Outils pour la mise en œuvre de systèmes Androïd

Serge Bordères

Centre d'Etudes Nucléaires de Bordeaux-Gradignan
Observatoire des TEchnologies Nomade et de l'Internet pour la Recherche

Grenoble – 13 novembre 2013

RESINFO

cnrs
dépasser les frontières

DEVLOG

Quelques mots sur OBTENIR

Observatoire des Technologies Nomades et de l'Internet

http://www.obtenir.cnrs.fr

- Favoriser l'émergence d'une expertise dans le domaine des technologies nomades et de l'Internet.

- Favoriser la diffusion des connaissances par des formations, conférences, articles...

- Prospecter les technologies.

- Connaître et faire connaître l'usage de ces technologies au service de la recherche et plus particulièrement au sein des expériences.

Quelques mots sur OBTENIR

Observatoire des Technologies Nomades et de l'Internet

Site web => http://www.obtenir.cnrs.fr

Liste de diffusion => nomadisme ///AT/// service.cnrs.fr

- Un groupe d'une dizaine de personnes
- Parrainé par RESINFO et DEVLOG
- S'adresse à TOUS les acteurs de la recherche, informaticiens et utilisateurs
- Concerne tous les aspects du nomadisme physique (les postes de travail) ou virtuels (les données les services).

Pour commencer...deux citations

1ère citation

Document « Recommandations de sécurité relatives aux ordiphones » de l'ANSSI

Il est illusoire d'espérer atteindre un haut niveau de sécurité avec un ordiphone ou une tablette ordinaire, quel que soit le soin consacré à son paramétrage.

2ème citation

Patrick Paillou, directeur de l'ANSSI

[...]Je vais vous dire ma vision des choses : il faut entrer en résistance contre la liberté totale dans l'usage des technologies de l'information. [...] La sécurité c'est aussi avoir le courage de dire non

Pour commencer...quelques constats

Depuis les débuts de l'informatique nous utilisons des technologies pétries de problèmes de sécurité, et qui ne pouvaient « espérer atteindre un haut niveau de sécurité »....

« L'Histoire nous apprend que l'Histoire ne nous apprend rien »

- Comme d'habitude, la nouvelle vague technologique des mobiles n'a pas suffisamment intégré la sécurité dés le départ.
- Il sera (est) impossible de les ignorer et de leur interdire indéfiniment l'accès au système d'informations.

Pour la première fois de nouvelles technologies sont massivement introduites d'abord dans la sphère privée puis professionnelle

Des mobiles dans le système d'information
Est-ce raisonnable ?

- L'absence de politique vis à vis des mobiles n'est plus raisonnable !
  Les utilisateurs n'ont souvent pas besoin de demander la permission (par ex. lorsqu'il suffit d'un login/mot de passe pour accéder à quelque chose)

- Il n'est pas raisonnable d'affaiblir des procédures de sécurité déjà existantes sous le prétexte de permettre l'accès aux mobiles. C'est eux qui doivent s'adapter aux conditions déjà existantes.

- Il n'est pas raisonnable de « tout interdire ». Il n'est pas raisonnable de laisser faire « n'importe quoi » et de ne pas gérer la situation.

- Il est raisonnable de se former, de mettre en place des solutions qui permettent une certaine ouverture, puis de progresser pas à pas.

Quels problèmes pour l'ASR ?

- Comment garder le contrôle sur le parc ?
  Il ne suffit plus de savoir qui accède au réseau mais avec quoi (BYOD)

- Eviter la dissémination des identifiants professionnels sur toute sorte de matériels non référencés

- Connaître le fonctionnement « système et réseau » des systèmes d'exploitation (principalement Androïd et IOS)

- Ne pas affaiblir la sécurité déjà existante du S.I

Le B.Y.O.D, nouveau défi des ASR

Bring Your Own Device

Pratique qui consiste à utiliser un matériel personnel dans l'activité professionnelle

Soit il s'agit d'une politique volontaire d'un établissement
Soit il s'agit d'un fait accompli

Pour la première fois les nouvelles technologies sont introduites d'abord dans la sphère privée puis professionnelle

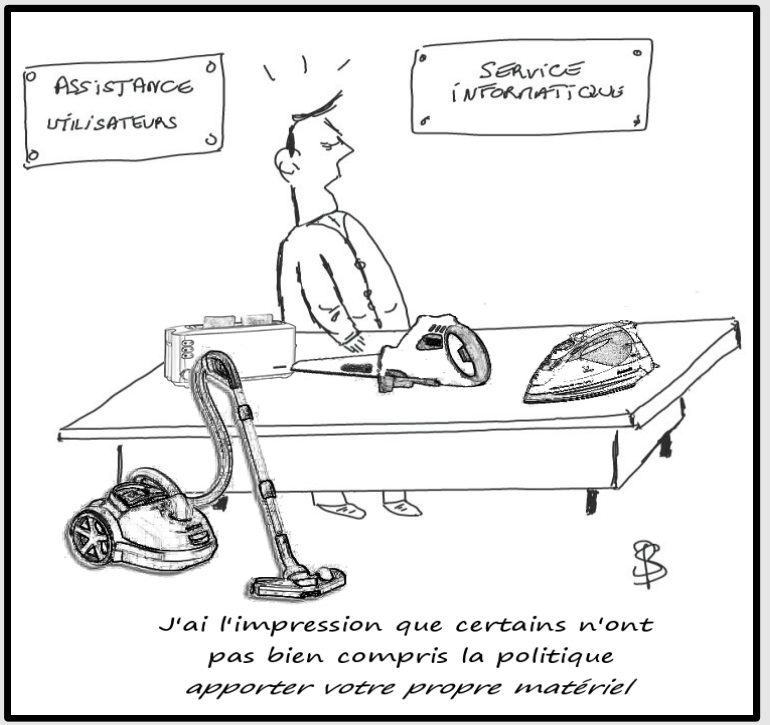
Un mobile professionnel (acheté par l'employeur) devient très vite personnel

Le B.Y.O.D, nouveau défi des ASR

Le BYOD c'est comme la fin des uniformes dans les écoles

Citation magazine MISC N°66 mars/avril 2013

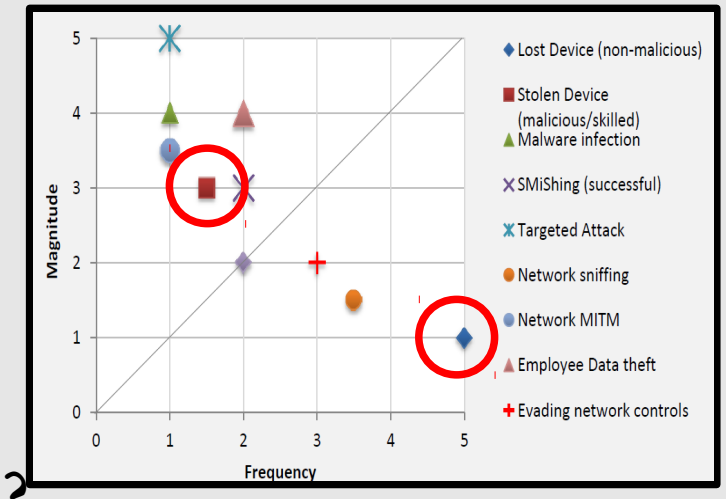
- La diversité des OS est **une contrainte** pour les informaticiens mais une tendance naturelle (bio-diversité plus riche qu'uniformité)
- L'ensemble des terminaux qui accèdent au S.I n'est plus complètement géré ni même connu.
  - **Un utilisateur peut posséder plusieurs mobiles**
  - **Que devient un mobile en fin de vie ?**
    - cédé à un membre de la famille ?
    - Vendu ?
    - Dans tous les cas a-t-il été remis en configuration d'usine et toutes les informations qu'il contient effacées (notamment les mots de passe) ?
  - **En cas de panne, l'opérateur le remplace. Les informations sont-elles effacées ?**



Le B.Y.O.D, nouveau défi des ASR

Forte volatilité des mobiles

- Les mobiles sont **plus exposés** aux pertes ou vols
  - On les amène partout
  - Produits technologiques très recherchés
  - Peu de sécurité et potentiellement beaucoup d'informations (y compris mots de passe...) qui peuvent alimenter un marché de revente.
- Des matériels personnels perdus/volés **sont-ils déclarés ?**
  - Risques de fuites d'informations non répertoriées
  - Pas de blocage des autorisations d'accès.



Mobility Security risk report (viaForensics)
https://viaforensics.com/resources/reports/mobile-security-risk-report/

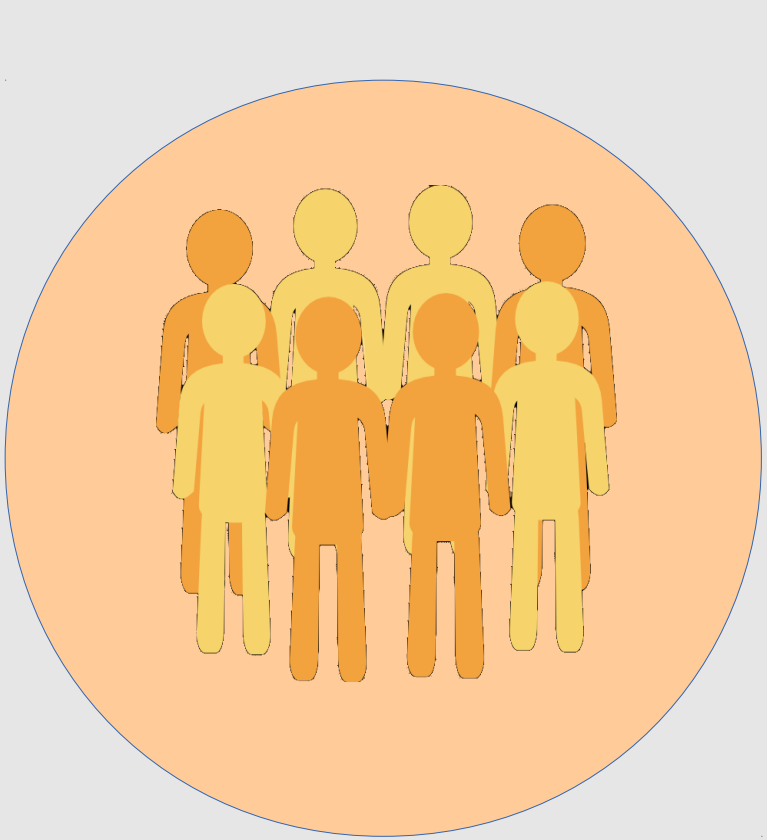
- Outils très personnel, insouciance
- Matériels conçus avant tout pour une diffusion grand public.
- Manque de formation ou sur-estimation de connaissances

Le B.Y.O.D, nouveau défi des ASR

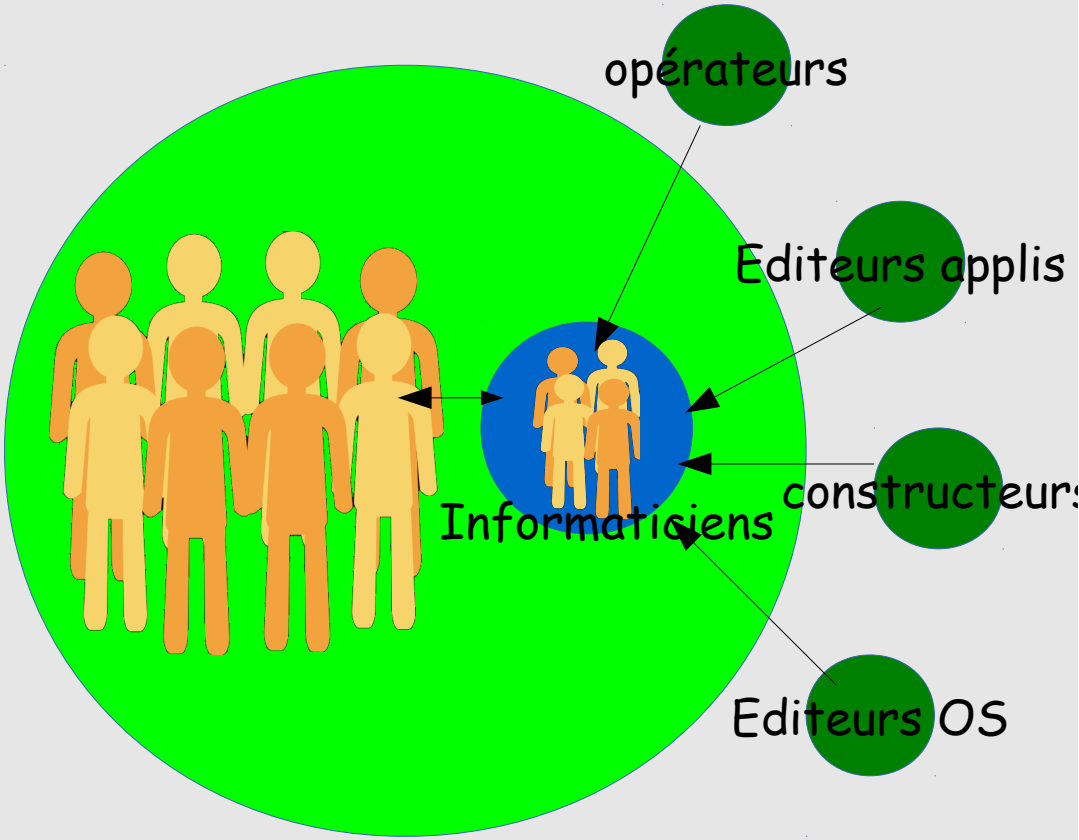
- Sécuriser au mieux les mobiles
- Connaître le parc
- Sécuriser l'accès au SI

Votre mobile nous intéresse !

Avant le monde était simple....tout passait par les informaticiens



Sphère privée

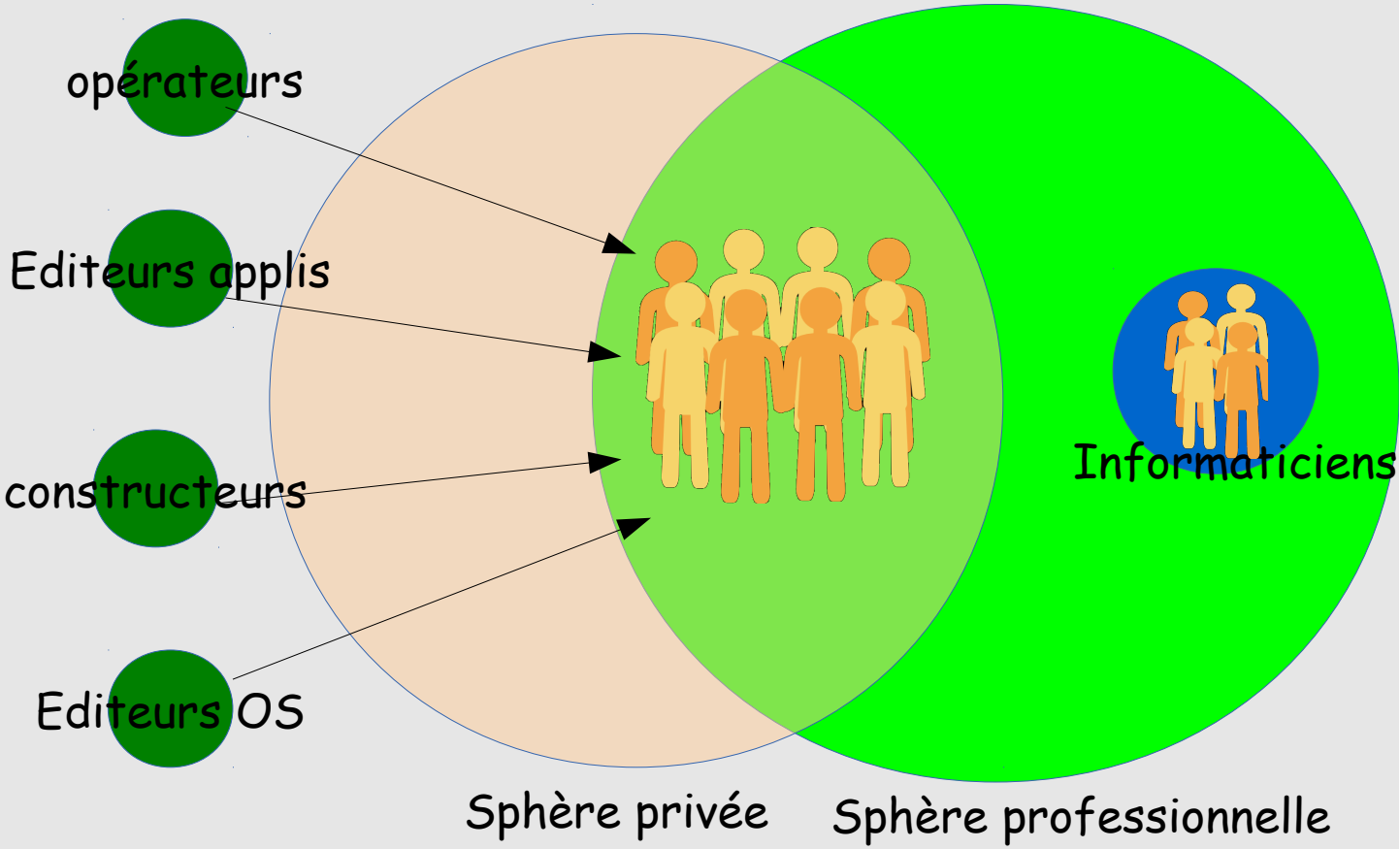


Sphère professionnelle

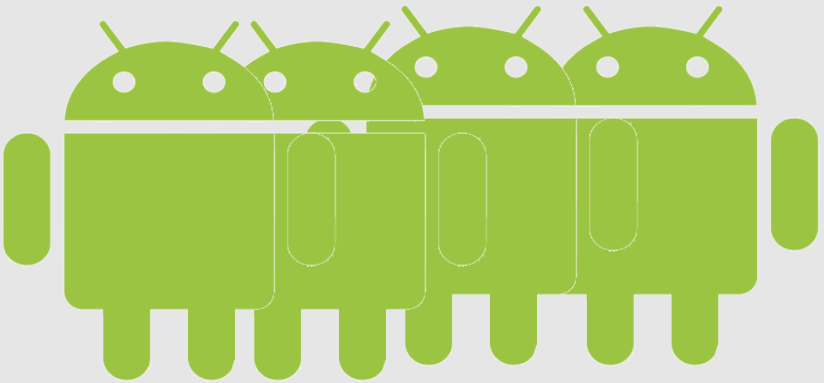
Votre mobile nous intéresse !

