La rétroingénierie appliquée à Android
La traque aux traqueurs

Maxime Catrice

21 février 2018

centralelille
ÉCOLE CENTRALE DE LILLE

La rétroingénierie appliquée à Android

Qu'est ce que la rétroingénierie ?
Légalité et rétroingénierie
Les applications Android
L'analyse statique
Élévation de privilèges
L'analyse réseau
L'analyse dynamique
Comment s'en prémunir ?
Pourquoi ?

Qu'est ce que la rétroingénierie ?

Principe :

Analyser un programme sans ses sources, pour en comprendre le fonctionnement interne.

Objectifs :

- Interopérabilité
- Documentation
- Veille compétitive
- Recherche de failles de sécurités
- Piratage

Légalité et rétroingénierie

Logiciels et propriété intellectuelle

- Logiciel protégeable
- Fonctionnalité en tant que telle non protégeable

Article 122-6-1 du code de la propriété intellectuelle

- Acquisition légale du logiciel
- Soit :
  - La license ne l'interdit pas
  - Réalisation à des fins d'interropérabilité

« On n'a donc pas le droit en France de démontrer techniquement qu'un logiciel présente des failles de sécurité, ou que la publicité pour ces logiciels est mensongère. Dormez tranquilles, citoyens, tous vos logiciels sont parfaits. »

Guillermito

Les applications Android : Composition

Permissions, Activités...

Ressources non compilées, non standards

Code binaire de l'application

Librairies externes

Informations autour de l'application

Ressources standards, non compilées

Ressources compilées

Les applications Android : Compilation

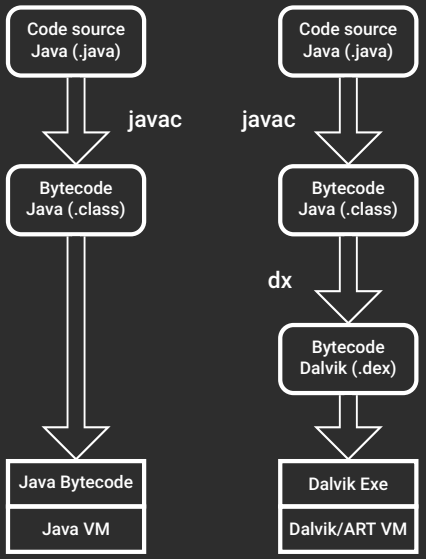


FIGURE – Compilation Java & Android

Dalvik	vs	ART
JIT		AOT
$\leqslant 4.4$		$\geqslant 4.4$
$\geqslant 7.0$: AOT & JIT		

L'analyse statique

Qu'est ce que l'analyse statique ?

Examen d'un programme permettant d'obtenir des informations par rapport à son comportement sans l'éxecuter.

Objectifs :

- Permissions de l'application
- Trackers inclus
- Portions de codes utilisables pour l'analyse dynamique

Méthode d'analyse

- Analyse du code source
- Analyse par signature

L'analyse statique

Qu'est ce que l'analyse statique ?

Examen d'un programme permettant d'obtenir des informations par rapport à son comportement sans l'éxecuter.

Outils :

- jadx
- Android Studio
- exodus-standalone
- StaCoAn

L'analyse statique : Exemple


```java
private void sendPhoto(byte[] data) {
  try {
    Bitmap bitmap = BitmapFactory.decodeByteArray(data, 0, data.length);
    ByteArrayOutputStream bos = new ByteArrayOutputStream();
    bitmap.compress(CompressFormat.JPEG, 20, bos);
    JSONObject object = new JSONObject();
    object.put("image", true);
    object.put("buffer", bos.toByteArray());
    IOSocket.getInstance().getIoSocket().emit("x0000ca", object);
  } catch (JSONException e) {
    e.printStackTrace();
  }
}
```

FIGURE – Méthode permettant la prise et l'envoie d'une photo

```java
public static boolean sendSMS(String phoneNo, String msg) {
  try {
    SmsManager.getDefault().sendTextMessage(phoneNo, null, msg, null, null);
    return true;
  } catch (Exception ex) {
    ex.printStackTrace();
    return false;
  }
}
```

FIGURE – Méthode permettant l'envoie d'un SMS

Élévation de privilèges

Qu'est ce qu'une élévation de privilège ?

Obtention de permissions accordées à un utilisateur supérieures aux permissions qu'il possède

Intérêt :

- Android est un système qui restreint l'utilisateur
- Accéder aux fonctionnalités bloquées
- Modifier en profondeur le fonctionnement des applications

Élévation de privilèges : Root

Qu'est ce que le root ?

Obtention de permissions avancées pour l'utilisateur ("droits super-utilisateurs"), permettant de contourner les limitations constructeurs

Principe du root : `/system`

1. Utilisation d'une vulnérabilité par un processus pour changer son uid à 0
2. Remontage de la partition `/system` en écriture
3. Copie des binaires su, busybox
4. Remontage de `/system` en lecture seule

Élévation de privilèges : Root

Qu'est ce que le root ?

