Introduzione

Android ha impiegato un tempo relativamente breve per divenire la piattaforma mobile più diffusa nel mondo. Sebbene in origine sia stato progettato per gli smartphone, oggi è utilizzato su tablet, televisori e dispositivi indossabili; a breve sarà disponibile anche sulle automobili.

Android viene sviluppato a ritmi incessanti: in media assistiamo a due release principali l'anno, ognuna delle quali introduce una nuova interfaccia utente, miglioramenti a livello di prestazioni e una serie di nuove funzionalità utente ampiamente discusse nei blog ed esaminate fin nei più piccoli dettagli dagli appassionati del sistema.

Uno degli aspetti della piattaforma Android che ha subìto notevoli miglioramenti negli ultimi anni, pur non suscitando grande attenzione tra il pubblico, è la sicurezza. Negli anni Android è divenuto sempre più resistente alle tecniche comuni di exploit (come l'overflow del buffer), il suo isolamento delle applicazioni (sandboxing) è stato rafforzato e la sua superficie attaccabile si è notevolmente ridotta grazie a una diminuzione importante del numero di processi di sistema eseguiti come root. Inoltre, nelle ultime versioni sono state introdotte importanti funzionalità di protezione quali i profili utente con una migliore e differenziata gestione dei permessi, la crittografia dell'intero disco, l'archiviazione delle credenziali con supporto hardware e il supporto per il provisioning, la gestione dei dispositivi centralizzati. Con Android 5 sono stati annunciati altri miglioramenti alla sicurezza e nuove funzionalità *enterprise*, come il supporto a una gestione separata dei profili utente e delle relative app associate, il potenziamento della crittografia dell'intero disco e il supporto per l'autenticazione biometrica.

Parlare dei miglioramenti all'avanguardia nel campo della sicurezza è sicuramente interessante, ma è molto più importante capire l'architettura di protezione di Android esaminandola dal basso verso l'alto, visto che ogni nuova funzionalità di sicurezza viene costruita partendo dal modello di protezione della piattaforma e integrata al suo interno. Il modello di sandboxing di Android (in cui ogni applicazione viene eseguita come un utente Linux distinto e ha una directory dati dedicata) e il suo sistema di autorizzazioni (che richiede a ogni applicazione di dichiarare esplicitamente le funzionalità della piattaforma che intende utilizzare) sono ben documentati e chiari. Tuttavia, i componenti interni di altre funzionalità fondamentali della piattaforma che incidono sulla sicurezza dei dispositivi, come la gestione dei package e la firma del codice, sono generalmente considerati come una "scatola chiusa" dalla community che si occupa di ricerca nell'ambito della sicurezza.

Tra i motivi per cui Android è così popolare vi è sicuramente la facilità relativa con cui un dispositivo può essere "flashato" con una build personalizzata di Android, sottoposto a "root" con l'applicazione di un package di aggiornamento di terze parti o comunque personalizzato con altre tecniche. I forum e i blog degli appassionati di Android sono pieni di numerose guide pratiche che accompagnano gli utenti nelle procedure necessarie per sbloccare un device e applicare vari package personalizzati; tuttavia, queste fonti offrono informazioni molto poco strutturate sul funzionamento "dietro le quinte" degli aggiornamenti di sistema e sui relativi rischi.

Questo libro mira a colmare queste lacune con una descrizione dell'architettura di protezione di Android dal basso verso l'alto, addentrandosi nell'implementazione dei principali sottosistemi Android e nei componenti legati alla sicurezza dei dati e dei dispositivi. Tra gli argomenti affrontati vi sono questioni che interessano tutte le applicazioni, come la gestione di utenti e package, i permessi e le policy del dispositivo, ma anche spiegazioni più specifiche come quelle sui provider di crittografia, l'archiviazione delle credenziali e il supporto di elementi sicuri.