Aujourd'hui,
tous les acteurs s'adressent directement aux utilisateurs

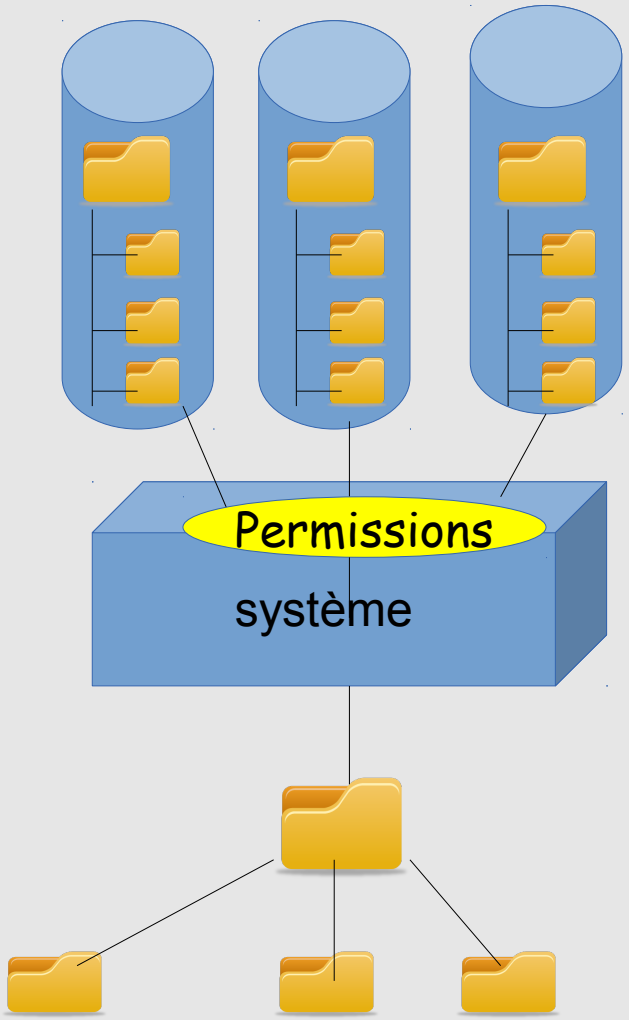
- **Discours**
  - Marketing
  - Économique
  - Technologique
  - Pas du tout sécuritaire !
  - Nous avons des solutions pour votre entreprise
  - Venez, venez dans notre cloud



Modèle de base d'Androïd



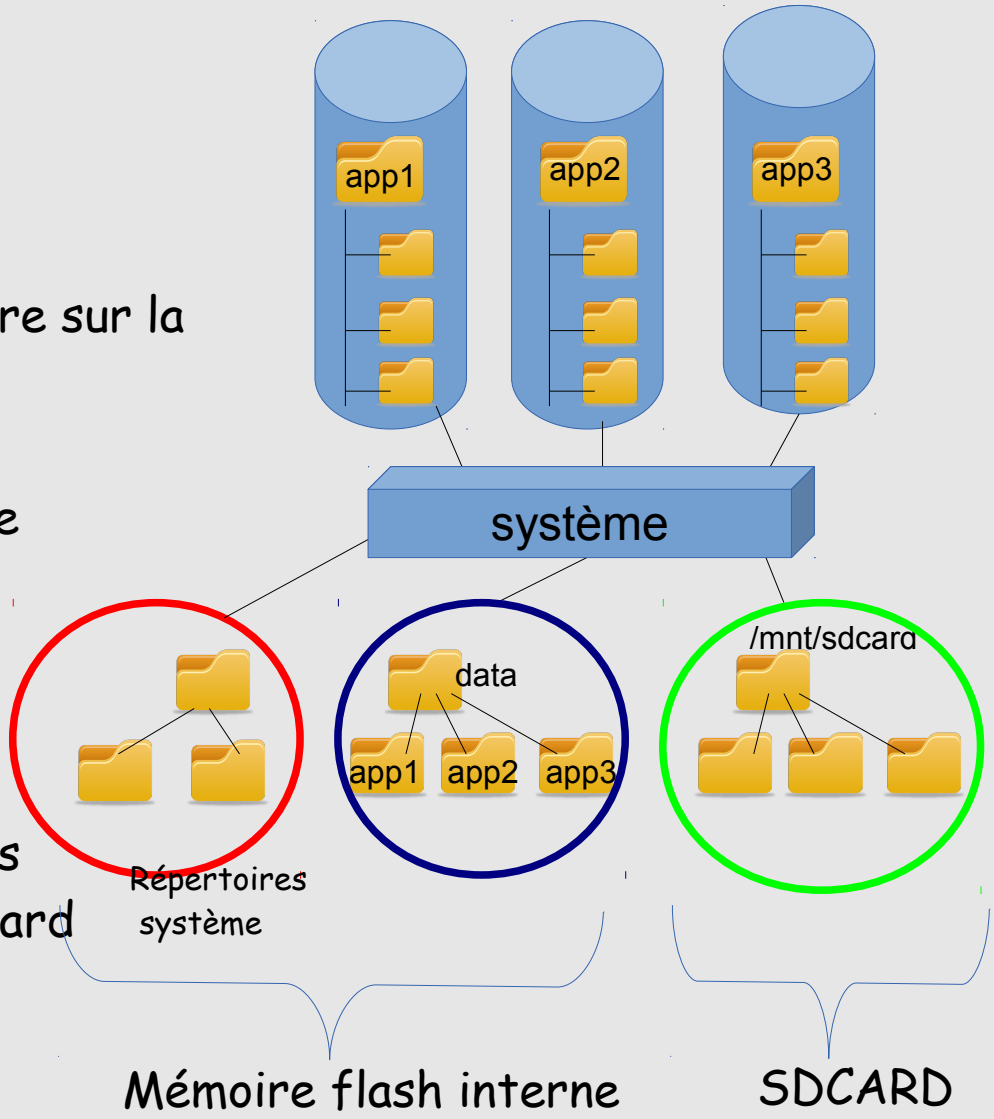
Le bac à sable



- Chaque application tourne dans un conteneur (machine virtuelle/sandbox) qui l'isole des autres applications
- Dans Androïd une application ne peut interagir avec ce qui lui est extérieur qu'au travers des permissions qui lui sont accordées.
- Chaque application a sa propre arborescence
- Mais il existe une zone partagée (sdcard)

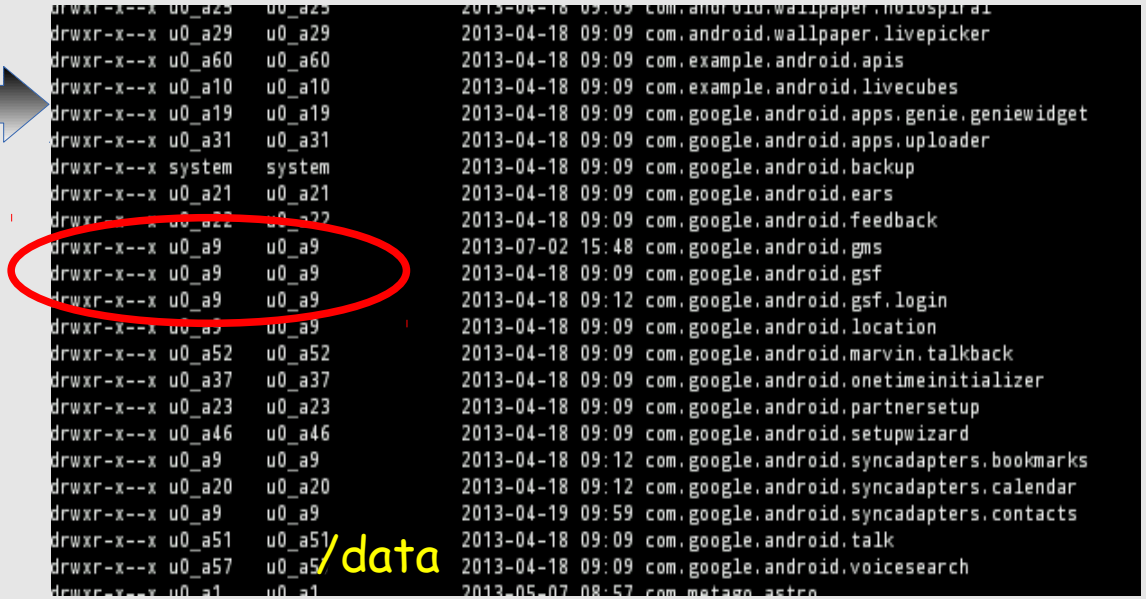
Système de fichiers

- Androïd dispose de deux types d'emplacements de stockage :
  - La mémoire flash dite « interne »
  - La SDcard dite « externe »
- Lorsqu'une application est autorisée à lire/écrire sur la SDcard elle peut tout y lire/modifier
- L'utilisateur peut créer sa propre arborescence contenant toutes sortes de documents
- Structure en projet possible
- Il est possible de copier toute sorte de fichiers depuis un ordinateur ou une clé USB sur la SDcard

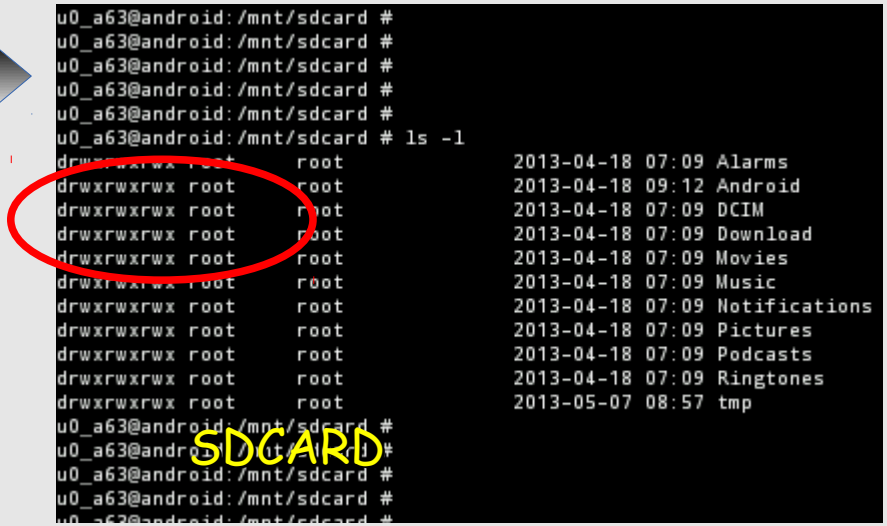


Système de fichiers

Dans la mémoire interne chaque application est propriétaire de son arborescence (chaque appli a un UID et un GID)



Sur la carte SD tous les répertoires sont accessibles en RWX



Donc accessibles pour toutes les applications qui ont la permission de lire/écrire sur la SDCARD(Modifier/supprimer le contenu du stockage USB)

Permissions

- Chaque application doit déclarer les permissions dont elle a besoin (manifest)
- Lors de l'installation, l'utilisateur doit accepter, ou pas, de délivrer les permissions à l'application
- Obligation d'accepter tout ou rien

Des outils pour apprendre … la virtualisation d'Androïd

Virtualisation d'Androïd

Utilité

- Faire tourner un système Androïd dans une machine virtuelle.
- Très proche d'un système réel
- Pour se former
- Faire des démos
- Tester des versions récentes d'Androïd ou des applications

Les solutions existantes

Androïd x86 : Portage d'Androïd dans l'environnement x86
Fonctionnement en machine virtuelle ou en natif

GenyMotion (anciennement AndroVM) => commercial: distribution d'une machine virtuelle pour VirtualBox

WindowsAndroïd : Il s'agit en réalité d'un émulateur qui ne nécessite pas de machine virtuelle.

Virtualisation d'Androïd : Androïd X86

- http://www.android-x86.org/
- Projet de portage d'Androïd sur les plate-formes X86
- Le projet fourni des images ISO de systèmes Androïd (4.3)
- Ces **images** permettent une utilisation en Live sans installation ou d'installer le système sur un disque virtuel ou physique.
- Le réseau est disponible
- Google Play est disponible : Installation des applications comme sur un smartphone/tablette

- Utilisations possibles :
  - Sur un système physique
  - Dans une machine virtuelle (Dans n'importe quelle solution de virtualisation)
  - Sur une clé USB

Virtualisation d'Androïd : Androïd X86

L'image ISO peut être utilisée soit pour booter en Live, soit pour installer le système sur disque

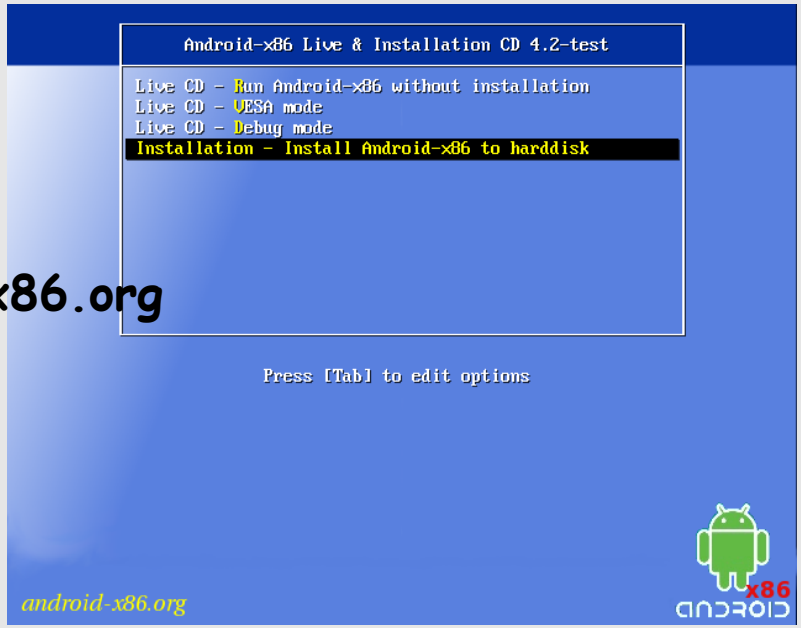
- **Il faut d'abord créer la machine virtuelle (par exemple Virtual Box)**

Exemple de configuration :

1 Go de mémoire
1 disque virtuelle 1Go (uniquement pour faire l'installation)
1 Lecteur CD/DVD qui pointe sur l'image ISO.
Option de démrrage : boot sur le CD/DVD

- **Télécharger l'image ISO depuis le site android-x86.org**

- **Booter la machine virtuelle**

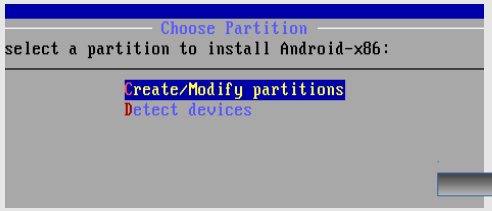


Virtualisation d'Androïd : Androïd X86

Pour installer Androïd sur le disque virtuel

- Booter sur l'image ISO
- Choisir l'option « Install Androïd to harddisk »

Choisir Create/Modify Pour créer la partition

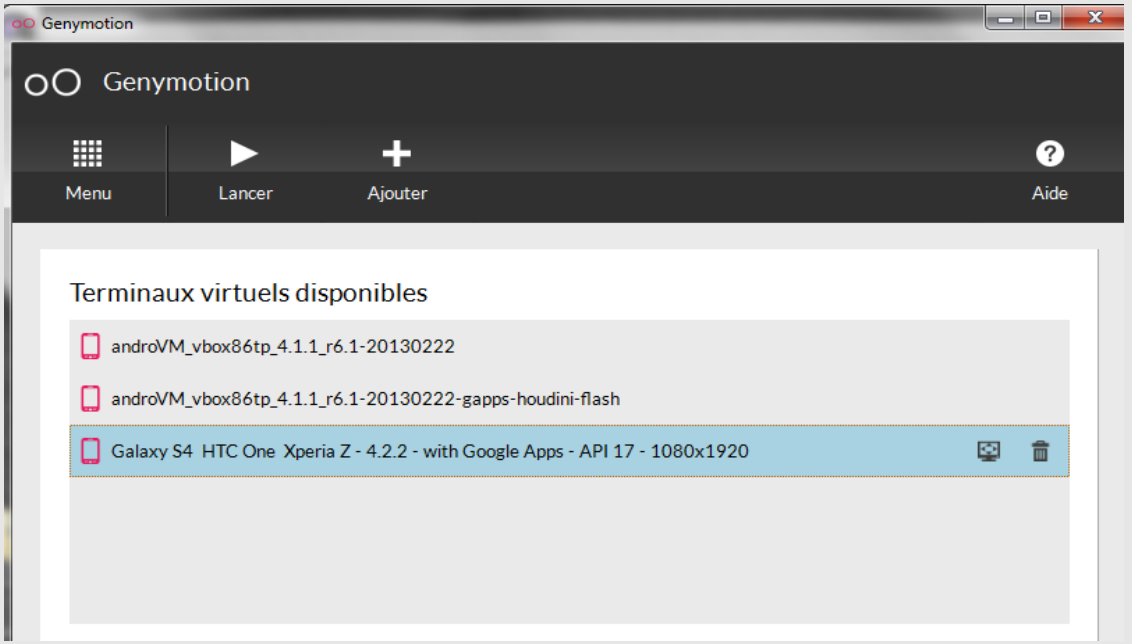


Dans la fenêtre cfdisk créer une partition bootable sur la totalité du disque.

