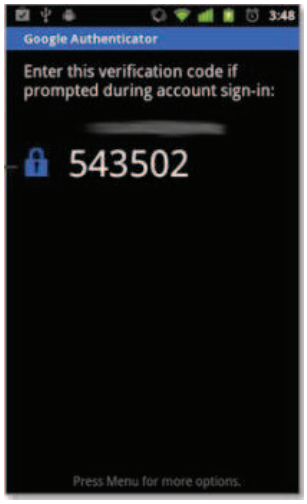

Figure 5. OTP généré par Google Authenticator sur un smartphone Android

Le flux d'authentification est décrit plus loin dans ce document à la Figure 6.

III. LES ATOUTS DES OPERATEURS DE RESEAUX MOBILES

L'audience des acteurs "Over The Top" de l'internet, tels que Google, Facebook, Yahoo ou Microsoft, revendiquant un grand nombre d'utilisateurs inscrits, leur permet de se positionner légitimement comme fournisseurs d'identité. Cependant, aucune de ces sociétés n'est actuellement capable de vérifier et de garantir les données saisies par les internautes, ceux-ci pouvant tout à fait déclarer une identité fictive pour utiliser les services, afin par exemple de préserver leur vie privée, mais cette identité déclarée ne peut être utilisée dans tous les contextes. L'état ou un organisme sous son autorité, voire une société commerciale offrant un service réel, sont les seuls à pouvoir garantir l'authenticité des informations d'identité de l'utilisateur. C'est notamment le cas des opérateurs de téléphonie mobile qui sont à même de disposer pour chacun de leurs abonnés de données d'identité fiables et réutilisables dans tous les contextes.

Chaque client mobile possède un élément de sécurité qui est l'UICC (Universal Integrated Circuit Card) appelée couramment mais improprement carte SIM (Subcriber Identify Module), ce terme désignant en fait l'application GSM résidant sur la carte. Avec 5 milliards d'abonnements mobiles dans le monde, l'UICC associée au terminal mobile constitue le token idéal utilisable par tous en tant que facteur matériel pour réaliser une authentification forte. En effet, les capacités cryptographiques de l'UICC, et la possibilité d'y stocker une clé secrète de manière sécurisée sous le contrôle d'un code PIN, lui permettent de calculer un OTP avec une sécurité équivalente à celle d'un token dédié. D'autre part, le numéro unique IMSI (International Mobile Subscriber Identity) inscrit dans la SIM permet d'authentifier celle-ci de manière sûre sur le réseau mobile, via le mécanisme d'authentification GSM.

En résumé, les opportunités des opérateurs mobiles pour se positionner comme fournisseurs d'identité et proposer des solutions d'authentification forte sont les suivantes :

- Un management d'identité fiable et pointu qui permet de se positionner comme fournisseur d'identité, notamment vis-à-vis de fournisseurs de services requérant l'identité réelle, mais il est essentiel que l'opérateur soit le garant du respect de la vie privée, en permettant à l'utilisateur de garder le contrôle des informations qu'il entend divulguer au fournisseur de service,
- Le téléphone mobile associé à l'UICC, de part sa très large diffusion, constitue un facteur matériel de choix dans la mise en place d'une authentification forte, sans surcoût pour l'utilisateur,
- Les capacités cryptographiques de l'UICC et sa capacité à stocker des secrets de manière sécurisée permettent de prouver de façon sûre que l'utilisateur est bien en sa possession,
- De même, l'attachement de la SIM au réseau fournit un mécanisme d'authentification que l'opérateur peut utiliser pour obtenir la preuve que l'utilisateur est bien en possession de son mobile. Ce mécanisme d'authentification réseau permet également de réaliser une authentification forte sur des canaux séparés pour la fourniture de chacun des facteurs : on parle alors d'authentification multicanaux, le mécanisme SMS/OTP évoqué plus haut en est un exemple.

