Modeli sigurnosnog rješenja za mobilne uređaje zasnovanih na Android operativnom sistemu

Miroslav Ćajić , Bogdan Brkić

Sadržaj — U ovom radu predstavljeni su modeli sigurnosnog rješenja za tajni prenos podataka između dva mobilna uređaja koji rade pod Android operativnim sistemom. Izvršena je analiza postojećeg načina komunikacije i predložena su sigurnosna rješenja od kojih je jedno sa vlastitim kriptografskim algoritmom. Kao primarni izvor podataka koristi se Android aplikacija.

Ključne riječi — Android, arhitektura, vlastiti algoritam, kriptografski sistem.

I. UVOD

POVEĆANA potreba za fleksibilnošću softvera mobilnog telefona dovela je do razvoja novog operativnog sistema,koji sa sobom donosi niz pogodnosti koji običnom korisniku olakšavaju rukovanje uređajem. Neke od osobina su:

- Prilagodljivost korisniku,
- Korisnički interfejs,
- Korisničke aplikacije,
- Integracija više uređaja u jednom,
- Upravljivost uređajem i
- Baza podataka.

Da bi usavršili upravljivost mobilnih teleofna, proizvođači su za svaki novi hardver pravili i odgovarajući softver. Izlazak na tržište nove serije određenog tipa telefona zahtijevalo je i novu verziju softvera ili reviziju već postojećeg. Situacija se promijenila proizvodnjom operativnih sistema koji su mnogo doprinijeli poboljšanju funkcionalnosti i upravljivosti mobilnih telefona.

U ovom radu razmatra se nekoliko scenarija za zaštićeni prenos podataka za mobilne telefone čiji se rad bazira na Android arhitekturi. Razmatrani operativni system je pogodan za realizaciju zaštićenih aplikacija. Posebno je interesantan profesionalnim organizacijama kao što su: vojska, MUP, službe bezbjednosti, državne organizacije, banke i sl. Pomenute organizacije iz bezbjedonosnih razloga uvijek imaju interes za realizaciju vlastitih kriptografskih rješenja. Za realizaciju vlastitih, ili za modifikaciju postojećih rješenja zaštite neophodan uslov koji se odnosi na dostupnost izvornog (source) kôda. Pored kriptografskih algoritama posebna pažnja se posvećuje distribuciji kriptoloških ključeva. Poznato je da je u korektno realizovanim rešenjima ukupna bezbjednost svedena na bezbjednost i kvalitet kriptoloških ključeva koji se primjenjuju u konkretnok kriptološkom rješenju. U ovom radu razmatrana su standardna rješenja distribucije ključeva i predložena je njihova modifikacija u cilju postizanja većeg stepena bezbjednosti i povjerenja u ukupan bezbjedonosni sistem.

Miroslav Ćajić dipl.ing.inf.-master, MikroByte doo, Vesa Rackovića 16, 73220 Rogatica, tel.: +387 65/631-675, fax: 387 58/417-517, e-mail: info@mikrobyte.com

Bogdan Brkić dipl.ing.inf.-master, Ministarstvo finansija Republike Srpske, Kralja Petra I Karađorđevića, 78000 Banja Luka, tel.: +387 65/567-941, e-mail: bbogdan@teol.net

II. ANDROID ARHITEKTURA

Android je operativni program namijenjen mobilnim telefonima, koji se sastoji od operativnog sistema, međuslojeva i ostalih ključnih programa. Jezgro Androida čini Linux Kernel, verzija 2.6., što omogućava sistemu povećanu sigurnost kao temelj stabilnosti sistema. Odlikuje se sistemom za upravljanje memorijom i procesima kao i mrežnim uslugama. Kernel takođe ima ulogu međusloja između hardvera i ostatka softverskog dijela. Za Android možemo reći da je zasnovan na Linuxu ali Android, nije Linux. Od Linuxa se razlikuje po tome što:

-Ne podržava windowing system
- Ne podržava glibc biblioteke
-Ne uključuje cijeli skup standardnih Linux komunikacionih usluga.
-Podržava samo standardni Linux2.6.24 Kernel i više
-Koristi Kernel Enhancements, za Android podršku

