KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS

Algis Kulbačiauskas

Apsauga nuo slapta daromų garso ir vaizdo įrašų Android išmaniuosiuose telefonuose

Vadovas

Doc. dr. Jonas Čeponis

KAUNAS, 2017

KAUNO TECHNOLOGIJOS UNIVERSITETAS

INFORMATIKOS FAKULTETAS

KOMPIUTERIŲ KATEDRA

Apsauga nuo slapta daromų garso ir vaizdo įrašų Android išmaniuosiuose telefonuose

Baigiamasis magistro darbas

Informacijos ir informacinių technologijų sauga (kodas 621E10003)

Vadovas

(parašas) Doc. dr. Jonas Čeponis

(data)

Recenzentas

(parašas) Doc. Ignas Martišius

(data)

Projektą atliko

(parašas) Algis Kulbačiauskas

(data)

KAUNAS, 2017

KAUNO TECHNOLOGIJOS UNIVERSITETAS

Informatikos fakultetas

(Fakultetas)

Algis Kulbačiauskas

(Studento vardas, pavardė)

621E10003 Informacijos ir informacinių technologijų sauga

(Studijų programos pavadinimas, kodas)

„Apsauga nuo slapta daromų garso ir vaizdo įrašų Android išmaniuosiuose telefonuose“

AKADEMINIO SĄŽININGUMO DEKLARACIJA

20 17 m. 05 22 d.

Kaunas

Patvirtinu, kad mano **Algio Kulbačiausko** baigiamasis projektas tema „Apsauga nuo slapta daromų garso ir vaizdo įrašų *Android* išmaniuosiuose telefonuose“ yra parašytas visiškai savarankiškai, o visi pateikti duomenys ar tyrimų rezultatai yra teisingi ir gauti sąžiningai. Šiame darbe nei viena dalis nėra plagijuota nuo jokių spausdintinių ar internetinių šaltinių, visos kitų šaltinių tiesioginės ir netiesioginės citatos nurodytos literatūros nuorodose. Įstatymų nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs.

Aš suprantu, kad išaiškėjus nesąžiningumo faktui, man bus taikomos nuobaudos, remiantis Kauno technologijos universitete galiojančia tvarka.

(vardą ir pavardę įrašyti ranka) *(parašas)*

Kulbačiauskas, A. „Apsauga nuo slapta daromų garso ir vaizdo įrašų *Android* išmaniuosiuose telefonuose“. Magistro baigiamasis projektas / vadovas doc. dr. Jonas Čeponis; Kauno technologijos universitetas, Informatikos fakultetas, Kompiuterinių sistemų inžinerijos katedra.

Kaunas, 2017. XX p.

1. SANTRAUKA

Visuomenė sparčiai artėja prie ribos, kada gyvenimas be technologijų yra sunkiai įsivaizduojamas – jos naudojamos beveik kiekviename šiuolaikinio žmogaus žingsnyje. Jei anksčiau technologijos, pavyzdžiui, kompiuteris, buvo naudojamas tam skirtoje patalpoje ar bent prie darbo stalo, šiandien dauguma planetos gyventojų technologijas – išmanųjį telefoną – turi visada su savimi. Čia kyla vis didesnė grėsmė žmogaus privatumui: daugėjant išmaniųjų telefonų, atsirado ir daugiau programišių, kurie geba įsilaužti į šiuos įrenginius ir geba pasižiūrėti, ką žmogus veikia, ar geba pasiklausyti pokalbių telefono savininkui apie tai nežinant. Antivirusinių programų kūrėjai turi sudarytą neoficialių privačių organizacijų sąrašą, kuris siekia šimtus įvairaus pobūdžio virusų kūrėjų, kurie bando pasipelnyti vykdydami nelegalią veiklą. „Kaspersky“ - antivirusinių programų kūrėjai – pateikia statistiką, jog jie aptikto daugiau nei 427 programėlių, kurios veikė kaip virusai ir buvo atsiųsti iš oficialių programų platintojų svetainių. Apsisaugoti nuo tokių virusų, ypač informatikos žinių neturinčiam žmogui yra beveik neįmanoma, todėl atsiranda poreikis turėti pagalbines priemones, kurios padėtų apsaugoti telefono savininko privatumą.

Remiantis [25] šaltinio duomenimis, apie 27% visų *Android* išmaniųjų telefonų virusų sudaro Trojos arkliai, kurie gali perimti pagrindines telefono funkcijas. O didžioji dalis susideda iš „malware“ ir „rat“ tipo virusų, kurie yra ne tokie pavojingi, tačiau rizika privatumui vis tiek išlieka. Visi šie trys tipai virusų veikia skirtingu principus, tačiau jie visi sukelia informacijos nutekėjimo grėsmę. Galime įsivaizduoti scenarijų, kada aukšto rango politiko arba verslininko yra klausomasi nuotoliniu būdu jam apie tai nieko nežinant, valstybės lygio paslaptys arba privataus verslo sandoriai bei sprendimai gali būti išgirsti trečių šalių - tai gali nuvesti iki įmonės bankroto. O kameros įjungimas gali išduoti įslaptintų arba privačių dokumentų turinį. Dokumentų turinys kartais gali būti svarbesnis už išgirstą pokalbį, todėl apsauga nuo neautorizuoto vaizdo įrašinėjimo yra labai aktuali.

Rinkoje yra sukurti keli sprendimai slapta daromų įrašų *Android* išmaniuosiuose telefonuose prevencijai, tačiau jie vartotojui yra nepatogūs. Pavyzdžiui, norint atsiliepti į gaunamą skambutį, vartotojas pirmiausia atsidaryti programėlę ir išjungti garso blokavimą. Pabaigus pokalbį, vėl rankiniu būdu įjungti garso blokavimą. Analogiškas nepatogumas yra apsisaugant nuo slapta daromo vaizdo įrašo. Šio darbo tikslas yra atlikti detalią situacijos analizę ir pasiūlyti sprendimą, kuris neturėtų nurodytų trūkumų.

Pirmoje darbo dalyje yra aptariama, kokie virusai yra sukurti ir kokią riziką jie kelia vartotojui. Virusai gali būti parsisiunčiami pačio telefono savininko arba priverstinai atsiunčiami ir įdiegiami iš trečių asmenų. Šiame darbe koncentruojamasi į apsaugą nuo plačiausiai paplitusių Trojos arklių ir „Rat“ (angl. žiurkė) tipo virusų.

Antroje darbo dalyje yra atliekama analizė, tiriant koks būdas yra efektyviausias kameros ir mikrofono blokavimui. Analizuojamos *Android* infrastruktūrinės bibliotekos būtinos atlikti pagrindinėms funkcijoms. Taip pat nagrinėjamas procesų išrinkimo metodas, padedantis išjungti blokavimus bet kuriai programai, kuri buvo įtraukta į neblokuojamų programų sąrašą.

Trečioje darbo dalyje atliekamas realios situacijos tyrimas bandant tiesiogiai įsibrauti į telefoną, įdiegus kenkėjišką plėtinį, kurio pagalba galima prisijungti į viruso sukurtą serverį. Tyrimui buvo pasirinkti trys populiariausi Trojos arklio tipo virusai, tarp kurių buvo mokami ir laisvai visiems

prieinamo atviro programinio kodo. Bandymai buvo atlikti su „DroidJack“, „SpyNote“ ir „DroidRat“ Trojos arkliais. Kiekvienu atveju vaizdo ir garso blokavimas buvo sėkmingas, tačiau virusai atlikti garso ir vaizdo įrašo nesugebėjo.

Kulbačiauskas, A. Research On Methods Of Application Level Iot Objects Identification: Master‘s thesis in „Protection from secretly made audio and video recordings in *Android* smartphones“ / supervisor assoc. prof. Dr Jonas Čeponis. The Faculty of Informatics, Kaunas University of Technology.

Research area and field: *Android* audio and video security.

Key words:

Kaunas, 2017. XX p.

2. SUMMARY

Society quickly goes to the edge where life without technology is unimaginable. Also, devices are getting smaller and smaller while security threat is growing because almost every person owns a smartphone and they might be under surveillances that uses their personal all the time possibly even without knowing that. Antivirus companies collect statistics of the unofficial companies which codes various types of malware viruses specifically for *Android* phones and there are hundreds of them. According to Kaspersky antivirus designer there were 427 apps which were detected as viruses and they were sent from official Google Play site. It is hard to protect ourselves from virus attacks without any knowledge of IT, so additional programs come in handy to do it for us.

Almost 25% of all detected *Android* smartphone viruses are Trojan horse type viruses, which can take over critical phone functions. Another part of viruses are malware and rat type viruses which are less dangerous but they still are considered as a threat. We can imagine an example when high rank politician or businessman are exposed to video and audio surveillance by third party persons using their phones, which in turn could cause national security threat exposure or may be the cause of business bankrupt under that circumstances. Camera surveillance can lead to national security leak of classified documents and this can be even more dangerous than just listening for conversations, therefore, securing video camera is more important than microphone.

Blocking camera and microphone also brings a lot of inconvenience for the user. For example, when he or she receives a phone call, the person would need to disable any mic blocking before answering the call, same with camera. It brings a lot of inconvenience when first thing needs to be done before taking a picture is disabling camera blocking, so workarounds are made to bypass this issue in this paper.

In the first part of the work the overview of the available viruses and the risk they expose is provided. Viruses can be downloaded by the users or by exposing security gaps of the phone. To make a protection from the viruses which uses *Android* design flaws is extremely hard, so we will focus to secure *Android* smartphones from viruses such as Trojan horse, malware, and rat type viruses which are most popular.

Second part of the work is analysis of best way to block microphone and camera. Analysis of *Android* infrastructure design to choose best method for camera and audio blocking. Selecting best method of how to exclude particular applications from the blocking list.

In third part, real situations will be tested by exploiting phone with viruses and testing how newly created prototype application will block audio and video input. For testing purposes three type of trojan horse viruses were choses to see if they can be blocked. Testing were made using the most popular and free to use viruses named „DroidJack“, „SpyNote“ and „DroidRat“. Testing was successful and all three trojan horse viruses were unable get any response from the camera or microphone.

3. Turinys

4. Paveikslėlių sąrašas

5. TERMINŲ IR SANTRUMPŲ ŽODYNAS

PID - Proceso identifikavimo numeris (angl. Process identification numer).

API - programos sąsaja (angl. Application program interface).

SD – Išorinė kietoji laikmena (angl. Solid drive).

JNI – Java vietinė sąsaja (angl. Java Native Interface)

HAL – Aparatinės įrangos apibendrintas lygis (angl. Hardware Abstraction Layer).

OSS – Atvira garso sistema (angl. Open Sound System).

ALSA -Pažangi Linux garso architektūra (angl. Advanced Linux Sound Architecture).

IPC – Tarp procesorinis bendravimas (angl. interprocessor communication)

LRU – Mažiausia paskutinė panauda (angl. least recently used).

RAM – Atsitiktinai pasiekiama atmintis (angl. Random access memmory).

IMSI – Tarptautinis mobilaus abonento identifikatorius (angl. International Mobile Subscriber Identity).

6. ĮVADAS

Magistro baigiamajame darbe „Apsauga nuo slapta daromų garso ir vaizdo įrašų *Android* išmaniuosiuose telefonuose“ siūlomi ir tiriami apsaugos metodai, padėsiantys apsaugoti žmogaus privatumą blokuojant vartotojo ir išorinių programų prieigą prie kameros ir mikrofono. Nagrinėjama kameros ir mikrofono blokavimo metodai, nes tai yra paprasčiau ir saugiau, nei bandyti skenuoti visas įdiegtas programas telefone ir ieškoti jose žalingo kodo. Siekiant sukurti metodą, padėsiantį išsaugoti telefono savininko privatumą, darbe nagrinėjami praktikoje naudojami būdai, kurie efektyviai padėtų uždrausti prieigą prie kameros ir mikrofono, nes apsaugos metodas atimti galimybę atlikti žalingus veiksmus yra saugesnė ir patikimesnė, nei bandymas surasti, kuri telefone įdiegta programa yra žalinga ir kelia pavojų privatumui. Atliekant šį darbą buvo sukurtas prototipas ir išbandytas jo veikimas su virtualiomis ir fizinėmis mašinomis, kas leistų įsitikinti, jog sukurtas prototipas veikia. Pritaikant *Android* operacinės sistemos jau paruoštas bibliotekas ir vadovaujantis *Android* pagalbine projektuotojo duomenų baze (angl. *Android Knowlege Center*) yra atliekamas tyrimas pasirenkant efektyviausią metodą, kuris bereikalingai nenaudotų telefono resursų, taip neblogindamas vartotojo patirties naudojantis telefonu, būtų patikimas ir sunkiai apeinamas perrašant programinį kodą ir perskaičius programos paketą, naudojantis atvirkštiniu programavimo metodu (angl. *Reverse enginering*), kad būtų labai sunku sukurti programą, kuri sugebėtų apeiti visas sukurtas apsaugas.

Bandymo dalyje yra aprašomi keli metodai ir atliekamas tyrimas išrinkti tinkamiausiam metodui. Pirmuoju metodu yra nagrinėjamas būdas, kada bandoma sustabdyti kamerą bandant stabdyti procesus renkantis pagal jų PID (angl. *process identification*) numerį. Šio būdo privalumai yra tai, kad tai leidžia sustabdyti programos kreipimąsi į kameros ir mikrofono įrašymo API, tačiau šį metodą lengva apeiti perrašius *Android* OS naudojamą API įdiegiant jį į pačios programos paketą, tokiu būdu programa nebesikreipia į specialias iš anksto parašytas bibliotekas taip nesužadindama API sąsajos panaudojimo trigerio, šį metodą naudoja antivirusinės programos [24]. Kitas metodas yra skenuoti programas ir ieškoti žalingo kodo jose. Ne visada atsiųsta programa iš oficialios parduotuvės bus saugi, nes galima atsisiųsti tik dali programos, o likusią jos dalį atsiunčiant iš trečiųjų šalių serverių su žalingu kodu, tokiu būdu yra apeinama pirminė apsauga, kada bendrovė „Google“ patikrina programos patikimumą prieš dedant ją „Google Play“ parduotuvę leisdama ją visiems atsisiųsti. Taip iš trečiųjų šalių atnaujinta programa praeina nepatikrinta ir aptikti tokios programos veiksmus yra labai sunku ir ne visada patikima. Metodas, kuris bus išsamiai aprašomas darbe, tai tiesioginis kameros blokavimas naudojant administratoriaus sisteminį vartotoją iškviečiant „setCameraDisabled()“ metodą, kuris yra integruotas į pačią „Media“ klasę. Garsas tokios klasės neturi, tačiau garsui sustabdyti bus naudojama kita klasė, „UserRestriction“, kuri uždeda vartotojui draudimą taip neleisdama jam atlikti tam tikrų veiksmų be sisteminio vartotojo sutikimo, kuris įgalino draudimą.

6.1. **Darbo problematika ir aktualumas**

Taip sparčiai tobulėjant technologijoms ir mažėjant kompiuterių dydžiui žmonės juos pradeda nešiotis kartu su savimi, kas stipriai padidina privatumo pažeidimo riziką. Nuo to gali nukentėti ne tik civiliai, bet ir valstybės tarnautojai, kurie saugo valstybės lygio saugumo paslaptis. Remiantis „Kaspersky“ atliktais tyrimais [25] nustatyta, jog 2016 metais buvo aptikta per 427 programėlių, kurios veikė kaip virusai, o 2015 buvo aptikta per 245. Tai parodo, jog 2016 metais su *Android* susijusių pranešimų apie virusus padidėjo beveik 25 procentais, prognozuojama jog 2017 m. šis

skaičius gali padidėti iki 35 procentų. Vaizdu ir garsu yra perduodama daugiau informacijos nei vien tik iš elektroninių dokumentų, todėl apsauga nuo jų yra labai aktuali. Nesugebėjus apsaugoti verslo nuo trečiųjų šalių atliekamo konferencijų pokalbių, nutarimų, verslo pasiūlymų įrašinėjimo, gali įmonė žlugti.

6.2. **Darbo tikslas ir uždaviniai**

Darbo tikslas: pasiūlyti ir ištirti galimus metodus bandant apsaugoti telefoną nuo nuotolinio kameros ar mikrofono įjungimo be savininko žinios, pasirinkus priimtiniausią būdą, kuris nesunkintų telefono savininko patirties juo naudojantis ir bereikalingai neapkrautų telefono resursų, taip paprastindamas telefono veikimo spartą ir sutrumpindamas baterijos tarnavimo laiką. Ištestuoti programos veikimą skirtinguose įrenginiuose (fiziniuose arba virtualiuose), pamatuoti jos resursų sunaudojimą, ištestuoji, ar ji atlieka pagrindines savo funkcijas.

6.3. **Darbo struktūra**

Šis dokumentas sudarytas iš keturių pagrindinių skyrių:

- išsami problemos analizė su pavyzdžiais
- rinkoje esančių analogų tyrimas ir analizė
- pasirinkto metodo įgyvendinimas
- bandymo rezultatai

7. BŪDAI SLAPTA DARYTI GARSO IR VAIZDO ĮRAŠUS

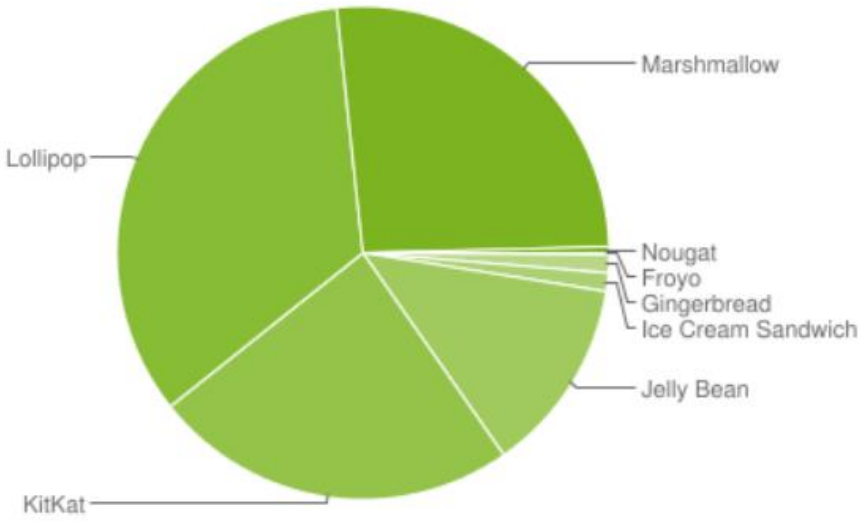
Remiantis [24], [25], [26] šaltinių duomenimis 86 % visų kenkėjiškų programų ir kompiuterinių virusų yra platinama juos užmaskavus tarp populiarių, gerą funkcionalumą turinčių ar net geros reputacijos programų. Pavyzdžiui, QR kodui skaityti skirtos programos, kurioms reikalinga prieiga prie kameros, kad galėtų nuskaityti QR kodą ir jį atpažinti, tačiau tokio tipo programėlės yra užmaskuojamos kaip piktavališkos programos, skirtos sekti ar šnipinėti.

Android operacinės sistemos problemos nėra išsprendžiamos su antivirusiniėmis programomis, nors ir visi antivirusinių programų kūrėjai tvirtina, kad jų programa gali padėti apsaugoti jų telefonus, bet didžiausia problema ta, kad dauguma *Android* įrenginių negauna saugumo spragų pataisymų, o tai piktavaliams padeda ilgą laiką naudotis rastais operacinės sistemos trūkumais [1].

