Sigurnosni aspekti pametnih telefona

Medakov, Viktor

Sveučilište Jurja Dobrile u Puli
Fakultet Informatike u Puli

VIKTOR MEDAKOV

SIGURNOSNI ASPEKTI PAMETNIH TELEFONA

Završni rad

Pula, rujan 2020. godine

Sveučilište Jurja Dobrile u Puli
Fakultet Informatike u Puli

VIKTOR MEDAKOV

SIGURNOSNI ASPEKTI PAMETNIH TELEFONA

Završni rad

JMBAG: 0303075408, redoviti student
Studijski smjer: Informatika
Kolegij: Osnove IKT
Znanstveno područje : Društvene znanosti
Znanstveno polje: Informacijsko-komunikacijske znanosti
Znanstvena grana: Informacijski sustavi i informatologija
Mentor: Doc.dr.sc. Snježana Babić

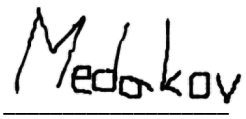
Pula,rujan 2020. godine

IZJAVA O AKADEMSKOJ ČESTITOSTI

Ja, dolje potpisani Viktor Medakov, kandidat za prvostupnika informatike ovime izjavljujem da je ovaj Završni rad rezultat isključivo mojega vlastitog rada, da se temelji na mojim istraživanjima te da se oslanja na objavljenu literaturu kao što to pokazuju korištene bilješke i bibliografija. Izjavljujem da niti jedan dio Završnog rada nije napisan na nedozvoljen način, odnosno da je prepisan iz kojega necitiranog rada, te da ikoji dio rada krši bilo čija autorska prava. Izjavljujem, također, da nijedan dio rada nije iskorišten za koji drugi rad pri bilo kojoj drugoj visokoškolskoj, znanstvenoj ili radnoj ustanovi.

U Puli, 15.09.2020. (datum)

Student :



IZJAVA

o korištenju autorskog djela

Ja, Viktor Medakov dajem odobrenje Sveučilištu Jurja Dobrile u Puli, kao nositelju prava iskorištavanja, da moj završni rad pod nazivom SIGURNOSNI ASPEKTI PAMETNIH TELEFONA koristi na način da gore navedeno autorsko djelo, kao cjeloviti tekst trajno objavi u javnoj internetskoj bazi Sveučilišne knjižnice Sveučilišta Jurja Dobrile u Puli te kopira u javnu internetsku bazu završnih radova Nacionalne i sveučilišne knjižnice (stavljanje na raspolaganje javnosti), sve u skladu s Zakonom o autorskom pravu i drugim srodnim pravima i dobrom akademskom praksom, a radi promicanja otvorenoga, slobodnoga pristupa znanstvenim informacijama. Za korištenje autorskog djela na gore navedeni način ne potražujem naknadu.

U Puli, 15.09.2020.(datum)

Potpis

SADRŽAJ

Uvod

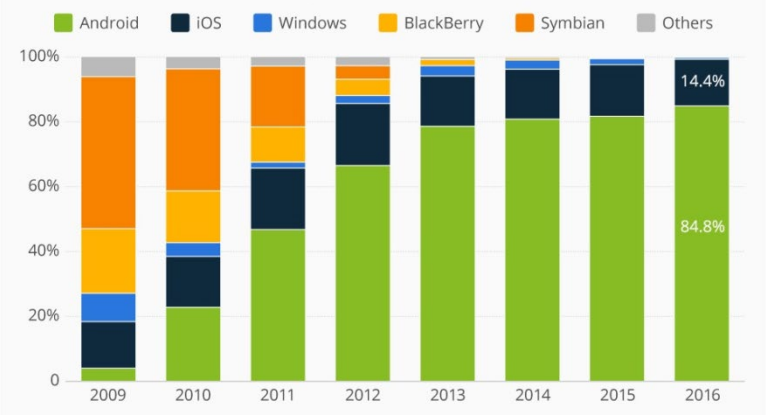
Danas, pametni telefoni i njihove aplikacije dosegli su vrhunac popularnosti gdje više od trećine stanovništva posjeduje pametni telefon. Kroz prošlo desetljeće, mogućnosti i usluge koje pametni telefoni pružaju kao što su usluge plaćanja putem pametnih telefona, tržište mobilnih aplikacija, dijeljenje podataka kao i vrste mrežne povezanosti kao što su 3G,4G i Wi-fi značajno su se povećale. 190 milijardi aplikacija preuzeto je samo u 2019. godini. To sa sobom donosi velike izazove u polju sigurnosti i privatnosti jer se isto tako povećao se i broj prijetnji koji je velik i svatko tko koristi pametni telefon može biti napadnut. Korisnici su tim prijetnjama izloženi dok pretražuju internet ili preuzimaju aplikacije od nepoznatog, nepouzdanog ili neprovjerenog izvora. Budući da je *Android OS* najkorišteniji operacijski sustav za pametne uređaje te je uz to i *open-source* što znači da svatko može kreirati svoje aplikacije koje iskorištavaju ranjivosti operacijskog sustava sa namjerom da dođe do podataka bez autorizacije korisnika. Korisnici su svjesni o raznim specifikacijama svojih pametnih telefona i kompanijama koje ga proizvode no nisu svjesni o ranjivostima mobilnih operacijskih sustava i njihovim sigurnosnim aspektima te je kao takav vrlo bitan i ljuski faktor i koliko su korisnici osviješteni o rizicima kojima su izloženi prilikom korištenja pametnog telefona. Tu dolazimo do pojma *„Social engineering"* koji označava širok spektar zlonamjernih aktivnosti kojima se korisnika psihološkim trikovima kao što su lažne e-mail i SMS poruke te reklame prevari da napravi određenu interakciju kojom će preuzeti zlonamjerni kod ili aplikaciju te sebe i svoje podatke izložiti napadu. U ovom radu će biti obrađena povijest pametnih telefona od njihovih početaka i svega nekoliko funkcionalnosti do njihovog uspona i postanka neizostavnim faktorom u današnjem životu. Obraditi ću sigurnost *Android* i *iOS* operacijskih sustava, model prijetnji koji obrađuje moguće aktivne napade na korisnika putem raznih tehnologija kao što su Wi-fi ili Bluetooth koje svakodnevno koristimo ne razmišljajući o tome kako ih netko može iskoristiti protiv nas da bi pristupio privatnim podacima. Također obradit ću ne samo softverske napade koje iskorištavaju ranjivosti operacijskog sustava putem zlonamjernih aplikacija već i one hardverske ranjivosti primjerice putem USB konekcije kao i protumjere za one napade koje je moguće spriječiti jer se direktne napade koji su usmjereni na individualnu osobu ili organizaciju vrlo teško može izbjeći za razliku od napada koji su usmjereni na nasumične korisnike koji ovise o sebi i svojim odlukama.

1. Povijesni razvoj mobilnih platformi

Početak popularnosti pametnih telefona počeo je 2007. godine kada je Steve Jobs, jedan od osnivača Apple kompanije svijetu predstavio *iPhone*, revolucionarni mobilni uređaj sa zaslonom na dodir te mogućnošću komuniciranja putem interneta na razini Desktop uređaja kao i mogućnost korištenja takozvanih *maps* za navigaciju. iPhone je uveo potpuno novo korisničko sučelje i revolucionarni softver koji je promijenio industriju mobilnih uređaja. Iako je *iPhone* na tržište izašao 2007. godine pametni telefoni su postojali još od 1993. godine. Razlika između današnjih pametnih telefona i onih na samim počecima je ta što su prvi pametni telefoni bili namijenjeni za korisnike koji su radili u poslovnim poduzećima te si obični korisnici iste nisu mogli priuštiti. Era pametnih telefona može se podijeliti u 3 faze (Sarmar & Soomro, 2013) .

Prva faza je bila isključivo namijenjena za poslovna poduzeća i korporacije te su se njihove mogućnosti prilagođavale po potrebama svakog poduzeća. Prvu fazu započeo je pametni telefon pod imenom *„IBM Simon"* kojega je proizvela kompanija IBM. *Simon* je bio u mogućnosti upravljati faks uređajem te korisničkim podacima. Također je imao aplikacije za adresar i kalkulator. Prema (Sarmar & Soomro, 2013) *Blackberry* je smatran revolucionarnim uređajem prve faze kojeg je proizvela kompanija RIM (*Reasearch in motion)* , danas pod imenom Blackberry Limited. *Blackberry* uređaji imali su nove funkcionalnosti kao što su Email, pretraga Web-a, integrirani mikrofon i zvučnik te možda najznačajnije funkcionalnosti, kameru i QWERTY tipkovnicu. *Blacberry* je možda bio revolucionaran sa svojim funkcionalnostima, finski proizvođač mobitela Nokia bio je vođa na tržištu mobitela te im nije dugo trebalo da se prilagode novim izazovima koje su rane 2000-te godine donosile kao što su promjene u bežičnoj i internetskoj tehnologiji te 2001. godine na tržište izbacuje model *Nokia 7650*, njihov prvi pametni telefon sa kamerom i zaslonom u boji. Važnu ulogu ima *Symbian* operacijski sustav koji se na svojim počecima zvao *EPOC* razvijen od kompanije *Psion.* U zajedničkom pothvatu sa globalnim proizvođačima mobitela Nokia,Ericsson i Motorola, Psion je postao *Symbian* a *EPOC* je postao *Symbian OS.* Nokia je poslovala odlično na tržištu sa svojim pametnim telefonima pogonjenim *Symbian OS* operacijskim sustavom te je 2003. prodala 250 milijuna jedinica modela *Nokia 1100* čineći ga najprodavanijim uređajem na svijetu u to vrijeme (Sarmar & Soomro, 2013).

Druga faza počinje dolaskom već spomenutog revolucionarnog *iPhone* pametnog telefona na tržište. Prije dolaska *iPhone* pametnog telefona i *iOS* operacijskog sustava,*Symbian OS* bio je vodeći operacijski sustav bez ikakve konkurencije na tržištu. Također je krajem 2007. godine Google prikazao svoj *Andorid* operacijski sustav sa namjerom da uđe na tržište pametnih mobitela. U ovoj fazi najbitnije je bilo uvesti nove mogućnosti za korisnike te pritom zadržati cijene pametnih telefona povoljnima kako bi se privukla pozornost kupaca. Mogućnosti kao integracija društvenih mreža te audio/video sadržaji bili su posebno naglašeni. Treća faza se odnosi na smanjenje razlike između pametnih uređaja namijenjenih za poslovna poduzeća te opće tržište kao i stabilizacija mobilnih operativnih sustava te njihovo daljnje poboljšanje i uvođenje novih mogućnosti. Kao što je i vidljivo na slici ispod, tržište operacijskih sustava pametnih telefona znatno se mijenjalo tijekom godina. Neki su operacijski sustavi ugašeni dok su neki kao zavladali tržištem. Podaci na slici ispod prikazuju podatke od 2009. do 2016. godine. *Symbian OS* 2009. bio je vodeći na tržištu a *Blackberry* na drugom mjestu. Dvije godine kasnije stvari se mijenjaju i *Android* operacijski sustav preuzima vodstvo na tržištu. Apple-ov *iOS* dijeli drugo mijesto sa *Blackberry* operacijskim sustavom. Kroz naredne 4 godine *Android* i *iOS* potpuno vladaju tržištem dok se ostali operacijski sustavi gotovo pa i ne upotrebljavaju. Razlog velike popularnosti *Android* operacijskog sustava je to što je *open-source* te ga svaki proizvođač pametnih telefona može koristiti i prilagođavati svojim potrebama unutar određenih granica dok je *iOS* ostao zatvoren za javnost i uređaji koji koriste *iOS* su proizvedeni isključivo od Apple kompanije (Sarmar & Soomro, 2013) .



Slika 1 Tržište operacijskih sustava , F,Richter,
https://www.statista.com/chart/4112/smartphone-platform-market-share/ , 14.09.2020.

2. Pregled sigurnosti operacijskih sustava

U okviru ovog poglavlja objasnit će se ključni sigurnosni mehanizmi koji su implementirani u *Andorid* te *iOS* operacijske sustave kako bi se postigla maksimalna moguća zaštita za korisničke podatke.

2.1 Android OS

Preko 2 milijarde uređaja pokrenuto je na Android sustavu te kako bi se zaštitili podaci korisnika koji koriste sustav potrebno je imati dobre obrambene mehanizme od potencijalnih štetnih aplikacija (eng.Potentialy harmful applications). Kako bi to osigurao Google je 2017 godine uveo Google Play Protect,najrasprostanjenija zaštita od prijetnji pametnim uređajima. Google Play Protect na dnevnoj bazi skenira i potvrđuje sigurnost za više od 500 000 aplikacija koje ulaze na tržište. Kako bi dodatno zaštitio korisnike Google je uveo i SafetyNet, niz servisa koji se mogu integrirati sa aplikacijama i igricama. Ti servisi provjeravaju da li je netko promijenio izvorni kod aplikacije, da li one imaju zloćudan kod ili ako sadržaj na određenom URL-u ima zloćudne aplikacije. Prema Google-u u 2018. godini instalacija potencijalno štetnih aplikacija unutar Google Play Store-a bila je na 0.04% dok je izvan Google Play Store-a bila na 0.92% (Android, 2019) .