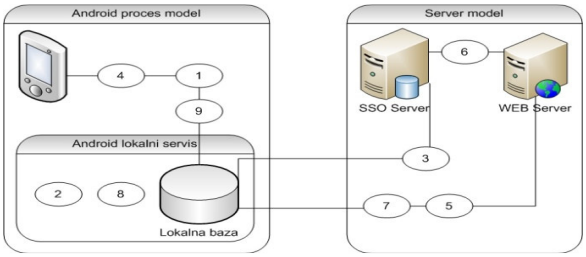
Android je u početku razvijan samostalno od strane Google-a, da bi se poslije pridružio u OHA (Open Handset Alliance). Jedan dio Android operativnog sistema je razvijan privatno od strane različitih programera i naziva se *Cupcake.* Cupake je naziv za update sistema koji nije zvanično potvrđen od Google-a. Android je sastavljen od nekoliko bitnih i zavisnih dijelova kao što su:

- **Linux kernel** operativni sistem - omogućava komunikaciju na nivou hardvera, upravlja memorijom, kontroliše procese, vrši optimizaciju aplikativnog softvera za mobilne uređaje.
- **Hardverski referentni dizajn** - opisuje niz sposobnosti kako bi bila obezbijeđena softverska podrška potrebna za mobilne uređaje.
- **Open Source biblioteke** – koriste se za razvoj aplikacija, uključujući SQLite, WebKit, OpenGL, i Media Manager.

- **Run Time** - koristi se za izvršavanje dominantnih Android aplikacija, uključujući DVL (Dalvik Virtual Mashine) kao i za osnovne biblioteke za pružanje specifične Android funkcionalnosti. Run Time je osmišljen kako bi se ostvarila mala učinkovitost u radu memorije pri upotrebi mobilnih uređaja.
- **Dalvik virtual machine -** vrši optimizaciju softvera za mobilne uređaje.
- **Application framework** - vrši prezentaciju usluga aplikativnom sloju, omogućava ponovnu upotrebu i zamjenu komponenti uključujući Window Manager, Content Providers, Location Manager, telefoniranje i peer-to-peer servise.

III. PRIMJENA JEDNOSTAVNE ANDROID ARHITEKTURE

Jednostavan Android proces model sastoji se od strane koja šalje podetke (pošiljalac) i strane koja prima podatek (primalac). Svaka komunikacija sastoji se od niza elemenata koji prethode sigurnoj komunikacionoj transmisiji. U ovom slučaju jedna od komunikacionih strana je Android proces model a druga strana je Server model. Komunikacija se vrši posredstvom Android lokalnog servisa.



Sl.1. Primjer primjene jednostavne Android arhitekture.

Sl. br. 1 opisuje sledeću situaciju:

1. Prijava na lokalni servis, direktno na mobilnom uređaju.
2. Android aplikacija posjeduje određeni SSL sertifikat na osnovu kog pristupa bazi podataka.
3. Korisiti se međusobni ključ za autentifikaciju. Nakon uspješne razmjene ključeva potvrđuje se autentifikacija. SSO token određene dužine (128 ili 256 bita-a) je vraćen u Android lokalnu bazu.
4. Korisnik vrši zahtjev za podacima iz lokalne baze podataka.
5. Ako su traženi podaci u lokalnoj bazi oni se odmah dobavlju, u suprotnom pravi se zahtjev prema WEB serveru. U tom slučaju korisničko ime i lozinka se ne šalju WEB serveru već se samo šalje SSO token.
6. WEB server zahtijeva autentifikaciju od SSO servera na osnovu SSO tokena.
7. Ako je SSO token ispravan, traženi podatak se vraća u lokalnu bazu podataka.
8. Podatak se čuva u lokalnoj bazi podataka.