Non è raro che tra una release e l'altra interi sottosistemi Android vengano sostituiti o riscritti; tuttavia, lo sviluppo relativo alla sicurezza è conservativo per natura. Quindi, anche se il comportamento descritto potrebbe cambiare o essere integrato da una release all'altra, l'architettura di sicurezza fondamentale di Android dovrebbe rimanere piuttosto stabile anche nelle versioni future.

Destinatari del libro

Questo libro è utile per chiunque sia interessato a saperne di più sull'architettura di sicurezza di Android. Le descrizioni di alto livello di ogni funzionalità di sicurezza e i dettagli di implementazione forniti sono un punto di partenza utile sia per i ricercatori nel campo della sicurezza, interessati a valutare il livello di sicurezza di Android nel suo complesso o di un sottosistema specifico, sia per gli sviluppatori della piattaforma, che si occupano della personalizzazione e dell'estensione di Android, entrambi interessati a comprendere il codice sorgente della piattaforma sottostante. Gli sviluppatori di applicazioni possono comprendere meglio il funzionamento della piattaforma per scrivere applicazioni più sicure e sfruttare meglio le API di sicurezza che questa fornisce. Anche se parte del libro è fruibile da un pubblico senza elevate competenze tecniche, la maggior parte della trattazione è strettamente legata al codice sorgente o ai file di sistema di Android, pertanto è utile una conoscenza dei concetti base relativi allo sviluppo del software in ambiente Unix.

Prerequisiti

Il libro presume che i lettori conoscano almeno i fondamenti dei sistemi operativi in stile Unix, preferibilmente Linux; i concetti comuni quali processi, gruppi di utenti, permessi dei file e così via non verranno quindi spiegati. Le funzionalità del sistema operativo specifiche di Linux o aggiunte di recente sono generalmente introdotte brevemente prima di parlare dei sottosistemi di Android che le utilizzano. La maggior parte del codice della piattaforma presentato proviene dai daemon del core (solitamente implementati in C o C++) e dai servizi di sistema (generalmente implementati in Java) di Android, pertanto è richiesta una certa dimestichezza con almeno uno di questi linguaggi. Alcuni esempi di codice contengono sequenze di chiamate del sistema Linux, quindi può essere utile conoscere la programmazione in ambiente Linux per capire il codice (sebbene non sia assolutamente obbligatorio). Infine, anche se la struttura di base e i componenti fondamentali (quali activity e servizi) delle app di Android sono brevemente descritti nei capitoli iniziali, si presume che il lettore conosca almeno le basi dello sviluppo di questo sistema operativo.

Versioni di Android

La descrizione dell'architettura e dell'implementazione di Android proposta in questo libro (fatta eccezione per numerose funzionalità di proprietà di Google) si basa su codice sorgente rilasciato al pubblico come parte dell'*Android Open Source Project* (AOSP). La maggior parte della trattazione e dei frammenti di codice si riferisce ad Android 4.4. A volte si fa riferimento anche al *ramo master* di AOSP, in quanto l'adesione a esso è in genere una valida indicazione della direzione che prenderanno le release future di Android. Va però sottolineato che non tutte le modifiche al ramo master vengono integrate senza variazioni nelle release al pubblico, pertanto è possibile che nelle versioni future alcune delle funzionalità presentate vengano modificate o persino rimosse.

Una versione di anteprima per sviluppatori di Android 5 è stata annunciata mentre questo libro veniva completato. Al momento di questa stesura il codice sorgente di Android 5 non era disponibile e la data esatta di rilascio al pubblico era sconosciuta. Per quanto la release di anteprima includesse nuove funzionalità di sicurezza, quali miglioramenti alla crittografia del dispositivo, alla gestione dei profili e dei dispositivi, nessuna di queste caratteristiche era definitiva. Ecco perché nel libro queste nuove funzionalità non vengono affrontate. Per quanto sia possibile introdurre alcuni dei miglioramenti alla protezione di Android 5 basandosi sul comportamento osservato, senza il codice sorgente sottostante una qualunque descrizione della relativa implementazione sarebbe stata incompleta e teorica.

