KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
KOMPIUTERIŲ KATEDRA

Justinas Grauslis

BELAIDŽIŲ 802.11 STANDARTO TINKLŲ KADRO SEKOS NUMERIO ANALIZE PAREMTO APSAUGOS ALGORITMO SUDARYMAS IR TYRIMAS

Magistro baigiamasis darbas

Vadovas
lekt. dr. Dangis Rimkus

KAUNAS, 2011

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
KOMPIUTERIŲ KATEDRA

BELAIDŽIŲ 802.11 STANDARTO TINKLŲ KADRO SEKOS NUMERIO ANALIZE PAREMTO APSAUGOS ALGORITMO SUDARYMAS IR TYRIMAS

Magistro baigiamasis darbas

Recenzentas
prof. dr. Rimantas Plėštys
2011 05

Vadovas
lekt. dr. Dangis Rimkus
2011 05

Atliko
IFN 9/3 gr. stud.
Justinas Grauslis
2011 05

KAUNAS, 2011

TURINYS

PAVEIKSLĖLIŲ SĄRAŠAS

LENTELIŲ SĄRAŠAS

1. ĮVADAS

Belaidžiai paketinio perdavimo tinklai duomenis siunčia elektromagnetinėmis bangomis bendru eteriu. Palyginus su laidiniais tinklais, belaidžiai teikia paklausias ryšio mobilumo, prieinamumo, ir paprastumo diegti savybes. Dėl šių savybių belaidžiai tinklai tapo viena iš pagrindinių informacijos perdavimo sistemų platformų, o didėjant interneto ir lokaliųjų privačių tinklų plėtrai, IEEE (angl. *Institute of Electrical and Electronics Engineers*) organizacijos sudarytos 802.11 standartų šeimos belaidžiai tinklai ir jų technologiniai aspektai tapo ypač aktualūs.

Saugumo atžvilgiu, dėl architektūrinių bendros perdavimo terpės ir radijo transliacijos savybių, belaidžiai tinklai tapo labiau pažeidžiami nei laidiniai, todėl buvo sukurti specializuoti belaidžių tinklų vartotojų autentifikaciją, duomenų integruotumą ir konfidencialumą užtikrinantys protokolai: WEP (angl. *Wired Equivalent Privacy*), WPA (angl. *WiFi Protected Access*) ir kiti. Šie protokolai neužtikrina tinklo paslaugų prieinamumo, o tai sukuria daug belaidžio tinklo saugos spragų atsisakymo aptarnauti atakoms, dar vadinamomis DoS (angl. *Denial Of Service*) atakomis. Šių atakų metu, pasinaudojant konkrečiais belaidžio tinklo standarto ar galutinės įrangos pažeidžiamumais, sutrikdomos arba visiškai nutraukiamos tinklo teikiamos paslaugos legaliems vartotojams.

Šiai dienai nėra nė vieno standarto, kuris numatytų apsaugos priemones prieš belaidžių tinklų DoS atakas. 802.11 standarto plėtinys 802.11w sprendžia tik nedidelę dalį belaidžio tinklo saugumo spragų atsisakymo aptarnauti atakų atžvilgiu, todėl ši problema reikalauja nuodugnesnių tyrinėjimų ir naujų apsaugos nuo atsisakymo aptarnauti atakų metodų bei algoritmų plėtojimo.

Siekiant apsaugoti belaidžius tinklus nuo DoS atakų, būtina išanalizuoti tinklo architektūrą, konkrečius jos pažeidžiamumus, tais pažeidžiamumais paremtas atakas visuose OSI (angl. *Open Systems Interconnection*) modelio lygmenyse, modeliuoti galimus atakų apsaugos algoritmus ir tirti jų praktinio panaudojimo galimybes.

Šio magistrinio darbo *tyrinėjimų sritis* – belaidžiai vietiniai kompiuterių tinklai. *Tyrimų objektas* – apsaugos nuo belaidžio tinklo atsisakymo aptarnauti atakų metodai ir algoritmai. Keliamas *tikslas* sudaryti kadro sekos numerio analize paremtą apsaugos nuo DoS atakų algoritmą ir jį ištirti. Tikslo siekiui būtina įvykdyti šiuos *uždavinius*:

- Išanalizuoti belaidžio tinklo architektūrines savybes ir saugumo spragas DoS atakų atžvilgiu;
- Išanalizuoti esamus apsaugos nuo DoS atakų algoritmus;
- Sudaryti sekos numerio analize paremtą saugos algoritmą ir suprojektuoti jo tyrimų modelį;
- Sudaryti tyrimų matodiką ir ją taikant ištirti sudarytą algoritmą.

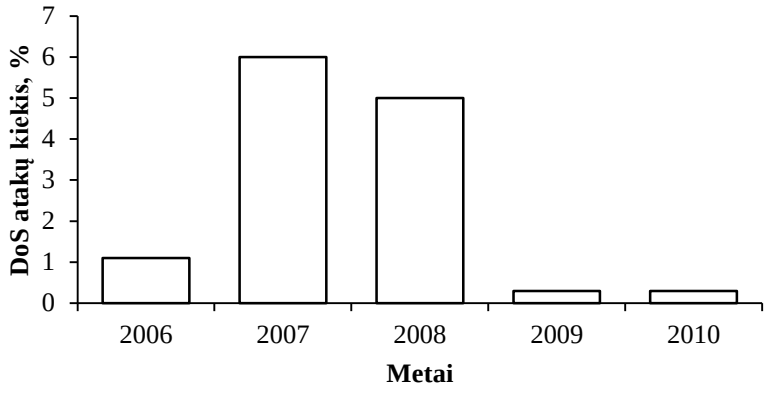
2. BELAIDŽIO TINKLO SAUGOS NUO ATSISAKYMO APTARNAUTI ATAKŲ ANALIZĖ

2.1. Tiriama sritis ir problematika

Belaidžio tinklo, kaip ir bet kokios kitos informacinės sistemos, apsaugos strategiją sudaro trijų saugumo tikslų siekimas: *konfidencialumo, integruotumo* ir *prieinamumo*. Atsisakymo aptarnauti atakos pažeidžia belaidžio tinklo *prieinamumo* tikslą, fiziniais, technologiniais ir organizaciniais metodais sutrikdydamos ar visiškai nutraukdamos tinklo ar jo komponento teikiamas paslaugas legaliems vartotojams [3,17].

Technologiniai atsisakymo aptarnauti atakų metodai remiasi paties standarto, kuris aprašo konkretų bevielį tinklą, bei jo programinės ir aparatinės realizacijos saugumo pažeidžiamumais. Įvairūs tyrimai parodė, jog 802.11 belaidžio tinklo standartas bei konkrečios jo programinės ir aparatinės realizacijos turi daug pažeidžiamumų, kuriais yra realizuojamos atsisakymo aptarnauti atakos [1,3,6,7].

2006-2010m Lietuvos Respublikos elektroninių ryšių ir informacijos saugumo padalinio statistikos duomenimis[1] (žr. 1 pav.), atsisakymo aptarnauti atakos sudarė 1 - 6% bendro incidentų kiekio.



1 pav. *Lietuvos DoS atakų statistika*

Nors bendrame incidentų kontekste atsisakymo aptarnauti atakos neužima didelės dalies, šios atakos yra nukreiptos prieš tarpinius tinklo įrenginius ir potencialiai gali sukelti daugiau žalos nei kitos atakos, todėl būtina įvertinti ir ištirti belaidžiame tinkle taikomų atsisakymo aptarnauti atakų metodus, taip pat apsaugos algoritmus nuo šių atakų.

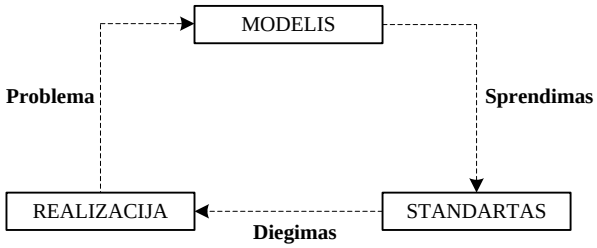
Toliau darbe minimos sąvokos *DoS* ir *atsisakymo aptarnauti ataka* yra laikomos sinonimais.

[1] Duomenys viešai pateikiami CERT LT svetainėje, adresu https://www.cert.lt/statistika.html

2.1.1. Belaidžio tinklo atsisakymo aptarnauti ataka

Belaidis tinklas – sudėtinga sistema, todėl pradedant analizuoti atakas ir jų apsaugos algoritmus, pagrįstus tinklo posistemių ir procedūrų elementais, pravartu apsibrėžti aspektus, kurių atžvilgiu bus analizuojamos saugumo problemos. Belaidžio tinklo problemas architektūriniu atžvilgiu galima analizuoti trimis aspektais:

- *Modelio* – tai saugumo problemų analizavimas tinklo (ar posistemio, algoritmo) modeliu, nepaisant konkrečių standartų ar sprendimų, o tinklą analizuojant matematiniais, kompiuteriniais ar kitais modeliavimo metodais. Šiam aspektui priskirtinas ir OSI modelis, bet kokį tinką abstrakčiai apibrėžiantis sluoksniais, kuriems būdingi tam tikri funkciniai ypatumai [3].
- *Standarto* – tai konkreti specifikacija, aprašanti konkretaus tinklo architektūrą fiziniu ir loginiu atžvilgiu. Šiame darbe analizuojamos IEEE 802.11 standartų šeimos (802.11a/b/g/n) belaidžio tinklo saugumo problemos ir jų sprendimo algoritmai, susiję su atsisakymo aptarnauti atakomis [1,3].
- *Realizacijos* – tai konkretaus standarto galutinė programinė ir (arba) aparatinė realizacija. Šiam aspektui būdinga tai, jog kai kurios saugumo spragos atsiranda standarto programinės ar aparatinės realizacijos metu, pavyzdžiui, pažeidžiamumu pasižyminčios tinklo įrangos tvarkyklės ar netobulas operacinės sistemos TCP/IP steko kodas [7].



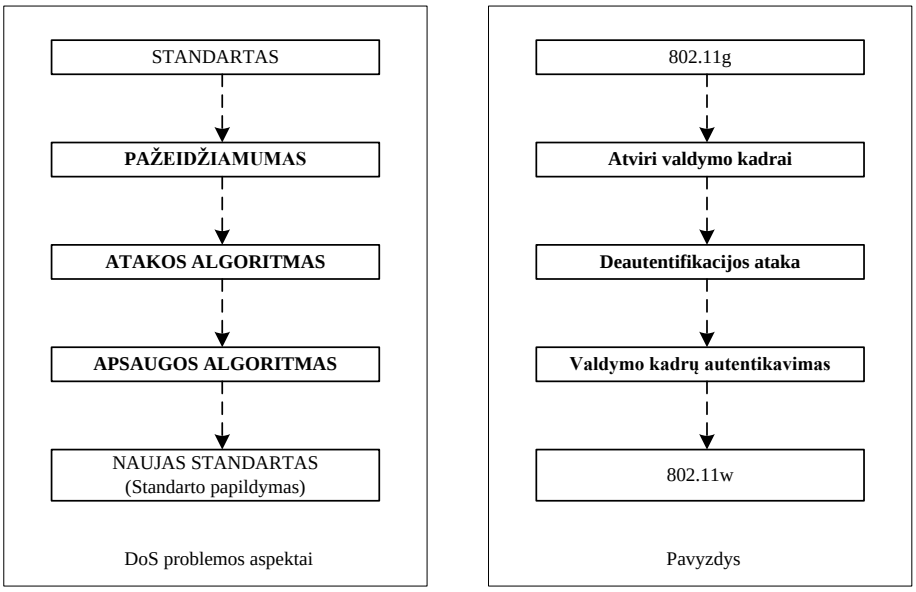
2 pav. *Belaidžio tinklo saugos analizės aspektai*

Šie trys aspektai yra susiję (žr. 2 pav.). Šiame darbe, belaidžio tinklo saugos analizė bus atliekama *modelio* ir *standarto* atžvilgiu.

Belaidžio tinko atsisakymo aptarnauti atakų problema susideda iš trijų neatsiejamų dedamųjų: *pažeidžiamumo*, *atsisakymo aptarnauti atakos algoritmo* ir *apsaugos nuo atsisakymo aptarnauti atakos algoritmo* [1,2].

- *Pažeidžiamumas* yra konkreti tinklo architektūros spraga, potencialiai sudaranti galimybę sutrikdyti ar nutraukti tam tikros tinklo funkcijos veiklą.
- *Atakos algoritmas* – tai konkretūs veiksmai, kurie išnaudodami tinklo pažeidžiamumą realiai atlieka tinklo funkcijos sutrikdymą arba nutraukimą,. Veiksmai realizuoti programinėmis bei techninėmis priemonėmis.
- *Apsaugos algoritmas* – konkretūs veiksmai, skirti užkirsti kelią atakos algoritmui arba jį slopinti.

Išspręstos saugumo problemos yra įtraukiamos į naujus standartus ar standartų papildymus. Problemų sprendimo loginė seka pateikta 3 pav.



3 pav. *Tinklo saugos problemų sprendimo loginė seka: a) apibendrinta DoS sritis; b) pavyzdys*

2.1.2. IEEE 802.11 standarto belaidžio tinklo apibendrinta architektūra

IEEE standartizacijos komitetas išleido IEEE 802.11 standartų šeimą, kuri aprašo belaidžio vietinio tinklo WLAN (angl. *Wireless Local Area Netwo*rk) ir belaidžio miestinio tinklo WMAN (angl. *Wireless Metropolitan Area Network*) specifikacijas. Remiantis OSI modeliu, šie standartai detalizuoja fizinio ir kanalinio lygmens parametrus ir procedūras. Konkrečių standartų architektūriniai aspektai skiriasi, todėl šiame skyrelyje aprašoma 802.11g standarto, kaip populiariausio WLAN standarto, architektūriniai aspektai, aktualūs atsisakymo aptarnauti atakų atžvilgiu [15].

IEEE 802.11 tinkluose yra naudojami trijų tipų kadrai:

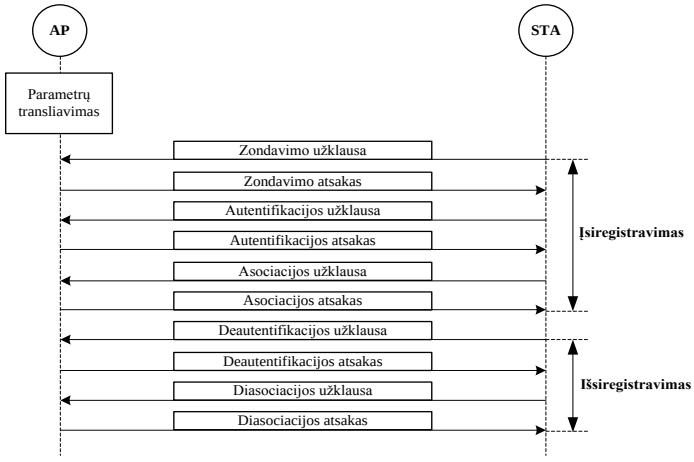
- *Valdymo* – skirti valdyti tinklinius sudarymus, keistis parametrais.
- *Kontrolės* – skirti valdyti perduodamus duomenis.
- *Duomenų* – aukštesnio lygmens paketus perduodantys kadrai.

Kiekvienas iš šių tipų turi tam tikrą kiekį būdingųjų kadrų, kurių pasiskirstymas pagal grupes anglų kalba pateiktas lentelėje Nr. 1.

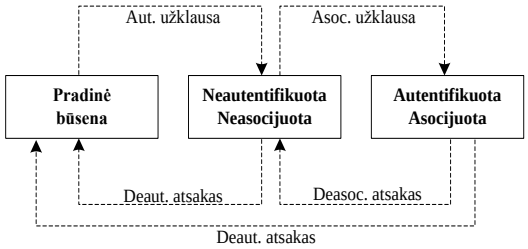
Lentelė Nr. 1 *IEEE 802.11 standarto šeimos belaidžių tinklų kadrų suvestinė [15]*

Valdymo kadrai (Management)	Kontrolės kadrai (Control)	Duomenų kadrai (Data)
Association request	PS-Poll (Power Save Poll)	Data
Association response	RTS (Request To Send)	Data + CF-ACK
Reassociation request	CTS (Clear To Send)	Data + CF-Poll
Reasociation response	ACK (Achnowledgement)	Data + CF-ACK + CF-Poll
Probe request	CF End (Contention Free End)	Null Function
Probe response	CF End + CF-ACK	CF-ACK
Beacon		CF-Poll
ATIM (Ad Hoc traffic Indication Map)		
Diassociation request		
Diassociation response		
Authentication request		
Authentication response		

Valdymo kadrų siuntimu realizuota belaidžio tinklo įrenginio įsiregistravimo į tinklą procedūra pavaizduota 4 pav., o besiregistruojančio įrenginio būsenos pateiktos 5 pav.



4 pav. *Įsiregistravimo į belaidį tinklą procedūra [8]*

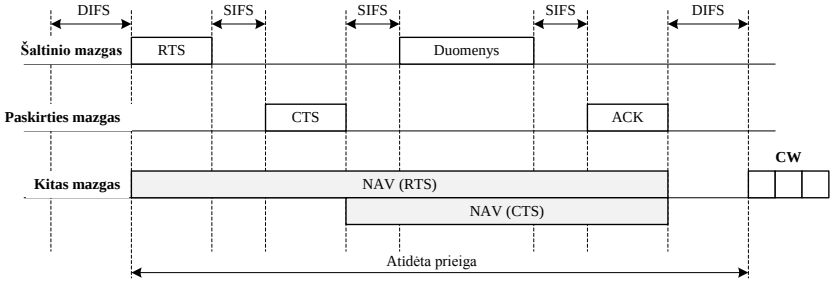


5 pav. *Besiregistruojančio į tinklą įrenginio būsenos [8]*

Belaidis tinklas naudoja bendrą perdavimo terpę – eterį. Tam, kad įvairių mazgų signalai būtų apdoroti suderintai, naudojamas bendros terpės valdymo mechanizmas CSMA/CA (angl. *Carrier Sense Multiple Access/Collision Avoidance*). Šio mechanizmo laiko diagrama pateikta 6 pav. Paprastai visi mazgai būna imtuvo režime, o jei prireikia išsiųsti kadrus, mazgai persijungia į siųstuvo režimą ir

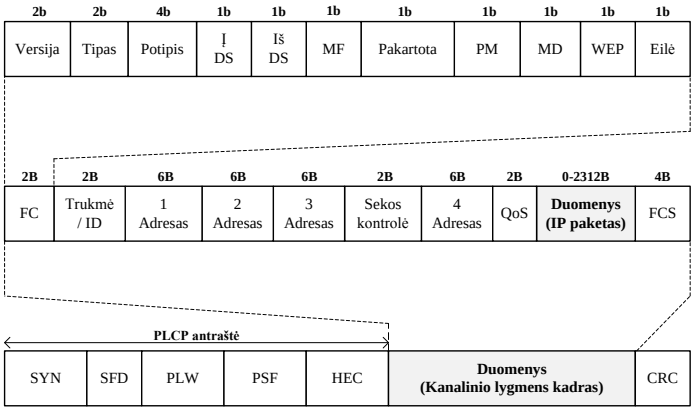
pasirinkę atsitiktinį laukimo laiką, kurį nustato specialus kintamasis CW (angl. *Contention Window*), pradeda tirti eterį, ar galima siųsti kadrus [15,8].

Eterio užimtumo identifikavimui naudojamas specialus modulis CCA (angl. *Clear Channel Assesment*). Kuomet CCA identifikuoja, jog terpė laisva, laukiamas tarpkadrinio laukimo laiko intervalo DIFS (angl. *Distributed Inter–Frame Space*) trukmės laikotarpis.



6 pav. *CSMA/CA mechanizmo laiko diagrama*

Jei kanalas lieka laisvas DIFS trukmę, mazgas mažina laukimo laiką kaskart, kuomet kanalas aptinkamas laisvas. Kuomet laukimo laikas išsenka, mazgas siunčia RTS (angl. *Request To Send*) žinutę aptarnaujančiam įrenginiui, pranešdamas ketinimą okupuoti kanalą. Aptarnaujantis įrenginys atsako CTS (angl. *Clear To Send*) kadru. Tuomet mazgas siunčia duomenų kadrus, o aptarnaujantis įrenginys juos patvirtina ACK (angl. *Acknowledgement*) kadrais. Pagal 802.11 standartą, RTS/CTS mechanizmas yra pasirenkamas – formaliai tinklas gali veikti ir be jo. Esant aktyviai komunikacijai, įrenginiai laukia trumpą tarpkadrinį laiko intervalą SIFS (angl. *Short Inter–Frame Space*). Jeigu perdavimas nepavyksta, laukimo laikas dvigubinamas. Kadruose esantis trukmės laukas savo reikšme nurodo aktyvios komunikacijos trukmę mikrosekundėmis ir suformuoja tinklo išnaudojimo vektorių NAV (angl. *Network Allocation Vector*) kiekvienoje kaimyninėje stotyje. Kanalo naudojimas kitam įrenginiui leidžiamas iki tol, kol NAV išsenka [15].



7 pav. *Belaidžio tinklo kadrų formatai [15]*

Fizinio lygmens duomenys, kurie yra kanalinio lygmens kadras, įterpiami į fizinio lygmens PLCP (angl. *Physical Layer Convergence Protocol*) protokolo segmentą, kuris tam tikra prasme yra "fizinio lygio kadras". Protokolas naudojamas įrenginių sinchronizacijai, fizinio lygmens patikros sumai ir kitoms žemo lygio operacijoms. Fizinio ir kanalinio lygmens kadrų formatai pateikti 7 paveikslėlyje, o lentelėje Nr. 2 pateikiami laukų aprašymai.

Lentelė Nr. 2 *Fizinio ir kanalinio lygmens kadrų antraštės laukai [15]*

Laukas	Aprašymas
SYN	Sinchronizacijai skirtas laukas. Dažnio keitimo atveju lygi 80 bitų, skleisto spektro – 128 bitai. Tai besikeičiančių 0 ir 1 seka (angl. *Synchronization*).
SFD	16 bitų laukas, žymintis kadrų pradžią (angl. *Start Frame Delimiter*).
PLW	12 bitų skaičius, žymintis baitų kiekį pakete (angl. *PLCP_PDU Length Word*).
PSF	Laukas, nurodantis MAC apkrovos siuntimo greitį (angl. *PLCP Signaling Field*).
HEC	16 bitų laukas, skirtas PLCP antraštės klaidų aptikimui (angl. *Header Error Check*).
CRC	32 bitų laukas skirtas identifikuoti kadro bitų klaidas (angl. *Cyclic Redundancy Check*).
FC	Kadro kontrolės laukas (angl. *Frame Control*).
Versija	Protokolo versijos numeris.
Tipas	2 bitų laukas nurodantis, ar kadras yra valdymo ar kontrolės ar duomenų.
Potipis	Nurodo, koks konkrečiai kadras yra iš tam tikro tipo kadrų aibės.
Į DS	Nustatoma, jeigu kadras siunčiamas iš bevielės prieigos stotelės į paskirstymo sistemą (angl. *Distribution System*)
Iš DS	Nustatoma, jeigu kadras gautas ir paskirstymo sistemos.
MF	Naudojama esant fragmentuotam kadrui (angl. *More Fragments*).
Pakartota	Nustatoma, jeigu kadras siunčiamas pakartotinai.
PM	Nurodoma kokiame galios režime buvo mazgas, kai išsiuntę kadrą (angl. *Power Management*).
MD	Naudojama stotelės energijos taupymo režime esančių mazgų kadrų valdymui (angl. *More Data*).
WEP	Nustatoma, jeigu naudojamas WEP apsaugos protokolas.
Eilė	Nustatoma, jeigu paketas buvo išsiųstas SOC režimu (angl. *Strictly Ordered Class*).
Trukmė/ID	Nurodo komunikacijos trukmę mikrosekundėmis
Adresas 1	Imtuvo adresas. Jei naudojama „į DS", tuomet tai bevielės prieigos stotelės adresas, jeigu naudojama „iš DS" – kito aptarnaujamo mazgo.
Adresas 2	Siųstuvo adresas. Jei naudojama „iš DS", tuomet tai bevielės prieigos stotelės adresas, jeigu naudojama „į DS" – kito aptarnaujamo mazgo.
Adresas 3	Jeigu Adresas 1 saugo paskirties adresą, šis laukas saugos šaltinio adresą. Jeigu Adresas 2 saugo šaltinio adresą, šis laukas saugos paskirties adresą.
Adresas 4	Jeigu naudojama WDS (angl. Wireless Distribution System) belaidė paskirstymo sistema, laukas saugos šaltinio mazgo adresą.
Sekos knt.	Saugo fragmento ir sekos numerį.
Duomenys	Enkapsiuliuotas aukštesnio lygmens paketas, paprastai IP (angl. *Internet Protocol*), iki 2312 oktetų.

2.2. Belaidžio tinklo atsisakymo aptarnauti atakos ir bendrieji saugos metodai

Belaidžio tinklo atsisakymo aptarnauti atakos realizuojamos naudojant įvairius tinklo pažeidžiamumus, todėl siekiant aiškumo, atakas patogiausia skirstyti pagal tai, kokį OSI tinklo modelio (žr. 8 pav.) lygmenį ataka naudoja.

Pirmieji keturi lygmenys tradiciškai priskiriami tinklo, kaip duomenų persiuntimo paslaugas teikiančio objekto, pusei. Fizinis lygmuo apima fizinę tinklo terpę, todėl atakos šiame lygmenyje

realizuojamos fiziniu įrangos sugadinimu ir triukšmo signalo generavimo metodais, siekant iškraipyti naudinguosius signalus.

8 pav. *Belaidžio tinklo lygių architektūra*

Kanalinio lygmens atakos naudojasi trūkumais, esančiais žemiausioje tinklo logikoje – kadrų formavimo procese, kadrų valdymo procese bei valdymo kadrais, naudojamais tinklo įrenginių tarpusavio kanalinių parametrų sukonfigūravimui. Pastarieji du lygmenys glaudžiai susiję su tinklo standartais, todėl juose neįmanoma realizuoti papildomų saugumo funkcijų nekeičiant standarto arba standarto realizacijos. Tinklo lygmens atsisakymo aptarnauti atakos susietos su tinklo protokolų saugos ribotumais bei maršrutizacija, o transporto lygmens atakos naudoja TCP (angl. *Transmission Control Protocol*) sesijų saugos pažeidžiamumus bei nesaugų UDP (angl. *User Datagram Protocol*) protokolą [6].

Paskutinieji trys lygmenys tradiciškai siejami su vartotoju (galiniu įrenginiu, serveriu), besinaudojančiu tinklo paslaugomis. Šių lygmenų saugumo ypatumai yra glaudžiai susieti su programine įranga, kuri realizuoja šiuos lygmenis vartotojo ar tarnybinės stoties pusėje. Šiuose lygmenyse išnaudojamas netobulas taikomųjų protokolų, programų veikimo algoritmas ir tinklinė sąveika, siekiant įvykdyti atsisakymo aptarnauti atakas. Sekančiose šio skyriaus dalyse detaliau aprašomi belaidžio tinklo saugumo pažeidžiamumai, jais pagrįstos konkrečios atakos bei apsaugos nuo atakų algoritmai pagal OSI lygmenis.

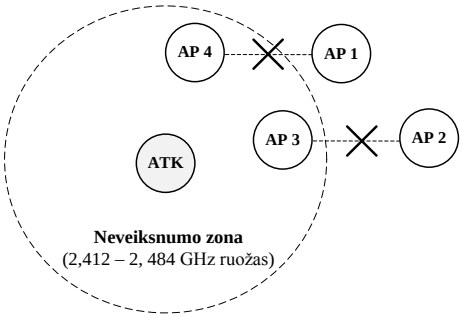
2.2.1. Fizinio lygmens atsisakymo aptarnauti atakos

Fizinio lygmens atakos literatūroje dažniausiai įvardijamos kaip slopinimo (*angl. Jamming*) atakos. Jų skiriamasis bruožas yra įvairaus profilio fizinio lygmens antraščių pakeitimai, kurie realizuojami siunčiant tam tikrų šablonų piktybinius signalus. Tuo siekiama, kad tinklo kadrai nebūtų perduodami arba būtų iškraipyti [6,18].

Vaizduodami atakų mechanizmus, naudosime šiuos sutartinius žymenis: **AP** (angl. *Access Point*) – legali belaidės prieigos stotelė; **ATK** (angl. *Attacker*) – atakuotojo įranga; **K** – kadras; ✕ - tinklo komunikacija visiškai nutraukiama; ⊣ - tinklo komunikacija yra sulėtinama ar sutrikdoma, bet veikianti.