Choisir cette partition et la formater en ext3

Virtualisation d'Androïd : GenyMotion

- https://cloud.genymotion.com
- Nécessite de s'enregistrer et télécharger un gestionnaire de téléchargement
- Nécessité d'installer VirtualBox
- Plusieurs versions de systèmes (4.2.2) fournis sous forme de machine virtuelle Virtual Box prête à l'emploi.



Principaux dispositifs de sécurité gérables

Verrouillage d'écran

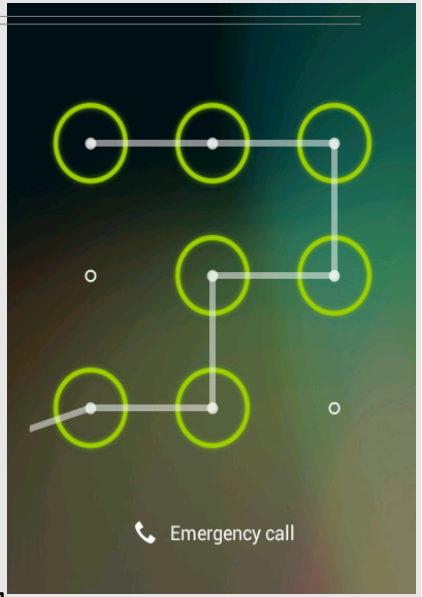
- Le verrouillage d'écran est la protection minimale et obligatoire

Tout mobile en relation avec le S.I doit verrouiller son écran

- Même si cette protection est considérée comme peu robuste

- Rien à voir avec le pincode de la carte SIM qui protège uniquement l'utilisation de cette carte

Verrouillage d'écran : Par modèle



Consiste à dessiner un motif entre les points contigüs

- La robustesse est proportionnelle à la complexité du motif

- Réputé peu robuste car le nombre de combinaisons est faible :
  - 4 points => 1624 solutions
  - 5 points => 7152 combinaisons
  - 6 points => 26016 combinaisons ==> plus de combinaisons qu'un pincode à 4 chiffres
  - 9 points => 140704 combinaisons
- Attention aux traces de doigts
- Une attaque force brute sur des combinaisons à 9 points révèle aussi les combinaisons de 4 à 8 points.
- Cela nécessite root du mobile, ou mauvais paramétrage (USB debbugging)

0	1	2
3	4	5
6	7	8

En 2012 le FBI n'a pas pu déverrouiller un smartphone verrouillé par un modèle et a dû demander l'aide de Google

Verrouillage d'écran : Par pincode

Verrouillage par une suite de 4 à 16 chiffres

- Robustesse proportionnelle au nombre de chiffres.
- En général 4 chiffres utilisés => 10000 combinaisons seulement
- Attaque force brute par root, pas directement sur l'écran

Pour info : IOS

- Plusieurs vulnérabilité publiées sous IOS
  (combinaison de touches permettant de contourner simplement le pincode
  http://www.zdnet.com/blog/security/iphone-passcode-lock-bypass-vulnerability-again/7544)

- La méthode pour casser le pin code sous IOS a été publiée
(consiste à booter sur un ramdisk ou jailbreaker)

Affaire Pistorus :
La police Sud-africain n'a pas pu déverrouiller l'iphone 5 d'Oscar Pistorius et a demandé l'aide d'Apple

Verrouillage d'écran : par mot de passe alpha-numérique

Utilisation d'un « vrai » mot de passe alpha-numérique

- Méthode la plus robuste
- Mais la moins pratique
- Difficile de tapez un mot de passe contenant chiffres, lettres, caractères spéciaux sur un smartphone :
  - basculement entre les claviers virtuels (numérique, alpha)
  - gros doigts – petites touches
  - Soleil…

Un intérêt important d'un smartphone est sa capacité multifonction qui nécessite de pouvoir le « dégainer » rapidement (téléphone, photo, gps, accès Internet, ou autre application).
Un mot de passe alpha-numérique est un gros frein qui fait perdre de l'intérêt au smartphone qui découragera les utilisateurs.

Gestion multi-utilisateurs

- Androïd 4.2 introduction de la possibilité de créer plusieurs comptes utilisateurs sur un mobile
- Androïd 4.3 introduction des profiles restreints

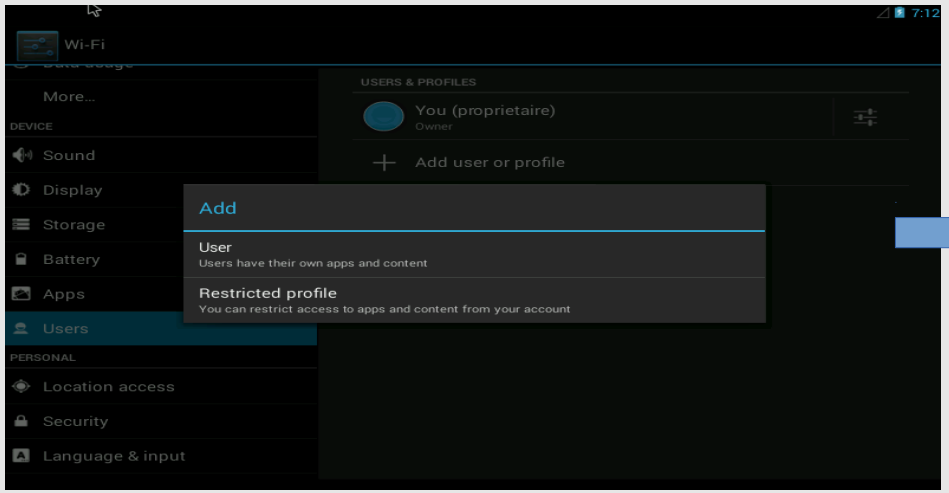
Intérêts

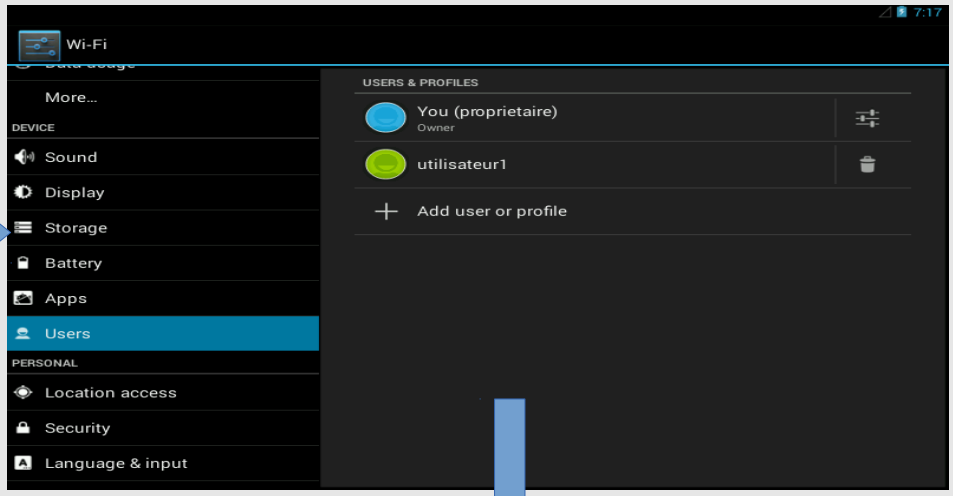
- Au départ : pouvoir partager un mobile (tablette) entre plusieurs personnes en préservant l'environnement de chacun ou pour restreindre les applications.

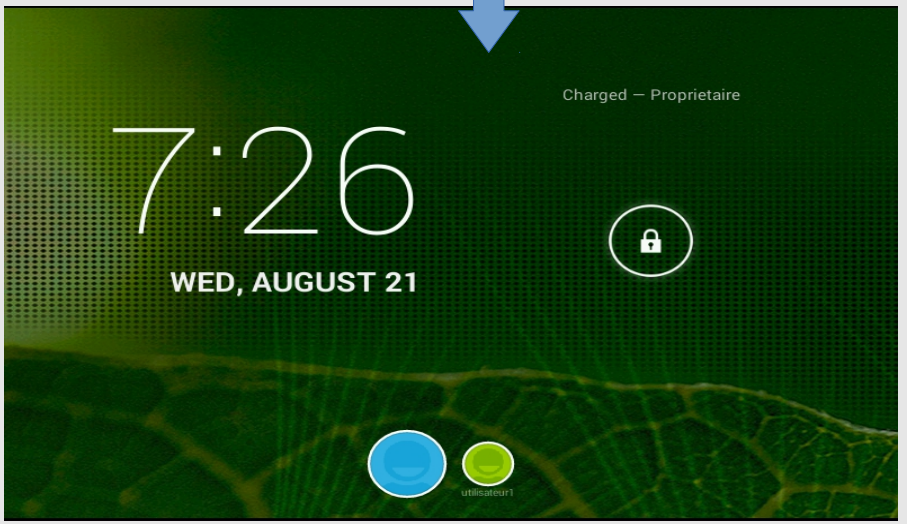
- Intéressant aussi pour séparer l'environnement professionnel du reste.

Gestion multi-utilisateurs

Comptes utilisateurs







- Le compte *utilisateur1* a la possibilité de définir son propre compte Google
- Il voit les applications de base
- Il ne voit pas les applications installées par les autres utilisateurs.
- Il peut installer ses propres applications
- Il peut définir son propre code de verrouillage.
- Etc…

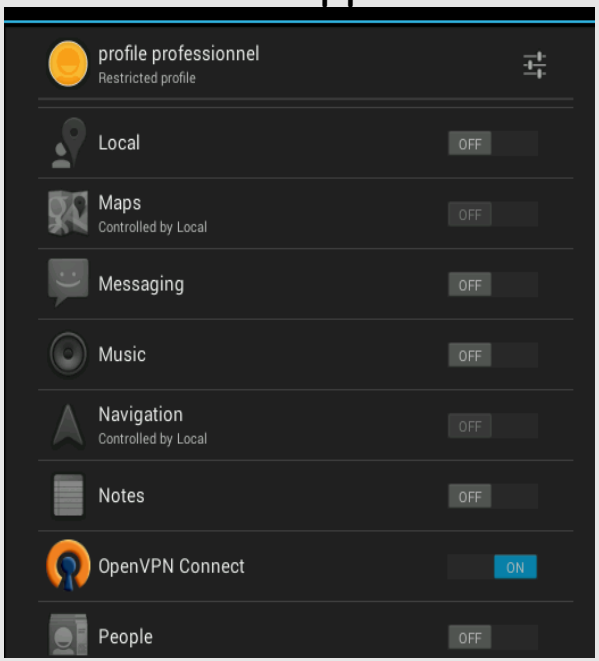
Gestion multi-utilisateurs

Profiles restreints

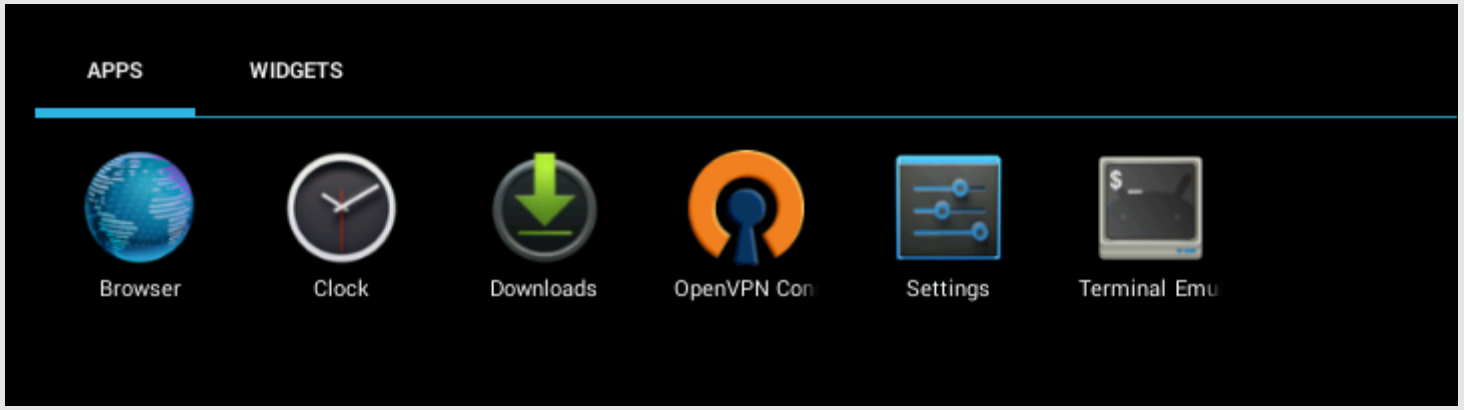
Mêmes caractéristiques que les comptes utilisateurs, avec en plus :

- Le compte du propriétaire du mobile doit obligatoirement avoir un code de verrouillage
- Seules les applications sélectionnées sont utilisables par le profile.

Sélection des applications



Applications utilisables depuis le profile



Le verrouillage d'écran est-il suffisant ?

Non !

- On n'est pas sûr que l'utilisateur a bien positionné un moyen de verrouillage, ou qu'il ne l'a pas supprimé
- On n'est pas sûr que le code de verrouillage n'est pas trivial
- Des vulnérabilités existent
  - Exemple : jusqu'à Androïd 4.0.4, la connexion USB sur un ordinateur n'exigeait pas le déverrouillage préalable du mobile.

Il va falloir faire en sorte d'être sûr que chaque mobile en rapport avec le SI possède un code de verrouillage :
L'usage de certificats sera un bon moyen et permettra, en plus, d'identifier chaque mobile =>>>> voir plus loin.

Stockage des codes de verrouillage ?

- **Verrouillage par motif**
  - Stocké sous forme d'un hash SHA-1 sans salt dans le fichier /data/system/gesture.key
  - Accessible uniquement par le système, sauf si le mobile est rooté
  - Le fichier est attaquable par force brute ou peut être simplement détruit sur un mobile rooté (si le mobile n'est pas crypté==> voir plus loin)
  - Attaquable aussi par dump du système

- **Verrouillage par pincode et mot de passe**
  - Stocké sous forme d'un hash SHA-1 de 40 caractères hexa et un hash MD5 de 32 caractères hexa dans /data/system/password.key et avec un salt aléatoire de 64 bits, stocké dans /data/data/com.android.providers.settings/databases/setting.db
  - Accessible uniquement par le système, sauf si le mobile est rooté
  - Attaquable par force brute (si rooté) ou par dump.