IV. AUTHENTIFICATION FORTE

Dans le cadre d'une étude interne sur les moyens d'authentification forte utilisables pour l'accès aux offres de Cloud Computing d'Orange, nous avons élaboré quelques mécanismes que nous allons détailler dans ce paragraphe. Une authentification forte n'a de sens que si elle permet d'augmenter la confiance de l'utilisateur, par ailleurs elle ne doit entrainer pour celui-ci ni complexité d'utilisation ni surcoût supplémentaires. Un état de l'art préliminaire nous a permis d'établir les caractéristiques d'une telle authentification :

- Authentification à (au moins) deux facteurs de classes différentes, voire multicanaux,
- Utilisation de mécanismes cryptographiques mis en œuvre dans un support physique,
- Le mécanisme mis en œuvre doit permettre de créer la confiance : pour cela l'utilisateur doit avoir le sentiment que l'authentification forte lui assure qu'il est le seul à pouvoir se connecter à son compte, d'autre part il doit pouvoir garder le contrôle des informations

d'identités qu'il divulgue au fournisseur de service. Pour ces raisons, le mécanisme d'authentification ne peut être complètement transparent pour l'utilisateur,

- Pour autant, il ne doit pas engendrer de complexité accrue : la simplicité d'usage doit être du même niveau que la simple saisie du couple identifiant/mot de passe, ce qui exclut les manipulations multiples et les saisies de champs supplémentaires,
- Pas de surcoût, ce qui exclut les clés USB, cartes à puces ou tokens matériels dédiés, ou bien les capteurs biométriques spécifiques,
- Simplicité de déploiement et mécanisme d'enrôlement rapide et aisé, ce qui exclut l'utilisation d'une infrastructure PKI et les certificats X.509,
- Conformité à des standards ouverts, pour permettre l'interopérabilité.

1) Google Authenticator : Nous nous sommes intéressés à la solution Google Authenticator que Google propose aux utilisateurs de ses services Google Apps, et que nous avons évoqué au paragraphe II.B.4. Cette solution, reposant sur la génération d'un OTP synchrone par le smartphone de l'utilisateur, est décrite à la Figure 6 :

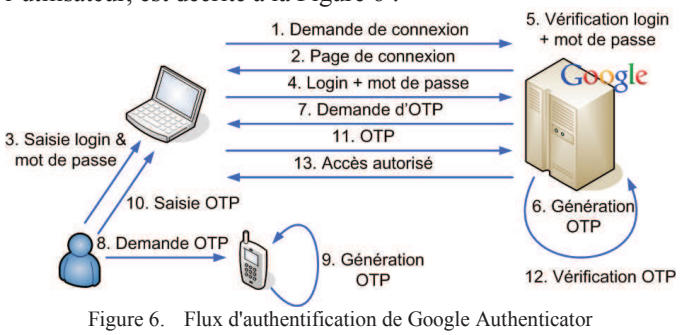

Figure 6. Flux d'authentification de Google Authenticator

Google Authenticator est annoncé conforme à OATH [7] (initiative for Open AuTHentication) dont le but est de définir une architecture standard d'authentification forte s'appuyant sur des standards ouverts préexistants, et regroupant entre autres des acteurs tels que Gemalto, SanDisk, VeriSign, Entrust, ou encore Symantec. OATH définit trois méthodes de génération d'OTP :

- TOTP (Time based One Time Password RFC 6238) [8]
- HOTP (HMAC based One Time Password RFC 4226) [9]
- OCRA (OATH Challenge-Response Algorithm RFC 6287) [10]

Google Authenticator génère un OTP synchrone selon la méthode TOTP et l'affiche sur le smartphone de l'utilisateur, qui devra le saisir ensuite dans le champ prévu sur la page de connexion, comme montré à la Figure 7 :

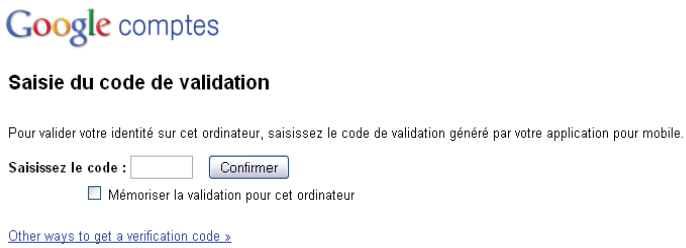

Figure 7. Saisie de l'OTP sur la page de connexion de Google

L'enrôlement est plutôt aisé : l'utilisateur doit préalablement charger l'application sur son smartphone, puis au moment de l'inscription, il scanne un code QR qui va permettre le chargement de la clé secrète dans le smartphone. Pour confirmer l'enrôlement, l'utilisateur doit saisir sur la page d'inscription un code à 6 chiffres affiché sur le smartphone après le scan du code QR.