Sigurnosni model *Android OS* operacijskog sustava uzima prednosti koje pruža od *Linux kernel* na kojemu je i sam android sustav sagrađen. Prilikom samog pokretanja uređaja provjerava se njegova ispravnost to jest da li je izvorni kod Android sustava kompromitiran putem procesa koji se naziva *Verified boot.* Prilikom *Verified boot*-a aktiviran je i proces nazvan *Rollback protection* koji sprječava da se uz sve dodatne provjere sigurnog ažuriranja ne instalira starija i slabije zaštićena verzija sustava koju napadač može iskoristiti. Važan pojam kod sigurnosnog modela android sustava je *Application sandboxing*. Android automatski dodjeljuje jedinstvenu identifikacijsku oznaku (*eng. app ID*) svakoj aplikaciju koja se instalira ili je već instalirana na uređaju. Svakoj aplikaciji dodijeljen je vlastiti direktorij u sustavu i samo ta aplikacija ima dozvolu da provodi operacije čitanja i dodavanja (*eng. read and write rules*) u svoj direktorij čime se ostalim aplikacijama ograničava pristum tom direktoriju, te od tuda naziv *sandbox*. Slijedeći bitan pojam su dozvole (*eng. permissions*). Kako bi aplikacije mogle pružati više funkcionalnosti moraju dohvatiti ostale resurse koji se nalaze u

sustavu, no budući da su aplikacije ograničene na svoje zasebne direktorije i procese, da bi dohvatili dodatne resurse koji su im potrebni moraju zatražiti dozvolu od korisnika. Pomoću dozvole od korisnika koji im daje prava da pristupe ostalim resursima aplikacije imaju pristup hardveru (*eng. hardware*), internetskoj povezanosti , podacima i ostalim uslugama koje pruža operacijski sustav. Kako bi aplikacije unutar sustava mogle međusobno komunicirati koristi se unutar procesna komunikacija (*eng. IPC – Inter Process Communication*). To je potrebno jer kao što je već spomenuto aplikacije se izvode u zasebnim procesima. Jedan od *IPC* sustava naziva se *„Binder“* . On garantira da procesi aplikacija ne mogu biti krivotvoreni. Sve aplikacije moraju biti potpisane od strane proizvođača uključujući već unaprijed instalirane aplikacije na uređaju. Android koristi te potpise kako bi kontrolirao da kada se određena aplikacija ažurira,da bude ažurirana od strane istog proizvođača (Elenkov, 2015).

Android u novijim verzijama koristi *Trusty.* To je sigurni operacijski sustav koji omogućava sigurnosnu egzekucijsku okolinu odnosno na engleskom *TEE* (*eng. Trusted execution environment*). *Trusty OS* se izvodi na istom procesoru kao i *Android OS* no on je izoliran od ostatka sustava od strane hardvera i softvera. *Trusty* i *Androd* se izvode paralelno jedan do drugoga. Izolacija koju *Trusty* pruža štiti korisnika od zlonamjernih aplikacija koje može instalirati na uređaj te od potencijalnih prijetnji koje se mogu u budućnosti otkriti unutar samog Android sustava. Prednost no i mana Android sustava je to što je vrlo rasprostranjen i tu dolazimo do pojma fragmentacije. Uz podsjetnik da je Android *open source* sustav, proizvođači pametnih telefona, unutar granica, mogu raditi sa njime što požele te svaki proizvođač postavlja svoja ažuriranja sustava kada oni to požele. Problem je što neće svaki proizvođač ostati konzistentan sa ažuriranjima softvera, tako će neki korisnici imati novije a neki starije verzije Android sustava. Google nastoji poboljšati situaciju fragmentacije projektom „*Treble“,*projekt kojim bi se promijenila arhitektura Android sustava kako bi se odvojile osnovne inačice sustava od glavnog izvornog koda. Na taj način proizvođači ne bi trebali čekati na ostale dijelove sustava da se ažuriraju kako bi mogli izdati novu verziju sustava (Markota, 2018).

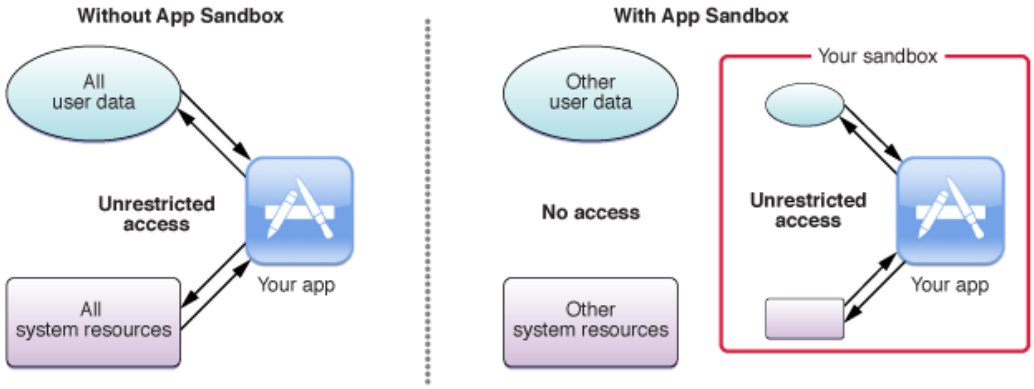
2.2 Apple iOS

iOS je dizajniran sa namjerom da bude maksimalno siguran. Čuvanje informacija sigurnim na pametnim telefonima je nužno, bilo da su to podaci određene kompanije i njezinih korisnika ili privatne fotografije, privatni bankovni podaci i ostale vrste podataka jer podaci svih korisnika su bitni a *iOS* uređaji su napravljeni da održavaju visoku razinu sigurnosti. Prilikom pokretanja, aktivira se *Secure boot chain* te se pokreče kod iz *BOOT ROM-a*. On sadrži Apple certifikat sa javnim ključem kojim se potvrđuje da je sustav prilikom pokretanja valjan i da nije kompromitiran. Ako jedan korak u *secure boot* chain-u nije u mogućnosti verificirati i prijeći na slijedeći, proces se stopira te se korisniku prikaže odgovarajuća poruka sa daljnjim uputstvima. Kako bi se sprječilo da se verzija sustava poništi i instalira starija verzija sustava sa određenim sigurnosnim manama, Apple ima proces koji se naziva *System software personalization* gdje server provjerava koje je verzije sustava dozvoljeno instalirati. Većina sigurnosnih mehanizama je napravljeno tako da ih korisnik ne može podešavati kako njemu odgovara time štiti korisnika da ne isključi sigurnosne postavke slučajnom greškom (Apple, 2012).

Gotovo identično kao i kod Android sustava da bi se provjerilo kako sve aplikacije dolaze od poznatog i potvrđenog izvora one moraju biti potpisane certifikatom kojeg je izdala Apple kompanija. Certifikat se dobiva tako što se svi developeri i kompanija za koju razvijaju aplikacije registriraju u Apple-ovu bazu podataka. Time se proizvođača aplikacije drži odgovornim ukoliko kreira zlonamjernu aplikaciju te daje korisnicima sustava povjerenje da su aplikacije koje koriste i kupuju sigurne. Uz sve ostale metode, Apple koristi tehnologiju *eng. Data protection* kako bi bolje zaštitio podatke spremljene u brzoj memoriji (*eng.Flash memory*). To omogućuje korisniku da odgovori na događaje kao što su nadolazeći pozivi bez potrebe za dekriptiranjem osjetljivih podataka i preuzimanja novih informacija dok je zaključan. Za razliku od Android operacijskog sustava, iOS ne dopušta korisnicima da preuzimaju *eng.* Third *party* aplikacije sa nepoznatih i potencijalno sumnjivih izvora na internetu. Kao i kod Android OS-a *eng. Third Party* aplikacije su *„sandboxed“* i onemogućen im je pristup ostalim datotekama osim svoje. iOS koristi *eng. Address Space Layout Randomization* ili skraćeno ASLR kako bi zaštitio podatke od određenih vrsta napada koji zahtijevaju od napadača da zna gdje se koji program nalazi u memoriji. ASLR nasumično razmješta različite dijelove

programa u memoriji te time onemogućuje napadaču da sazna lokaciju gdje mora ubaciti zlonamjerni program (Maheshika, 2019).

Na *Slika 2* ispod prikazan je već spomenuti *„sandbox“* koji prikazuje sa lijeve strane model sustava kojemu je dopušteno preuzimanje aplikacija sa nepoznatih izvora te sa desne strane *iOS* sustav koji to korisnicima ne dopušta (Medium, 2017, *https://medium.com/@robdeans/exploring-ioss-sandbox-b72e4697ab2f*).



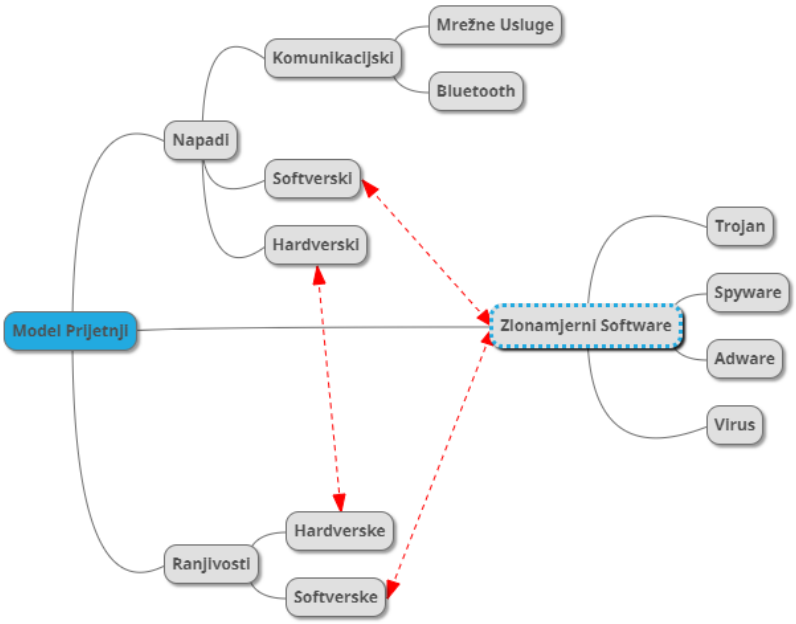
Slika 2 iOS „Sandbox“, R.Deans,

https://medium.com/@robdeans/exploring-ioss-sandbox-b72e4697ab2f , 14.09.2020.

3. Model prijetnji

Pametni mobiteli korisnicima pružaju informacije i usluge koje su im nadohvat ruke kroz razne tehnologije te ih to i čini žarištem osjetljivih informacija do kojih napadači žele doći. Zbog toga privatnost i zaštita podataka ostaju jedan od najbitnijih aspekata te je bitno prikazati kakvim prijetnjama su izloženi pametni telefoni. Kako bi se sastavio kvalitetan model prijetnji, prema (Theoharidou, Mylonas, & Gritzalis, 2012) potrebno je odrediti sredstva i resurse koje pametni telefoni koriste :

- *Podaci* – Podaci su najbitnije i jedino sredstvo koje pametni telefon posjeduje te ono čemu napadač želi pristupiti. Budući da su pametni telefoni vrlo personalizirani uređaji zahvaljujući svojoj višenamjenskoj svrsi , kako će netko zaštiti svoje podatke zavisi od jednog pojedinca do drugog te je teško procijeniti sam rizik od napada.
- *Povezivost* – Mogućnost pametnog telefona da se poveže na razne mreže i koristi bežične tehnologije povećava rizik i otvara vrata raznim napadima.
- *Aplikacije* – Aplikacije su zaštitni znak pametnih mobitela te je njihova popularnost i raznolikost dovela pametne telefone na vrhunac popularnosti no kao i u svim ostalim aspektima , napadači su našli način kako ih iskoristiti za provođenje raznih napada.



Slika 3 Model Prijetnji , Izvor : Autor rada

Prilikom istraživanja iz svih nabrojenih literatura u ovome radu sastavio sam model prijetnji koje mogu naštetiti povjerljivosti, integritetu te dostupnosti podataka kojima se korisnik koristi i koji je prikazan na *Slika 3*. U većini literature zlonamjerni softver te zlonamjerne aplikacije uvijek su u zasebnoj cjelini te se posebno obrađuju kao što su na tu cjelinu nadovezane softverske ranjivosti te napadi. Napadi su podijeljeni na one koje koriste komunikacijsku tehnologiju i bežične mreže te softverski i hardverski napadi. Hardverski napadi iskorištavaju hardverske ranjivosti stoga su oni također povezani. Izazovno je bilo sastaviti ovaj model budući da u nijednoj literaturi on nije potpuno i konkretno napravljen te puno autora ima svoj različit model. U sljedećim poglavljima proći će se kroz navedene prijetnje, načine na koje funkcioniraju te kako se od njih obraniti ili spriječiti da do takvih napada uopće ne dođe.