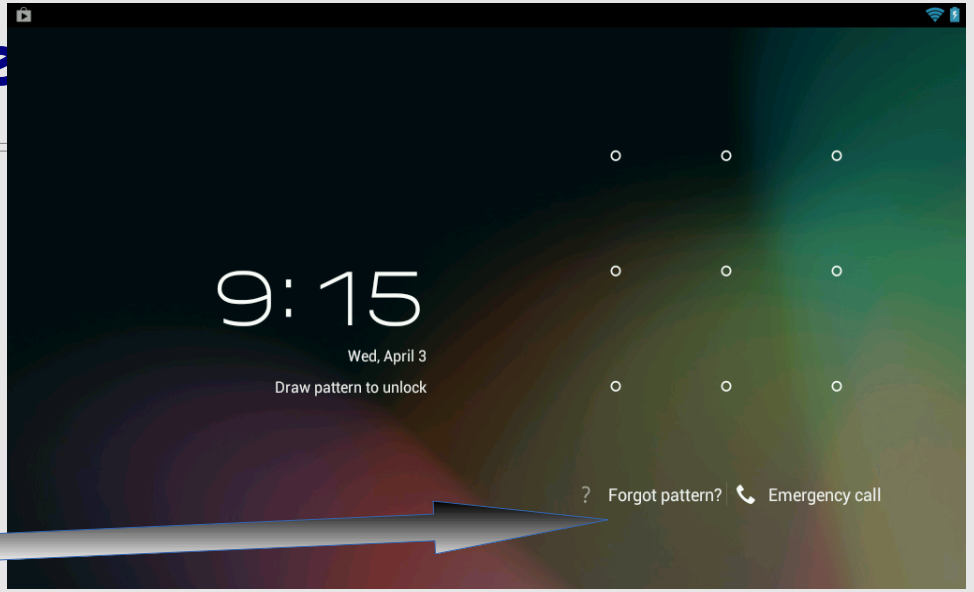
Et le mot de passe Google dans Google Play ?

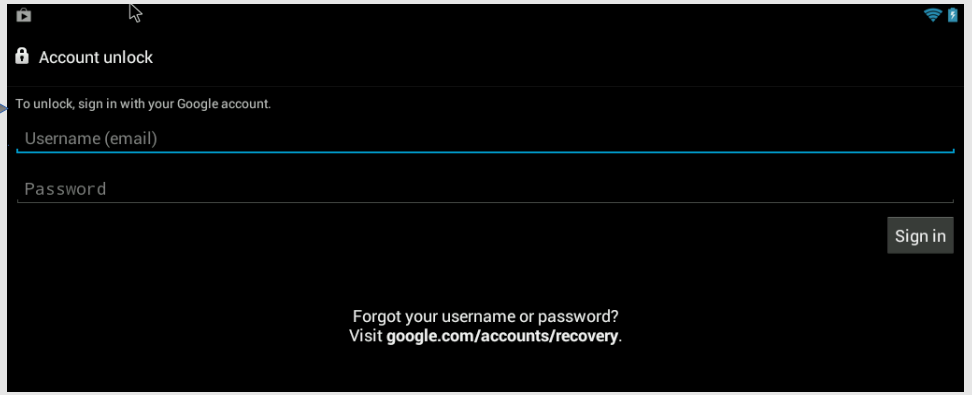
- Tout utilisateur d'un mobile Androïd doit disposer d'un compte Google (et donc d'un mot de passe) pour pouvoir accéder à Google Play.
- Ce mot de passe n'est pas stocké dans le mobile.
- Lors de la première authentification un token est téléchargé sur le mobile (en clair dans /data/system/users/0/accounts.db)
- Ce token ne peut servir qu'à partir du mobile pour Google play et pas pour se connecter sur le compte Google avec un navigateur.
- Le changement de mot de passe Google, invalide le Token.

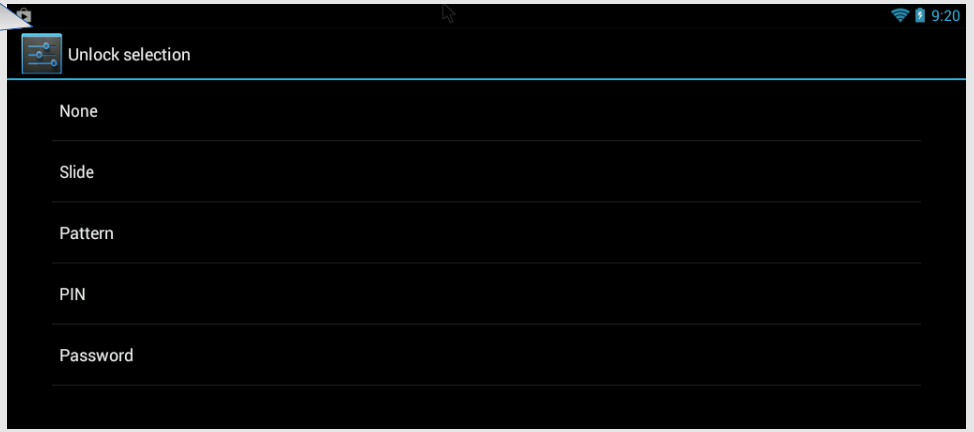
Perte du code de verrouillage

- Un mobile Androïd se bloque pendant 30 secondes à la cinquième tentative de déverrouillage avec un code erroné. (et ainsi de suite toutes les 5 tentatives)
- Il est possible alors de sélectionner l'option « Mot de passe oublié »
- Le login et le mot de passe du compte Google sont alors demandé.
- Une fenêtre propose de choisir un nouveau mode de verrouillage.

Cela marche uniquement si le réseau est activé et connecté







Utilisation des certificats

Utilisation des certificats

Quels intérêts ?

- Chaque mobile peut être identifié par un certificat
- L'importation d'un certificat force l'usage d'un code de verrouillage du mobile
- Un certificat a une durée de vie limitée (un certificat compromis finit par être inopérant)
- En cas de compromission du mobile/certificat , il suffit de révoquer le certificat, seul le mobile est impacté.
- Permet l'accès à un réseau Wifi sécurisé (EAP-TLS)
- Permet l'accès distant sécurisé via VPN

Utilisation des certificats

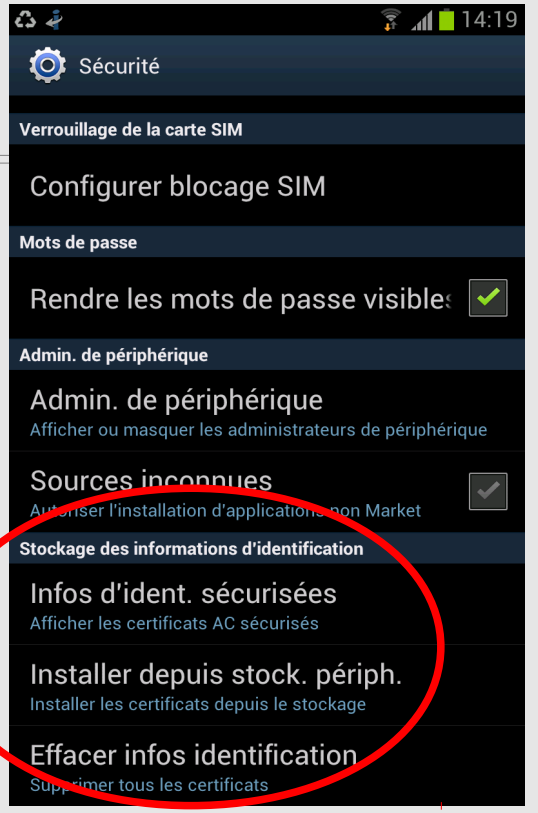
Structure nécessaire

- Il faut disposer d'une **IGC** pour créer les certificats
- Il faut mettre en place une procédure pour que le certificat arrive jusqu'au mobile
- Pour l'autorisation d'accès, un serveur RADIUS

Quels certificats ?

- Techniquement les certificats CNRS sont utilisables
- Comme il s'agit d'une utilisation purement local ce n'est pas très souhaitable (procédure de création, renouvellement, diffusion très lourde et peu adapté)
- La création d'une IGC local peut rendre le processus très simple (et bénéficier à l'ensemble du parc).

Utilisation des certificats

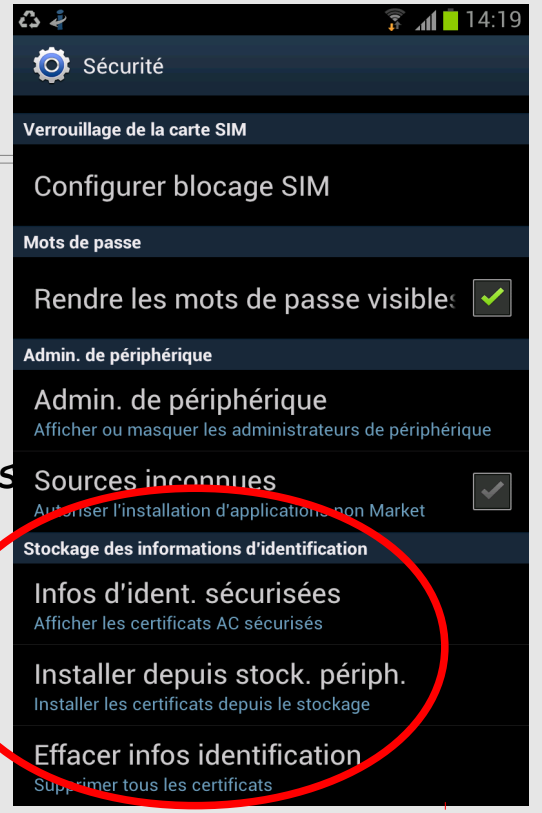


- Androïd dispose d'un magasin de certificats (Keychain)
  A la différence d'IOS, Androïd ne stocke que des certificats dans le keychain.

- La gestion se fait au travers du menu « *Paramètres/sécurité* » puis le paragraphe « **Stockage des informations d'identification** »

- L'importation se fait à partir de fichiers PKCS12 (P12) ou CRT pour les certificats des autorités.

Utilisation des certificats



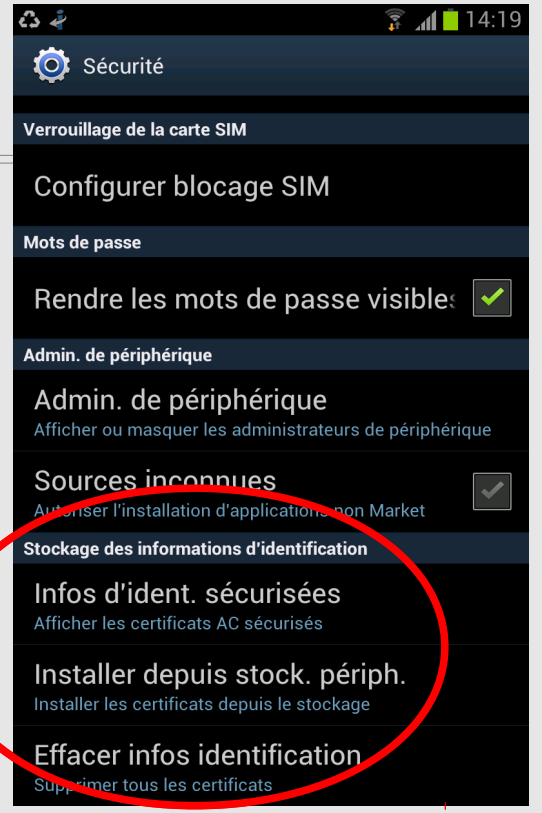
Trois fonctions disponibles

- **Afficher les AC sécurisés**
  - Permet d'afficher la liste des autorités de certification enregistrées (celle initialement incluses et celles rajoutées)
  - Il est possible de les supprimer individuellement

- **Installer les certificats depuis le stockage**
  - Pour installer des certificats dans le magasin (y compris ceux des AC)
  - Les certificats doivent être contenus dans des fichiers P12 ou CRT à la racine de la SDCARD.
  - Après l'installation le fichier est détruit

- **Supprimer tous les certificats**
  - Supprime tous les certificats créé par l'utilisateur (on ne peut pas les supprimer individuellement)

Utilisation des certificats



- Il n'est pas possible d'afficher la liste des certificats enregistrés avec ce menu
- La liste sera visible lorsqu'il s'agira d'autoriser un certificat dans une application. Le système demandera explicitement à l'utilisateur si l'application peut utiliser le certificat.
- Il n'y a pas de fonction d'exportation de la clé privée du certificat
- L'enregistrement du premier certificat exigera que le mobile soit protégé par un moyen de verrouillage (modèle, pin code ou mot de passe).

Utilisation des certificats : diffusion

1ère méthode :
On branche le mobile par USB sur un ordinateur sur lequel se trouve le fichier P12 et on le copie à la racine de la carte SD d'où il sera importer dans le keychain.

2ème méthode :
Si on veut intégrer aussi des mobiles IOS la copie par USB ne fonctionne pas. Avec IOS il faut télécharger le certificat par Wifi. Or, si le certificat est utilisé pour un accès au Wifi sécurisé il faut disposer d'un autre ssid pour le téléchargement. C'est un peu « bête » de devoir ouvrir en permanence un réseau non-sécurisé, pour télécharger des informations confidentielles.

On peut alors utiliser un SSID éphémère.

Utilisation des certificats : diffusion

Méthode du réseau Wifi éphémère

1. On cré un SSID de service qui sera utilisé uniquement pour faire les téléchargements de certificats. Il s'agit d'un réseau ouvert mais qui fait une authentification par adresse MAC. Par défaut rien n'est autorisé à utiliser ce SSID.

2. Le certificat est créé par l'IGC et déposé sur un serveur Web (interne) dans un répertoire protégé.

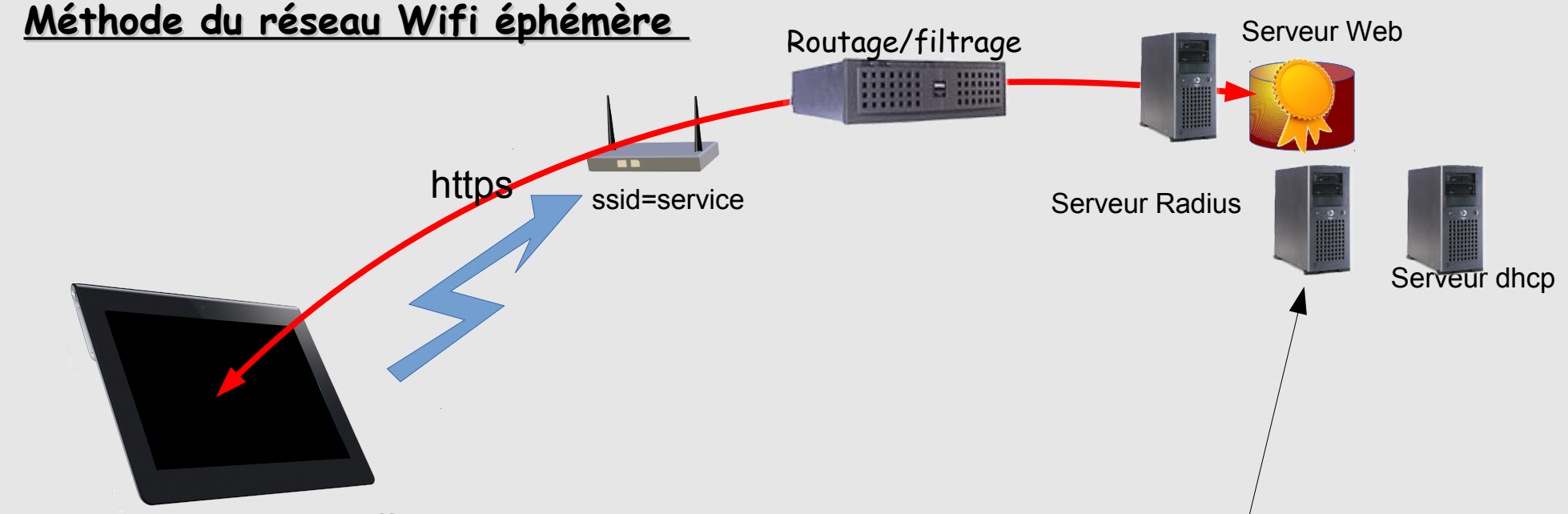
3. Quand on veut permettre un téléchargement de certificat, l'adresse MAC est autorisée sur ce SSID dans le serveur Radius pour une durée de 30 mn par exemple. La machine est aussi déclarée dans DHCP. Après quoi l'autorisation d'accès est révoquée.

4. Le mobile voit ce réseau comme un réseau Ouvert et n'a pas besoin de configuration particulière pour s'y connecter (une fois enregistrée comme précédemment).
5. En fonction des infos envoyées par le serveur Radius à la borne, le mobile est placé sur le Vlan souhaité et reçoit une adresse IP dédiée.

6. A partir de cette adresse le mobile n'a le droit que d'aller sur le serveur Web (https) sur lequel son certificat a été déposé et peut ainsi le télécharger.