Android programos yra paleidžiamos „smėliadėžės“ (angl. *sandbox*) aplinkoje, tokiu būdu ji yra izoliuota nuo sistemos ir nuo bendros *Android* operacinės sistemos resursų, nebent leidimai yra suteikti pačio naudotojo, kuris instaliavo programą. Labiausiai pažeidžiamos yra 4.4 „Kitkat“ ir senesnės versijos *Android* operacinės sistemos. Kaip matyti 1 pav., 5.0 ir žemesnės „Lollipop“ versijos naudotojų yra daugiau negu 50 procentų. Senesni įrenginiai, naudojantys *Android* kaip operacinę sistemą, yra labai pažeidžiami, nes turi daugiau pažeidžiamumų nei naujesni modeliai, išleidžiami su naujesnėmis operacinės sistemos versijomis, nors ir yra žinoma, kad senesnės kartos telefonai yra labiau pažeidžiami, nes jiems nėra siunčiama jokių naujinimų, nes naujinimai yra įdiegiami su nauja *Android* versija, kuri paprastai yra palaikoma tik naujausių išmaniųjų telefonų,

todėl apsisaugoti senesnį telefoną nuo kibernetinių atakų yra pačio savininko rūpestis. Prieš įdiegiant programą iš oficialios „Google Playstore“ parduotuvės iššoka pranešimas apie programai reikalingas teises į telefoną, kad galėtų naudotis tam tikromis aparatinėmis įrangos dalimis, bet vartotojas dažnai nesusimąsto, kad kai kurių specifinių leidimų tai programai nereikia, tuo pasinaudoję trečiųjų šalių žmonės į jas įrašo žalingą kodą. Dažniausiai populiariąsias programas, žaidimus arba kitokio turinio pagalbines programas nusipirkę arba nelegaliai įdėję įskiepį programišiai, priverčia telefono savininką atsisiųsti naujinimus į telefoną jam nepastebėjus, tokiu būdu vartotojas gauna ir kenkėjiškus pakeitimus. Telefono savininkas turėtų pats pasirūpinti savo telefono saugumu. Įdiegiant žaidimą ar kokią kitą programą vartotojas turi patikrinti ir įsitikinti, ar leidimai, kurių prašo suteikti ta programa, tikrai yra reikalingi jai veikti. Tokios privilegijos, kaip įjungti vibratorių arba įrašyti duomenis į išorinę atmintį, yra reikalingos, tačiau jai greičiausiai nereikia prieigos prie SMS, kontaktų sąrašo, kameros arba mikrofono. Telefono savininkas turėtų atkreipti dėmesį jeigu programa, kurią norima įdiegti, prašo šių privilegijų suteikimo: kameros, mikrofono, garso padidinimo ar sumažinimo galimybės, taip pat leidimą sustabdyti telefoną nuo užmigimo. Šios keturios privilegijų rūšys yra pačios populiariausios, nes Java klasės *Android* operacinės sistemos yra suprogramuotos taip, kad darant fotografiją yra skleidžiamas garsas, tam, kad būtų pranešama apie bandymą pasinaudoti kamera, o garso nutildymas šią funkciją išjungia. Google korporacija sukūrė programą, kuri nuolat tikrina telefone vykstančius procesus ir žiūri, ar programos nebando daryti žalingų veiksmų, šis išmaniųjų telefonų programų tikrinimo metodas yra prieinamas nuo 2.3 *Android* versijos. Tačiau pavojus slypi ne tik iš programinės puses, bet ir nuo skylių, paliktų pačioje *Android* operacinėje sistemoje jos kūrimo metu. Šiuos trūkumus išnaudojęs piktavalis dažniausiai gauna teisę naudoti pagrindines „root“ vartotojo teises telefone, tai jam leidžia prieiti prie bet kurio leidimo be vartotojo sutikimo. Visos išorinės programos kreipiasi į „manifest“ leidimus. Vienas iš sprendimo būdų, tai sekti, kada yra duota prieiga į specifinius leidimus, tai užfiksavus galima sustabdyti nesankcionuotus veiksmus.

Version	Codename	API	Distribution
2.2	Froyo	8	0.1%
2.3.3 - 2.3.7	Gingerbread	10	1.2%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	1.2%
4.1.x	Jelly Bean	16	4.5%
4.2.x		17	6.4%
4.3		18	1.9%
4.4	KitKat	19	24.0%
5.0	Lollipop	21	10.8%
5.1		22	23.2%
6.0	Marshmallow	23	26.3%
7.0	Nougat	24	0.4%



1 Pav. *Android* OS paplitimas.

2013-aisiais metais visuomenei buvo paskelbtas straipsnis, kad Jungtinės Amerikos Valstijos ir Didžiosios Britanijos žvalgybos agentūros turi visišką prieigą prie telefone esančių tokių duomenų, kaip SMS, GPS koordinatės, elektroninis paštas ir kitos sritys, keliančios grėsmę saugumui. Sustabdyti kenkėjiškus veiksmus, padarytus su aukščiausio lygio sisteminiu vartotoju naudojant išorines pagalbines priemones gali būti labai sunku norint sustabdyti nepageidautina veiklą be vartotojo žinios, kadangi šios spragos, per kurias gali patekti valdžios agentūros, yra iš anksto numatytos telefono gamintojų. Norint žinoti, kaip būtų galima apsisaugoti nuo visų atakų, reikėtų sužinoti, kokiais būdais yra patenkama į telefoną [13].

7.1. **Kuo skiria „*Windows*“ ir „*Android*“ antivirusinės programos**

Tokios antivirusinės programos, kaip „Windows Defender“ yra automatiškai įdiegiamos kartu su „Windows 8“ versija operacinės sistemos diegimo metu. Ji skenuoja failus jų atidarymo metu, tikrina tinklu perduodamus duomenis ir atsisiunčia naujausius patobulinimus iš „Microsoft Update“ duomenų bazės. „Windows“ operacinės sistemos antivirusinės programos stebi tinklu perduodamus duomenis, kaip prevenciją atsisiųsti virusų naudodamos vadinamą „Smart screen filter“ būdą ir tuo pačiu apsaugo nuo „exploit“ tipo virusų, kurie plinta atidarius paleidžiamąjį failą. „Chrome“ ir „Firefox“ naršyklės taip pat blokuoja pavojingas svetaines, kuriose yra virusų. Naršyklės ir jų papildai yra labiausiai atakuojamos aplikacijos, nes per jas į asmeninius kompiuterius patenka daugiausiai virusų. „Java“ ar „Flash“ yra vienos iš labiausiai naudojamų programavimo kalbų, kuriomis plinta virusai internetu, neapsaugota nuo jų ir *Android* [2]

Kitaip nei „Windows“, *Android* antivirusinės sistemos turi limituotą galimybę prieiti prie sistemos vien dėl *Android* operacinės sistemos dizaino savybių. Visos išmaniųjų telefonų antivirusinės programos yra įdiegiamos „smėliadėžės“ aplinkoje ir turi labai limituotas galimybes apsaugoti nuo „protingų“ virusų. Kenkėjiškos programos, kurios patenka į telefoną kartu su naudingo turinio programėlėmis kaip įskiepiai, yra lengviau pastebimos dėl jų specifinio veikimo principo, kuris yra lengvai pastebimas tikrinat kreipinius į *Android* API. Kitaip nei Windows operacinės sistemos, *Android* antivirusinės programos negali gauti aukščiausio lygio vartotojo dėl saugumo, o tai labai apriboja jos veikimo galimybes. Suteikus aukščiausio lygio vartotojo teises kiltų dar didesnė rizika privatumui [2].

Antivirusinės programos taip pat gali turėti failų skanavimo funkciją siūlydamos patikrinti SD ar vidinės atminties failų talpyklas ir tik tuos failus, kuriuos gali pažiūrėti pats vartotojas, tai jeigu virusas yra sisteminiame lygmenyje, antivirusinė programa neturi galimybių jį aptikti. Antivirusinės programos gali skanuoti ir APK formato programinių paketų failų paketus SD laikmenoje, bet tik tuos, kurie gali būti prieinami paties žmogaus. *Android* platformų antivirusinės programos gali skenuoti tinklu siunčiamus paketus, taip iš žalingų svetainių gali uždrausti ateinančius nesaugius paketus taip užkertant kelią atsisiųsti kenksmingas programas.

Android jau turi tam tikrų saugumą užtikrinančių būdų tokiu kaip „Google Plays Store“. *Google* prieš įdėdami programą į savo parduotuvę ją perskanuoja, jeigu kenkėjiški pakeitimai buvo padaryti vėliau su patobulinimais tai *Google* ištrina kenkėjišką programą iš savo parduotuvės ir net gali ištrinti iš paties telefono nuotoliniu būdu. Taip pat *Google Chrome* naršyklė turi įdiegtą funkciją, kuria žmogus yra perspėjamas apie galimai žalingo turinio svetainę, kurioje gali būti virusas ir leidžia pasirinkti, kokius veiksmus galima atlikti.

Antivirusinės programos telefonams nėra taip stipriai rekomenduojamos ,kaip Windows sistemai, nes dėl operacinės sistemos architektūros ji tampa į kartus saugesnė nei Windows sistema, ir gali net pabloginti telefono darbą, greičiau iškraudamos bateriją, sulėtinti telefono veikimą.

Prastesnės reputacijos antivirusinės programos siunčia melagingus pranešimus, neva yra rastas virusas ir jį ištrinti galima tik nusipirkus pilną antivirusinės programos versiją [4].

7.2. **„Stagefright“ virusas**

Vienas iš labiausiai paplitusių „Exploit“ tipo virusų yra „Stagefright“ virusas. Tai yra *Android* komponento saugumo spraga multimedijos grotuve, per kurią labai lengvai galima įsilaužti į mobilųjį *Android* telefoną tik nusiuntus MMS (angl. *multimedia messaging service*) žinutę su nekaltai atrodančiu tekstu, kuriame kaip įskiepis yra įdiegtas žalingas kodas, jis paleidžiamas per multimedijos komponentą žinutės perskaitymo metu.

Ši spraga atsirado, kada dauguma *Android* telefonų gamintojų neatsakingai suteikė „Stagefright“ sistemai vos silpnesnes sisteminį vartotoją nei „ROOT super user“ saugumo privilegijas. Įsilaužėlis atsiuntęs MMS žinutę į telefoną, kuri paleista naudojant "media“ Java klase kartu su "root" privilegijomis. Atsižvelgiant į saugumo konfigūracijas telefone įsilaužėlis gali gauti telefonui visas administratoriaus privilegijas, vadinasi, kad jis nepastebėtas gali padaryti beveik bet ką. Šis metodus buvo plačiai naudojamas prisijungti prie telefono nuotoliniu būdu.

Pagrindinis dėmesys buvo skiriamas „Stagefright“ problemai spręsti, tačiau tai nėra vienintelė saugumo spraga. „Mediaserver“ pažeidžiamumas nėra išnaudojamas vien tik MMS pagalba, dar vienas būdas užkrėsti telefoną yra per naršyklę. Paleistas MP4 tipo vaizdo failas su kenkėjišku kodu per interneto naršyklę taip pat gali perimti telefono valdymą. Tai yra yra labai efektyvus viruso platinimo būdas, nes užtenka tik nueiti įinternetinę svetainę ir naudojant *Java Script*, paleisti MP4 failą , o kartu su juo ir kenkėjišką kodą. Tuo pačiu būdu gali būti paleidžiamas virusas ir per *Android* programą su įdiegtu kenkėjišku vaizdo failu. Telefono pažeidžiamumą virusu „Stagefright“ galima patikrinti viešai platinama programa, kurią sukūrė „Zimperium“ komanda. Jie ir aptiko šia saugumo spragą, o vėliau ir sukūrė programą, kuri patikrina, ar telefone buvo ištaisyta ši problema.

Antivirusinės programos ant *Android* operacinės sistemos elgiasi kiek kitaip, jos bando paleisti visas programas smėlio dėžėje ir apriboti leidimus, kuriuos jos turi. Kadangi antivirusinė turi per mažas teises, ji negali nei uždrausti sustabdyti atsiųstos žalingos programos, nei sustabdyti vartotoją, kad jos negalėtų atsiųsti ar startuoti. Sustabdyti iš naršyklės paleisto „Exploit“ tipo viruso telefone taip pat negali, tad nėra būdo, kaip būtų galima nuo jų apsisaugoti naudojant antivirusinę programą. Jeigu virusas jau telefone, tai antivirusinė programa su juo nieko negali padaryti dėl per mažų jai suteiktų teisių, taigi žalinga programa gali sau toliau netrukdoma veikti, nes ji turės didesnes „root“ teises nei pati antivirusinė [3] [31].

7.3. **Būdai apsisaugoti nuo „Stagefright“ atakų**

Android antivirusinės programėlės negali apsaugoti nuo šio tipo atakų, nes dauguma telefonų turi užrakintas „root“ teises. Antivirusinės programėlės negali patikrinti gauto MMS pranešimo ir pažiūrėti, ar ta žinutė nebando pasinaudoti „Stagefright“ bibliotekos spraga. „Google“ nepataiso šios problemos, nes jų tikslas pakeisti *Android* infrastruktūrą iš pagrindų ir įdiegti vartotojui naują versiją su jau pataisytomis klaidomis. Kada naujas telefono gamintojas nori išleisti telefoną, jis privalo gauti leidimą iš „Google“, kad galėtų įdiegti „Google Play“ aplikaciją į savo gaminamą telefoną. Susitarimo sąlygos yra tokios, kad „Google“ bet kada gali įdiegti atnaujinimus ar pakeitimus be vartoto ar gamintojo sutikimo ir visa tai vyksta sisteminiame lygmenyje. „Google“ dažnai įdiegia įvairiausius atnaujinimus per „Google Play“ sistemą iki seniausių versijų tokių kaip 2,2 „Froyo“,

pavyzdžiui, „Google“ pridėjo *Android* įrenginio sekimo funkciją į beveik visus *Android* įrenginius be vartotojo sutikimo. „Google“ pataisė „Stagefright“ spragą su 5.0 „Lollipop“ versija, tačiau daugybė senų telefonų vis dar turi šią spragą, nes senųjų telefonų operacinės sistemos nebeatnaujinamos.

Apsisaugoti nuo „Stagefright“ galima uždėjus draudimą MMS programai automatiškai paleisti MMS žinutes, o tai reikštų išvis išjungti “MMS auto-retrieval” nustatymus , tada MMS žinutė nebus automatiškai paleidžiama. Išjungus šią funkciją, žinutė automatiškai nebus paleista, o norint ją perskaityti reikės ant MMS paspausti, tada ji automatiškai atsisiųs. Jeigu MMS atėjo iš pažįstamo asmens, tai dar nereiškia, kad ši žinutė yra be virusų, nes asmuo, kuris siuntė gali nežinoti, kad jo telefonas yra užkrėstas kirminu arba veikiant programiniam kodui, kirminas gali pradėti daugintis, taip išsiųsdamas žinutę visiems kontaktams iš adresų knygutės [5].

7.4. **Būdai įjungti kamerą nuotoliniu būdu telefone**

Paviešintoje nacionalinio saugumo informacijoje yra atskleidžiami būdai, kaip yra pažeidžiama žmonių teisė į privatumą. Per žinių kanalą NBC „Wiki Leaks“ įkūrėjas Edward Snowden atskleidė, jog nacionalinis saugumas naudoja metodą, vadinamą DROPOUTJEEP įsilaužimui į mobiliuosius telefonus ir garso ir vaizdo įrašų darymui gaunant pilnas „super user“ teises į telefoną. DROPOUTJEEP - tai programa, kuri, anot Edward Snowden, buvo sukurta išnaudoti specialiai paliktomis skylėmis telefonuose pačių telefonų gamintojų gyventojų šnipinėjimo tikslais. Ši programa gali išsiųsti ar atsiųsti failus, gauti SMS turinį, visus kontaktus, balso paštą, esamą būvimo vietą pasinaudojus GPS arba ryšio antenos apytiksliai nustatyta vieta, mikrofonu ar kamera. Visa gauta informacija yra išsiunčiama ir užšifruojama tam, kad nebūtų įmanoma susekti, kokio tipo informacija buvo išsiųsta. DROPOUTJEEP programą programišiai gali paleisti telefone nuotoliniu būdu. Šis metodas yra dažniausiai naudojamas „IPhone“ telefonuose, tačiau *Android* - ne išimtis kaip ir *Blackberry*. Kadangi *Android* telefonai užima kiek daugiau nei 50% visos telefonų rinkos, todėl šiame darbe didesnis dėmesys buvo skirtas jiems [6].

7.5. **Būdai daryti vaizdo įrašą**

Techniškai daryti vaizdo įrašą telefone to nežinant pačiam vartotojui, atrodytų, yra sudėtinga, nes *Android* yra parašytas taip, kad norėdamas filmuoti privalai rodyti vaizdą ekrane, t.y, fone su aktyvia programa slapto vaizdo įrašo daryti neišeina. Tačiau faktiškai tai apeiti yra labai paprastas būdas: programėlės vaizdą padaryti 1x1 pikselio dydžio, kas yra praktiškai nepastebima, net ir žinant kur tiksliai yra 1x1 pikselio dydžio taškas.

Kaip būtų galima apsisaugoti nuo tokių aplikacijų paprasto būdo nėra, nes, kad ir koks atidus ar atsargus bebūtų vartotojas, vis tiek bus rasti būdai kaip įdiegti žalingas programas į telefoną. Tačiau vienas iš elementariausių patarimų yra atkreipti dėmesį į programas ir kokios prieigos jos prašo, kaip pavyzdys, tinkamam skaičiuotuvo funkcionavimui nereikia prieigos prie kameros ar mikrofono. Jei diegiama skaičiuotuvo programėlė prašo tokių teisių, yra rizika, jog programėlė gali savyje turėti virusą.

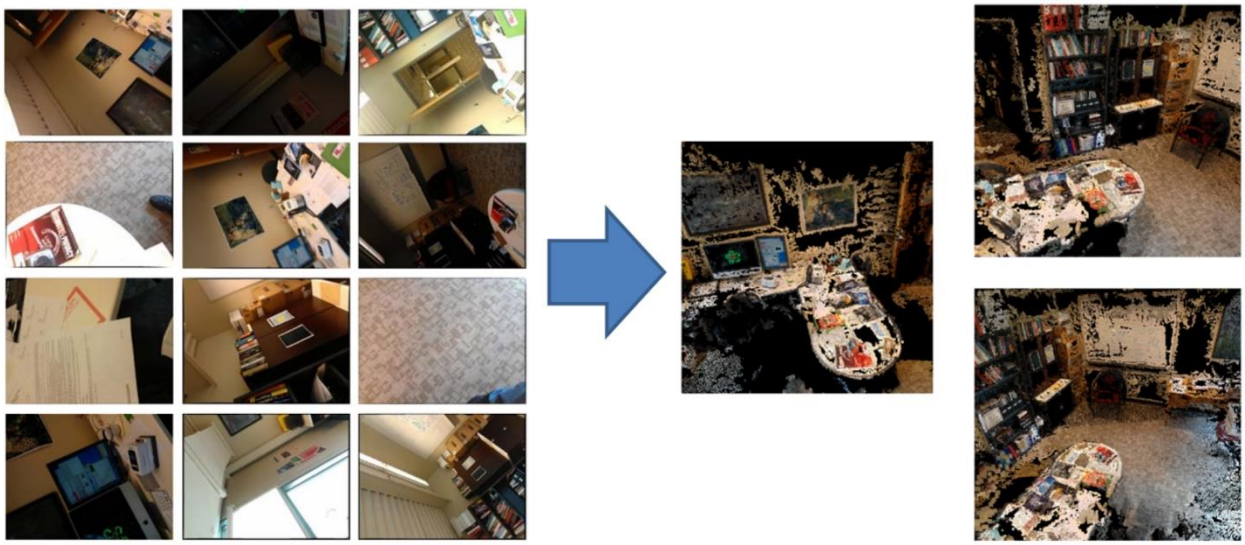
Vienas iš labiausiai paplitusių būdų įsilaužti į telefoną yra per vartotojo sąsają, o tai reiškia, kad įsilaužus į vartotojo „Google“ prieigą, turėsi ir prieigą į patį telefoną, o įsilaužėlis tada, galės padaryti labai daug. Siekiant eliminuoti šią riziką, vartotojai turėtų reguliariai ištrinti nenaudojamas programas, nes laikui einant jos gali būti nupirktos blogų ketinimų turinčių įmonių ar žmonių ir su kitu atnaujinimu gali būti įdiegtas kenkėjiškas kodas. Pastebėjus didesnį duomenų perdavimą ar

baterijos sunaudojimą, neaiškius procesus ir resursų sunaudojimą galima įtarti, kad kažkas ne taip. Tokios programos kaip skaičiuotuvas neturėtų veikti fone ir gali būti traktuojamos kaip virusas dėl neįprasto elgesio [7].

Kadangi jau savaime yra padaromos skylės ir galimybės įsilaužti į telefoną pačių gamintojų, tai vienas iš tokių metodų yra: įsilaužėliui reikalinga turėti mobili stotis kurios pagalba jis gali perimti telefono signalo ryšį taip apgaudamas telefoną, kad jis pradeda manyti, jog yra prisijungęs prie pagrindinės ryšio tiekėjo stoties, o jau tada galima nesunkiai pradėti siųsti kenkėjiškas programas, tačiau tokia įranga ne kiekvienam prieinama. Nors kompanija pavadinimu „OMA (Open Mobile Alliance)“ kuri ir paliko šią skyle visuose įrenginiuose, kurie jungiasi radijo ryšiu, tvirtina pataisę šią spragą tačiau ją vis dar galima išnaudoti įvairiausio tipo įrenginiuose tokiuose kaip telefonai, modernūs automobiliai, laisvų rankų įranga ir kita, o ši spraga suteikia įsilaužėliui pilną teisę į tą įrenginį. Buvo pastebėta, kad per WiFi buvo galima nuotoliniu būdu įjungti kamera ar valdyti Bluetooth ausines [8].

