

Sigurnost operacijskih sustava i aplikacija

Android

Tin Bušić, 04.04.2025.

Pregled predavanja

- Pitanja za ispite
- Motivacija
- Uvod u sigurnost Androida
- Detaljna analiza arhitekture sigurnosti Androida
- Pohrana s ograničenim pristupom: evolucija i implementacija
- Uobičajene prijetnje Android aplikacijama
- Sigurnost međukomponentne komunikacije
- Izazovi i ograničenja u sigurnosti Androida
- Budući smjerovi razvoja
- Zaključak

Pitanja za ispite

- Koji su osnovni principi sigurnosnog modela Androida?
- Kako višestrana suglasnost poboljšava sigurnost platforme?
- Što je pohrana s ograničenim pristupom i zašto je uvedena?
- Kako Android postiže sigurnost međukomponentne komunikacije?
- Koji su glavni sigurnosni izazovi Android platforme?
- Koje su najčešće prijetnje Android aplikacijama?

Motivacija

- Android – najrašireniji mobilni OS (>70% tržišta)
- Ogromna količina osjetljivih korisničkih podataka
- Kompleksan sigurnosni model → programer ključna karika
- Sigurnosni propusti → ozbiljne posljedice (gubitak povjerenja, uklanjanje s Google Play-a)
- Potrebna integracija sigurnosti u svaki korak razvoja
- Korištenje alata za statičku i dinamičku analizu (Lint, SonarQube...)

Uvod u sigurnost Androida (1/2)

- Definiranje sigurnosti Androida
 - Proces koji traje kroz životni ciklus aplikacije, od dizajna do održavanja
 - Zaštita od zlonamjernog koda, neovlaštenog pristupa i krađe podataka
- Temeljni ciljevi sigurnosti
 - Osiguravanje povjerljivosti (*engl. confidentiality*), integriteta (*engl. integrity*) i dostupnosti (*engl. availability*) podataka
 - Zaštita korisničkih podataka i privatnosti

Uvod u sigurnost Androida (2/2)

- Programer odgovoran za implementaciju sigurnosnih mehanizama
 - Važnost praćenja sigurnosnih smjernica i redovitih ažuriranja
- Uloga otvorenosti Androida
 - Prednosti i nedostaci otvorenosti u kontekstu sigurnosti
 - Omogućuje transparentnost i prilagodbu

Ključni principi sigurnosnog modela Androida

- Model višestranе suglasnosti (*engl. multi-party consent*)
 - Radnje zahtijevaju pristanak korisnika, programera i platforme
 - Korisnik: odobravanje dozvola
 - Programer: deklariranje dozvola i poštivanje pravila
 - Platforma: provedba sigurnosnih mehanizama

Ključni principi sigurnosnog modela Androida

- Slojevita sigurnost (*engl. defense in depth*)
 - Više slojeva zaštite
 - Jezgra operacijskog sustava: izolacija aplikacija
 - Posrednički sloj: upravljanje dozvolama
 - Aplikacijski okvir: sigurnosne politike
- Princip najmanjih privilegija
 - Aplikacijama se dodjeljuju minimalne potrebne dozvole
 - Pri implementaciji tražiti samo neophodne dozvole
 - Korištenje AndroidX i Jetpack sigurnosnih biblioteka
 - Smanjuje rizik od zlouporabe resursa

Arhitektura Androida – Linux Kernel

- Temelj sigurnosnog modela
- Izolacija procesa i upravljanje resursima
 - Svaka aplikacija ima svoj UID (*user identifier*)
- Upravljanje memorijom i resursima
- SELinux (Security-Enhanced Linux) - kontrola pristupa resursima
- Ograničenja sigurnosti jezgre operacijskog sustava
 - Ovisnost o pravilnoj konfiguraciji i ažuriranjima
 - Ne može zaštititi od svih vrsta napada

Arhitektura Androida – posrednički sloj

- Uloga posredničkog sloja (*engl. middleware*) u sigurnosti
 - Posrednik između aplikacija i jezgre operacijskog sustava
 - Upravljanje dozvolama i API-jima
 - Provodi dozvole *sandboxing*
- Android Runtime (ART)
 - Provjera bajtkoda prije izvršavanja
 - Sprječavanje izvršavanja zlonamjernog koda
- Pri implementaciji koristiti moderne SDK (*software development kit*) alate i sigurni IPC (*interprocess communication*)

Arhitektura Androida – aplikacijski sloj

- Sigurnosne politike na razini aplikacija
 - Šifriranje podataka (npr. pohrana lozinki)
 - Korištenje *EncryptedSharedPreferences* i *Jetpack Security*
 - Zaštita od neovlaštenog pristupa API-jima
- Upravljanje dozvolama
 - Normalne, opasne i posebne dozvole
- Sigurna međukomponentna komunikacija (ICC)
 - *Intents*: sigurna razmjena poruka između aplikacija
 - Ograničavanje tko može slati i primati određene *Intents*
- Kontinuirano testiranje prema *OWASP Mobile Top 10*