Utilisation des certificats : diffusion

Méthode du réseau Wifi éphémère

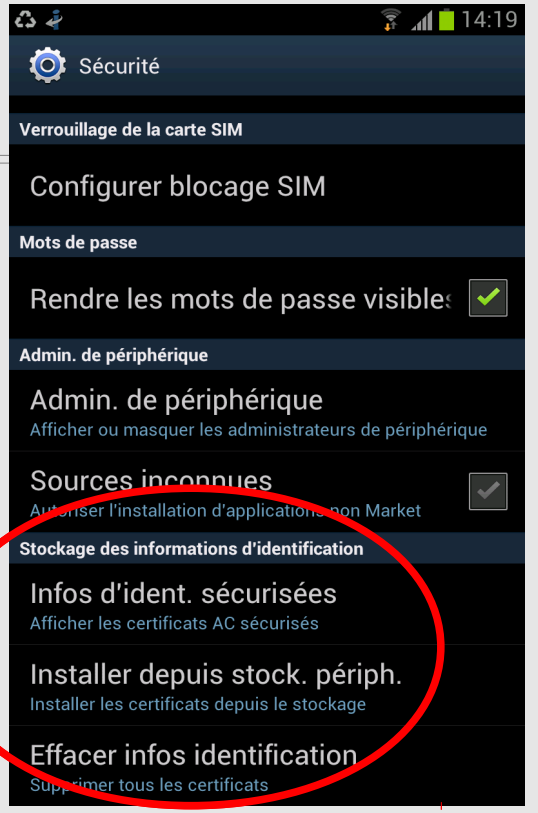



```
expire=$(date --date "30 minutes" +"%d %b %Y %H:%M:%S")
cat > /path/wifimere << !!
$MAC2   Auth-Type := Accept,Calling-Station-Id == "$MAC1",Expiration := "$expire",Colubris-AVPair == "ssid=service"
          Tunnel-type  = VLAN,
          Tunnel-Medium-Type = 6,
          Tunnel-Private-Group-ID=50
DEFAULT   Auth-Type := Reject, Colubris-AVPair == "ssid=service"
!!
service radiusd restart

cat > /path/dhcp-wifimere << !!
host wifimere {
hardware ethernet $MAC1 ;
Fixed-address 172.16.12.1 ;
}
!!
echo "Relance de dhcp"
Service dhcpd restart
```

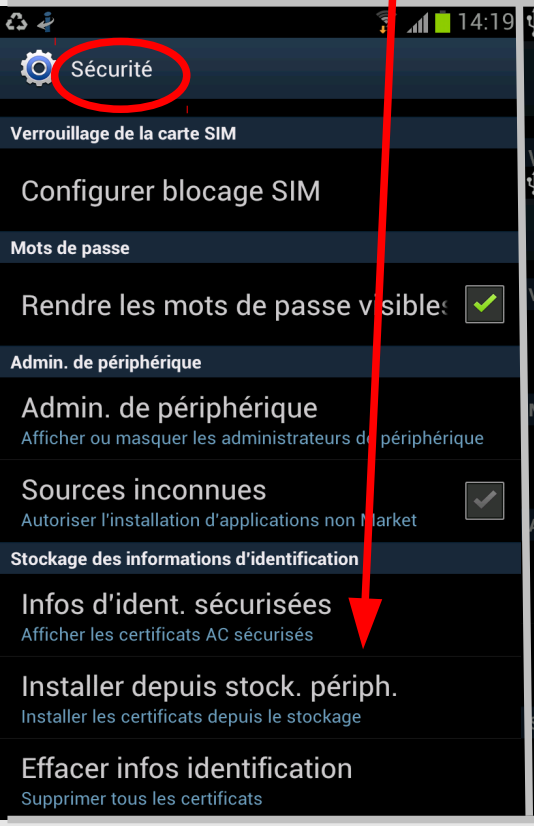
Utilisation des certificats : importation

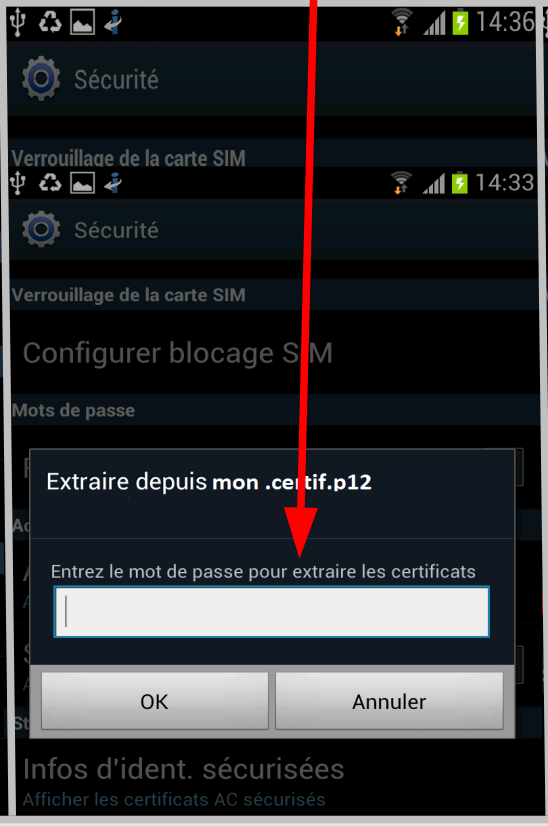
- On suppose que le fichier P12 est à la racine de la Sdcard
- Le fichier P12 est protégé par un mot de passe (utile uniquement pendant l'importation)
- C'est le Common Name (CN) contenu à l'intérieur du certificat qui servira pour l'authentification
- Le nom du fichier P12 est complètement indépendant du CN
- Le fichier P12 peut contenir le certificat de l'utilisateur et celui de l'AC

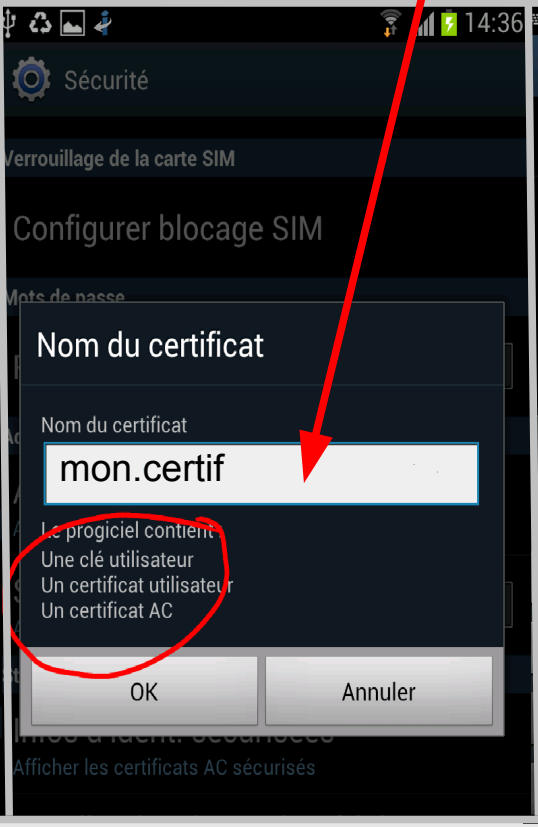


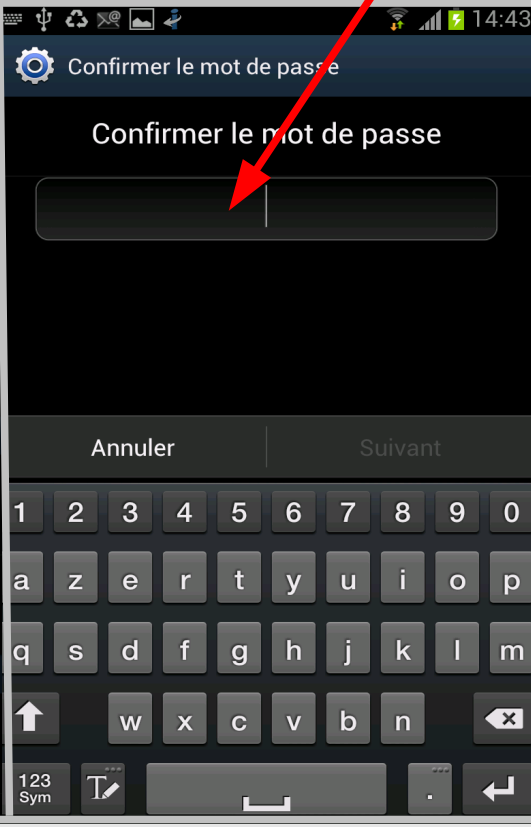
Utilisation des certificats : importation

1	2	3	4
• Paramètres/sécurité • Installer depuis stock.périph	▪ Le mot de passe du fichier p12 est demandé ▪ Il ne sera plus utilisé par la suite	▪ Le nom du fichier est pris comme nom du certificat (alias). Il ne s'agit pas du CN mais d'un nom qui permettra ensuite de repérer le certificat. ▪ Ce nom peut être changé par ce que l'on veut.	▪ Le mot de passe demandé ici est le mot de passe de verrouillage du mobile pour permettre d'écrire dans le magasin. ▪ Le certificat est enregistré dans le magasin.









Utilisation des certificats pour le Wifi (EAP/TLS)

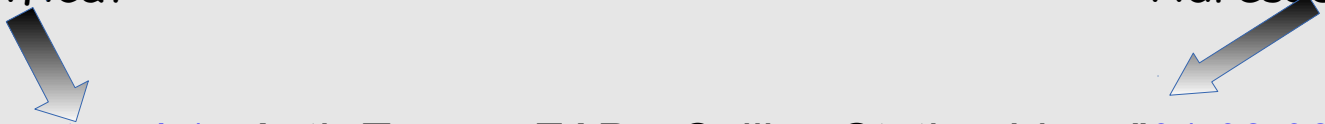
Pourquoi utiliser le protocole EAP/TLS ?

- Authentifier les mobiles avec leur certificat (et croiser avec l'adresse MAC : La connexion ne sera autorisée que si le certificat est soumis depuis un matériel qui possède l'adresse MAC enregistrée)
- Positionner les mobiles dans leur sous-réseau (pas forcément le même pour tous)
- Les mobiles authentifient aussi le certificat du serveur

L'authentification est réalisée par un serveur Radius avec une entrée du style :

CN du certificat

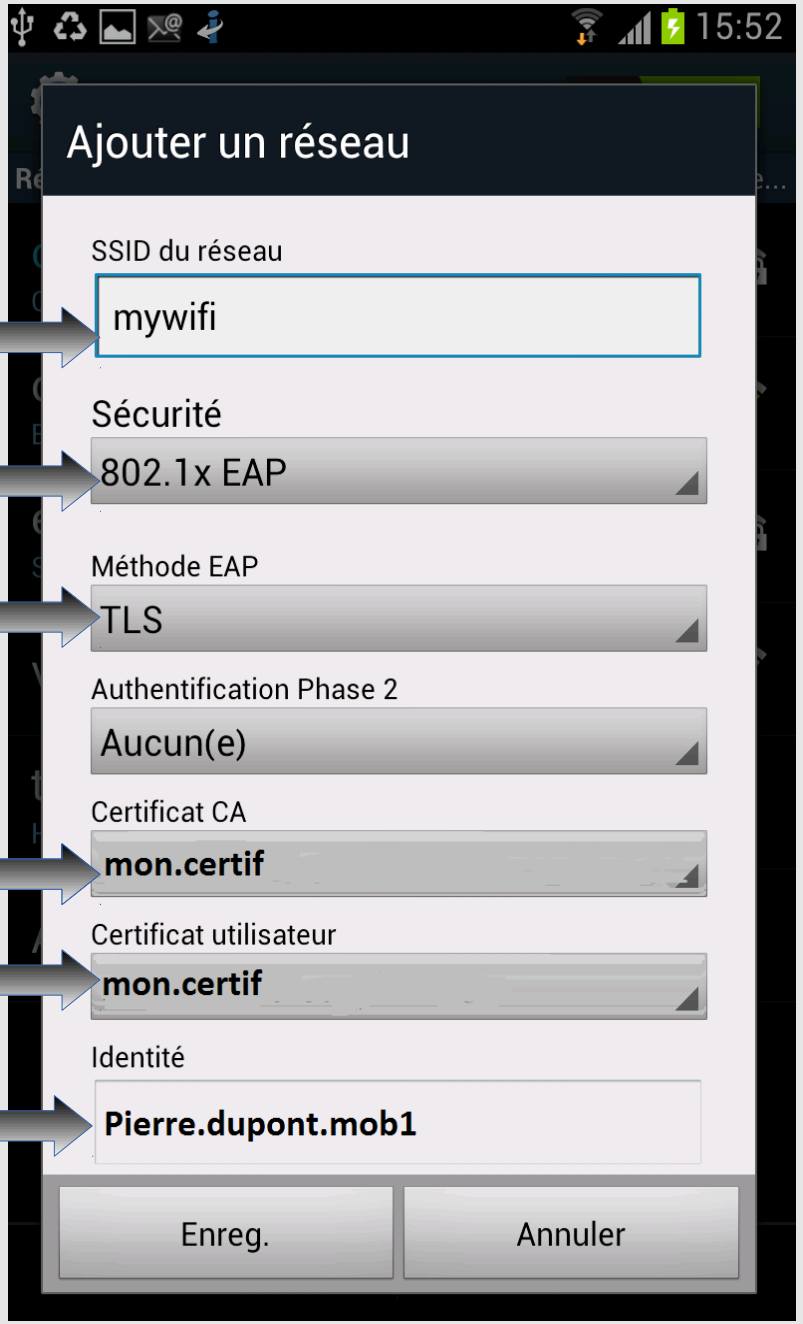
Adresse MAC du mobile



```
pierre.dupont.mob1   Auth-Type := EAP ,  Calling-Station-Id == "01:02:03:04:05:06"
        Tunnel-type  = VLAN,
        Tunnel-Medium-Type = 6,
        Tunnel-Private-Group-ID =10
```

VLAN

Utilisation des certificats pour le Wifi (EAP/TLS)



Le nom du SSID

Type de sécurité

Méthode

Certificat CA

Choix (dans la liste déroulante) du certificat à utiliser.

Le CN du certificat (c'est sur lui que se fera l'authentification)

Stockage des certificats : le keychain

- Sur un mobile non-rooté le keychain n'est accessible qu'au travers du service **keystore**.
- **Keystore** est lancé au démarrage par /init.rc

```
service keystore /system/bin/keystore /data/misc/keystore
    class main
    user keystore
    group keystore
    socket keystore stream 666
```

- Les certificats sont stockés dans **/data/misc/keystore**
- Accessible directement sur un mobile rooté (=> intérêt des MV pour tester)

```
root@android:/data/misc/keystore # ls -al
-rw------- keystore keystore      84 2013-04-15 16:02 .masterkey
-rw------- keystore keystore    1156 2013-04-15 16:24 1000_CACERT_jeanduponttest
-rw------- keystore keystore    1156 2013-04-15 16:02 1000_CACERT_pierreduponttest
-rw------- keystore keystore    1300 2013-04-15 16:24 1000_USRCERT_jeanduponttest
-rw------- keystore keystore    1300 2013-04-15 16:02 1000_USRCERT_pierreduponttest
-rw------- keystore keystore    1524 2013-04-15 16:24 1000_USRPKEY_jeanduponttest
-rw------- keystore keystore    1524 2013-04-15 16:02 1000_USRPKEY_pierreduponttest
root@android:/data/misc/keystore #
```

- Ici un certificat P12 a été importé
- *1000* est l'UID du créateur (system)
- *pierreduponttest* est le CN du certificat
- Le keystore peut être manipulé avec la commande (root) *keystore_cli*
- Les autorités de certification ajoutées sont dans **/data/misc/keychain/cacerts-added**

Protection du keychain

- Le Keychain d'Androïd ne stocke que des certificats.
- Il est chiffré par une clé dérivée du code de verrouillage.
- Ce chiffrement 'semble' assez robuste. (http://nelenkov.blogspot.fr/2011/11/ics-credential-storage-implementation.html)
- Dans Androïd 4.3 : Possibilité de « hardware-backed » pour protéger les clés (Nexus 4)

Pour aller plus loin sur les attaques possibles (Androïd et IOS)

Platform	Circumvent Simple Passcode	Circumvent Complex Passcode
Android	Sometimes, device dependent	Sometimes, device dependent
Blackberry	Rarely	Rarely
iOS	Always	Always, however complex passcodes with at least 6 alphanumeric characters can provide protection to some encrypted files

Contournement du passcode

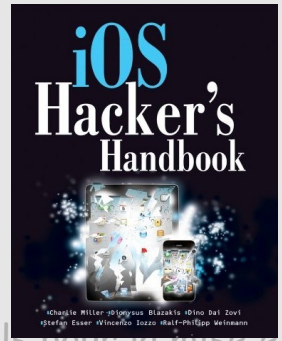
Mobility risk report :Understanding the security impact of IOS and Androïd in the enterprise
https://viaforensics.com/resources/reports/mobile-security-risk-report/

Voir aussi :Lost iPhone? Lost Passwords! Practical Consideration of iOS Device Encryption Security
Fraunhofer Institute for Secure Information Technology

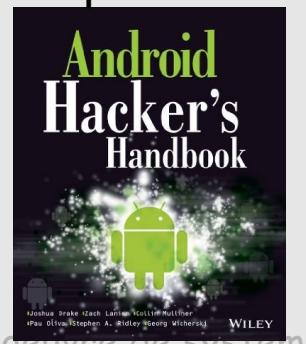
http://sit.sit.fraunhofer.de/studies/en/sc-iphone-passwords.pdf
http://sit.sit.fraunhofer.de/studies/en/sc-iphone-passwords-faq.pdf

After using a jailbreaking tool, to get access to a command shell, we run a small script to access and decrypt the passwords found in the keychain. The decryption is done with the help of functions provided by the operating system itself (…) overall approach takes six minutes.