2.2.1.1. Neribotų resursų ataka

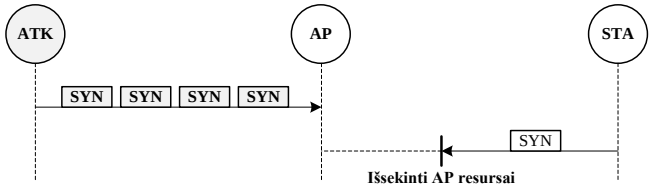
Neribotų resursų ataka (žr. 9 pav.) yra primityviausias fizinio lygmens slopinimo metodas, realizuotas stipraus trukdančio signalo (arba triukšmo) nuolatiniu generavimu plačiame dažnių diapazone, kurį naudoja konkreti belaidžio ryšio sistema. Šiai atakai reikalingas didelis energijos kiekis, nes naudojami galingi radijo siųstuvai [8].



9 pav. *Neribotų resursų ataka*

2.2.1.2. Preambulės ataka

Preambulės ataka (žr. 10 pav.) pagrįsta nuolatinio signalo su sinchronizaciniu SYNC šablonu siuntimu į belaidės prieigos įrangą, kuri atakos metu negali sinchronizuotis su legalia įranga. Atakos metu gali būti sukeliamas didelis bitų klaidų santykis. Atakai nereikalingas galingas siųstuvas – sėkmingai sutrikdyti įrangos darbą užtenka trimis eilėmis mažesnio atakuojančio įrenginio signalo lygio nei legalūs signalai [8].

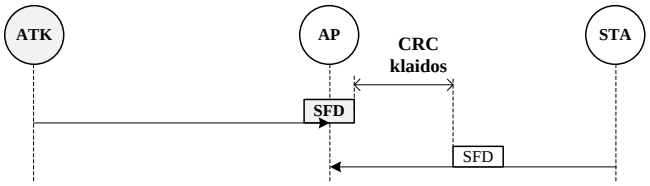


10 pav. *Preambulės ataka*

2.2.1.3. Konvergencijos žymos ataka

Fizinio lygmens konvergencijos procedūros preambulėje yra SFD (angl. *Start Frame Delimiter*) laukas, skirtas žymėti konvergencijos antraštės pradžią. Ši ataka pagrįsta atakuojančio įrenginio signale esančio SFD šablono siuntimu į tinklo prieigos įrangą, prieš ją pasiekiant legaliems signalams (žr. 11 pav.). Priėmęs atakuotojo SFD šabloną, aptarnaujantysis įrenginys nedelsiant pradeda interpretuoti gaunamus klaidingus bitus kaip konvergencijos procedūros antraštę, nors tai atliekama prieš gaunant SFD

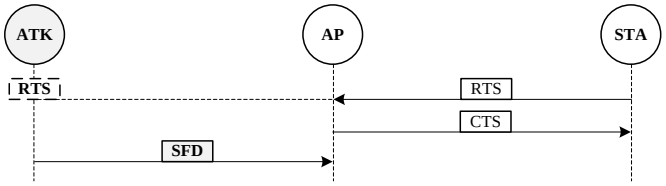
žymą iš legalaus vartotojo. Taikant šį metodą, iškraipoma legalaus signalo bitų tvarka ir gaunamos patikros klaidos tiek konvergencijos procedūros antraštėje, tiek kanalinio lygmens kadre [8].



11 pav. *Konvergencijos žymos ataka*

2.2.1.4. Reaktyvioji ir atsitiktinio trukdžio ataka

Naudojant aktyvius atakų metodus reikalingas pastovus energijos šaltinis, maitinantis trukdžius generuojantį įtaisą, kuris paprastai būna gana didelės galios. Naudojant reaktyviąją ataką, piktybinis įtaisas stebi eterį ir generuoja trukdžius tik tuomet, kai aptinka perduodamus kadrus (žr. 12 pav.). Šis metodas pagrįstas paskirstytąja koordinacijos funkcija DCF (angl. *Distributed Coordination Function*), kuomet bandymas siųsti duomenis aptinkamas nustačius eteryje siunčiamus RTS ar CTS kadrus. Aptikus aktyvumą tinkle, toliau naudojamas tam tikras atakos metodas, pavyzdžiui SFD ataka, o po to vėl laukiama aktyvumo [8].

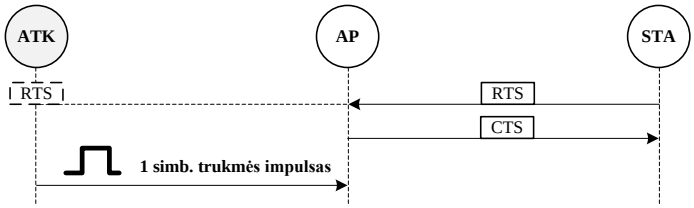


12 pav. *Reaktyvioji ataka*

Atsitiktinio trukdžio ataka pasižymi trukdžius generuojančio įrenginio atsitiktiniu aktyvumu: jeigu įrenginys veikia pastoviai, jo vietą lengva nustatyti, todėl atakuojantis įrenginys sukonfigūruojamas trukdžius generuoti atsitiktiniais laiko momentais. Tiek reaktyvioji, tiek atsitiktinio trukdžio atakos sudarytos siekiant taupyti atakuojančio įrenginio energiją bei apsunkinti jo radimą. Šios atakos daugiau nusako atakos tipą, nei konkretų algoritmą.

2.2.1.5. Simbolių ataka

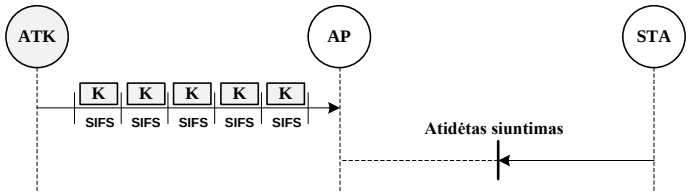
Šis atakos tipas būdingas 802.11 ir 802.11b standarto tinklams, kuriuose nėra numatyta persiuntimo klaidų koregavimo funkcija FEC (angl. *Forward Error Correction*). Tokiu atveju, atsiradus nors vieno simbolio klaidai, yra sugadinamas visas kadras. Atakos mechanizmas pavaizduotas 13 pav. Trukdantis įtaisas, aptikęs aktyvumą, generuoja vieno simbolio trukmės stiprų impulsą. Nesant FEC funkcijai, yra didelė tikimybė, jog pavyks sugadinti visą kadrą [8].



13 pav. *Simbolių ataka*

2.2.1.6. Monopolizacijos ataka

Monopolizacijos ataka (žr. 14 pav.) pagrįsta CSMA/CA mechanizmu: belaidžiame tinkle cirkuliuojantys kadrai atskiriami trumpu laiko tarpu – tarpkadriniu laiku. Trumpas tarpkadrinis laikas SIFS naudojamas atskirti kadrus, kurie yra ankstesnių kadrų komunikacijos dalis (pavyzdžiui, ACK kadrams), o prailgintos trukmės tarpkadrinis laikas DIFS naudojamas tarp mazgų, besirengiančių inicijuoti naują kadrų siuntą. Mažiausias laikas, kurį privalo laukti įrenginiai prieš siunčiant informaciją belaidžiame tinkle yra SIFS. Atakuotuojas gali siųsti trumpus kadrus tarpkadriniu laiku taip monopolizuodamas ryšio kanalą [8].



14 pav. *Monopolizacijos ataka*

2.2.2. Fizinio lygmens apsaugos metodai

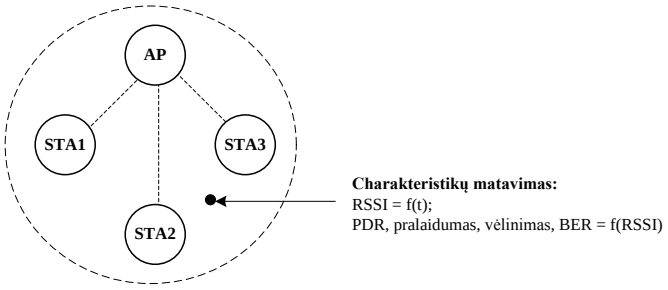
Fizinio tinklo lygmens apsaugos veiksmus galima skirstyti į dvi grupes [8]:

- *Fizinio lygmens atsisakymo aptarnauti atakos identifikacija.*
- *Identifikuotos atakos neutralizavimo priemonių naudojimas.*

Pirmuoju atveju, naudojant atakų nustatymo metodikas, identifikuojama, ar tinklas yra atakuojamas, ar tiesiog blogai funkcionuojantis. Antru atveju, nustačius konkrečią tinklo fizinio lygmens DoS ataką, imamasi konkretaus metodo, kuris apsaugo nuo nustatytos atakos. Šiam atvejui taip pat priskirtini specializuotieji tinklo plėtojimai, skirti tinklą padaryti atsparesnį atsisakymo aptarnauti atakoms fiziniame lygmenyje.

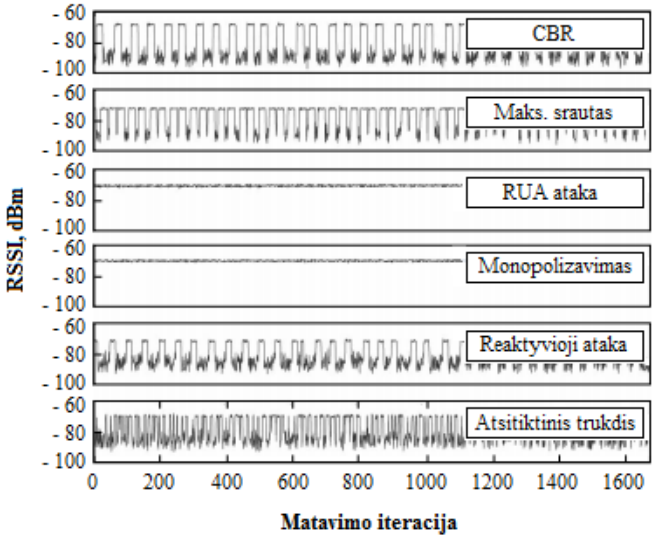
2.2.2.1. Fizinio lygmens atsisakymo aptarnauti atakų identifikacija

Fizinio lygmens atsisakymo aptarnauti atakos veikia manipuliuodamos fiziniais belaidžio tinklo signalo parametrais. Tipinė fizinio lygmens atakų identifikacijos schema pateikta 15 pav. Tiriant minėtuosius parametrus, naudojami specializuoti tinklų testeriai, kurie zonduoja tinklą ir pateikia fizines tinklo charakteristikas [29].



15 pav. *Fizinio lygmens atakų identifikacija*

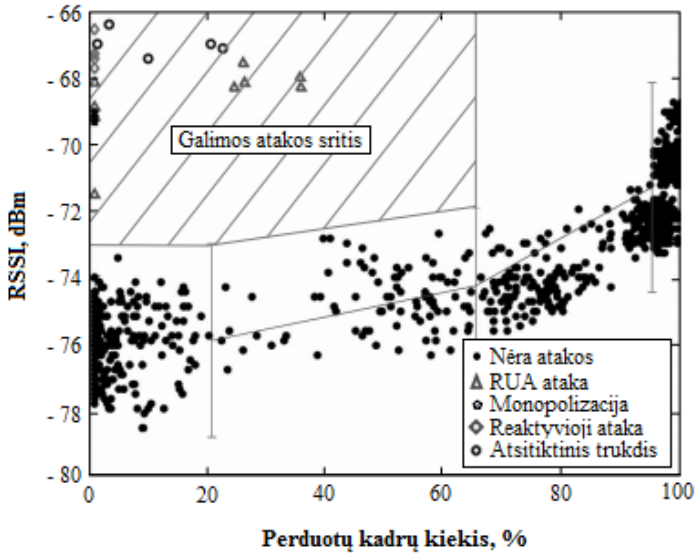
Kiekvienas atakos tipas suformuoja savitą signalo stiprumo priklausomybės nuo laiko funkciją (žr. 16 pav.). Atakos aptinkamos identifikavus anomalius fizinio lygmens parametrus ar charakteristikas. Paprastai, jeigu užtikrinamas stiprus belaidžio ryšio signalas RSSI (angl. *Received Signal Strengh Indication*) ir įrenginiai yra sąlygiškai arti vienas kito, tai kiti fizinio lygmens parametrai būna taip pat geri: mažas bitų klaidų santykis BER (angl. *Bit Error Ratio*), aukštas perduotų paketų santykis PDR (angl. *Packet Delivery Ratio*), didelis pralaidumas ir mažas vėlinimas.



16 pav. *Fizinio lygmens atakų signalų charakteristikos [29]*

Kuomet signalas yra stiprus, bet minėtieji parametrai rodo prastą tinklo veiklą, tikėtina, jog vykdoma DoS ataka. Vienintelis būdas nustatyti atakos vykdymo faktą (ir tipą) yra signalo stiprumo

lyginimas su kitais fiziniais parametrais. Wenyuan Xu, Ke Ma, Wade Trappe ir Yanyong Zhang savo straipsnyje pateikia atakų nustatymo skalę (žr. 17 pav.) pagal fizinius parametrus, gautą remiantis eksperimentų su realia MICA2 moduline įranga rezultatais [29].



17 pav. *Atakų identifikacija lyginant signalo stiprumą su prarastais kadrais [29]*

Autorių atliktų eksperimentų metu buvo vykdomos fizinio lygmens DoS atakos ir matuojami tinklo fizinio lygmens parametrai ties kiekviena ataka. Remiantis matavimais, buvo sudaryta perduotų paketų kiekio priklausomybės nuo signalo sitiprumo skalė, kurioje sužymėtos kiekvienai atakai būdingos šios priklausomybės reikšmės.

2.2.2.2. Fizinio lygmens apsauga ir nustatytų atakų neutralizavimas

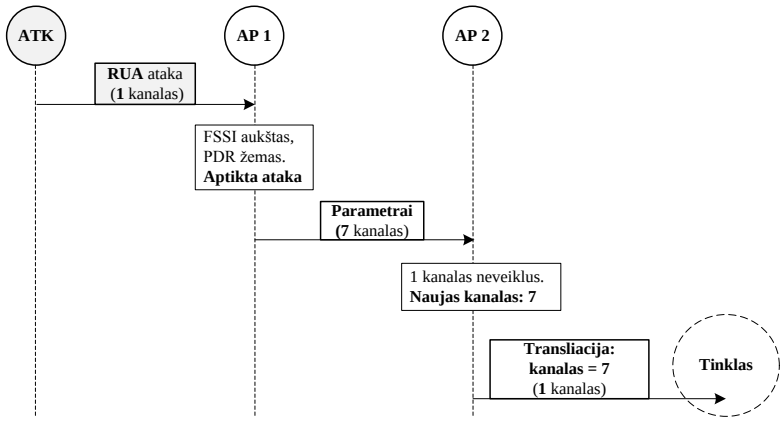
Aptikus vykdomą belaidžio tinklo fizinio lygmens ataką imamasi atsakomųjų priemonių. Fizinio lygmens atakas neutralizuoti galima dviem budais: fiziškai pašalinant ataką sukeliančius įrenginius ir tinklo fizinį lygmenį darant atsparesnį atakoms.

Fizinis įrenginio pašalinimas racionalus, jeigu trukdantį įrenginį įmanoma operatyviai rasti nedideliame tinkle, kuomet įrenginys naudoja neribotų resursų ataką. Kuomet atakuojama iš už tinklo aprėpties zonos dideliame tinkle ir piktavališki įrenginiai naudoja maskuojančiuosius atakos režimus (reaktyviąją ataką, atsitiktinį trukdį), fizinis jų pašalinimas tampa sudėtingas [31].

Šiuo atveju taikomas tinklo atsparumo DoS atakoms didinimas, kurį pagrindžia dvi pagrindinės strategijos:

- *Dažninis atakos išvengimas*.
- *Erdvinis atakos išvengimas*.

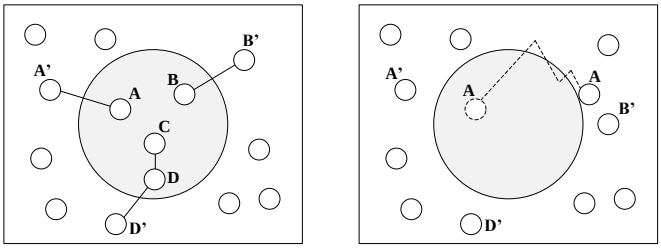
Dažninis išvengimas pagrįstas tinklo kanalo dažnio keitimu (žr. 18 pav.). Dažnis keičiamas pseudoatsitiktine tvarka tarp tinklo periferinių įrenginių: įrenginys, identifikavęs ataką, keičia veiklos kanalą ir nauju kanalu siunčia signalinį kadrą, informuodamas kitus įrenginius. Kiti įrenginiai pradeda zonduoti tinklą visais kanalais ir aptikę naują kanalą, kuriuo dirba pirminis įrenginys, grįžta į pirminį kanalą ir transliaciniu režimu persiunčia visiems tinklo įrenginiams pranešimą naudoti naująjį kanalą [29].



18 pav. *Dažninis fizinio lygmens atakų išvengimas*

Šio metodo trūkumas – reikalingas tam tikras laikas visam tinklui persikonfigūruoti. Jei tinklas didelis, gali kilti nesklandumų su kanalų pasirinkimais, todėl galimi ir daliniai dažnio keitimai tik tose srityse, kurias apima atakos zona [30].

Erdvinis atakos išvengimas (žr. 19 pav.) efektyvus mobiliuosiuose belaidžiuose tinkluose. Jis pagrįstas tinklo mazgų pasišalinimu iš atakuojamos zonos. Paveikslėlio a) dalyje pavaizduota tinklo topologija prieš prasidedant atsisakymo aptarnauti atakai fiziniame lygmenyje (jos zona pažymėta pilkai). Mazgas A tiesiogiai sujungtas su mazgu A', mazgas B – su mazgu B' ir t.t. Po to, kaip prasideda ataka, atakos zonoje esantys mazgai ją identifikuoja naudodami atakos radimo algoritmą ir šalinasi iš atakos zonos atsitiktiniu maršrutu tol, kol palieka atakos zoną. Tuomet mazgai tikrina galimą susijungimą su kitais mazgais naujoje vietovėje ir jei tokios galimybės nėra, juda apie atakuojamos zonos ribas tol, kol vėl susijungia su tinklu. Paveikslėlio b) dalyje pateiktas A mazgo judėjimo maršrutas erdvinio atakos išvengimo procedūros metu [16,17].



19 pav. *Erdvinis atakos išvengimas: a) prieš prasidedant atakai; b) identifikavus ataką [29]*

2.2.3. Kanalinio lygmens atsisakymo aptarnauti atakos

Belaidžio tinklo kanalinio lygmens atsisakymo aptarnauti atakos pagrįstos tinklo žemo lygio logikos mechanizmų (bendros terpės valdymo, autentifikacijos ir kt.) saugumo trūkumais. Atakuojant tinklą kanaliniame lygmenyje, naudojami legalūs fizinio signalo parametrai, todėl šių atakų metu atakuojantiems įrenginiams reikia mažiau energijos nei fizinio lygmens atakavimo atveju. Šių atakų metu naudojami suklastotieji MAC (angl. *Media Access Control*) adresai [6,18].

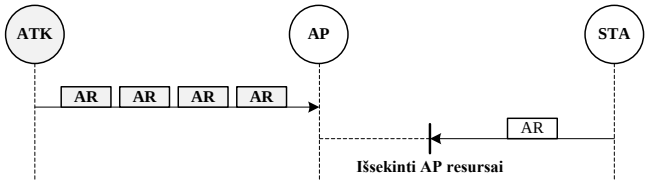
Kanalinio lygmens atsisakymo aptarnauti atakas galima skirstyti į dvi grupes:

- *Selektyviosios atakos* – tai tokios kanalinio lygmens atakos, kai sutrikdomas tik vieno tinklo įrenginio darbas. Paprastai šių atakų tikslas būna sutrikdyti tinklo paslaugas konkretiems vartotojams [8].
- *Pilnosios atakos* – jų metu sutrikdomas viso (arba kiek įmanoma didesnio mazgų skaičiaus tinkle) darbas. Šios atakos paprastai atliekamos prieš pagrindinius tinklo mazgus (prieigos stoteles, kontrolerius), teikiančius paslaugas tinklo vartotojams [8].

Šiame skyrelyje aprašomos populiariausios belaidžio tinklo kanalinio lygmens atsisakymo aptarnauti atakos.

2.2.3.1. Autentifikacijos ir asociacijos užtvindymo ataka

Nepriklausomai nuo naudojamų apsaugos protokolų, belaidės prieigos įranga autentifikuoja kiekvieną klientą (net ir esant atvirai autentifikacijai), jam išskirdama sisteminius resursus. Autentifikacijos ir asociacijos tvindymo atakos metu, atakuotojas siunčia daug autentifikacijos ir asociacijos užklausų prieigos įrangai, taip išeikvodamas įrangos atminties ir skaičiuojamuosius resursus (žr. 20 pav.). Kitiems vartotojams paslaugų teikimas sulėtėja arba iš vis nutrūksta [8].

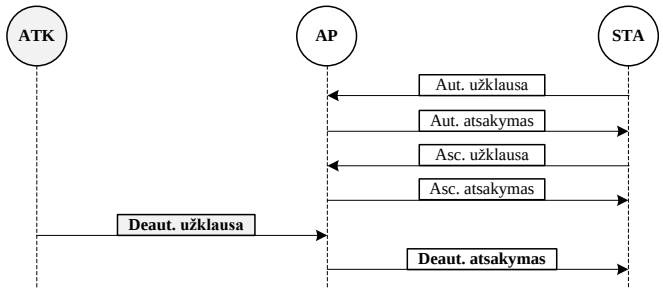


20 pav. *Autentifikacijos ir asociacijos tvindymo ataka*

2.2.3.2. Deautentifikacijos ir diasociacijos ataka

Belaidžio tinklo kanalinio lygmens valdymo kadrai nėra apsaugoti kriptografinėmis autentifikacijos priemonėmis, todėl autentifikacijos ir asociacijos dialogas tarp tinklo vartotojo ir prieigos įrangos yra lengvai pažeidžiamas piktybiniais suklastotais kadrais - prieigos įranga visus paketus, ateinančius iš to

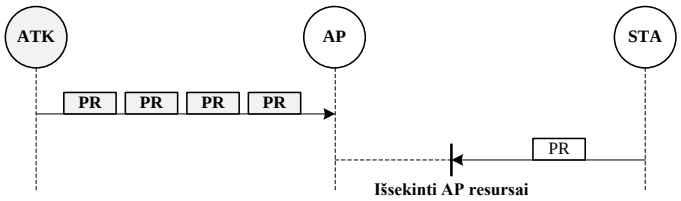
paties MAC adreso, traktuoja kaip legalius. Deautentifikacijos ir diasociacijos atakos metu, atakuojantis įrenginys klausosi tinkle perduodamų valdymo kadrų ir aptikęs įrenginio bandymą autentifikuotis arba asociuotis, siunčia diasociacijos arba deautentifikacijos kadrą to įrenginio adresu, taip jį atjungdamas nuo tinklo (žr. 21 pav.). Analogiškai galima atjungti ir jau įsiregistravusį ir autentifikuotą tinklo įrenginį. Periodiškai atliekant šią procedūrą, atakuojami įrenginiai yra pastoviai atjungti nuo tinklo [8,10].



21 pav. *Deautentifikacijos ir diasociacijos ataka*

2.2.3.3. Zondavimo užtvindymo ataka

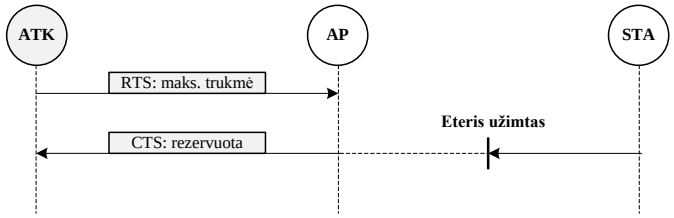
Tinklo vartotojai, skenuodami tinklo paslaugas teikiančią įrangą, naudoja zondavimo kadrus. Prieigos įranga, gavus zondavo užklausą, vartotojui siunčia zondavimo atsaką, kuriame nurodomi tam tikri parametrai, reikalingi vartotojui susijungti su prieigos įranga: kanalas, SSID (angl. *Service Set Identifier*), BSSID (angl. *Basic Service Set Identifier*) ir kiti. Šios atakos metu, atakuojantis įrenginys, siunčia didelį kiekį zondavimo užklausų prieigos įrangai su skirtingais šaltinio MAC adresais (žr. 22 pav.). Prieigos įranga, apdorodama užklausas, eikvoja papildomus atminties ir procesoriaus resursus, todėl legaliems vartotojams tinklo veikla sulėtėja arba nutrūksta [8].



22 pav. *Zondavimo tvindymo ataka*

2.2.3.4. Siuntimo atidėjimo ataka

Siuntimo atidėjimo ataka (žr. 23 pav.) pagrįsta belaidžio tinklo architektūriniu aspektu – bendros terpės valdymo mechanizmu. Belaidžiame tinkle naudojamas bendras eteris, todėl vienoje aprėpties zonoje vienu metu priimamas kadras tik iš vieno vartotojo mazgo. Kadruose nurodoma kiek laiko užims esama komunikacija. Atakuojantis įrenginys nustato kuo ilgesnes laiko reikšmes savo kadro „duration“ lauke, taip legaliem tinklo įrenginiam padidindamas laukimo etapą iki siuntimo [8].

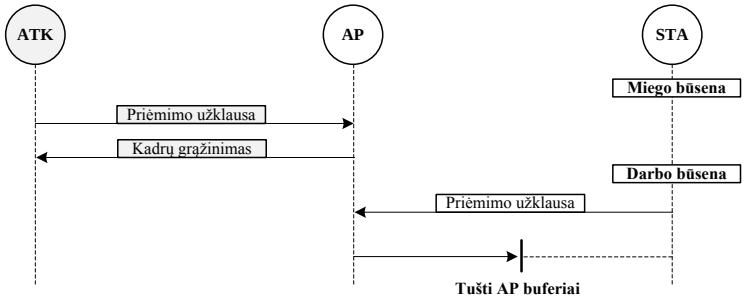


23 pav. *Siuntimo atidėjimo ataka*

2.2.3.5. Energijos taupymo režime esančių mazgų ataka

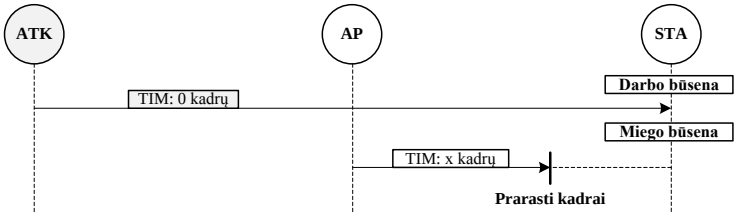
IEEE 802.11 belaidžio tinklo standartas numato energijos taupymo režimą. Klientai, taupydami energiją, pereina į „miego“ režimą, kurio metu negali nei priimti, nei išsiųsti kadrų, prieš tai apie tai pranešę prieigos įrangai. Prieigos įranga kaupia konkrečiam mazgui, esančiam energijos taupymo režime, skirtus kadrus. Mazgas, grįžęs į normalų darbo režimą, siunčia bevielės prieigos stotelei paketų atgavimo kadrą ir priima jam sukauptus paketus, o stotelė išvalo savo atmintį Pasinaudojant tuo, jog miego režimo valdymo komunikacija yra atvira, galimi keli šiuo mechanizmu pagrįsti DoS atakų atvejai [8,10].

Atakuotojas, apsimesdamas tam tikru legaliu vartotojo mazgu, kol legalus mazgas yra energijos taupymo režime, siunčia paketų surinkimo kadrą stotelei (žr. 24 pav.). Stotelė atiduoda sukauptuosius kadrus ir išvalo buferius, tokiu būdu legalus mazgas netenka jam skirtų kadrų.



24 pav. *Klaidingo kadrų buferio atlaisvinimo ataka*

Atakuotojas, apsimesdamas prieigos įranga, siunčia suklastotus kadrus su TIM (angl. *Time Identification Map*) žyma, kurioje nurodoma, kad klientui nėra sukaupta jokių kadrų, nors kadrai yra sukaupti (žr. 25 pav.). Klientas toliau pereina į energijos taupymo režimą negavęs savo kadrų.



25 pav. *Kadrų su TIM lauku klastojimas*

2.2.4. Kanalinio lygmens apsaugos metodai

Visi kanalinio lygmens atsisakymo aptarnauti antpuoliai remiasi tuo, jog trukdantysis įrenginys turi galimybę prieiti prie tinklo išteklių naudodamas tam tikrą suklastotą kadrą, todėl daugumos kanalinio lygmens apsaugos algoritmų principai remiasi kanalinio lygmens kadrų autentifikacija ir MAC adresų valdymu realizuota tinklo prieigos kontrole [21,23].