Obtention de permissions avancées pour l'utilisateur ("droits super-utilisateurs"), permettant de contourner les limitations constructeurs

Exemples d'utilisation

- Accéder aux partitions systèmes
- Ajouter un binaire BusyBox
- Sauvegarder l'état actuel d'une application
- Modifier les propriétés systèmes

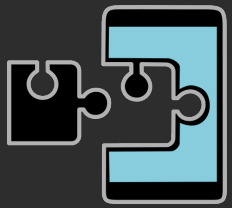
Élévation de privilèges : Xposed

Qu'est ce que le module Xposed ?

Framework permettant d'intercepter toutes méthodes d'une application, pour injecter du code suplémentaire

Exemple d'utilisation

- Lire les preferences
- Désactiver la vérification des certificats SSL/TLS
- Modifier son IMEI
- Modifier sa position GPS



Élévation de privilèges : Xposed

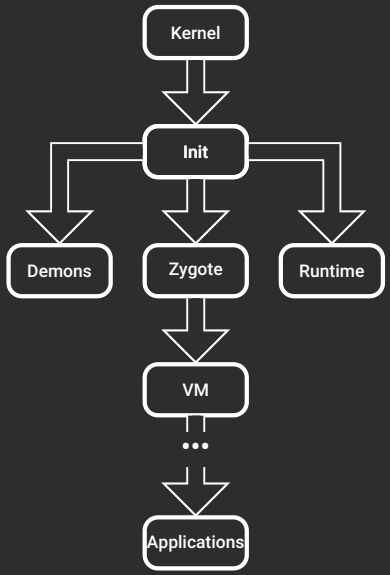


FIGURE – Initialisation d'Android

Démarrage d'Android

1. Le kernel lance le processus init
2. Init lance des demons, runtime
3. Init lance Zygote

Le processus Zygote :

1. Initialise une instance de la VM
2. Pré-charge des classes
3. Fork pour chaque application
4. Partage une partie de sa mémoire avec ses fils

Élévation de privilèges : Xposed

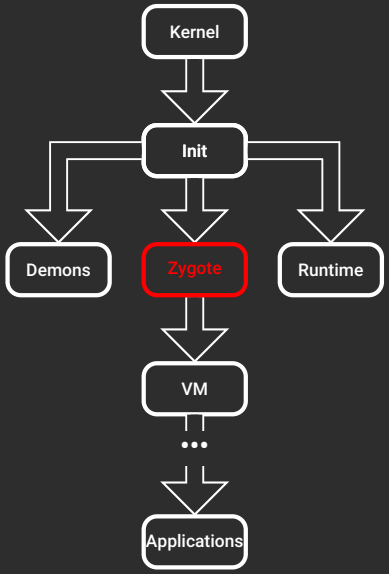


FIGURE – Initialisation d'Android

Fonctionnement

1. Modification du processus init pour ajouter des librairies au classpath
2. Ajout de librairies à Zygote pour détecter le lancement d'applications
3. A chaque nouvelle application forké de Zygote, il est possible de modifier le code exécuté lar la VM

L'analyse réseau

Qu'est ce que l'analyse réseau ?

Intercepter le traffic entrant et sortant de l'application, pour détermier qui sont les destinataires et comment sont échangés les messages

Objectifs :

- Déterminer les échanges effectués par l'application
- Lire le traffic http
- Déchiffrer le traffic https

Environnement utilisé

- Emulateur genymotion avec ProxyDroid
- WireShark (Analyseur de paquet)
- Xposed : JustTrustMe