4. Napadi bazirani na komunikaciji

Napadi bazirani na komunikaciji najčešći su putem bežičnih mreža a sigurnost bežičnih mreža nije ništa drugo nego zaštita osobnih računala, pametnih telefona, tableta i ostalih prijenosnih uređaja koji koriste bežične mreže od mogućih prijetnji. Kada je riječ o bežičnoj tehnologiji prvo je potrebno objasniti pojam bežične komunikacije. Bežična komunikacija je bilo koji tip razmjene podataka između dviju strana koja se odvija bežično to jest „over the air". Primjeri bežične komunikacije su (Islam & Jin, 2019):

- *Wi-fi mrežna komunikacija*
- *Bluetooth komunikacija*
- *Satelitska komunikacija*

Wi-fi odnosno *„wireless fidelity"* odnosi se na bežičnu lokalnu mrežu te je baziran na *IEEE 802.11* standardu. Danas ovakav tip mreže susrećemo gotovo svugdje oko nas. Budući da se ovakav tip mrežne komunikacije tako često koristi, on je također jedan od najvećih problema u aspektu sigurnosti i zaštite podataka. Kada je riječ o Wi-fi bežičnoj komunikaciji uvijek se spominju 3 sigurnosna protokola (Lehembre, 2005):

1. *WEP*
2. *WPA*
3. *WPA2*

WEP ili *Wired Equivalent Privacy* je prvi sigurnosni protkol za Wi-fi mrežnu komunikaciju i bio je standardni protokol od 1999. do 2004. godine. U samim počecima koristio je 64 bitnu enkripciju te je ona kroz godine povećana na 128 bitnu te u konačnici na 256 bitnu enkripciju. Usprkos svim promjenama i unaprjeđenjima kroz vrijeme su se otkrivale mane u ovome protokolu. Kako je računalna snaga rasla postajalo je sve lakše te mane iskorištavati te se ovaj protokol zamijenilo sa boljim. Kako bi se popravile funkcije WEP protokola 2003. godine uveden je WPA ili *Wifi Protected Access*. Ovo se ispostavilo kao privremeno rješenje te je i ono imalo relativno slabu zaštitu no ipak se lakše konfiguriralo. Kako se prelazilo na ovaj protokol morali su se zadržati pojedini elementi njegovog prethodnika kako bi bio kompatibilan sa svim uređajima no to je značilo i zadržavanje određenih sigurnosnih mana. Godinu dana kasnije u 2004. godini WPA2 postao je dostupan. Glavna razlika između WPA i WPA2 protokola je ta što WPA2 koristi napredni enkripcijski protokol to jest *AES* (*eng. Advanced Encryption*

Protocol) koji je mnogo snažniji i najbolja opcija za enkripciju podataka. Jedina značajna ranjivost ovog protokola je što ukoliko naoadač dobije pristup mreži, može napasti ostale uređaje koji su spojeni na tu istu mrežu (Nguyen, 2018), (NetSpot, 2020 https://www.netspotapp.com/wifi-encryption-and-security.html)**.**

4.1 Algoritmi enkripcije

Kako bi se zaštitilo podatke prilikom prijenosa, potrebno ih je kriptirati. Enkripcija općenito je proces transformiranja podataka tako da oni budu nečitljivi onome koji ne posjeduje određeno znanje to jest ključ koji te podatke može dekriptirati i pročitati. Dva su najčešća algoritma bežične enkripcije (Nguyen, 2018) :

1. *Stream cipher* – Kriptira podatke u kriptirani tekst bit po bit.
2. *Block cipher* – Izvršava se na blokovima podataka fiksnih veličina

„Stream cipher" algoritam

Ovaj algoritam kriptira podatke jedan po jedan bit. Koristi beskonačni niz bitova nasumičnih podataka kao ključ. Kako bi ova implementacija zaštite podataka ostala zaštićena, generator nasumičnih podataka treba biti nepredvidiv i isti ključ se nikad ne smije ponovno iskoristiti. Najpopularniji i najviše korišten *„stream cipher"* je *RC4* (*eng. Rivest Cipher 4*). Koristi se u sigurnosnim bežičnim protokolima kao *WEP* i *WPA* te također u *TLS* protokolu. Jednostavan algoritam no i dosta ranjiv te se ne preporučuje za korištenje u novijim sustavima (Nguyen, 2018)**.**

„Block cipher" algoritam

Enkripcija podataka se vrši na način da se kriptira fiksna veličina *n*-bitova podataka odnosno jedan dio ili *block*. Najčešće fiksne veličine jednog dijela su 64 bita, 128 bita i 256 bita. Naprimjer 64 bitni *block cipher* algoritam uzeti će 64 bita podataka te ih kriptirati. Popularni *block cipher* algoritmi (Rawal, Chhikara, Kaur, & Khanna, 2019) , (Nguyen, 2018):

- *DES* – odnosno *Data Encryption Standard* , prije je bio najpopularniji *block cipher*

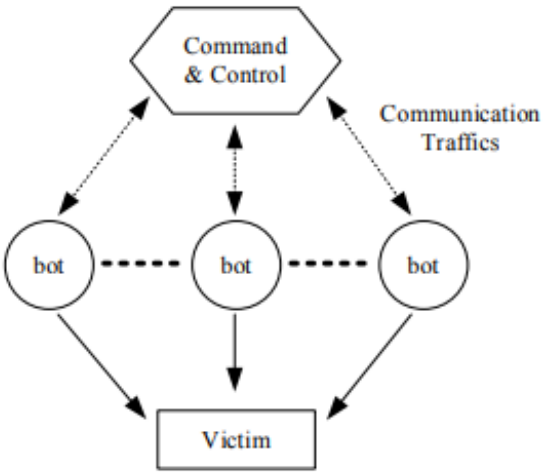
algoritam no vrlo ranjiv na takozvane *„brute force“* napade.

- *3DES* – identičan svom prethodniku *DES*-u, jedina razlika je ta što se *3DES* izvršavao 3 puta. Jači algoritam no 3 puta sporiji te se zbog toga nije uspio probiti na vrh popularnosti kao njegov prethodnik.
- *AES – Advanced Encryption Standard* je najkorišteniji *block cipher* algoritam na svijetu. Veličina *block*-a je 128 bita te podržava 3 veličine ključa : 128,192 i 256 bitova. Što je duža veličina ključa, to je jača enkripcija.

4.2 Napadi putem mrežnih usluga

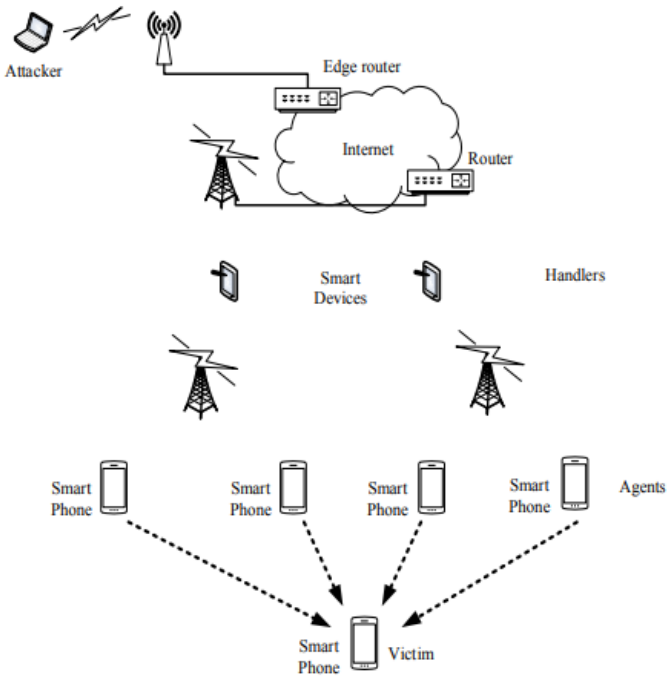
„Distributed Denial of Service“

Napadi koji su usmjereni na onemogućavanje mreže ili smanjivanje njezinih performansi pripisuju se *„DDoS“* (*eng. Denial of service*) napadima. Prema (Cusack, Lutui, & Khaleghparast, 2016) *„Distributed Denial of Service“* napadi su napadi na dostupnost (*eng. availability*) i definirani su kao pokušaj da usluge kao mrežnu povezanost ili servere onemoguće legitimnim korisnicima tako da zaguše servere sa takozvanim botovima (*eng. bots*). Bot je softver isprogramiran da izvršava određen zadatak, uz to je automatiziran što znači da ne treba biti pokrenut od strane čovjeka. Botovi obično izvršavaju repetitivne zadatke i pokušavaju imitirati ljudsko ponašanje. *DDos* napad zahtijeva od napadača da ima pristup mreži uređaja kako bi izvršio napad. Na *Slika 4* ispod prikazan je model standardnog *Ddos* napada.



Slika 4 Standardni DDos napad , (Cusack, Lutui, & Khaleghparast, 2016)

Zaražena računala postaju botovi, time napadač ima mrežu zaraženih računala odnosno *„botnet“*. Jednom kada je *botnet* mreža uspostavljena napadač može usmjeriti botove na IP adresu mete te će time oni potencijalno usporiti ili potpuno onemogućiti mrežu. *Slika 5* ispod prikazuje kako napadač može koristiti Wi-fi da kreira *botnet* te pokrene napad putem mreže telefona. Zlonamjerne mobilne aplikacije koje korisnik preuzme mogu se koristiti i koriste se kako bi se dobila kontrola nad uređajem koji bi bio jedan od mnogih u mreži botova pomoću kojih se napad vrši kao što je i prikazano na *Slika 5*. Kod ovakve vrste napada možete biti žrtva ali i nesvjesno sudjelovati. *DDos* napadi čine puno direktne štete jer blokiraju promet kompanijama koje ciljaju te to rezultira u smanjenom dohotku te kompanije kao i gubitak povjerenja klijenta (Cusack, Lutui, & Khaleghparast, 2016).

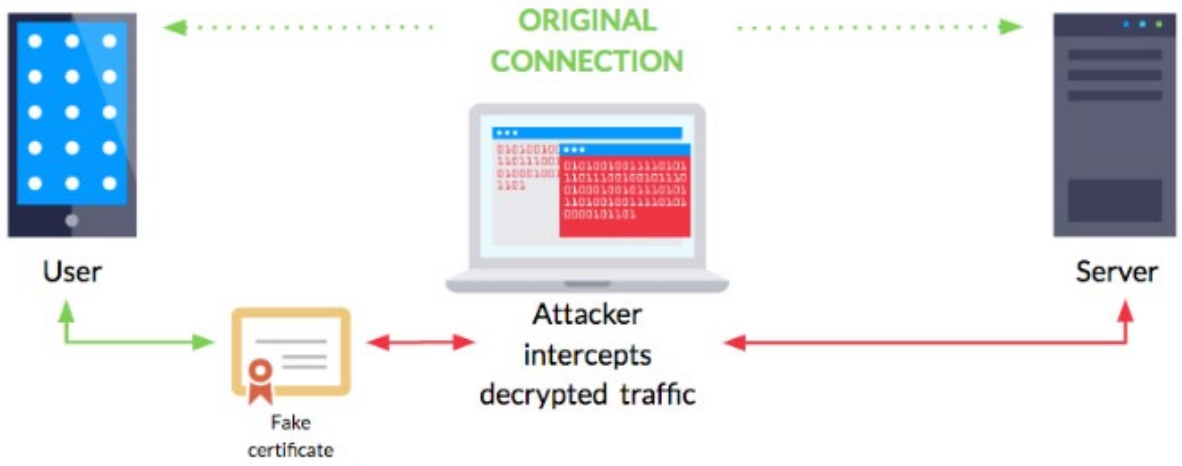


Slika 5 DDos napad putem mreže pametnih telefona , (Cusack, Lutui, & Khaleghparast, 2016).

Postoje određene protumjere za *„DdoS“* napade koje mogu biti korisne ali ne i potpuno spriječiti napad a jedna od njih je implementacija *IPS* (*eng. Intrusion Prevention System*) sustava koji nadzire i prati mrežu za bilo kakve ranjivosti no njegove mogućnosti i efikasnost je ograničena. Protiv ovih napada teško se braniti zbog načina na koji se distribuira (Douligeris & Mitrokotsa, 2003).

„Man in the middle“

Na pametnim telefonima ovakvi napadi se najčešće dešavaju kada aplikacija komunicira sa serverom te napadač neopaženo presretne komunikaciju kako bi pratio radnje i osjetljive podatke korisnika kao što su korisnička imena i lozinke. Napadi ne moraju nužno završiti samo na prisluškivanju podataka već napadač može promijeniti sadržaj koji se šalje sa servera. Mnogo je načina na koje napadač može presresti mrežnu komunikaciju. Jedan od češćih načina je putem nesigurne i nepouzdane Wi-fi veze na koju se korisnici spajaju misleći da je sigurna. Uz karakteristiku presretanja mrežnog prometa bitna karakteristika je i manjak autentifikacije. Internet se oslanja na HTTP i HTTPS protokole za mrežnu komunikaciju. HTTPS je sigurniji i preporučeno je koristiti taj protokol zbog kriptografskih metoda zaštite podataka koji HTTP nema. Na tržištu još uvijek ima aplikacija koje koriste HTTP protokol. Na svakih 5 aplikacija za Android uređaje jedna koristi HTTP protokol te za iOS uređaje jedna od sedam (Moonsamy & Batten, 2014.) .