Organizzazione del libro

Il libro prevede 13 capitoli da leggere in sequenza. In ogni capitolo viene affrontato un aspetto o una caratteristica della sicurezza di Android, e i capitoli successivi si basano sui concetti introdotti in quelli precedenti. Anche se conoscete già l'architettura e il modello di sicurezza di Android e siete interessati soltanto ai dettagli relativi a un argomento specifico, è consigliabile che leggiate comunque i Capitoli da 1 a 3, in quanto i concetti che contengono formano le basi per tutto il resto del libro.

- Il Capitolo 1, "Modello di sicurezza di Android", offre un'introduzione di alto livello all'architettura e al modello di sicurezza di Android.
- Nel Capitolo 2, "Permessi", vengono spiegate la dichiarazione, l'uso e l'applicazione dei permessi Android.
- Il Capitolo 3, "Gestione dei package", affronta la firma del codice e i dettagli relativi al funzionamento dell'installazione e della gestione di applicazioni Android.

- Nel Capitolo 4, "Gestione degli utenti", viene esaminato il supporto multiutente di Android e viene descritta l'implementazione dell'isolamento dei dati nei dispositivi multiutente.
- Il Capitolo 5, "Provider di crittografia", offre informazioni generali sul framework *Java Cryptography Architecture* (JCA) e descrive i provider di crittografia JCA di Android.
- Il Capitolo 6, "Sicurezza di rete e PKI", introduce l'architettura del framework *Java Secure Socket Extension* (JSSE) e ne esamina l'implementazione in Android.
- Il Capitolo 7, "Archiviazione delle credenziali", descrive lo store delle credenziali di Android e introduce le API fornite alle applicazioni che necessitano di memorizzare in sicurezza le chiavi di crittografia.
- Nel Capitolo 8, "Gestione degli account online", si parla del framework di gestione degli account online di Android e dell'integrazione in Android del supporto per gli account Google.
- Il Capitolo 9, "Sicurezza aziendale", si occupa invece del framework di gestione dei dispositivi di Android, descrivendo nei dettagli l'implementazione del supporto VPN e addentrandosi nel supporto per l'*Extensible Authentication Protocol* (EAP).
- Il Capitolo 10, "Sicurezza del dispositivo", introduce il boot verificato, la crittografia del disco e l'implementazione della schermata di blocco di Android, mostrando anche l'implementazione del debug USB sicuro e del backup del dispositivo crittografato.
- Nel Capitolo 11, "NFC ed elementi sicuri", vengono presentati lo stack NFC di Android, l'integrazione di elementi sicuri (SE, *Secure Element*), le API e l'emulazione di schede basata su host (HCE, *Host-based Card Emulation*).
- Il Capitolo 12, "SELinux", inizia con una descrizione dell'architettura e del linguaggio per le policy di SELinux, spiega nei dettagli i cambiamenti apportati a SELinux per l'integrazione in Android e presenta la policy SELinux base di Android.
- Nel Capitolo 13, "Aggiornamenti di sistema e accesso root", si parla dell'uso del bootloader e del sistema operativo di recovery di Android per l'esecuzione di aggiornamenti sull'intero sistema e delle modalità con cui ottenere l'accesso root sulle build Android di engineering e produzione.

Convenzioni

Questo libro si occupa principalmente dell'architettura e dell'implementazione di Android, pertanto contiene numerosi frammenti di codice e listati, a cui si fa riferimento in maniera costante nei paragrafi che li seguono. Per distinguere questi riferimenti (che di solito includono più costrutti del sistema operativo o del linguaggio di programmazione) dal resto del testo sono state adottate alcune convenzioni di formattazione.

Comandi, nomi di funzioni e variabili, attributi XML, nomi di oggetti SQL, parole chiave, nomi di file, utenti, gruppi di Linux, URL, processi e altri oggetti del sistema operativo sono resi in `monospaziato` (per esempio, "il file `packages.xml` ", "l'utente `system`", "il daemon `vold`" e così via). Anche i valori letterali stringa sono in questo formato; quando li utilizzate in un programma, dovete generalmente racchiuderli tra virgolette singole o doppie (per esempio, `Signature.getInstance("SHA1withRSA", "AndroidOpenSSL")`).