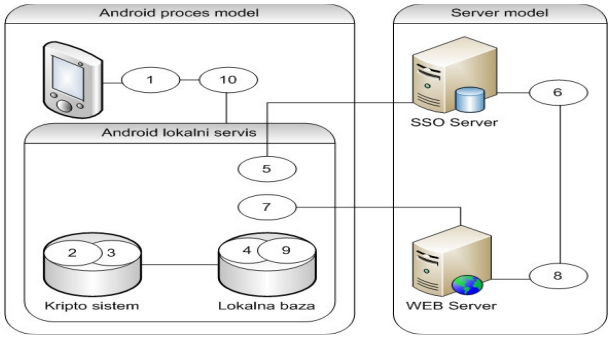
Da bi se omogućio povratak korisničkih podataka na lokalnu bazu potrebno je u nekim dijelovim Android arhitekture izbjeći složenosti operacija. Ograničenje koje se javlja u ovom dijelu arhitekture odnosi se na lokalnu bazu podataka koja ne provjerava ispravnost SSO tokena sa SSO servisom. Takođe, šifrovanje nije izvedeno na lokalnom nivou, kao i problem slanja korisničkog imena i lozinke preko mreže.

IV. PRIMJENA SLOŽENE ANDROID ARHITEKTURE

Jedan od uslova za uspješno implementirano rješenje sigurnosnog modela predstavlja i njegova jednostavnost upotrebe. Cilj je da krajnji korisnici što manje u radu osjećaju prisustvo kriptografskog sistema. To podrazumijeva jednostavnost posla koji obavljaju, statičnost postojećih funkcija i vemenskog perioda potrebnog za obavljanje konkretnog poslovnog zadatka.

Model implementacije bezbjedonosnog rješenja komunikacionog kanala obezbeđuje odličan metod za zaštitu osjetljivih podataka koji se prenose od predajne do prijemne strane. Ipak, usložnjavanje modela dovodi do smanjenja performansi i otežava konkretnu upotrebu.

Složeni Android proces model za osnovu ima već pomenuti jednostavni proces model koji je nadograđen još sa kripto sistemom.



Sl..2. Primjer primjene složene Android arhitekture.

Prema slici br.2. imamo sledeći scenario:

1. Korisnik unosi korisničko ime i lozinku u telefon,
2. Korisničko ime i lozinka ovog puta se šalju kao šifrovani podaci na osnovu kojih se provjerava identitet korisnika. Ovaj podatak se šifruje ključem dužine 256 bita, upotrebom AES ili nekog drugog algoritma. U upotrebi će biti asimetrični šifarski sistem sa javnim i privatnim ključevima.
3. Privatni ključ se upotrebljava za dešifrovanje simetričnog niza. Simetrični niz se koristi za šifrovanje i dešifrovanje bilo kog lokalno sačuvanog podatka.
4. Svi podaci koji se nalaze u lokalnoj bazi podataka su šifrovani pomoću simetričnog kripto sistema.
5. Ako se traženi podaci ne nalaze u lokalnoj bazi onda se pravi zahtjev prema SSO serveru. Ako SSO server nema tražene podatke, onda prosljeđuje zahtjev za traženin podacima prema WEB serveru. U ovom

dijelu scenario se poklapa sa prethodnim gdje se koristi privatni ključ za pretragu podataka. Korisničko ime i lozinka se ne šalju preko mreže. SSO server prepoznaje korisnika i na osnovu ovh zahtjeva vraća alfa-numerički znak određene dužine.

6. SSO server generiše token koji će automatski isteći nakon određenog vremenskog perioda.
7. Nakon prijema tokena Android servisna aplikacija može početi komunikaciju sa WEB serverom koja se zasniva na istom nivou na kom se nalazi SSO usluga. Javna odnosno privatna infrastruktura se ponovo koristi za podešavanje sigurnosnog kanala u komunikaciji između telefona i servera. Potvrda servisa koji ima glavnu WEB uslugu proističe iz iste potvrde o autorstvu koja je dobijena sa telefonom.
8. Po prijemu zahtjeva, SSO token je izdvojen iz samog zahtjeva.
9. Po prijemu podataka, vrši se šifrovanje prije nego što podaci budu upisani u bazu podataka,
10. Podaci su vraćeni korisniku.