Pohrana s ograničenim pristupom– Motivacija (1/2)

- Problem: neograničen pristup datotečnom sustavu
 - Prije pohrane s ograničenim pristupom (*engl. scoped storage*), aplikacije su imale preširok pristup datotekama korisnika
 - Svaka aplikacija je mogla čitati i pisati u vanjski spremnik (*external storage*)
- Rizici povezani s neograničenim pristupom
 - Curenje osjetljivih podataka: aplikacije su mogle krasti slike, dokumente i druge datoteke
 - Zlonamjerne aplikacije: iskorištavale su pristup za širenje zloćudnog koda ili praćenje korisnika

Pohrana s ograničenim pristupom– Motivacija (2/2)

- Zahtjevi za privatnošću i sigurnošću
 - Korisnici su trebali bolju kontrolu nad svojim podacima
 - Programeri su trebali sigurniji način za pohranu i pristup datotekama
- Cilj pohrane s ograničenim pristupom
 - Ograničiti pristup aplikacija samo na vlastite datoteke
 - Osigurati da aplikacije trebaju dozvolu za pristup zajedničkim datotekama

Pohrana s ograničenim pristupom – karakteristike (1/2)

- Ograničen pristup datotečnom sustavu
 - Aplikacije imaju pristup samo vlastitim direktorijima (*engl. app-specific directories*)
 - Ovi direktoriji su privatni i zaštićeni od drugih aplikacija
- *MediaStore API* za pristup zajedničkim datotekama
 - Aplikacije moraju koristiti *MediaStore API* za pristup slikama, videozapisima i audio datotekama
 - *MediaStore API* pruža kontrolirani pristup zajedničkim datotekama

Pohrana s ograničenim pristupom – karakteristike (1/2)

- Zahtjevi za specifičnim dozvolama
 - Aplikacije moraju tražiti dozvolu za pristup određenim vrstama datoteka (npr. slike, videozapisi)
 - Korisnici mogu odobriti ili odbiti dozvole
- Ograničenje pristupa "*legacy*" pohrani (/sdcard)
 - Direktan pristup "*legacy*" pohrani je ograničen
 - Aplikacije moraju koristiti *API* pohrane s ograničenim pristupom za pristup datotekama u toj pohrani

Pohrana s ograničenim pristupom

- Prednosti pohrane s ograničenim pristupom
 - Smanjenje rizika od curenja podataka: aplikacije imaju pristup samo onim datotekama koje su im potrebne
 - Povećanje privatnosti korisnika: korisnici imaju veću kontrolu nad svojim datotekama
 - Bolja organizacija datotečnog sustava: aplikacije ne mogu stvarati datoteke bilo gdje
- Nedostaci pohrane s ograničenim pristupom
 - Kompromisi u funkcionalnosti aplikacija: neke aplikacije mogu trebati više pristupa datotekama nego što je sada dopušteno
 - Kompleksnost za programere: programeri moraju naučiti novi API za pristup datotekama, više rubnih slučajeva za pokriti
 - Mogući problemi kompatibilnosti sa starijim aplikacijama

Sigurnost međukomponentne komunikacije (ICC)

- Međukomponentna komunikacija (*engl. inter-component communication, ICC*) između različitih komponenti unutar iste ili različitih aplikacija
 - Osigurava funkcionalnost i interoperabilnost Android sustava
 - Komunikacija između aktivnosti, servisa, *broadcastova*
- Kako Android osigurava međukomponentnu komunikaciju
 - *Intents*: sigurna razmjena poruka između aplikacija
 - Dozvole: kontrola pristupa komponentama i podacima.
 - *Sandboxing*: izolacija aplikacija kako bi se spriječio neovlašten pristup

Sigurnost međukomponentne komunikacije (ICC)

- Uloga *Intents* u sigurnosti
 - Eksplicitni: ciljaju određenu komponentu aplikacije
 - Implicitni: deklariraju radnju koju treba izvršiti, ali ne ciljaju određenu komponentu
- Izbjegavati slati osjetljive podatke kroz implicitne *Intents*
 - Ograničavanje intentu unutar paketa
 - Ručno validiranje *Intents* izvana
- Potencijalni problemi s ICC sigurnošću
 - Ranjivosti zbog neispravnog rukovanja *Intents*
 - Curenje podataka kroz nezaštićene *Intents*
 - Lažiranje *Intents* (*Intent spoofing*)

Uobičajene prijetnje Android aplikacijama

- Zlonamjerne aplikacije
 - Načini distribucije: trgovine aplikacijama, phishing, sideloading
 - Primjeri: virusi, trojanci, špijunski programi, ucjenjivački kod
 - Kako se zaštititi: provjera dozvola, instaliranje aplikacija iz pouzdanih izvora, korištenje antivirusnog softvera
- Phishing i društveni inženjering
 - Kako funkcionira: lažne poruke, web stranice, e-mailovi
 - Primjeri: lažni zahtjevi za lozinkom, nagradne igre, hitne situacije
 - Kako se zaštititi: provjera autentičnosti, oprez pri klikanju na linkove, ignoriranje sumnjivih poruka