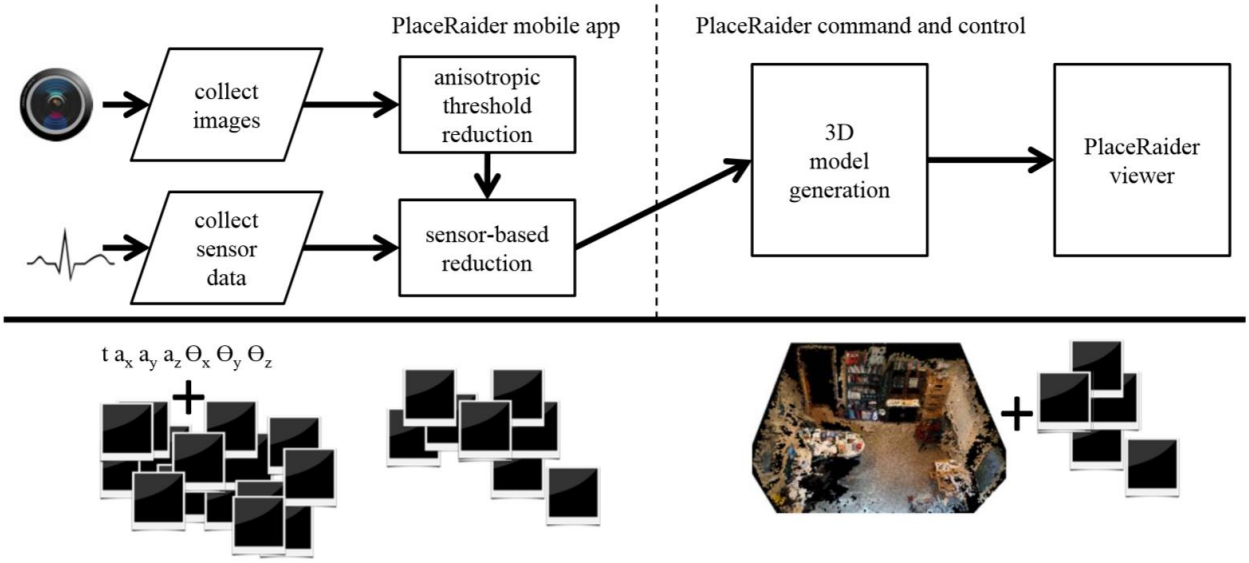
7.6. **„Placeraider“ virusas**

„Placeraider“ virusas buvo sukurtas karinei pramonei ir jo galimybės yra didžiulės su „root“ teisėmis kaip ir dauguma kitų virusų, tačiau jis yra paruoštas šnipinėjimui, todėl jo funkcionalumas yra bemaž didžiausias palyginus su kitais virusai. „Placeraider“ sugeba ne tik pajausti, kokius judesius telefono savininkas atlieka pirštais, kokioje padėtyje jis yra, guli, stovi ar sėdi, tačiau didžiausias jo pranašumas yra tame, kad jis gali padaryti patalpos 3D vaizdo modelį naudodamas giroskopu nustatyta pozicija ir iš atskirų nuotraukų sudėlioti pilną patalpos ar aplinkos planą. Šio tipo virusai yra vadinami „sensory malware“ arba sensoriai virusai, nes jie ne tik renka duomenis, bet ir atlieka skaičiavimus, reikalingus nustatyti žmogaus judesį ar vietą. Pasitelkus telefonų didelės raiškos kameras ir giroskopo parodymus, galima apskaičiuoti ir padaryti 3D modelį, tačiau tuo užsiima jau pats atakuotojas, nes telefonu padarytas didelės raiškos nuotraukas su visais apskaičiavimais telefone užimtų nemažai laiko, kad būtų suformuotas pilnas modelis ir be to aukai gali kilti įtarimų dėl pasikeitusių telefono resursų per trumpą laiko tarpą.



2 Pav. suformuotas vaizdas iš atskirų nuotraukų

„Placeraider“ virusas naudojasi 3 Pav. pavaizduota architektūra. Duomenys surenkami nuotoliniu būdu per mobilųjį telefoną, tada pati telefono operacinė sistema atlieka nuotraukų sumažinimus, panaikindama nenaudingą informaciją nuotraukose ir palieka juodas dėmes, kurios padeda sumažinti paruošto siųsti paketo dydį kartu su visa giroskopo surinkta informacija.



3 Pav. Duomenų apdorojimo ir išsiuntimo schema.

Šis virusas yra platinamas kartu su panašius leidimus turinčiomis programomis įdiegiant jį į pagrindinį instaliacinį paketą. Įdiegtas jis veikia telefono fone ir nepriklausomai nuo atsiųstos programos yra be jokio vaizdinio apipavidalino. Šis virusas atpažįsta dokumentus esančius ant stalviršių. Pavyzdžiui kada yra dirbama prie stalo ir yra paimamas telefonas į ranką parašyti žinutę, tuo momentu kamera yra puikioje padėtyje stalo atžvilgiu ir gali padaryti fotografijas su visais dokumentais esančiais ant jo, galbūt banko kortelėmis ar Amerikoje populiariomis čekių knygelėmis. Tokiu būdu nesunkiai galima nuskaityti kalendorių esantį ant stalo ir turėti informaciją kada auka nebus namuose, arba kurią valandą ir kur jis tiksliai bus. Šiam virusui tereikia vos kelių leidimų ir net nesuprasi, kad paprasta fotokameros programa su naudingomis funkcijomis yra „Tojos arklys“ įsikūręs telefone. Reikalingos prieigos yra leidimas įjungti kamerą, įrašyti bei ištrinti failus, valdyti garsą ir naudoti internetą. Beveik visos šiais laikais suinstaliuotos programos telefone prašo tų pačių leidimų, todėl išskirti iš visų kitų aplikacijų ir laikyti ją žalinga tiesiog nekiltų klausimas.

Informacijos apdorojimui ir sumodeliuoti 3D paveiksliuką pačiame telefone naudojant 1 procesorių užtruktų keletą valandų tačiau taip sparčiai didėjant telefonų GPU ir CPU pajėgumams tokius paveiksliukus galės suformuoti ir pačiame telefone, o tai smarkiai sumažintų siunčiamų duomenų kiekį. Dabar tokio tipo virusą galima paleisti ir per reklamas kurios ateina su dauguma aplikacijų jos yra atsiunčiamos į telefoną be vartotojo sutikimo ir su jomis gali ateiti virusas.

Daryti fotografijas nei daryti vaizdo įrašą yra geriau dėl kelių priežasčių tokių kaip: vaizdo failas užima daugiau vietos ir daugiau eikvoja baterija tačiau yra susiduriama su kita problema, tai yra fotografijos sukeltas garsas, arba rodomas kameros vaizdas ekrane. Tai galima apeiti tik keliais būdais tai užtildyti garsą ir padaryti fotografija tada vėl pagarsinti ir reikia padaryti taip, kad nebūtų

galima matyti kameros rodomo vaizdo ant ekrano, o tai galima apeiti padarius, aplikaciją vieno pikselio dydžio arba padaryti, kad rodomas vaizdas butu visiškai permatomas, bet pastarasis būdas yra įmanomas ne ant visų *Android* versijų [9].

7.7. **Mygtukų registras pasinaudojus kamera**

Kembridžo mokslininkai pademonstravo galimai naują būdą kaip galima kameros ir mikrofono pagalba nuspėti kokius mygtukus spaudo telefono savininkas, tad kibernetiniams nusikaltėliams būtų paprasta sužinoti kokį PIN kodą suspaudo žmogus. Iš 200 bandymų spėjimų tikslumas buvo apie 60 procentų ir mažesnius slaptažodžius atspėjo 50 procentų tikslumu. Buvo sukurta programa, kurios pagalba programa galėjo mokintis, kokius judesius daro žmonės spausdami tam tikrus mygtus. Pavaizduota 4 paveikslėlyje.

Figure 15: Areas reachable by the thumb with little help from supporting fingers.

Figure 16: PIN-pad with renamed digit for a right-handed user.

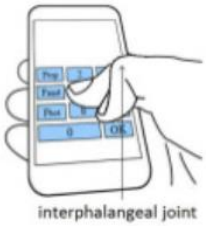


Figure 17: Rotation of thumb with interphalangeal joint.

Figure 18: Rotation of thumb with carpometarcarpal joint.

4 Pav. pirštų judesiai spaudant mygtukus

7.8. **Stacionarus telefono bokšto generatorius „StingRay“**

“Stingray” tai yra prietaisas Dar vadinamas “IMSI-catcher” veikianti per SS7 protokolą išnaudodamas SS7 pažeidžiamumo galines duris (angl. back door) ir skirtas imituoti GSM telefoninį bokštą, tam, kad būtų galima perimti telefono signalą bet kurio dažniu. Šis aparatas buvo sukurtas “Harris Corporation” įmonės, kuri priklauso kariuomenė ir yra labiausiai naudojamos Jungtinėse Amerikos Valstijose saugumo agentūros ir “Stingray” pavadinimas dabar yra priskiriamos ir kitiems aparatams su tokiu pačiu funkcionalumu. Toks stebėjimo būdas yra vienas pavojingiausių, nes telefonas su bokštu bendrauja labai silpnu užšifruotu kanalu. Kas nėra paslaptis, jeigu ryšio tiekėjas paprašo telefono bendrauti nesaugiu ryšiu, telefonas iš karto patvirtina nesaugų ryšį ir toliau bendrauja atviru tekstu, o taip pat gali pareikalauti įrašyti meta-duomenis į vidinę atmintį, perimti pokalbių duomenis, perimti visus kriptografinius raktus iš telefono.

“Stingray” perima telefono signalą padidinęs signalo stipruma ir todėl telefonas automatiškai atsijungia nuo legalaus ryšio tiekėjo bokšto ir prisijungia prie mobilaus “Stingray” aparato. Tinkle telefonai yra atpažįstami pagal jų unikalius IMSI ir EMS kodus, o netikras bokštas jos parsisiunčia tiesiai iš telefono atminties. Naudojantis šiuo aparatu neįmanoma sužinoti koks telefonas yra sekamas, kartą jau prisijungus prie telefono ir gavus telefono duomenis, vėliau tą telefoną galima

surišti su žmogumi kuris jį nešioja. “Stingray” bando automatiškai palaikyti stabilų ryšį su telefonu žiūrėdamas, kad nereikia būtų per daug galios tam, kad taupyti akumuliatorių energiją. Jis automatiškai supranta kada žmogus pradeda garsiau kalbėti ir pagal tai automatiškai padidino galios išėjimas tam kad ryšys nenutrūktų ir telefonas vis dar būtų jungiamos prie mobilios aparatūros.

5 Pav. „Stingray“ ryšio stotelė

Ši „žmogus viduryje ataka“ (angl. *man in the middle attack*) yra pavojinga tuo, kad perimus telefono signalą galima ne tik klausytis pokalbių, bet ir rinkti visus duomenų perdavimo paketus ir, jeigu jie yra šifruoti, tai galima nesunkiai dešifruoti super kompiuterių pagalba per kelias minutes, nes telefonai dažniausiai naudoja 1024 bitu RSA kriptografinius raktus. Kartu su duomenų perdavimu piktavaliai gali atsiųsti paleistuką (angl. *exploit*) ir tokiu būdu galima gauti ROOT teises į telefono operacinę sistemą atsiuntę SMS, MMS žinutę kuri panaši į paprastą žinutę ir pasinaudodami „Stagefright“ pažeidžiamumu jį aktyvuoti. Tai reiškia, kad yra suteikia pilna prieiga prie duomenų, sisteminiu registrų, kurie kaupia visa informacija o iš jų galima nuskaityti viską net ir ištrintus duomenis tokius žinutės, taip pat gali nuotoliniu būdu įjungti kamerą. Gavus pilną prieigą su ROOT teisėmis užpuolikas gali išjungti telefone einančius procesus kurie gali trukdyti jam atlikti piktavališkus veiksmus [17].

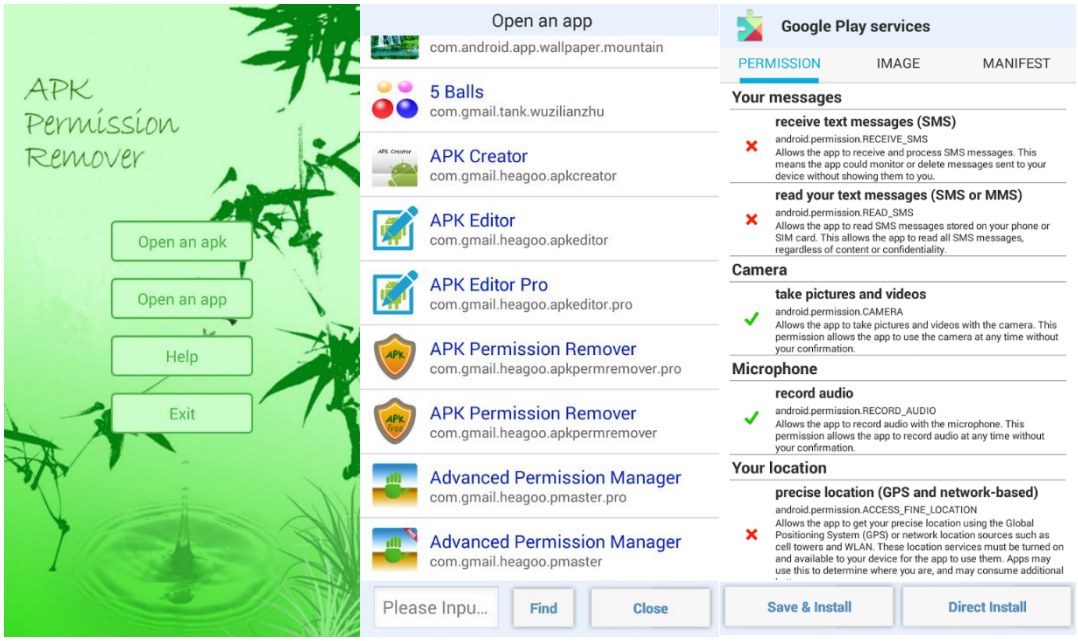
7.9. **Esami problemos sprendimo įrankiai**

Rinkoje jau egzistuoja aplikacijos, kurios padeda užblokuoti kitų aplikacijų prašomas privilegijas, kas apsaugo telefoną nuo žalingų aplikacijų uždraudžiant joms atlikti tam tikrus veiksmus. Tačiau tokiu metodu pašalinus privilegijas ar jas užblokavus, programa nustoja veikti, kada ji kreipiasi į privilegijų sąrašą, kuris jai buvo numatytas iš anksto.

Darbe įvertintos kelios aplikacijos, kurios buvo išbandyto ir kurios funkcionuoja. Šiuo metu yra keturios programos kurios geriausiai apsaugo telefonas nuo šnipinėjimo ir ne tik uždraudžia prieiga prie kameros ir mikrofono bet taip pat atlieka ir operacijos apsaugoti nuo bendro duomenų gavimo telefone. Esamų įrankių trūkumas yra funkcionalumo stoka, t.y., įrankiai nesuteikia galimybės neblokuoti vartotojo pasirinktų programų. Tokią galimybę suteikia sprendimo būdas, kuris buvo sukurtas šio tyrimo metu.

7.9.1. **Programa „APK Permission Remover“**

Ši programa yra naudinga tuo, kad ji gali skenuoti visas programas, esančias vartotojo išmaniajame telefone, ir surinkti visų aplikacijų prašomos privilegijas vienoje vietoje. Tada vartotojas, pasirinkęs leidimą, kurį pageidauja nutraukti, ir programa pašalina leidimą.

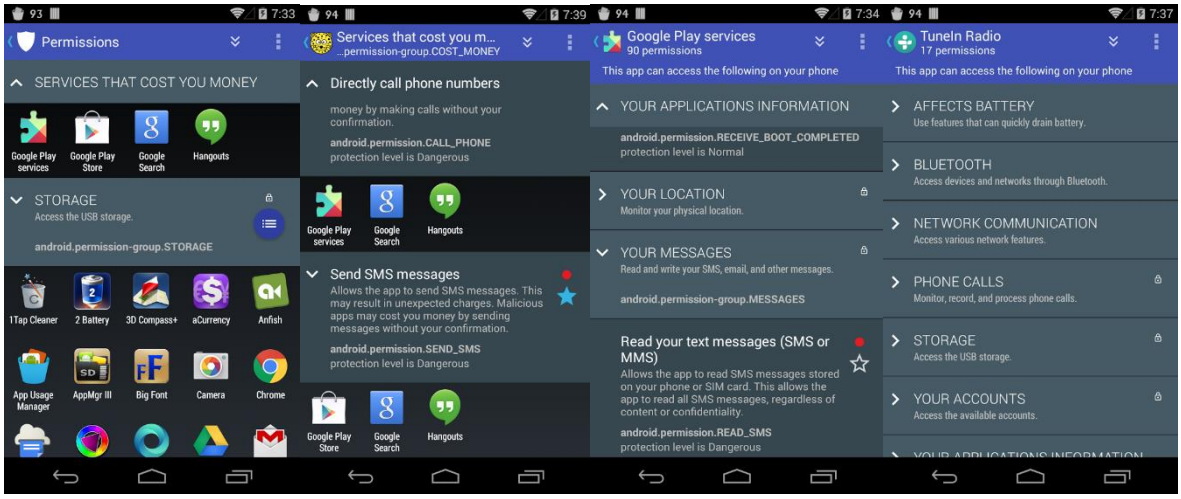


6 **Pav**. „APK Permission Remover“

„APK Permission Remover“ nereikalingos “ROOT” teisės, todėl ja gali naudotis visi vartotojai. Ši programa gali panaikinti privilegijas prieš įdiegiant naują aplikaciją arba pakeisti seniau įdiegtoms aplikacijoms suteiktas privilegijas. Programa pakeičia aplikacijų APK failą, kuriame yra aprašytos programai reikalingos privilegijos, tokiu būdu galima įsidiegti aplikacijos, kurios nori ir gali būti tikras, kad toje programa galimai esantys šnipinėjimo įrankiai negali būti naudojami. Tokiu būdu pašalinus programos privilegijos ji nebeveikia, nes nebegali tęsti numatytų funkcijų.

7.9.2. **Programa „aSpotCat“**

“aSpotCat” yra nemokama programa, kuri suteikia dideles galimybes išfiltruoti ir atrinkti visas privilegijas visų aplikacijų, kurios yra įdiegtos į telefoną ir surūšiuoti jas į grupes. Programa „aSpotCat“ taip pat pateikia kiekvienos privilegijos aprašymą, kuriame nurodomas privilegijos rizikos lygis. Naudodamasis aprašymu ir nurodytu rizikos lygiu, vartotojas gali priimti sprendimą, kurias programas vartotojas turėtų vėliau ištrinti. Programos trūkumas yra tas, jog pati programa prevencijos nevykdo, ji tik pateikia aprašymą. Vartotojas, turinti mažiau techninių žinių, gali netinkamai įvertinti programos nurodomą rizikos lygį ir nesiimti veiksmų ar imtis neefektyvių veiksmų.



7 **Pav**. “aSpotCat” programosvaizdas.

7.9.3. **Programa „D-Vasive Pro“**

Viena iš didelį potencialą turinčių aplikacijų yra “D-Vasive Pro”. Jos pagrindinis tikslas yra pranešti vartotojui apie jo telefone daromus veiksmus arba juos iš karto uždrausti. Ši programa uždraudžia tik specifinius aplikacijų veiksmus, tokius kaip: draudimas išjungti kamerą, mikrofoną, *bluetooth*, *wifi* ryšį, taip pat turi prevencija prieš “stingray” puolimo būdą. Kiekvienoje iš išvardintų prevencijų yra galimybė pranešti vartotojui apie veiksmus, vykdomus telefone, tačiau programa „D-Vasive Pro“ leidžia naudotis visomis funkcijomis visų aplikacijų ,esančių telefone. Kai D-Vasive Pro” aptinka galimą pažeidimą, programa parodo pranešimą, kad buvo bandyta aktyvuoti mikrofoną arba kamerą. Programa veikia šiuo principu: pilnai uždraudus prieigą į kamerą, kas buvo atlikta darbo autoriui testuojant programą, aplikacijos, kurios bando naudoti kamerą, nustojo veikti dėl sutrikusio kameros darbo.

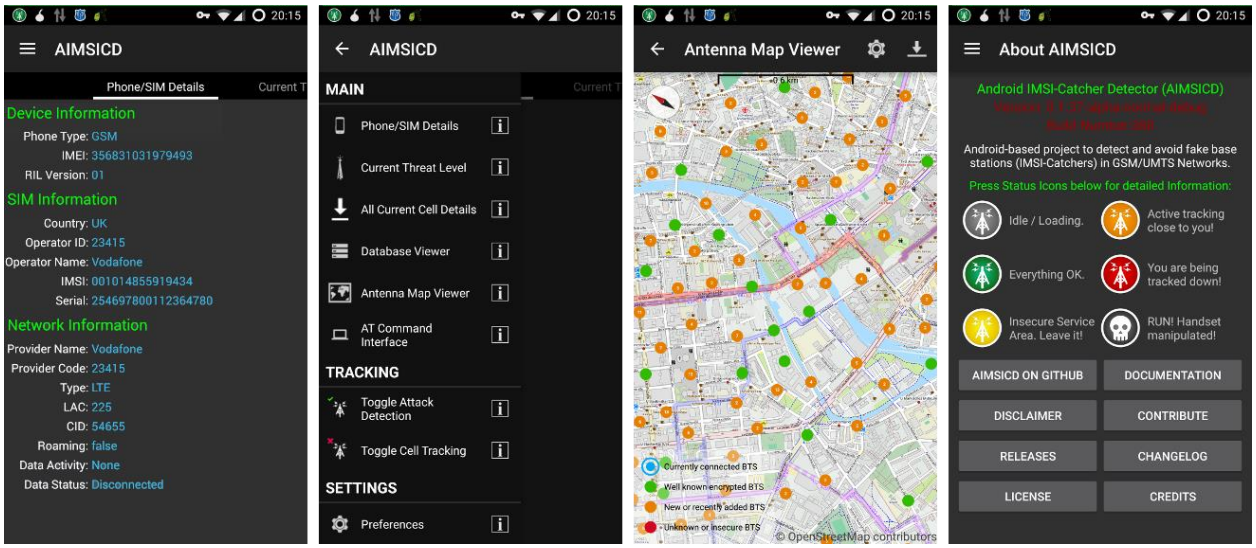
8 Pav. "D-Vasive Pro" programosfunkcionalumas

Šios programos privalumas yra toks, kad ji gali kaupti visus įrašus, kuriuos galima vėliau peržiūrėti ir pastebėjus įtartiną programos veikimą, vartotojas gali imtis veiksmų. Kaip ir kitos aukščiau apžvelgtos programos, „D-Vasive Pro" nesuteikia galimybės vartotojui pasirinkti programas, kurios nebūtų blokuojamos nepaisant jų atliekamų veiksmų. Tai yra vienas iš programos „D-Vasive Pro" tobulino galimybių.