2.2.4.1. Kadrų klastojimo nustatymas

Kadrų klastojimo nustatymo metodai paprastai naudojami tinklų įsibrovimo nustatymo ar prevencijos sistemose IDS/IPS (angl. *Intrusion Detection/Prevention System*). Pagrindiniai metodai, aprašomi mokslinėje literatūroje:

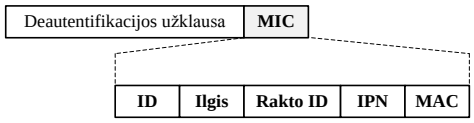
- *Legalaus MAC adreso nustatymas.* Visi 802.11 tinklo standartą atitinkančias tinklo plokštes gaminantys gamintojai privalo gauti unikalų kanalinio lygmens adreso identifikatorių iš IEEE organizacijos. Identifikatorius – tai pirmi trys MAC adreso baitai, nurodantys konkretų gamintoją, o likusius 3 gamintojas gali keisti savo nuožiūra. Vadovaujantis šiuo požymiu, tinkle siunčiamus kadrus galima tikrinti pagal tai, ar jų MAC adresai yra legalūs pagal identifikatorių t.y. ar pirmi trys baitai yra išduotų identifikatorių sąraše. Nors toks metodas veiksmingas prieš atsitiktiniu MAC adresų generavimu pagrįstas DoS atakas, tačiau dabartiniai atakų algoritmai sugeba keisti MAC adresus taip, jog jie atitinka išduotus identifikatorius [23].
- *Sekos numerio tikrinimas.* 802.11 standartas numato 16 bitų kadro antraštės lauką, skirtą kadrų sekos kontrolei. Laukas padalintas į 4 bitų lauką, žymintį fragmento numerį, ir 12 bitų lauką, žymintį sekos numerį. Jeigu kadrą prireikia skaidyti į kelis t.y. atlikti fragmentaciją, fragmento numeriai didinami po 1, o sekos numeris paliekamas toks pats. Priešingu atveju, jeigu siunčiami nefragmentuoti kadrai, fragmento numeris būna lygus 0, o sekos numeris didinamas po 1, kas kiekvieną kadrą iki 4096. Sekos numerio lauko reikšmę gali keisti tik tinklo adapterių žemiausio lygio programinė įranga saugoma lustuose (angl. *Firmware*), todėl norint jį pakeisti, reiktų keisti lustų programų funkcionalumą bei turėti tų programų išeities kodą, kuris paprastai būna uždaras ir slepiamas kaip gamintojo komercinė paslaptis. Dėl šių priežasčių įsibrovėliui techniškai sudėtinga keisti sekos numerį ir MAC adresų klastojimas gali būti nustatytas tikrinant sekos numerius pagal ne įprastas jo kitimo charakteristikas, pavyzdžiui, gaunant kadrus su tuo pačiu MAC adresu bet dideliu tarpu

besiskiriančiais sekos numeriais. Būtent šio tipo algoritmai bus tiriami šiame darbe, o detalūs algoritmų aprašymai pateikti 2.3 skyriuje [10,25,32].

- *Diasociacijos atidėjimas.* Pagal standartą, diasociacija vykdoma siunčiant vieną kadrą. Diasociacijos atakos naudoja seriją diasociacijos kadrų vienu aukos MAC adresu. Atidedant diasociaciją 5-10 sekundžių ir stebint, ar bus gauta papildomų kadrų, galima nustatyti, ar diasociacija teisinga [4].
- *RARP protokolas* (angl. *Reverse Adress Resolution Protocol*). Klastojant MAC adresus, tinkle gaunami tie patys MAC adresai prie skirtingų IP adresų. Naudojantis šiuo požymiu, RARP užklausų pagalba, galima nustatyti tokius MAC adresus. Metodas nėra patikimas, nes dažniausiai klastojami ir IP adresai [4].
- *Santraukos funkcija valdymo kadrams.* Esami WPA/WPA2 sprendimai neautentifikuoja valdymo kadrų. Išplėtus IEEE 802.11i (WPA2) standartą, kiekvieną valdymo kadrą galima autentifikuoti santraukos funkcija (angl. *Hash*). Metodas remiasi didelių skaičių sandaugos skaidymo dauginamaisiais problema [4].

2.2.4.2. IEEE 802.11w standartas

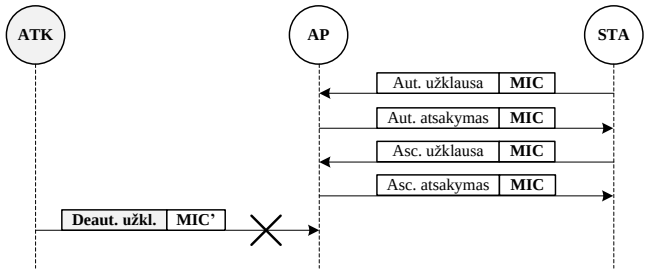
IEEE organizacija patvirtino 802.11w belaidžių tinklų standartą, kaip 802.11standarto papildymą. Šio standarto tikslas yra užtikrinti belaidžio tinklo valdymo kadrų saugumą. Standarto teikiami apsaugos metodai užkerta kelią diasociacijos ir deautentifikacijos atsisakymo aptarnauti atakoms kanaliniame lygmenyje. Esminis 802.11w standarto ypatumas atsisakymo aptarnauti atakų atžvilgiu – valdymo kadrų integruotumo ir autentiškumo apsaugojimas naudojant papildomus integruotumo kodus MIC (angl. *Message Integrity Code*), kurių struktūra pateikta 26 pav. Šis kodas sudaromas naudojant bendrą slaptą raktą tarp tinklo prieigos įrangos ir vartotojo įrangos [16].



26 pav. *Kadro saugos lauko MIC struktūra*

- *ID* – informacijos elemento numeris.
- *Rakto ID* – nurodo rakto numerį, generuojant MIC.
- *IPN* – naudojamas atsakymo apsaugai. Didinamas teigiamu skaičiumi kas kadrą (angl. *IGTK Packet Number*).
- *MAC* – šifruota santraukos funkcija (angl. *Message Authentication Code*).

Kadangi deautentikacijos pranešimai yra apsaugoti integruotumą ir autentiškumą užtikrinančiu MIC kodu, nepakanka žinoti vien MAC adreso, norint legalų vartotoją atjungti nuo tinklo naudojant piktybinius deautentifikacijos ir diasociacijos kadrus (žr. 27 pav.).



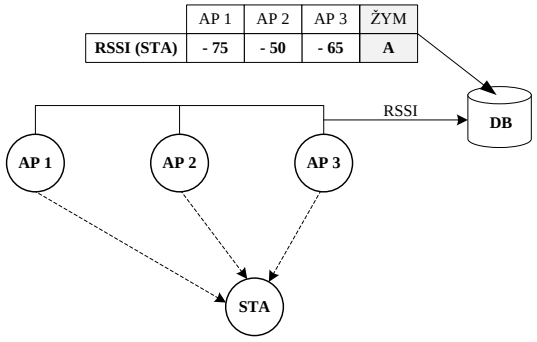
27 pav. *Kadro MIC lauko panaudojimas apsaugai nuo deautentifikacijos*

Šis apsaugos metodas neapsaugo nuo užtvindymo atakų, nes jos vykdomos MIC kodu neapsaugotais kadrais.

2.2.4.3. Signalo charakteristikų žymos

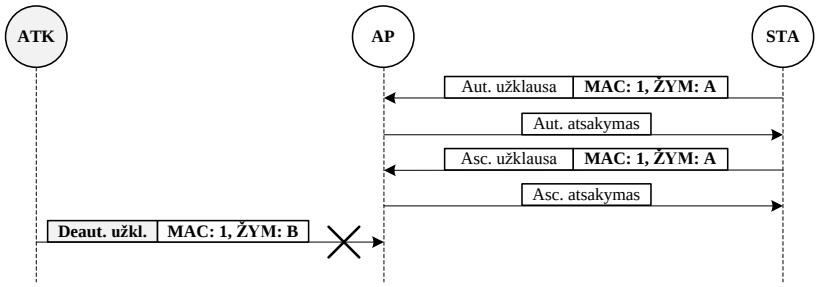
Šis metodas skirtas autentifikuoti tinkle esančius mazgus pagal jų generuojamų signalų charakteristikas, siekiant užkirsti kelią nelegalaus įrenginio bandymui gauti prieigą prie tinklo ir atlikti atsisakymo aptarnauti ataką. Signalo charakteristikų panaudojimo schema pateikta 28 paveikslėlyje.

Vadovaujantis prielaida, jog tinklo mazgų būsena tinkle yra stacionari, kiekvieną iš mazgo ateinantį signalą galima apibūdinti vidutine jam būdinga signalo stiprumo reikšme priėmimo vietoje RSSI, matuojama dBm. Jeigu tinkle yra naudojamos kelios belaidės prieigos stotelės, kiekviena iš jų gali matuoti įrenginio siunčiamų signalų stiprumą ir perduoti duomenis į centralizuotą mazgą – serverį, arba tiesiog didesnės skaičiuojamosios galios belaidės prieigos stotelę. Naudojant atitinkamą algoritmą, suformuojamas visų priimtų reikšmių masyvas, kuris ir naudojamas mazgų autentifikavimui pagal signalo lygį [6,27].



28 pav. *Tinklo mazgų autentifikacija pagal jų signalų charakteristikas*

Tokiu būdu, tam tikrus valdymo kadrus galima autentifikuoti naudojantis šiomis žymomis ir užkirsti kelią diasociacijos ir deautentifikacijos atakoms, atmetant deautentifikacijos ir diasociacijos užklausų kadrus, kurie netenkina signalo žymų, nors MAC adresas yra suklastotas (žr. 29 pav.).



29 pav. *Autentifikacijos pagal signalus panaudojimas apsaugai nuo deautentifikacijos*

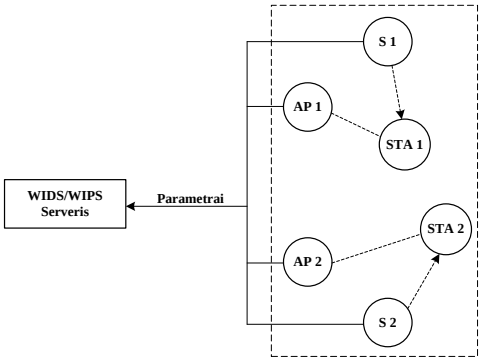
2.2.4.4. Belaidžio tinklo įsibrovimo nustatymo ir prevencijos sistemos

Tai ne konkretus metodas apsaugai nuo konkrečių atsisakymo aptarnauti atakų, o kompleksinis sprendimas, realizuotas aparatine ir programine įranga bei apjungiantis daugelį apsaugos algoritmų į viena sistemą [32].

Skiriamos dvi sistemų rūšys:

- *Įsibrovimo nustatymo sistemų* WIDS (angl. *Wireless Intrusion Detection System*) tikslas yra identifikuoti tinkle įsibrovimus ir apie tai informuoti atsakingą asmenį.
- *Įsibrovimo prevencijos sistemų* WIPS (angl. *Wireless Intrusion Prevention System*) tikslas yra užkirsti kelią sėkmingam įsibrovimui, aptikus bandymo įsibrauti požymius.

Šių sistemų konceptinė ir struktūrinė sudėtis priklauso nuo konkretaus tinklo ir konkrečių reikalavimų. 30 pav. pateikta schema vaizduoja apibendrintąjį jų veikimą. Šias sistemas sudaro belaidžiai davikliai, kurie skenuoja radijo signalus tinklo aprėpties zonoje ir informaciją siunčia į centralizuotą mazgą – sistemos serverį, kuriame daviklių priduota informacija yra toliau apdorojama pagal sudarytas saugumo taisykles. Esant tikėtinam įsilaužimui ar kitam incidentui, generuojamas aliarmas [14,28].



30 pav. *WIDS/WIPS sistemos schema*

2.2.5. Aukštesnių lygmenų atsisakymo aptarnauti atakos

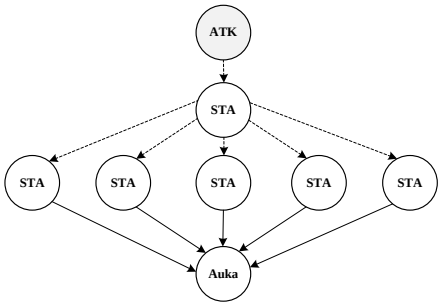
Fizinio ir kanalinio lygmens atakos yra glaudžiai susijusios su tinklo architektūra, kurią apibrėžia standartas IEEE 802.11-2007. Šių atakų metu išnaudojamos konkretaus standarto belaidžio tinklo signalų charakteristikos, tinklo valdymo ir duomenų srauto kontrolės mechanizmai, kadrų formate esantys saugos ribotumai.

Tarpinę padėtį tarp kanalinio ir tinklo lygmenų užima ARP (angl. *Address Resolution Protocol*) transliacijos audros ataka, kuri realizuojama puolančiajam teikiant ARP užklausas su neegzistuojančiais IP adresais. Tinkle kyla transliacijų audra, nes mazgai vienas po kito transliaciniu režimu siuntinėja užklausas apie nežinomus IP adresus [24].

Tinklo ir transporto lygmens atsisakymo aptarnauti atakų ypatumai remiasi TCP/IP platformos, kuri naudojama belaidžiame ir kitų standartų tinkluose, pažeidžiamumais. Saugos ribotumai atsiranda dėl nesaugios ICMP (angl. *Internet Control Message Protocol*), IP, TCP, UDP ir kitų protokolų architektūros [24].

Sesijos pateikimo ir taikomojo lygmenų atakų pobūdis remiasi konkrečios taikomosios programinės įrangos, naudojamos tinklo klientų ir tinklo tarpinių įrenginių, atsisakymo aptarnauti atakų pažeidžiamumais, o taip pat ir taikomųjų protokolų pažeidžiamumais.

Dėl aukštesnių lygmenų architektūrinių savybių, kurios yra atskirtos nuo konkretaus tinklo standarto, atakuotojai gali naudoti kitų standartų tinklų resursus ir ataką atlikti iš didelio skaičiaus atakuojančių mazgų. Tokios atakos vadinamos DDoS (angl. *Distributed Denial of Service*) arba paskirstytomis atsisakymo aptarnauti atakomis (žr. 31 pav.). Jų esminis požymis – tai didelis užkrėstų kompiuterių kiekis piktybine programine įranga (angl. *Daemon*), kuri pastoviai laukia signalo tam tikru transportinio lygmens prievadu, ir gavusi signalą, pradeda ataką. Signalą generuoja centrinis mazgas, o visi užkrėstieji mazgai atakuoja vieną aukos mazgą ar sistemą [12,19].



31 pav. *Taikomojo lygmens DDOS atakos schema*

Aukštesnių nei kanalinis lygmenų atakos nėra tiesiogiai susijusios su IEEE 02.11 standartu, todėl šiame skyrelyje jos apžvelgiamos bendrais aspektais.

2.2.5.1. ICMP protokolo atakos

ICMP protokolas buvo sukurtas tinklo įrenginiams informuoti vienas kitą apie tinkle iškilusias problemas ICMP žinučių pagalba, tačiau specialiai siunčiant ICMP žinutes su nustatytais klaidingais pranešimais arba siunčiant didelį kiekį ICMP žinučių į tam tikrą tinklo mazgą, galima sulėtinti arba nutraukti mazgo darbą [21,24].

Būdingiausios ICMP atsisakymo aptarnauti atakos:

- *Šaltinio sulėtinimas* – siunčiant ICMP paketus su nustatyta lauko žyme „šaltinio pristabdymas“, sulėtinama šaltinio sparta.
- *Siuntimo nutraukimas* realizuojamas siunčiant ICMP paketus su nustatyta „adresatas nepasiekiamas“ žyme. Siunčiantysis įrenginys gali būti klaidingai informuotas, jog šaltinis negali priimti paketų ir nutraukti sujungimą.
- *„Mirtina“ ataka* (angl. *Ping of Death*) realizuojama siunčiant didesnį nei 65535 baitų dydžio ICMP užklausos paketą. Atakuojamas įrenginys gavęs tokias užklausas persikrauna arba pakimba.
- *Užtvindymo ataka* (angl. *Ping Flood*) realizuojama ICMP paketą siunčiant transliaciniu režimu į visus tinklo mazgus, nurodant suklastotą aukos sistemos IP adresą. Visi įrenginiai atsakinėja į užklausą, gaunamas ICMP atsakymų perteklius ir sutrikdomas aukos įrenginio darbas dėl perkrovos.

2.2.5.2. IP protokolo atakos

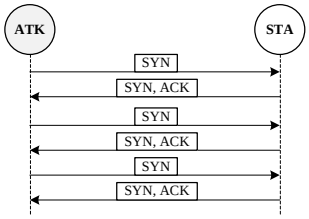
Naudojant IP protokolą, organizuojamos šios atakos:

- *IP tvindymas* – tai paskirstytoji atsisakymo aptarnauti ataka. Tinklo vartotojų kompiuteriai apkrečiami piktybine programine įranga, kuri gavus signalą siunčia IP užklausas į vieną mazgą. Mazgas patiria perkrova ir nustoja funkcionavęs [24].
- *IP fragmentavimas* realizuojamas tikslingai blogai surenkant IP paketus iš fragmentų, siekiant gauti fragmentų persidengimą, įtalpinimą ar ilgio viršijimą. Jeigu tokius paketus priimantysis mazgas neturi specialiai šiai atakai atspariai suprogramuoto TCP/IP modulio, mazgas persikrauna arba pakimba [24].

2.2.5.3. TCP protokolo atakos

TCP yra persiuntimo kokybę užtikrinantis transportinio lygmens protokolas, atsisakymo aptarnauti atakos naudojasi šio protokolo komunikacinių dialogų tarp mazgų trūkumais [21,24].

- *SYN užtvindymas* (žr. 32 pav.) realizuojamas siunčiant didelį kieki TCP paketų su nustatyta SYN vėliavėle į atakuojamą mazgą. Atakuojamas mazgas ties kiekvienu gautu SYN paketu rezervuoja sisteminius resursus kitiems dialogo etapams ir tęsiant ataką atakuojamo mazgo resursai išsenka, jis tampa neveiklus.



32 pav. *TCP sujungimo tvindymo ataka*

- *Sujungimo atsisakymo* ataka realizuojama TCP sujungimo sudarymo etape naudojant IP paketus su vienodais šaltinio ir paskirties adresais. Šiuo atveju sistema bando jungtis pati su savimi ir esant nenumatytai situacijai sistemos TCP/IP programiniuose moduliuose, sistema gali persikrauti arba pakibti.
- *TCP sujungimo nutraukimas* realizuojamas atakuotojui siunčiant TCP paketą su nustatyta RST (angl. *Reset*) arba FIN (angl. *Final*) vėliavėle aukos sistemai. Jeigu puolantysis nurodo teisingą sekos numerį, sistema išardo susijungimą.

2.2.5.4. Taikomojo lygmens protokolų ypatumai

Taikomojo lygmens atsisakymo aptarnauti atakos realizuojamos siunčiant didelius tam tikro taikomojo lygmens protokolų užklausų kiekius, kuriuos aptarnauja procesai tarnybinėse stotyse [24].

- *HTTP užtvindymo* ataka realizuojama siunčiant didelį kiekį HTTP (angl. *Hyper Text Transfer Protocol*) protokolo paketų su HTTP GET metodu. Ši ataka išeikvoja daug skaičiuojamosios įrenginio galios, todėl pastoviai siuntinėjant HTTP užklausas į tinklo įrenginių WEB konfigūravimo sąsaja teikiančius procesus, galima sulėtinti jų veiklą.
- *FTP prisijungimo užtvindymo* ataka realizuojama siunčiant didelį kiekį FTP (angl. *File Transfer Protocol*) sujungimo inicijuoti raginančių paketų į atakuojamą mazgą ir išnaudojant atakuojamo mazgo resursus.

2.2.6. Aukštesnių lygmenų apsaugos metodai

Tipinės fizinio ir kanalinio lygmens atakos paprastai vykdomos is vieno piktybinio mazgo, nes atakuojamas tinklas yra apibrėžtas vienu standartu ir atakuotojas fiziškai negali atakuoti tinklo, naudodamas kitų standartų tinklų resursus.

Aukštesni lygmenys yra labiau pažeidžiami dėl to, jog 802.11 standarto belaidis tinklas naudoja standartinę TCP/IP platformą, o sesijos, pateikimo ir taikomojo lygmenis realizuoja konkretūs programiniai taikomieji procesai. TCP/IP platformą naudoja ir kitų standartų tinklai, pavyzdžiui, IEEE 802.3 (Ethernet), todėl atsiranda papildoma galimybė atakuoti belaidžio tinklo įrenginius pasitelkiant ir kitų tinklų resursus – naudojant paskirstytąją atsisakymo aptarnauti ataką. Dauguma aukštesniųjų lygmenų apsaugos algoritmų yra būtent paskirstytųjų atsisakymo aptarnauti atakų apsaugos algoritmai.

2.2.6.1. Bendroji aukštesnių lygmenų apsauga

Šiai dienai, daugelis autorių mano, jog nėra sukurta efektyvių saugos algoritmų, kurie visiškai apsaugotų nuo pilnųjų paskirstytųjų atsisakymo aptarnauti atakų – ir tai yra didelė problema, tačiau yra tam tikri bendrieji apsaugos metodai, pagerinantys tinklo mazgų saugumą šių atakų atžvilgiu [9].

Tipiniai aukštesnių lygių apsaugos nuo DoS metodai [24]:

- *Užkardos* bendrąja prasme filtruoja tinklo srautą ir tikrina jo legalumą, taip pat dažnai būna integruotos daugumoje maršrutizatorių. Priklausomai nuo užkardos tipo, naudojant filtravimo taisykles, paremtas tinklo, transporto ir aukštesnių lygmenų paketų antraštine informacija, galima sumažinti paskirstytųjų atsisakymo aptarnauti atakų galimybę draudžiant tam tikrų potinklių prieigą, neleidžiant paketų su specifiniais laukų nustatymais ar netinkamais dialogo proceso vykdymais. Užkardose taip pat galima riboti paskiriems mazgams išskiriamus resursus ir kt.
- *Įsibrovimo nustatymo sistemos* analizuoja tinklo srautą ir taikydamos atakos identifikavimo algoritmus, praneša aliarmu apie galimą ataką. Šios sistemos automatiškai aptinka nelegalų srautą pagal netaisyklingai sudarytus paketų laukus arba numatytąsias saugumo taisykles.
- *Įsibrovimo prevencijos sistemos* veikia analogiškai nustatymo sistemoms, tačiau jų pagrindinis tikslas yra užkirsti kelią prasidėjusiai atakai.
- *Tinklo įrenginių konfigūracija* svarbi tuo, jog gali panaikinti tam tikrą tinklo pažeidžiamumą, paremtą nereikalingu funkcionalumu, ir užkirsti kelią atsisakymo aptarnauti atakoms. Pagrindiniai konfigūraciniai aspektai:

- IP transliavimo išjungimas tinkle ir maršrutizatorių ICMP paketų ribojimas per tam tikrą laiko tarpą užkerta kelia ICMP užtvindymo atakai. Užkardose ICMP žinučių su „šaltinio sulėtinimas“ blokavimas ir tarpinių tinklo įrenginių funkcijos reaguoti į šį pranešimą išjungimas užkerta kelią šaltinio sulėtinimo atakoms.
- Nereikalingų ir nenaudojamų paslaugų išjungimas padeda išvengti UDP ir TCP tvindymo tam tikrais prievadais. „Timeout“ intervalo mažinimas ir ribotų sujungimo kiekio nustatymas TCP dialogo procesuose sumažina TCP SYN atakos padarinius.
- Saugumo programinių priemonių naudojimas pašalina programinius atsisakymo aptarnauti atakų pažeidžiamumus, o antivirusinės sistemos apsaugo nuo galimų piktybinių programų, naudojamų paskirstytose atakose. Saugumo lygį kelia kiekvienos atskiros programos saugumo nustatymų tinkamas derinimas, atnaujinimai.

2.2.7. Bendrųjų apsaugos metodų apibendrinimas

Visų iki šiol išanalizuotų atakų ir jų apsaugos metodų apibendrinimas pateiktas lentelėje Nr. 3.

Lentelė Nr. 3 *Bendrųjų belaidžio tinklo apsaugos metodų apibendrinimas*

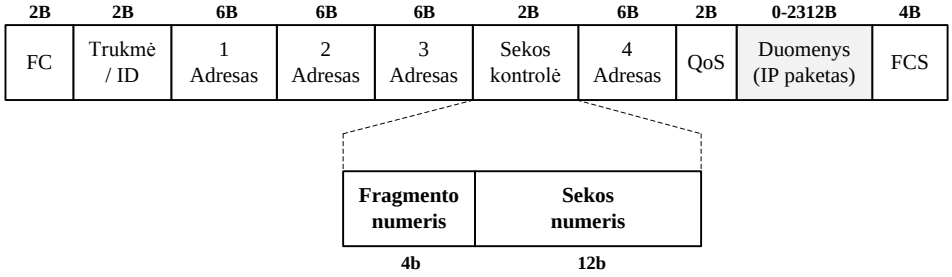
Lygmuo	Apsaugos metodas	Malšinamos atakos	Sudėtingumas
Aukštesni lygiai (ne standarto)	Užkardos	ARP audra TCP tvindymas ICMP protokolo atakos IP tvindymas	Vidutinis/Aukštas
	Įsibrovimo prevencijos sistemos	IP fragmentacija IP tvindymas	Vidutinis/Aukštas
	Tinklo įrenginių saugi konfigūracija: transliavimo išjungimas, ICMP ribojimas, nenaudojamų paslaugų išjungimas ir kt.	ARP audra TCP tvindymas ICMP protokolo atakos TCP sesijos nutraukimas HTTP/FTP tvindymas	Žemas
	Programinės priemonės: sistemų atnaujinimai, antivirusinės sistemos ir kt.	OS/Firmware DoS exploit Taikomojo lygio DDOS	Žemas
Kanalinis	Sekos numerio tikrinimo algoritmai: GAP, SNRA, FRR	Autentifikacijos tvindymas Deautentifikacija Zondavimo tvindymas Siuntimo atidėjimo ataka Energijos taupymo režimo kompromitacija	Vidutinis
	Signalo charakteristikų žymės		Vidutinis/Aukštas
	Kadrų autentifikacimas santraukos funkcijomis		Aukštas
	Diasociacijos atidėjimas		Žemas
	RARP tikrinimas		Žemas
	MAC adreso legalumo nustatymas		Žemas
Fizinis	Erdvinis išvengimas	Neribotų resursų Preambulės Konvergencijos žymos Atsitiktinis trukdis Simbolių ataka Monopolizacija	Aukštas
	Dažninis išvengimas		Aukštas
	Fizinis atakuojančio įrenginio pašalinimas		Aukštas

Lentelėje Nr. 3 atakų apsaugos metodai įvertinami kokybiškai, pagal jų realizavimo sudėtingumą: *Žemas* – metodo realizacijai užtenka pakoreguoti įrenginių konfigūraciją; *Vidutinis* – realizacijai reikalingi papildomi programiniai paketai ir konfigūracija; *Aukštas* – realizacijai reikia naujos aparatinės ir programinės įrangos, reikalingas standarto keitimas ar darbo režimas ne standartinėmis specifikacijomis.

2.3. Operacijomis su kadro sekos numeriu paremtų apsaugos algoritmų analizė

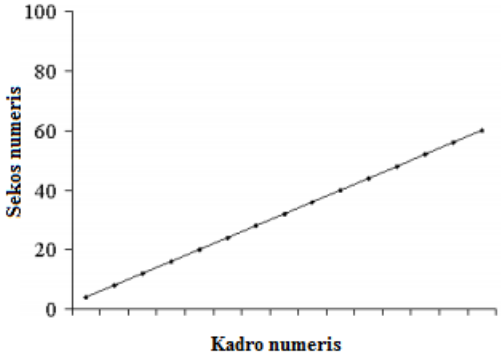
Ankstesniuose skyriuose buvo aprašytos standarto fizinio ir kanalinio lygių bei aukštesnių lygių tipinės atsisakymo aptarnauti atakos ir apsaugos algoritmai nuo jų. Remiantis šiuo aprašymu, galima konstatuoti faktą, jog yra daug ir įvairių metodų, kaip sutrikdyti ar visiškai nutraukti belaidžio tinklo veiklą įvairiose OSI lygiuose, todėl ir apsaugos algoritmai turėtų būti diegiami visose šiuose lygiuose. Deja, tačiau saugos metodų realizacija fiziniame ir kanaliniame lygiuose reikalauja standarto arba jo realizacijos keitimo. Siekiant išvengti praktinių problemų dėl standarto ar jo realizacijos keitimo, tyrimas bus atliekamas modeliuojant algoritmų veiklą, o tiriami bus tik tokie kanalinio lygio algoritmai, kurių veikimas paremtas 802.11 standarto tinklo kadro sekos numerio SEQ (angl. *Sequence number*) tikrinimu.