Cette solution répond à certains de nos critères : elle est de nature à augmenter la confiance puisque le smartphone est nécessaire pour s'authentifier, elle n'induit pas de surcoût pour l'utilisateur, elle est simple à utiliser et l'enrôlement est plutôt aisé, et enfin elle est conforme à OATH dont la vocation est de devenir un standard.

Cependant, la génération de l'OTP ne repose pas sur un élément matériel sécurisé : le calcul est effectué par l'application installée sur le smartphone, avec la clé secrète stockée en mémoire de masse. Si le mécanisme utilisé garantit bien que l'utilisateur est en possession du couple identifiant/mot de passe et de la clé secrète, il ne prouve pas de manière formelle que l'OTP a été généré par le smartphone, en raison de la relative vulnérabilité du stockage de la clé. L'authentification est donc basée sur deux facteurs de connaissance, le mot de passe de connexion et la clé secrète, plutôt que sur la combinaison d'un facteur de connaissance et d'un facteur matériel. L'authentification n'est donc pas aussi « forte » qu'il y parait à première vue.

Notre approche a donc consisté à élaborer des schémas d'authentification dérivés de Google Authenticator, en y incluant des mécanismes pouvant établir de manière sûre la possession du terminal mobile par l'utilisateur.

2) Calcul de l'OTP par l'UICC : Cette approche conserve le flux d'authentification précédent, mais la clé secrète est stockée dans une zone protégée de l'UICC, qui génère l'OTP. Cette méthode garantit que l'utilisateur est bien en possession de son mobile au moment de l'authentification. La génération de l'OTP est effectuée par une application SIM Toolkit, ce qui rend possible l'utilisation d'un terminal mobile de base au lieu du smartphone. La Figure 8 montre le flux d'authentification, avec un OTP synchrone généré par l'algorithme TOTP :

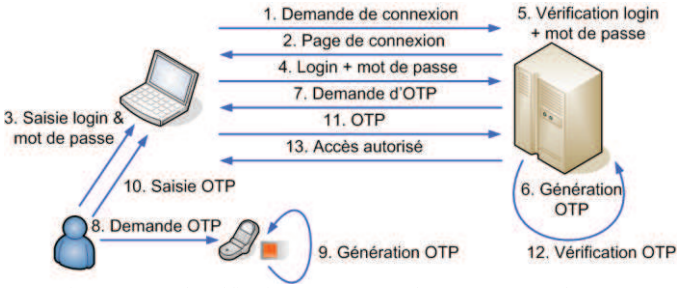

Figure 8. Authentification par OTP synchrone généré par l'UICC

La Figure 9 représente un flux d'authentification utilisant un OTP asynchrone généré à partir d'un challenge fourni par le serveur, selon l'algorithme OCRA. Ce flux est un peu plus complexe que le précédent pour l'utilisateur, puisque qu'il nécessite la saisie du challenge sur le terminal mobile.

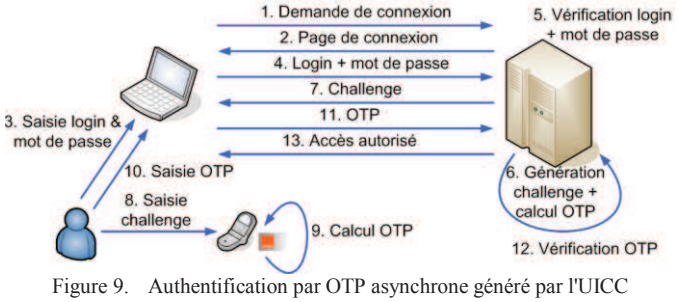

Figure 9. Authentification par OTP asynchrone généré par l'UICC

3) Utilisation du canal USSD : Comme nous l'avons vu au paragraphe III, l'attachement de la SIM au réseau fournit un mécanisme d'authentification que l'on peut utiliser pour obtenir la preuve que l'utilisateur est bien en possession de son mobile. Le mécanisme SMS/OTP décrit au paragraphe II.B.3 utilise ainsi l'authentification GSM : si l'utilisateur a reçu l'OTP envoyé via SMS sur son mobile, c'est qu'il est bien en possession de celui-ci, et surtout de la carte SIM qui y est insérée. Les principaux inconvénients de SMS/OTP sont le le délai d'acheminement non garanti, et la vulnérabilité à certaines attaques du type « Man-in-the-mobile ».