I nomi delle classi Java sono generalmente nel formato non qualificato, senza il nome del package (per esempio, "la classe `Binder`"); i nomi completi sono utilizzati solamente quando nell'API o nel package in discussione sono presenti più classi con lo stesso nome, o quando per altre ragioni è importante specificare il package contenitore (per esempio, "la classe `javax.net.ssl.SSLSocketFactory`"). Quando referenziati nel testo, i nomi di metodi e funzioni sono mostrati con le parentesi, ma i loro parametri sono omessi per ragioni di spazio (per esempio, "il metodo factory `getInstance()`"). Consultate la documentazione di riferimento pertinente per la firma completa della funzione o del metodo.

I percorsi, le cartelle/directory, le partizioni, i nomi degli elementi dell'interfaccia, come menu, opzioni, finestre di dialogo, le chiavi, i permessi, i livelli di protezione, le librerie e i moduli sono invece riportati in *corsivo*.

La maggior parte dei capitoli contiene diagrammi che illustrano l'architettura o la struttura del sottosistema o del componente di protezione in discussione. Tutti i diagrammi seguono lo stile informale fatto di caselle e frecce, ma non si conformano rigorosamente a un formato specifico. Detto questo, molti prendono in prestito idee dai diagrammi di distribuzione e dalla classe UML; le caselle generalmente rappresentano classi e oggetti, mentre le frecce indicano le dipendenze o i percorsi di comunicazione.

Ringraziamenti

Desidero ringraziare tutto il personale di No Starch Press che ha collaborato a questo libro. Un ringraziamento speciale va a Bill Pollock, che ha reso intelligibili i miei sproloqui, e ad Alison Law, per la sua pazienza durante la trasformazione degli stessi in un libro vero e proprio.

Un grande ringraziamento è riservato a Kenny Root, che si è occupato della revisione dei capitoli e ha condiviso alcuni aneddoti relativi alle funzionalità di sicurezza di Android.

Ringrazio Jorrit "Chainfire" Jongma per la manutenzione di SuperSU, uno strumento impareggiabile per frugare tra i componenti interni di Android, e per aver rivisto la mia descrizione dello stesso nel Capitolo 13.

Grazie a Jon "jcase" Sawyer per aver continuato a mettere in dubbio le nostre supposizioni sulla sicurezza di Android e per aver contribuito al mio libro scrivendone la prefazione.

L'autore

Nikolay Elenkov ha lavorato a progetti di sicurezza per grandi aziende negli ultimi dieci anni. Ha sviluppato software per la sicurezza su varie piattaforme, da smart card e HSM a server Windows e Linux. Si è interessato ad Android subito dopo il rilascio iniziale al pubblico e sviluppa applicazioni per questo sistema operativo dalla versione 1.5. L'interesse di Nikolay per i componenti interni di Android si è intensificato con il rilascio di Android 4.0 (Ice Cream Sandwich). Negli ultimi tre anni ha documentato le sue scoperte nel suo blog dedicato alla sicurezza Android, consultabile all'indirizzo `http://nelenkov.blogspot.com/`.

Il revisore tecnico

Kenny Root è un collaboratore fondamentale di Google nell'ambito della piattaforma Android dal 2009 e rivolge la sua attenzione prevalentemente alla sicurezza e alla crittografia. È l'autore di ConnectBot, la prima app SSH per Android; contribuisce inoltre allo sviluppo del mondo open source. Quando non si sta occupando di hacking, trascorre il tempo con la moglie e i due figli. Ha studiato alla Stanford University, alla Columbia University, alla Chinese University di Hong Kong e al Baker College, ma proviene da Kansas City, la città delle grigliate migliori del mondo.