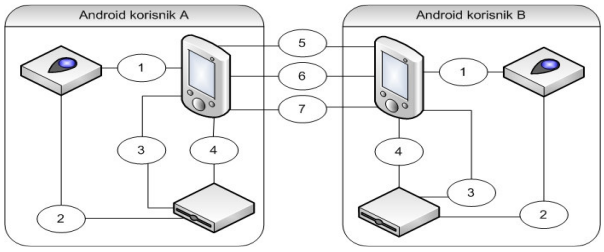
A. *Sigurnosne napomene*

Javni odnosno privatni ključ koji je generisan od strane korisnika, nalazi se direktno na telefonu u vrijeme kada je unesen u telefon, i kao takav, privatni ključ nikada nije napustio telefon niti je bio prenesen preko bilo koje mreže. U vrijeme kada se korisnički podaci unose u telefon, potvrda o korisniku mora biti najmanje jednom unesena u SSO aplikaciju. Važno je napomenuti da je aplikacija razvijana sa CA sertifikatima od određene organizacije čiji je i WEB server i kao takva nije podložna "Man-in-the-middle" napadima.Ako korisnik izgubi uređaj, svi lokalno sačuvani podaci su nečitljivi. Simetrični ključ koji se koristi za šifrovanje je takođe nekoristan. Javni, odnosno, privatni ključevi koji se koriste kao osnova za šifrovanje su zaštićeni sigurnosnom lozinkom koja je u ovom lancu najslabija karika. Ako korisnik unese veoma jednostavnu lozinku, neovlašteni korisnik može lako doći do privatnog ključa a samim tim i do lokalnih podataka. Takođe, moguće je sprovesti politiku jakih lozinki direktno iz korisničke aplikacije. Gledano sa korisničke strane ovaj scenario iako je dosta komplikovaniji, za korisnika je isti. Potrebno je da korisnik unese određeno korisničko ime i lozinku koja će biti verifikovana unutar sigurnosnog sistema.

Ovaj sigurnosni model može se nadograditi prilikom procesa prijavljivanja korisnika na uređaj, kao i dodatnom upotrebom kriptografskog sistema, što je prikazano u nastavku ovog rada.

V. ANDROID SIGURNOSNO RJEŠENJE U KOMUNIKACIJI

Dosadašnja dva primjera pokazuju sigurnosnu intervencuju kada je u pitanju dobavljanje podataka u mobilni telefom. Sledeći primjer je baziran na zaštiti govornog kanala koji se uspostavlja između dva ili više korisnika. Ovaj scenario, prikazan slikom br.3., je dodatno proširen u odnosu na prethodni. Proširenje se ogleda u sigurnosnoj nadogradnji prilikom procesa prijavljivanja korisnika na uređaj, kao i načina šifrovanja i dešifrovanja ulazno-izlaznog podatka iz mobilnog uređaja. Prijemna i predajna strana posjeduju identičan komunikacioni element koji je, u ovom slučaju externi memorijski disk.



Sl. 3. Primjer primjene Android sigurnosnog rješenja u komunikaciji.