IEEE 802.11 standartas numato 16 bitų kadro lauką, skirtą kadrų sekos numeriui žymėti (žr. 33 pav.). Laukas padalintas į 4 ir 12 bitų laukus – framgento ir sekos numeriui saugoti. Jeigu kadrą prireikia skaidyti į kelis t.y atlikti fragmentaciją, fragmento numeriai didinami po 1 ties kiekvienu fragmentu, o sekos numeris paliekamas toks pats. Priešingu atveju, jeigu siunčiami nefragmentuoti kadrai, fragmento numeris būna lygus 0, o sekos didinamas po 1 kas kiekvieną kadrą iki 4096 [10].

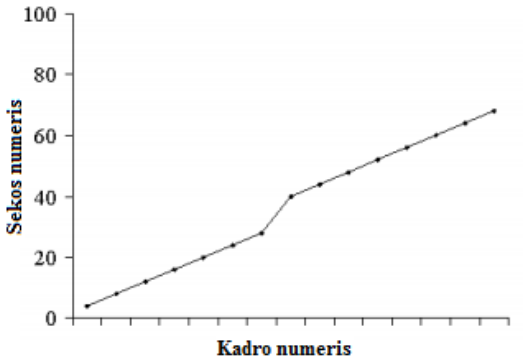


33 pav. *Kadro sekos kontrolės lauko struktūra [15]*

Įsibrovėliui techniškai sudėtinga keisti sekos numerį, todėl MAC adresų klastojimas gali būti nustatytas tikrinant sekos numerius. Idealiu atveju, sekos numeris turėtų kisti tiesiškai, priklausomai nuo kadrų skaičiaus, ir periodiškai kartotis kas 4096 kadrai (žr. 34 pav.). Atsitiktinai pasitaikantys kadrų praradimai koreguoja šią priklausomybę ir kitimas gaunamas nenuoseklus (žr. 35 pav.).

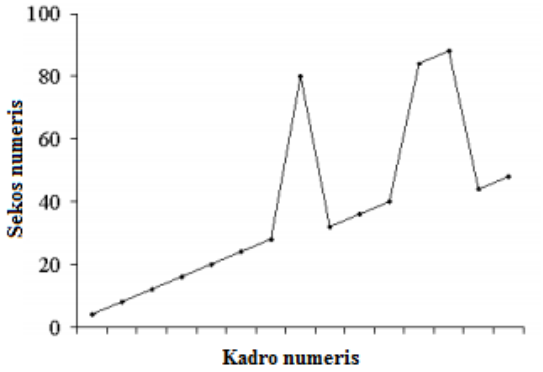


34 pav. *Tipinis sekos numerio kitimas [4]*



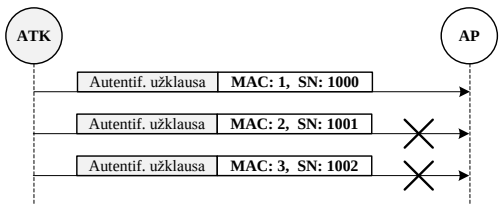
35 pav. *Sekos numerio su praradimais [4]*

Vykdant kanalinio lygmens atakas, pagrįstas MAC adresų klastojimu, iš to paties adreso siunčiami kadrai šio dėsningumo nesilaiko ir gaunami sekos numerio kitimo iškraipymai. 36 pav. pateiktas MAC adreso klastojimo atvejo sekos numerio kitimo grafikas. Grafike pastebimas kiekvienas nukrypimas nuo idealios kitimo kreivės gali būti traktuojamas kaip klastojamo MAC aptikimas su tam tikra tikimybe.



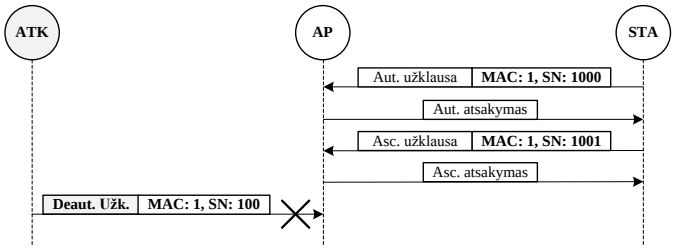
36 pav. *Sekos numerio kitimas esant atakai [4]*

Minėtąsias sekos numerio kitimo savybes galima panaudoti realizuojant apsaugos nuo atsisakymo aptarnauti algoritmus. Esant tvindymo atakai, prieigos įranga identifikuoja, jog skirtingų MAC adresų kadrai pasižymi nuosekliu sekos numerio kitimu. Vadinasi, tokias užklausas galima atmesti, kaip nelegalias (žr. 37 pav.).



37 pav. *Sekos numerio analizės panaudojiams apsaugai nuo autentifikacijos tvindymo*

Esant deautentifikacijos arba diasociacijos atakai, prieigos įranga patikrina, ar deautentifikacijos/diasociacijos užklausos kadro sekos numeris yra nuoseklus: jeigu ne, kadras nustatomas kaip nelegalus ir užklausa ignoruojama (žr. 38 pav.).



38 pav. *Sekos numerio analizės panaudojimas apsaugai nuo deautentifikacijos*

Sekos numerio analize paremtų apsaugos prieš DoS atakas algoritmų veiklą galima dalinti į dvi sudedamąsias dalis:

- *Atakos identifikavimas* – algoritmo dalis, nustatanti, ar vykdoma ataka. Ši dalis yra analogiška algoritmams, kuriuos naudoja atakų aptikimo ir prevencijos sistemos, tačiau esminis skirtumas tas, kad vietoje aliarmo generavimo, šiuo atveju yra atliekami aktyvūs atakos užkirtimo veiksmai. Ši dedamoji pasižymi didesniu teoriniu pagrįstumumu, apibrėžianti kaip apskritai galima nustatyti ataką tinkle su tam tikra tikimybe, operuojant sekos numerio kitimo matematiniais įvertinimais.
- *Atakos neutralizavimas* – algoritmo dalis, kuri atlieka specifinius atakos neutralizavimo veiksmus, tokius kaip MAC filtravimas, paketų naikinimas ar ignoravimas. Ši dedamoji yra konkrečios realizacijos aspektas, nes aktyvūs neutralizavimo veiksmai glaudžiai susieti su įrangos, kuri atliks neutralizavimą, kanalinio lygio valdymo funkcionalumu ir programine struktūra.

Sekančiuose skyreliuose (2.3.1 – 2.3.5) pateikiama esamų naudojamų ir siūlomų algoritmų, paremtų sekos numerio stebėjimu, analizė. Pažymėtina, kad analizuojami algoritmai realizuoja tik pirmąją - identifikavimo dedamąją, todėl neutralizavimo dedamoji darbe turės būti atskirai suprojektuota.

2.3.1. Sekos numerio skirtumo algoritmas (GAP)

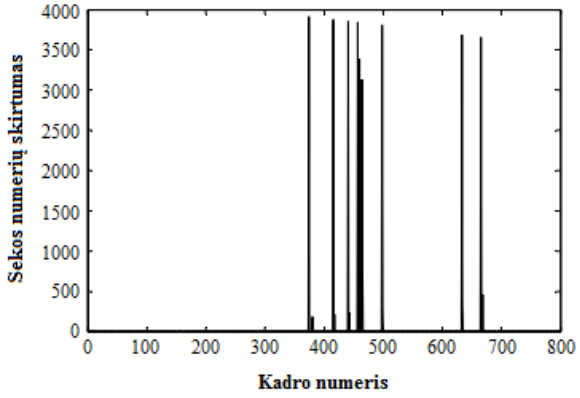
Šis identifikacijos metodas pagrįstas monotoniškų ir aiškiu sekos numerio kitimu ir dėdėjimu vienetu. Nukrypimas nuo tipinio sekos numerio kitimo dėsningumo traktuojamas kaip anomali veikla ir identifikuojamas kaip galima ataka. Metodo esmę sudaro slenksčio apskaičiavimas ir įvertinimas [5].

Slenkstis – tai skirtumas (angl. *GAP*) tarp dviejų laiko atžvilgiu gretimų kadrų sekos numerių. Idealiu atveju jis visada lygus 1. Neigiamas skirtumas gaunamas esant iškraipytai kadrų sekai, o didesnis nei 1– prie atakos arba netinkamo tinklo veikimo. Algoritmas pseudokodo pavidalu pateiktas lentelėje Nr.4.

Lentelė Nr. 4 *GAP apsaugos algoritmo pseudokodas [5]*

```
gauti S(i);
G =  (S(i) - S(i-1)) /(mod(G, 4096);
jeigu (G >= P)
{
    nustatyta ataka;
}
kitu atveju
{
    testi;
}
```

Šį metodą pasiūlę autoriai atliko eksperimentus ir 39 pav. pavaizduota šio metodo veikimo rezultatai. Autorių tyrimo metu, algoritmas nesugebėjo identifikuoti 41,25 procentų atakų (angl. *False Negative*), ir identifikavo 33,19 procentų atakų, kuomet jų nebuvo (angl. *False Positive*). Teisingai atpažinta buvo 39,12 procentų atakų.



39 pav. *GAP algoritmo identifikacijos rezultatai [5]*

Matomi aiškūs sekos numerio neatitikimai byloja anomalią veiklą tinkle. Nustatant kokio dydžio neatitikimą traktuoti kaip ataką, galima reguliuoti teisingų ir klaidingų nustatymo kiekius.

2.3.2. Sekos numerio kitimo spartos algoritmas (SNRA)

Sekos numerio dažnumo analizė SNRA (angl. *Sequence Number Rate Analysis*) pagrįsta belaidės stotelės transliuojamų kadrų perdavimo dažnumu. Perdavimo dažnumas apskaičiuojamas dviejų gretimų kadrų sekos numerių skirtumą dalinant iš tų kadrų atvykimo laiko skirtumo [5].

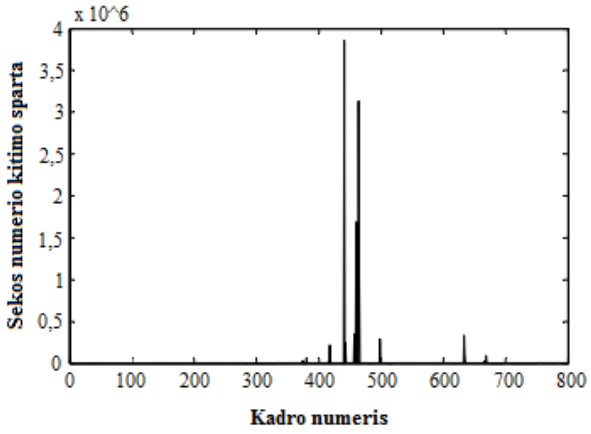
Apibendrintas algoritmas pseudokodu pateiktas lentelėje Nr. 5.

Lentelė Nr. 5 *SNRA apsaugos algoritmo pseudokodas [5]*

```
gauti S(i), T(i);
G =  (mod(S(i) – S(i-1), 4096))/(T(i)-T(i-1));
jeigu (G >= P)
{
    nustatyta ataka;
}
kitu atveju
{
    testi;
}
```

Konkretus 802.11 standartas turi teorinę siunčiamų paketų kiekio per sekundę ribą. Jeigu apskaičiuotas sekos numerio dažnumas yra didesnis už šią ribą, laikoma, kad identifikuota ataka.

Šį metodą pasiūlę autoriai atliko eksperimentus ir 40 pav. pavaizduota šio metodo veikimo rezultatai. Autorių tyrimo metu, algoritmas nesugebėjo identifikuoti 44,18 procentų atakų, ir identifikavo 35,32 procentų atakų, kuomet jų nebuvo. Teisingai atpažinta buvo 35,02 procentų atakų.



40 pav. *SNRA algoritmo identifikacijos rezultatai [5]*

Esminis šio metodo privalumas prieš slenksčiu paremtą metodą tas, kad išvengiama neteisingų identifikavimų dėl natūralaus paketų praradimo. Teorinė maksimali riba apskaičiuojama imant maksimalią įmanoma standarto pralaidą (802.11b atveju 11Mb/s) ir mažiausią kadro dydį (ACK ir CTS atveju 14B). Tokiu atveju riba yra 98,214 kadrų/s, vadinasi, numerio kitimo sparta neturėtų viršyti 98,3.

2.3.3. Sekos numerio intervalo analizės algoritmas (FRR)

Prieš tai aptartieji metodai naudoja individualius paketus arba gretimus paketus. FRR (angl. *Forge Resilient Relationship*) algoritmo esmė yra operavimas tam tikru nuosekliu paketų kiekiu, vadinamu „langu“ su *N* paketų [5].

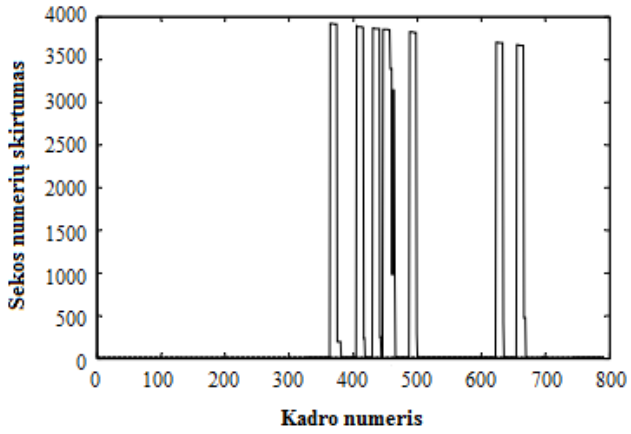
Algoritmas apskaičiuoja *N-1* sekos skaičių skirtumus *(d1, d2, d3....dn-1)* tarp kiekvienų gretimų kadrų. Šiuo atveju, identifikavimo sprendimus galima grįsti eilia faktorių, gautų iš skirtumų sekos. Bazinis sprendimas priimamas pagal maksimalų skirtumą [13,26].

Apibendrintas algoritmas pseudokodu pateiktas lentelėje Nr. 6.

Lentelė Nr. 6 *FRR apsaugos algoritmo pseudokodas [5]*

```
gauti w(k) = (S(k), S(k-1),...,S(k-N-1));
d = (d1,d2,d3,…DN-1);
dn = (S(k-N+1)-S(k-N)(mod 4096));
jeigu (MAX(di)>= T)
{
    nustatyta ataka;
}
kitu atveju
{
    testi;
}
```

Šį metodą pasiūlę autoriai atliko eksperimentus ir 41 pav. pavaizduota šio metodo veikimo rezultatai. Autorių tyrimo metu, algoritmas nesugebėjo identifikuoti 1,72 procentų atakų, ir identifikavo 81,91 procentų atakų, kuomet jų nebuvo. Teisingai atpažinta buvo 94,48 procentų atakų.



41 pav. *FRR algoritmo identifikacijos rezultatai [5]*

2.3.4. Pseudoatsitiktinio sekos numerio algoritmas

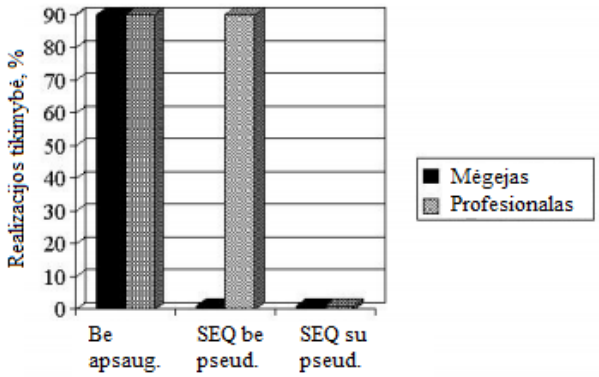
Šio metodo pagrindą sudaro ne pagal standartą numatyto nuoseklaus, o pseudoatsitiktinio sekos numerio generavimas. Tikrinančioji pusė galėtų autentifikuoti kadrus su pseudoatsitiktiniais sekos numeriais tikrindama, ar sekos numeris yra tam tikroje priimtinoje riboje. Jeigu sekos numeris nepatenkina tikrinimo sąlygos, kadras yra naikinamas, jei patenkina – apdorojamas [4].

Algoritmas pseudokodu pateiktas lentelėje Nr. 7.

Lentelė Nr. 7 *Pseudoatsitiktinio sekos numerio algoritmas [4]*

```
PSN = (PRF (PSK, generavimas,
MIN(AA,SPA)||MAX(AA,SPA)) XOR SNp);
jeigu (PSN = [A,B])
{
    trinti kadra;
}
kitu atveju
{
    testi;
}
PSN – Pseudo Random Sequence Number
PRF – Pseudo-Random Function
SNp – Previuosly used Sequence Number
AA – Aps MAC address
SPA – Client's MAC address
```

Šį metodą pasiūlę autoriai atliko eksperimentus ir 42 pav. pavaizduota šio metodo veikimo rezultatai, kurie vaizduoja tikimybę, jog ataka bus sėkminga, priklausomai nuo algoritmo tipo ir atakuotojo patirties.



42 pav. *Pseudoatsitiktinio sekos numerio algoritmo rezultatai [4]*

2.3.5. Sekos numerio analizės algoritmų apibendrinimas

Sekos numerio analize paremtų belaidžio tinklo atsisakymo aptarnauti atakos identifikavimo algoritmų apibendrinimas pateiktas lentelėje Nr. 8. Remiantis lentele, FRR algoritmas generuoja potencialiai mažiau klaidingų identifikacijų, tačiau dėl grupinio kadrų analizavimo, taip pat identifikuoja potencialiai mažiau anomalijų, todėl jo architektūra yra tobulintina. Pseudoatsitiktinio generavimo algoritmas reikalauja standarto pakeitimo, todėl nepaisant savo privalumų, toliau darbe analizuojamas nebus.

Lentelė Nr. 8 *Kadro sekos numerio analizės apsaugos algoritmų apibendrinimas*

Algoritmas	**Sekos numerio tikrinimo kriterijus**	**Atsparumas prarastiems kadrams**	**Dubliuoto MAC adreso identifikavimas**	**Vienu metu analizuojamų kadrų kiekis**	**Reikalingas standarto keitimas**
GAP	Skirtumas tarp dviejų gretimų kadrų sekos numerių	Ne	Taip	2	Ne
SNRA	Sekos numerio kitimo sparta lyginant dviejų gretimų kadrų numerius ir laiko žymas	Taip	Taip	2	Ne
FRR	Didžiausas skirtumas tarp n kadrų tarpusavyje gretimų kadrų skirtumų	Taip	Ne	n	Ne
PSEUDO	Tuo pačiu skaičiu inicijuotų šaltinio ir gavėjo generatorių išduotų sekos numerių sulyginimas	Taip	Taip	1	Taip

2.4. Analizės išvados ir darbo tikslas

IEEE 802.11 standartų šeimos belaidžiai tinklai visuose lygiuose pagal OSI modelį turi pažeidžiamumų, kuriais potencialiai gali būti realizuojami ir realiai yra realizuojami atsisakymo aptarnauti atakų algoritmai, tačiau aktualiausia yra fizinio ir kanalinio lygmenų apsauga, nes juose yra realizuotas standartas. Šių lygmenų apsaugos nuo atsisakymo aptarnauti algoritmus galima realizuoti keičiant tinklo standartą arba apsaugai panaudojant aukštesnius lygmenis.

Fizinio lygmens atsisakymo aptarnauti atakos yra sudėtingiausiai išvengiamos ir sukuria didžiausius nuostolius, nes manipuliuodamos belaidžio tinklo signalų charakteristikomis, sukuria fizines sąlygas: didelį triukšmų lygį, iškraipytą sinchronizaciją ir kita. Esant šioms sąlygoms, tinklas apskritai negali veikti. Efektyviausias šių atakų malšinimo metodas yra atakuojančio įrenginio fizinė paieška ir pašalinimas, tačiau taip pat taikomas dažninis ir erdvinis atakuojančio įrenginio išvengimas.

Kanalinio lygmens atsisakymo aptarnauti atakos veikia pagal legalius šio lygio mechanizmus: autentifikacijos, bendros terpės valdymo ir kitus. Dėl standartinių mechanizmų naudojimo, kanalinio lygio atakos formalių standarto taisyklių nepažeidžia, tačiau generuoja suklastotus tinklo kadrus, kurie klaidingai inicijuoja tam tikrus įvykius tinkle: deautentifikaciją, diasociaciją, sukauptų buferių atlaisvinimą ir kita. Šie klaidingai inicijuoti veiksmai suformuoja atsisakymo aptarnauti atakos efektą. Kanalinio lygmens atakas galima efektyviai malšinti stipria kadrų autentifikacija ir tikrinimu.

Sekos numerio analize paremti algoritmai naudojami atakų identifikacijai ir veikia su tam tikra sėkmingos identifikacijos tikimybe, tačiau šiuos algoritmus galima panaudoti apsaugai prieš DoS atakas, juose realizuojant reaktyvią logiką ir naikinant galimai nelegalius tinklo kadrus.

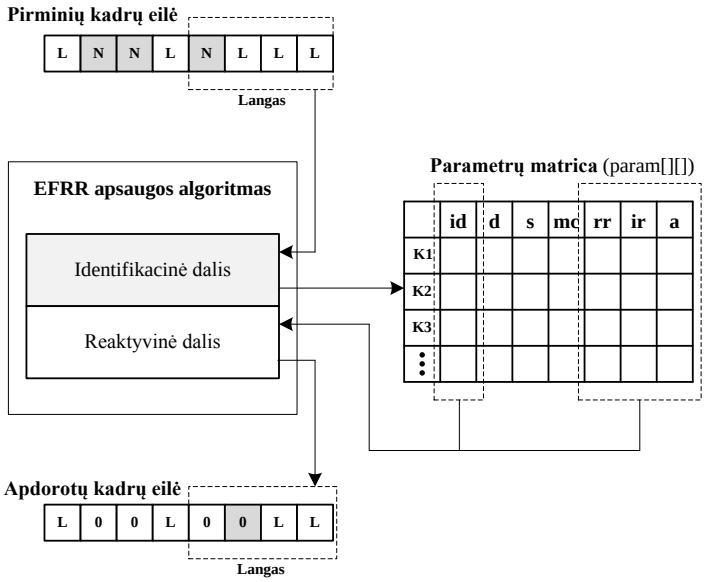
Šio magistrinio darbo tikslas yra sudaryti kadro sekos numerio analize paremtą apsaugos nuo DoS atakų algoritmą ir jį ištirti. Siekiant aprašytų tikslų ir rezultatų, būtina įvykdyti šiuos uždavinius:

1. Išanalizuoti esamus literatūroje publikuojamus apsaugos nuo atsisakymo aptarnauti atakų belaidžiame tinkle algoritmus;
2. Pasiūlyti savo apsaugos algoritmą, paremtą kadrų sekos numerio analize;
3. Sudaryti kompiuterinį tyrimo modelį, su galimybe simuliuoti tipines belaidžio tinklo atakas ir tiriamuosius apsaugos algoritmus;
4. Sudaryti algoritmų tyrimo metodiką ir ją taikant, atlikti atakų ir apsaugos algoritmų modeliavimus;
5. Remiantis tyrimų rezultatais, įvertinti sudarytąjį algoritmą ir suformuoti galutines darbo išvadas.

3. SEKOS ANALIZE PAREMTO APSAUGOS ALGORITMO IR PROGRAMINIO MODELIO PROJEKTAVIMAS

3.1. Išplėstasis FRR algoritmas (EFRR)

Išplėstasis FRR algoritmas (angl. *Extended Forge Resilient Relationship*) – tai šiame darbe sudarytas apsaugos nuo atsisakymo aptarnauti atakų algoritmas, paremtas kadrų sekos numerio analize ir FRR algoritmo savybėmis. Apibendrinta algoritmo schema pateikta 43 pav.



43 pav. *EFRR algoritmo konceptinė schema*

EFRR algoritmą struktūriškai sudaro dvi pagrindinės dedamosios: identifikacinė dalis ir reaktyvinė dalis. Identifikacinė dalis atsakinga už priimtų kadrų analizę ir galimai nelegalių kadrų identifikavimą. Ši dalis gali būti naudojama atskirai, kaip įsibrovimo aptikimo algoritmas ir panaudojama WIDS/WIPS sistemose. Reaktyvioji dalis, panaudodama identifikacinės dalies rezultatus, aktyviai blokuoja kadrus, kurie įtariami kaip nelegalūs. Savaime suprantama, identifikacinė dalis yra svarbiausia, nes būtent nuo jos priklauso algoritmo efektyvumas atakos malšinimo atžvilgiu.

Algoritmo identifikacinė dalis kadrus analizuoja grupėmis, kurios vadinamos langais. Sisteminiu požiūriu, langas yra tam tikro ilgio C kalbos struktūrų, realizuojančių virtualius kadrus, masyvas, kuris schemoje atitinka pirminių kadrų eilės dalį. Gavęs tam tikro lango ilgio kadrus, algoritmas atlieka kiekvieno lange esančio kadro analizę ir užpildo parametrų matricą, kurios santrumpų reikšmės pateiktos lentelėje Nr. 9. Užpildžius parametrų matricą, ji perduodama algoritmo reaktyviajai daliai, kuri analizuoja

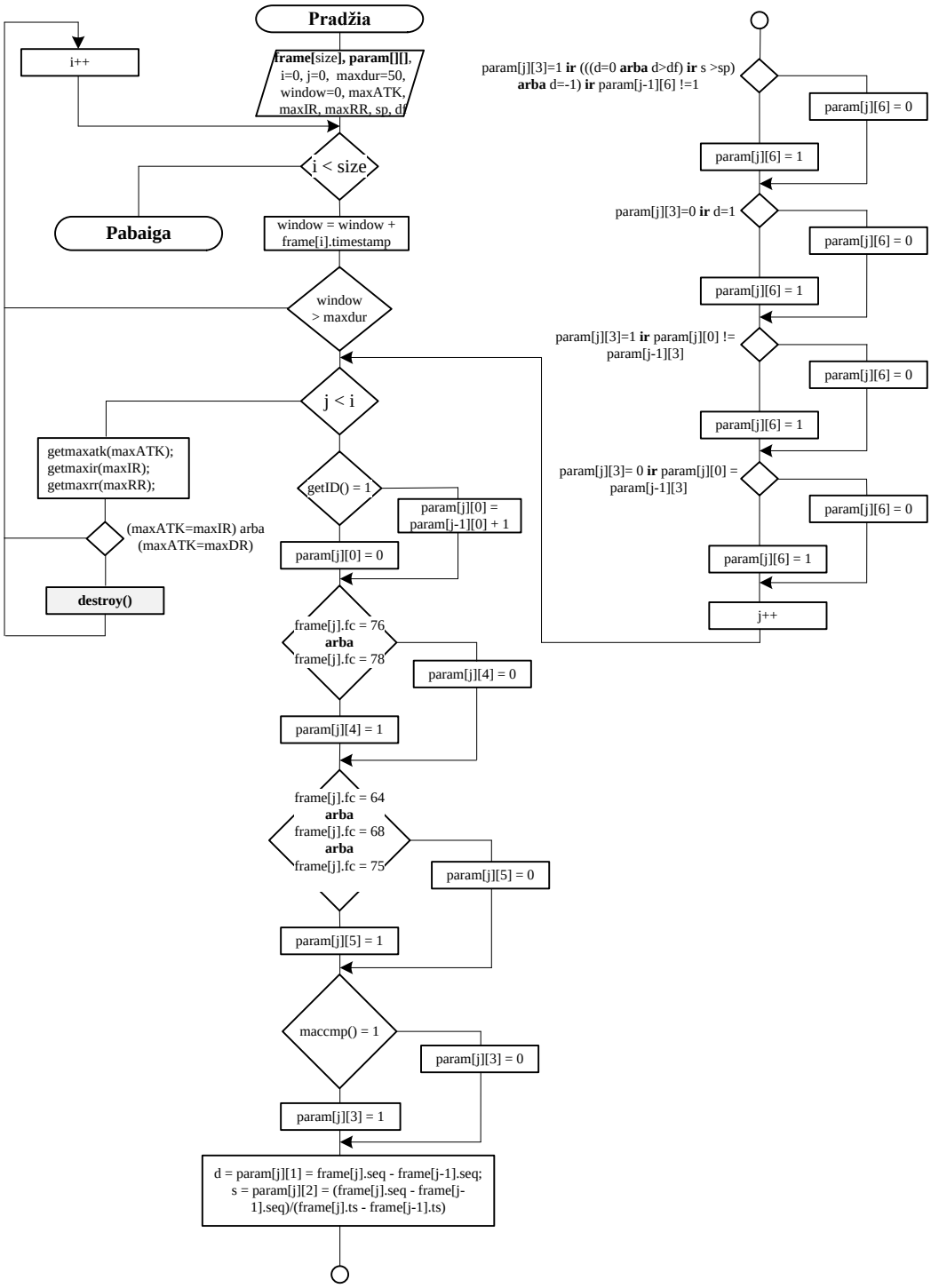
matricoje pateiktus parametrus ir nusprendžia, kuris iš `id` lauke įvardintų adapterių yra galimai[2] nelegalus, bei naikina visus to adapterio kadrus. Po to, priimamas naujas kadrų langas ir visa procedūra kartojama iš naujo.