Voir aussi



A paraître



Chiffrement des mobiles Androïd

Principes du chiffrement

Pourquoi chiffrer ?

- Pour protéger toutes les informations que contient un mobile en cas de perte ou de vol
  - Mieux protéger les « authentifiants »
  - Protéger les données utilisateurs

Caractéristiques du chiffrement Androïd

- A la livraison un mobile Androïd n'est pas chiffré
- Il n'y a pas de processeur cryptographique dans le matériel
- Androïd permet de chiffrer la partition ***/data*** et la carte SD.
- Chiffrement de surface logiciel (dmcrypt)
- Cette opération est très facile à réaliser.

Principes du chiffrement

Caractéristiques du chiffrement Androïd

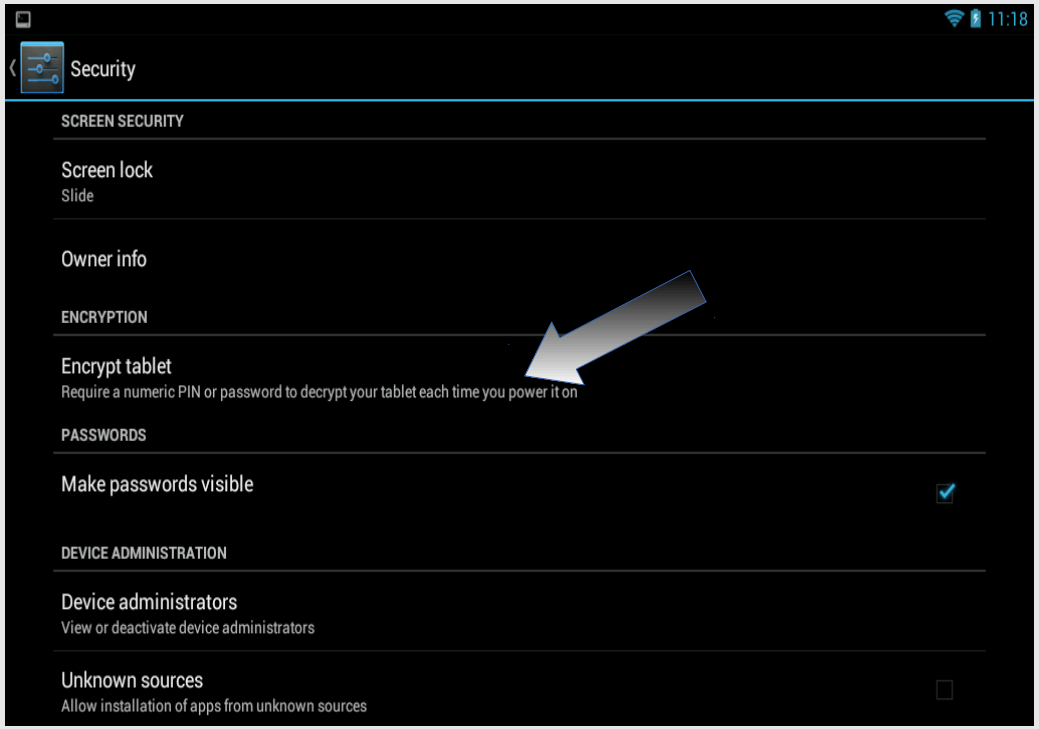
- Pour chiffrer, Androïd impose le positionnement d'un mot de passe alphanumérique pour le verrouillage du mobile (en tout cas sur Samsung).

- Mis à part ce mot de passe, le chiffrement est complètement transparent pour toutes les applications. Tous les fichiers créés par les applications sont donc protégés par le chiffrement.

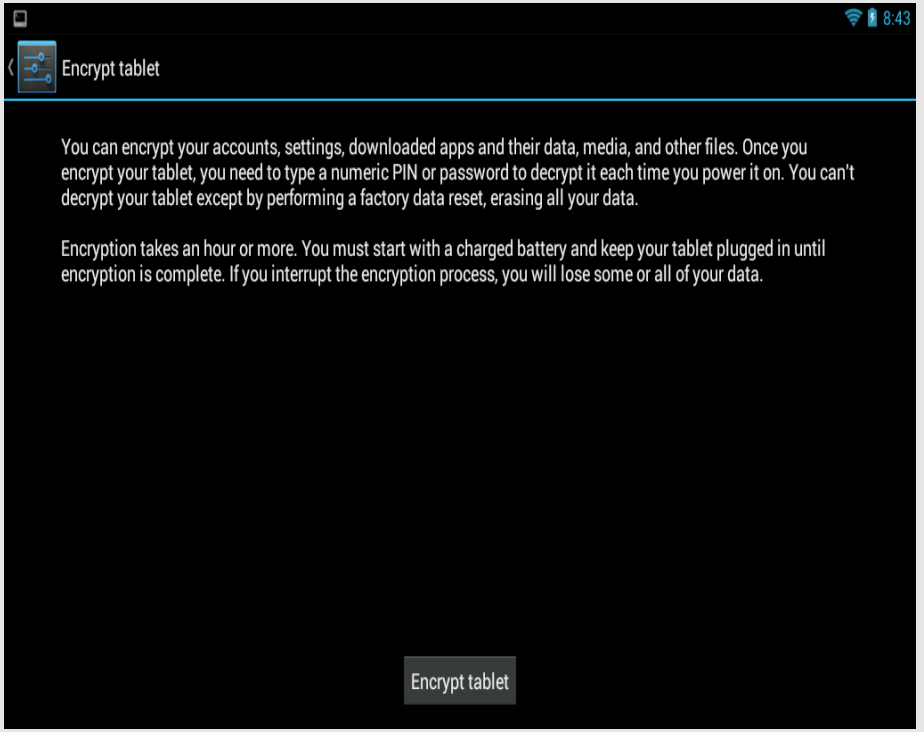
- Pas de problème de performance constaté (aujourd'hui les mobiles sont équipés d'au moins un double cœur et les plus récents de 4 voire 8)

Procédure pour chiffrer

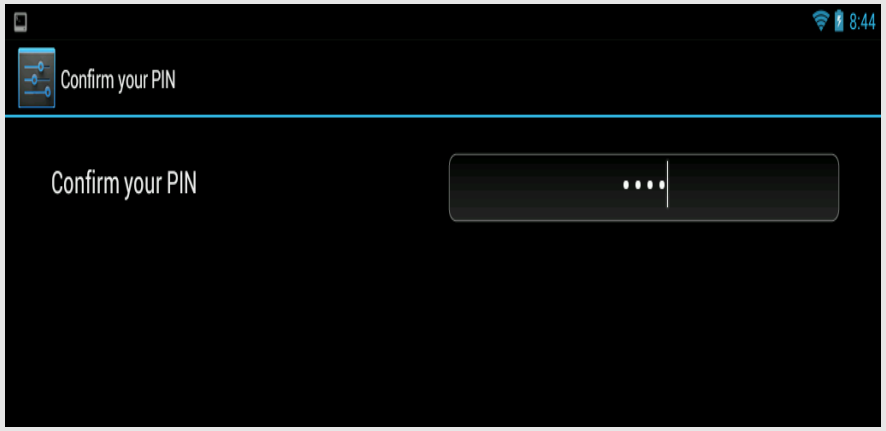
1



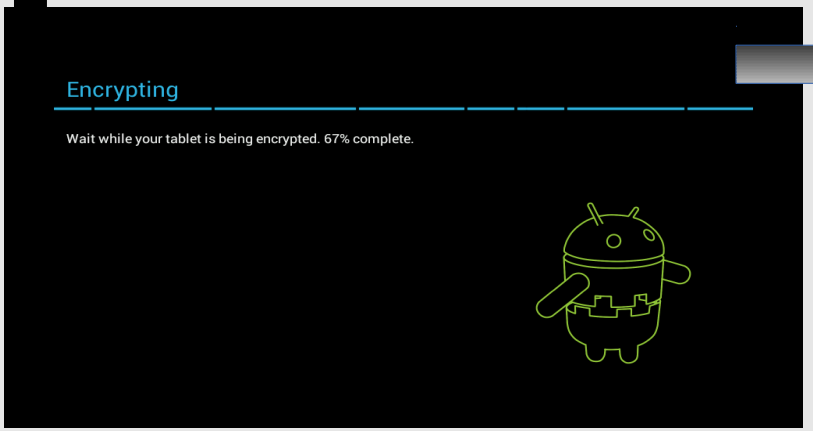
2



3



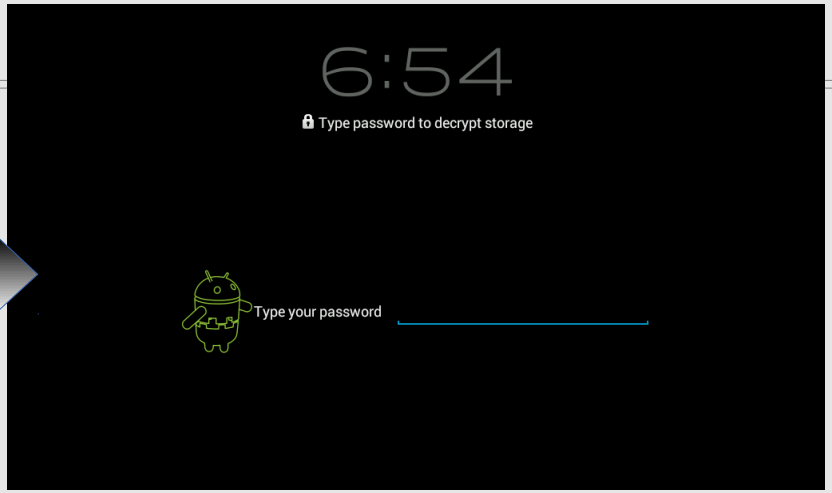
4



reboot

Procédure pour chiffrer

Après le reboot le pin code
ou mot de passe est demandé
pour déverrouiller le mobile



Montage avant chiffrement

```
root@android:/ # mount
rootfs / rootfs ro,relatime 0 0
tmpfs /dev tmpfs rw,nosuid,relatime,mode=755 0 0
devpts /dev/pts devpts rw,relatime,mode=600 0 0
proc /proc proc rw,relatime 0 0
sysfs /sys sysfs rw,relatime 0 0
tmpfs /mnt/asec tmpfs rw,relatime,mode=755,gid=1000 0 0
tmpfs /mnt/obb tmpfs rw,relatime,mode=755,gid=1000 0 0
/dev/block/sda6 /system ext4 ro,relatime,data=ordered 0 0
/dev/block/sdb1 /cache ext4 rw,nosuid,nodev,relatime,data=ordered 0 0
/dev/block/sdb3 /data ext4 rw,nosuid,nodev,relatime,data=ordered 0 0
/dev/block/sdc /mnt/sdcard vfat rw,relatime,fmask=0000,dmask=0000,allow_utime=0022,coo
root@android:/ #
```

Montage après chiffrement

```
rootfs / rootfs ro,relatime 0 0
tmpfs /dev tmpfs rw,nosuid,relatime,mode=755 0 0
devpts /dev/pts devpts rw,relatime,mode=600 0 0
proc /proc proc rw,relatime 0 0
sysfs /sys sysfs rw,relatime 0 0
tmpfs /mnt/asec tmpfs rw,relatime,mode=755,gid=1000 0 0
tmpfs /mnt/obb tmpfs rw,relatime,mode=755,gid=1000 0 0
/dev/block/sda6 /system ext4 ro,relatime,data=ordered 0 0
/dev/block/sdb1 /cache ext4 rw,nosuid,nodev,relatime,data=ordered 0 0
none /mnt/shared/android-extension vboxsf rw,nodev,relatime 0 0
/dev/block/sdc /mnt/sdcard vfat rw,relatime,fmask=0000,dmask=0000,allow_utime=0022,coo
/dev/block/dm-0 /data ext4 rw,nosuid,nodev,relatime,data=ordered 0 0
root@android:/ #
```

Quand un mobile devient fournisseur d'accès
Le mode modem

Le mode modem

A quoi ça sert ?

Permet de connecter un ordinateur sur Internet via un smartphone équipé de la 3G (ou mieux)

Comment ?

- En connectant le smartphone sur l'ordinateur en USB
  (il suffit de connecter le câble USB)

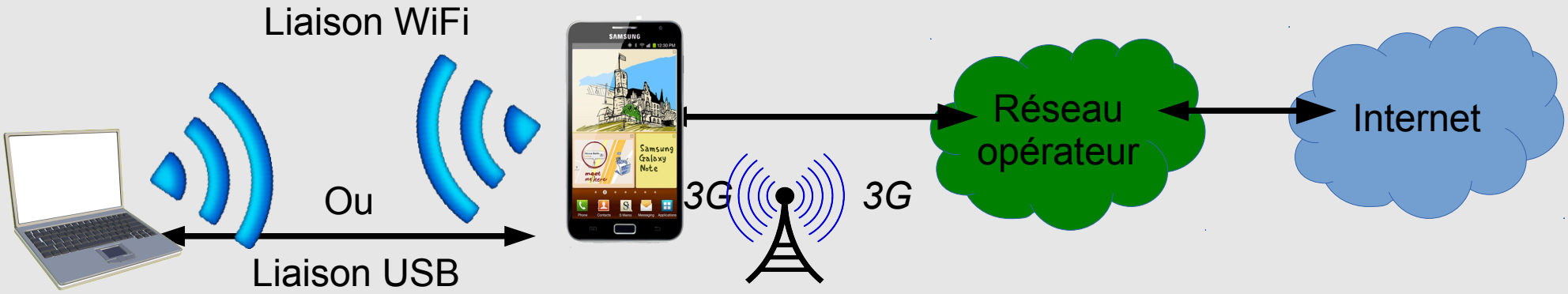
  Ou bien

- En se servant du smartphone comme une borne WiFi sur laquelle l'ordinateur se connecte

Condition ?

- Le smartphone dispose d'une connexion DATA
- L'opérateur autorise le mode modem

Le mode modem



Le mode modem

Une tierce personne peut-elle utiliser la connexion Internet du smartphone ?

- Le smartphone diffuse comme une borne Wifi
- Tous ceux qui sont à proximité captent le SSID
- Si la sécurité est « ouvert », tout le monde peut utiliser le réseau WiFI créé par le smartphone.
- Idem si le mot de passe est connu ou trop évident.

Le mode modem

Y a-t-il un risque d'interconnexion du réseau interne avec l'extérieur ?

- Cela suppose que le pirate trouve le mobile à travers Internet et le réseau de l'opérateur et qu'il soit en mode modem à ce moment-là.
- Cela suppose qu'il forge les paquets pour qu'ils soient vus comme des réponses.
- Cela suppose que le smartphone route les paquets vers l'ordinateur connecté
- Cela suppose que l'ordinateur route les paquets vers sa liaison Ethernet libre vers une cible potentiel.
- Compte tenu que le réseau ordinateur-smartphone est éphémère cela semble très compliqué, en tout cas sans une « aide » sur l'ordinateur et ou smartphone (genre virus par exemple).

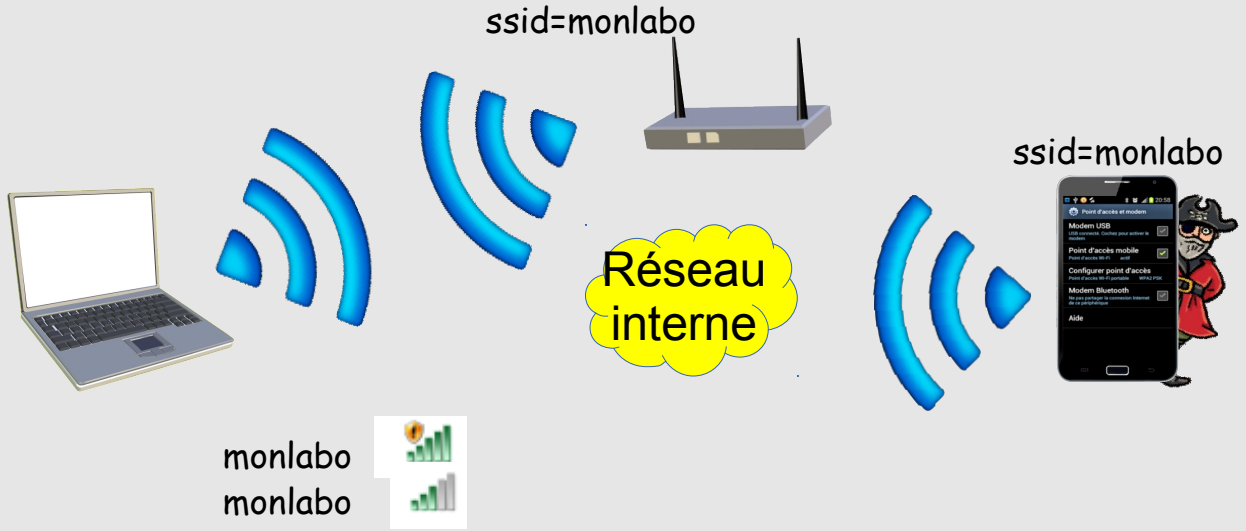
Le mode modem

Y a-t-il un risque de piratage d'un SSID du réseau local?