7.9.4. **„AIMSICD" programa**

"AIMSICD" yra programa, orientuota į "Stingray" arba dar kitaip vadinamas "IMSI-Catcher" atakas. Ji veikia stebėdama telefoninį ryšį ir stebi, ar telefonas nėra peradresuojamas prie kito tinklo tiekėjo (šis klausimas bus išsamiau apžvelgtas kitoje dalyje apie "Stingray" atakas). "AIMSICD" programosprivalumas yra tas, kad jis seka telefoninių bokštų būklę bei jų parametrus, o tai leidžia efektyviai nustatyti pokyčius vietos atžvilgiu ir daryti išvadą, ar telefoninis bokštas yra tinklo tiekėjų, ar "Stingray" aparatūros. Ši programa turi kelis perspėjimo būdus tokius kaip: „telefonas saugus", „aplinka nesaugi", „galimas netikras bokštas yra labai arti", „esi susektas", „telefono kontrolė perimta". Programa suteikia galimybę pasižiūrėti į žemėlapį ir pamatyti visus aktyvius bokštus, prie kurių telefonas yra kažkada prisijungęs. Žemėlapis padeda nustatyti, kurie bokštai yra saugūs, t. .y, naudoja šifruotą ryšį ir yra legalūs tinklo tiekėjų bokštai, taip pat aplikacijoje galima peržiūrėti bokšto siunčiamą identifikacijos informaciją. Ši programa seka ne tik bokštų padėtį, bet ir stebi žalingas SMS žinutes, per kurias dažnai ateina žalingi įskiepiai į telefoną. Per tuos įskiepius puolėjas gauna ROOT teises į telefono operacinę sistemą, o tai reiškia, kad įdiegtos apsaugos nebepadės sustabdyti įsilaužėlio: turint ROOT teises programišius gali uždaryti bet kurį procesą, kuris jam trukdo atlikti jo siekiamus veiksmus, todėl uždrausti įjungti kamerą arba mikrofoną (kas yra šio darbo tikslas), kad būtų galima pasiklausyti pokalbių, yra labai sunku, o gal net ir neįmanoma.

AIMSICD programa yra atviro kodo, todėl galime laikyti saugesne, nes prieš ją įdiegiant galima pasižiūrėti, kokius veiksmus programa atlieka.



9 Pav. "AIMSICD" programosvaizdas.

7.10. **Analizės apibendrinimas**

Atlikus kenkėjiškų programų analizę, galima daryti išvadas, kad rizika privatumui yra labai didelė dėl plataus kenkėjiškų programų paplitimo ir gebėjimo užsimaskuoti tarp naudingo funkcionalumo ar nekaltai atrodančių programų. Pagal paplitimą pirmauja „DroidRat" ir sudaro apie 72% visų kenkėjiškų programų. Iš likusių kenkėjiškų programų apie 24% sudaro Trojos arklio tipo virusai, kurie pagal savo galimybes yra gerokai pavojingesni nei „DroidRat" virusai. Į 4% įeina labai didelę grėsmę keliantys virusai kuriuos aptikti yra ypatingai sunku, nes paprastai jie išnaudoja *Android* operacinės sistemos dizaino trūkumus dėl ko kyla sunkumu juos aptinkant. Tiriant analogus buvo pastebėta, jog rinkoje jau yra sukurta programėlių kurios blokuoja vaizdo ir garso signalus, taip pat yra programų kurios skirtos specifiniams virusams aptikti kurių neranda antivirusinės ir kitokio apsauginio turinio programos. Atlikus analize matomas akivaizdus poreikis prevencijai, prieš kenkėjiško turinio programas, keliančias grėsmę savininko ir aplinkinių privatumo saugumui.

8. *ANDROID* PROGRAMĖLIŲ TYRIMAS IR REALIZACIJA

Pasaulyje yra apie 1,4 milijardo *Android* įrenginių ir, 2017 metų duomenimis, yra parašyta daugiau nei 2,8 milijono programėlių, skirtų *Android* išmaniesiems telefonams [22]. Didesnė dalis šių programėlių yra naudingos vartotojui. Tačiau yra daugiau nei 40 skirtingų kompanijų, kurios užsiima kenkėjiškų programėlių kūrimu, yra žinomi ir jų pavadinimai [23]. Kenkėjiško programėlės, kurios užsiima nuo reklamos siuntimo iki privačių duomenų rinkimo, vis dažniau atsiranda ir legalioje „Google" autorizuotoje duomenų bazėje. Kadangi ne visos pagalbinės antivirusinės programos gali jas aptikti, todėl vartotojai dairosi ir į privačių asmenų kurtas programėles, skirtas apsaugoti

privatumą. Mažiausiai 5% visų žalingų programų, esančių rinkoje, gali slapta įjungti kamerą ar pasiklausyti pokalbio, įjungdami pasiklausymo galimybę nuotoliniu būdu. Statistikos, kiek yra tokiu būdų padaryta nusikaltimų, nėra, nes padaryti statistikos tyrimus yra beveik neįmanoma, nes aptikti tokį įsilaužimą yra labai sunku. Jeigu populiariausios antivirusinės programos galėtų aptikti ir sustabdyti tokį įvykį, jį būtų galima įtraukti į statistiką. Todėl reikalingos kitos priemonės, kurios drastiškai panaikina galimybę naudotis kamera ar pasiklausyti pokalbių. Kadangi apsisaugoti nuo visų įmanomų pavojų, susijusių su informacijos rinkimu, dėl rizikų masto ir skirtingų rizikų pobūdžio vienas sprendimas nesuteiks galimybių, todėl darbo apimtis yra siaurinama: darbe bus apžvelgtos tik tos programėlės, kurios saugo žmogaus privatumą uždrausdamos prieigą prie aparatinės sistemos.

Atliekant šį tyrimą, lygintos geriausios programėlės, esančios „Google Play“ oficialioje *Android* programėlių duomenų bazėje. Visos ištirtos populiariausios ir geriausius įvertinimus turinčios programėlės naudoja tą patį metodą blokuoti kamerai - „setCameraDisabled()“ metodą. Šis metodas populiarius, nes blokuodamas kamerą, jis veikia neapkraudamas procesoriaus. Šiam metodui veikti reikia programėlei suteikti administratoriaus teises, nes „setCameraDisabled()“ metodas yra priskiriamas „DeviceAdminReceiver“ klasei.

Kitaip nei kamera, garso blokavimui yra naudojamas metodas „setMicrophoneMute()“, kuris išjungia mikrofono naudojimą ir tokiu būdu prisijungia mikrofono procesą ir neleidžia kitoms programėlėms jo naudoti, tačiau jį nesunkiai galima vėl įjungti uždarius veikiančia programą. Metodas, kuris yra nagrinėjamas šiame darbe, skiriasi nuo tų, kurie yra naudojami visiems prieinamose apsauginėse programėlėse. Programą, startavus su administratoriaus vartotoju, gali išjungti tik tas pats administratoriaus vartotojas, kuriuo buvo startuota programa, todėl kitos programos jos išjungti negalės. Garsui užblokuoti bus naudojamas buferinio proceso priverstinis uždarymas vos tik jam atsiradus.

Žemiau pateiktoje lentelėje yra atliekamas populiariausių programų, surūšiavus jas pagal atsiuntimų ir įvertinimo skaičių, palyginamasis ir tyrimas. Pirmu numeriu pažymėtoje eilutėje yra pateikiami duomenys siūlomo prototipo, kurio kūrimas yra aprašomas programos kūrimo dalyje, likusiose aprašomos 8 lentelės eilutėse yra išvardinti pagrindiniai funkciniai privalumai populiariausių programų.

Tyrimui atlikti buvo naudojami keli metodai, tai įdiegtos programos įvertinimas tikrinant visus programos funkcionalumus ir antras, tai programos išskleidimas ir išskaidymas į pirminį programinį kodą. Išskaidyta programa į pirminį kodą gali pasakyti daugiau nei tik grafinė sąsaja. Programos išskleidimui naudojama programa „Dex2Jar“. Ši programa skirta įdiegtam programos paketui išmaniajame telefone perversti į programinį kodą. Šiam veiksmui įvykdyti atliekami tokie žingsniai:

- Telefone yra surandama norima programėlė „/data/app“ kataloge. Tada visą programos paketą, kurio pabaiga yra „.apk“, atsiunčiame į kompiuterį ir pervadiname į „.zip“ formatą.

- Pakeitus į „.zip“ formatą, paketą galima išpakuoti kaip suarchyvuotą failą. „Dex2Jar“ programą perrašyti į išpakuotą katalogą.

- Cmd.exe komandinėje eilutėje parašyti „dex2jar classes.dex“ komandą. Programa išpakuoja ir pateikia „classes.clas“. Išskleistą programą pasiverčia į „.jar“ formatą.

- Su „JD-GUI“ programa atidaromas „.jar“ sukurtas failas. Šios programos pagalba galima naršyti po naujai sukurtą biblioteka, kurioje matosi pirminis programos kodas, o „.class“ failai išdėlioti lengvai skaitoma tvarka.

8.1. *Android* programų tyrimo rezultatai ir jų paaiškinimas

Atlikus veiksmus aukščiau išvardinta seka, galima sužinoti, kokius veiksmus atlieka programa. Pasinaudojant šiuo metodu buvo išnagrinėtos žemiau lentelėje pateiktos programos.

Funkcionalumo palyginimas su esamomis rinkoje programa ir aprašoma darbe.

Nr.	Logotipas	Pavadinimas	Garso blokavimas	Vaizdo blokavimas	Pranešimu rodymas	Registrų rinkimas	Išskirti neblokuojamas programas
1.		Camera Block - Spyware protect Microphone Block -Anti malware	TAIP	TAIP	TAIP	NE	NE
2.		Camera Blocker - Anti Spyware	NE	NE	TAIP	NE	NE
3.		D-Vasive Anti-Spy	TAIP	TAIP	TAIP	TAIP	NE
4.		Cameraless - camera block	NE	TAIP	TAIP	NE	NE
5.		Camera Guard™ Blocker FREE	NE	TAIP	TAIP	NE	NE
6.		Cam Locker	NE	TAIP	TAIP	NE	NE
7.		Camera Blocker Free	NE	TAIP	NE	NE	NE
8.		Camera Block -Protect your cam	NE	TAIP	TAIP	TAIP	NE

10 Pav. Analogų palyginimas

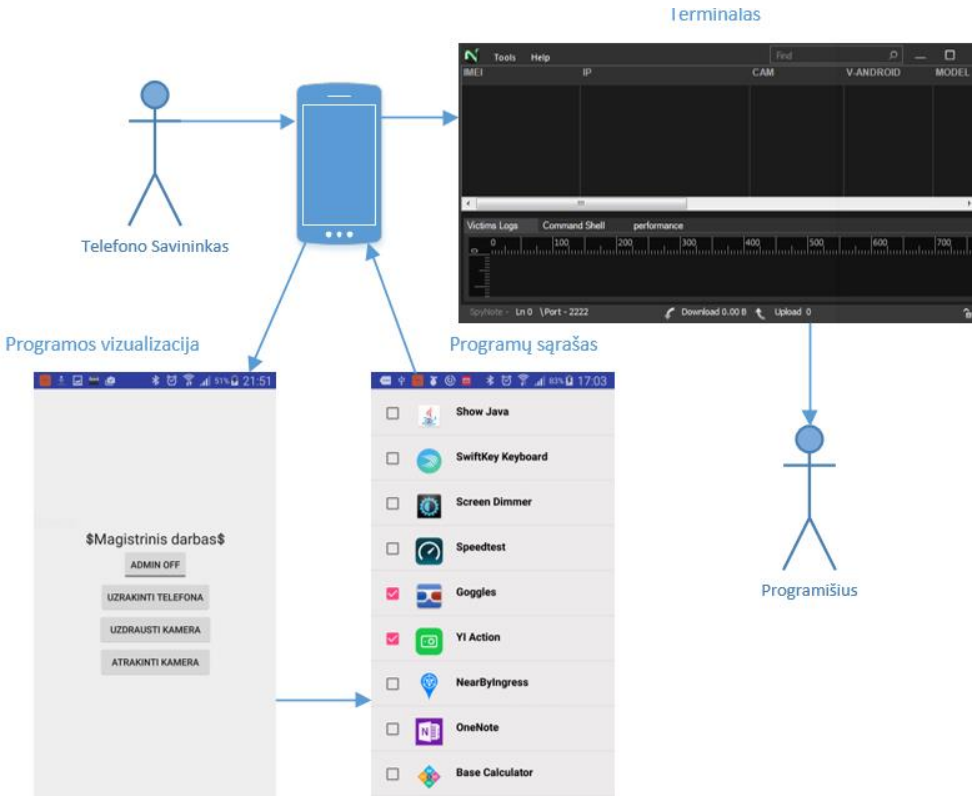
Iš atliktų tyrimo rezultatų matyti, jog visos programėlės tarpusavyje yra skirtingos ir turi skirtingus funkcionalumus, tačiau jos visos turi vieną bendrą bruožą - negali išskirti, kurias programas blokuoti, o kokių - ne. Telefono savininkui yra sukeliami papildomi sunkumai naudojantis telefonu, atsiradę dėl supaprastinto programų veikimo principo: programos blokuoja vaizdą ir garsą visoms programoms, nepriklausomai nuo jų panaudos svarbos vartotojui. Norint greitai užfiksuoti įvykį vaizdo kamera, tai padaryti yra neįmanoma, nes norint įjungti kamerą, pirma reikia išjungti kameros blokavimą ir tik tada yra leidžiama vėl daryti fotografijas ar filmuoti. Ta pati problema yra ir su mikrofonu: sulaukus skambučio, pašnekovas nieko negirdės tol, kol nebus išjungtas mikrofono blokavimas.

Pirmu numeriu pažymėtoje eilutėje yra pateikiami atsakymai, kokias funkcines galimybes yra siūloma įdiegti. Autoriui padarė išvadą, jog rinkoje yra reikalingas programos patobulinimas, kuris suteiktų galimybę vartotojui išsirinkti, kurių programų vartotojas norėtų neblokuoti. Toliau darbe yra pasirenkamas saugumo metodas, kuris blokuoja viską, tačiau leidžia išsirinkti, ko neblokuoti. Šis atvirkštinio saugumo metodas užtikrina maksimalų saugumą, nes nereikia išrinkinėti potencialiai pavojingas programas ir jas užblokuoti, o paprasčiau išrinkti, ko neblokuoti.

8.2. *Aandroid* programos realizavimo metodai ir reikalavimai

Android sistemos komponentai yra paleidžiami ant Linux branduolio, todėl *Android* sistema iš esmės yra procesų rinkinys įskaitant „daemon“, „mediaserver“ , „systemserver „ ir pačios *Android* programos procesus. Rašant *Android* programą galima naudoti numatytas *Android* klases į kurias yra kreipiamasi per „set“ kreipinį į metodą arba perrašant visą klasę, taip sutrumpinant ir paliekant tik tai kas bus naudojama, išnaudojant paskutinį metoda ir atsiranda kenkėjiškos programos kurių negali aptikti antivirusinės programos, nes tos kenkėjiškos programėlės nesikreipia per pagrindinę *Android* API biblioteką.

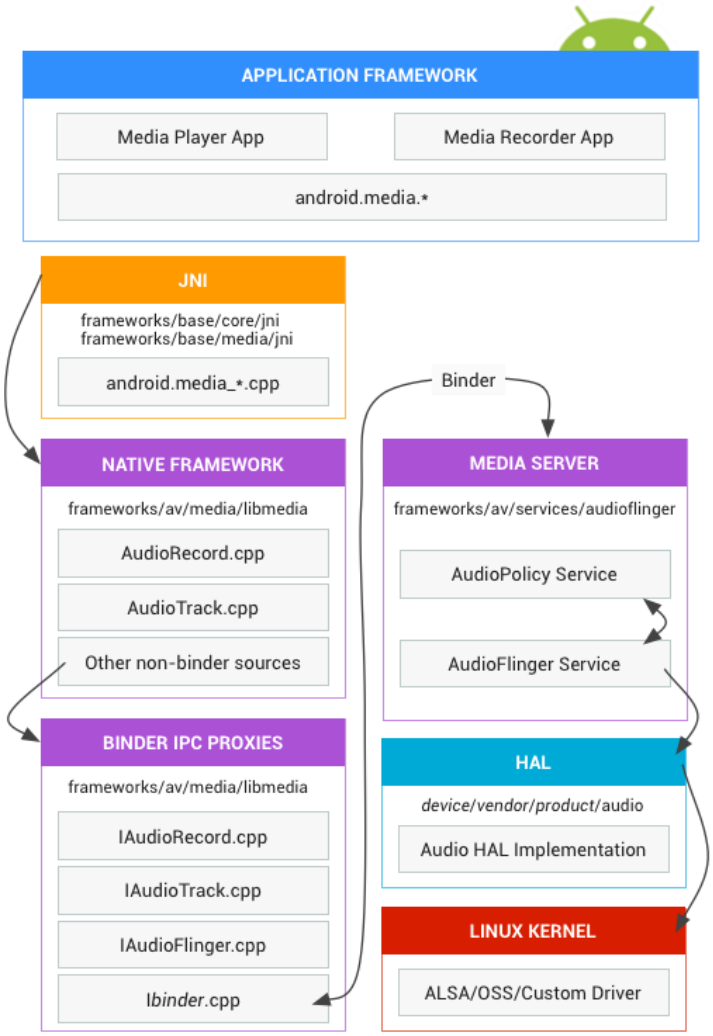
Šiame darbe bus aprašomas programos veikimas tarp skirtingų sistemos lygių, bibliotekų ir aparatinės įrangos. Programos prototipui įgyvendinti bus naudojami šios pagrindinės klasės: „MediaServer“ tai *Android* klasė kuri atsakinga už vaizdo valdymą telefone, nuo pat aparatinės įrangos iki vaizdo išvesties į ekraną. „AudioServer“ tai klasė atsakinga už garso valdymą kuri taip pat kaip ir „MediaServer“ klasė, apjungia visus programos lygių sluoksnius, nuo pat mikrofono aparatinės įrangos iki vartotojo išvesties į ekraną arba garsiakalbį.



11 Pav. Prototipo veikimo vizija.

Įsivaizduojama prototipo vizualizacija susideda iš kelių pagrindinių elementų tokių kaip telefono, apsauginės programos, kenkėjiškos programos. Kenkėjiškų programų kūrėjai darosi vis išradingesni ir vis dažniau išnaudoja dar neištaisytus *Android* programinio kodo trūkumus arba prisidengia kitomis programomis taip patekdamos į telefonus. Kuriamos programos tikslas yra apsaugoti telefoną nuo didžiausią grėsme privatumui keliančių metodų, tai yra kamera ir mikrofonas. Kuriamos programos prototipas turi apsaugoti kamerą ir mikrofoną užblokuodami bet kokį bandymą jais pasinaudoti, nes metodas viską uždraudžiant yra saugesnis nei metodas bandant surasti kenkėjišką programą ir bandyti stabdyti jos veiksmus. Vartotojas iš įdiegtų programų sąrašo gali pasirinkti programas kurių neblokuoti tam, kad telefonas neprarastų savo pagrindinės paskirties - skambinti. Užblokavus kamerą arba mikrofoną programa pradeda laukti kol bent viena iš pasirinktų programų pradės veikti ir atsiras procesų sąraše, radus tokį procesą programa automatiškai išjungia vaizdo ir garso blokavimus, o joms užsidarius blokavimai įsijungs automatiškai.

8.3. **Garso operacinė įranga**



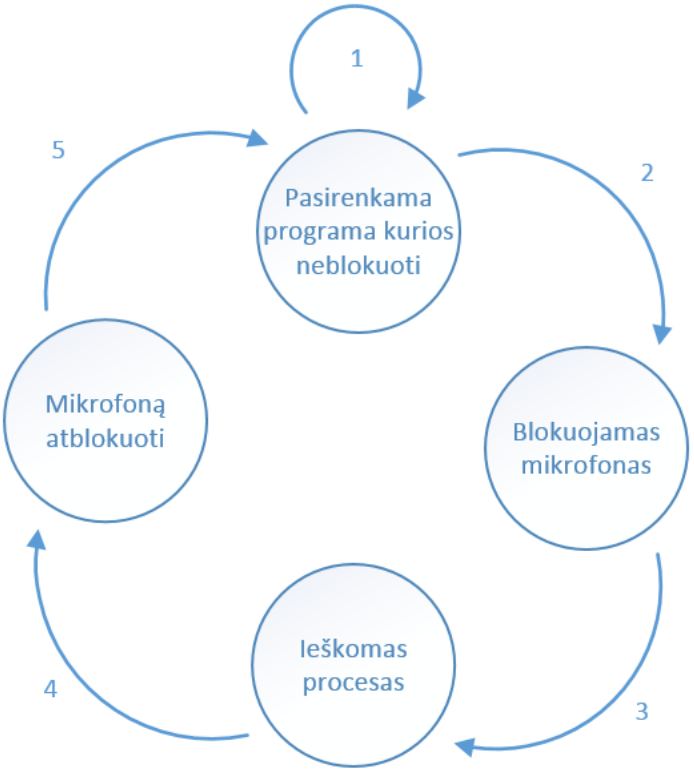
12 Pav. Garso operacinės sistemos sluoksniai.