Prilikom odabira zaštićenog načina komuniciranja, korisnik umjesto sigurnosne lozinke i korisničkog imena treba da ostavi određeni biometrijski dokaz. To može biti otisak prsta, sken zjenice oka ili glasovna komanda. Umjesto ove biometrijske kontrole moguće je koristiti neki drugi akreditiv kao što je smart kartica ili slično. Svaki učesnik u razgovoru koristi isti externi nosač podataka na kom je smješten GNK (Generisani Niz Ključeva). Za svaku započetu iteraciju koristi se jedan ključ koji se nakon završene komunikacije odbacuje na određeni način. Odbacivanje se može vršiti fizičkim brisanjem tog ključa ili uvođenjem indexa upotrebe za svaki pojedinačni ključ. Sledeći redni ključ za šifrovanje je sledeći ključ koji je na redu u nizu ključeva, ili je to pseudo ključ. Upotreba pseudo ključeva moguća je samo kod asimetričnog kriptografskog sistema pošto se pseudo ključevi različito generišu kod svakog korisnika. Nastavak ovog scenarija ograničen je samo na sihroni kriptografski sistem. Pošto je jednom upotrebljen ključ za šifrovanje odbačen, na osnovu izvještaja IUK (Index Upotrebljenih Ključeva) sigurnosni servis se pozicionira na sledeći ključ. U slučaju totalne ili djelimične upotrebljenosti ukupnog broja ključeva, svaki korisnik ponovo dobija novu externu memorijsku jedinicu sa novim GNK. Ukoliko se komunikacija proširi na više novih korisnika koji posjeduju validnu externu memorijsku jedinicu, generator IUK mora se resetovati na određenu vrijednost koja je zajednička za sve korisnike u komunikaciji. Vrijednost na osnovu koje se resetuje generator ključeva ne mora biti tajna, ali se ne smije ponoviti.

U slučaju gubitka externe memorijske jedinice od strane jednog korisnika, potrebno je sve jedinice koje sadrže GNK u potpunosti zamijeniti drugim jedinicama. Ako se biometrijski podaci na osnovu kojih se vrši akreditovanje korisnika nalaze pohranjeni na ovoj memorijskoj jedinici i sa njih se vrši akreditovanje korisnika, u slučaju gubitka ili oštećenja iste nije potrebno vršiti zamjenu ostalih jedinica. Razlog tome je što se neovlašteni korisnik na osnovu svojih biometrijski parametara ne može validno predstaviti sistemu i samim tim nije u mogućnosti da ispravno koristi mobilni uređaj.

Postupak komuniciranja kroz šifrovani kanal bio bi sledeći:

1. Korisnik šalje zahtjev za upotrebu EM (Externa Memorija).

2. Na osnovu svojih biometrijskih parametara korisnik se prijavljuje u sistem.
3. Ukoliko je biometrijski dokaz validan uspostavlja se interni komunikacioni kanal između EM i telefona korisnika.
4. Da bi se komunikacioni kanal sihronizovao potrebno je da sigurnosni servis korisnika koji započinje itreciju, dobije određenu vrijednost na osnovu IUK-a.
5. Vrijednost IUK-a se šalje kroz komunikacioni kanal. Na strani pošiljaoca, vrijednost IUK-a dobija drugu veličinu a stara vrijednost se privremeno memoriše u RAM memoriji telefona. Sigurnosni servis prijemnog telefona prihvata ovu vrijednost i na osnovu nje pozicionira se za čitanje određenog ključa.
6. Kada je sihronizacija kanala završena, strana pošiljaoca dobija povratni signal za početak slanja signala. Vrijeme potrebno za sihronizovanje komunikacionih strana zavisi do kvaliteta komunikacionog kanala.
7. Pošiljalac počinje da emituje podatke koji su šifrovani na osnovu vrijednosti iz GNK. Šifrovanje se vrši pomoću vlastitog šifarskog algoritma. Prijemna strana prihvata šifrovani signal i dešifruje ga na osnovu iste vrijednosti iz svog GNK.