Notre approche consiste à utiliser le canal USSD (Unstructured Supplementary Service Data) en lieu et place du SMS. Comme le SMS, il s'agit une fonctionnalité des réseaux GSM, utilisable par l'ensemble du parc des téléphones mobiles, et destinée à transmettre de l'information sous forme texte sur le canal de signalisation. Le volume d'information maximal contenu dans une trame USSD est du même ordre de grandeur (182 caractères) que le SMS (160 caractères), mais contrairement au SMS, aucune donnée n'est stockée dans la chaine de transmission et de traitement USSD. L'USSD est généralement associé à des services de consultation, interactifs ou non. On y accède en composant sur le clavier du téléphone un « shortcode » commençant et finissant par un ou plusieurs caractères '*' ou '#'. Par exemple, les abonnés d'Orange peuvent consulter leur compte en composant le '#123#'. Les temps de réponse sont de l'ordre de la seconde, alors qu'un SMS met plusieurs secondes pour être remis à son destinataire.

Un premier schéma, représenté à la Figure 10, consiste en une variante de SMS/OTP :

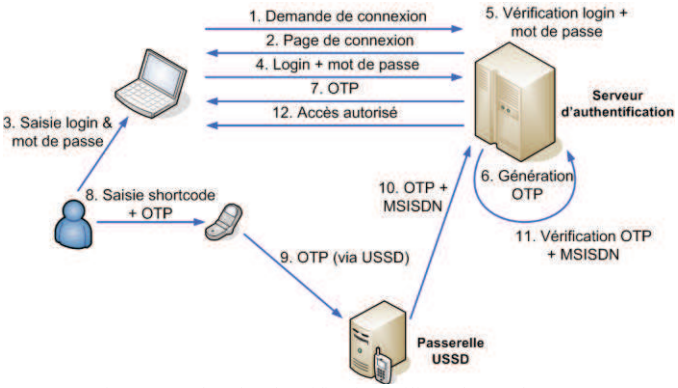

Figure 10. Flux d'authentification utilisant le canal USSD

6. Après vérification de l'identifiant et du mot de passe (étapes 1 à 5), le serveur génère un OTP,
7. puis envoie une page invitant l'utilisateur à composer le « shortcode » d'accès au service USSD, suivi de l'OTP. Par exemple, si le shortcode est '*#149#', et l'OTP '123456', alors la page indiquera à l'utilisateur de composer '*#149#123456#',
8. L'utilisateur compose le numéro demandé,
9. ce qui permet l'envoie de l'OTP vers la passerelle USSD,
10. La passerelle renvoie au serveur d'authentification le numéro de mobile MSISDN (Mobile Station ISDN Number) de l'utilisateur et l'OTP qu'elle vient de recevoir,
11. Le serveur vérifie le numéro MSISDN de l'utilisateur et la concordance de l'OTP,
12. Ce qui permet d'autoriser l'accès.

On voit que ce schéma diffère de SMS/OTP par le sens de circulation de l'OTP : celui-ci est cette fois lu sur l'interface et envoyé via le mobile de l'utilisateur sur le réseau GSM. Les caractéristiques en sont assez semblables, avec l'avantage d'une meilleure réactivité pour l'USSD :

- Utilisable sur tous les téléphones mobiles,
- Pas d'installation d'application dans le mobile ou la carte SIM,
- La preuve de possession du mobile est basée sur l'authentification GSM,
- Authentification à la fois bi-facteur et bi-canal,
- Contraintes ergonomiques équivalentes, la saisie de l'OTP sur la page de connexion étant remplacée par une saisie sur le mobile.