Android operacinės sistemos sluoksniai jungiasi prie aukščiausio sluoksnio garso specifinio struktūrinio rėmo „*Android*.media“, kuris yra pagrindinė sąsaja atsakinga už garso ir vaizdo programinės ir operacinės įrangos valdymą. 12 Pav. vaizduoja kaip sąveikauja garso operacinės sistemos sluoksniai su operacine, technine įranga bei vartotojo grafine sąsaja. Žemiau yra detaliai pasakojama kiekvienas operacinės sistemos sluoksnis, jo paskirtis ir sąveika tarp operacinės, techninės įrangos ir grafine išvestimi.

- Programos struktūra – programos struktūrai priklauso programos kodas kuris naudoja „*Android*.media“ klasę bendrauti su garso technine įranga. Pati operacinė sistema iškviečia atitinkamą JNI (iš 12 paveikslėlio) klasę kuri apjungia programuotojo parašytą programinį kodą ir operacinės sistemos programinės įrangos JAVA klases, taip įvyksta techninės įrangos sužadinimas.
- JNI (angl. Java Native Interface) - Java vietinis kodas siejasi su „*Android*.media“ klase ir kviečia žemesnio lygio kodą pasiekti garso techninę įrangą.

- Vietinė sąsaja (angl. *Native Interface*) – suteikia tokia pat prieigą kaip *Android*.media paketas kuris kviečia jungiamąjį IPC (angl. interprocessor communication) kad pasiektų garso specifinius servisus iš garso valdymo serverio.
- Jungiamasis įgaliojimas (angl. *Binder proxy*) – palengvina bendravimą už proceso ribų. Įgaliojimai yra laikomi frameworks/av/media/libmedi ir prasideda su raide „I“.
- Garso ir vaizdo serveris (angl. *Media Server*) – turi garso servisų programinį kodą kuris bendrauja su HAL (angl. *Hardware Abstraction Layer*) media serveris yra frameworks/av/services/audioflinger.
- HAL – apibrėžia standartinę struktūrą į kurią kreipiasi garso tvarkymo servisas ir HAL yra būtinas, kad garso operacinė įranga veiktų kaip numatyta. Visa HAL struktūra yra laikoma hardware/libhardware/include/*hardware.*
- *Kernel valdikliai (angl. Kernel* driver) – garso valdikliai glaudžiai susiję su aparatinės įranga ir HAL interpretatoriumi. Šiems valdikliams priklausomai nuo gamintojo gali būti naudojami ir kiti valdikliai tokie kaip: pažangi Linux garso architektūra (angl. Advanced Linux Sound Architecture (ALSA)) arba atvira garso sistema (OSS) (angl. *Open Sound System*) [20].

Planuojamo garso užblokavimo prototipo įgyvendinimo būsenų diagrama.

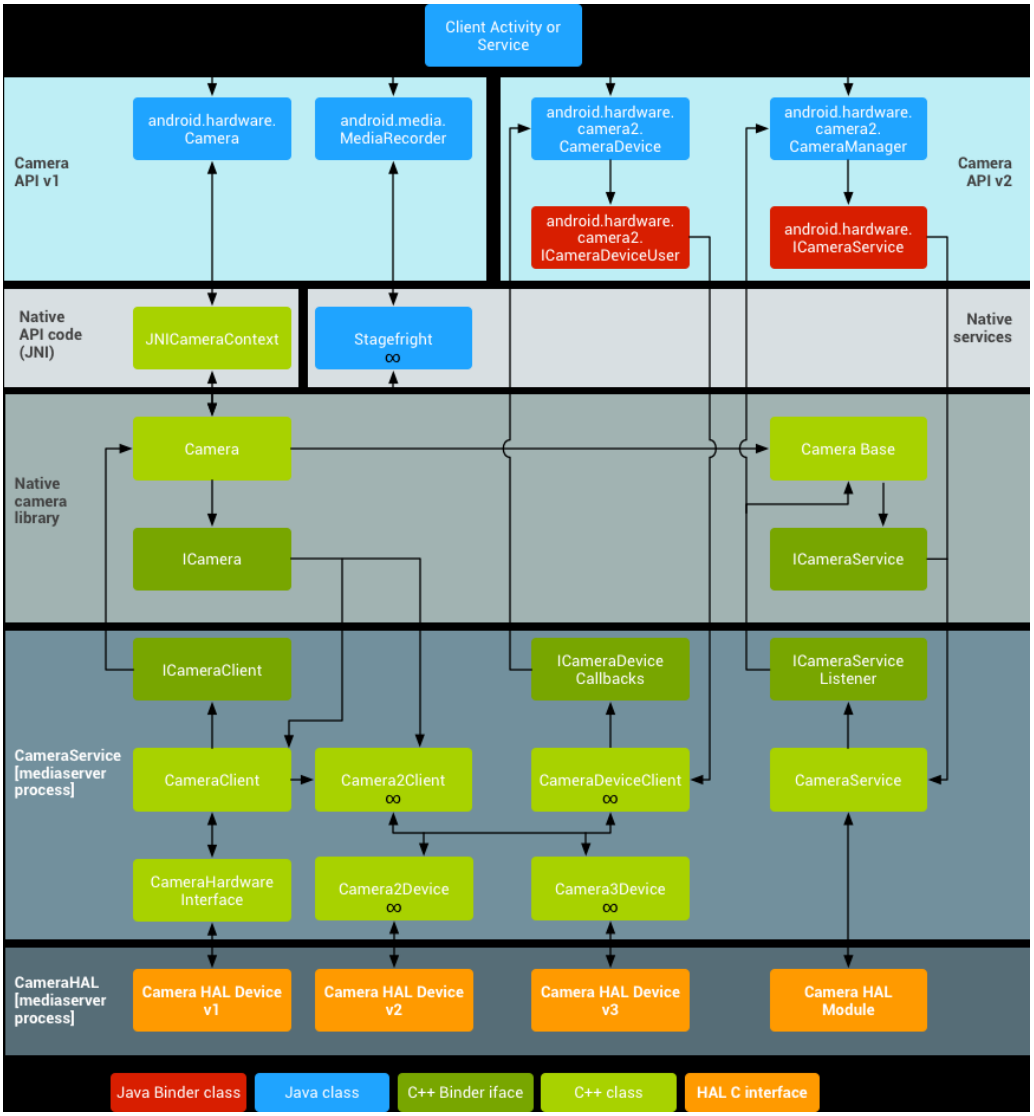


13 Pav. Garso būsenų diagrama.

Garso blokavimui planuojamas prototipo modelis, kuris turėtų būti sudarytas iš kelių procesų, tokių kaip programos pasirinkimas iš įdiegtų sisteminių arba vartotojo pasirinktų programų, kurios

vėliau galėtų būti išskiriamos kaip neblokuojamos programos dėl jų funkcionalumo ir dažnos panaudos. Pirmu numeriu pažymėtoje proceso dalyje programų pasirinkimas naudojamas kaip grįžtamas procesas kurį galima pakartoti arba papildyti kitomis programomis kurias norima eliminuoti iš blokuojamų sąrašo. Antru numeriu pažymėta proceso eiga blokuoja visas programas išskyrus tai atvejais kada „ieškomas procesas“ metu yra randama programa kuri yra įtraukta į neblokuojamų sąrašą. Ketvirto etapo metu yra sužadinamas mikrofono atblokavimo procesas ir taip ciklas kartojasi tok kol programa veikia [30].

8.4. **Vaizdo operacinė sistema.**



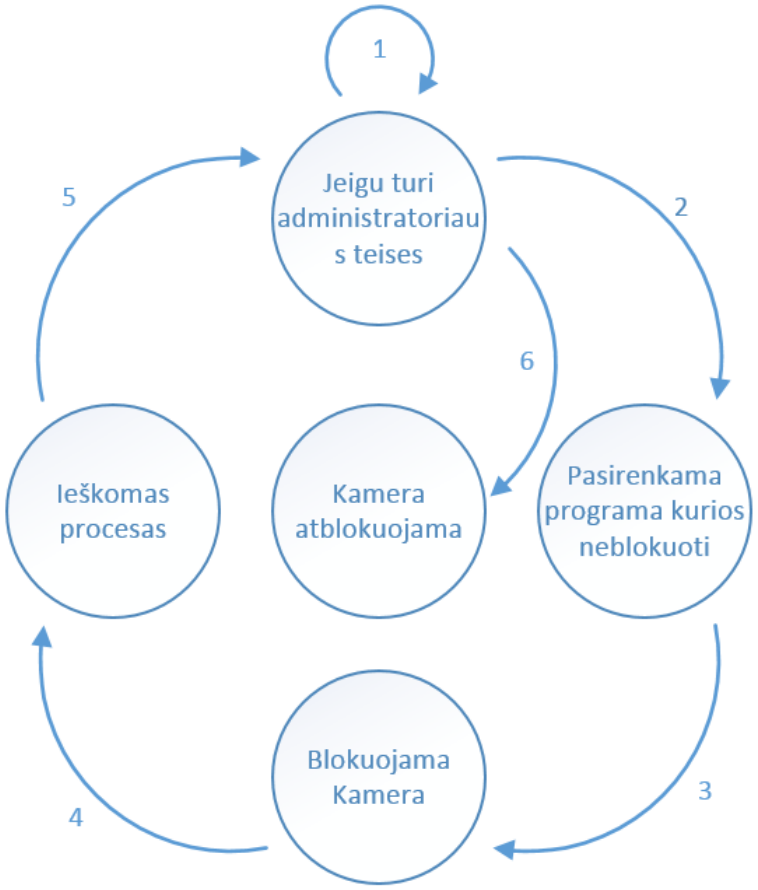
14 Pav. Kameros architektūra.

Aukščiau pavaizduotame paveikslėlyje yra parodomi visi operacinės ir aparatinės sluoksniai ir kaip veikia „Android.hardware“ API. HAL Kameros aparatinės įrangos (angl. *Hardware Abstraction*

Layer) lygiai kurie reikalingi veikti kameros aparatinei įrangai. I kameros posistemę įeina ir juostiniai komponentai, tuo tarpu kameros aparatinės įranga suteikia sąsają.

- Programinės įrangos sąsaja – tai yra programėlės kodas kurį rašo programos dizaineris kuris išnaudoja Android.hardware.Camera programos sąsają, kad galėtų pasiekti kameros aparatinę įrangą, atitinkamai parašytas kodas iškviečia „JNI glue“ Java klasę, kad pasiektų pirminį kodą kuris valdo kameros aparatinę įrangą.
- JNI – tai yra Java vietinė sąsaja (angl. *Java native interface*) ir ji yra reikalinga, kad būtų galima pasiekti Android.hardware.Camera objektą esantį frameworks/base/core/jni/*Android*_hardware_Camera.cpp. Šis kodas šaukia žemesnio lygio vietinį programinį kodą, kad gauti prieigą į fizinę kamerą ir gražinti duomenis atgal į Android.hardware.Camera objektą pagrindinės struktūros lygyje.
- Vietinė struktūra (angl. *Native Framework*) – yra laikoma frameworks/av/camera/Camera.cpp klasėje kuri suteikia atitikmenį vietinei Android.hardware.Camera klasei. Ši klasė iškviečia IPC sujungimo įgaliojimą, kad gautų leidimą pasiekti kameros komponentams.
- Jungiamieji IPC įgaliojimai – taip pat kaip ir garsui leidžia bendrauti už proceso ribų. Viso yra trys jungiamosios klasės ir jos laikomos frameworks/av/camera kataloguose komunikuoti su kameros servisais. ICameraService yra sąsaja su kameros aparatine įranga, o ICamera yra sąsaja su kažkuria iš esamų kamerų, norint gražinti atsakymą iš kameros į kuriamą programą reikia naudoti ICameraClient metodą.
- Kameros valdikliai (angl. *Camera Services*) – yra laikomos frameworks/av/services/camera/libcameraservice/CameraService.cpp ir tai yra kodas kuris tiesiogiai siejasi su HAL (angl. *Hardware Abstraction Layer*)
- HAL – tiesiogiai bendrauja su kameros aparatine įranga ir ji yra būtina, kad kamera aparatinė įranga veiktų kaip priklauso.
- Kernel valdikliai (angl. *Kernel Driver*) – tai kameros sąsaja su pačia aparatine kamera sujungta kartu su HAL. Kamera ir valdikliai privalo palaikyti YV12 ir NV21 paveiksliukų formatus, kad butų galima peržiūrėti padarytas nuotraukas ekrane. YV12 ir NV21 tai nuotraukų formatai kuriu atspalviai yra dalinami i 2x2 pikselių laukelius, šio vaizdo formato laikmenos posistemė yra MPEG-4 [21].

Planuojamo prototipo vaizdo blokavimų būsenų diagramos vizualizacija.



15 Pav. Kameros blokavimo būsenų diagrama

Vaizdo blokavimui planuojami 5 procesai kurie atliktų pagrindinius veiksmus bandant uždrausti prieigą prie kameros. Pirmoje būsenoje procesas kreipiasi pats į save tikrindamas ar administratoriaus vartotojas vis dar yra įjungtas. Kameros blokavimui nėra būtina pasirinkti kurią programą norima įtraukti į neblokuojamų sąrašą. Šiame pavyzdyje programos išskyrimas bus įtraukiamas antruoju proceso metu. Programos pasirinkimas atliekamas paspaudus sąraše norimą programą piktogramą iš sąrašo. Kameros blokavimui naudojamas administratoriaus klasės metodas kuris turi aukštesnes nei „VideoManager“ klasė. Trečiuoju procesu kameros naudojimas yra visiškai uždraustas visoms programoms. Ketvirto proceso etapu vyksta esamų procesų ieškojimas tol kol bet kuris rastas procesas atitinka ieškomąjį. Radus procesą programa vėl kreipiasi į administratoriaus klasę patikrinti ar administratoriaus teisės vis dar yra aktyvios. Šeštuoju etapu kameros blokavimas yra išjungiamas, visos programos gali ją naudoti be papildomų suvaržymų. Ciklas yra kartojamas tok kol visi blokavimai yra neišjungiami, programos uždarymas nesustabdo kameros blokavimo.

8.5. Sisteminiai leidimai *Android* operacinėje sistemoje.

Android yra privilegijomis atskirta operacinė sistema, kurioje kiekviena programa veikia su jai priskirtais specifiniais sistemos ypatumais (Linux vartotojo vardu ir grupės vardu). Dalis operacinės sistemos yra išskirta į išskirtines privilegija. Linux/*Android* operacinė sistema išskiria programėlė nuo sistemos, o papildomos menkiausios saugumo charakteristikos suteikiamos per leidimą mechanizmą, kurios uždraudžia naudoti specifines operacijas, kurias paprasti procesai gali naudoti be suvaržymų ir per URI leidimus suteikti prieigą prie specifinių duomenų dalių.

Pagrindinis *Android* saugumo sprendimo dizainas yra uždrausti programai turėti standartinių prieigų kurios galėtų pakenkti kitoms programoms telefone ar pačiai operacinei sistemai. Į draudimą įeina ir naudotojo ar kitos programos duomenų nuskaitymą, rašymą, interneto ryšio naudojimą, kamerą ir neleisti telefonui užmigti. Kadangi visos programos yra paleidžiamos atskiroje smėlio dėžėje, tai kiekvienai iš naujai įdiegtų programų reikia patvirtinti leidimų sąrašą.

Visos *Android* programos turi sufiksą (.apk) taip galima atpažinti ar tai yra įdiegimui paruoštas paketas, jis turi būti pasirašytas kūrėjo privačiuoju raktu, tačiau nėra reikalaujama , kad privatusis raktas būtų pasirašytas sertifikatų centro, o tai padeda operacinei sistemai atskirti programos autorius ir juos priimti arba ne.

Kiekvieno programos įdiegimo metu, *Android* įdiegimo paketui suteikia išskirtinį Linux vartotojo vardą ir šis vartotojo vardas išlieka nepakitęs visą programos gyvavimo laiką OS, ant skirtingų telefonų unikalus vartotojo vardas gali skirtis. Tačiau galima priversti nurodžius „AndroidManifest.xml's manifest“ faile parametrą, kad naudotų tą patį vartotojo vardą uždėjus parametrą „sharedUserId“. Bet kokie kiti programos laikomi duomenys, bus priskirti prie tos programos vartotojo vardo ir turės teisę prie kitu paketų [29] [31].

8.6. Leidimų naudojimas

Programai suteikti leidimą reikia „AndroidManifest.xml's manifest“ faile, kur reikia pridėti vieną ar daugiau <uses-permission> etiketę. Pavyzdžiui žemiau pavaizduotame programiame kode yra privilegija leidžianti stebėti ateinančias žinutes.

```
<manifest xmlns : android = "http://schemas.android.com/apk/res/android"
        package = "com.android.app.myapp" >
        <uses - permission android : name = "android.permission.RECEIVE_SMS" / >
...
</ manifest>
```

Kadangi yra dviejų lygiu leidimai, aukštos arba žemos rizikos tai žemos rizikos programoms, sistema automatiškai suteikia visas teises reikalingas tai programai veikti, o jeigu „manifest.XML“ aprašytos aukštos rizikos privilegijos, kurios gali pakenkti vartotojo privatumui, tada reikalingas jo patvirtinimas, kad sistema leistų programai naudoti prašomus leidimus. Nuo *Android* 6.0 versijos (API lygis 22) ir aukštesnės, programa turi vykdyti `targetSdkVersion` komandą ir tikrinti ar versija yra aukštesnė nei 23, tokiu atveju programa turi tikrinti privilegijų prieinamumą prieš ją įdiegiant arba pradedant veikti. Vartotojas bet kuriuo metu gali pakeisti programai suteiktus leidimus. Jeigu *Android* versija yra 5.1 (API lygis 22) arba žemesnė ir programos `targetSdkVersion` parametras yra 22 arba mažesnis, tada sistema prašo suteikti programai leidimus prieš ją įdiegiant ir vėliau pati *Android* sistema jų keisti nebegali dėl *Android* OS dizaino. Uždrausti programai naudoti tam tikrus leidimus, senesnėse *Android* OS versijose, yra tik programos ištrynimas. Visi programos kreipimaisi į „*Android* Manifest.XML“ failą yra įrašomi į sisteminį žurnalą [29].

8.7. **Didelės ir mažos rizikos leidimai**

Leidimai skirstomi į du rizikos lygius: didelės ir mažos rizikos leidimus. Normalios rizikos leidimai apima sritis, kada programai reikia leidimo pasiekti duomenis ar resursus kurie nėra smėlio dėžėje, tačiau jie turi labai mažą riziką vartotojo privatumui, todėl atskiro privilegijų patvirtinimo nereikia, kad programa funkcionuotų kaip numatyta. Pavyzdžiui, leidimas pakeisti laiko zoną yra mažos rizikos leidimas ir jie turi būti aprašyti Manifest.XML faile, kad sistema suteiktu visas privilegijas be vartotojo sutikimo. Aukštos rizikos privilegijoms yra priskiriami tie leidimai, kurie prašo prieigų prie vartotojo asmeninių resursų arba duomenų, kurie tiesiogiai gali pakenkti vartotojo privatumui.[15].

8.8. **Administratoriaus leidimas**

Prototipo kūrimui, kuris aprašomas „Tyrimas ir prototipo kūrimas“ dalyje yra būtina sąlyga, kad programai būtu suteikta administratoriaus teisė, nes metodai kurie yra iškviečiami *Android* programoje neveikia be administratoriaus teisių. Toliau detaliau panagrinėsime kaip veikia administratoriaus prieiga, kaip jam yra suteikiamos teisės, kokią riziką tai kelia ir kokią naudą tai duoda.

Rašoma programa tampa administratoriaus programa kada yra kreipiamasi į „DeviceAdminPolicy“ klasę, kuri turi tam tikras administracines funkcijas, kuris gali būti labai destruktyvios, tokias kaip: telefono užmigimo būsenos keitimas, slaptažodžio keitimas, ilgalaikės atminties užkodavimo pakeitimai, kameros ir mikrofono blokavimai, gamyklinių parametrų atstatymas. Prieš įjungiant administratoriaus prieigą telefone, į ekraną iššoka pranešimas supažindinantis apie galimus veiksmus po administratoriaus prieigos įjungimo su kuriomis vartotojas privalo sutikti norint suteikti prieigą programėlei naudotis „DeviceAdminPolicy“ *Android* klase. Vartotojui atsisakius patvirtinti ir suteikti programai administratoriaus teises, programa niekur nedingsta, tačiau tampa neaktyvi, todėl yra prarandama daug programos teikiamų funkcionalumų. Norint ištrinti įdiegtą programą pirma reikia išjungti programos administratoriaus registracijos prieigą ir tik tada bus leista ją ištinti.

