

雲端儲存再進化

免 root 之 Android 應用程式資料備份、分享 (AppSync)

指導教授: 吳育松

組員: 王傑民、孫羽柔、王永慶

專題研究動機與目的





隨著智慧裝置的普及，每個手上的智慧裝置也越來越多，當手上的智慧裝置越來越多時，我們會希望把所有的 app data (app data 為 app 在 /data/data 下所存放的屬於該 app 的系統資料，內容包括了該 app 的設定、有關該 app 的所有的資料。ex: 若該 app 是一款手機遊戲，在該目錄下所存放的 app data 可能就包括目前的遊戲紀錄。若該 app 是一款通訊軟體，該 app data 就可能存放著通訊歷史紀錄等。) 都能夠保存下來，並且讓不同裝置間的 app 資料能夠透過雲端或者直接在本地端進行同步，能對使用者帶來極大的便利性。

而目前 Android 上，對於 app data 的備份的解決方法也有一些工具可以使用，但是目前的解決辦法都需要使用者對他們 Android device 進行 root 的動作，對於一般的使用者來說，這是

一個非常不友善而且危險的方法，幾乎所有的手機廠都不支持使用者進行 root 的動作，一旦有 root 的行為，該裝置就會失去保固，這是一般使用者所不樂見的，所以我們希望能夠提供一個不需要 root 的方法進行備份、同步、分享的動作。

現有相關研究概況及比較

目前市面上有提供 app data 備份的軟體如圖表所示，唯一也能夠做到不需要 root 就能夠進行 app data 的備份軟體為 Carbon Backup，他們所採用的方式為透過 android 4.0 以上所原生支援的 adb backup 和 adb restore 功能，但是此方法在每次進行備份、還原都過程中都必須要使用者先將裝置與電腦連結，並且安裝他們所提供的 PC Client 才能夠進行工作，相當的不方便，而我們所設計的 AppSync 軟體能夠直接在手機端進行操作，更為方便！

	Titanium Backup	My Backup Pro	App Backup & Restore	Carbon Backup	AppSync
Cloud Storage	Dropbox、GDrive、Box	No	No	GDrive	AppSync cloud storage
Backup apk	Yes	Yes	Yes	Yes	Yes
Backup app data	Yes	Require root	Require root	Yes	Yes
Sync with other devices	No	No	No	Require root	Yes
Supports non-rooted devices	No	Apk only	Apk only	connect to PC require	Yes
					

設計原理、研究方法與步驟

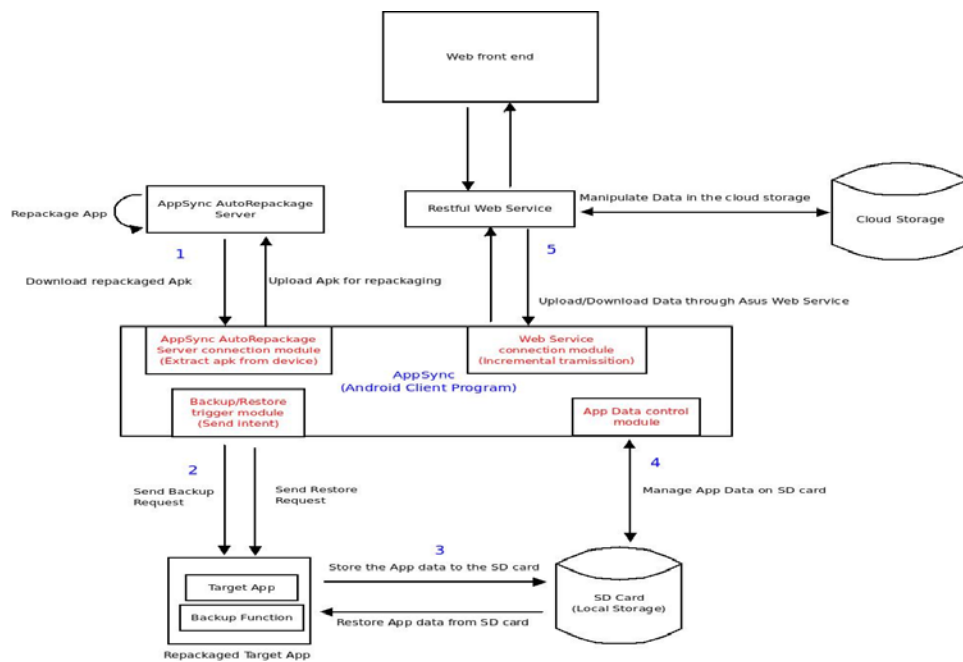
AppSync 的程式架構流程如圖所示，首先在步驟 1，我們會先讓使用者選取它想要賦予能夠取出 app data 能力的 app，選取後 AppSync 就

會自動將該 app 的 apk 檔取出，傳上我們設計的 AppSync AutoRepackage Server，它會對 apk 進行 repackage 的動作，我們在突破讓 non-rooted Android device 可以取出 app data 的部份是採取

repackage 的技術，基本概念就是將原本的 app 嵌入一段可以執行將 app data 取出放到 SD 卡上的程式碼，和另外一段可以將 app data 從 SD 卡上還原到系統資料夾的程式碼，這段程式碼會被我們的 AppSync 所觸發執行這段功能，如此一來在 non-rooted Android device 上就能夠作到取出和還原 App Data 的功能了。接下來經過 repackage 的 apk 會透過 AppSync 快速的安裝到使用者的裝置上，並且覆蓋掉原本的 app (透過 app 更新的方式，讓使用者可以保留原本的 app

data)，如此一來我們就賦予了該 app 具有取出 app data 的能力了。

接下來在步驟 2 中，我們會根據使用者的操作，對經過我們 repackage 過後的特定 app 送入相對 intent，這時候我們嵌入該 app 的相對應程式碼就會被觸發開始作用，在步驟 3 中，將 app data 放到 local storage 上，如此一來，使用者就可以在沒有 root 的情況下取出 app data 了！至於步驟 4，則是 AppSync 會去管理放在 local storage 上面的資料。



AppSync 設計概觀

為什麼要 Repackage? Repackage 架構

在原本的 Android 架構裡，app data 存放在每個 app 專屬的 /data/data/<package>/ 目錄裡，但是該目錄只有自己的 app 能夠存取，目前市面上的 app data 備份軟體都是透過先將 Android device root 過後 (root 指的是取得最高使用者權限的過程)，才能夠去存取該目錄，但是 root 這個動作的缺點如前面所提，因此代替 root 我們用採取 repackage 的方法來達到取出 app data 的目標。

在 repackage 的部份，我們會先對我們想要 repackage 的目標 app 進行反組譯的動作，

翻譯成 dalvik assembly，另外我們再將我們寫好的取出/還原 app data 的模組也反組譯成 dalvik Assembly，再將兩個部份再 assembly 階段結合在一起，再組譯成最終的 repackaged apk。

結論

我們所設計的 AppSync 在技術實作上已經相當完整並且可以穩定運作，也已經在 Google Play Store 上架，希望未來能造福廣大的使用者。