Slika 6 „Man in the middle“ napad, https://www.nowsecure.com/blog/2019/11/20/the-analysts-guide-to-mitm-issues-in-mobile-apps/

Detektiranje *„Man in the middle“* napada može biti teško bez odgovarajuće pripreme te ukoliko konstantno ne pretražujete da li je vaš internet presretan, ovakvi napadi mogu vrlo lako proći neopaženo. Najbolje mjere zaštite protiv ovakvih napada su imati jaku *WEP/WAP* enkripciju na pristupnoj točki. Slaba enkripcija može omogućiti napadaču da na takozvani *„brute force“* način uđe na mrežu te pokrene napad. Još jedna dobra opcija je *VPN* (*eng. Virtual private network*) pomoću koje se kreira sigurna okolina ukoliko se

barata osjetljivim informacijama. *VPN* koristi enkripciju baziranu na ključevima za sigurnu komunikaciju. Na taj način čak iako napadač uspjeti presresti mrežni promet, neće moći dešifrirati podatke unutar *VPN* mreže. (Varonis, 2020, https://www.varonis.com/blog/man-in-the-middle-attack/)

„IMSI Catcher"

Napad korištenjem takozvanog *IMSI Catcher-a* usmjeren je na globalni sustav za mobilne komunikacije (*eng. Global system for Mobile Communicatons*). Prvo je potrebno objasniti par pojmova koji su potrebni za razumijevanje ove vrste napada (Threat Lab, 2019) :

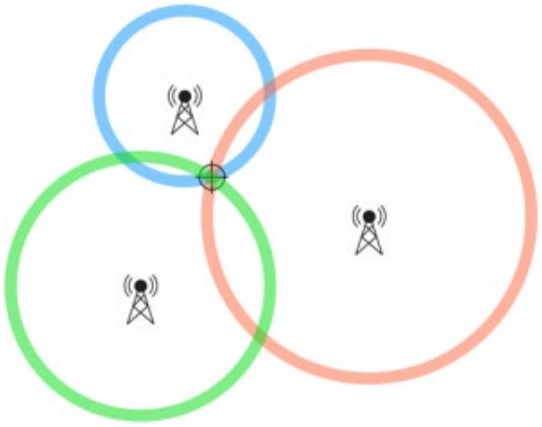
- *IMSI (eng. International Mobile Subscriber Identity)* – Jedinstveni ID povezan na *SIM* karticu uređaja te jedan od potrebnih podataka kojim se autentificira i spaja korisnika na mrežu.
- *TMSI (eng. Temporary Mobile Subscriber Identifier)* – Prilikom povezivanja na mrežu, mreža će pitati korisnika za *IMSI* te će mu dodijeliti *TMSI* dok je na mreži. Cilj ovoga je da se oteža bilo kome tko bi mogao prisluškivati podatke da ih poveže sa određenim korisnikom.
- *IMEI (eng. International Mobile Equipment Identity)* – Jedinstveni ID mobitela.

„IMSI Cather" ili *Css (eng. Cell site simulator)* je radio uređaj koji predstavlja lažnu baznu stanicu te omogućava prisluškivanje podataka korisnika koji su u dometu *Css*-a aktivno se ubacujući u mrežu gdje se predstavlja kao lažna bazna stanica ostalim mobilnim uređajima. Važno je razumijeti na koji se način *Css* spaja sa mobitelima koji koriste *LTE (eng. Long term evolution)* standard odnosno *4g* mrežu te kako zaobilazi sigurnosne protokole koji štite *GSM.* U *GSM* mreži , mobiteli uvijek traže baznu stanicu sa najvećim signalom na koji će se spojiti. Jedna od tehnika pomoću kojih napadač može preusmjeriti mobitel da se spoji na *Css* je da se zamaskira u baznu stanicu primjerice postavljajući se na istu frekvenciju i odašiljajući jaču snagu čekajući dok se mobitel ne spoji. Ipak postoji brži način te se oslanja na to da *LTE* standard ima pravilo da kada mobitel pronađe baznu stanicu sa frekvencijom koja ima veći prioritet nego ona na kojoj je trenutno spojen, mora se prebaciti na tu baznu stanicu bez obzira na jačinu signala. Kako bi se

doznalo frekvenciju sa najvećim prioritetom u području sve što je potrebno je izvući podatke iz ne kriptiranih konfiguracijskih poruka baznih stanica koje prema (Shaik, Borgaonkar, Niemi, & Seifert, 2017) svatko može nadzirat. Koristeći ove tehnike, napadač može natjerati mobitel da se spoji na *Css* te koji otkriva napadaču *IMSI* te omogućuje sljedeće napade (Threat Lab, 2019):

1. *Lociranje mobitela*
2. *Degradiranje protokola*

Lociranjem mobitela napadač dobiva točne ili približne GPS koordinate. U ovom primjeru pretpostavljamo da je napadač uspio namamiti mobitel da se spoji na *Css* koristeći se tehnikama navedenih u odlomku iznad. Nakon spajanja napadač kreira naredbu *„RRC Connection Reconfiguration"* koja sadrži ID od barem 3 najbliže bazne stanice te njihove frekvencije koju šalje na žrtvin mobitel kao što je i prikazano na *Slika 7*. Inače se ova naredba koristi za modificiranje već postojeće konekcije za baznu stanicu no napadača zanima samo prvobitni odgovor žrtvinog mobitela na poslanu naredbu. Taj odgovor sadrži jačine signala prethodno navedenih baznih stanica pomoću kojih se može dobiti lokacija mobitela triangulacijom. Degradiranje protokola je moguće i omogućava da se žrtvin mobitel degradira na manje sigurne protokole na koje se mogu pokrenuti jači napadi. Kao i u prethodnom napadu, pretpostavka je da se žrtva uspješno spojila na *Css*. Nakon spajanja mobitel šalje *TAU* (*eng. Tracking Area Update Request*). Ovom porukom mobitel obavještava baznu stanicu na koju se spojio o svojoj lokaciji kako bi mreža preusmjeravala promet brže. *Css* na tu poruku odgovara sa *„TAU Reject"* porukom koju konfigurira da vraća odgovor *„LTE services not allowed"*. Nakon zaprimanja poruke mobitel briše sve informacije o mreži na koju je prethodno poruci bio spojen te postavlja svoju SIM karticu u stanje koje je neodgovarajuće za *LTE* standard te traži 3G i GSM mreže na koje će se spojiti te u tom stanju ostaje sve dok se mobitel ne resetira (Threat Lab, 2019).

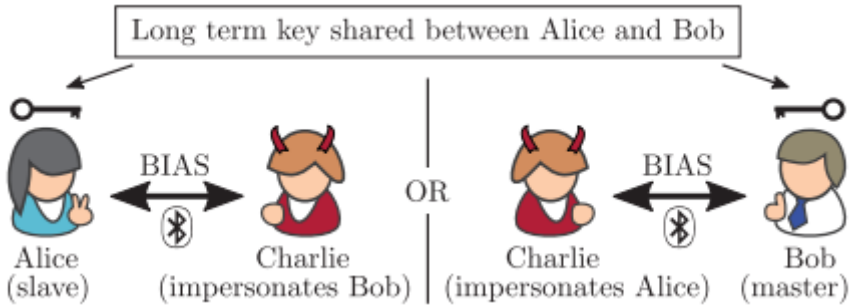


Slika 7 Triangulacija putem baznih stanica

4.3 Bluetooth napadi

„BIAS“

Tim znanstvenika koji su ujedno i autori rada (Antonioli, Tippenhauer, & Rasmussen, 2020) na koji se referenciram otkrili su određene ranjivosti u Bluetooth tehnologije te su izveli svoju vrstu napada koju su nazvali *BIAS* (*eng. Bluetooth impersonation attacks*) kako bi potkrijepili svoje tvrdnje. Ovim napadom potvrdili su da Bluetooth standard sadrži ranjivosti koje omogućavaju napadaču da se predstavi kao uređaj koji žrtva smatra sigurnim te uspostavi sigurnu konekciju sa žrtvinim uređajem bez potrebnog kriptiranog ključa kojeg prilikom uparivanja moraju imati oba uređaja. Ovaj napad specifično cilja autentifikacijsku fazu prilikom sigurne uspostave konekcije, točnije, napada proceduru *legacy authentication procedure* koja se koristi za uspostavu *Legacy Secure* (*eng. LSC*) konekcije te također *Secure Connection* (*eng SC*) proceduru. Napadač ne mora biti prisutan prilikom uparivanja dva uređaja te prethodno ovome napadu pretpostavlja se da je barem jednom bila uspostavljena konekcija između dva uređaja od kojih će jedan biti žrtva napada. U svojemu radu uspješno su napali 28 Bluetooth uređaj koji su proizvedeni od velikih i poznatih proizvođača uključujući Apple i Samsung. Uzmimo kao primjer dvije osobe koje su obje žrtve, osoba *„Alice“* te osoba *„Bob“* putem sigurne Bluetooth veze uspostave konekciju. Također pretpostavljamo da su *„Alice“* i *„Bob“* podijelili kriptirani ključ za uspostavljanje sigurne veze koji je nepoznat napadaču. Cilj napadača to jest u ovom slučaju osobe *„Charlie“* je da uspostavi sigurnu konekciju sa *„Bob“* pretvarajući se da je *„Alice“* ili obrnuto. (Antonioli, Tippenhauer, & Rasmussen, 2020)

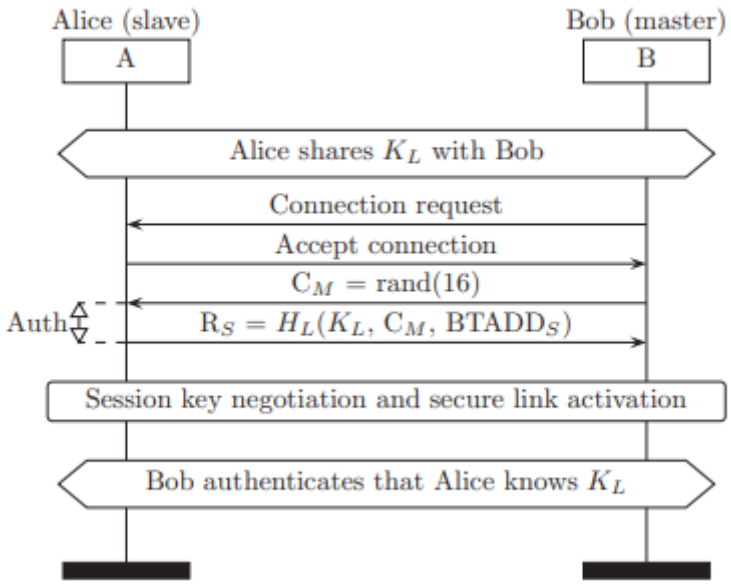


Slika 8 BIAS napad, (Antonioli, Tippenhauer, & Rasmussen, 2020)

„Bob“ i „Alice“ posjeduju kriptirani ključ koji se naziva *link key* koji su dobili prilikom uspostave prve Bluetooth konekcije. U ovom primjeru *„Bob“* je *master* te on želi uspostaviti konekciju sa *„Alice“* koja je *slave* koristeći spomenuti ključ, *„Alice“* će prihvatit zahtjev za konekcijom te se upravo prilikom uspostave konekcije provodi napad. 3 parametra koja označuju njihovu jedinstvenu vezu su *link key* te Bluetooth adrese *master* i *slave* uređaja $BTADD_A$ i $BTADD_B$. Napadač zna javne informacije ove veze kao što su Bluetooth imena, adrese,brojeve verzija protokola te mogućnosti uređaja itd. Kada se predstavlja kao *„Alice“* ili *„Bob“* , može mijenjati adrese iz $BTADD_A$ u $BTADD_B$ no ne može dokazati vlasništvo *link key* ključa koji mu nije poznat. To je temeljna pretpostavka Bluetooth autentifikacijskog procesa koja korisnicima garantira da je njihova veza sigurna, no ipak u svojem radu autori su dokazali da (Antonioli, Tippenhauer, & Rasmussen, 2020) :

- *Bluetooth uspostava konekcije nije sigurna niti kriptirana*
- *Legacy Secure Connection ne zahtijeva zajedničku autentikaciju*

U svojemu radu autori su naveli 4 podvrste napada. 2 napada prilikom *Legacy Secure Authentication* procesa gdje napadač može sebe predstaviti kao *master* ili *slave* uređaj te 2 napada na *Secure Connection* proces gde napadač također može sebe predstaviti kao *master* ili *slave* uređaj. U ovom radu prikazati ću dvije podvrste napada,jednu vrstu za svaki tip uspostave konekcije. Prva podvrsta napada je na *Legacy secure* konekciju odnosno *LSC*. Ovaj tip konekcije prati proceduru koja je prikazana na *Slika 9* ispod.