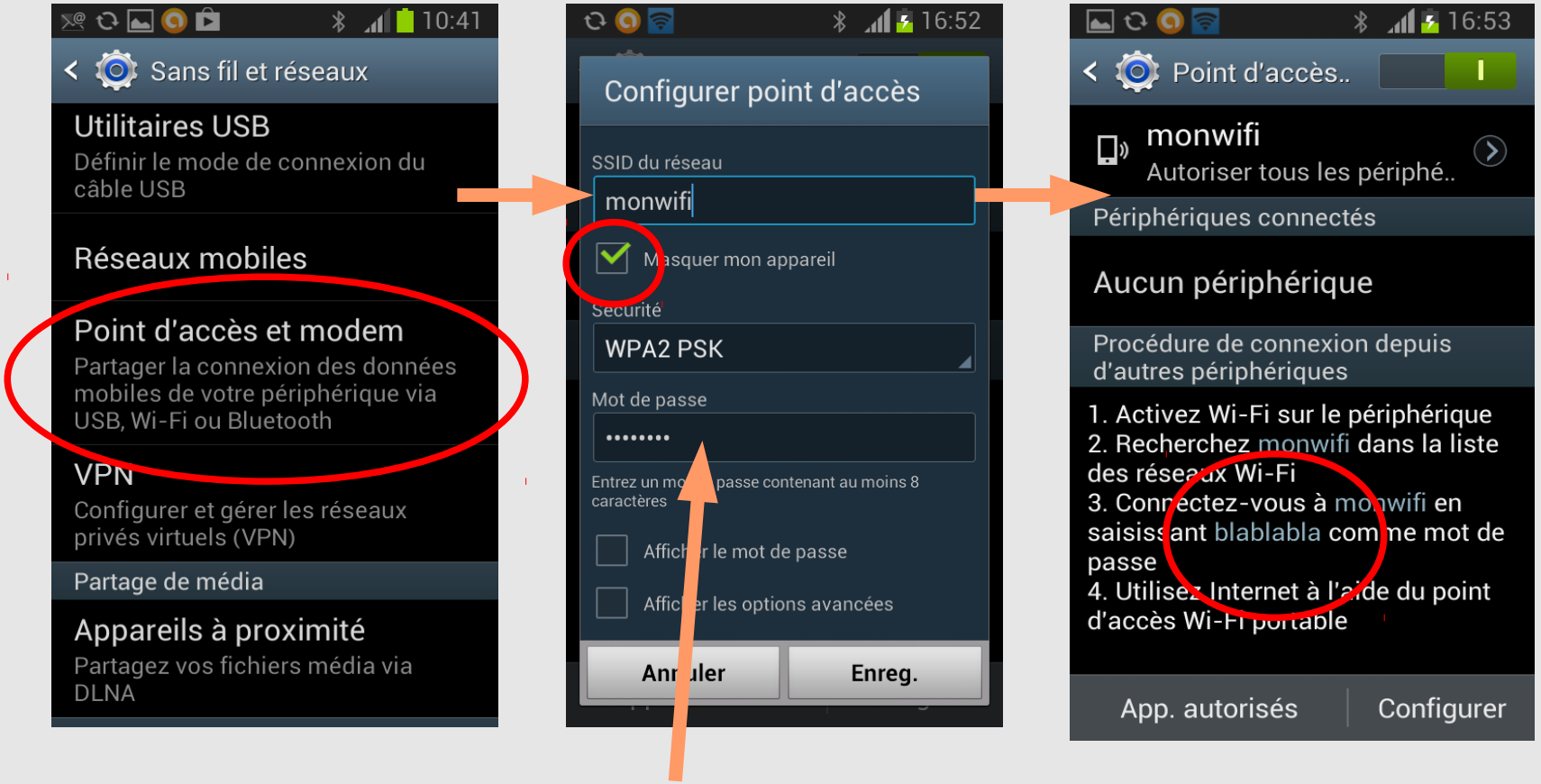
C'est probablement le plus gros risque du mode modem !!

- N'importe qui peut configurer un point d'accès avec un SSID, ouvert, de même nom qu'un SSID existant. C'est pas nouveau, mais beaucoup plus facile et accessible sans grande compétence.

- Un utilisateur, sur n'importe quel ordinateur WiFi, verra deux SSID de même nom. S'il sélectionne le mauvais il se connectera au réseau via le mobile pirate qui pourra agir comme « mobile-in-the-middle »

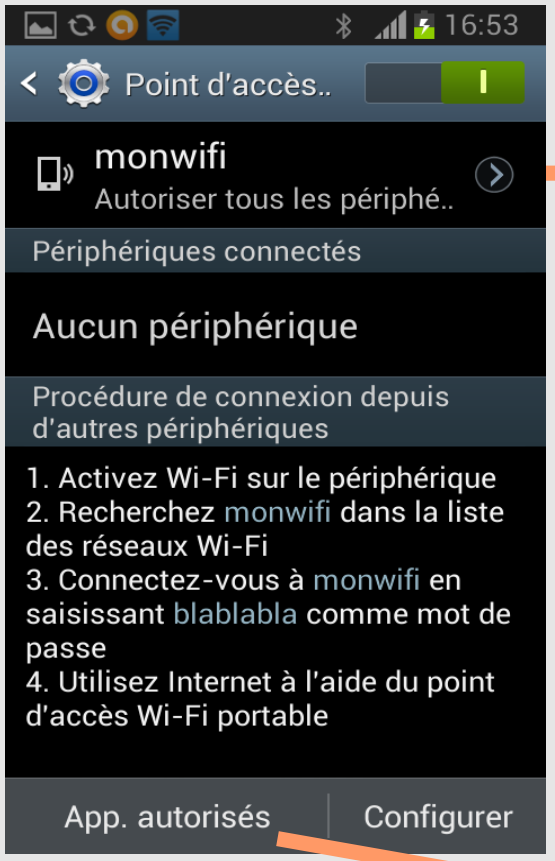


Le mode modem configuration

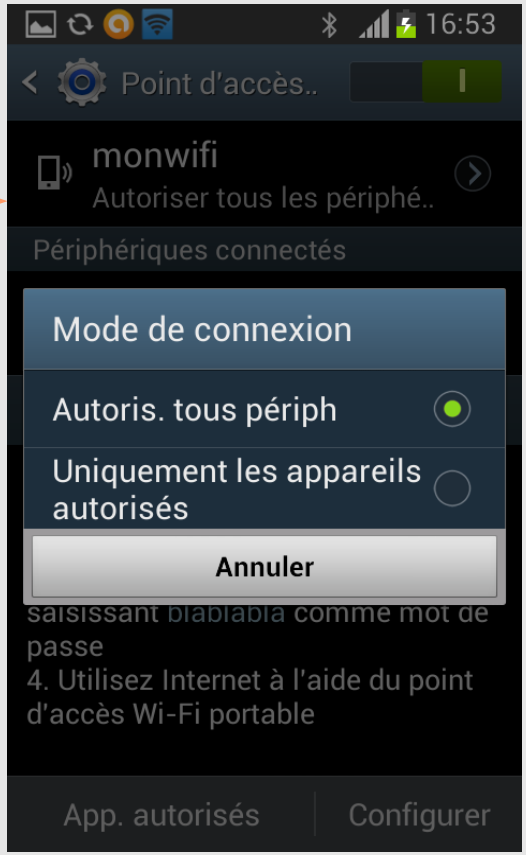


- **SSID** : Ne pas utiliser son propre nom ou un SSID existant
- **Sécurité** : Ne pas utiliser une sécurité « ouverte »
- **Mot de passe** : na pas utiliser le mot de passe d'un compte (il est stocké en clair)

Le mode modem configuration



De préférence autoriser
uniquement ses propres matériels



Traiter la sécurité SUR les mobiles n'est pas suffisant

Protection du Système d'Information et de la sphère professionnelle

Les mobiles, vecteurs de compromissions

- Nous évoluons rapidement d'une situation de rares matériels nomades, **gérés**, vers une situation d'une **multitude** de mobiles, **non gérés**, qui contiendront, **TOUS**, toutes les clés nécessaires pour attaquer le Système d'Information !! (les « credentials »).

- L'ergonomie des mobiles impose pratiquement toujours l'enregistrement des mots de passe dans les mobiles.
  - Trop compliqué de taper un mot de passe complexe
  - Processus qui tournent en arrière-plan.
  - Les applications stockent ces mots de passe de façon complètement incertaines, et probablement en clair

- Si rien n'est mis en place, rien ne prouve que l'utilisateur à un passcode ou qu'il n'est pas triviale, rien ne permet de savoir quel mobile peut se connecter, rien ne permet de savoir que devient un mobile dans le temps.

Les mobiles vont de plus en plus devenir des tirelires à mots de passe (tirelires en porcelaine)

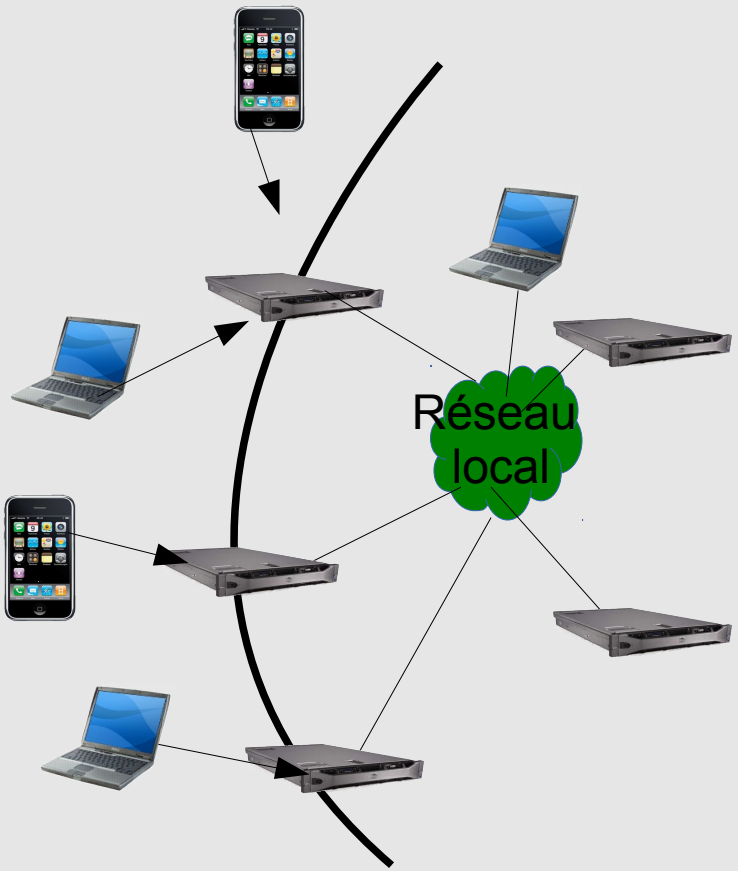
Les mobiles, vecteurs de compromissions

- Le problème ne vient pas que du côté du mobile (et peut se généraliser à tous les postes nomades)
- Les réseaux des établissements ont (ou auront) une vitrine de services accessibles depuis Internet … en général avec de simples mot de passe (messagerie, agenda, synchronisation de fichiers…)
  - Parce que le mot de passe ne coûte pas cher
  - Finalement plus pratique/facile pour l'utilisateur

- La mise en ligne directe sur Internet de services Web, souvent pleins de vulnérabilités, nécessite un niveau de réactivité élevé pour répondre aux alertes.

Mots de passe stockés + services directement sur Internet = danger

Il va peut-être falloir reconsidérer sa vitrine…



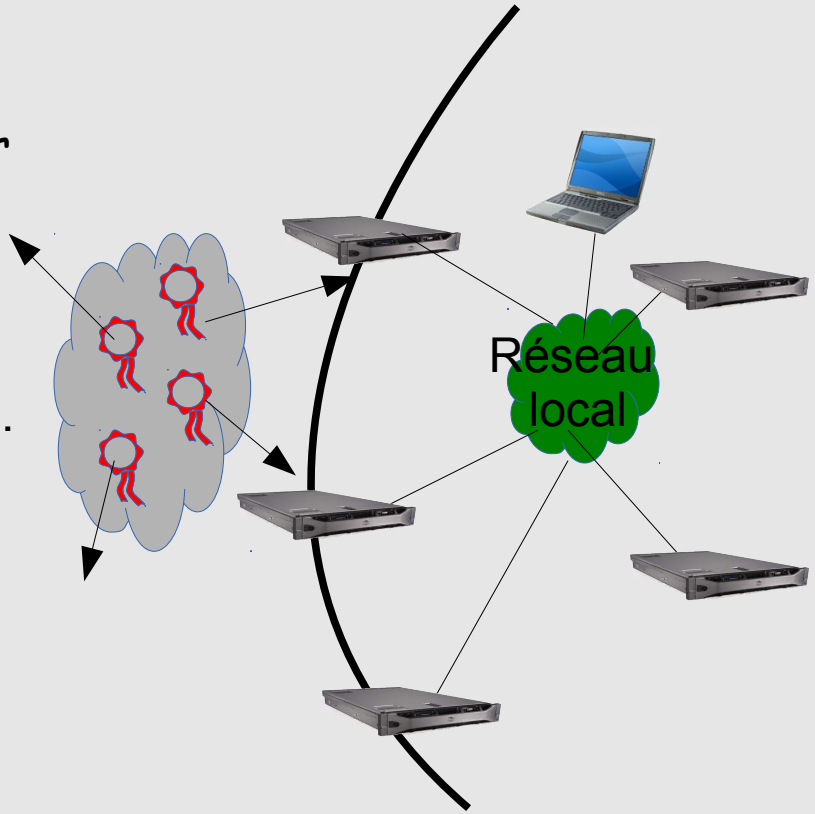
Les mobiles, vecteurs de compromissions

- Le problème ne vient pas que du côté du mobile (et peut se généraliser à tous les postes nomades)
- Les réseaux des établissements ont (ou auront) une vitrine de services accessibles depuis Internet … en général avec de simples mot de passe (messagerie, agenda, synchronisation de fichiers...)
  - Parce que le mot de passe ne coûte pas cher
  - Finalement plus pratique/facile pour l'utilisateur

- La mise en ligne directe sur Internet de services Web, souvent pleins de vulnérabilités, nécessite un niveau de réactivité élevé pour répondre aux alertes.

Mots de passe stockés + services directement sur Internet = danger

Il va peut-être falloir reconsidérer sa vitrine...

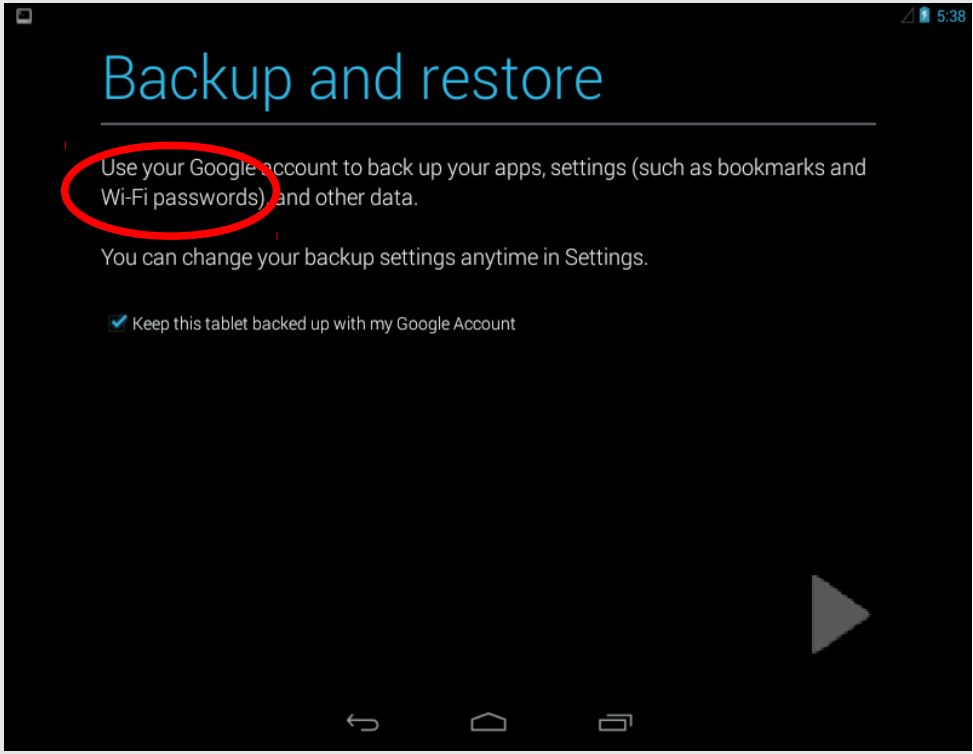


Exemple de possibilité de fuites de mot de passe

- Androïd propose une option permettant de sauvegarder les mots de passe WiFi chez Google. Si mot de passe WiFi = mot de passe du compte utilisateur => Risques

- D'autres applis peuvent proposer cela.