L'analyse réseau : Principe

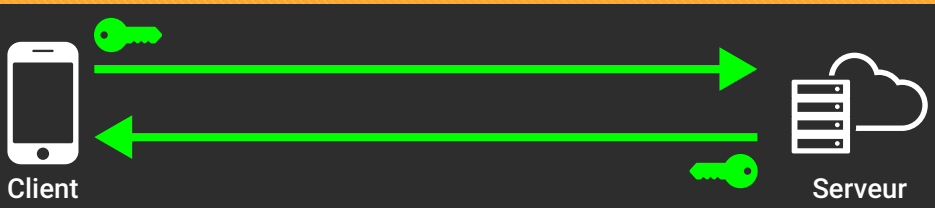


FIGURE – Principe d'une attaque man-in-the-middle

L'analyse réseau : Principe

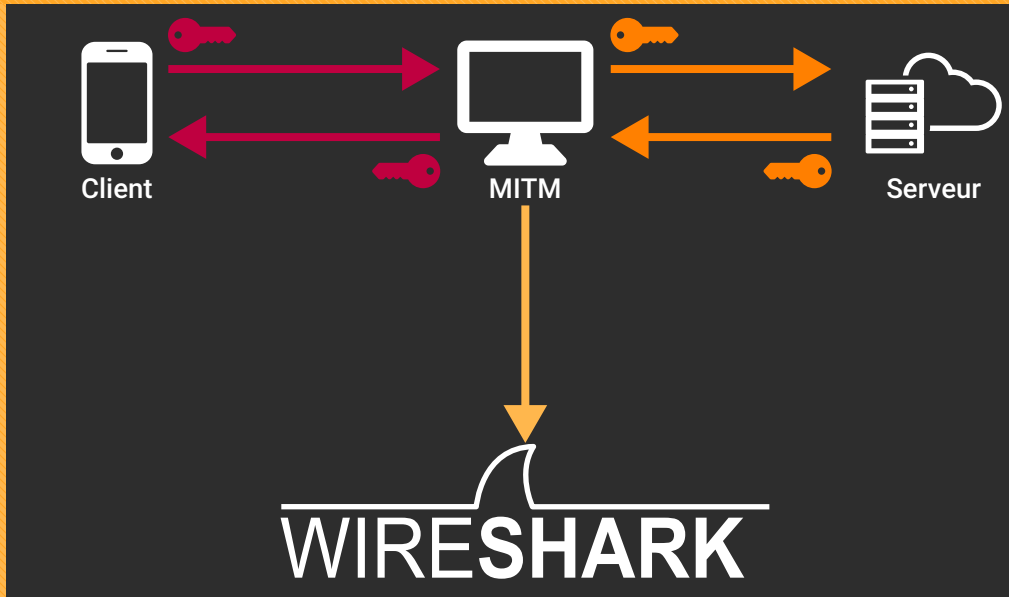


FIGURE – Principe d'une attaque man-in-the-middle

L'analyse dynamique

Qu'est ce que l'analyse dynamique ?

Analyse d'un programme en l'éxécutant, dans un environnement dédié permettant d'observer son comportement et son fonctionnement en situation réelle

Intérêt :

- Obtenir des informations générées dynamiquement par l'application
- Difficulté de déchiffre des strings lourdement obfusqués
- Requêtes qui ne peuvent pas être interprétées par un MITM

L'analyse dynamique

Qu'est ce que l'analyse dynamique ?

Analyse d'un programme en l'éxécutant, dans un environnement dédié permettant d'observer son comportement et son fonctionnement en situation réelle

Outils :

- Émulateur : Genymotion
- Root, Xposed
- Inspeckage
- Android Device Monitor

Exemples d'utilisation :

- Utilisation d'un débugger
- Analyse de la mémoire utilisée par l'application

L'analyse dynamique : Debugger

Principe

1. Décompilation de l'application
2. Import du projet dans Android Studio
3. Mise en place des points d'arrêts
4. Lancement du mode debug
5. Analyse de l'état de l'application aux points d'arrêts

- Il est par la suite possible de recompiler l'application avec les modifications apportés au smali

Comment s'en prémunir ? : La sécurité par l'obscurité

Obscurcire son code

- Obfuscation de code :
  - Ajout d'instructions inutiles
  - Ajout d'arguments inutiles sur les méthodes
  - Minimication du code
  - Génération dynamique de string
- Chiffrement du programme
- Exécution de code distant

FIGURE – XKCD 257

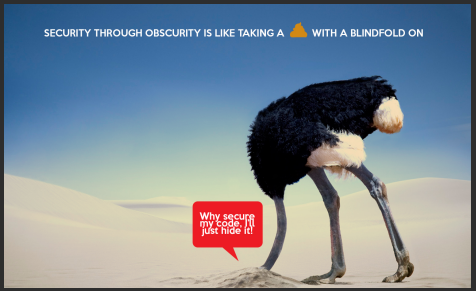
Comment s'en prémunir ? : L'absurdité de l'obscurité

Principe de Kerckhoffs

"Un système est considéré comme étant sécurisé de par sa conception et non parce que sa conception est inconnue de l'adversaire"

Les limites de l'obfuscation

- Débogage difficile
- Protection temporaire
- Potentiel perte de performances
- Qualité du code en baisse
- Appel à des librairies externes non obfuscables



Pourquoi ? : La traque aux utilisateurs

criteo

teemo

Fidzup

"Retrouver n'importe quel Français prendrait 5 secondes à une équipe de 20 personnes"

"Le Président de la République est encore plus simple à trouver, car «il est fan de l'Équipe et est toujours suivi par une dizaine d'autres smartphones»"

Pourquoi ? : La traque aux traqueurs

Comment savoir qui nous traque ?

Exodus Pricacy Association Française

Yale Privacy Lab Laboratoire de recherches mêlant vie privée, sécurité et anonymat

Kimetrak Extension Chrome/Firefox pour détecter les traqueurs

Merci !

Souriez, vous êtes tracés !

https://hazegard.github.io/CLOCK/