Prijetnje

- Nedovoljno šifriranje
 - Ako pohranjujemo korisničke podatke lokalno (npr. token), koristi Android Keystore
 - *SharedPreferences* ≠ sigurno pohranjivanje – koristiti *EncryptedSharedPreferences*
 - Izbjegavati hardkodirane lozinke i API ključeve u kodu – koristi gradivne varijable (*BuildConfig*)
- Loša autorizacija
 - Provjeriti prava pristupa na serveru, ne samo na klijentu
 - Nikad ne oslanjati se isključivo na UI za sakrivanje funkcionalnosti
 - Koristiti sigurne sesije i provjeri identitet korisnika pri svakom zahtjevu

Izazovi – statičke dozvole

- Dozvole koje se aplikacijama dodjeljuju prilikom instalacije
- Korisnik odobrava dozvole prije instalacije
- Nedostaci statičkih dozvola
 - Aplikacije mogu tražiti previše dozvola
 - Korisnici često ne razumiju čemu služe dozvole
 - Nema mogućnosti dinamičkog mijenjanja dozvola tijekom korištenja aplikacije
- Posljedice statičkih dozvola
 - Zlonamjerne aplikacije mogu zloupotrijebiti dozvole
 - Korisnici mogu biti izloženi riziku od praćenja i krađe podataka

Izazovi – Kontrola protoka informacija

- Praćenje i kontroliranje načina na koji se podaci kreću kroz sustav
- Osiguravanje da osjetljivi podaci ne cure na neovlaštena mjesta
- Nedostatak holističke sigurnosti
 - Android nema ugrađene mehanizme za potpunu kontrolu protoka informacija
 - Aplikacije mogu dijeliti podatke na nesiguran način
- Posljedice nedostatka kontrole protoka informacija
 - Curenje osjetljivih podataka
 - Ranjivost na napade koji iskorištavaju protok informacija

Budući smjerovi razvoja

- Dinamičniji modeli dozvola
 - Korisnik može odobriti ili odbiti dozvole u bilo kojem trenutku
 - Prednosti: veća kontrola korisnika, bolja zaštita privatnosti
- Bolji alati za praćenje sigurnosti
 - Potreba za alatima koji mogu otkriti zlonamjerno ponašanje aplikacija
 - Analiza koda, prometa i resursa aplikacija, automatsko otkrivanje ranjivosti

Minimiziranje sigurnosnih prijetnji

- Redovito ažuriranje sigurnosnih protokola
- Edukacija korisnika i programera
 - Što bi korisnici trebali znati: dozvole, izvori aplikacija, sumnjive poruke
 - Što bi programeri trebali znati: sigurno kodiranje, zaštita podataka, testiranje sigurnosti
- Praćenje novih sigurnosnih prijetnji
 - Kako pratiti: sigurnosni blogovi, izvješća, alarmi
 - Koristiti *Play App Signing*

Zaključak

- Android sigurnost kontinuirana evolucija
- Višeslojna arhitektura je temelj
- Pohrana s ograničenim pristupom štiti privatnost
- Razumijevanje prijetnji neophodno
- Izazovi traže rješenja
- Dinamične dozvole obećavaju napredak
- Edukacija i suradnja ključni
- Kontinuirano poboljšanje nužno

Literatura

- Mayrhofer, René, et al. "The Android Platform Security Model." ACM Transactions on Privacy and Security (TOPS) 24.3 (2021): 1-35.
- Lee, Yu-Tsung, Haining Chen, and Trent Jaeger. "Demystifying Android's Scoped Storage Defense." IEEE Security & Privacy 19.5 (2021): 16-25.
- Enck, William, Machigar Ongtang, and Patrick McDaniel. "Understanding Android Security." IEEE Security & Privacy 7.1 (2009): 50-57.

Dodatna literatura

- [Android Developers - Security Overview](#)
- Bathia, Gaurav, et al. "Security vulnerabilities in android applications." International Journal of Computer Applications 45.18 (2012).
- [OWASP Mobile Security Project. OWASP Mobile Security Project](#)
- Zhou, Yajin, and Xuxian Jiang. "Dissecting Android malware: Characterization and evolution." 2012 IEEE Symposium on Security and Privacy. IEEE, 2012.
- Nash, Kristen, et al. Challenges and Directions for Android App Security. Proceedings of the 2018 IEEE Conference on Communications and Network Security (CNS), IEEE, 2018.
- Wei, Xuxian, et al. A permission-based model for Android security analysis. Proceedings of the 4th ACM conference on Security and Privacy in Wireless and Mobile Networks, 2011.

Hvala!