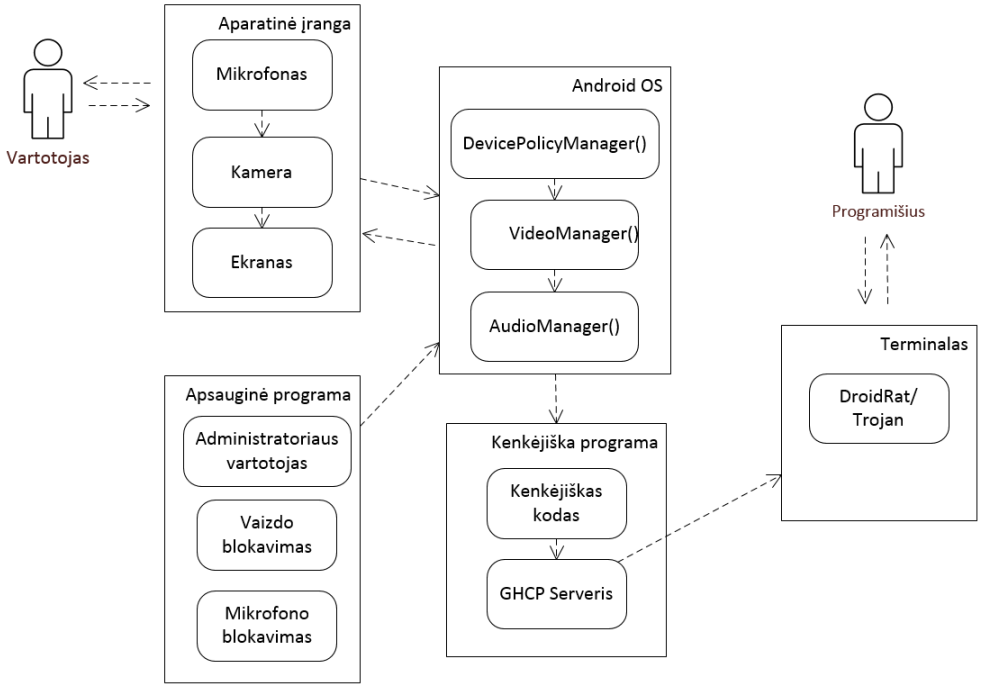
8.9. **Projektavimo išvados**

Atlikus projektavimo tyrimą, buvo pastebėta, kad rinkoje esančių analogų funkcionalumas yra nepakankamas kada norima neprarasti pagrindinės telefono funkcijos, nes prieš naudojantis kamera arba skambinimo funkcija, vaizdo ir garso blokavimo funkcijos turi būti išjungtos, o tai sukelia didelių nepatogumų telefono savininkui. Matomas aiškus poreikis turėti galimybę išsirinkti kurių programų neblokuoti, todėl buvo pasirinktas metodas leidžiantis išskirti specifines programas iš įdiegtųjų programų sąrašo, kurioms atsiradus aktyvių procesų sąraše, kameros ir vaizdo blokavimai būtų automatiškai išjungiami. Tyrimas parodė, jog geriausias metodas blokuoti kamerą yra pasinaudoti administratoriaus prieiga, kurio pagalba galima visiškai užblokuoti kameros įvadą. Garso blokavimui yra pasirenkamas metodas, išnaudojus Android operacinės sistemos trūkumus, kada dvi skirtingos programos negali naudotis mikrofonu vienu metu. Nuolatinis garso įrašinėjimas ir jo užtildymas, padėtų apsaugoti nuo kenkėjiškų programų bandymo pasinaudoti mikrofonu.

9. TYRIMAS IR PTOTOTIPO KŪRIMAS

Trečiame skyriuje aptarėme kokios programos yra siūlomos kaip sprendimo būdas i slapta daromus garso ir vaizdo įrašus. Nekeičiant operacinės sistemos ar kitaip nebandant apeiti esamų saugumo įgyvendinimų tokių kaip vartotojo užrakinimas smėlio dėžėje po žemo lygio sisteminio vartotojo privilegijomis, aptarsime kokį metodą naudosime užtikrinti saugumui nuo piktavališkų programų ar įsilaužėlių. Šiame skyriuje aptarsiu metodus kurie yra mažiausiai įsibraunantys į patį *Android* operacinės sistemos veikimą, taip apsaugant nuo potencialių grėsmių sutekus didesnes vartotojo privilegijas nei tas kurias nustatė pats gamintojas. Šio sprendimo įgyvendinimo metu, programėlei yra prašoma suteikti administratoriaus teises kurios yra aukščiausios kurias gali būti suteikiamos *Android* be papildomų operacinės sistemos pakeitimų ir pasinaudodamas siūlomomis komandomis iš oficialios *Android* kūrėjų svetainės.

Tyrimo metu buvo imituotas viruso paleidimas ant „Samsun Galxy S4 GT-I95505“ serijos telefono kuriame yra įdiegta *Android* 5.0 „Lollipop“ versija. Naudojantis dviem skirtingais „Trojos arklio“ tipo virusais kurie buvo įdiegti paslėpus programinį kodą tarp nekaltai atrodančios programos. Paketo generavimo metu yra pasirenkamas portas ir IP adresas arba DSN, kuris turi būti kompiuterio kurio pagalba bus jungiamasi prie telefono. Terminalo programa veikia kaip nuolatinis klausytojas ir laukia kol įdiegta programa telefone užmegs ryšį su terminalu. Programos funkciniam tyrimui atlinkti bus naudojami šie „Trojos arklio“ tipo virusai: „SpyNone 2.4.1“ ir „DroidJack 4.4“ šių virusų paplitimas ura didžiausias dėl lengvai prieinamo pirminio kodo, o tai suteikia galimybe šiuos virusus išnaudoti platesniam programišių ratui. Virusams veikti yra būtina Java RTE sistema leidžianti naudotis sukurtu terminalu. Virusų veikimo principas yra paremtas serverio naudojimusi, kuris yra sukuriamas asmeniniame kompiuteryje iš kurio yra vykdomos atakos, paleidus programą telefone ji automatiškai prie jo prisijungia, tokiu būdu leisdama perduoti užklausas iš terminalo į telefoną gauti įvairiai informacijai įskaitant tiesioginį garso pasiklausymą arba vaizdo peržiūrą.



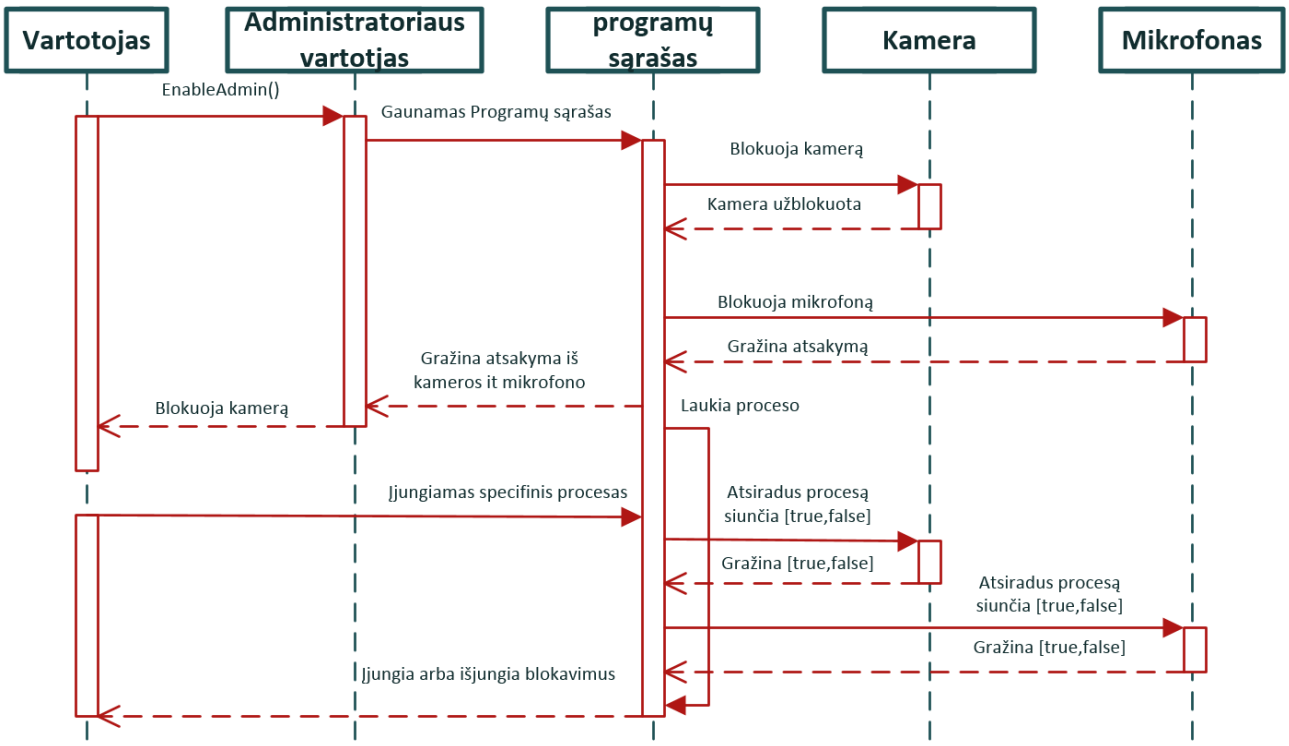
16 Pav. Programos prototipo veikimo procesų diagrama.

Aukščiau esančiame paveikslėlyje yra pavaizduota kaip virusas sąveikauja su *Android* operacine sistema ir kuriamu programos prototipu.

9.1. **Sprendimo metodas**

Android operacinė sistema yra apsaugota ir gauti ROOT teisių į įrenginį tiks suvedus slaptažodį nepavyks, nes aukščiausios teisės kurias gali suteikti kitoms programėlėms yra administratoriaus teisės. Tai yra padaryta tam, kad būtų sumažinta bet kokia galimybė padaryti pokyčius pačioje operacinėje sistemoje, nes taip atsitikus telefonas bus visam laikui pažeidžiamas, nes su tokia programa gali būti įdiegtas ir žalingas kodas, kuris paprastai yra nerandamas nei pačio telefono savininko, nei antivirusinių programėlių. Todėl savo sprendimui įgyvendinti naudosiu pačias aukščiausias teises kurios yra suteikiamos programėlėms įdiegtoms smėlio dėžėje ir tai yra administratoriaus teisės.

Prototipo sekų diagramoje yra patyti pagrindiniai programinio kodo veikimo komponentais pradedant nuo vartotojo veiksmų baigiant pagrindinių veikimo funkcijų įgalinimu. Vartotojui įjungus programą ji įgauna administratoriaus teises kurių „Trojos arklio“ tipo virusai atimti nebegali dėl savo per mažų teisių ir ribotų funkcinių galimybių. Išnaudojant šiuos virusų trūkumus prototipo kūrime mes užtikrinam, kad paleista programa nebus priverstinai uždaryta taip bandant apeiti kameros ir garso blokavimus.



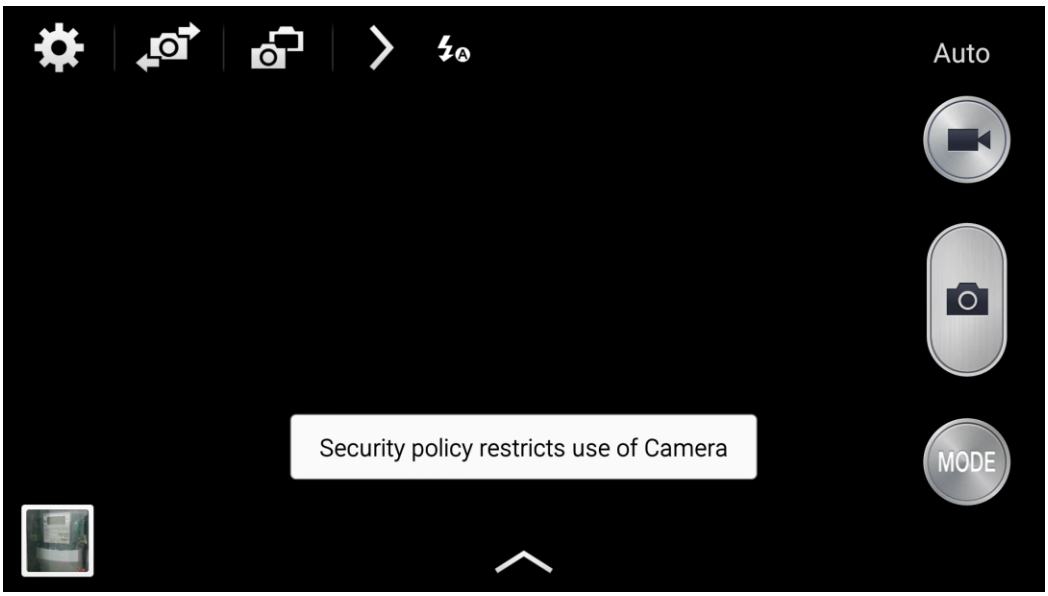
17 Pav. Prototipo veikimo sekų diagrama.

9.2. **Vaizdo kameros blokavimas**

Android operacinė sistema yra labai artima *Linux* OS. Tiek *Android* ar *Linux* visas programa paleidžia jos pačios smėlio dėžėje, jeigu programėlė nebuvo prieš tai įdiegta aukščiausiomis ROOT sisteminio vartotojo teisėmis, tai daryti bet kokius pakeitimus operacinės sistemos kataloguose ar kitose prieš tai įdiegtose programėlėse neturės jokių teisių. Tas pats principas galioja ir *Android* OS tik su dar stipresniais apribojimais ir tai yra prevencija daryti bet kokius pokyčius pačiai operacinei sistemai.

Nuo 14 *Android* platformos versijos (API 14) buvo pridėta papildoma funkcija, kuri leidžia uždrausti kameros naudojimą. Tai buvo padaryta kada buvo sugalvota sukurti nuorodą į kamerą esant užrakintam telefonui, bet naudodamas tą pačią funkciją galimas kameros blokavimas visoms aplikacijoms įdiegtoms telefone. „setCameraDisabled„ tai yra metodas, kuris kreipiasi tiesiai į įrenginį ir valdo visas aparatinės įrangos kameras įdiegtas telefone jas išjungdamas. Naudojant šį metodą vietoje būdo kuri naudoja antivirusinių programų gamintojai, kada yra be perstojo ieškoma aktyvaus proceso kuris iškviečia kamerą, yra sumažinama procesoriaus užimtumas, RAM atminties sunaudojimas programai gyvuoti ir tuo pačiu padidėja baterijos gyvavimo laikas. Kad veiktų „setCameraDisabled„ metodas būtina programai suteikti administratoriaus teises, be jų šis metodas neveiks. Administratoriaus sisteminis vartotojas privalo turėti įjungtą USES_POLICY_DISABLE_CAMERA metodą kitaip programa parodys klaidą apie saugumo pažeidimo pranešimą.

Užblokavus kamerą ekrane pasirodo toks pranešimas ir kamera vaizdo nebetransliuoja.



18 Pav. Užblokuotos kameros vaizdas.

Kitaip nei vaizdo įrašo blokavimas, su garsu nėra jokių metodu kurie padėtų tiesiog užrausti garso įrašinėjimą todėl tai šiek tiek labiau apsunkina užblokuoti garso įrašo darymą. Operacinės sistemos moduliai garso nuskaitymo metu signalą paima iš aparatinės įrangos ir įrašo į buferį. Programėlės naudodamos “AudioRecord“ metodą gali nuskaityti garsą įrašytą į buferį ir jį perduoti

vartotojui per aplikaciją. Garso įrašo buferio dydis priklauso nuo operacinės sistemos konfigūracijų, ir buferiui užsipildžius pradedama perrašinėti nuo seniausio įrašo.

9.3. **Garso uždraudimas.**

Garso uždraudimui taip pat kaip ir vaizdo yra naudojami *Android* operacinės sistemos numatyti Java „AudioManager“ klasės „setMikrophoneMute()“ metodai. Kitaip nei vaizdo blokavimui „AudioManager“ klasė skirta blokuoti mikrofono įvadą tik toje pačioje programoje kurioje jį ir iškviečiame. Tai apeiti galima nuolat įrašinėjant garsą su nutildytu mikrofono, vykstant šiam procesui jokia kita programa mikrofono naudoti nebegalės.

Sėkmingam garso blokavimui nuolatinis garso išjungimas yra būtinas norint apeiti *Android* „AudioManager“ klasės veikimą, nes kiekviena įdiegta programa atstato „AudioManager“ klasės parametrus į pradinę būseną, kas automatiškai išjungia bet kokius mikrofono blokavimus. Išnaudojus *Android* operacinės sistemos trūkumus kada dvi skirtingos programos negali naudotis mikrofonu vienu metu dėl užimto garso įrašinėjimo buferio. Mikrofono užtildymui yra naudojamas setMicrophoneMute() metodas kuris nuolat paleidžiamas cikle, kad kenkėjiškos programos negalėtų mikrofono vėl įjungti. Įgyvendinus šiuos metodus kitos programos gražina klaidą, kad mikrofonas jau įrašinėja.

19 Pav. Atsiradusi žinutė užrakinus mikrofoną.

9.4. **Įdiegtų programų sąrašas ir jų išskyrimas**

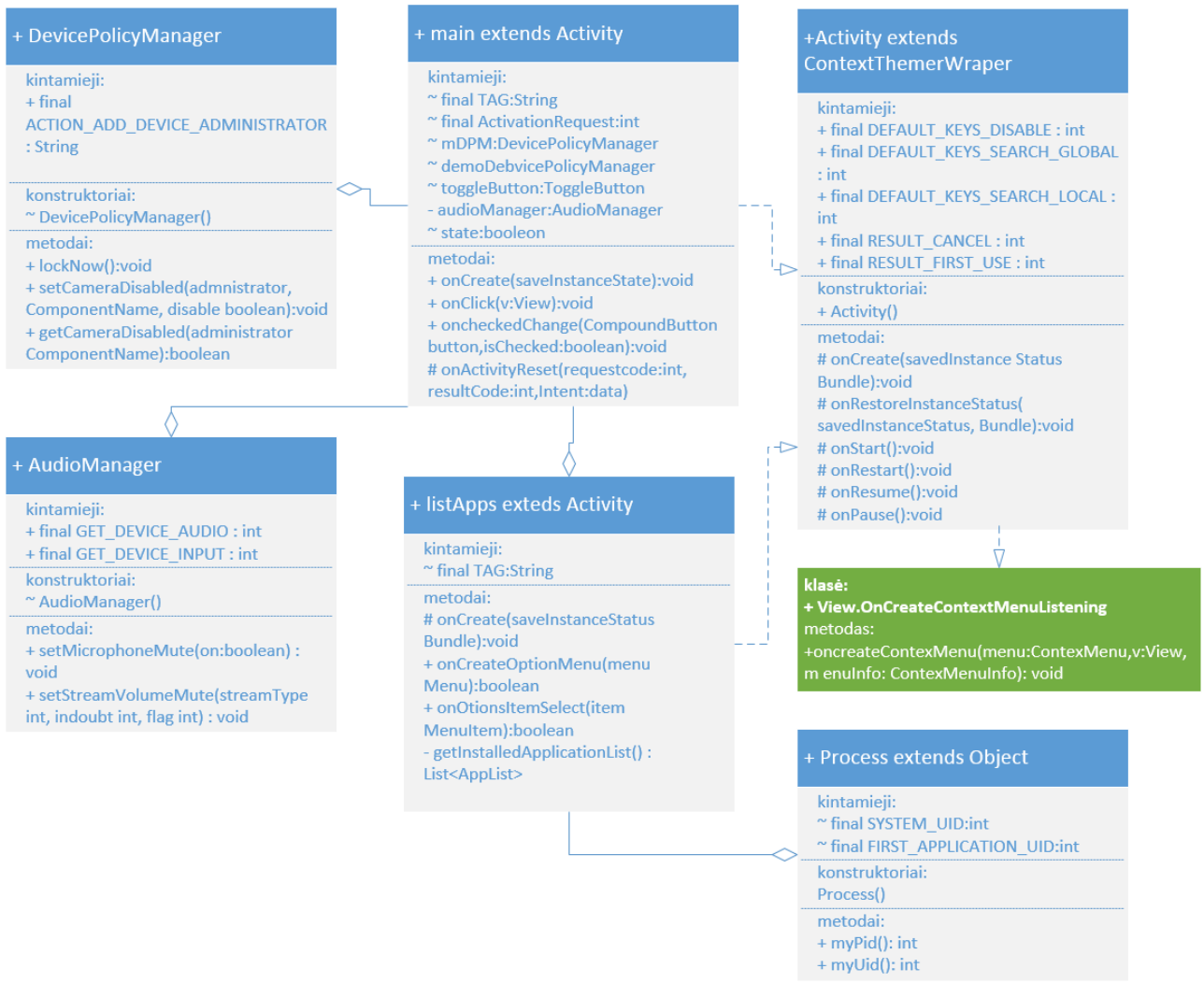
Kadangi daug programų esančių rinkoje jau siūlo panašius sprendimus garso ir vaizdo blokavimui, tačiau nei viena programa nesiūlo palengvinimo, kuris padėtų telefono savininkui laisvai išnaudoti visus išmanaus telefono privalumus ir tuo pat metu jaustis saugiai žinant, kad jo privatus gyvenimas nebus atskleistas. Tai įgyvendinti reikia turėti galimybę automatiškai išjungti blokavimus, jeigu kažkuri specifinė programa yra paleista, pavyzdžiui standartinė telefono kamera arba yra atliekamas skambutis, nes niekas nenorėtų galvoti apie tai kad prieš skambinant arba atsiliepiant į skambutį, mikrofono blokavimas turi būti išjungtas, kad pašnekovas girdėtų ką sako telefono savininkas.

Šiam sprendimui įgyvendinti yra pasirenkami metodai kurie parodo įdiegtų programų sąrašą, leidžia juos pasirinkti ir ieško aktyvaus proceso iš procesų sąrašo. Programoms pasirinkti kuriamas naujas langas ir jam priskiriama nauja „Activity“ klasė, ten yra aprašoma „ListView“ klasė kurios pagalba yra gaunamas visas įdiegtų programų sąrašas. Pasirinktos programos identifikavimo numeris yra persiunčiamos į pagrindinę „Main“ klasę kur programos identifikavimo numerio atitikmuo yra ieškomas tarp esamų procesų getProcessList() metodo pagalba. Radus proceso atitikmenį visi blokavimai yra išjungiami tol kol aktyvus procesas egzistuoja.

