Firma digitale delle app Android

Bryan Corradino – A.A. 21/22

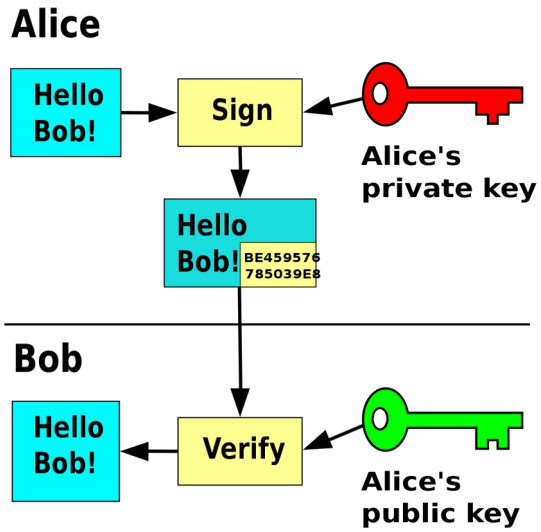
Presentazione per Sistemi Mobile

Firme digitali: rudimenti

- **Ogni soggetto possiede una coppa di chiavi**
  - Una *pubblica*, da condividere con i destinatari dei propri messaggi
  - Una *privata*, da mantenere segreta
- **Le chiavi vengono generate da un algoritmo crittografico**
  - L'algoritmo deve garantire che un contenuto crittografato con la chiave privata possa essere decriptato solo con la corrispondente chiave pubblica (e viceversa) *(crittografia asimmetrica)*
- **Si vogliono garantire *autenticità*, *integrità* e *non ripudiabilità***

Firme digitali: rudimenti

- **Alice vuole inviare un messaggio a Bob**
  - Calcola un hash del messaggio (detto *digest*), lo cripta con la propria chiave privata e lo allega al messaggio originale
- **Bob riceve il messaggio**
  - Decripta il digest usando la chiave pubblica di Alice, calcola un nuovo digest del messaggio e lo confronta con quello allegato da Alice
  - Se coincidono, Bob ha la certezza che il messaggio non sia stato manomesso da terzi
  - Tuttavia, nulla garantisce che il mittente sia davvero Alice...



Certificate Authority (CA)

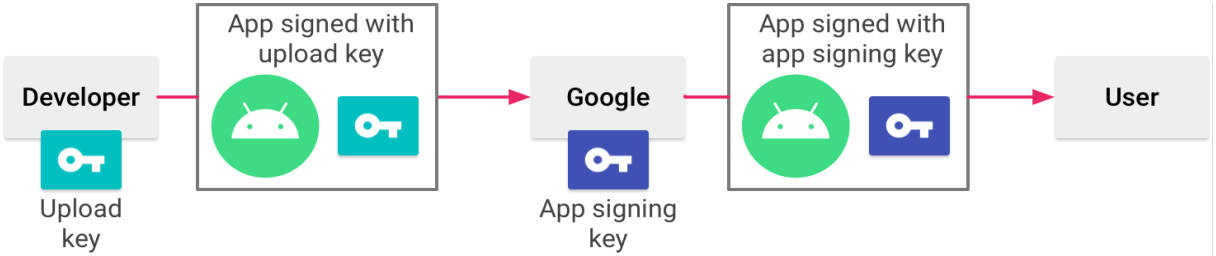
- **Sono soggetti terzi di fiducia che fanno da garanti della corrispondenza tra chiave e proprietario**
- **In Android non è necessario farsi rilasciare un certificato digitale da una CA**
  - Sono ammesse app con certificati *self-signed*

Perché firmare digitalmente un'app?

- **Motivazioni**
  - Si vuole rendere possibile identificare il soggetto che ha compilato e distribuito un'app
  - Eventuali manomissioni si rilevano con facilità
  - App che condividono lo stesso certificato possono comunicare/condividere file tra di loro più facilmente
- **Punto cruciale: Android non permette l'installazione di app non firmate**

Modalità di firma delle app

- **Play App Signing**
  - Servizio offerto da Google
  - Fa uso di *upload key* (custodita dallo sviluppatore) e *signing key* (custodita da Google ed eventualmente anche dallo sviluppatore)
  - Requisito per generare i cosiddetti App Bundle (obbligatori su Google Play da agosto 2021)



- **Firma manuale**
  - Lo sviluppatore genera le proprie chiavi ed è sua responsabilità proteggere la chiave privata
- **Firma di debug**
  - Eseguita da Android Studio quando si compila l'app, non viene accettata dagli Store

Play App Signing: ne ho bisogno?

- **È praticamente obbligatorio se si vogliono distribuire app sul Play Store**
- **Revoca della upload key**
  - Se lo sviluppatore smarrisce la propria upload key o se questa viene compromessa, può richiederne la revoca a Google
- **È impossibile perdere la signing key**
  - Smarrire la signing key utilizzata per firmare la propria app equivale a perdere qualsiasi autorità su quell'app
  - Dal momento che Google custodisce una copia della signing key, è teoricamente impossibile smarrirla
- **Tuttavia…**

Procedimento

- **In Android Studio**
  - *Build -> Generate Signed Bundle/APK*
  - Selezionare *Android App Bundle* o *APK*
  - *Key store path -> Create new*
  - Fornire i dati per la creazione del *keystore* e della *upload key*
- ***Build -> Generate Signed Bundle/APK***
  - Selezionare *Android App Bundle* o *APK*
  - Specificare percorso e dati del keystore appena creato
  - *Next*, poi specificare il percorso di output dell'app firmata e concludere
- **Accedere alla propria *Play Console* (https://play.google.com/console/)**
  - Creare una nuova *release* (passaggi non mostrati)
  - Scegliere se fornire una propria signing key o delegarne la generazione a Google