L'un de nos critères cités en introduction de ce paragraphe pour la mise en place d'une authentification forte est la simplicité d'usage, or tous les schémas présentés jusqu'ici impliquent la saisie de l'OTP par l'utilisateur, donc une complexité accrue par rapport à la simple saisie du couple identifiant/mot de passe. L'idéal serait de pouvoir à la fois

générer l'OTP et l'envoyer vers le serveur d'authentification lors d'une même manipulation. Nous avons donc élaboré un autre schéma, montré à la Figure 11, utilisant l'USSD et dérivé de Google Authenticator. Il s'agit d'installer sur le smartphone de l'utilisateur une application générant l'OTP, et envoyant celui-ci via le canal USSD, plutôt que de le saisir manuellement sur la page d'authentification.

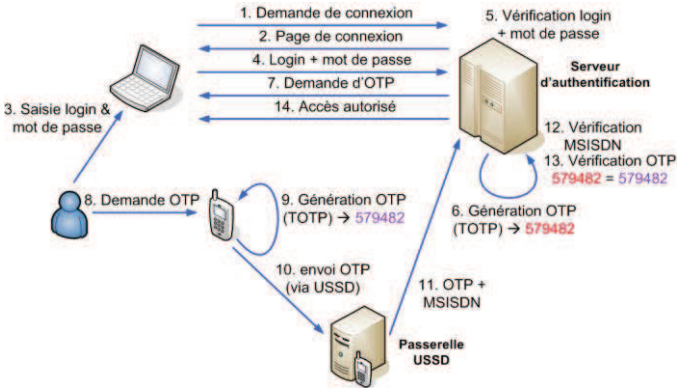

Figure 11. Flux d'authentification avec génération d'OTP synchrone par le smartphone selon l'algorithme TOTP, et envoi via USSD

Par rapport à Google Authenticator, ce schéma a l'avantage de fournir un mécanisme prouvant de manière formelle, via le mécanisme d'authentification GSM, la détention du mobile et de la carte SIM par l'utilisateur. Comme dans le schéma représenté à la Figure 10, on a donc bien une solution d'authentification forte, à la fois bi-facteur et bi-canal.

L'autre avantage de cette solution, par rapport à tous les schémas décrits précédemment, est de simplifier l'usage : il n'y a plus de saisie d'OTP, source d'erreurs éventuelles, et la durée de l'authentification s'en trouve raccourcie.

L'OTP a une durée de vie limitée – à 30 secondes par défaut – ce qui impose à l'utilisateur d'actionner l'application sur son smartphone dans ce laps de temps, au bout duquel le serveur annule la transaction. Il est également nécessaire de bloquer les requêtes suivantes tant que le serveur n'a pas reçu l'OTP ou que celui-ci n'a pas expiré. En effet, si plusieurs transactions sont initiées au nom de l'utilisateur – ce qui pourrait être réalisé par un malware – pendant la durée de validité d'un OTP, il n'existe pas de moyen de savoir quelle est celle que l'OTP valide, ce qui pourrait être exploité par un attaquant. Une alternative est l'utilisation de HOTP, qui permet la génération d'un OTP basé sur un compteur, ce qui permet d'associer de manière non ambigüe un OTP avec une transaction initiée par le poste client.

Une autre solution est la génération d'un OTP asynchrone, selon l'algorithme OCRA. Dans ce dernier cas, afin d'éviter la saisie du challenge, celui-ci est affiché sous forme d'un code QR que l'application installée sur le smartphone récupère pour générer ensuite l'OTP. Le flux d'authentification par OTP asynchrone est décrit à la Figure 12 :

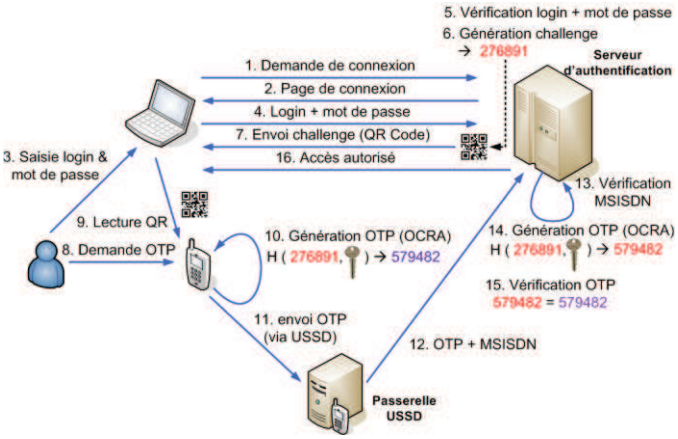

Figure 12. Flux d'authentification avec génération d'OTP asynchrone par le smartphone selon l'algorithme OCRA, et envoi via USSD

Nous avons implémenté cette solution en tant que « proof of concept » lors de notre étude, et une demande de brevet a été déposée en septembre 2011 à l'INPI [11].