9.5. **Programos kūrimas**

Programos kūrimui yra naudojama, specialiai *Android* programoms kurti skirtas įrankis pavadinimu „Android Studio“, kuris yra nemokamas ir visiems laisvai prieinamas. Šio įrankio pagalba yra sukuriamas vaizdas ir programos turinys. Grafinės sąsajos kūrimas yra aprašomas XML programavimo kalbos pagalba, ten apsirašo kiekvienas atskiras programėlės langas ir funkciniai mygtukai, kuriu identifikacinis numeris yra perduodamas pagrindinei „Main“ klasei, kur ir apsirašo visas programos veikimas. Programai veikti reikalingos administratoriaus teisės, kurios yra suteikiamos per „DevicePolicyManager“ klasę. Visas programos veikimas yra pavaizduojamas klasių diagrama.

Pagrindinės klasės išvardintos žemiau esančiame paveikslėlyje yra būtinos tinkamam programos funkcionavimui. Kiekviena klasė turi metodus kurie yra aprašomi atlikti tam tikriems veiksmas. Klasės gali būti perrašytos naudojant „@override“ komandą, o tai leidžia sumažinti programos dydi ir supaprastinti kodo rašymą, sumažinant metodų skaičių kurie yra nenaudojami.



20 Pav. Programos prototipo klasių diagrama.

9.6. **Realizacijos išvados**

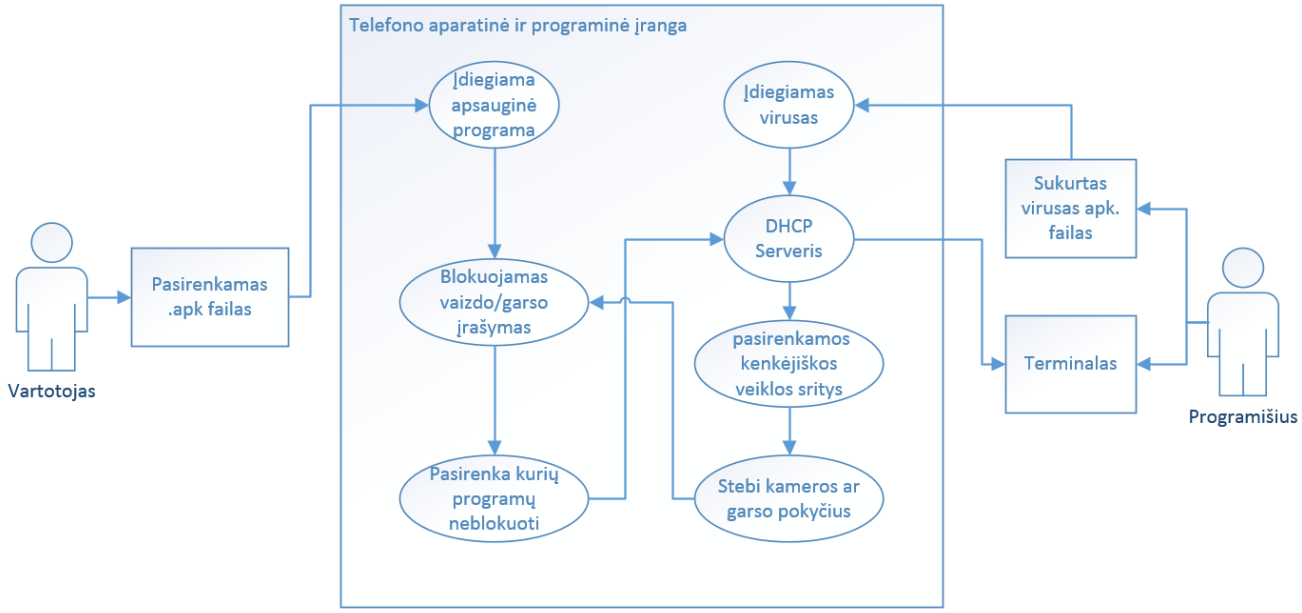
Realizavus prototipą buvo pastebėta kad, kameros ir garso blokavimas veikia visoms nekenkėjiškoms programoms. Kada paleidžiamos telefone esančios programos, kurios naudojasi garsu arba kamera, jos gražina skirtingas klaidos žinutes apie nepavykusi bandymą įrašinėti vaizdą ir garsą. Užblokavus kamerą buvo pastebėta jog yra gražinama saugumo pažeidimo žinutė į išvesties ekraną pranešanti, kad kameros naudojimas yra uždraustas saugumo politikos. Užblokavus garso įvestį, paleistos programos gražina klaidos žinutę apie nepavykusį bandymą įrašyti garso, nes mikrofonas jau yra naudojamas kitose programėlėse. Programos išskyrimui ištestuoti buvo pasirinkta standartinė Android operacinės sistemos programa pavadinimu „Camera“. Jai atsirandant aktyvių procesų sąraše, kameros ir garso blokavimas buvo sėkmingai išjungtas, kas parodo jog sukurtas prototipas atlieka projektuojamas pagrindines funkcijas.

10. EKSPERIMENTAS NAUDOJANT KENKĖJIŠKAS PROGRAMAS

Prieš tai aptartame *Android* programėlių tyrimo ir realizavimo dalyje, buvo atliktas prototipo bandymas su gamyklinėmis *Android* operacinės sistemos programomis, kurios buvo sėkmingai užblokuotos pasinaudojus sukurto prototipo programa. Šiame skyriuje bus atliktas eksperimentas parodantis kaip sukurtas prototipas veikia su kenkėjiškomis programomis. Bus ištestuojami trys populiariausi Trojos arklio tipo virusai, o tyrimų rezultatai aprašomi išvadose.

10.1. Tyrimas naudojant „Trojos arklio“ tipo žalingas programas

Remiantis [25] ir [26] šaltiniais *Android* „Trojos arklio“ tipo virusų yra apie 27 proc. visų viešai prieinamų programų „Google Play“ parduotuvėje, tai leidžia įsivaizduoti jų paplitimą. Tai yra nesudėtingo turinio ir atviro kodo virusai kurie prieinami profesionalui ir pradedančiajam, todėl apsaugoti nuo tokio tipo kenkėjiškų programų yra geras rezultatas. Remiantis [25] šaltiniu 2016 metais buvo aptikta apie 28 neoficialios organizacijos kurios užsiimdavo kenkėjiško turinio programų kūrimu. Žemiau esančiame paveikslėlyje yra pavaizduotas planas kaip kuriamas prototipas sąveikauja su kenkėjiška programa [31].



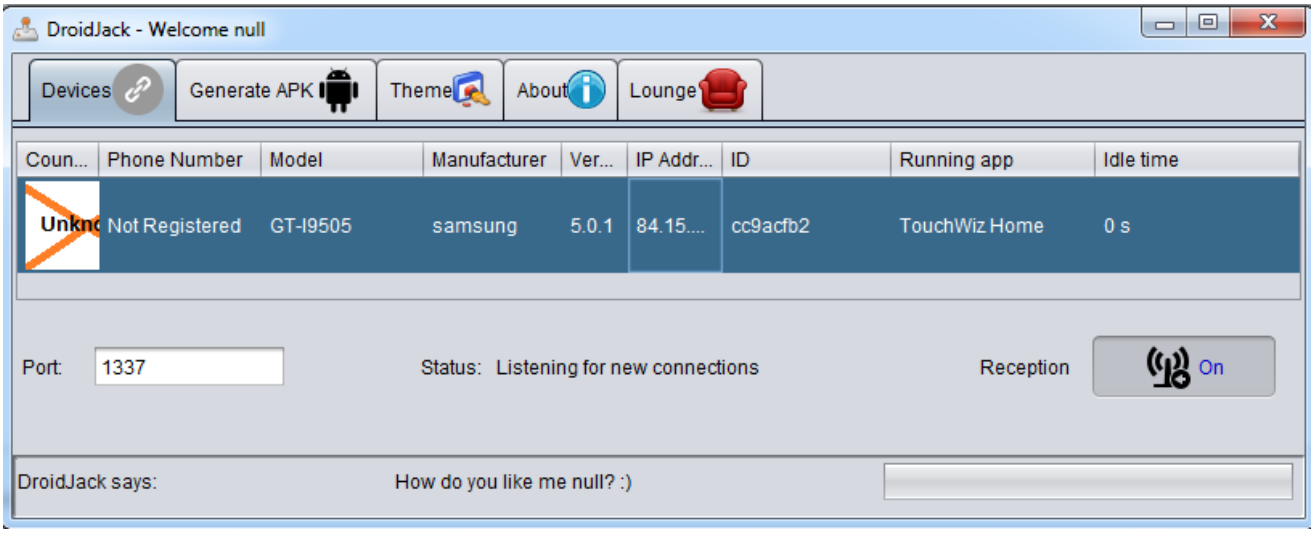
21 Pav. Kenkėjiškos programos sąveika su kuriamu prototipu.

10.1.1. „DroidJack“ 4.4 Trojos arklys

Viena iš mokamų ir laisvai prieinamų programų yra „DroidJack“ 4.4 kurios pagalba bus atliktas tyrimas parodantis ar galima apsisaugoti nuo šio Trojos arklio. Šis virusas yra labai universalus, jo pagalba galima padaryti įskiepį į esamą programinį kodą be jokiu papildomų programavimo žinių, tada įdiegti jį į aukos telefoną ir pradėti vykdyti žalingus veiksmus. „DroidJack“ veikimui būtinas serveris kurį jis susikuria naudodamas tą patį terminalą per kurį yra vykdomos

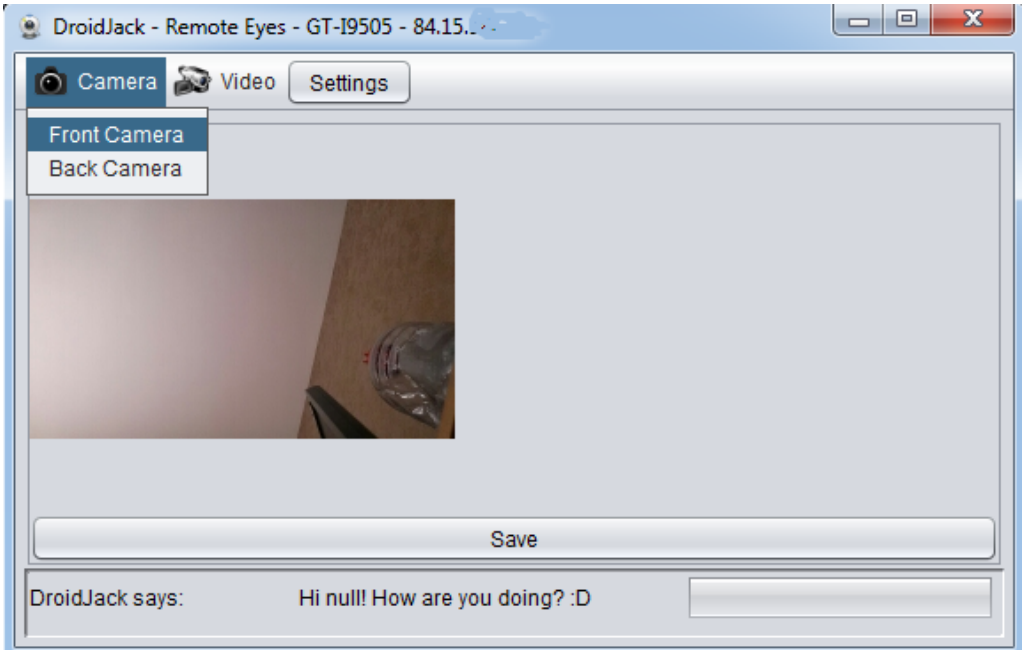
užklausos į telefoną. Naudojant DNS adresą užklausos į telefoną gali būti siunčiamos per bet koki tinklą, bet kurioje pasaulio vietoje. Tyrimui atlikti buvo sugeneruotas įdiegiamasis paketas kuris užima 275 kilo baitus ir privalo būti įdiegtas į telefoną kaip bet kuri kita programa. Iš visų ištirtų Trojos arklių tipo virusų „DroidJack“ yra vienas geriausių pagal siūlomas galimybes tokias kaip automatiniai failų persiuntimai, garso, vaizdo pasiklausimas vos tik yra sužadinamas telefonas arba GPS sekimas. „DroidJack“ siūlo pasirinkimą prisegti savo žalingą programinį kodą prie jau parašytos programėlės kuri vėliau bus įdiegiama į busimų aukų telefonus per oficialius tinklus. Po sėkmingo prisijungimo atsiranda telefoną identifikuojantys parametrai kurie lengviau padeda nustatyti kuriam telefonui norima siųsti komandas.

Įdiegus ir atidarius sugeneruotą paketą serveris kaip mat susijungia su telefonu, atsiranda naujas įrašas rodantis, kad telefonas yra pasiekiamas ir jam galima pradėti siųsti kenkėjiškas komandas. Mus domina tik Vaizdo ir garso užblokavimo galimybės, todėl kitokios komandos nei garso ir vaizdo transliavimui nėra nagrinėjamos.



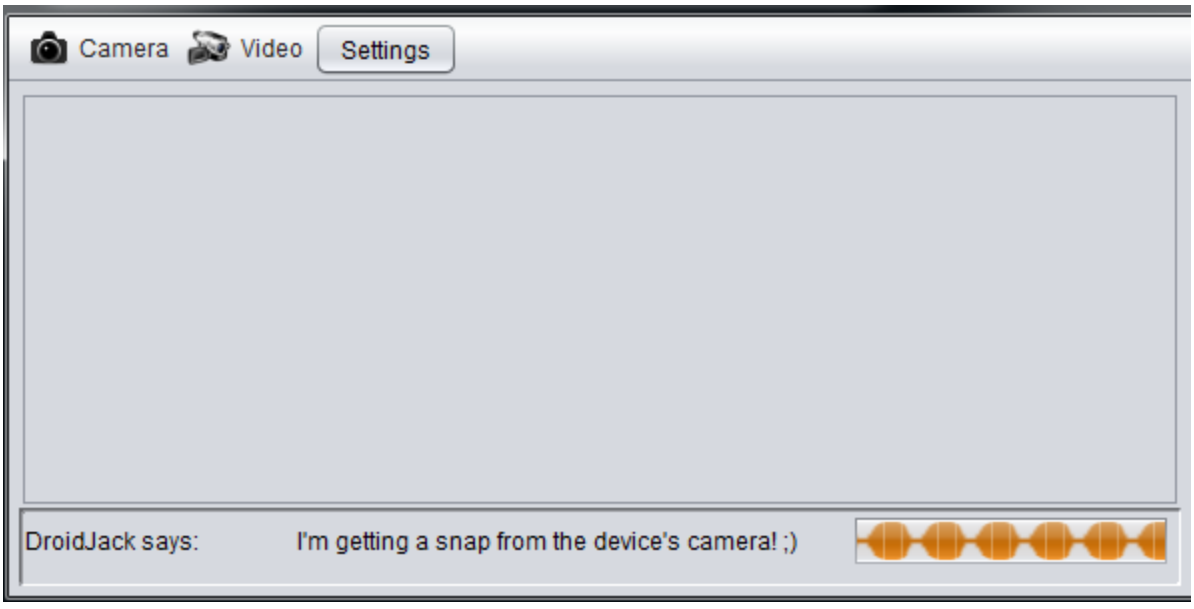
22 Pav. „DroidJack“ 4.4 terminalas

Po sėkmingo prisijungimo galima pradėti siųsti komandas į telefoną, kurių atsakymas grąžinamas i terminalo langą. Žemiau pavaizduotame paveikslėlyje „DroidJack“ Trojos arklio pagalba yra padaryta, ant stalo padėto telefono, nuotrauka per priekyje esančią kamerą. Užklausa iš telefono grąžinta sėkmingai.



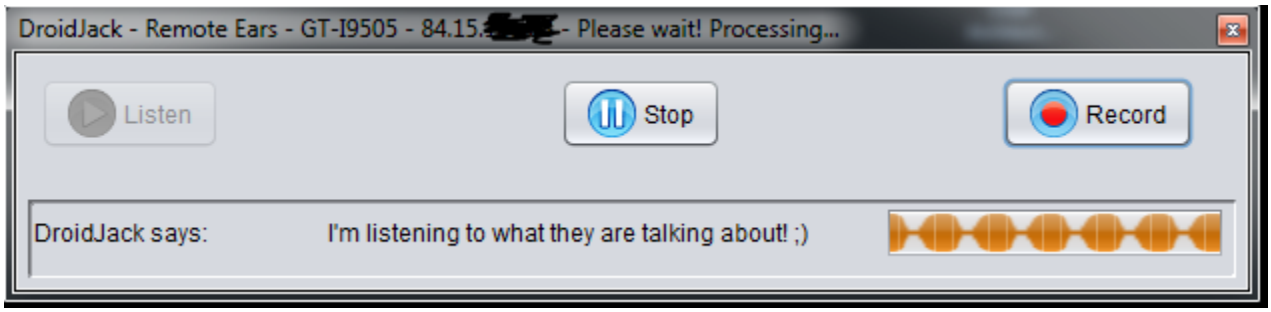
23 Pav. „DroidJack“ pagalba padarytas vaizdas, priekine kamera.

Naudojantis sukurtu prototipu užblokuojami bet kokie ketinimai pasinaudoti kamera. Bandoma pakartoti tuos pačius veiksmus gauti vaizdui terminale, darant fotografiją per priekinę kamerą.



24 Pav. „DroidJack“ nepavykęs bandymas nuskaityti kamerą

Kameros nuskaitymas buvo nesėkmingas ir joks vaizdas nebuvo grąžintas. Šis bandymas įrodo, kad kameros blokavimas veikia norint uždrausti kenkėjiškoms programoms ja pasinaudoti.

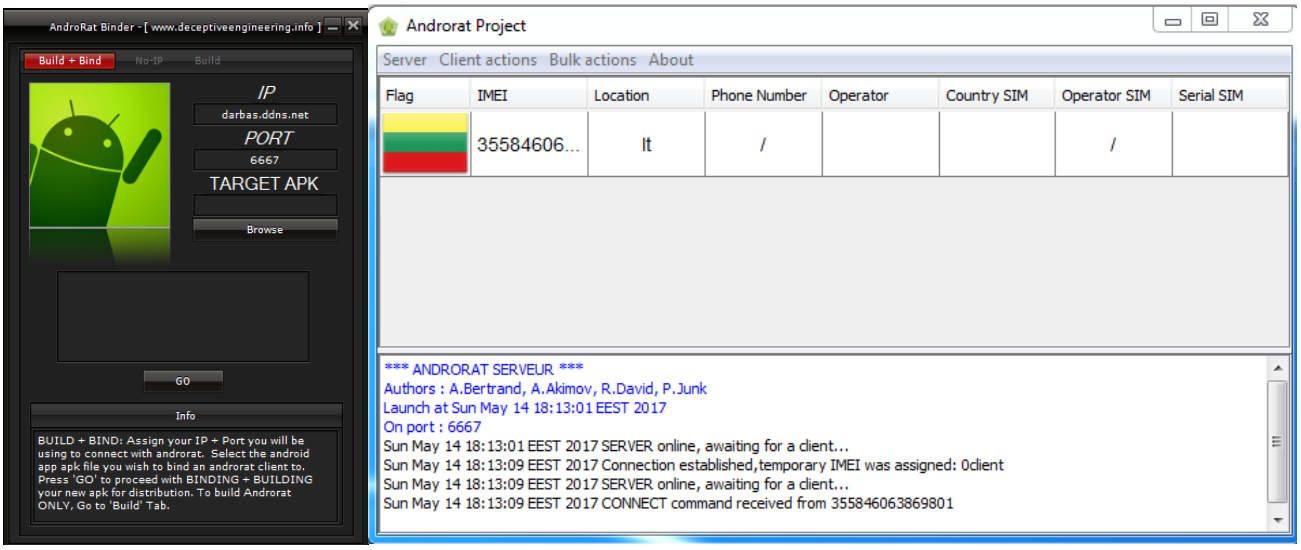


25 Pav. Bandymas padaryti garso įrašą.

Taip pat kaip su vaizdu taip ir su garsu po mikrofono užblokavimo buvo galima girdėti tik tylą. Šie bandymai įrodo, kad vaizdo ir garso užblokavimas buvo sėkmingas ir „DroidJack“ virusas atlikti kenkėjiškų veiksmų nesugebėjo. Pasirinkus standartinę fotografijoms daryti programą telefone kaip neblokuojamą, kameros ir vaizdo blokavimas buvo sėkmingai išjungtas vos tik ji atsirado aktyvių procesų sąraše.