Lentelė Nr. 9 *Identifikacinių parametrų aprašymas*

id	Tinklo plokštės reliatyvus identifikatorius. Kinta nuo 0 iki *n*, kur *n* – nustatytų skirtingų tinklo plokščių kiekis. Skirtingos tinklo plokštės nustatomos lango ribose analizuojant kadrų sekos numerio kitimo dėsningumą, ir vadovaujantis standarto sąlyga, kad tos pačios tinklo plokštės kadrų numeriai skiriasi vienetu.
d	Sekos numerių skirtumas tarp esamo ir prieš tai buvusio kadro
s	Sekos numerių kitimo sparta lyginant esamo kadro numerį ir laiko žymą su prieš tai gauto kadro numeriu ir laiko žyma.
mc	MAC adresų įvertinimo parametras: 1, jeigu dabartinio ir prieš tai buvusio kadro MAC adresai sutampa; 0 – jeigu ne.
rr	Valdymo kadro identifikavimo parametras: 1, jeigu kadras inicijavo autentifikaciją, asociaciją ar zondavimą; 0 – jeigu ne.
ir	Valdymo kadro identifikavimo parametras: 1, jeigu kadras reikalavo deautentifikacijos, diasociacijos arba reasociacijos; 0 – jeigu ne.
a	Atakos nustatymo parametras: 1 – jeigu nustatoma, jog kadras galimai atakos; 0- jeigu ne.

Detalesnį algoritmo veikimą vaizduoja 44 pav. paveikslėlyje pateikta algoritmo blokinė schema. Algoritmo vykdymo pradžioje yra apibrėžiami pradiniai parametrai, reikalingi algoritmui: tuščia parametrų matrica `param[n][7]` (čia `n` – lango ilgis; `1-7` nurodo 48 pav. schemoje ivardintus parametrus); iteracijų kintamieji `i`, `j`; sekos numerių skirtumo slenkstis `df`, sekos numerių kitimo spartos slenkstis `sp` ir kt. Po to, tikrinama, ar kadrų laiko žymių suma neviršija tam tikros ribos. Šis mechanizmas naudojamas tam, kad kadrų lango ilgis būtų dinamiškai parenkamas atsižvelgiant į kadrų gavimo spartą. Jeigu kadrų lango ilgis visada būtų statinis, tai esant mažai kadrų siuntimo spartai algoritmai ilgai lauktų, kol visa eilė būtų užpildyta ir tai sukeltų didelį kadrų vėlinimą. Visas sugaištamas laikas apibrėiamas tik kaip priimamų kadrų laiko žymų skirtumo suma. Algoritmas priima tiek kadrų, kad jų užvėlinimas neviršytų `maxdur` parametro reikšmės, kitaip sakant, šiuo atveju algoritmas paketų „laukia“ ne ilgiau nei 50ms. Priėmus per 50ms „sulauktus“ kadrus, algoritmas pradeda vykdyti identifikacinę savo dalį ir pildo minėtąją parametrų matricą `param[n][7]`. Parametrų matrica pildoma analizuojant priimo kadro antraštę (kontrolės lauką `fc`, sekos lauką `seq`, adresų laukus `addr` ir pan.) ir lyginant su prieš tai gauto kadro antrašte. Po to, matrica perduodama reaktyviajai algoritmo daliai, kuri pagal 44 pav. pateiktoje schemoje pavaizduotas logines sąlygas tikrina parametrų matricos elementų reikšmes ir priima sprendimą dėl galimai nelegalių kadrų pašalinimo, kuris sistemiškai atliekamas ištrinant C struktūrą.

[2] Šio darbo kontekste dažnai pasitaikantis įvardijimas „galimai“ nurodo, kad identifikacija įvyksta tik su tam tikra teisingos identifikacijos tikimybe, todėl bendru atveju kadrai nustatomi kaip galimai nelegalūs.



44 pav. *EFRR algoritmo blokinė schema*

Atvaizduoti visus algoritmo veiksmus detalizuojant iki visų kviečiamų funkcijų algoritmų būtų pernelyg sudėtinga, todėl dalis algoritmo blokinėje schemoje pavaizduotų veiksmų nedetalizuoti, o

pateikti apibendrintomis[3] C kalbos funkcijomis, pvz. `destroy()`, kurios aprašytos lentelėje Nr. 10. Detalų algoritmą C išeities kodo pavidalu galima rasti šio darbo prieduose.

Lentelė Nr. 10 *Apibendrinančių funkcijų aprašymas*

Apibendrinta funkcija	Aprašymas
`getID()`	Nustato kadrą sugeneravusios tinklo plokštės reliatyvų identifikatorių (nustato matricos `id` parametrą)
`getmaxatk(maxATK)`	Nustato, kurio parametrų matricoje identifikuoto adapterio kadrų yra daugiausiai nustatytų kaip nelegalių, ir to adapterio ID gražina per `maxATK` kintamąjį.
`getmaxir(maxIR)`	Nustato, kurio parametrų matricoje identifikuoto adapterio kadrų yra daugiausiai nustatytų su autentifikacijos, asociacijos arba zondavimo užklausomis, ir to adapterio ID gražina per `maxATK` kintamąjį.
`getmaxrr(maxRR)`	Nustato, kurio parametrų matricoje identifikuoto adapterio kadrų yra daugiausiai nustatytų su deautentifikacijos, diasociacijos ir rasociacijos užklausomis, ir to adapterio ID gražina per `maxATK` kintamąjį.
`destroy()`	Naikina kadrą – ištrina iš eilės (nunilina kadrą modeliuojančia struktūrą).
`maccmp()`	Palygina dviejų kadrų MAC adresus ir grąžina 1 jei jie sutampa, arba 0, jeigu ne.

3.2. Priemonės

Pagrindinis šo darbo tyrimų akcentas yra specializuotas modelis, kuriame programiškai realizuoti tiriamieji algoritmai. Modelio realizacijai panaudotos priemonės pateiktos lentelėje Nr. 11. C programavimo kalba buvo pasirinkta todėl, kad leidžia tiesiogiai valdyti atminties išskyrimus, operacijas su simbolių eilutėmis ir kitus žemesnio lygio aspektus - tai yra paranku šio darbo kontekste. C kalba taip pat yra viena populiariausių sistemų prorgamavimo kalbų, kuriam priskirtinos analizuojamų algoritmų realizacijos.

Lentelė Nr. 11 *Modelio realizacijai naudotos priemonės*

Integruota plėtojimo aplinka (IDE)	CodeBlocks 8.04
Programavimo kalba	C (Ansi C)
Naudotos bibliotekos	Standartinė C biblioteka 2.13
Kompiliatorius	GCC (GNU C Compiler) 4.4.2
Operacinė sistema	Ubuntu Linux 10.04 LTS
Aparatinė dalis	x86 architektūros PC

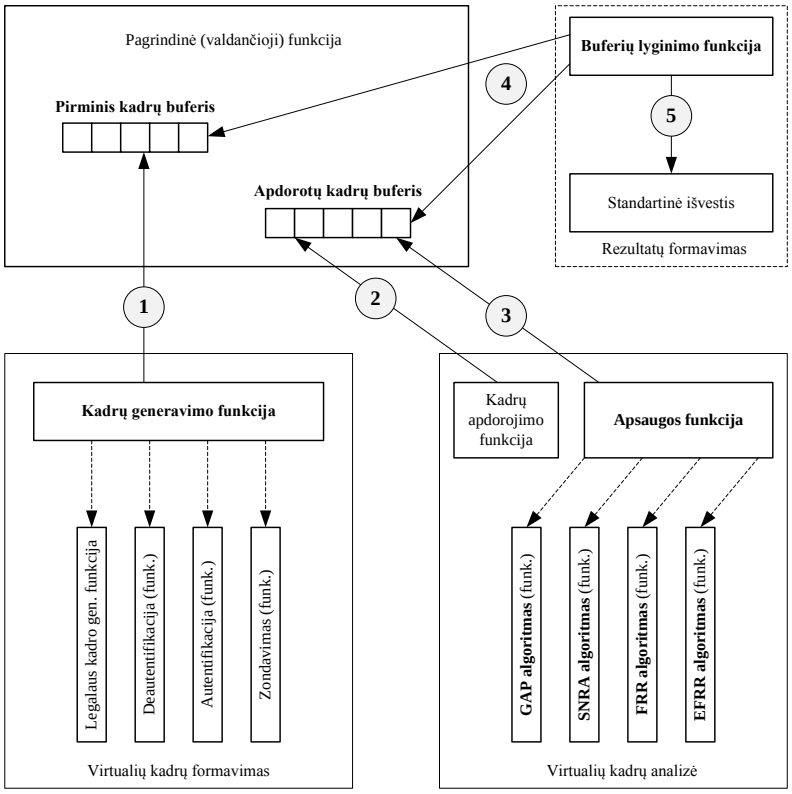
“Linux” operacinė sistema naudojama dėl patogaus procesų įvesties ir išvesties organizavimo - tai sumažina modelio apimtį, pavyzdžiui, nereikia programuoti rašymo į failus, nes tai atlieka pati operacinė sistema.

[3] Apibendrinta funkcija – tai funkcijos užrašas, atitinkantis arba tikrą C funkcija algoritmo C išeities kode, arba logiškai apibendrinantis tam tikrą veiksmų seką, kuri sugeneruoja funkcijos apraše pateiktą rezultatą ir kuri nėra aktuali pačio algoritmo analizės kontekste, todėl nėra nedetalizuojama.

3.3. Modelis

Panaudojant aprašytas priemones, buvo realizuotas specializuotas tyrimo modelis. Sąvoka „specializuotas“ šio darbo konkteskte reiškia, jog modelis buvo sukurtas tik šio darbo tyrimams ir nebuvo naudota jokia egzistuojanti tinklų modeliavimo platforma. Toks variantas buvo pasirinktas todėl, kad esamos tinklų modeliavimo platformos yra diskretinio laiko ir įvykių, o tai nėra tinkama modelio architektūra siekant tirti analizuojamus algoritmus jų apdorojamų tinklo kadrų atžvilgiu. Modelio struktūrinė schema pateikta 45 pav. Schemoje brūkšnine linija pažymėtos esminės modelio dalys:

- *Valdančioji funkcija* – funkcija, kuri nustato modelio parametrus ir kviečia kitas funkcijas.
- *Virtualių kadrų formavimas* – funkcijų rinkinys, kuris C kalbos duomenų struktūrų pavidalu formuoja virtualius IEEE 802.11 standarto tinklo kadrus ir perduoda kitoms funkcijoms. Šioje dalyje formuojami tiek legalūs, tiek ir nelegalūs (tam tikros atakos) kadrai, kurie generuojami pagal nustatytus parametrus.
- *Virtualių kadrų analizė* – funkcijų rinkinys, priimantis kadrus ir atliekantis veiksmus su jais: identifikuojantis atakas, trinantis galimai nelagalius kadrus ir kt.
- *Rezultatų formavimas* – funckijos, skirtos išvesti tirtųjų statistikų reikšmes po simuliacijos.



45 pav. *Tyrimo modelio struktūrinė schema*

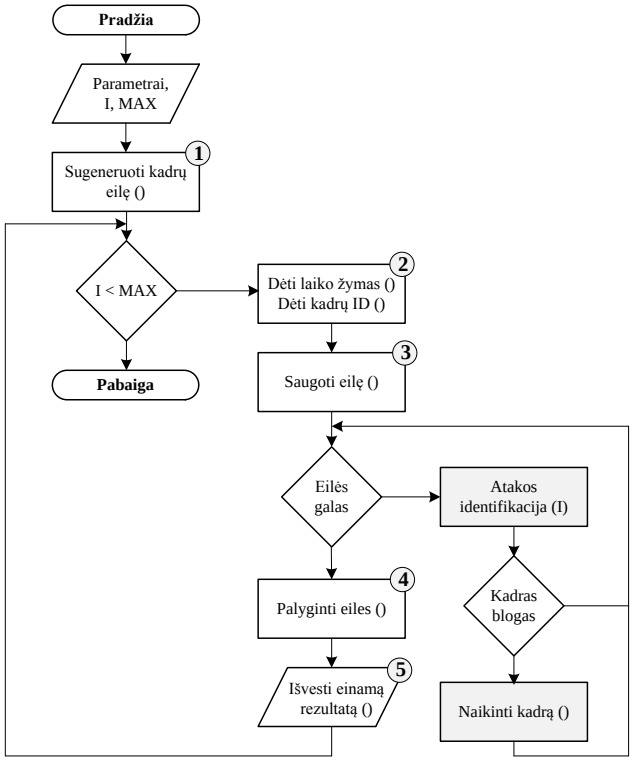
Schemoje rodyklėmis pažymėti ryšiai tarp esminių modelio funkcijų, o numeriai apskritimuose nurodo vieno modeliavimo ciklo etapus:

1. Pagal tyrėjo nurodytus parametrus (kadrų kiekį, nelegalių kadrų santykį, atakos tipą ir pan.) generuojama kadrų eilė. Eilė sudaryta iš tam tikro kiekio legalių kadrų bei atsitiktine tvarka tarp jų pasiskirsčiusių nelegalių tam tikros atakos kadrų. Sugeneruota kadrų eilė sisteminiu požiūriu saugoma kaip C struktūrų masyvas pirminiame 2MB buferyje. Šis buferis modeliuoja kadrus tokius, kokie jie perduodami belaidžio tinklo eteriu (realizuotas loginis IEEE 802.11 kanalinio lygmens kadro formatas);
2. Kadrų apdorojimo funkcija priima pirminį buferį su kadrais ir suformuoja apdorotų kadrų buferį. Šiame buferyje saugomi kadrai išlaiko savo pirminę struktūrą, tačiau jiems yra priskiriamos naujos savybės, pavyzdžiui, laiko žyma. Šis buferis modeliuoja paskirties mazgo operacinės sistemos priimtus kadrus;
3. Apsaugos funkcija priima apdorotų kadrų buferyje saugomus virtualius kadrus ir jiems taiko apsaugos algoritmus, kurie analizuoja kadrus ir ištrina iš eilės tuos kadrus, kurie yra galimai nelegalūs;
4. Buferių lyginimo funkcija sulygina pirminį kadrų buferį su apdorotų kadrų buferiu, kuriame saugojami apsaugos algoritmų apdoroti kadrai. Kadangi iš anksto žinoma kokių ir kiek kadrų buvo sugeneruota, ši funkcija nustato, kiek buvo ištrinta legalių ir nelegalių kadrų, kitaip tariant – ši funkcija formuoja algoritmų efektyvumo statistikas;
5. Lyginimo funkcijos rezultatai per OS standartinę išvestį pateikiami tyrėjui ekrane (tyrėjas juos gali nukreipti į bylą);

Šiais skaičiais pažymėti punktai logiškai atitinka sekančiuose modelio aprašo skyreliuose skaičiais pažymėtus punktus, todėl pakartotinai jie nebus aprašomi.

3.3.1. Modelio veikimo algoritmas

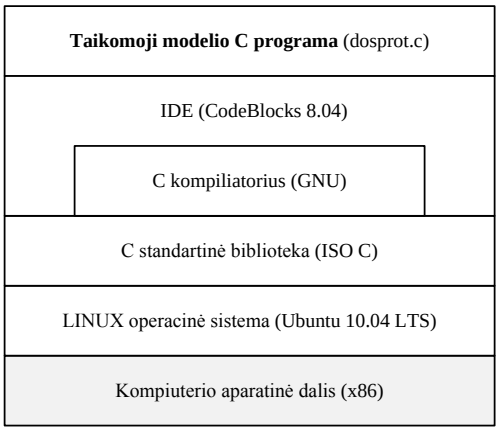
Detalesnį tyrimo modelio veikimą vaizduoja 46 pav. pateiktas apibendrintas modelio veikimo algoritmas. Simuliacijos pradžioje nustatomi kadrų pasiskirstymo, algoritmų veikimo režimo, sisteminių savybių ir kt. parametrai, kurių atžvilgiu tiriami algoritmai. Pagal šiuos parametrus generuojama virtualių kadrų eilė, kuri po to apdorojama formuojant kadrų priėmimo laiko žymas. Suformavus laiko žymas, kadrai perduodami apsaugos algoritmui, kuris naudodamas atakos identifikaciją nustato galimai blogus kadrus ir juos ištrina. Ciklo gale lyginimo funkcija išveda simuliacijos rezultatus. Skaičiais pažymėti etapai atitinka 3.3 skyrelyje aprašytus modelio simuliacijos ciklo žingsnius.



46 pav. *Tyrimo modelio veikimo algoritmas*

3.3.2. Modelio realizacija

Modelis realizuotas naudojant 3.2 skyrelyje aprašytas priemones. Sluoksninė realizuoto modelio schema pavaizduota 47 pav. Programinė dedamoji veikia x86 architektūros kompiuterio aparatinėje dalyje, o pačio modelio struktūrinės dalys yra realizuotos C programavimo kalbos funkcijomis.



47 pav. *Modelio realizacijos sluoksninė schema*

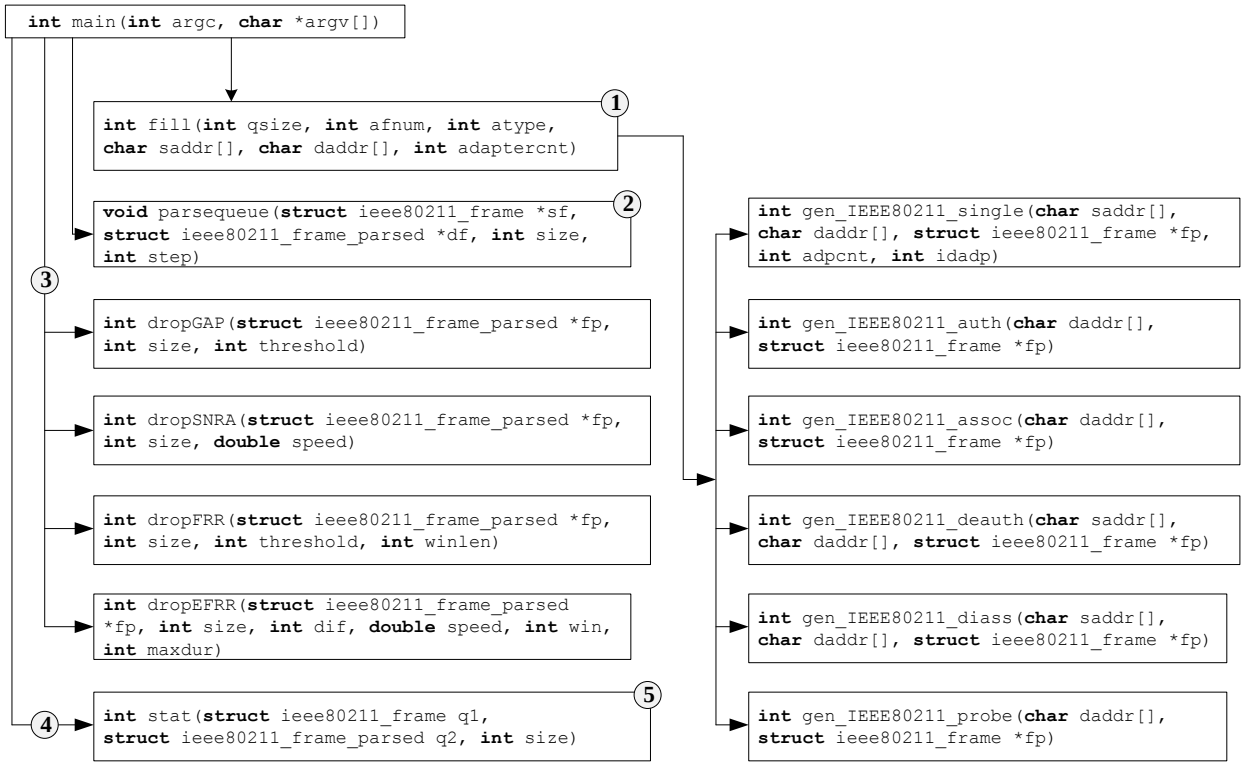
Virtualus kadras realizuotas kaip C kalbos struktūra, kuri pateikta lentelėje Nr. 12. Šią struktūrą per argumentus naudoja kadrų generavimo ir apdorojimo funkcijos. Kiekvienas struktūros elementas realizuoja IEEE 802.11 standarto kanalinio lygmens kadro lauko reikšmę, kurios simuliacijos metu yra

keičiamos pagal tai, kaip tai apibrėžia standartas. Modelis nerealizuoja viso standarto, tačiau pateikia supaprastintą kadrų generavimo ir apdorojimo mechanizmą, modeliuojantį tikro šio standarto tinklo kadrų tėkmę.

Lentelė Nr. 12 *Kadrą modeliuojanti C duomenų struktūra*

```
struct ieee80211_frame {
        u_int8_t        fc[2];
        u_int8_t        dur[2];
        u_int8_t        addr1[IEEE80211_ADDR_LEN];
        u_int8_t        addr2[IEEE80211_ADDR_LEN];
        u_int8_t        addr3[IEEE80211_ADDR_LEN];
        u_int8_t        seq[2];
        u_int8_t        addr4[IEEE80211_ADDR_LEN];
        u_int8_t        qos[2];
        u_int8_t        pld[12];
        u_int8_t        fcs[4];
    };
```

48 pav. pavaizduoti esminiai modelio C funkcijų prototipai ir ryšiai, kurie nurodo funkcijų tarpusavio kvietimus. Numeriais pažymėtos tos funkcijos, kurios realizuoja 3.3 skyrelyje tokiais pačiais numeriais pažymėtus modelio simuliacijos žingsnius.



48 pav. *Modelio realizacijos C funkcijų ryšiai*

Funkcijų prototipai detaliau aprašyti lentelėje Nr. 13, kurioje pateikti tik esminių [4] modelį realizuojančių funkcijų apibūdinimai. Funkcijos komunikuoja per buferį, kuriame jau minėtos

[4] Visą modelio C išeities tekstą galima rasti laikmenoje, pridėtoje prie šio magistrinio darbo.

ieee80211_frame struktūrų masyvo pavidalu saugomi virtualūs kadrai. Buferis yra 2MB dydžio ir simuliacijos metu saugo pastovaus 50000 kadrų ilgio eilę. Visi parametrai modeliui yra perduodami per pagrindinės funkcijos main() argumentus, kurie yra nurodomi per operacinės sistemos komandinę eilutę. Konkrečios parametrų reikšmės parenkamos pagal tyrimo metodiką, tačiau visais atvejais laikoma, jog kadrai virtualiai keliauja pastoviu greičiu su pastoviu vėlinimu ir pastoviu pasiskirstymu. Pastarieji požymiai realizuojami kadrams priskiriant pastovaus skirtumo laiko žymes, taigi modelyje kadrų siuntimas modeliuojamas kaip struktūrų masyvo rašymas ir skaitymas.

Lentelė Nr. 13 *Modelį realizuojančių C funkcijų prototipų aprašymas*

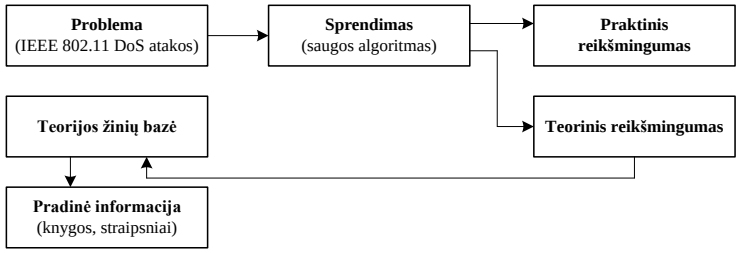
Funkcijos prototipas	Funkcijos aprašymas
`int main(int argc, char *argv[])`	Pagrindinė funkcija, priimanti `argc` kiekį argumentų iš komandinės eilutės sąsajos ir suformuojanti `argv` adresų masyvą į gautus argumentus. (Argumentais nustatomi modelio parametrai).
`int fill(int qsize, int afnum, int atype, char saddr[], char daddr[], int adaptercnt)`	Užpildo `qsize` ilgio `queue` eilę `ieee80211_frame` struktūromis, iš kurių kiekviena modeliuoja virtualų kadrą. Argumentuose nurodoma: blogų kadrų kiekis `afnum`, atakos tipas `atype`, šaltinio adresas `saddr`, paskirties adresas `daddr` ir legalių tinklo adapterių kiekis `adaptercnt`.
`void parsequeue(struct ieee80211_frame *sf, struct ieee80211_frame_parsed *df, int size, int step)`	Priima `size` ilgio `sf` eilėje saugomus kadrus ir kiekvienam kadrui uždeda `step` dydžio laiko žymą bei sisteminį identifikatorių. Apdoroti kadrai kaip struktūrų masyvas įrašomi į `df` eilę.
`int dropGAP(struct ieee80211_frame_parsed *fp, int size, int threshold)`	Tikrina `size` ilgio `fp` eilėje saugomus kadrus ir ištrina tuos, kurie GAP algoritmu identifikuojami kaip galimai blogi. Argumentuose nurodoma: algoritme naudojamas sekos numerių skirtumas `threshold`.
`int dropSNRA(struct ieee80211_frame_parsed *fp, int size, double speed)`	Tikrina `size` ilgio `fp` eilėje saugomus kadrus ir ištrina tuos, kurie SNRA algoritmu identifikuojami kaip galimai blogi. Argumentuose nurodoma: algoritme naudojamas sekos numerių kitimo greitis `speed`.
`int dropFRR(struct ieee80211_frame_parsed *fp, int size, int threshold, int winlen)`	Tikrina `size` ilgio `fp` eilėje saugomus kadrus ir ištrina tuos, kurie FRR algoritmu identifikuojami kaip galimai blogi. Argumentuose nurodoma: algoritme naudojamas lango ilgis `winlen` ir sekos numerių skirtumas `threshold`.
`int dropEFRR(struct ieee80211_frame_parsed *fp, int size, int dif, double speed, int win, int maxdur)`	Tikrina `size` ilgio `fp` eilėje saugomus kadrus ir ištrina tuos, kurie EFRR algoritmu identifikuojami kaip galimai blogi. Argumentuose nurodoma: algoritme naudojamas lango ilgis `win`, sekos numerių skirtumas `dif`, sekos numerių kitimo greitis `speed` ir maksimali laukimo trukmė `maxdur`.
`int stat(struct ieee80211_frame q1, struct ieee80211_frame_parsed q2, int size)`	Sulygina `size` ilgio `q1` ir `q2` eilėse saugomus kadrus ir išspausdina blokuotų kadrų statistikas.
`int gen_IEEE80211_single(char saddr[], char daddr[], struct ieee80211_frame *fp, int adpcnt, int idadp)`	Sugeneruoja legalų virtualų kadrą kaip `ieee80211_frame` struktūrą bei patalpina į `fp` rodykle nurodyta vietą. Argumentuose nurodoma: šaltinio adresas `saddr`, paskirties adresas `daddr`, adapterių kiekis `adpcnt` ir adapterio ID `idadp`.
`int gen_IEEE80211_auth(char daddr[], struct ieee80211_frame *fp)`	Sugeneruoja autentifikacijos tvindymo atakos kadrą kaip `ieee80211_frame` struktūrą bei patalpina į `fp` rodykle nurodyta vietą. Argumentuose nurodoma: paskirties adresas `daddr`.
`int gen_IEEE80211_assoc(char daddr[], struct ieee80211_frame *fp)`	Sugeneruoja asociacijos tvindymo atakos kadrą kaip `ieee80211_frame` struktūrą bei patalpina į `fp` rodykle nurodyta vietą. Argumentuose nurodoma: paskirties adresas `daddr`.
`int gen_IEEE80211_deauth(char saddr[], char daddr[], struct ieee80211_frame *fp)`	Sugeneruoja deautentifikacijos atakos kadrą kaip `ieee80211_frame` struktūrą bei patalpina į `fp` rodykle nurodyta vietą. Argumentuose nurodoma: šaltinio adresas `saddr`, paskirties adresas `daddr`.
`int gen_IEEE80211_diass(char saddr[], char daddr[], struct ieee80211_frame *fp)`	Sugeneruoja diasociacijos atakos kadrą kaip `ieee80211_frame` struktūrą bei patalpina į `fp` rodykle nurodyta vietą. Argumentuose nurodoma: šaltinio adresas `saddr`, paskirties adresas `daddr`.
`int gen_IEEE80211_probe(char daddr[], struct ieee80211_frame *fp)`	Sugeneruoja zondavimo tvindymo atakos kadrą kaip `ieee80211_frame` struktūrą bei patalpina į `fp` rodykle nurodyta vietą. Argumentuose nurodoma: paskirties adresas `daddr`.