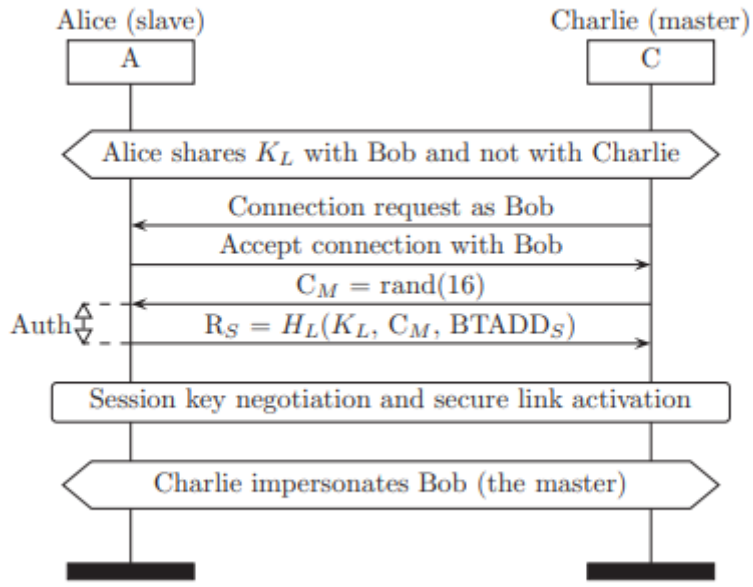


Slika 9 LSC Autentikacija, (Antonioli, Tippenhauer, & Rasmussen, 2020)

Master uređaj šalje zahtjev za uparivanjem te *slave* dijeli *link key* sa *master* uređajem. Također razmijene svoje Bluetooth adrese te nakon toga *master* generira nešto što se zove *challenge protocol* koji je na *slika 9* označen sa *„C_M"* te ga šalje *slave* uređaju koji generira *challenge response protocol* putem hash algoritma koristeći *link key*, svoju adresu *$BTADD_S$* te *C_M* nazvan *„R_S"* kojeg šalje natrag *master* uređaju koji napravi isto sa svojim parametrima. Ukoliko su vrijednosti iste *master* autentificira da je povezan sa *slave* uređajem. Problemi prilikom uspostave *LSC* konekcije su sljedeći (Antonioli, Tippenhauer, & Rasmussen, 2020):

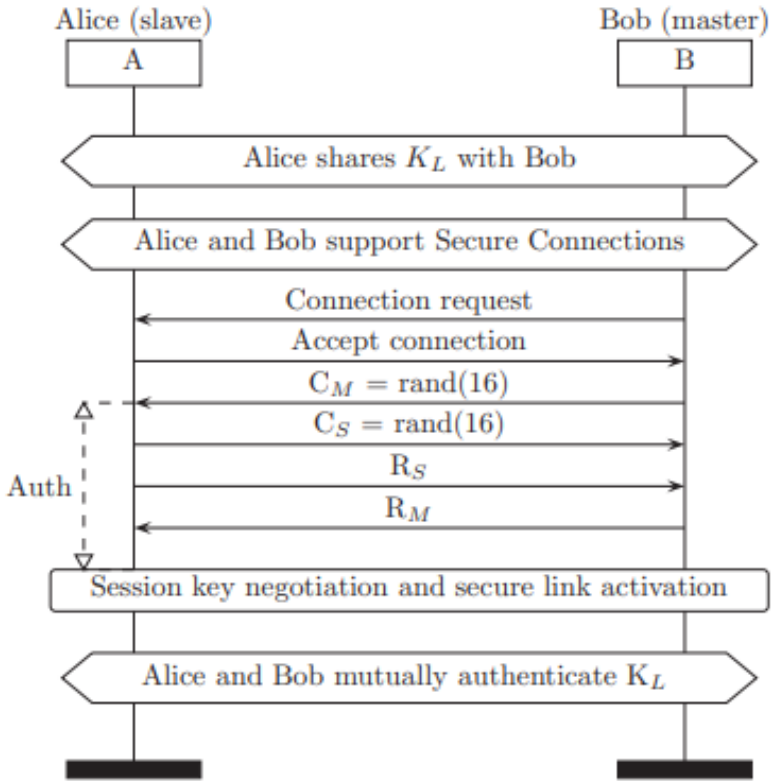
- *LSC autentifikacija je jednosmjerna (master provjerava slave uređaj) što znači da slave ne može provjeriti da li je master onaj za koga se predstavlja.*
- *Uređaj može promijeniti ulogu prije nego li procedura počne*

Upravo te slabosti su autori iskoristili kako bi proizveli vlastiti napad koji je prikazan na *slika 10* ispod.



Slika 10 BIAS LSC napad kao „master“ uređaj, (Antonioli, Tippenhauer, & Rasmussen, 2020)

Slave dijeli *link key master* uređaju misleći da je to *„Bob“*. *„Charlie“* šalje Bluetooth adresu od *„Bob“* nazad do *„Alice“* koja šalje nazad svoju adresu. Slijedi već poznata procedura autentifikacije gdje *„Charlie“* predstavlja sebe kao *„Bob“* te je u mogućnosti izvesti daljnje napade na uređaj dok *slave* uređaj ne može nikako autentificirati integritet *master* uređaja. Druga podvrsta napada odnosi se na *Secure connection* autentifikaciju koja je prikazana na *Slika 11*. Ovaj tip autentifikacije koristi dvosmjernu autentifikaciju gdje *„Bob“* i *„Alice“* razmjene svoje adrese zatim oboje kreiraju *challenge protocole* i razmjene ih te autentificiraju jedan drugog na temelju vrijednosti *challenge response protcol* vrijednost (Antonioli, Tippenhauer, & Rasmussen, 2020).

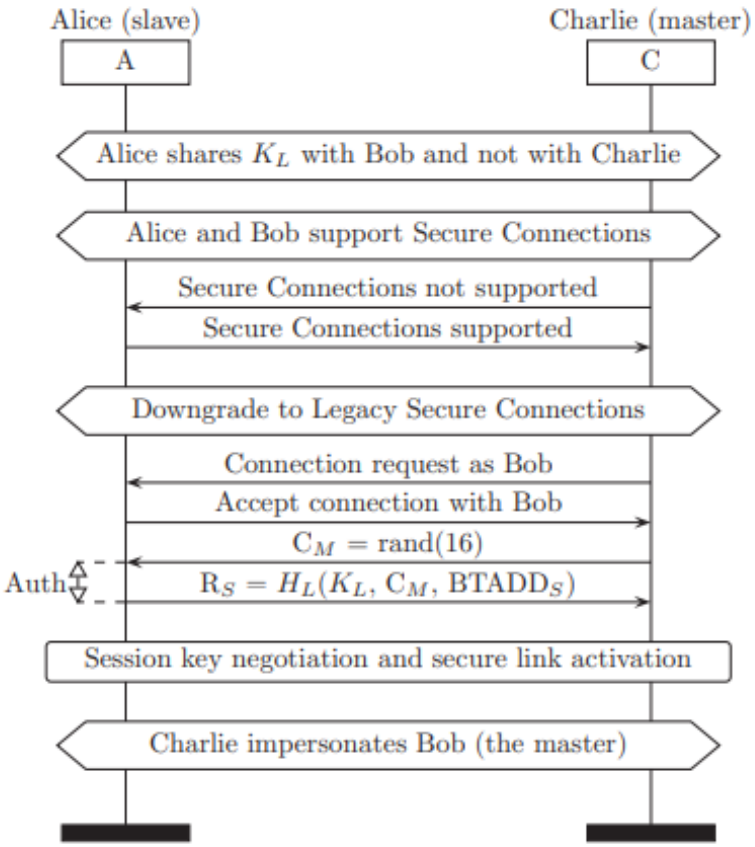


Slika 11 SC autentikacija, (Antonioli, Tippenhauer, & Rasmussen, 2020)

Iako je kod ovog tipa autentifikacija međusobna, ona još uvijek ima ranjivosti koje se mogu iskoristiti a one su (Antonioli, Tippenhauer, & Rasmussen, 2020) :

- *SC autentifikacija se može pregovarati (objašnjenje slijedi u slijedećem paragrafu).*
- *SC autentifikacija nije prisilna.*

SC autentifikacija se može pregovarati, što znači da napadač koji zna mogućnosti uređaja može namjerno promijeniti metodu autentifikacije iz *SC* u *LSC* što znaći da opet samo on ukoliko se predstavlja kao *master* koristi jednosmjernu autentifikaciju. Napad na *SC* konekciju gdje se napadač predstavlja kao *master* radi na sljedeći način prikazan na *Slika 12* **.** (Antonioli, Tippenhauer, & Rasmussen, 2020)



Slika 12 BIAS SC napad kao „master“ uređaj, (Antonioli, Tippenhauer, & Rasmussen, 2020)

„Alice“ dijeli svoj *link key* sa *„Bob“* misleći da je veza sigurna. *„Charlie“* dijeli adresu od *„Bob“* te laže o mogućnostima uređaja da njegov uređaj ne podržava *SC* konekciju već samo *LSC*. Iako *slave* uređaj podržava *SC* konekciju on se mora spustiti na razinu koju zahtijeva *master* a to je *LSC* te u tom slučaju gubi mogućnost da autentificira *master* uređaj i time je integritet narušen i napadač je uspio u svojoj namjeri da se predstavi kao *„Bob“.*

Slijedi slika sa popisom uređaja na koje je uspješno izveden barem jedan od 4 podvrste *BIAS* napada. (Antonioli, Tippenhauer, & Rasmussen, 2020)

		LSC		SC	
Chip	Device(s)	MI	SI	MI	SI
Bluetooth v5.0					
Apple 339S00397	iPhone 8	●	●	●	●
CYW20819	CYW920819EVB-02	●	●	●	●
Intel 9560	ThinkPad L390	●	●	●	●
Snapdragon 630	Nokia 7	●	●	●	●
Snapdragon 636	Nokia X6	●	●	●	●
Snapdragon 835	Pixel 2	●	●	●	●
Snapdragon 845	Pixel 3, OnePlus 6	●	●	●	●
Bluetooth v4.2					
Apple 339S00056	MacBookPro 2017	●	●	●	●
Apple 339S00199	iPhone 7plus	●	●	●	●
Apple 339S00448	iPad 2018	●	●	●	●
CSR 11393	Sennheiser PXC 550	●	●	-	-
Exynos 7570	Galaxy J3 2017	●	●	-	-
Intel 7265	ThinkPad X1 3rd	●	●	-	-
Intel 8260	HP ProBook 430 G3	●	●	-	-
Bluetooth v4.1					
CYW4334	iPhone 5s	●	●	-	-
CYW4339	Nexus 5, iPhone 6	●	●	-	-
CYW43438	RPi 3B+	●	●	●	●
Snapdragon 210	LG K4	●	●	●	●
Snapdragon 410	Motorola G3, Galaxy J5	●	●	●	●
Bluetooth v≤ 4.0					
BCM20730	ThinkPad 41U5008	●	○	-	-
BCM4329B1	iPad MC349LL	●	●	-	-
CSR 6530	PLT BB903+	●	●	-	-
CSR 8648	Philips SHB7250	●	●	-	-
Exynos 3470	Galaxy S5 mini	●	●	-	-
Exynos 3475	Galaxy J3 2016	●	●	-	-
Intel 1280	Lenovo U430	●	●	-	-
Intel 6205	ThinkPad X230	●	●	-	-
Snapdragon 200	Lumia 530	●	●	-	-

Slika 13 Ranjivih uređaji, (Antonioli, Tippenhauer, & Rasmussen, 2020)

„ Bluesnarf "

Bluetooth napadi bili su aktivni u samim počecima pametnih telefona 2004. godine dok su danas gotovo pa nepostojeći i mane u Bluetooth tehnologiji koje postoje danas ne često se iskorištavaju no ipak potrebno je spomenuti jedan tip napada , *„Bluesnarf"* napad. Ovaj napad omogućava napadaču da pristupi podacima unutar mobitela povezivanjem žrtve spajajući se putem Bluetooth veze. Prema (Wong, 2005) kako bi se izvršio ovaj napad potrebno je iskoristiti sigurnosne mane protokola za razmjenu objekata, odnosno *OBEX* (*eng. Object exchange protocol*). Napadač se mora spojiti na

uslugu koja se zove *OBEX Push Service* koja ne zahtijeva autentifikaciju i koja je optimizirana za jednostavnu razmjenu podataka. Jednom kada je *OBEX* protokol kompromitiran, napadač je u mogućnosti sinkronizirati svoj sustav sa žrtvinim uređajem. Nakon toga se poziva GET metoda na *filenames* to jest ime podatka ili mape primjerice *telecom/cal.vcs* za kalendar uređaja ili *"telecom/pb.vcf"* za popis kontakata na uređaju. Ako je *firmware* na uređaju loše implementiran,napadač može dobiti pristup podacima čija su imena poznata napadaču ili koja može na sreću pogoditi. Neki od ostalih napada slični ovome putem Bluetooth tehnologije su (Becker, 2007) :

- *BlueBug*
- *Bluejack*

5. Zlonamjerni softver

U ovom poglavlju objasnit ću na koji način funkcionira, njegovu definiciju te vrste zlonamjernog softvera namijenjenog pametnim telefonima , na engleskom jeziku poznatiji i pod nazivom *mobile malware*. Nastao od spoja dvije riječi *malicious* i *software* , *malware* je zlonamjerni softver koji cilja i iskorištava mane u operacijskom sustavu kako bi naštetio korisniku na bilo koji način. Kada se radi o pametnim telefonima posebno se izdvaja u kategoriju *mobile malware* koja cilja specefično operacijske sustave mobilnih uređaja. U određenoj literaturi *mobile malware* je ekvivalent zlonamjernim aplikacijama (*eng. Mobile malicious application*). Prema (Ahvanooey, Li, Rabbani, & Rajput, 2017),zlonamjerna aplikacija je skriveni zlonamjerni softver koji se nalazi i izvodi u pozadini napadnutog uređaja.

5.1 Faze funkcioniranja

Nakon pregleda glavnih vrsta zlonamjernog softvera biti će objašnjene glavne faze to jest korake koje *maleware* prođe kako bi zarazio ciljani uređaj. Prema (Becher, 2009) zlonamjerni softver ima 3 glavne faze :

1. Zaraziti uređaj (*eng. infecting*)
2. Širenje na ostale uređaje (*eng. spreading*)
3. Zlonamjerna funkcionalnost (*eng. malicious functionality*)

5.1.1 Zaraza uređaja

Zaraza je prvi korak gdje se *malware* inflitrira u uređaj. Ovaj korak može se dodatno kategorizirati prema razini interakcije koja je potrebna da korisnik aktivirao *malware*. (Becher, 2009) je podijelio razine interakcije na 4 vrste (Becher, 2009) :

1. *Explicit permission* – Direktno se pita korisnika da li želi zaraziti svoj uređaj jasno dajući doznanja korisniku o namjerama. Ovo je tipičan primjer takozvanog *proof-of* koncepta koji se koristi u testne ili edukacijske svrhe.
2. *Implicit permisson* – Ova kategorija predstavlja standardni niz pitanja prilikom instalacije nepotpisane ili nepoznate aplikacije. Korisnika se navodi na instaliranje aplikacije na razne načine koristeći društveni inžinjering (*eng. Social engineering*) o kojemu više u sljedećim poglavljima.