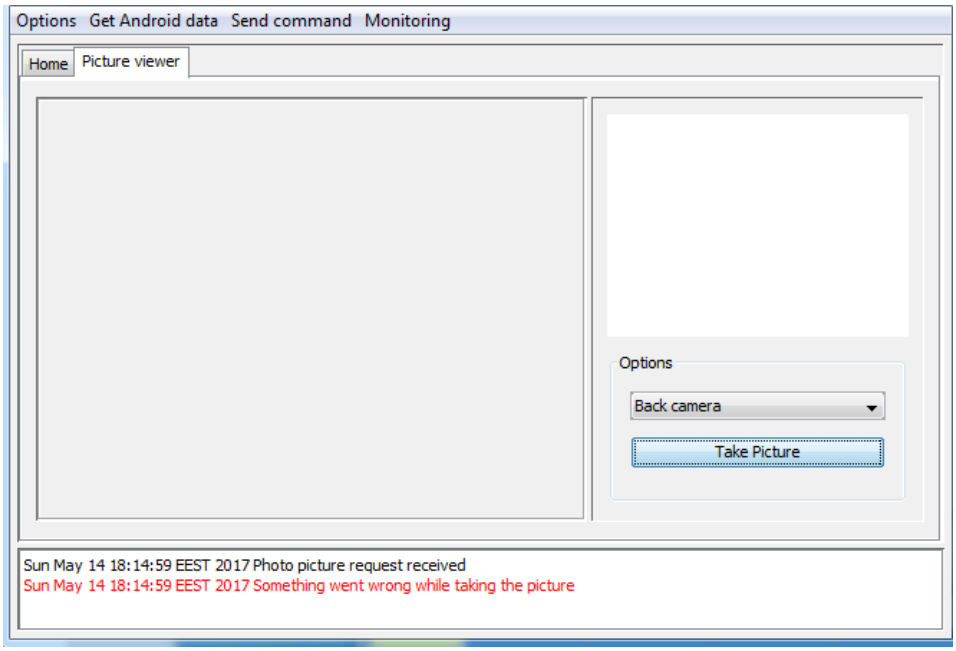
10.1.2. **„Androrat“ Trojos arklys**

„Androrat“ yra laisvai prieinamas ir nemokamas Trojos arklys parašytas A. Bertrand, R.David ir A. Akimov autorių naudojant Java. Šis virusas skiriasi nuo kitų nes jo kenkėjiško kodo instaliacinis paketas užima vos 57 kilo baitus vietos. Šio virus veikimo paruošimui reikalingos dvi atskiros programos paketo sukūrimui ir terminalas komandų vykdymui. Kaip ir dauguma Trojos arklių taip iš šiam virusui reikalingas serveris kuris yra sukuriamas programišiaus asmeniniame kompiuteryje. Įdiegtas paketas aukos telefone pradeda veikti vos tik yra atidaroma programa.



26 Pav. „Androrat“ terminalo užmegztas ryšys su telefonu

Po sėkmingo prisijungimo yra užmezgamas ryšys tarp „Androrat“ terminalo ir telefono. Tik prisijungus prie terminalo iš karto yra parodomas telefono identifikacijos numeris IMEI, telefono numeris ir kiti konfidencialūs duomenys. „Androrat“ turi mažesnį pasirinkimą komandų kurias galima įvykdyti, bet garso ir vaizdo įrašymo pasirinkimai yra. Pirmuoju bandymu be kameros užblokavimo, nuotraukos padarymo komanda suveikė ir vaizdas atsirado terminalo lange, o po kameros užblokavimo buvo gražinta klaida apie nepavykusį bandymą. Mikrofono blokavimas kaip ir prieš tai darytame tyrime suveikė sėkmingai ir po bandymo pasiklausyti buvo girdima tyla.



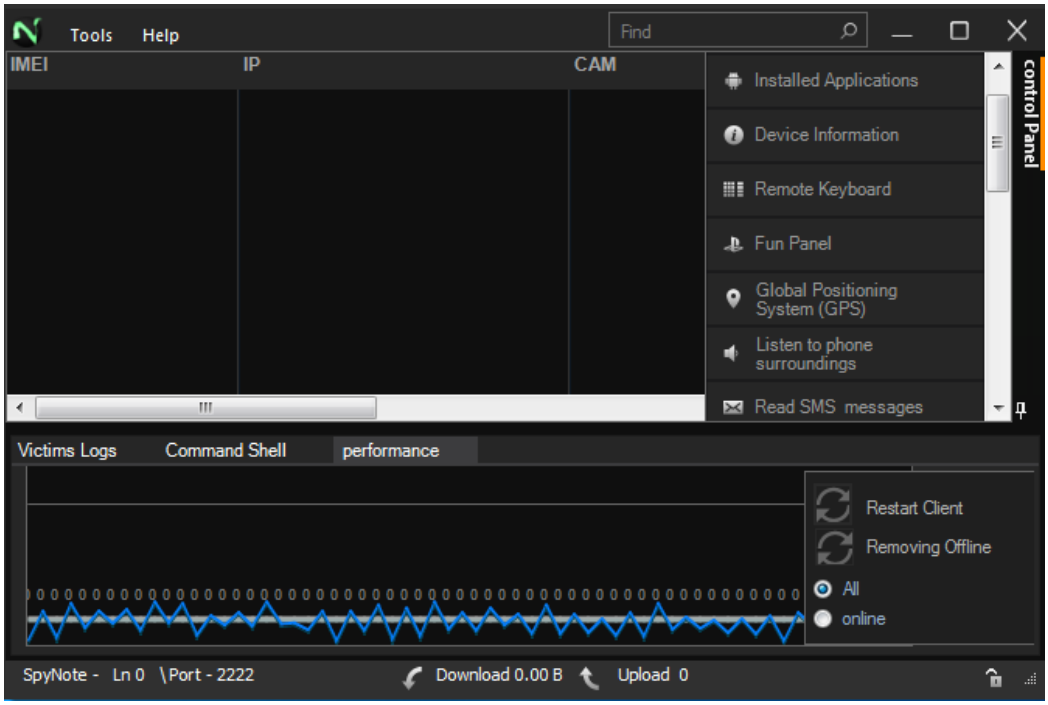
27 Pav. „Androrat“ nesėkmingas bandymas padaryti fotografiją

Tyrimas parodo, kad sukurtas prototipas atlieka savo pagrindinę funkciją uždrausdamas, bet kokius bandymus pasinaudoti kamera ir mikrofonu.

10.1.3. **„SpyNote“ 2.4.1 Trojos arklys**

„SpyNote“ yra Trojos arklys skirtas prisijungimui prie telefono per terminalą. Kaip prieš tai aptarti virusai „SpyNote“ turi daugiausia komandų kurias galima įvykdyti kenkėjiškiems tikslams atlikti. Šis Trojos arklys kitaip nei kiti, susikuria serveri prie kurio galima prisijungti nuotoliniu būdu per interneto naršyklę. Prisijungimai naudojamas dinaminis DNS adresas. Jungiantis tokiu būdu nereikia nuolat tikrinti ar nepasikeitė IP adresas tam, kad prisijungimas būtų sėkmingas. Kadangi šis virusas yra laisvai prieinamas ir nemokamas todėl jis turi daugiau funkcionalumo klaidų ir kai kurios komandos neveikia.

Po sėkmingo prisijungimo buvo galima tiesiogiai transliuoti vaizdo ir garso įrašus tiesiai terminale. Naudojant sukurtą prototipą po garso ir vaizdo užblokavimo telefone „SpyNote“ nebegalėjo gauti tokio atsakymo iš kameros ir gražino klaidą, o garso nuskaitymo metu buvo girdima tyla.



28 Pav. „SpyNote“ terminalo langas

Bandymo rezultatai parodo, kad sukurtas prototipas atlieka savo pagrindines funkcijas ir sėkmingai apsaugo telefono savininką nuo didžiausią grėsmę keliančių privatumui spragų. Pasirinkus standartinę fotografijoms daryti programą telefone kaip neblokuojamą, kameros ir vaizdo blokavimas buvo sėkmingai išjungtas vos tik ji atsirado aktyvių procesų sąraše.

10.2. **Eksperimento rezultatai**

Atlikus eksperimentą, buvo pastebėta, kad sukurtas prototipas sėkmingai užblokuoja tris populiariausius ir daugiausiai paplitusius Trojos arklius skirtus išmaniesiems telefonams. „DroidJack“ blokavimo eksperimento metu, buvo pastebėta, kad į „DroidJack“ terminalą yra grąžinama žinutė apie galimą programos išaiškinimą, dėl kilusių sunkumų nuskaitant kameros ir garso įvesties signalus. Garso ir vaizdo blokavimas buvo sėkmingas, virusas kenkėjiškų veiksmų atlikti nesugebėjo. „DroidRat“ Trojos arklys po kameros ir garso užblokavimų iš karto grąžino klaidos žinutę į terminalą, apie nepavykusį bandymą padaryti fotografiją ir garso įrašą. „SpyNote“ tyrimo rezultatai taip pat buvo sėkmingi, po garso ir vaizdo užblokavimo. Atlikus eksperimentą, galima daryti išvadas, kad sukurtas prototipas, sėkmingai užblokuoja garso ir vaizdo išvesties signalus kenkėjiškose programose, o procesų sąraše atsiradus programai, kurios norima neblokuoti, visi blokavimai buvo išjungti sėkmingai, o jai išnykus, garso ir vaizdo blokavimas buvo vėl aktyvuotas.

11. IŠVADOS

Atlikus kenkėjiškų programų analizę, galima daryti išvadas, kad rizika privatumui yra labai didelė dėl plataus kenkėjiškų programų paplitimo ir gebėjimo užsimaskuoti tarp naudingo funkcionalumo ar nekaltai atrodančių programų. Pagal paplitimą pirmauja „DroidRat“ ir sudaro apie 72% visų kenkėjiškų programų. Iš likusių kenkėjiškų programų apie 24% sudaro Trojos arklio tipo virusai, kurie pagal savo galimybes yra gerokai pavojingesni nei „DroidRat“ virusai. Į 4% įeina labai didelę grėsmę keliantys virusai kuriuos aptikti yra ypatingai sunku, nes paprastai jie išnaudoja *Android* operacinės sistemos dizaino trūkumus, dėl ko kyla sunkumu juos aptinkant. Tiriant analogus buvo pastebėta, jog rinkoje jau yra sukurta programėlių kurios blokuoja vaizdo ir garso signalus, taip pat yra programų kurios skirtos specifiniams virusams aptikti, kurių neranda antivirusinės ir kitokio apsauginio turinio programos. Atlikus analize matomas akivaizdus poreikis prevencijai, prieš kenkėjiško turinio programas keliančias grėsmę savininko ir aplinkinių privatumo saugumui.

Atlikus projektavimo tyrimą, buvo pastebėta, kad rinkoje esančių analogų funkcionalumas yra nepakankamas kada norima neprarasti pagrindinės telefono funkcijos, nes prieš naudojantis kamera arba skambinimo funkcija, vaizdo ir garso blokavimo funkcijos turi būti išjungtos, o tai sukelia didelių nepatogumų telefono savininkui. Matomas aiškus poreikis turėti galimybę išsirinkti kurių programų neblokuoti, todėl buvo pasirinktas metodas leidžiantis išskirti specifines programas iš įdiegtųjų programų sąrašo, kurioms atsiradus aktyvių procesų sąraše, kameros ir vaizdo blokavimai būtų automatiškai išjungiami. Tyrimas parodė, jog geriausias metodas blokuoti kamerą yra pasinaudoti administratoriaus prieiga, kurio pagalba galima visiškai užblokuoti kameros įvadą. Garso blokavimui yra pasirenkamas metodas, išnaudojus *Android* operacinės sistemos trūkumus, kada dvi skirtingos programos negali naudotis mikrofonu vienu metu. Nuolatinis garso įrašinėjimas ir jo užtildymas, padėtų apsaugoti nuo kenkėjiškų programų bandymo pasinaudoti mikrofonu.

Realizavus prototipą buvo pastebėta kad, kameros ir garso blokavimas veikia visoms nekenkėjiškoms programoms. Kada paleidžiamos telefone esančios programos, kurios naudojasi garsu arba kamera, jos gražina skirtingas klaidos žinutes apie nepavykusi bandymą įrašinėti vaizdą ir garsą. Užblokavus kamerą buvo pastebėta jog yra gražinama saugumo pažeidimo žinutė į išvesties ekraną pranešanti, kad kameros naudojimas yra uždraustas saugumo politikos. Užblokavus garso įvestį, paleistos programos gražina klaidos žinutę apie nepavykusį bandymą įrašyti garso, nes mikrofonas jau yra naudojamas kitose programėlėse. Programos išskyrimui ištestuoti buvo pasirinkta standartinė *Android* operacinės sistemos programa pavadinimu „Camera“. Jai atsirandant aktyvių procesų sąraše, kameros ir garso blokavimas buvo sėkmingai išjungtas, kas parodo jog sukurtas prototipas atlieka projektuojamas pagrindines funkcijas.

Atlikus eksperimentą, buvo pastebėta, kad sukurtas prototipas sėkmingai užblokuoja tris populiariausius ir daugiausiai paplitusius Trojos arklius skirtus išmaniesiems telefonams. „DroidJack“ blokavimo eksperimento metu, buvo pastebėta, kad į „DroidJack“ terminalą yra grąžinama žinutė apie galimą programos išaiškinimą, dėl kilusių sunkumų nuskaitant kameros ir garso įvesties signalus. Garso ir vaizdo blokavimas buvo sėkmingas, virusas kenkėjiškų veiksmų atlikti nesugebėjo. „DroidRat“ Trojos arklys po kameros ir garso užblokavimų iš karto grąžino klaidos žinutę į terminalą, apie nepavykusį bandymą padaryti fotografiją ir garso įrašą. „SpyNote“ tyrimo rezultatai taip pat buvo sėkmingi, po garso ir vaizdo užblokavimo. Atlikus eksperimentą, galima daryti išvadas, kad sukurtas prototipas, sėkmingai užblokuoja garso ir vaizdo išvesties signalus kenkėjiškose programose, o procesų sąraše atsiradus programai, kurios norima neblokuoti, visi blokavimai buvo išjungti sėkmingai, o jai išnykus, garso ir vaizdo blokavimas buvo vėl aktyvuotas.

12. LITERATŪROS SĄRAŠAS.

1. Hoffma, C.," *Android* Has a Big Security Problem, But Antivirus Apps Can't Do Much to Help", 2015-10-29 [žiūrėta 2016-01-20] prieiga per internetą <http://www.howtogeek.com/232436/*Android*-has-a-big-security-problem-but-antivirus-apps-cant-do-much/>

2. Hoffma, C.," What's the Best Antivirus for Windows 10? (Is Windows Defender Good Enough?)", 2015-10-08 [žiūrėta 2016-01-20] prieiga per internetą <http://www.howtogeek.com/225385/what%E2%80%99s-the-best-antivirus-for-windows-10-is-windows-defender-good-enough/>
3. Hoffma, C.," *Android*'s Stagefright Exploit: What You Need to Know and How to Protect Yourself", 2015-11-08 [žiūrėta 2016-01-20] prieiga per internetą <http://www.howtogeek.com/225834/stagefright-what-you-need-to-know-and-how-to-protect-yourself/>
4. Hoffma, C.," *Android*'s Stagefright Exploit: What You Need to Know and How to Protect Yourself", 2015-10-29 [žiūrėta 2016-01-20] prieiga per internetą <http://www.howtogeek.com/232436/*Android*-has-a-big-security-problem-but-antivirus-apps-cant-do-much/>
5. Hoffma, C.," Not Getting *Android* OS Updates? Here's How Google Is Updating Your Device Anyway", 2014-12-14 [žiūrėta 2016-01-20] prieiga per internetą <http://www.howtogeek.com/179638/not-getting-*Android*-os-updates-heres-how-google-is-updating-your-device-anyway/>
6. Robin Liss," How Your Phone Camera Can Be Used to Spy on You ", 2014-05-30 [žiūrėta 2016-01-20] prieiga per internetą <http://cameras.reviewed.com/features/how-your-smartphone-camera-can-be-used-to-spy-on-you>
7. Kimm Zetter," Snacks for your mind ", 2014-05-22 [žiūrėta 2016-01-20] prieiga per internetą <http://snacksforyourmind.blogspot.co.uk/2014/05/exploring-limits-of-covert-data.html>
8. Kimm Zetter," Hackers Can Control Your Phone Using a Tool That's Already Built Into It ", 2014-07-31 [žiūrėta 2016-01-20] prieiga per internetą <http://www.wired.com/2014/07/hackers-can-control-your-phone-using-a-tool-thats-already-built-into-it/>
9. Robert Templeman, Zahid Rahman, David Crandall, Apu Kapadia," PlaceRaider: Virtual Theft in Physical Spaces with Smartphones ", 2012-09-27 [žiūrėta 2016-01-20] prieiga per internetą <http://arxiv.org/pdf/1209.5982v1.pdf>
10. Robert Templeman, Zahid Rahman, David Crandall, Apu Kapadia," PlaceRaider: Virtual Theft in Physical Spaces with Smartphones ", 2012-09-27 [žiūrėta 2016-01-20] prieiga per internetą <http://arxiv.org/pdf/1209.5982v1.pdf>
11. Wish Wu," MMS Not the Only Attack Vector for "Stagefright" ", 2015-07-31 [žiūrėta 2016-01-20] prieiga per internetą <http://blog.trendmicro.com/trendlabs-security-intelligence/mms-not-the-only-attack-vector-for-stagefright/>
12. Ross Anderson," PlaceRaider: Virtual Theft in Physical Spaces with Smartphones ", 2012-09-27 [žiūrėta 2016-01-20] prieiga per internetą <http://www.cl.cam.ac.uk/~rja14/Papers/pinskimmer_spsm13.pdf>
13. https://en.wikipedia.org/wiki/*Android*_%28operating_system%29
14. Kannon Yamada,"The Seven Deadly *Android* Permissions: How to Avoid the Sin of Slothful Preparedness" 2013 Kovo 16 [žiūrėta 2016-05-15] prieiga per internetą. http://www.makeuseof.com/tag/the-seven-deadly-*Android*-permissions-how-to-avoid-the-sin-of-slothful-preparedness/
15. Google Inc. „*Android* Reference manuals." System permissions https://developer.*Android*.com/guide/topics/security/permissions.html

16. Google Inc. „*Android* Reference manuals.“ *Android* Security [žiūrėta 2017-01-05] prieiga per internetą. https://source.*Android*.com/security/
17. Wikipedia „Stingray phone tracke“ [žiūrėta 2017-01-05] prieiga per internetą. https://en.wikipedia.org/wiki/Stingray_phone_tracker
18. Google Inc. „*Android* Reference manuals.“ [žiūrėta 2017-01-05] prieiga per internetą. https://developer.*Android*.com/guide/components/processes-and-threads.html
19. Google Inc. „*Android* Reference manuals.“ [žiūrėta 2017-01-05] prieiga per internetą. https://developer.*Android*.com/reference/*Android*/os/Process.html
20. Google Inc. „*Android* Audio Reference manuals.“ [žiūrėta 2017-03-05] prieiga per internetą. https://source.*Android*.com/devices/audio/
21. Google Inc. „*Android* Video Reference manuals.“ [žiūrėta 2017-03-05] prieiga per internetą. https://source.*Android*.com/devices/camera/index.html
22. „The Statistics Portal“ Statistics and Studies from more than 18,000 [žiūrėta 2017-03-05] prieiga per internetą. https://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-play-store/
23. „Spreitzenbarth mobile security and Forensis„ [žiūrėta 2017-03-05] prieiga per internetą. https://forensics.spreitzenbarth.de/*Android*-malware/
24. Hoffma, C.,“ How Your Phone Camera Can Be Used to Stealthily Spy on You: ransplantation Attacks against *Android* Camera Service“, 2015-10-08 [žiūrėta 2016-01-20] prieiga per internetą <http://dl.acm.org/citation.cfm?id=2699103&CFID=923922343&CFTOKEN=97857369>
25. Newsletter “ Kaspersky security bulletins“, 2016, [žiūrėta 2017-03-20] prieiga per internetą<https://kasperskycontenthub.com/securelist/files/2016/12/KASPERSKY_SECURITY_BULLETIN_2016.pdf>
26. Yajin Zhou ir Xuxian Jiang, "Dissecting *Android* Malware: Characterization and Evolution“, 2016, [žiūrėta 2017-04-05] prieiga per internetą <http://dl.acm.org/citation.cfm?id=2310710>
27. Wenrui Diao, Xiangyu Liu ir kiti, „Your Voice Assistant is Mine: How to Abuse Speakers to Steal Information and Control Your Phone“, 2014,07 [žiūrėta 2017-05-05] prieiga per internetą <http://dl.acm.org/citation.cfm?id=2666623&CFID=923922343&CFTOKEN=97857369>
28. Manar Mohamed ir kiti, „SMASheD: Sniffing and Manipulating *Android* Sensor Data“, 2016.09, [žiūrėta 2017-04-05] prieiga per internetą, <http://dl.acm.org/citation.cfm?id=2857749&CFID=923922343&CFTOKEN=97857369>.
29. H. Wang, A. Moshchuk, A. Felt, „Permission Re-Delegation: Attacks and Defenses“, 2013.10, [žiūrėta 2017-04-05] prieiga per internetą, <http://www.cs.columbia.edu/~lierranli/coms6998-10Spring2013/papers/perredel_usenixsec2011.pdf>.
30. G. Gokul, Y. Yan, K. Dantu, „Real Time Sound Processing on Android“, 2016.02, [žiūrėta 2017-04-05] prieiga per internetą, <http://dl.acm.org/citation.cfm?id=2990512&CFID=765219812&CFTOKEN=22437852>.
31. V. Singh, K. Sharma, „Smartphone Security: Review of Challenges and Solution“, 2016.05, [žiūrėta 2017-04-05] prieiga per internetą, < http://dl.acm.org/citation.cfm?id=2905214>.