4. SEKOS ANALIZE PAREMTŲ APSAUGOS ALGORITMŲ TYRIMAS

4.1. Tyrimo tipas

Šiame darbe yra naudojamas *taikomojo tyrimo tipas*[5], kuris šio darbo kontekste yra viešai prieinamos belaidžių tinklų saugos mokslo žinių bazės panaudojimas, siekiant sudaryti apsaugos nuo atsisakymo aptarnauti atakų algoritmą, pagrįstą kadrų sekos numerių analize, ir šį algoritmą galimai pritaikyti praktiškai. Kitaip tariant – mokslo žinių bazė naudojama siekiant išspręsti praktinę problemą.

Tyrimų metu yra vadovaujamasi *apibendrintu konstruktyviu tyrimo metodu*[6], kurio schema pateikta 49 pav. paveikslėlyje. Probleminė sritis, teorinė žinių bazė ir pradinė informacija yra aprašytos ir išanalizuotos analitinėje darbo dalyje, o apsaugos algoritmą ir programinį modelį sudarėme projektinėje dalyje. Tęsiant loginę darbo seką, tiriamojoje dalyje naudodami suprojektuotą programinį atakų ir apsaugos algoritmų tyrimo modelį, tirsime apsibrėžtas algoritmų charakteristikas ir pagal jas lyginsime sudarytąjį algoritmą EFRR su jau esamais algoritmais: GAP, SNRA ir FRR.



49 pav. *Apibendrintas konstruktyvusis tyrimo metodas*

4.2. Tyrimo metodika

Tyrimų metu, taikant šiame skyriuje aprašomą tyrimo metodiką, įvertinami analitinėje dalyje aprašyti apsaugos algoritmai, pagrįsti kadro sekos numerio analize:

- *GAP* – algoritmas, pagrįstas sekos numerio skirtumo nustatymu tarp gretimų kadrų.
- *SNRA* – algoritmas, pagrįstas sekos numerio kitimo greičio nustatymu tarp gretimų kadrų.
- *FRR* – algoritmas, pagrįstas kadrų eilės (lango) skirtumo skaičiavimu tarp kiekvienų gretimų eilės kadrų.

[5] Taikomasis tyrimas (angl. *Applied research*) – tai vienas iš tyrimo tipų, paremtas akadaminės ar specializuotos tyrėjų bendruomenės sukauptos žinių bazės (metodų, algoritmų, teorijų ir pan.) panaudojimu sprendžiant tam tikras taikomo ar praktinio pobūdžio problemas [22].

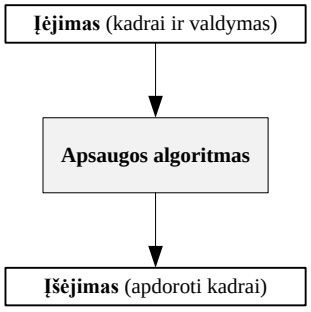
[6] Konstruktyvusis tyrimo metodas (angl. *Constructive research*) remiasi konstrukcijos (algoritmo, teorijos, modelio, programinės įrangos ar pan.) analitiniu plėtojimu ir bandymais, remiantis iš anksto apsibrėžtais kriterijais. Tai populiariausias metodas informatikoje [22].

- *EFRR* – algoritmas, pagrįstas dinaminiu kadrų eilės formavimu priklausomai nuo kadrų gavimo spartos ir rinkinio parametrų skaičiavimu kiekvienam eilės kadrui.

Sekančiuose skyreliuose pateikiami konrečių algoritmus įvertinančių parametrų ir charakteristikų aprašai bei tyrimų rezultatai.

4.2.1. Algoritmų įvertinimo parametrai

Algoritmų tyrimui naudojamas „juodos dėžės“ principas – kiekvienas algoritmas laikomas atskira posisteme, o jos veikimo rezultatai įvertinami operuojant įėjimo ir išėjimo duomenų lyginimo operacijomis (žr. 50 pav.).



50 pav. *Juodos dėžės principas*

Tyrimo kontekste, įėjimas suprantamas kaip algoritmui pateikiami tinklo kadrai ir valdymas, o išėjimas – apdoroti tinklo kadrai, gaunami blokuojant tikėtinai nelegalius tinkle kadrus įėjimo sraute. Remiantis 50 pav., visus įvertinimo parametrus galima skirstyti į tris grupes:

- *Algoritmo efektyvumo įvertinimo parametrai (išvestis).*
- *Srauto įvertinimo parametrai (įvestis).*
- *Algoritmo operaciniai parametrai.*

4.2.1.1. Efektyvumo įvertinimo parametrai

Algoritmų efektyvumas nustatomas lyginant jų įvestį su išvestimi, t.y. lyginant priimtus kadrus su apdorotais. Efektyvumą įvertinantys parametrai pateikti lentelėje Nr. 14.

Lentelė Nr. 14 *Algoritmų blokavimo efektyvumo įvertinimo parametrai*

Žymėjimas	Aprašymas	Mat. vienetai/išraiška
BN	Blokuotų nelegalių kadrų kiekis	%
BL	Blokuotų legalių kadrų kiekis	%

Tiriamieji algoritmai veikia su tam tikra sėkmingo nelegalių kadrų identifikavimo tikimybe, todėl be blokuotų nelegalių kadrų kiekio, kuris tiesiogiai įvertina algoritmo atakos malšinimo efektyvumą ir idealiu atveju turėtų visada būti lygus 100 %, taip pat naudojamas blokuotų legalių kadrų kiekio įvertinimas, siekiant įvertinti netinkamai identifikuotų kadrų kiekį. Savaime suprantama, blokuotų legalių kadrų kiekis idealiu atveju turėtų būti lygus 0%, tačiau kaip pamatysim iš tyrimo rezultatų, modeliavimo metu gauti parametrai yra tik artimi savo idealioms reikšmėms.

4.2.1.2. Apdorojamo srauto įvertinimo parametrai

Algoritmo srautą įvertinantys parametrai įvertina apsaugos algoritmui pateikiamos kadrų eilės savybes. Srauto parametrai apibendrinti lentelėje Nr. 15.

Lentelė Nr. 15 *Algoritmų apdorojamo srauto parametrai*

Žymėjimas	Aprašymas	Mat. vienetai/išraiška
N	Nelegalių kadrų kiekis įėjimo sraute	%
A	Nelegalių tinklo adapterių kiekis	vnt.
P	Prarastų kadrų kiekis	%
T	Efektyvus laikas[7]	s

Srauto įvertinimo parametrai naudojami modelio posistemėje, kurioje yra generuojami virtualūs kadrai. Keičiant šių parametrų reikšmes, keičiasi sugeneruotos kadrų eilės savybės (nelegalių kadrų kiekis, tinklo plokščių kiekis, prarastų kadrų kiekis ir pan.), kurios savo ruožtu keičia algoritmų veikimo efektyvumą. Šie parametrai apibrėžia modeliuojamo srauto savybes, o pačius algoritmus įtakoja netiesiogiai.

4.2.1.3. Operaciniai parametrai

Algoritmų operaciniai parametrai įvertina algoritmo darbo režimų įtaką veiklos rezultatams ir algoritmų įtaką sistemai, kurioje jie veikia. Parametrai pateikti lentelėje Nr. 16.

Lentelė Nr. 16 *Algoritmų operaciniai parametrai*

Žymėjimas	Aprašymas	Mat. vienetai/išraiška
L	Kadrų eilės (lango) ilgis	vnt.
D	Kadrų eilės numerio skirtumas	vnt.
S	Kadrų eilės numerio kitimo greitis	vnt./s
B	Sisteminis buferis	MB

[7] Tiriamasis modelis nėra realaus laiko, todėl laikas modeliuojamas įvertinant gautų kadrų kiekį ir sumuojant kiekvieno kadro laiko žymų skirtumus.

Kadrų sekos numerio skirtumas ir kitimo greitis įeina į visų algoritmų identifikacijos sąlygas. Kadrų eilės ilgis yra naudojamas FRR ir EFRR algoritmuose, todėl tikslinga ištirti algoritmų efektyvumo priklausomybę nuo šio parametro. Sisteminis buferis visų tyrimų metu yra laikomas patovus 2MB (arba 50000 kadrų) ilgio ir keičiamas tik tiriant algoritmo vykdymo laiką.

4.2.2. Tyrimo eiga

Tyrimo eigos metu nustatomi algoritmų veiklos scenarijai ir tiriamos charakteristikos, parenkamos konstantų vertės ir atliekamas modelio programos vykdymas (simuliacija). Tiriamų charakteristikų sudarymui naudojama 4.2.1 skyrelyje nusistatyi parametrai.

4.2.2.1. Tiriami scenarijai ir charakteristikos

Algoritmų tyrimų scenarijai parinkti pagal tai, kokia tipinė atsisakymo aptarnauti ataka naudojama modelyje:

- *Deautentifikacija* – šio scenarijaus metu naudojama deautentifikacijos ataka: priimamas srautas pasižymi atakos kadrais, nukreiptais selektyviai vieno vartotojo blokavimui. Diasociacijos ir reasociacijos ataka veikia analogiškai[8] ir algoritmuose interpretuojamos analogiškai šiai atakai, todėl nėra prasmės jų išskirti kaip atskirų scenarijų.
- *Autentifikacijos tvindymas* – šio scenarijaus metu naudojamas didelis kiekis deautentifikacijos kadrų siunčiamų įvairiais vartotojų adresais. Ataka nukrepta prieš prieigos įrangą. Zondavimo tvindymas ir asociacijos tvindymas yra analogiški šiai atakai, todėl nėra prasmės jų išskirti kaip atskirų scenarijų.

Tiriamos charakteristikos pateiktos lentelėje Nr. 17.

Lentelė Nr. 17 *Tiriamosios algoritmų charakteristikos*

Nr.	Charakteristika
1.	$BN = F(\mathbf{N}, (P, A, L, T) = \mathbf{const})$ $BL = F(\mathbf{N}, (P, A, L, T) = \mathbf{const})$
2.	$BN = F(\mathbf{P}, (N, A, L, T) = \mathbf{const})$ $BL = F(\mathbf{P}, (N, A, L, T) = \mathbf{const})$
4.	$BN = F(\mathbf{A}, (P, N, L, T) = \mathbf{const})$ $BL = F(\mathbf{A}, (P, N, L, T) = \mathbf{const})$
5.	$BN = F(\mathbf{L}, (P, A, N, T) = \mathbf{const})$ $BL = F(\mathbf{L}, (P, A, N, T) = \mathbf{const})$
6.	$BN = F(\mathbf{T}, (P, A, L, N) = \mathbf{const})$ $BL = F(\mathbf{T}, (P, A, L, N) = \mathbf{const})$

[8] Analogiškas atakų veikimas šio darbo kontekste reiškia, jog atakų mechanizmai skiriasi tik kadro antrašte, o visi kiti aspektai yra labai artimi (naudojamų kadrų kiekis, poveikis sistemoje, naudojamas tinklo pažeidžiamumas ir kt.)

Charakteristikos tiriamos ties *kiekvienu scenarijumi*. Bendu atveju, tiriamos charakteristikos nusako, kaip algoritmų efektyvumą įvertinantys blokuotų nelegalių kadrų kiekio ir blokuotų legalių kadrų kiekio parametrai priklauso nuo vieno iš srauto ar operacinių parametrų, kitus parametrus tiriamuoju momentu laikant pastovius.

Atskirai eksperimentiškai įvertinamas algoritmo vykdymo greitis. Charakteristika pateikta lentelėje Nr. 18. Ji pateikta atskirai todėl, kad tik tiriant šią charakteristiką yra keičiamas sisteminio buferio dydis, taip pat, ši charakteristika apibūdina algoritmo efektyvumą procesoriaus laiko eikvojimo prasme, o ne atakos malšinimo prasme.

Lentelė Nr. 18 *Algoritmų vykdymo spartos įvertinimo charakteristika*

Nr.	Charakteristika
1.	O = F(**B**, (N**,** P,A,R,L,T) = **const**)

Konstantinių parametrų reikšmės parenkamos taip, kad joms esant algoritmų efektyvumas būtų didžiausias ir rezultato įtaka labiausiai priklausytų nuo kintamo parametro. Naudotų konstantų reikšmės pateiktos lentelėje Nr. 19.

Lentelė Nr. 19 *Konstantos*

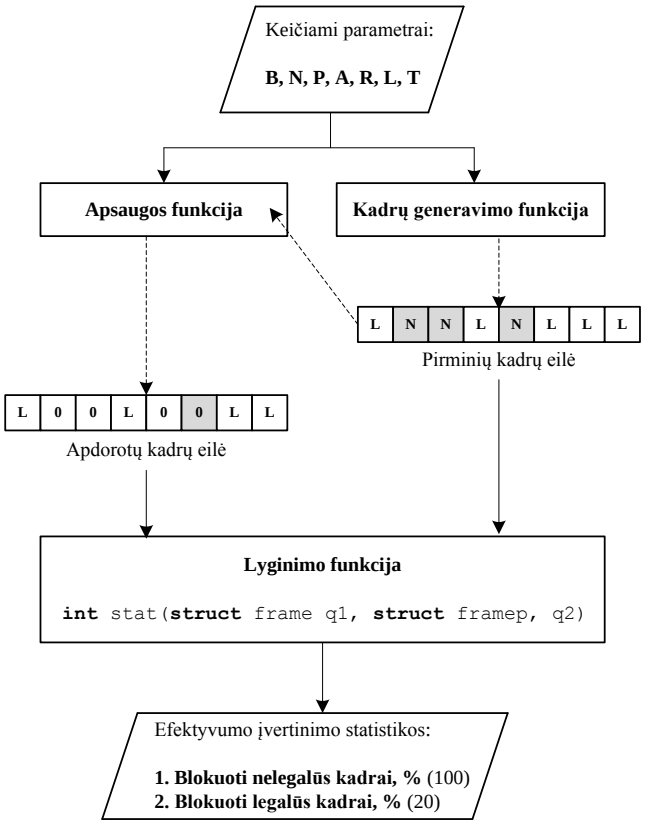
Konstanta	Reikšmė
N	50 %
P	0 %
A	1
L	5
T	300s
B	2MB (50000 vnt.)

4.2.2.2. Tyrimo schema

Loginę tyrimo veiksmų seką ir pagrindinius joje dalyvaujančius modelio realizacijos objektus vaizduoja tyrimo schema (žr. 51 pav.). Tyrimo pradžioje nustatomi parametrai, reikalingi modelio vykdymui. Šie parametrai valdo kadrų generavimo funkcijos ir apsaugos funkcijos (algoritmo) veikimą. Pagal nurodytus parametrus, kadrų generavimo funkcija sugeneruoja C struktūrų masyvą, kuris modeliuoja kadrų eilę ir yra patalpintas pirminių kadrų eilėje. Šį masyvą priima apsaugos funkcija, ir blokuodama identifikuotus nelegalius kadrus, perrašo jį į apdorotų kadrų eilę. Lyginimo funkcija priima abu minėtuosius buferius ir išduoda algoritmo vykdymo efektyvumo įvertinimo parametrų statistikas: blokuotų legalių ir nelegalių kadrų kiekius.

Visa ši seka kartojama keičiant tam tikro įėjimo parametro reikšmę ir fiksuojant efektyvumo parametrų reikšmes po simuliacijos. Taip gaunama efektyvumo įvertinimo parametrų priklausomybė nuo srauto ir operacinių parametrų, kuri reprezentuoja 4.2.2.1 skyrelyje nusistatytas tiriamas algoritmų

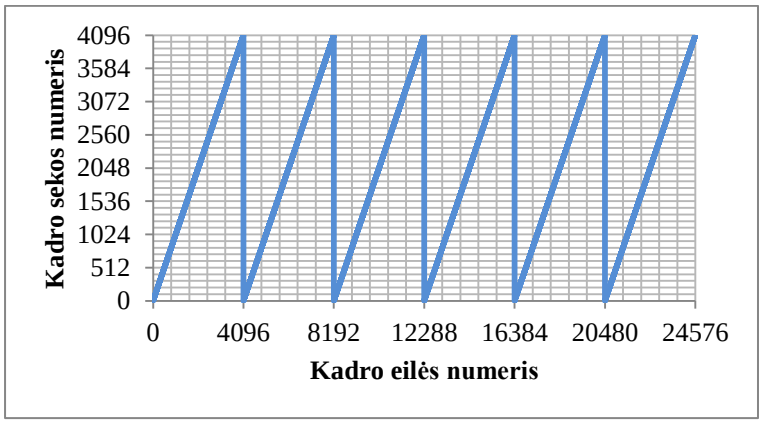
charakteristikas. Modelis statistikas spausdina į Linux standartinę išvestį, todėl naudojant operacinės sistemos išvesties nukreipimo į bylą mechanizmą, rezultatai išsaugomi tekstinėje byloje, kuri po to apdorojama su Excel programiniu paketu. Charakteristikų tyrimo rezultatai peteikti sekančiame skyrelyje.



51 pav. *Tyrimo vykdymo schema*

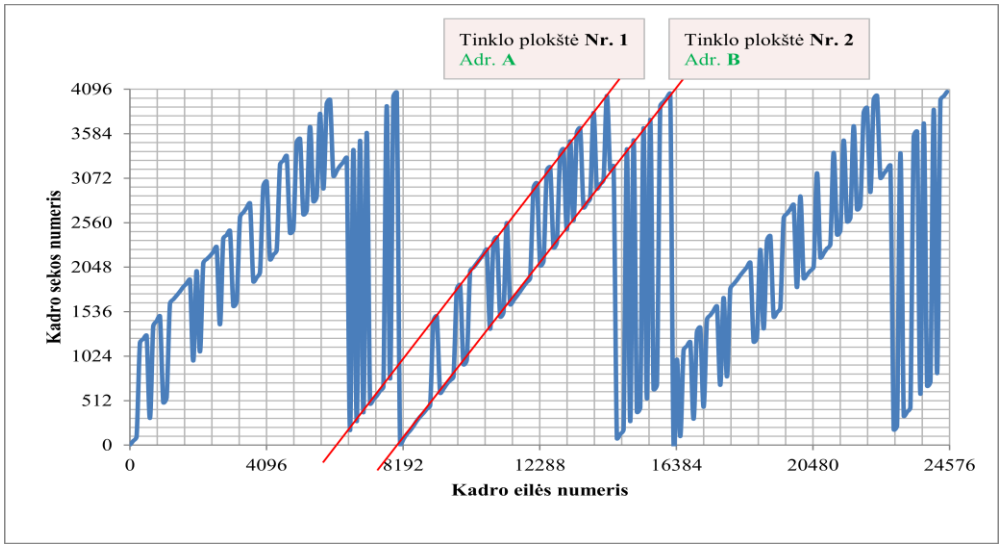
4.3. Rezultatai

Prieš pradedant tirti charakteristikas, modelis išbandomas modeliuojant legalių kadrų srautą. 52 pav. pateikta sumodeliuotos 24576 kadrų ilgio eilė sekos numerio priklausomybė nuo kadro eilės numerio.



52 pav. *Modelio demonstravimas. 1 plokštės sekos numerio kitimas*

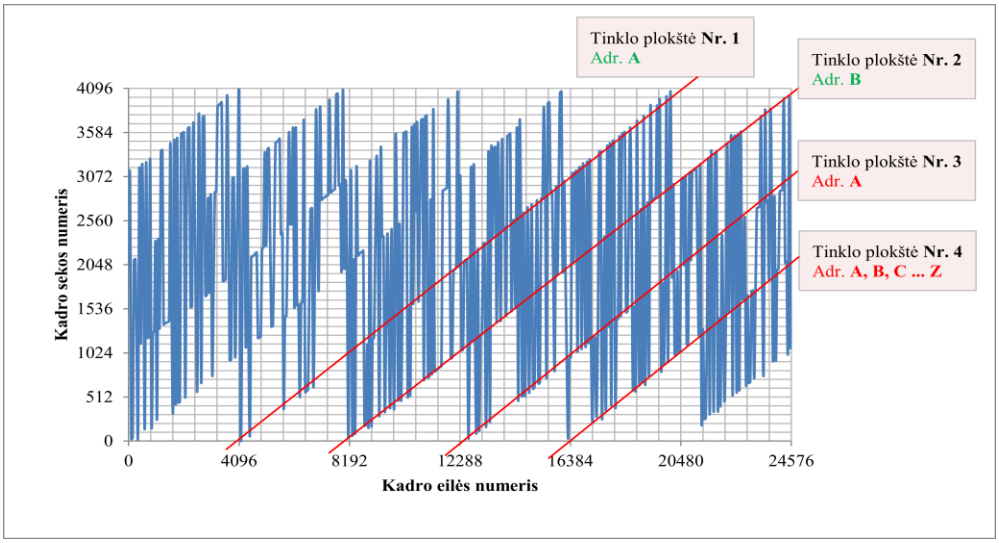
Modeliuojamas 1 tinklo plokštės srautas, todėl sekos numeris didėja tiesiškai ir kinta periodiškai, nes sekos numeriui skirtas laukas yra 12 bitų ilgio ir sekos numerio skaitliukas „persisuka“ kas 4096 kadrai. Didėjant tinklo plokščių skaičiui, tokio dėsningumo sekos numerio kitimas kiekvienos plokštės atžvilgiu atsispindi bendrame sekos numerio priklausomybės nuo kadro eilės numerio grafike. 53 pav. pavaizduota 2 legalių tinklo plokščių generuojamų kadrų sekos numerio priklausomybė nuo kadro eilės numerio. Skirtumas tarp plokščių sekos numerių šiuo atveju yra 1024. Raudonos linijos grafike žymi individualių ploščių sugeneruotų kadrų sekos numerio kitimo kreives. Modeliuojamos 2 tinklo plokštės generuoja tokį patį kiekį kadrų (tokia pačia sparta ir pasiskirstymu), todėl individualių plokščių sekos numeris periodiškai tampa lygus 0 kas 8192 kadrai, nes reikia dvigubai ilgesnės eilės, kad individualių plokščių sekos numerio skaitliukas „persisuktų“.



53 pav. *Modelio demonstravimas. 2 plokščių sekos numerio kitimas*

Remiantis tokiais dėsningumais, iš modeliavimo metu sugeneruoto srauto kadrų eilės galima nustatyti, kiek tinklo plokščių jį sugeneravo ir kiekvieną kadrą priskirti vienai iš nustatytų plokščių. Jeigu laikomės prielaidos, kad suklastoti sekos numerį yra neįmanoma arba labai sudėtinga, o skirtingos plokštės naudoja skirtingus MAC adresus, galima daryti išvadą, jog visi kadrai yra legalūs, tačiau galimos ir kitos situacijos. 54 paveikslėlyje pavaizduotame grafike analogišku būdu galima išskirti 4 tinklo plokščių generuojamus kadrus. Vadovaujantis minėtomis prielaidomis, 54 pav. pavaizduotų 1 ir 2 plokčių generuotas srautas gali būti laikomas kaip legalus, tačiau 3 plokštė dubliuoja 1 plokštės adresą, o 4 plokštės generuojami kadrai adresą apskritai keičia tam tikra tvarka. Esant tokiai situacijai, galima daryti išvadą, jog 4 plokštės generuojamas srautas yra nelegalus, nes nesilaiko standarto taisyklių, taip pat viena

iš 1 ir 2 plokščių yra nelegali, nes negali būti dvi tinklo plokštės su tuo pačiu adresu vienoje belaidės prieigos zonoje.



54 pav. *Modelio demonstravimas. 4 plokščių sekos numerio kitimas*

Grafikuose, pateiktuose 52-53 pav., rezultatai yra artimi analitinėje dalyje analizuotų autorių straipsniuose pateiktiems sekos numerio kitimo rezultatams (žr. 34-36 pav.) ir demonstruoja tik patį sekos numerio kitimo dėsningumą bei galimą jo pritaikymą identifikuojant nelegalius tinklo kadrus. Konkrečių scenarijų ir charakteristikų rezultatai pateikiami sekančiuose skyreliuose.

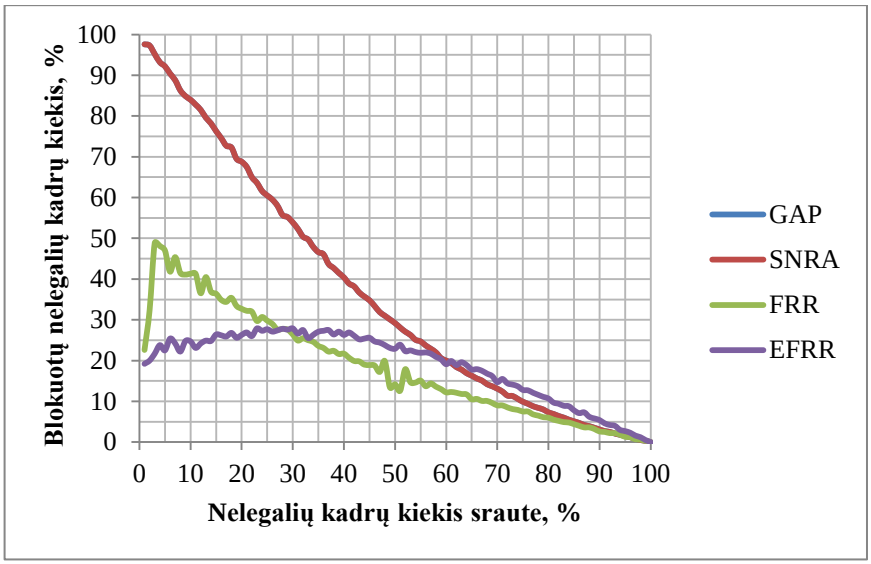
4.3.1. Deautentifikacijos scenarijus

Deautentifikacijos scenarijaus metu generuojami deautentifikacijos atakos kadrai, kurie atsitiktinai paskirstomi tarp legalių kadrų. Šis scenarijus taip pat reprezentuoja diasociacijos ir reasociacijos atakas, nes skirtumas tarp šių atakų mechanizmų yra tik kadro antraštės kontrolės lauko (FC) reikšmė. Šiame scenarijuje tiriamos algoritmų blokuotų legalių kadrų ir blokuotų nelegalių kadrų priklausomybės nuo srauto ir operacinių parametrų pagal 4.2.2.1 skyrelyje nusistatytas tiriamąsias charakteristikas. Visi parametrai išskyrus kintamąjį yra konstantos. Visų rezultatų kontekste naudojama sąvoka „efektyvumas“ reiškia blokuotų nelegalių kadrų ir blokuotų legalių kadrų kiekius, kaip pagrindinius parametrus, įvertinančius algoritmų veikimą.

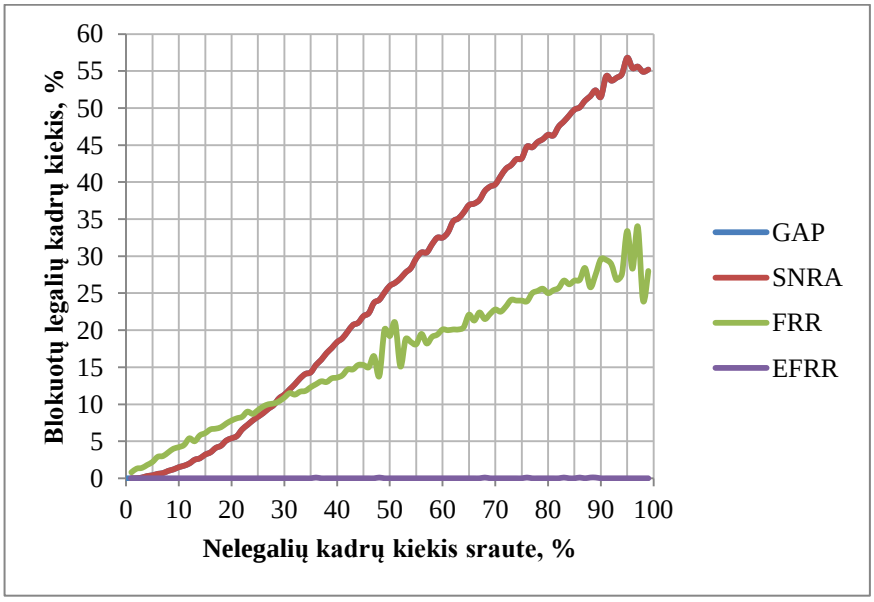
4.3.1.1. Efektyvumas nuo nelegalių kadrų kiekio

Nelegalių kadrų kiekio įtaka algoritmų veiklai tiriama keičiant nelegalių kadrų kiekį nuo 0 iki 100% ir fiksuojant algoritmo blokavimo savybes. 55 pav. pateikta blokuotų nelegalių kadrų kiekio priklausomybė, o 56 pav. – blokuotų legalių kadrų. Visiems algoritmams galioja bendras principas: kuo

daugiau nelegalių kadrų, tuo jų blokavimo efektyvumas mažėja. Nors sudarytojo EFRR algoritmo maksimalus blokuotų nelegalių kadrų kiekis buvo mažiausias iš visų algoritmų – 27,9% , tačiau jo blokuotų legalių kadrų kiekis visame kitimo diapazone neviršijo 0,1% ir vidutiniškai buvo lygus 0,008%, kai tuo tarpu GAP/SNRA[9] ir FRR algoritmai pasižymėjo atitinkamai 56,8% ir 34% maksimaliais legalių kadrų blokavimo kiekiais, kurie tiesiškai didėjo, didėjant nelegalių kadrų kiekiui.