3. *Common interaction* – Treća kategorija je akcija koja je česta prilikom korištenja mobitela gdje se uređaj zarazi prilikom otvaranja SMS poruke ili elektroničke pošte.
4. *No interaction* – Najgora vrsta *malware*-a prenosi se ovim putem,bez interakcije sa korisnikom.

5.1.2 Širenje zlonamjernog softvera na ostale uređaje

Neki napadi ciljaju posebno jednu specifičnu žrtvu pa cilj napada nije širenje *malware*-a no kad to nije slučaj *malware* se želi proširiti na što više uređaja kako bi napravio što veću štetu. Najčešći način na koji se *malware* može proširiti na više uređaja je putem zlonamjernih aplikacija koje korisnik preuzima od nepouzdanog izvora, iako se zna dogoditi da zlonamjerna aplikacija pronađe način da dospije i na tržište aplikacija koje se inače smatra sigurnim. Sumnjivi email-ovi ili SMS poruke su poslije zlonamjernih aplikacija možda i najčešći način kako se uređaj zarazi. Primjerice,korisnik dobije email ili SMS da je osvojio određenu nagradu,klikne na poveznicu te bude odveden na lažnu stranicu,korisniku izgleda da se ništa ne dešava te izađe sa web stranice no u međuvremenu *malware* je preuzet na uređaj bez korisnikova znanja. Jedan od načina također može biti žičanim putem kada se pametni telefon spoji žičano na osobno računalo radi kreiranja sigurnosnih kopija,ažuriranja ili sinkroniziranja podataka. O ovakvoj vrsti mogućih napada više u posebnom poglavlju (Becher, 2009)**.**

5.1.3 Zlonamjerna funkcionalnost

Jednom kada se nađe na uređaju,*malware* može raditi ono za što je kreiran. Prema (Becher, 2009) zlonamjerna funkcionalnost je podijeljena na novčanu štetu (*eng. monetary damage*) , podatkovnu štetu (*eng. data damage*) i skrivenu štetu (*eng. hidden damage*).

5.2 Pregled vrsta zlonamjernog softvera

Slijedeće su vrste zlonamjernih softvera (Khan & Kumar, 2019) :

- *Virus*
- *Spyware*

- *Trojan*
- *Ransomware*

Virus

Virus kod pametnih telefona je dio koda koji je u mogućnosti replicirati samog sebe kako bi zarazio ostale aplikacije ili datoteke na način da se replicirana verzija virusa doda u datoteku ili aplikaciju (Khan & Kumar, 2019)**.** Prvi virus koji je ciljao pametne telefone razvijen je 2004. godine i nazvan je *Cabir*. Glavna svrha ovog virusa bio je da se educira developere kako se zlonamjeran kod može napisati za specifičan operacijski sustav, u ovom slučaju *Symbian OS*. *Cabir* se prenosio kao .SIS datoteka ,instalacijska datoteka *Symbian OS* sustava,putem Bluetooth tehnologije no onda se zamaskirao kao dio sigurnosnog mehanizma. Nakon instalacije datoteke uređaj je bio zaražen. Svaki put kada bi se uređaj pokrenuo na ekranu bi se ispisala poruka *Caribe*. Prenosio se sa mobitela na mobitel tako što je automatski tražio ostale uređaje koji su imali uključen Bluetooth te slao svoji kopiju na taj uređaj. Ovaj virus nije bio opasan za korisnika budući da mu je svrha bila educirati developere i osvijestiti korisnike (Kaspersky, 2020, https://www.kaspersky.com/blog/cabir-10/5107/) **,** (Heera, 2008)**.**

Spyware

Spyware je zlonamjeran softver koji prati žrtvin uređaj kako bi pratio aktivnosti korisnika kao što su lokacija,pozivi,poruke i elektroničku poštu. Možda najpoznatiji *Spyware* koji je do danas otkriven naziva se *„Pegasus“*. Razvije od izraelske kompanije pod nazivom *NSO Group*. Širi se najčešće putem SMS ili email poruka te ukoliko dobije pristup uređaju, ima potpunu kontrolu i nadzor nad njim. Nakon skeniranja uređaja, *„Pegasus“* instalira potrebne module pomoću kojih nadzire poruke,elektroničku poštu,prisluškuje pozive,povijest pretraživanja itd. Zanimljiva je činjenica da *„Pegasus“* sam sebe uništi ukoliko nije uspio uspostaviti vezu sa takozvanim *Command and Control* serverom. U početku je ovaj *Spyware* bio namijenjen samo za iOS uređaje no 2017. razvijena je i Android verzija pod nazivom *„Chrysaor“*. Ova verzija vrlo je slična onoj koja cilja iOS uređaje po svojim sposobnostima no razlikuje se po načinu na koji prodire u uređaj (Khan & Kumar, 2019) , (Lookout, 2016) , (Lookout, 2017) , (Ahvanooey, Li, Rabbani, & Rajput, 2017).

Trojan

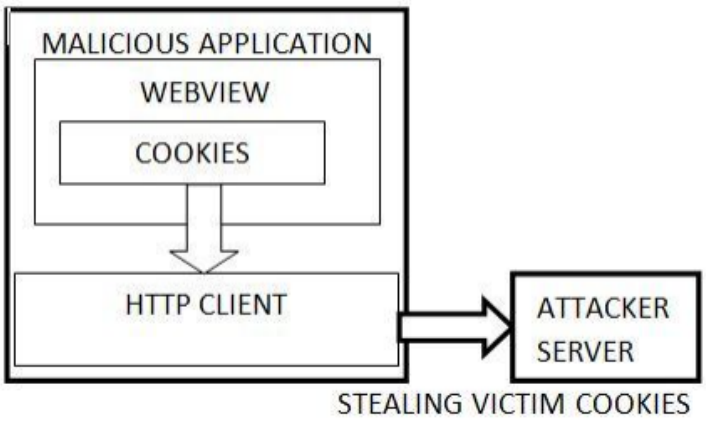
Ovakav tip zlonamjernog softvera koji omogućava pristup osjetljivim interakcijama korisnika kao što su kupovina putem aplikacija,mobilno bankarstvo itd tako što se skriva iza aplikacija za koje korisnik smatra da su sigurne i koje nisu dovoljno dobro provjerene (Ahvanooey, Li, Rabbani, & Rajput, 2017).

Ransomware

Ransomware je tip zlonamjernog softvera koji korisniku onemogučava da pristupi podacima na svom uređaju što spada u kategoriju *Lock-screen ranswomware* ili u potpunosti blokira korisničko sučelje uređaja što je *Crypto-ransomware* (Lipovský, Štefanko, & Braniša, 2016) .

5.3 WebView

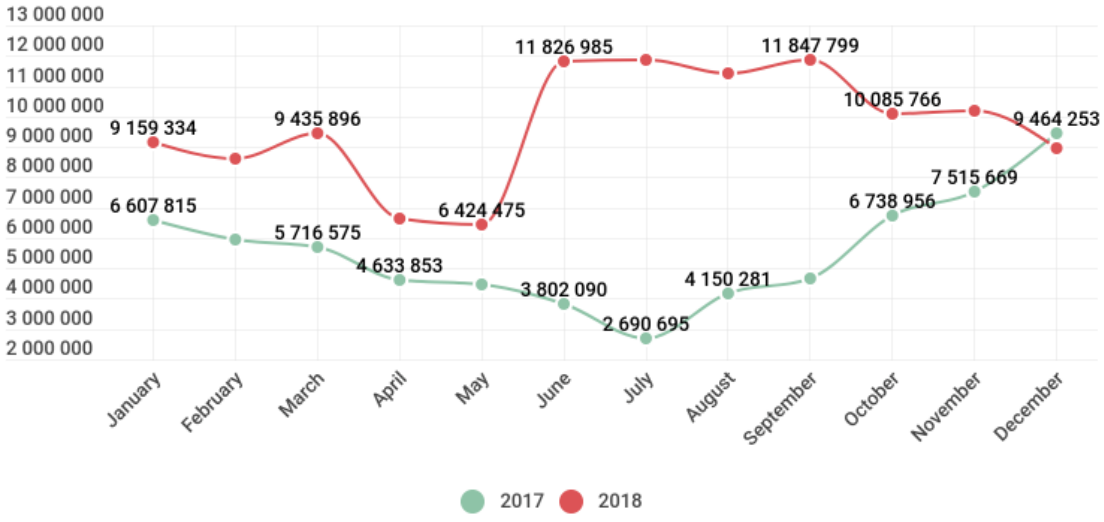
U okviru ovog poglavlja opisuje se sistemska komponenta *WebView* koja je namijenjena za mobilne operacijske sustave koja dozvoljava aplikacijama da prikazuju sadržaj sa web stranice direktno unutar aplikacije te dozvoljava aplikaciji interakciju sa ostalim web stranicama ili web aplikacijama. Kao sigurnosnu mjeru *WebView* u sebi ima ugrađen već spominjani *sandbox* mehanizam što znači da se Javascript kod može izvršavati samo u izoliranom okruženju i on nema mogućnosti pristupiti resursima uređaja. Napadi uzrokovani sigurnosnim manama *WebView* nazivaju se *Javascript injection* napadi. *Javascript injection* napad se oslanja na mehanizam *WebView* komponente da Javascript kod *„prizove“* Java kod od aplikacije.Kako bi se ovaj napad izvršio napadač mora kreirati zlonamjerni program za specifičnu aplikaciju koji potom pokrene na web stranici. Iako je *WebView* zaštićen *sandbox* mehanizmom on je bespomoćan ukoliko korisnik preuzme određeni sadržaj sa te web stranice. Jednu od ključnih uloga ima *WebView*-ov API *„loadUrl“.* Ovaj API se koristi da se učita specifični URL u *WebView* komponenti. Taj specifični URL može biti kreiran i od strane napadača te može dobiti pristup web stranici i njenim podacima kao što su *cookies* (Zhang, 2018).



Slika 14 Javascript Injection" napad(Zhang, 2018)

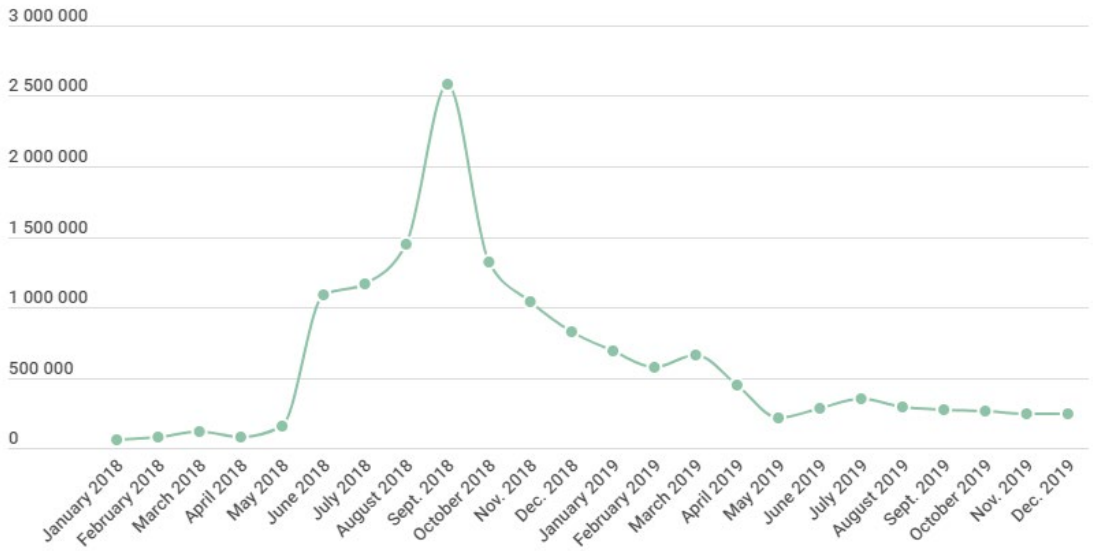
5.4 Statistika

Slijedeće podatke prikupila je međunarodna agencija za cyber sigurnost *„Kaspersky Lab"* nakon provedenog istraživanja. *Kaspersky* tehnologije u 2019. godini zabilježile su 3,503,952 zlonamjerna instalacijska paketa. Detektirano je najviše novih programa koji ciljaju bankovne mobilne transakcije,njih 69,777 te novih *ransomware* programa 68,362. Slika ispod prikazuje broj instalacijskih paketa u 2017. i 2018. godini koji su sadržavali zlonamjeran program u sebi. Iz grafa se jasno može vidjeti da je broj takvih paketa za samo jednu godinu porasao i konstantno bio veći sve do kraja godine u prosincu gdje je broj ipak bio manji nego prethodne godine. (Kaspersky, 2020 , https://securelist.com/mobile-malware-evolution-2019/96280/)



Slika 15 Zlonamjerni instalacijski paketi, V.Chebesyev, https://securelist.com/mobile-malware-evolution-2019/96280/ , 14.09.2020.