La génération d'un OTP synchrone ou asynchrone nécessite le partage d'une clé secrète entre le token de l'utilisateur – son smartphone dans le cas de notre application – et le serveur d'authentification. Cette clé partagée est en fait une clé fille obtenue lors de la phase d'enrôlement, en diversifiant une clé mère détenue uniquement par le serveur avec le numéro de téléphone mobile MSISDN, élément propre à la carte SIM. La clé fille est donc liée à la carte SIM de l'utilisateur. L'algorithme de diversification étant bien entendu à sens unique, il est impossible de reconstituer la clé mère à partir d'une clé fille.

La phase d'enrôlement consiste donc à générer la clé fille et à la stocker sur le smartphone, après installation de l'application. Le flux d'enrôlement est représenté Figure 13 :

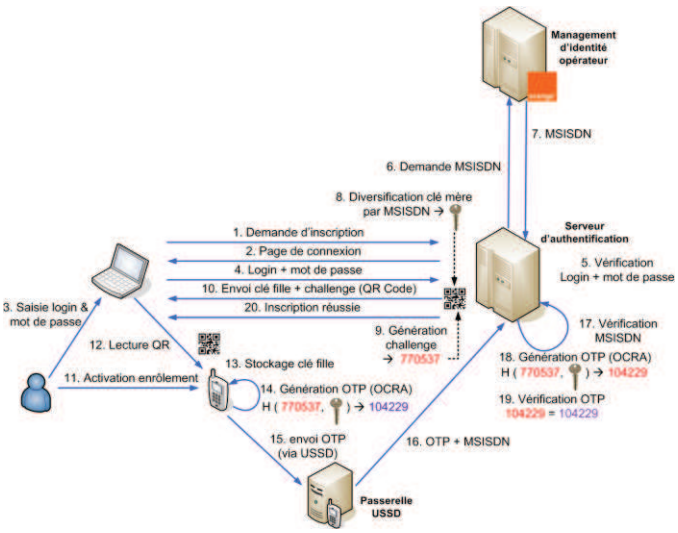

Figure 13. Flux d'enrôlement pour initialiser la clé fille nécessaire à la génération des OTP par le smartphone

La génération de la clé fille par diversification de la clé mère est simplifiée par le fait que l'opérateur dispose déjà du numéro MSISDN, lié de manière fiable à l'identité de l'utilisateur, lequel n'aura donc pas à le fournir à nouveau. L'enrôlement s'appuie sur le management d'identité de l'opérateur : après vérification de l'identifiant et du mot de passe de l'utilisateur, le serveur d'authentification récupère auprès du management d'identité le numéro MSISDN de l'utilisateur, qu'il utilise comme diversifiant pour générer la clé fille. A des fins de vérification, le serveur d'authentification génère également un challenge, et l'ensemble clé fille + challenge est envoyé à l'utilisateur sous forme d'un code QR qui sera scanné par le smartphone. Celui-ci en extrait la clé fille qu'il va stocker en mémoire de masse dans un container chiffré. Afin de valider l'enrôlement, il génère ensuite un OTP asynchrone à partir du challenge et l'envoie sur le canal USSD, selon un flux analogue à celui représenté à la Figure 12. La vérification de l'OTP permet de clore la phase d'enrôlement, à la suite de quoi le client sera ultérieurement reconnu par le serveur comme utilisateur d'authentification forte.

L'authentification forte dans le schéma décrit à la Figure 12 repose en fait sur l'attachement de la SIM au réseau, et non sur les clés qui sont stockés dans la mémoire de masse du smartphone. On peut donc le remplacer avantageusement par celui représenté à la Figure 14 : la génération de l'OTP est laissée au serveur d'authentification, ce qui simplifie l'application dont le rôle se limite à la lecture et à l'interprétation du code QR, et à la retransmission de l'OTP ainsi décodé via le canal USSD. Ceci permet également d'éviter la phase de diversification et le stockage de la clé secrète. On retrouve alors un schéma dont la cinématique est semblable à celle de la figure 10, mais l'utilisation d'un code QR pour transmettre l'OTP généré par le serveur permet d'en éviter la ressaisie et améliore l'expérience utilisateur.