VI. ZAKLJUČAK

U budućnosti složenu Android arhitekturu moguće je proširiti tako da se simetrični ključ zapisuje direktno na serveru. U slučaju gubljenja ključa, podaci na serveru su još uvijek nečitljivi zbog toga što se simetrični kriptovani niz još uvijek nalazi na serveru. Ako bi neovlašteni korisnik pokušao koristiti telefon, bilo bi dovoljno vremena da se prijavi mogući pokušaj neovlaštene upotrebe zbog krađe ili gubljenja uređaja, te da se izbriše određeni korisnički nalog sa servera. Osim toga, ključ koji se nalazi na serveru i dalje je šifrovan. Čak i ako je ovaj ključ presreten u komunikaciji, beskoristan je bez privatnog ključa. Ova arhitektura se mogla proširiti još u jednom smjeru. U toku prijema na lokalnu bazu može se zahtijevati SSO token i zatim zahtijevati potvrdu identiteta od SSO servera. Na ovakav način bi se spriječilo da pojedinac koji zna tuđe korisničke podatke pristupi podacima na lokalnoj bazi podataka.Pošto Android arhitektura nema vlasničke programe za upravljanje hardverom uređaja, programerima je dozvoljeno implementiranje vlasničkih drajvera u dijelu koji se odnosi na HAL. Pošto se HAL brine o ispravnom funkcionisanju drajvera, ovdje postoji mogućnost redirektovanja određenih instrukcija sigurnosnog sistema od strane programera. Posebna olakšica kod Android operativnog sistema je implementacija sopstvenog sigurnosnog rješenja kao što je prikazano u ovom radu. Ovo se odnosi na rješenja koja zahtijevaju određenu mjeru sigurnosti tokom prenosa govornog signala ili prenosa podataka. Implementacijom vlastitiog sigurnosnog rješenja.

Sigurnosno rješenje može se posmatrati kao komunikacioni sistem, gdje se informacija koja je generisana na predajnoj strani dostavlja željenom odredištu, odnosno, prijemnoj strani poruke. Poruka se šalje kroz otvoreni prostor i izložena je svi napadima koji se odnose na presretanje i modifikaciju podatka. Ključni elementi u komunikaciji između dvije strane su: povezivanje, generisanje signala, kriptološka sinhronizacija, razmjena podataka, otkrivanje i ispravljanje grešaka, zaštita, upravljanje, nadgledanje komunikacionog kanala i sl. Zadatak ovako formiranog sigurnosnog modela jeste poštovanje osnovnih principa bezbjednosti, odnosno, povjerljivosti i privatnosti, provjere identiteta, poštovanja integriteta, raspoloživosti podatka, kontrole sistema i neporicanja sprovedenih akcija.

Implementacijom vlastitog kriptografskog rješenja dobijamo potpuno nov, vlastiti proizvod, sa vlastitim rješenjem. Android platforma programerima pruža mogućnost izrade novog sigurnosnog servisa koji će u potpunosti biti prilagođen za potrebe određenih institucija čije se poslovanje bazira na tajnom prenosu podataka. Realno je očekivati da ćemo u narednih nekoliko godina imati kriptološke sisteme koji su integrisani u jednu cjeloviti komponentu čineći pouzdan i siguran sistem za prenos podataka, na bazi domaćeg rješenja.

LITERATURA

[1] Android A Programmers Guide, J.F. DiMarzio, *Mc Graw Hill*, 2008.
[2] Android Essentials, Chris Haseman, *Apress*, 2008.
[3] Applied Cryptanalysis, Mark Stamp, Richard M. Low, *Wiley Intersciene*, 2007.
[4] Applied Cryptography, Bruce Schneuer, 2en Edition, *WileyPublishing*, 2007.
[5] Professional Android Application Development, *Reto Meier, WROX*, 2008.
[6] Thoughts on Google Android, B. Smith, *Spectrum Data Technologies*, 2008.
[7] GSM DTI Operation and Maintenance, Ericsson RadioSystems AB, Stockholm, 1998.
[8] William Stallings, *Data and Computer Communication*, Pearson Prentice Hall, 2004, NJ, USA.
[9] George Beekman, Eugene J. Rathswohl, *Computer Confluence*, Prentice Hall, 2003, NJ, USA.
[10] Osnovi bezbednosti i zaštite informacionih sistema Milan Milosavljević i Gojko Grubor, *Univerzitet Singidunum*, 2006.
[11] http:// android.git.kernel.org
[12] http://android-developers.blogspot.com
[13] http://code.google.com
[14] http://developer.android.com
[15] http://source.android.com

ABSTRACT

This paper presents models of security solutions for the secret transfer of data between two mobile devices that operate under the Android operating system. In this paper analysis of the existing methods of communication and security solutions are proposed. Although, proprietary cryptographic algorithm is suggested. As the primary source of data the Android applications are used.

Models of security solutions for mobile devices based on Android operating system

Miroslav Ćajić, Bogdan Brkić