Slijedeća slika prikazuje graf na kojemu je prikazan broj zlonamjernog softvera koji cilja bankovne mobilne transakcije u periodu od 2018. do 2019. godine. Prema *Kaspersky Lab*-u broj takvih programa se prepolovio od 2017. godine no njhovi napadi su se povečali zbog smanjene aktivnosti ostalih vrsta zlonamjernih programa. (Kaspersky, 2020 , https://securelist.com/mobile-malware-evolution-2019/96280/)



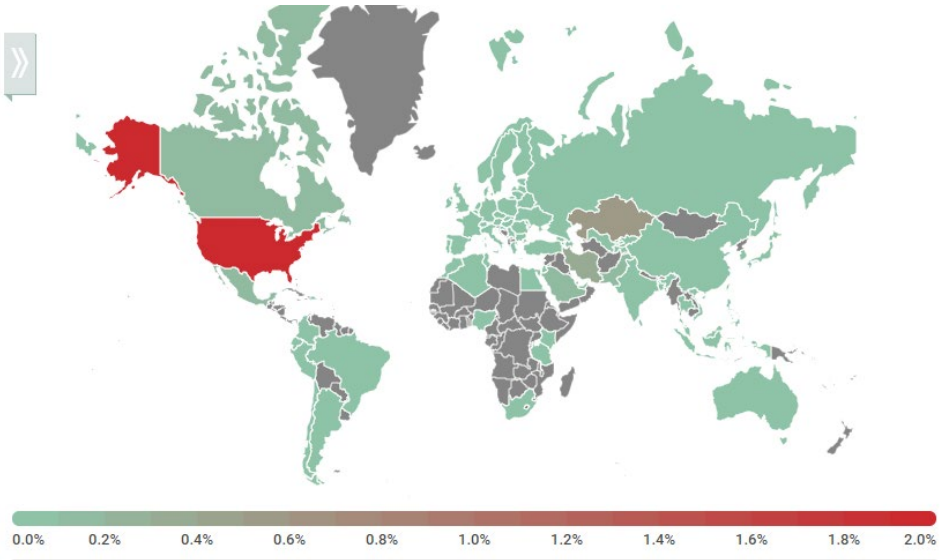
Slika 16 Instalacija „Banking Trojan" programa, V.Chebesyev, https://securelist.com/mobile-malware-evolution-2019/96280/ , 14.09.2020.

Najviše pojava takozvanog *„Banking Trojan"* programa ima u Rusiji 3 godine zaredom od 2017. do 2019. godine gdje je zadnju godinu postotak iznosio 0.72%. Razne vrste *Trojan malware*-a su fokusirane na krađu podataka i bankovnih informacija specifično od ruskih aplikacija. Slijede Južnoafrička republika sa 0.66% te Australia sa 0.59%.

	Country	%*
1	Russia	0.72
2	South Africa	0.66
3	Australia	0.59
4	Spain	0.29
5	Tajikistan	0.21
6	Turkey	0.20
7	USA	0.18
8	Italy	0.17
9	Ukraine	0.17
10	Armenia	0.16

Slika 17 Postotak „Banking Trojan“ programa u 2019. Godini, V.Chebesyev, https://securelist.com/mobile-malware-evolution-2019/96280/ , 14.09.2020.

Kada se radi o ucjenama korisnika putem već navedenog *ransomware*-a Sjedinjene Američke države su na prvom mjestu od 2017. do 2019. godine gdje u zadnjoj godini je postotak iznosio 2.03% što ih je stavilo na vrh ljestvice po napadima ove vrste.



Slika 18 Postotak ransomware napada u 2019. Godini, V.Chebesyev, https://securelist.com/mobile-malware-evolution-2019/96280/ , 14.09.2020.

5.5 Vrste detekcije

Detekcija zlonamjerna aktivnosti je tema koja se još raspravlja i nije se postignut zajednički dogovor o klasifikaciji metoda detektiranja zlonamjernog softvera. Prema (Kouliaridis, Barmpatsalou, Kambourakis, & Chen, 2020) metode detekcije dijele se na dvije glavne koje se kasnije dijele u tri podvrste :

1. *Signature based detection*
    1.1. *Static*
    1.2. *Behaviour*
    1.3. *Hybrod*
2. *Anomaly based detection*
    2.1. *Static*
    2.2. *Dynamic*
    2.3. *Hybrid*

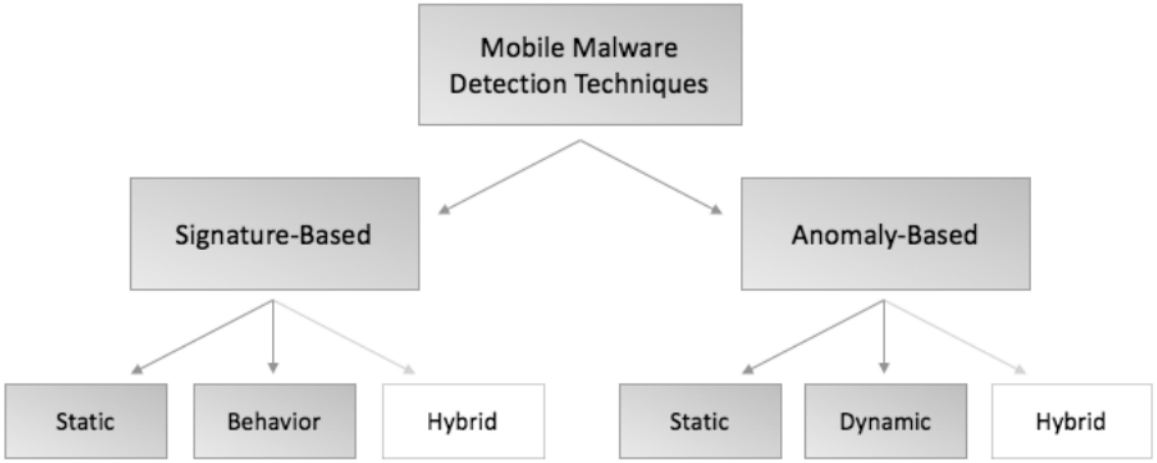
„Signature based detection“

Prva podvrsta ove metode je *„Static signature based detection“* koja koristi bazu podataka prikupljenih uzoraka i sekvenci bitova već poznatih zlonamjernih softvera te ih uspoređuje sa sumnjivim dijelovima koda u sustavu kako bi odredila da li je on zlonamjeran. *„Behaviour signature based detection“* se razlikuje po tome što on prikuplja podatke o potencijalno zlonamjernom kodu dok je on pokrenut i dok se izvodi. Zadnja podvrsta je *„Hybrid signature based detection“* te se ona sastoji od kombiniranje dvije navedene podvrste te funkcionira na *host* i *cloud* sustavu gdje se koriste razni servisi za praćenje prometa te napada na privatnost (Kouliaridis, Barmpatsalou, Kambourakis, & Chen, 2020) .

„Anomaly based detection“

Ova vrsta detekcije koristi manje ograničen pristup na način da se nezaraženi uređaj promatra na određeno vrijeme te se dobiveni podaci sa promatranja uređaja koriste kao referentna točka te svatko odstupanje od te točke se smatra zlonamjernom aktivnošću. *„Static anomaly based detection“* metoda ne zahtijeva da se zlonamjerni kod izvršava već je zadaća ove metode provjeriti kod za zlonamjerne funkcionalnosti i potencijalno ponašanje te na temelju toga utvrditi da li je opasnost stvarna. Ova metoda može rezultirati nečim što se naziva *„False positive“* detekcije, određeni programi, aplikacije ili dijelovi koda koji nisu zlonamjerni se označavaju kao zlonamjerni. Kod *„Dynamic anomaly based detection“* metode, detekcija se vrši prilikom pokretanja potencijalno zlonamjernog koda te ima mogućnost otkrivanja najnovijih prijetnji. Zadnja podvrsta *„Habryd anomaly based detection“* je metoda koja spaja prethodne dvije. Ovakva vrsta detekcije može se naći u servisu *„ScanMe mobile“* koji je također *host* i *cloud* servis koji

dopušta korisnicima da na *cloud* servisu ili u kontroliranom okruženju na mobitelu skeniraju aplikaciju koju žele instalirati (Kouliaridis, Barmpatsalou, Kambourakis, & Chen, 2020).



Slika 19 Metode detekcije, (Kouliaridis, Barmpatsalou, Kambourakis, & Chen, 2020)

6. Hardverske ranjivosti i napadi

6.1 Napadi putem USB konekcije

Danas svi pametni telefoni koriste USB (*eng. Universal Series Bus*) konekciju za punjenje uređaja te za prebacivanje i sinkronizaciju podataka između pametnih telefona međusobno ili pametnih telefona i osobnih računala. Štoviše USB konekcija omogućava svim developerima koji žele da se uključe u razvoj operacijskog sustava uređaja na koji su spojeni. Glavna svrha ovih napada je iskorištavanje ranjivosti na mobilnim operacijskim sustavima koje omogućava USB konekcija. To nalaže da napadač mora biti u posjedu žrtvinog uređaja ili jednog od uređaja na koji je žrtvin mobitel povezan (Pereira, Correia, & Brandão, 2017.)

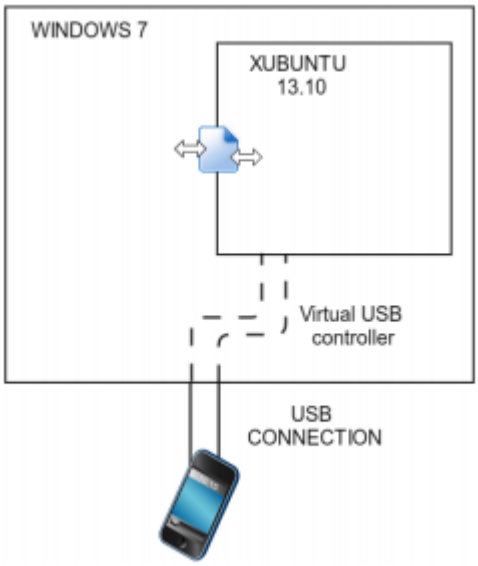
USB konekcija ne podržava mrežnu klasu, audio klasu ili ostale klase kao *MSC* (*eng. USB mass storage class*). Na Android uređajima *ADB* (*eng. Android Debugging Bridge*) je izvorno onemogućen te ukoliko je mobitel spojen na osobno računalo, to računalo gleda na mobitel samo kao masovnu pohranu bez ikakvih funkcionalnosti. Jednom kada je *ADB* omogućen unutar postavki, uređaj se može kontrolirati putem *ADB* alata koji je dostupan unutar *Android SDK* (*eng. Software development kit*). *ADB* je alat koji omogućava komunikaciju osobnog računala i mobitela putem USB konekcije te omogućava nekoliko akcija kao što su dohvat podataka sa pametnog telefona ili ubacivanje podataka na isti, instalacija ili *debugging* aplikacija. To je *„klijent – server“* program sa tri komponente (Amarante & Barros, 2017):

1. *Klijent* – Šalje naredbe te je pokrenut na računalu.
2. *„Daemon“* – Izvršava naredbe na uređaju i pokreće proces u pozadini uređaja.
3. *Server* – Upravlja komunikacijom između klijenta i *„Daemon*“ komponente. Server se izvršava u pozadini na računalu.

Arhitektura napada

Program kojim će napadač izvesti napad već mora biti spreman na računalu na koje će se mobitel spojiti. Također program mora biti brz, potpuno automatiziran te u mogućnosti da izvršava operacije na više razina operacijskog sustava. U radu (Pereira, Correia, & Brandão, 2017.) autori su kreirali sljedeći scenarijo. Iskorištene su funkcionalnosti dva opereacijska sustava, Windows i Linux. Windows je glavni

operacijski sustav kojeg nazivaju *host* dok je Linux to jest *Xubuntu* pokrenut na virtualnoj mašini pomoću programa *Virtual Box* i njega nazivaju *guest* te je u mogućnosti komunicirati sa *host* operacijskim sustavom. Razlog zbog čega se koristi *Xubuntu* je taj kako bi zlonamjerni program koji se izvodi sa *guest* operacijskog sustava mogao iskoristiti prednosti koje pruža Linux operacijski sustav. Nužno je da je *guest* operacijski sustav Linux a *host* Windows a ne obrnuto. Komunikacija između operacijskih sustava je neophodna kako bi *guest* koji radi većinu posla mogao reći *host* operacijskom sustavu kada i kako da napadne (Pereira, Correia, & Brandão, 2017.).



Slika 20 Arhitektura komuniciranja između „guest" i „host" OS, (Pereira, Correia, & Brandão, 2017.)