55 pav. *Deautentifikacija. Nelegalių kadrų blokavimas nuo atakos kadrų kiekio*

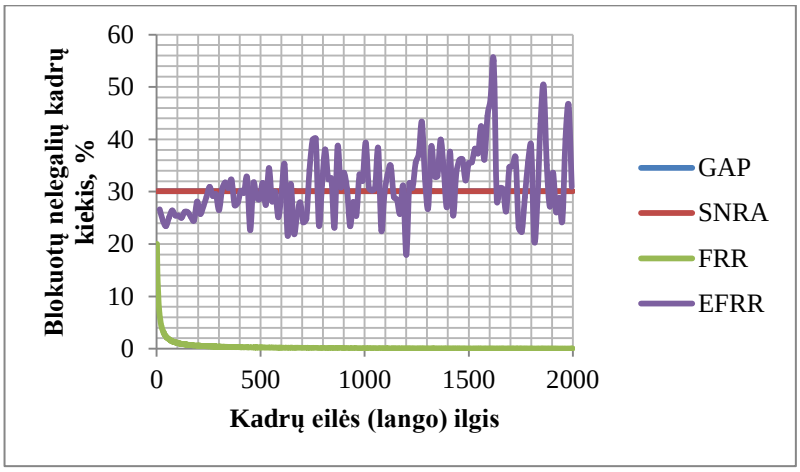


56 pav. *Deautentifikacija. Legalių kadrų blokavimas nuo atakos kadrų kiekio*

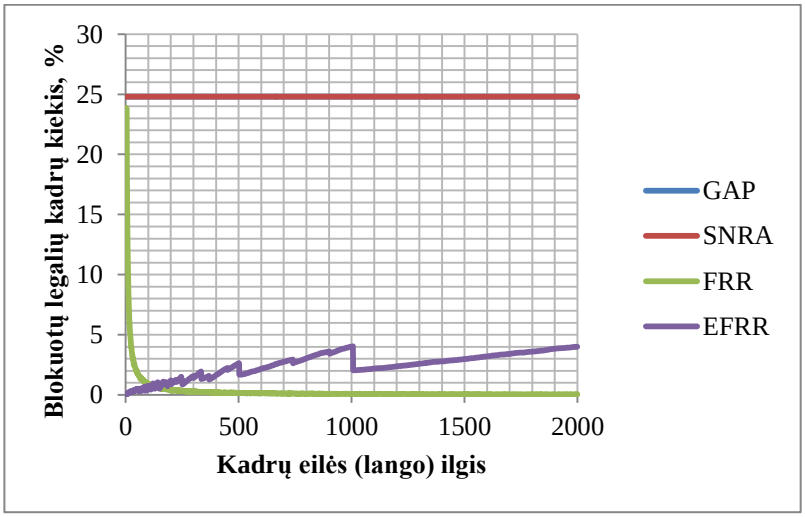
[9] GAP ir SNRA algoritmai, nesant prarastų kadrų, veikia identiškai ir grafikuose jų kreivės persidengia. Dėl šios priežasties, komentuojant rezultatus, jie abu vienu metu įvardijami žymenimi „GAP/SNRA“.

4.3.1.2. Efektyvumas nuo kadrų eilės ilgio

Kadrų eilės ilgis – tai tik FRR ir EFRR algoritmų naudojamas parametras, todėl GAP/SNRA algoritmo rezultatai nuo jo nepriklauso ir yra pastovūs. Lango ilgis buvo keičiamas nuo 1 (logiškai mažiausios reikšmės) iki 2000, kuri yra maksimali, siekiant nesukelti didesnio nei nusistatyto vėlinimo, kadrams laukiant eilėje. 57 pav. pateikta blokuotų nelegalių kadrų kiekio priklausomybė, o 58 pav. – blokuotų legalių kadrų. Sudarytasis EFRR algoritmas pasižymėjo atsitiktiniu blokuotų nelegalių ir legalių kadrų kiekio kitimu su vidutinėmis 31,05% ir 2.52% vertėmis. FRR algoritmo efektyvumas ryškiai sumažėja esant 200 kadrų ir ilgesnei eilei, vidutiniškai generuodamas 0,35% nelegalių ir 0,31% legalių kadrų blokavimo kiekius. GAP ir SNRA algoritmai pasižymėjo pastoviais 30,1% nelegalių ir 24,8% legalių kadrų blokavimo kiekiais.



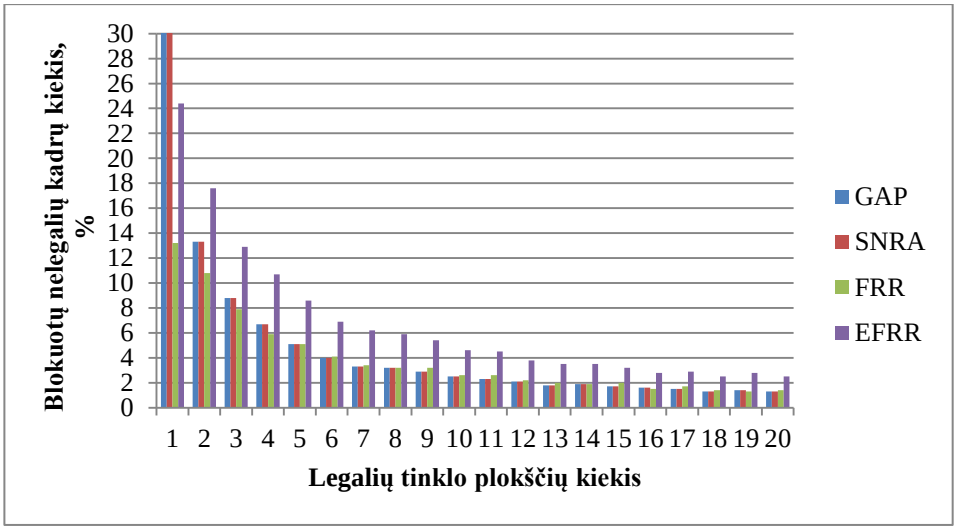
57 pav. *Deautentifikacija. Nelegalių kadrų blokavimas nuo eilės ilgio*



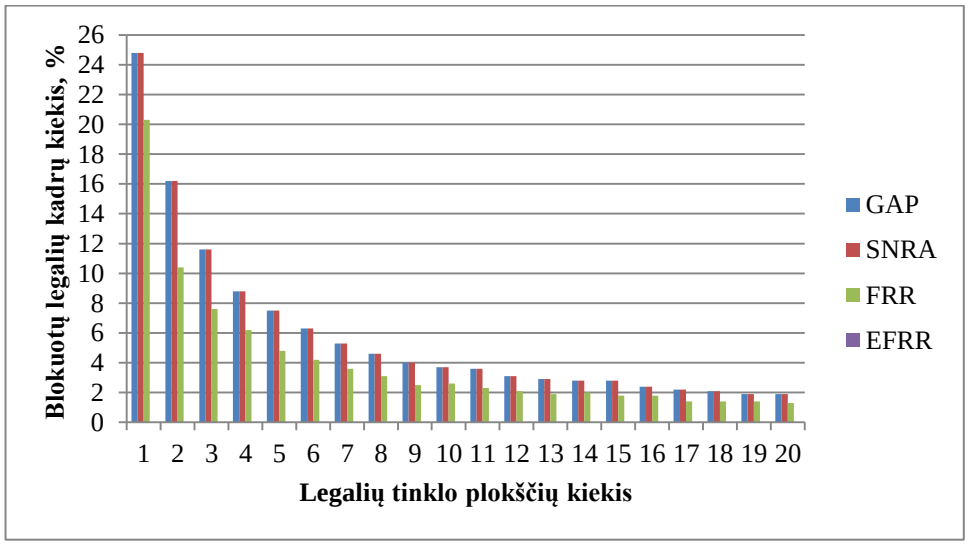
58 pav. *Deautentifikacija. Legalių kadrų blokavimas nuo eilės ilgio*

4.3.1.3. Efektyvumas nuo legalių tinklo plokščių skaičiaus

Plokščių kiekis keičiamas nuo 1 iki 20, laikantis prielaidos, jog tipinis vieno prieigos taško aptarnaujamų belaidžio tinklo mazgų kiekis yra ne didesnis nei 20 ir toks intervalas yra tikėčiausias realiomis sąlygomis. 59 pav. pateikta blokuotų nelegalių kadrų kiekio priklausomybė, o 60 pav. – blokuotų legalių kadrų. Didėjant adapterių skaičiui, blokavimo efektyvumas prastėja. Šioje charakteristikoje EFRR algoritmas pasižymi mažu blokuotų legalių kadrų kiekiu – ties visais adapterių kiekiais jis buvo lygus 0%, kai tuo tarpu GAP/SNRA algoritmas pasižymėjo 5,9% o FRR – 4,13% vidutiniais kiekiais su atitinkamai 24,8% ir 20,3% maksimaliomis vertėmis. EFRR algoritmo blokuotų nelegalių kadrų kiekis buvo didesnis už GAP/SNRA ir FRR algoritmų kiekius vidutiniškai 30 procentų.



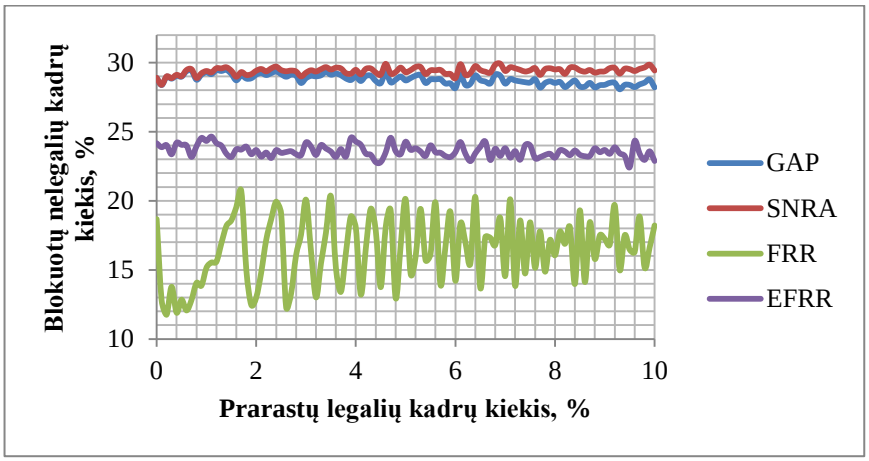
59 pav. *Deautentifikacija. Nelegalių kadrų blokavimas nuo plokščių kiekio*



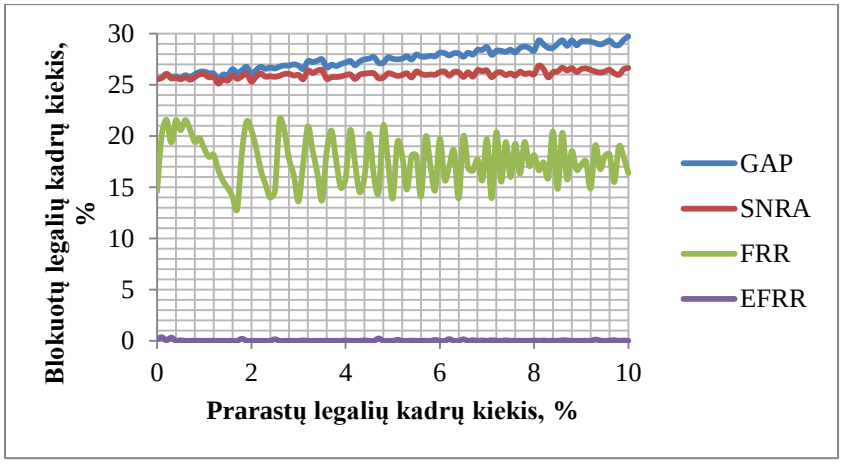
60 pav. *Deautentifikacija. Legalių kadrų blokavimas nuo plokščių kiekio*

4.3.1.4. Efektyvumas nuo prarastų legalių kadrų kiekio

Prarastų kadrų įtaką blokavimams įvertiname tirdami efektyvumo nuo prarastų kadrų kiekio charakteristiką. 61 pav. pateikta blokuotų nelegalių kadrų kiekio priklausomybė, o 62 pav. – blokuotų legalių kadrų. Prarastų kadrų kiekis didinamas nuo 0 iki 10%. Minėtuose grafikuose aiškiai išsiskiria GAP ir SNRA kreivės, kurios iki šiol buvo persidengusios, nes neegzistuojant prarastiems kadrams, GAP ir SNRA algoritmai veikė identiškai. Visų algoritmų blokuotų nelegalių kadrų kiekis kito atsitiktinai (GAP/SNRA ir EFFR atveju išsibarstymas buvo nedidelis), su atitinkamomis vidutinėmis vertėmis: GAP – 28,79%, SNRA – 29,4%, FRR – 16,42% ir EFRR – 23,59%. Sudarytojo EFRR algoritmo blokuotų legalių kadrų kiekis neviršijo 0,36% ir vidutiniškai buvo lygus 0,029%, kai tuo tarpu FRR pasižymėjo vidutiniu 17,54% kiekiu. GAP blokuotų legalių kadrų kiekis labiausiai augo priklausomai nuo prarastų kadrų kiekio, atitinkamai nuo 20,6 iki 29,8%.



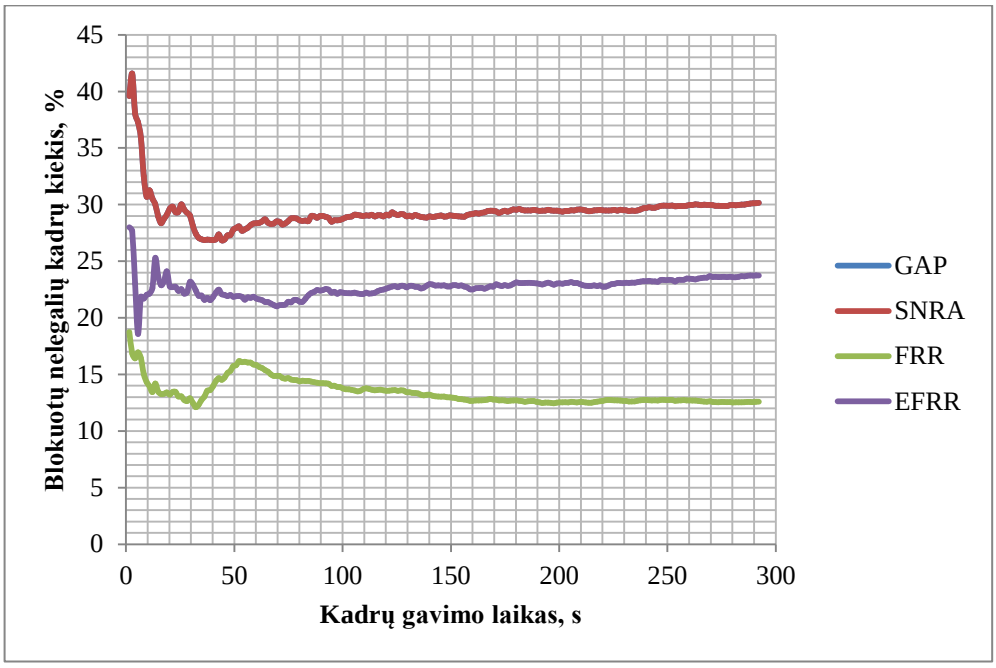
61 pav. *Deautentifikacija. Nelegalių kadrų blokavimas nuo prarastų kadrų kiekio*



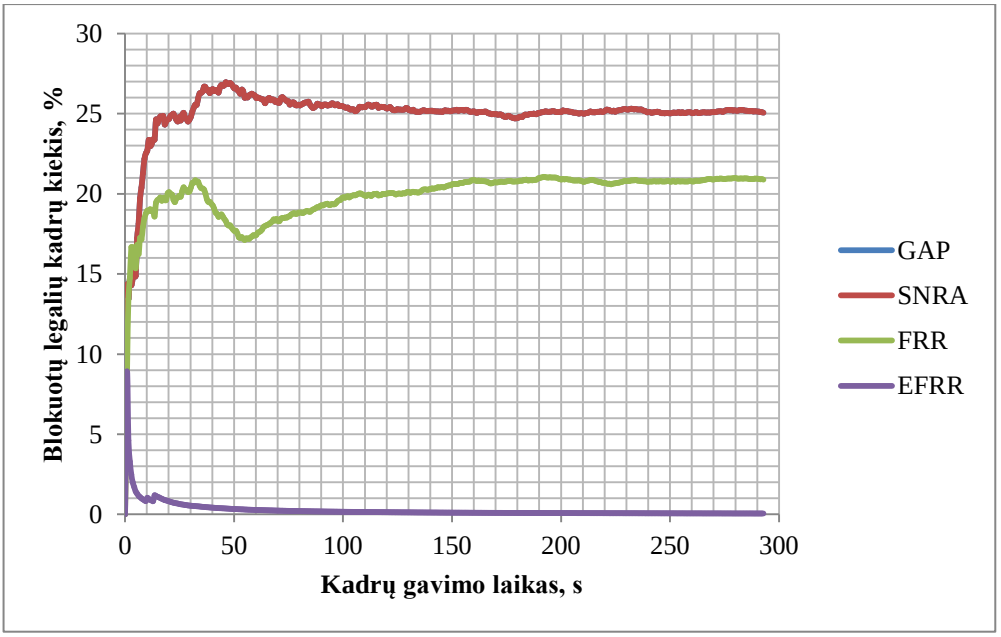
62 pav. *Deautentifikacija. Legalių kadrų blokavimas nuo plokščių kiekio*

4.3.1.5. Efektyvumas nuo sugaišto laiko

Sugaištas laikas modelyje matuojamas sumuojant kadrų laiko žymų skirtumus (modelis nėra realaus laiko). 63 pav. pateikta blokuotų nelegalių kadrų kiekio priklausomybė, o 64 pav. – blokuotų legalių kadrų. Sudarytasis EFRR algoritmas pasižymi mažiausiu 0,25% vidutiniu blokuotų legalių kadrų kiekiu, o blokuotų nelegalių kadrų kiekis vidutiniškai lygus 22,75%.



63 pav. *Deautentifikacija. Nelegalių kadrų blokavimas nuo laiko*



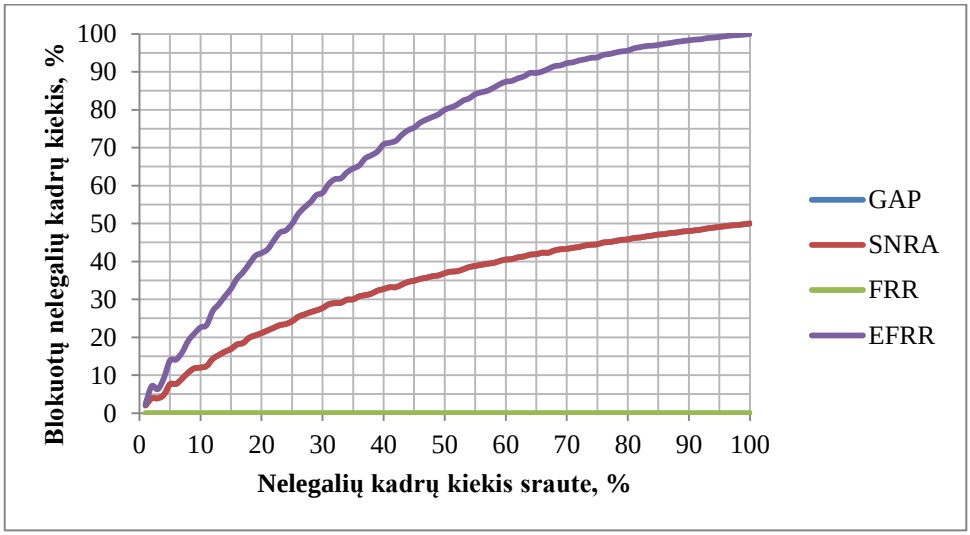
64 pav. *Deautentifikacija. Legalių kadrų blokavimas nuo laiko*

4.3.2. Autentifikacijos tvindymo scenarijus

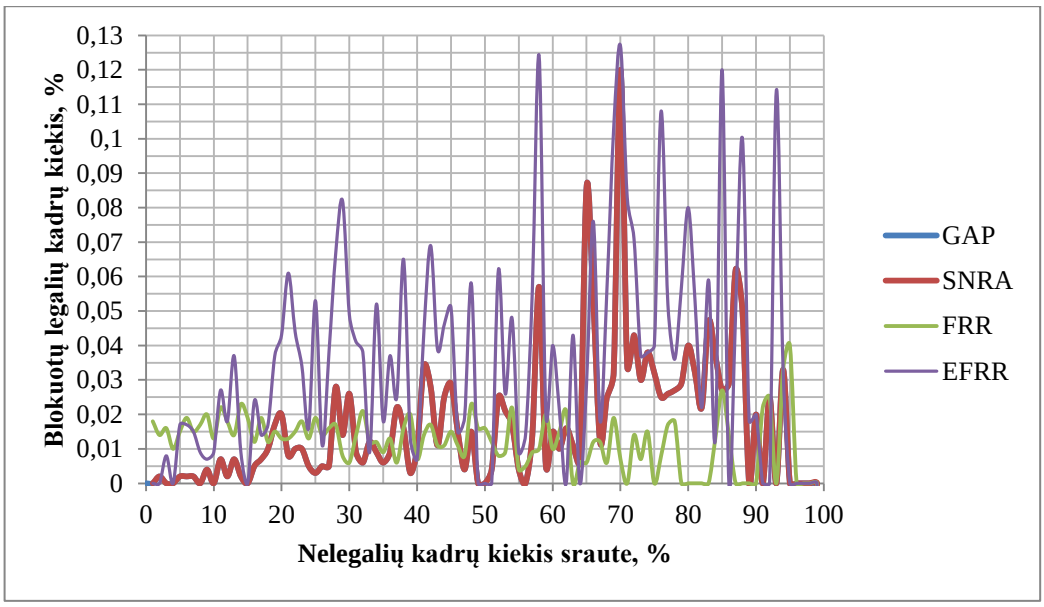
Autentifikacijos tvindymo scenarijaus metu generuojami autentifikacijos atakos kadrai, kurie atsitiktinai paskirstomi tarp legalių kadrų. Šis scenarijus taip pat reprezentuoja zondavimo ir asociacijos tvindymo atakas, nes esminis skirtumas tarp šių atakų mechanizmų yra tik kadro antraštės kontrolės lauko (FC) reikšmė. Šiame scenarijuje tiriamos algoritmų blokuotų legalių kadrų ir blokuotų nelegalių kadrų priklausomybės nuo srauto ir operacinių parametrų pagal 4.2.2.1 skyrelyje nusistatytas tiriamąsias charakteristikas. Visi parametrai išskyrus kintamąjį yra konstantos.

4.3.2.1. Efektyvumas nuo nelegalių kadrų kiekio

Nelegalių kadrų kiekis, kaip ir deautentifikacijos scenarijuje, keičiamas nuo 0 iki 100 procentų. 65 pav. pateikta blokuotų nelegalių kadrų kiekio priklausomybė, o 66 pav.– blokuotų legalių kadrų. Priešingai nei deautentifikacijos scenarijaus atveju, didėjant nelegalių kadrų kiekiui, blokavimo efektyvumas taip pat didėja. Tai galima paaiškinti tuo, kad tvindymo atakos naudoja didelį kiekį skirtingų paskirties MAC adresų ir algoritmai patikimiau atpažįsta nelegalius kadrus, esant didesniam jų kiekiui. FRR algoritmas šios atakos neblokuoja visiškai dėl savo architektūrinių savybių. GAP/SNRA algoritmo blokuotų nelegalių kadrų kiekis pasiekia maksimalią 49,8% ribą, tuo tarpu sudarytasis EFRR algoritmas geba blokuoti iki 100% visų nelegalių kadrų, tačiau tam reikia didelio kiekio atakos kadrų. Priešingai nei deautentifikacijos scenarijaus atveju, visų algoritmų blokuotų nelegalių kadrų kiekis sumažėjo ir neperžengia 0,13% ribos, atsitiktinai pasiskirstydamas apie vidutines reikšmes: GAP/SNRA – 0,017%, FRR – 0,012% ir EFRR – 0,035%.



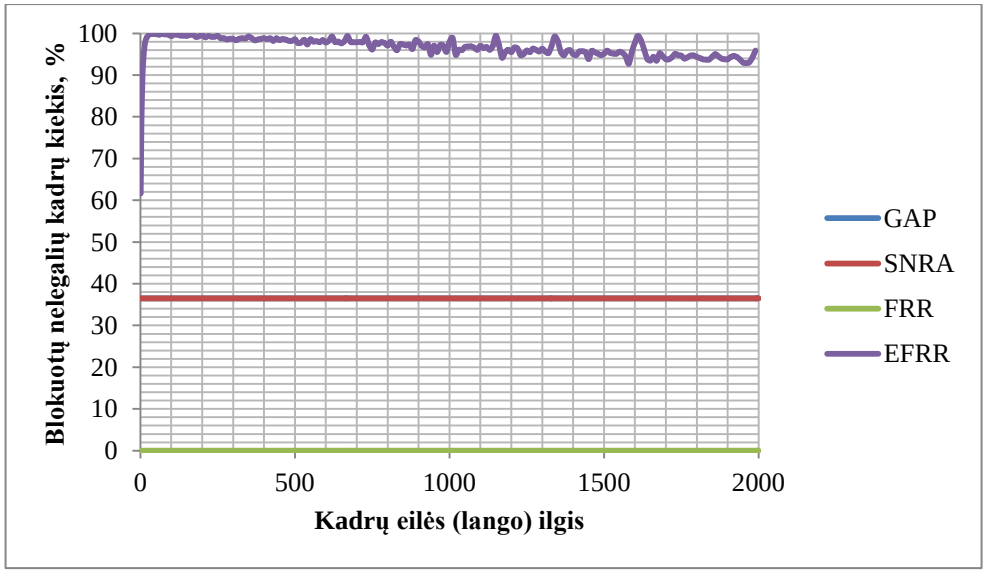
65 pav. *Autentifikacija. Nelegalių kadrų blokavimas nuo atakos kadrų kiekio*



66 pav. *Autentifikacija. Legalių kadrų blokavimas nuo atakos kadrų kiekio*

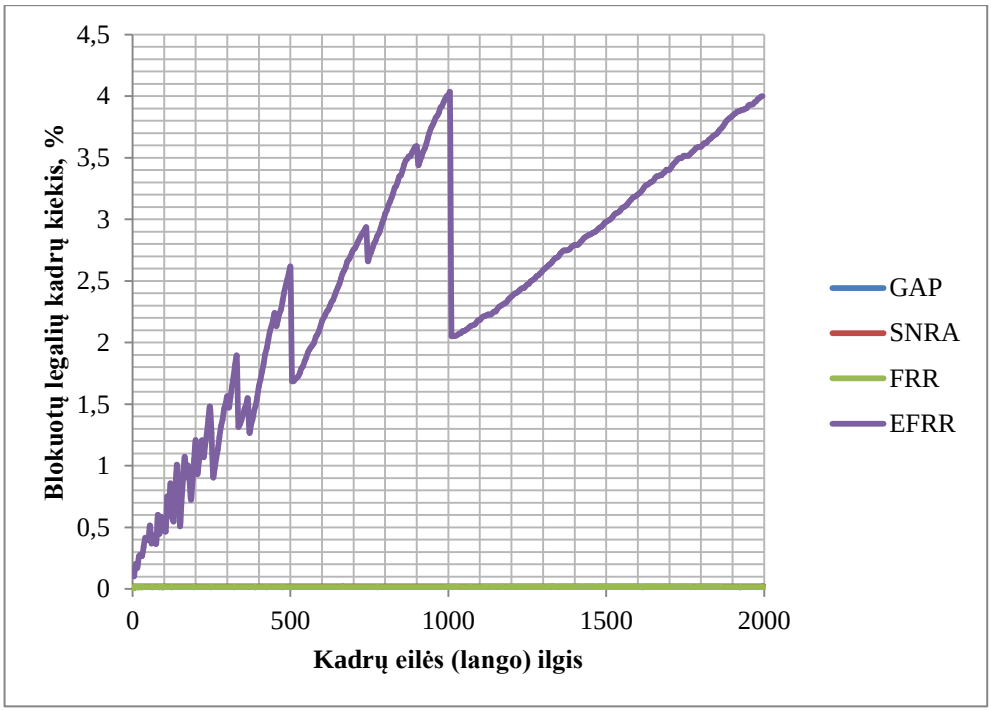
4.3.2.2. Efektyvumas nuo kadrų eilės ilgio

Kadrų eilę naudoja tik FRR ir EFRR algoritmai, todėl GAP/SNRA algoritmo efektyvumas nepriklauso nuo šio parametro ir yra pastovus su 36,5% blokuotų nelegalių kadrų ir 0,02% legalių kadrų kiekiais (žr. 67 pav. ir 68 pav.). FRR algoritmas apskritai negali blokuoti tvindymo atakų, todėl ši charakteristika apibūdina tik sudarytojo EFRR algoritmo savybes. EFRR blokuotų nelegalių kadrų kiekis ties 20 kadrų ilgiu pasiekia maksimalią 99,8% vertę ir toliau krenta visame tiriamajame ilgio intervale iki 92,4% su vidutine 96,8% verte.