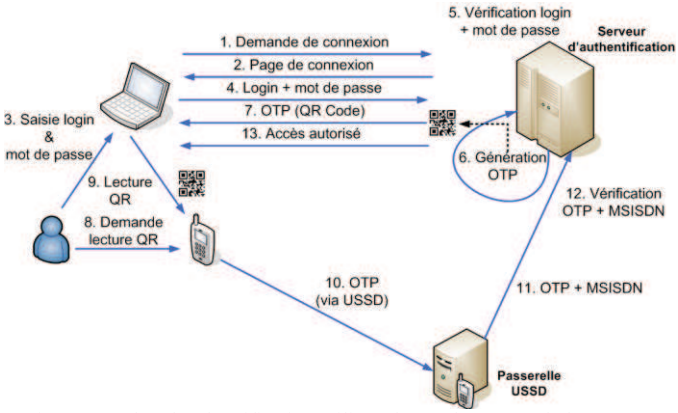

Figure 14. Flux d'authentification utilisant le canal USSD et la lecture par code QR d'un OTP généré par le serveur d'authentification.

V. CONCLUSION

Dans cette étude, nous avons présenté divers schémas d'authentification forte s'appuyant sur le téléphone mobile et la carte SIM. Ces solutions d'authentification permettent d'éviter au client le surcoût d'un token dédié et s'appuient, contrairement à un « soft-token », sur des mécanismes liés à l'UICC. Certains de ces schémas ne nécessitent aucune installation d'application, ni sur le terminal, ni sur l'UICC, et peuvent fonctionner avec un téléphone de base, d'autres sont basés sur l'emploi d'une application installée sur un smartphone mais offrent en contrepartie une grande simplification de l'expérience utilisateur.

Le déploiement de telles solutions d'authentification forte par un opérateur de téléphonie mobile, dont le management d'identité est par nature fiable, permettrait à celui-ci de se positionner comme fournisseur d'identité et tiers d'authentification, avec de forts atouts pour susciter la confiance des services consommateurs d'identité et des utilisateurs finaux. Enfin, pour couvrir l'ensemble des utilisateurs d'un territoire, il faudrait envisager une architecture multi-opérateurs basée sur des mécanismes tels que ceux décrits dans cet article.

REFERENCES

[1] OpenID, "Specification OpenID 2.0 Final", 2007. http://openid.net/specs/openid-authentication-2_0.html

[2] OASIS, "Security Assertion Markup Language v2", 2005. http://saml.xml.org/saml-specifications

[3] IETF, "The OAuth 1.0 Protocol", RFC5849, 2010.

[4] P. Mell, T. Grance, NIST. "The NIST Definition of Cloud Computing V15", NIST, July 2009.

[5] Kim Cameron, "The Laws of Identity", May 2005. http://www.identityblog.com/stories/2005/05/13/TheLawsOfIdentity.pdf

[6] David Barroso, "ZeuS Mitmo: Man-in-the-mobile", September 2010. http://securityblog.s21sec.com/2010/09/zeus-mitmo-man-in-mobile-i.html

[7] Initiative for Open AuTHentication, OATH Reference Architecture, Release 2.0, 2007.

[8] D. M'Raihi,.S. Machani, M. Pei, J. Rydell, "TOTP : Time-based One-Time Password Algorithm", IETF, RFC6238, May 2011.

[9] D. M'Raihi,.M. Bellare, F. Hoornaert, D. Naccache, O.Ranen, "HOTP : An HMAC-based One-Time Password Algorithm", IETF, RFC4226, December 2005.

[10] D. M'Raihi, J. Rydell, S Bajaj, S. Machani, D. Naccache; "OCRA : OATH Challenge-Response Algorithm", IETF, RFC6287, June 2011.

[11] J-J. Schwartzmann, C. Schutz, "Authentification forte par OTP via le canal USSD", INPI n° FR 11 57973, déposé le 08 septembre 2011.