Program pokrenut na *Xubuntu* virtualnoj mašini to jest *guest* OS zadužen je za (Pereira, Correia, & Brandão, 2017.) :

- *Detektiranje spojenih USB uređaja*
- *Identifikaciju vrste uređaja*
- *Identifikaciju poznatih ranjivosti za taj uređaj*
- *Napade iskorištavanjem tih ranjivosti*
- *Komunikaciju sa „host" OS ukoliko su potrebni dodatni alati za izvršavanje napada*
- *Identifikaciju vanjskih kartica USB uređaja*

Windows OS to jest *host* zadužen je za sljedeće (Pereira, Correia, & Brandão, 2017.) :

- *Komunikaciju sa „guest" OS kako bi saznao koji uređaj označit za napad ukoliko su*

potrebni dodatni alati

- *Identifikacijom firmware-a*
- *Pružanje dodatnih alata za napad „guest" OS*

6.2 Rooting / Jailbraking

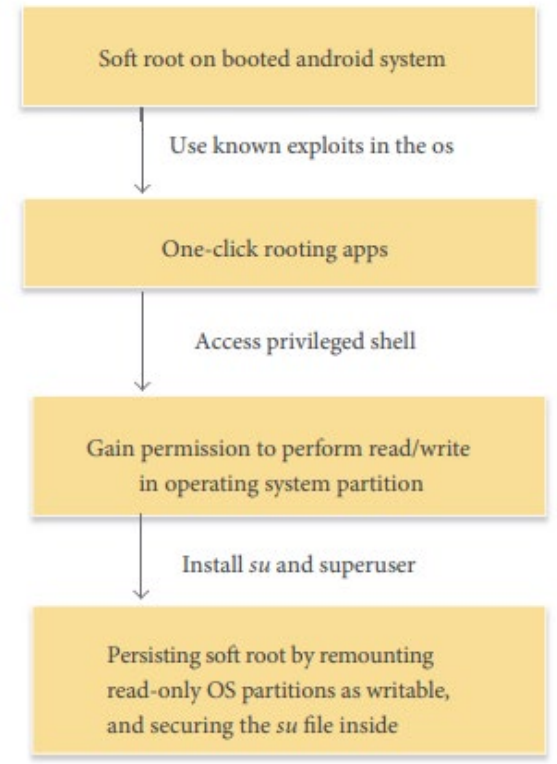
Android uređaji svojim korisnicima ne dopuštaju potpun pristup (*eng. root access*) operacijskom sustavu uređaja, za obične korisnike to ne predstavlja nikakav problem no korisnici koji žele raditi teške promjene sustava ili samo eksperimentirati sa operacijskim sustavom moraju napraviti proces koji se zove *„rooting"*. Ima puno razloga zašto bi netko htio *„root"-at* svoj uređaj. Primjerice Android uređaji dolaze sa već unaprijed instaliranim aplikacijama koje se ne mogu izbrisati i koje korisnik uopće ne koristi a zauzimaju memorijski prostor , RAM,resurse te troše bateriju uređaja. Takve aplikacije može izbrisati samo privilegirani korisnik koji oma *root access. Root* procesom se također mogu dodavati nove mogućnosti te micati stare te ostale modifikacije više razine. Prednosti izvođenja ovog procesa su već navedene modifikacije i popravak telefona no ima lošija strana (EDUCBA, https://www.educba.com/rooting-android/ , 14.09.2020.) :

- *„Bricking a device"* - što u doslovnom prijevodu znači pretvoriti svoj telefon u ciglu to jest on postane beskoristan ili ozbiljno pokvaren ukoliko se *root* napravi na pogrešan način.
- *„Tweaking"* - Uređaj se može početi ponašati na čudan način zbog modifikacija i promjena u samom korijenu operacijskog sustava mobitela.
- *Ažuriranja softvera* – uređaj koji je *„rootan"* vrlo vjerojatno neće moći ažurirati svoj mobitel zbog promjena u operacijskom sustavu što znači da starija verzija operacijskog sustava možda neće biti dobro zaštićena od napada kao ona novija verzija.

Prema (Nguyen-Vu, Chau, Kang, & Jung, 2017.) postoje dvije vrste kojima se ovaj proces može izvesti :

1. *Soft Root*
2. *Hard Root*

„Soft Root“ se izvodi na sustavu koji je već pokrenut i trenutno u funkciji to jest korisnik koristi taj mobitel. Ova tehnika se bazira isključivo na softveru i uključuje iskorištavanje ranjivosti u kernelu. Bazirano na ranjivostima u sustavu kreirane su mnoge takozvane *„One click rooting apps“* koje korisniku omogućavaju da jednim klikom u toj aplikaciji *„roota“* uređaj. *„Root“* proces ovom metodom može biti privremen ili krajnji ovisino o aplikaciji koja se koristila jer Android ima u sebi ugrađen mehanizam *„Verified Boot“* spomenut u prvom poglavlju koji provjerava vjerodostojnost uređaja te da li je njegov operacijski sustav modificiran što znaći da se prednosti dobivene ovim procesom mogu poništiti sljedeći put kada se uređaj resetira i pokrene. *„Hard Root“* zahtijeva fizičku interakciju sa uređajem. Glavni cilj ove metode je pokrenuti sustav u takozvani *„Recovery mode“* koji je izoliran te od tamo napraviti instalaciju potrebnih paketa kako bi se dobio status *„super user“* to jest status super korisnika (Nguyen-Vu, Chau, Kang, & Jung, 2017.).



Slika 21 „Soft Root“ (Nguyen-Vu, Chau, Kang, & Jung, 2017.)

ZAKLJUČAK

Sa ubrzanim razvojem pametnih telefona i njihovih funkcionalnosti nove prijetnje i mogući napadi razvijaju se također. Razlikujući se od osobnih računala , riješenja i metode prevencije suzbijanja prijetnji upućenih prema pametnim telefonima moraju uzeti u obzir više faktora , kao što su ograničeni dostupni resursi , razne funkcionalnosti koje napadači mogu iskoristiti kao što su razne metode konekcije, raznolikost aplikacija i njihovih izvora te ljudski faktor i činjenicu da je pametni telefon stalno u pokretu i lako ga je izgubiti te sa njime sve svoje podatke. Većina literature predlaže korisnicima imati barem jednu sigurnosnu aplikaciju koja će imati *anti-malware* i *anti-theft* funkcionalnosti. Najbolji način obrane od napada , te iskorištavanja ranjivosti pametnih telefona je biti informiran i objektivan o mogućim rizicima (*bežične mreže , nepouzdani izvori za aplikacije itd...)* kao i održavati mobitel ažuriranim i zaštićenim sa PIN lozinkom ili *touchscreen* uzorkom radi maksimalne sigurnosti podataka.

LITERATURA

Knjige :

- Becher, M. (2009). *Security of Smartphones.* Mannheim.
- (2015). Android Security Internals. U N. Elenkov, *Android Security Internals* (str. 12-19). USA: William Pollock.
- Mulliner, C. R. (2006). *Security of Smart Phones.* Santa Barbara.
- Abu-Nimeh, S., Becher, M., Fogie, S., Hernacki, B., Morales, J. A., & Wright, C. (2009). *Mobile Malware Attacks and Defense.* Burlington: Syngress Publishing, Inc.

Radovi i članci :

- Amarante, J., & Barros, J. P. (2017). *Exploring USB Connection Vulnerabilities on Android Devices.*
- Antonioli, D., Tippenhauer, N. O., & Rasmussen, K. (2020.). *BIAS: Bluetooth Impersonation AttackS.*
- Becker, A. (2007). *Bluetooth Security & Hacks.*
- Douligeris, C., & Mitrokotsa, A. (2003). *DDoS attacks and defense mechanisms: classification and state-of-the-art.* Piraeus.
- Khan, R., & Kumar, A. (2019). *A Malicious Attacks and Defense Techniques on Android-Based Smartphone Platform.* Blue Eyes Intelligence Engineering & Sciences Publication.
- Lehembre, G. (2005). *Wi-Fi security – WEP, WPA and WPA2.*
- Nguyen-Vu, L., Chau, N.-T., Kang, S., & Jung, S. (2017.). *Android Rooting: An Arms Race between Evasion and Detection.* Seoul.
- Pereira, A., Correia, M., & Brandão, P. (2017.). *USB connection vulnerabilities on Android smartphones:.* Center for Research in Advanced Computing Systems (CRACS-INESC LA);.
- Rawal, A., Chhikara, G., Kaur, G., & Khanna, H. (2019). *Cryptography Algorithm.* Haryana: Journal of Analog and Digital Communications.
- Shaik, A., Borgaonkar, R., Niemi, V., & Seifert, J. P. (2017). *Practical Attacks Against Privacy and Availability in 4G/LTE Mobile Communication Systems.*

- Zhang, L. (2018). *Smartphone App Security: Vulnerabilities and Implementations.* Zhang, Linxi.
- Ahvanooey, Q. Li, M. Rabbani, & R. A. Rajput, (2017). *A Survey on Smartphones Security: Software Vulnerabilities, Malware, and Attacks* (str. 30). China: International Journal of Advanced Computer Science and Applications.
- Android. (2019). *Android Security & Privacy 2018 Year in Review.*
- Apple. (2012). *iOS Security.*
- Cusack, B., Lutui, R., & Khaleghparast, R. (2016). Detecting Slow DDos Attacks on Mobile Devices . *Detecting Slow DDos Attacks on Mobile Devices* , 3-5.
- Google. (2018). *Android Security & Privacy.* USA: Google.
- Heera, A. B. (2008). *CELL PHONE VIRUS AND SECURITY.*
- Islam, M., & Jin, S. (2019). *An Overview Research on Wireless Communication Network.* Chongqing.
- Kouliaridis, V., Barmpatsalou, K., Kambourakis, G., & Chen, S. (2020). *A Survey on Mobile Malware Detection Techniques.*
- Lipovský, R., Štefanko, L., & Braniša, G. (2016.). *The Rise of Android Ransomware.*
- Lookout. (2016). *Technical Analysis of Pegasus Spyware.*
- Lookout. (2017). *Pegasus for Android Technical Analysis and Findings of Chrysaor.*
- Maheshika, P. (2019). iOS Security Model. *Medium*.
- Markota, K. (2018). *SVJESNOST KORISNIKA O FAKTORIMA RIZIKA PRILIKOM KORIŠTENJA MOBILNIH APLIKACIJA.* Osijek: Sveučilišni studij računarstva.
- Moonsamy, V., & Batten, L. (2014.). *Mitigating man-in-the-middle attacks on smartphones – a discussion of SSL pinning and DNSSec..*
- Nguyen, H. G. (2018). *Wireless Network Security.*
- Prey. (2018). *Mobile Theft &.* Prey.
- Sarmar, M., & Soomro, T. R. (2013). Impact of Smartphone's on Society. U M. Sarmar, & T. R. Soomro, *Impact of Smartphone's on Society* (str. 217). European Journal of Scientific Research.

- Theoharidou, M., Mylonas, A., & Gritzalis, D. (2012.). *A Risk Assessment Method for Smartphones.* Athens.

- Threat Lab. (2019). *Understanding how IMSI-Catchers exploit cell networks(probably).* Electronic Frontier Foundation.

- Wong, L. W. (2005). *Potential Bluetooth Vulnerabilities in Smartphones.* Lih Wern Wong.

Internet izvori :

- EDUCBA. (2020, https://www.educba.com/rooting-android/). *10 Advantages and Disadvantages of Rooting Android devices.*

- Kaspersky. (2020, https://securelist.com/mobile-malware-evolution-2019/96280/). *Mobile malware evolution 2019.*

- Kaspersky. (2014, https://www.kaspersky.com/blog/cabir-10/5107/). *10 interesting facts about Cabir, a decade-old smartphone virus.*

- NetSpot. (2020, https://www.netspotapp.com/wifi-encryption-and-security.html). *Wireless Security Protocols : WEP,WPA,WPA2 and WPA3*

- Varonis. (2020, https://www.varonis.com/blog/man-in-the-middle-attack/). *What is a Man-in-the-Middle Attack: Detection and Prevention Tips.*

POPIS SLIKA

Popis Slika :

SAŽETAK

Pametni telefoni, od svojih skromnih početaka ranog 21. stoljeća, postali su sastavni dio života ljudi. Gotovo polovica stanovništva na svjetskoj razini, oko 3.8 milijardi ljudi, koristi pametni telefon. Inovativnom tehnologijom i kompaktnim dizajnom, pametni telefoni pružaju bezbroj informacija nadohvat ruke svakom korisniku. Praksa je pokazala da mnogo ljudi zaboravlja da mnoštvo informacija s kojima raspolažu nisu jednosmjerna. Zlonamjerni korisnici mogu doći i do privatnih informacija mnogih korisnika zloupotrebljavajući tehnologije pomoću kojih funkcioniraju pametni telefoni. Napada koji ciljaju korisnike pametnih telefona ima puno i koriste se raznim tehnologijama koje korisnici upotrebljavaju gotovo svaki dan, kao što su : *Bluetooth , Wi-fi , Aplikacije itd...* Cilj ovog rada je ukazati na ulogu i značaj sigurnosnih aspekata pametnih telefona kako bi se kod šire javnosti više percipirala važnost upotreba mjera prevencija za moguće napade na pametne telefone.

Ključne riječi: sigurnost pametnih telefona, bežične tehnologije, zlonamjerne aplikacije, informacije

ABSTRACT

Since their humble beginnings in the early 21st century , smartphones have become integral part of people's lives where almost half of population , 3.8 billion people is an active smartphone user. With their innovative technology and compact design they offer us the world of information at the grasp of our hand. But almost everyone forgets that all that information that we have is a one-way relationship. We become vulnerable as someone also has the ability to access our personal information abusing the technologies on which smartphones operate. Attacks that target smartphone users are many and they use different technologies , ones we use every day, such as : *Bluetooth , Wi-fi*, Apps etc... Because of that, it is necessary that we pay attention on our personal data , the way we use smartphones and to take certain measures to prevent such attacks.

Key words: smartphone security, wireless technologies, malicious applications , data