Étude du protocole numérique de puces GPS embarquées

Les appareils embarqués, particulièrement les téléphones « smartphones » et tablettes, sont généralement pourvus de puces GPS permettant de fournir une localisation de l'appareil avec une précision variant de 3 à 50 mètres.

Ces puces sont généralement assimilables à des microcontrôleurs autonomes, reliés au processeur principal (qui est le plus souvent un Système sur un puce : System on a Chip — SoC en anglais) au travers d'un port d'entrée-sortie numérique (le plus souvent, une liaison série type UART).
La puce GPS utilise donc un protocole particulier pour communiquer avec le SoC. Alors que certains protocoles disposent de documentation publique et d'implémentations libres, on se propose d'étudier certaines puces pour lesquelles le protocole est tenu secret et n'est pas, à ce jour, documenté publiquement.

Il s'agit donc de comprendre et documenter le protocole mis en œuvre dans l'une des puces GPS proposées : la puce BCM4751 de Broadcom et la puce GSD4t de SiRF. Une implémentation propriétaire du logiciel côté processeur principal est disponible (sous forme binaire uniquement) et servira de base pour la compréhension du protocole. Il sera donc nécessaire d'analyser le comportement de cette implémentation, au travers de plusieurs approches :

- En effectuant une trace de l'exécution normale du logiciel
- En effectuant une trace de l'exécution du logiciel en contrôlant l'ensemble des commandes envoyées au logiciel
- En analysant les sorties de débogage du logiciel (c'est à dire les logs)
- En émulant la puce en envoyant des réponses capturées (replay) et en altérant les réponses
- Par une analyse statique du binaire de l'implémentation (c'est à dire de le décompiler)
- En écrivant une implémentation (incomplète) du protocole et analysant les réponses de la puce

Les puces en question sont présentes sur un grand nombre d'appareils utilisant le système d'exploitation Android, sur lequel le développement aura lieu. On utilisera le code source du système Replicant, dérivé entièrement libre d'Android comme base de développement. Plus particulièrement, la puce GPS BCM4751 est présente sur les appareils suivants, pris en charge par Replicant :

- Google Nexus S, Samsung Galaxy S, Samsung Galaxy S3, Samsung Galaxy Note, Samsung Galaxy Tab 2

La puce GSD4t est quand à elle présente sur les modèles suivants, pris en charge par Replicant :

- Google Galaxy Nexus, Samsung Galaxy S2

Chaque puce nécessite une séquence d'allumage et d'envoi d'un microcode (firmware), qui n'est pas traitée par ce sujet et qui est déjà implémentée par des utilitaires du projet Replicant.

Installation de CyanogenMod sur l'appareil :

Il est nécessaire d'installer CyanogenMod 9.1.0 sur l'appareil afin de disposer d'un environnement de développement et d'accès au compte root.
Les instructions d'installation sont disponibles sur le wiki du projet CyanogenMod :
<http://wiki.cyanogenmod.org/w/Devices>

Mise en place de l'environnement de développement :

Afin d'utiliser les outils d'analyse de l'implémentation propriétaire, il est nécessaire de mettre en place l'environnement de développement de Replicant.
Tout d'abord, installer les dépendances nécessaires (il est nécessaire d'utiliser un système GNU/Linux et il est recommandé d'utiliser Debian, version jessie) :
<http://redmine.replicant.us/projects/replicant/wiki/BuildDependencies>
Puis, télécharger le code source du système (entre 10 et 20 Gio) :
<http://redmine.replicant.us/projects/replicant/wiki/GettingReplicantSources>

Tracer l'implémentation propriétaire :

Il est nécessaire d'utiliser l'outil strace, qui va permettre d'afficher les appels système effectués par le logiciel. La page de manuel de strace explicite les options à utiliser. Il est recommandé de sauvegarder la sortie de strace dans un fichier (l'affichage sur le terminal pouvant ralentir l'exécution et poser des problèmes de timeout).

Compiler le noyau Linux :

Afin d'obtenir le plus de verbosité lors du débogage, il est préférable de recompiler un noyau avec la variante engineering (-eng) d'Android. Ceci peut par ailleurs permettre de tracer le protocole depuis le noyau. Les instructions sont disponibles pour différents appareils :
<http://redmine.replicant.us/projects/replicant/wiki#Building-Replicant>

Généralement, la règle de compilation pour le noyau est **bootimage**.
Les instructions d'installation du noyau sont précisées pour chaque appareil :
<http://redmine.replicant.us/projects/replicant/wiki#Installing-Replicant>
(Reflasher la partition **boot** avec **fastboot** pour les Nexus et **KERNEL** avec **heimdall** pour les Galaxy.)

Notes spécifiques au BCM4751 :

Une page ressource concernant le BCM4751 existe sur le wiki du projet Replicant :
<http://redmine.replicant.us/projects/replicant/wiki/BCM4751>

Notes spécifiques au GSD4t:

Une page ressource concernant le GSD4t existe sur le wiki du projet Replicant :
<http://redmine.replicant.us/projects/replicant/wiki/GSD4t>

Informations et aide technique :

Si des problèmes techniques sont rencontrés lors de la mise en place de l'environnement de développement, on pourra utiliser les forums, le canal IRC ou la liste de diffusion du projet Replicant :

- Forums : <http://redmine.replicant.us/projects/replicant/boards>
- IRC : #replicant sur irc.freenode.net
- Liste de diffusion : <http://lists.osuosl.org/mailman/listinfo/replicant>