- Que devient un mobile lorsque l'utilisateur en change ?
  N'est-il pas transféré à une autre personne...avec les mots de passe dedans ?



Peut-on interdire les mobiles ?

- De toute façon, si rien n'est fait, à partir du moment où il suffit de configurer un login/mot de passe, rien ne peut interdire à un matériel de se connecter (et de stocker le mot de passe)

Exemple: service de messagerie IMAP

- « L'idéale » est d'encapsuler la relation avec les mobiles dans un mode professionnel

Le mode professionnel

C'est quoi ?

- Sur le mobile séparer, et protéger, ce qui est professionnel du reste
- Les services du S.I ne sont accessibles qu'après établissement d'une liaison avec authentification sécurisée
- Les services et les mots de passe ne sont utilisables qu'après établissement de cette liaison.

Mode professionnel : Principe du Silo

Consiste à créer sur le mobile un conteneur sécurisé qui protège tous les éléments professionnels, données et « authentifiants ».

Divers solutions apparaissent :

- Samsung Knox
- Citrix
- Blackberry balance

- Il s'agit de solutions propriétaires
- Plus adaptées à une gestion de parc mobile d'entreprise
- Doivent fonctionner au moins sur les environnements IOS, Androïd, Windows RT...
- Quid du coût ?
- Quid de l'infrastructure côté SI ?

- Manque de maturité et d'expérience des offres
- Pérennité pas évidente

Mode professionnel : Principe du Silo

Le silo est-il efficace ?

- Oui, si les services ne sont accessibles qu'à travers le silo
- Non, s'il reste des services accessibles avec un simple login/mot de passe parce que dans ce cas les mobiles n'ont pas besoin du silo pour se connecter aux services (et donc stockent les mots de passe de façon non-sécurisée)

Un exemple d'utilisation de la méthode des « tokens aléatoires »

Un exemple d'utilisation de la méthode des « tokens aléatoires »

- **Contexte préexistant**

Le service de messagerie (IMAP) n'est accessible qu'au travers d'une connexion VPN (IMAP pas ouvert directement sur l'extérieur)

- **Problème à résoudre**

  - Fournir le même accès à la messagerie aux mobiles sans affaiblir la sécurité déjà existante et sans effet tirelire (pas de mot de passe enregistré)
  - Quel que soit le client de messagerie
  - Androïd ou IOS
  - 0 €
  - Unification des connexions depuis des ordinateurs « classiques » et mobiles
  - Méthode utilisable pour d'autres applications

Un exemple d'utilisation de la méthode des « tokens aléatoires »

- **Bénéfices**
  - Connaissance complète du parc autorisé à interagir avec le réseau du labo, même les BYOD
  - Amélioration de la sécurité pour l'ensemble des nomades y compris les portables « classiques »
  - Ré-utilisation des technologies déjà utilisées
  - Pas d'installation lourde ou intrusive sur les mobiles.
  - A terme, toutes les machines nomades utiliseront le même principe

Un exemple d'utilisation de la méthode des « tokens aléatoires »

- **Méthode générale**

  - L'utilisateur de se connecte via le serveur OpenVpn avec une authentification à double facteur :
    - Un certificat (ce que je possède)
    - Un mot de passe variable, donc non enregistré sur le mobile, indépendant du certificat, validé par le serveur (le secret que je connais)

  - L'utilisateur configure et utilise son application mail (imap) qu'au travers du VPN. Le mot de passe mail est inutilisable sans la connexion VPN

Un exemple d'utilisation de la méthode des « tokens aléatoires »

- **Connexion VPN, avec (le fameux) password variable**

La question à résoudre est : comment le client et le serveur VPN peuvent s'accorder sur mot de passe variable ?

En réalité le mot de passe est constitué de deux parties : une partie secrète et fixe et une partie variable. Il n'est pas utile que client et serveur s'accorde sur cette partie variable. Il suffit que le serveur vérifie qu'elle change d'une connexion sur l'autre.
Client et serveur doivent simplement s'accorder sur le format du mot de passe. Par exemple : chiffres suivis de deux caractères aléatoires.

12345XY

Un exemple d'utilisation de la méthode des « tokens aléatoires »

- **Enregistrement des mobiles**
  - Les mobiles sont enregistrés, comme n'importe quelle machine, pour utiliser le réseau Wifi (EAP/TLS) et l'accès par VPN.
  - Chaque mobile reçoit un certificat spécifique et unique qui l'identifie et installé par le service informatique

  - Le certificat possède une date d'expiration couvrant ce qu'on estime être la durée de vie du mobile
  - L'enregistrement (dans Radius) est valable un an (l'utilisateur reçoit un mail pour lui demander s'il utilise toujours ou plus le mobile).
  - Le même certificat est utilisé pour la connexion Wifi en interne et l'accès VPN.
  - Pour l'accès VPN, l'utilisateur devra aussi utiliser un passcode enregistré dans la table Radius (chiffré).

Un exemple d'utilisation de la méthode des « tokens aléatoires »

- **Enregistrement des mobiles : bénéfices**
  - Tous les mobiles sont connus
  - Unification des méthodes de connexion (le mobile est un ordinateur comme les autres)
  - Le fait de mettre un certificat sur le mobile oblige l'utilisateur à positionner un code de verrouillage sur le mobile.
  - Avec les dates d'expiration, un mobile non utilisé, perdu et non signalé, fini par ne plus pouvoir se connecter, même en connaissant le passcode VPN.

Un exemple d'utilisation de la méthode des « tokens aléatoires »

- **Enregistrement des mobiles : Concrètement dans les tables Radius**

CN du certificat

Adresse MAC du mobile

pierre.dupont.tab Auth-Type:= EAP, Calling-Station-Id == « AA-BB-CC-DD-EE-FF », Expiration:= « 12 Mar 2014 »...
Tunnel-Private-Group-ID = v

VLAN

CN du certificat

Partie secrète du passcode VPN (chiffrée)

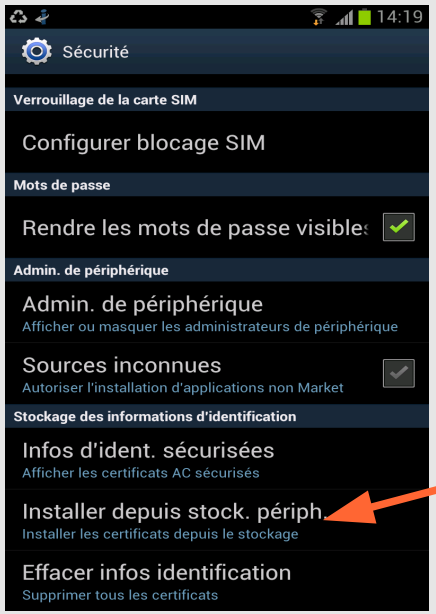
pierre.dupont.tab Crypt-password:= cygbsz678çka , NAS-Port:= v, expiration:= « 12 Mar 2014 » .....
Framed-IP-Address=a.b.c.d

Adresse IP renvoyée

Un exemple d'utilisation de la méthode des « tokens aléatoires »

- **Configuration sur le mobile, par le service informatique**

1) L'application Androïd ou IOS Openvpn Connect est installée

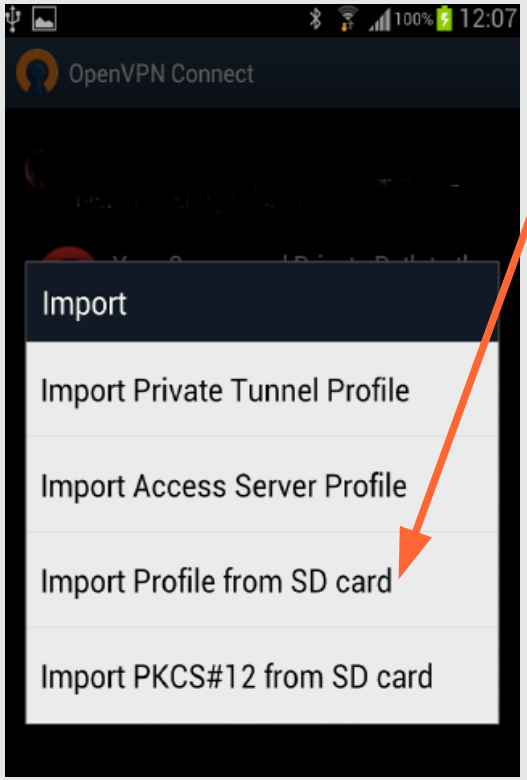


2) Le certificat du mobile est téléchargé et importé dans le keychain.

Un exemple d'utilisation de la méthode des « tokens aléatoires »

- **Configuration sur le mobile, par le service informatique**

3) La configuration Openvpn est téléchargée sur le mobile et importée dans Openvpn



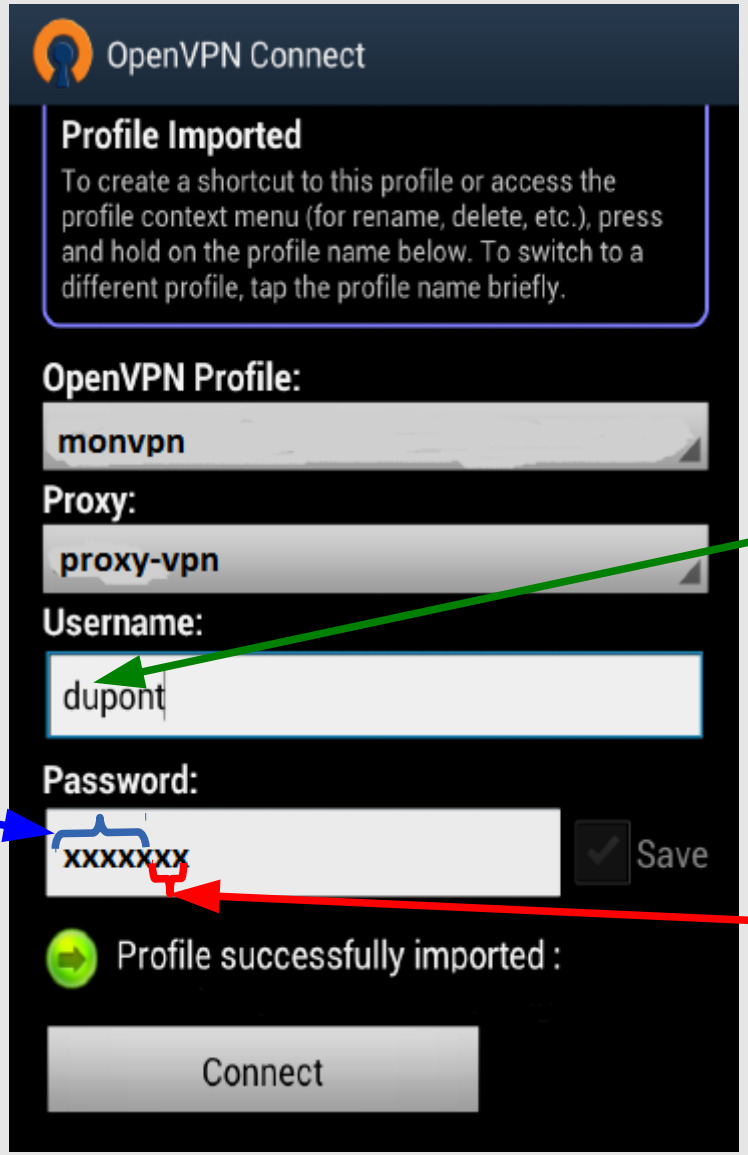

```
client
verb 2
connect-retry-max 5
resolv-retry 5
dev tun
http-proxy proxy-vpn.labo.fr 443
remote vpn.labo.fr pppp  tcp
auth-user-pass
management-external-key
comp-lzo
setenv ALLOW_PASSWORD_SAVE 0
auth-nocache
ns-cert-type server
reneg-sec 0
```

Port du daemon Openvpn pour cet utilisateur

Interdiction de stockage du mot de passe (mesure supplémentaire mais pas suffisante)

Un exemple d'utilisation de la méthode des « tokens aléatoires »

- **Connexion VPN, avec mot de passe variable, vue du côté client**



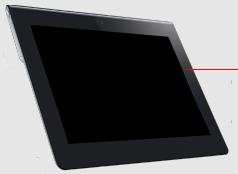
N'importe quoi car c'est le CN qui sera utilisé comme identité

Partie secrète

Partie variable aléatoire

Un exemple d'utilisation de la méthode des « tokens aléatoires »

- **Connexion VPN, avec passcode variable, vue du côté serveur(s)**



certificat+mot de passe

Tunnel chiffré

Serveur OpenVpn

Serveur Radius

➔ Authentification du certificat
- ➢ Vérifie expiration
- ➢ Vérifie liste de révocation

➔ Appel du script Auth-user-pass-verify qui reçoit le mot de passe en paramètre
- ➢ Extraction de la partie variable
- ➢ Comparaison avec la partie variable précédemment utilisée et stockée dans un fichier temporaire.
- ➢ Rejet si identique, sinon continue.

Fonction standard d'Openvpn

➔ Appel du script Client-connect
- ➢ Interroge le serveur Radius avec le CN et la partie fixe et secrète du mot de passe

➔ Interroge sa base de donnée avec CN+partie secrète

➔ Rejette ou renvoi l'adresse IP

Rejette ou affecte l'adresse IP

Un exemple d'utilisation de la méthode des « tokens aléatoires »

- **Configuration du client de messagerie**

1) Le client ouvre d'abord le VPN

2) Il entre dans les paramètres de configuration de son application mail

3) Dans le champ mot de passe, il tape n'importe quoi, aléatoirement sur son clavier. Ce mot de passe sera appelé TOKEN. Il est spécifique à ce mobile et pour cette application (imap). L'utilisateur n'a pas besoin de s'en souvenir, car le token est enregistré.

Le token n'est utilisable qu'après ouverture du VPN.

Un exemple d'utilisation de la méthode des « tokens aléatoires »

- **Configuration du client de messagerie**

Que se passe-t-il si l'utilisateur veut configurer un autre mobile puisqu'il ne connaît pas son token ?

- Dans ce cas, le 2ème mobile sera enregistré avec un autre certificat
- Il se connectera au VPN avec ce certificat et le même passcode VPN
- Il configura le mail de la même manière en tapant un token aléatoirement

Que se passe-t-il si l'utilisateur veut reconfigurer son appli mail sur un mobile déjà enregistré ?

- Il se connecte sur le VPN
- Re-configure son client de messagerie en indiquant une chaîne convenue pour reseter le token (par exemple $$)
- Une fois resetté il recommence la configuration en tapant un nouveau token

Un exemple d'utilisation de la méthode des « tokens aléatoires »

- **Connexion du client de messagerie, côté serveur(s)**

Login mail+token aléatoire
Tunnel chiffré

Serveur OpenVpn

Serveur imap

Authentification PAM
/etc/pam.d/imap

Auth requisite pam_radius_auth.so

Adr.IP VPN+login+imap

Serveur Radius

Script (bash) token_management

➔Le triuple IPvpn+login+appl est-il connu ?
Non => il est inscrit avec le token (chiffré) dans une table dynamique d'utilisateurs.
Oui => vérifie le token dans la table dynamique d'utilisateurs. Si c'est le bon token renvoi access-accept sinon access-reject . Si le token est reset => le triuple est retiré de la table dynamique.

login

Adresse ip vpn du client

Dupont auth-Type:= Accept, Calling-station-Id =~ « a.b.c.d »
Exec-Program-Wait = « /path/token_management »

Script de traitement

Moyens utilisés

- Ré-utilisation du dispositif déjà existant
  - un serveur VPN
  - un serveur Radius
  - Une IGC local
- Modification d'une ligne dans la configuration PAM du serveur de mail
- Une quinzaine de ligne en BASH dans le serveur Openvpn (vérification de la partie variable)
- Une trentaine de ligne de BASH dans le serveur Radius (enregistrement des tokens)

Avantages

- Ne dépend pas de l'application cliente
- Applicable pour d'autres applications

- Connaissance du parc de mobiles
  - Re-confirmation annuelle
  - Expiration des certificats : un mobile qui n'est plus utilisé perd la validité de son certificat.
- On sait qui se connecte, mais aussi avec quoi.