67 pav. *Autentifikacija. Nelegalių kadrų blokavimas nuo eilės ilgio*

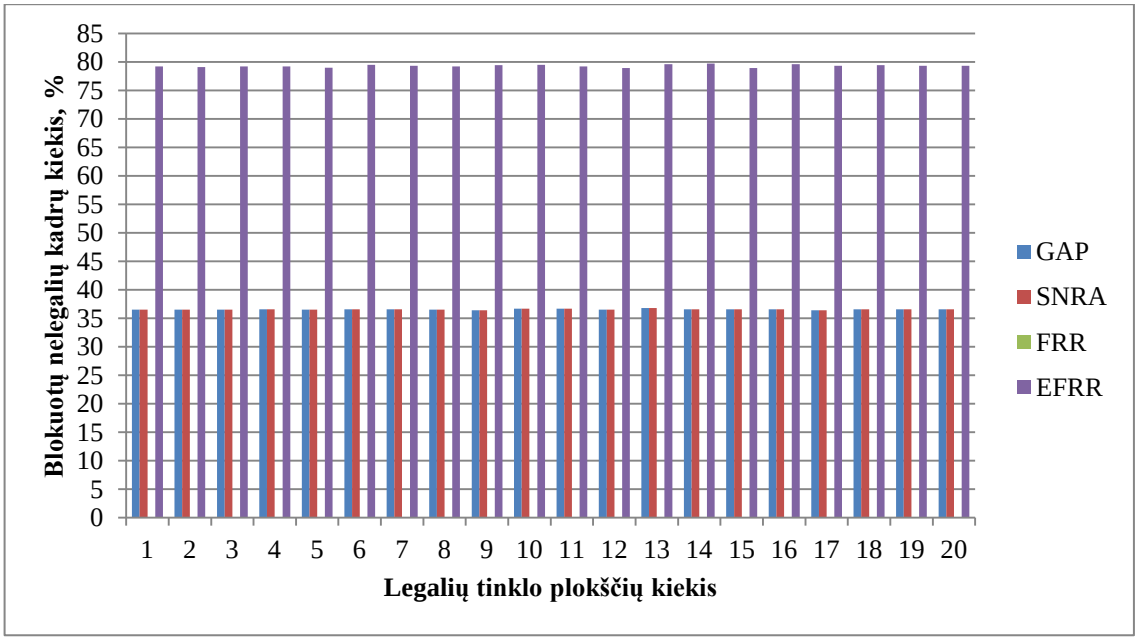
Legalių blokuotų kadrų kiekis tiesiškai didėja, didėjant eilės ilgiui iki 1000 ir pasiekia maksimalią 4,04% vertę. Remiantis grafikais, galima daryti tarpinę išvadą, jog optimalus EFRR algoritmo lango ilgis yra 20, kuriam esant, gaunamas didžiausias teisingo blokavimo kiekis ties pakankamai mažu 0,2% klaidingo blokavimo kiekiu.



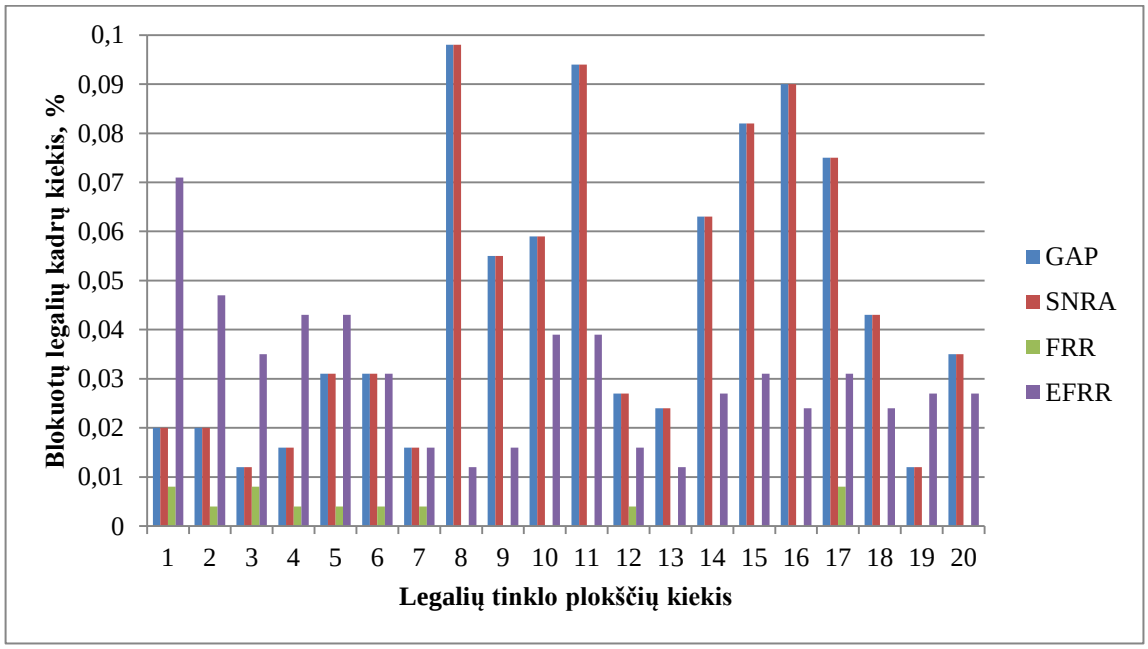
68 pav. *Autentifikacija. Legalių kadrų blokavimas nuo eilės ilgio*

4.3.2.3. Efektyvumas nuo nelegalių adapterių skaičiaus

Adapterių skaičius blokuotų nelegalių kadrų kiekį įtakoja minimaliai: EFRR algoritmas pasižymėjo geriausiu efektyvumu su vidutiniu 79,29% kiekiu, tuo tarpu GAP/SNRA blokuotų nelegalių kadrų kiekis buvo identiškas ir lygus vidutiniškai 36,57% (žr. 69 pav. ir 70 pav.). Tiek GAP/SNRA tiek EFRR algoritmų blokuotų nelegalių kadrų kiekio nuokrypis nuo vidurkio neviršijo 0,8%. Blokuotų legalių kadrų kiekio kitimas buvo atsitiktinis su vidutinėmis vertėmis: GAP/SNRA – 0,045%, FRR – 0,0024% ir EFRR – 0,03%.



69 pav. *Autentifikacija. Nelegalių kadrų blokavimas nuo plokščių kiekio*

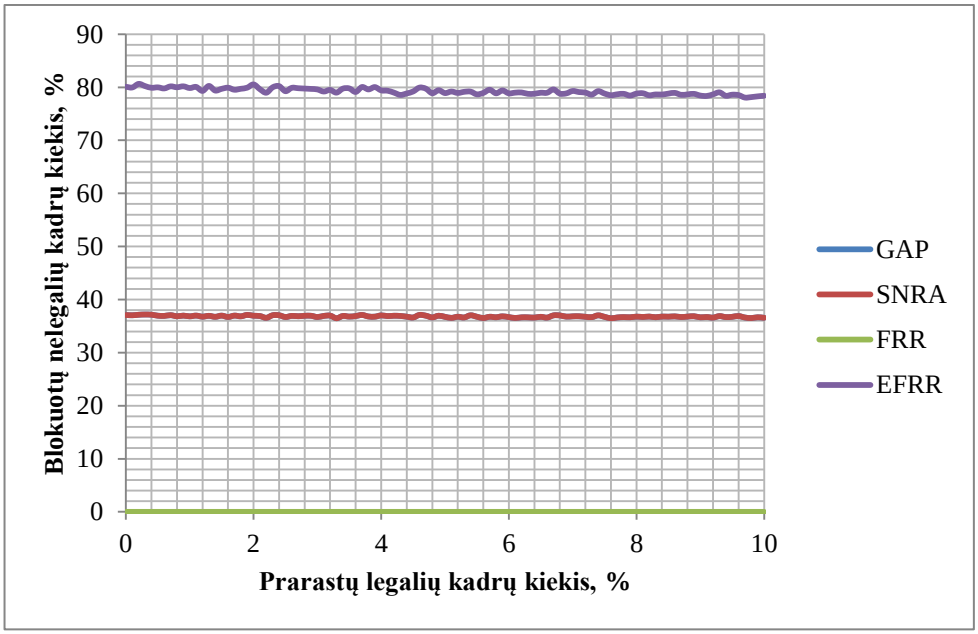


70 pav. *Autentifikacija. Legalių kadrų blokavimas nuo eilės ilgio*

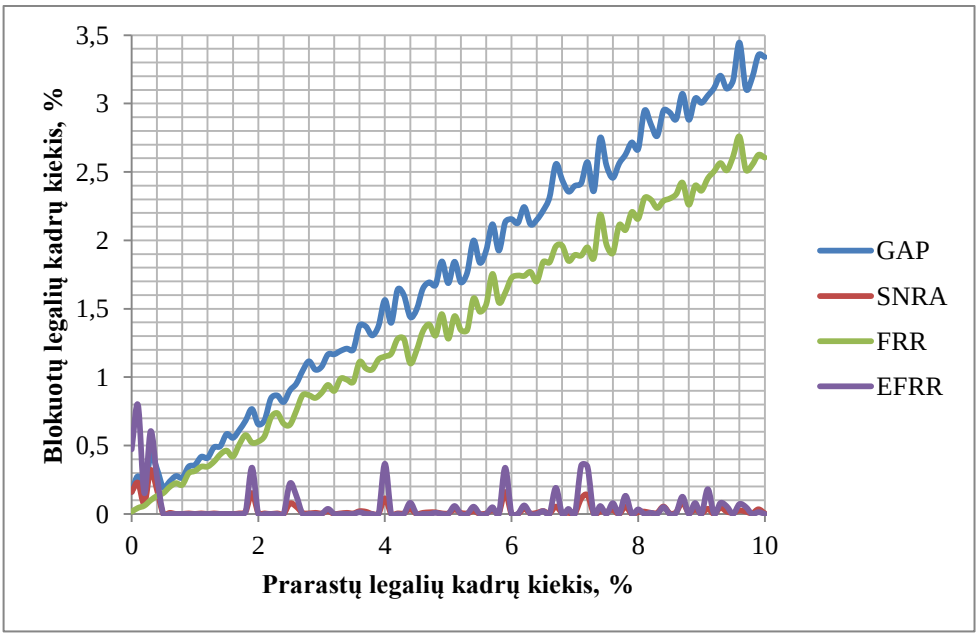
4.3.2.4. Efektyvumas nuo prarastų legalių kadrų kiekio

Didėjant prarastų legalių kadrų kiekiui, blokuotų autentifikacijos tvindymo atakos kadrų kiekis buvo vidutiniškai lygus 79,23% veikiant EFRR algoritmui ir 36,81% veikiant GAP/SNRA algoritmui (žr. 71 pav. ir 72 pav.). Priešingai nei deautentifikacijos atveju, prarastų kadrų faktorius neįtakojo to, kad GAP ir

SNRA algoritmų blokuotų nelegalių kadrų kiekiai išsiskirtų. Legalių blokuotų kadrų atžvilgiu, GAP ir SNRA algoritmai išsiskyrė: GAP algoritmo blokuotų legalių kadrų kiekis tiesiškai didėjo nuo 0,15% iki 3,44%, kai tuo tarpu SNRA pasižymėjo vidutinišku 0,0027% kiekiu.



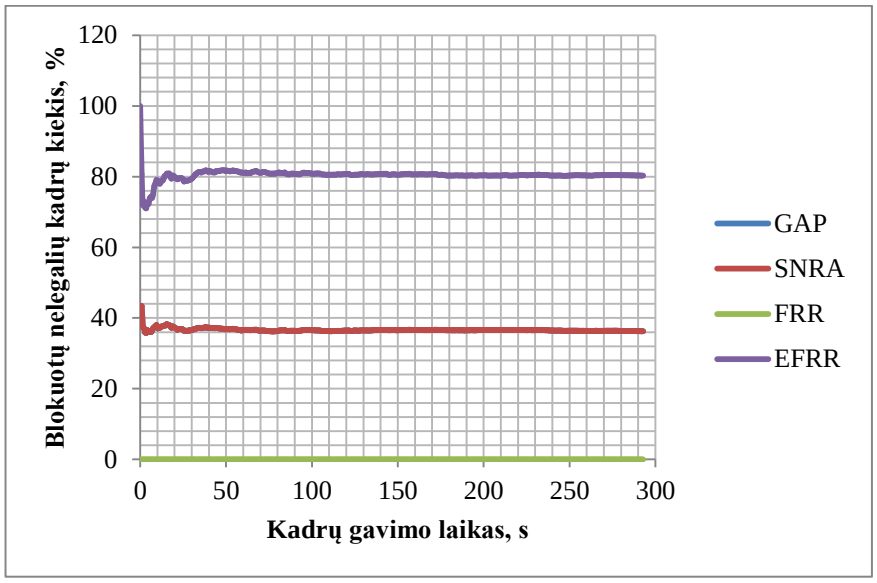
71 pav. *Autentifikacija. Nelegalių kadrų blokavimas nuo prarastų kadrų kiekio*



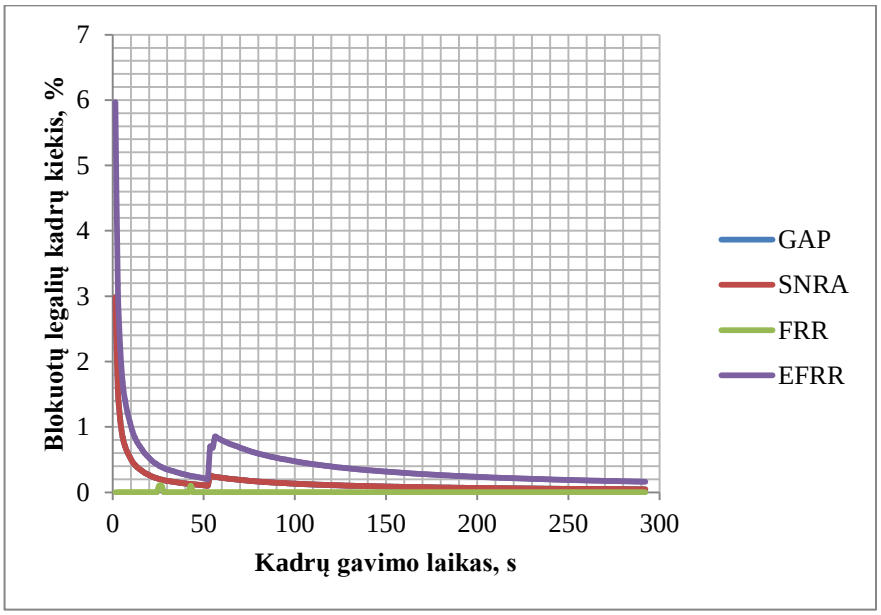
72 pav. *Autentifikacija. Legalių kadrų blokavimas nuo prarastų kadrų kiekio*

4.3.2.5. Efektyvumas nuo sugaišto laiko

Efektyvumas nuo sugaišo laiko pateiktas 73 pav. ir 74 pav. pateiktuose grafikuose. Modelis nėra realaus laiko, todėl laikas modeliuojamas sumuojant kadrų laiko žymas. Nelegalių blokuotų kadrų kiekis veikiant EFRR algoritmui buvo vidutiniškai lygus 80,36%, o veikiant GAP/SNRA algoritmams - 36,61%. EFRR ir GAP/SNRA algoritmų blokuotų legalių kadrų kiekis iki 50s mažėjo eksponentiškai, o FRR algoritmo blokuotų legalių kadrų kiekis visu modeliuotu laiku buvo artimas 0 su vidutine 0,0025% reikšme.



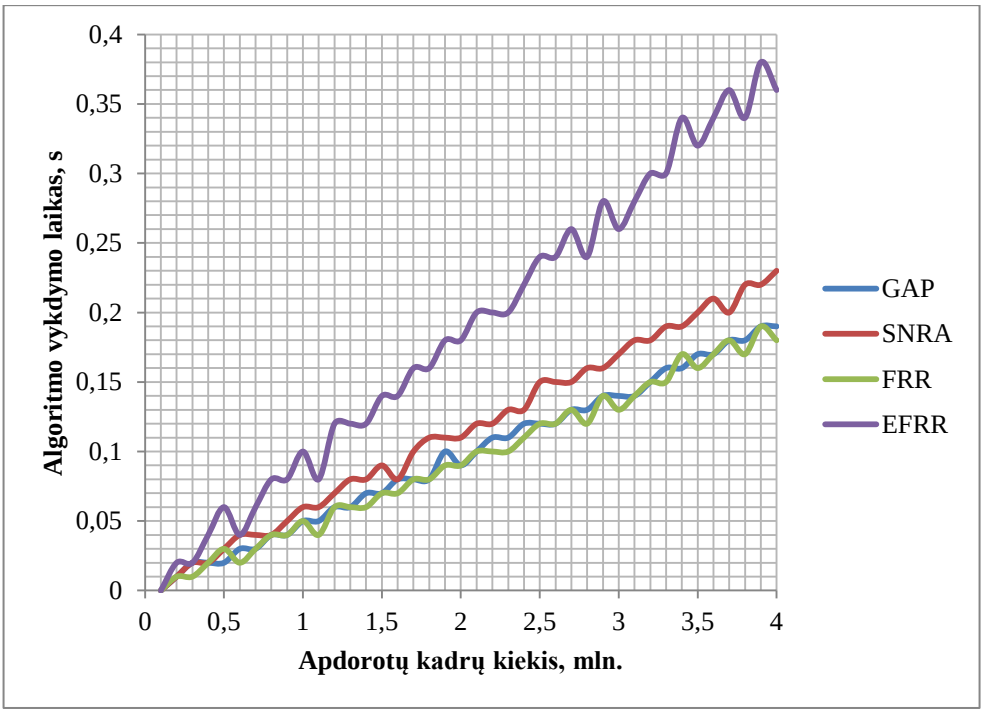
73 pav. *Autentifikacija. Nelegalių kadrų blokavimas nuo laiko*



74 pav. *Autentifikacija. Legalių kadrų blokavimas nuo laiko*

4.3.3. Vykdymo trukmė

Vykdymo trukmės nuo apdorojamų kadrų kiekio charakteristika įvertina algoritmų sudėtingumą panaudojamo procesoriaus laiko prasme. Charakteristikos grafikas pateiktas 75 pav.



75 pav. *Algoritmų vykdymo trukmė nuo apdorojamų kadrų kiekio*

Didėjant apdorojamų kadrų kiekiui, visų algoritmų vykdymo laikas didėja tiesiškai, kitaip sakant, visi algoritmai pasižymi *O(n)* sudėtingumu, tačiau skirtingas kreivių statumas reprezentuoja atskirų algoritmų vykdymo greičius: lėčiausiai vykdomas EFRR algoritmas, nes jį sudaro daugiausiai skaičiavimų. FRR ir GAP algoritmai vykdomi greičiausiai. Taikant optimizacijas, algoritmus galima paspartinti, tačiau tiesinio sudėtingumo pakeisti į efektyvesnį, tarkime, logaritminį, nepavyks, nes algoritmai turi analizuoti kiekvieną kadrą, ir negali eliminuoti dalį įvesties ties tam tikru vykdymo etapu.

4.4. Rezultatų apibendrinimas ir tyrimo išvados

Visi tyrimų rezultatai apibendrinti vidutinio blokuotų legalių kadrų kiekio ir vidutinio blokuotų nelegalių kadrų kiekio atžvilgiu bei pateikti lentelėse: deautentifikacijos scenarijaus rezultatus apibendrina lentelė Nr. 20, o autentifikacijos tvindymo – lentelė Nr. 21.

Lentelė Nr. 20 *Deautentifikacijos scenarijaus tyrimo rezultatų apibendrinimas*

	Vidutinis blokuotų nelegalių kadrų kiekis / Vidutinis blokuotų legalių kadrų kiekis, %				
	Nuo neleg. kadrų kiekio (0-100%)	Nuo plokščių sk. (1-20)	Nuo lango ilgio (1-2000)	Nuo praradimų (0-10%)	Nuo laiko (0-300s)
GAP	36,66 / 26,22	4,84 / 5,93	30,1 / 24,8	28,8 / 27,55	29,35 / 25,02
SNRA	36,66 / 26,22	4,84 / 5,93	30,1 / 24,8	29,4 / 26, 01	29,35 / 25,02
FRR	18,79 / 16,54	3,87 / 4,14	0,35 / 0,31	16,42 / 17,54	13,42 / 19,95
EFRR	19,15 / 0,008	6,76 / 0	31,05 / 2,52	23,6 / 0,029	22,76 / 0,252

Lentelė Nr. 21 *Autentifikacijos tvindymo scenarijaus tyrimo rezultatų apibendrinimas*

	Vidutinis blokuotų nelegalių kadrų kiekis / Vidutinis blokuotų legalių kadrų kiekis, %				
	Nuo neleg. kadrų kiekio (0-100%)	Nuo plokščių sk. (1-20)	Nuo lango ilgio (1-2000)	Nuo praradimų (0-10%)	Nuo laiko (0-300s)
GAP	33,53 / 0,017	36,57 / 0,045	36,5 / 0,02	36,82 / 1,75	36,62 / 0,159
SNRA	33,59 / 0,017	36,57 / 0,045	36,5 / 0,02	36,82 / 0,028	36,62 / 0,159
FRR	0 / 0,012	0 / 0,002	0,0006 / 0,016	0,00004 / 1,38	0 / 0,003
EFRR	70,1 /0,035	79,29 / 0,03	96,78 / 2,523	79,24 / 0,06	80,36 / 0,425

Remiantis tiriamosios darbo dalies rezultatais, galima formuluoti šias tyrimo išvadas:

1. Sudarytasis tyrimo modelis nėra realaus laiko ir tinklo kadrų srautus modeliuoja C kalbos struktūrų masyvais, kuriuose esanti kiekviena duomenų struktūra reprezentuoja modeliuojamą kadrą. Visa vidinė modelio veikla realizacijos prasme yra paremta šių masyvų perdavimo ir apdorojimu C funkcijose, kurių veiklą valdo per komandinę eilutę perduoti parametrai. Sudarytasis EFRR algoritmas buvo realizuotas kaip viena iš modelio funkcijų;
2. GAP ir SNRA algoritmų teisingo blokavimo kiekiai buvo gauti panašūs į analitinėje dalyje autorių gautus kiekius: tyrimo metu gautas vidutinis GAP ir SNRA blokuotų nelegalių kadrų kiekis buvo 35%, kai tuo tarpu Bansal R., Bansal D. ir Tiwari S. straipsnyje "Non-Cryptographic Methods of MAC Spoof Detection in Wireless LAN" pateiktuose rezultatuose gauta 39,12% esant GAP ir 35,02% esant SNRA algoritmams. Šis požymis pagrindžia modelio veiksnumą.
3. Apibendrinus visus scenarijus ir visas charakteristikas, sudarytasis EFRR algoritmas teikia iki 56 procentų mažesnį blokuotų legalių kadrų kiekį (neteisingą blokavimą) ir iki 63 procentų didesnį blokuotų nelegalių kadrų kiekį nei kiti tirtieji algoritmai: GAP, SNRA ir FRR.

5. GALUTINĖS DARBO IŠVADOS

1. IEEE 802.11 standartų šeimos tinklai pasižymi atsisakymo aptarnauti atakų (toliau DoS) pažeidžiamumais visuose OSI modelio lygiuose: fizinio ir kanalinio lygio pažeidžiamumai yra sąlygoti paties standarto saugos trūkumų ir netobulos standarto realizacijos – šiuos pažeidžiamumus galima pašalinti modifikuojant standartą arba panaudojant aukštesnius tinklo lygmenis atakų prevencijai; aukštesnių lygmenų pažeidžiamumai yra sąlygoti TCP/IP platformos ir taikomųjų protokolų saugos trūkumų, bei yra būdingi visų standartų tinklams naudojantiems šiuos protokolus , taip pat ir belaidžiams;
2. Fizinio lygmens DoS atakos, pagrįstos tinklo signalų iškraipymu, sukuria didžiausią žalą bei yra sunkiausiai išvengiamos, jų neutralizavimui taikomi fizinio atakos įrenginio pašalinimo, dažninio ir erdvinio išvengimo metodai. Kanalinio lygmens DoS atakos yra pagrįstos kadrų klastojimu, todėl šias atakas galima malšinti valdymo ir kontrolės kadrų autentifikacijos mechanizmais: kriptografiniu kadrų autentifikavimu, signalo charakteristikomis paremtu tinklo plokščių autentifikavimu ir srauto tikrinimu, paremtu kadrų sekos numerio analize;
3. Sudarytasis EFRR apsaugos nuo DoS atakų algoritmas yra pagrįstas kadrų klastojimo identifikacija pagal sekos numerio kitimo charakteristikas. Algoritmo blokavimo teisingumas yra tikimybinis, nes nelegalių kadrų teisingas arba klaidingas identifikavimas priklauso nuo srauto, kuris yra atsitiktinis;
4. EFRR algoritmas, kitų tirtųjų algoritmų atžvilgiu, pasižymėjo šiomis ypatybėmis:
   a) Priklausomai nuo nelegalių kadrų kiekio sraute, iki 56,7 procentų mažesniu blokuotų legalių kadrų kiekiu ir iki 70,3 procentų mažesniu blokuotų nelegalių kadrų kiekiu esant deautentifikacijos scenarijui, bei atitinkamai iki 0,08 procentų didesniu legalių ir 99,8 procentų didesniu nelegalių blokuotų kadrų kiekiais esant autentifikacijos scenarijui;
   b) Priklausomai nuo tinklo plokščių kiekio, iki 24,8 procentų mažesniu blokuotų legalių kadrų kiekiu ir iki 5,7 procentų mažesniu blokuotų nelegalių kadrų kiekiu esant deautentifikacijos scenarijui, bei atitinkamai iki 0,03 procentų mažesniu legalių ir 79,7 procentų didesniu nelegalių blokuotų kadrų kiekiais esant autentifikacijos scenarijui;
   c) Priklausomai nuo prarastų kadrų kiekio, iki 29,4 procentų mažesniu blokuotų legalių kadrų kiekiu ir iki 5,3 procentų mažesniu blokuotų nelegalių kadrų kiekiu esant deautentifikacijos scenarijui, bei atitinkamai iki 2,7 procentų mažesniu legalių ir 80,6 procentų didesniu nelegalių blokuotų kadrų kiekiais esant autentifikacijos scenarijui;

d) Priklausomai nuo kadrų eilės ilgio, iki 20,8 procentų mažesniu blokuotų legalių kadrų kiekiu ir iki 37,3 procentų didesniu blokuotų nelegalių kadrų kiekiu esant deautentifikacijos scenarijui bei atitinkamai iki 4 procentų didesniu legalių ir 99,8 procentų didesniu nelegalių blokuotų kadrų kiekiais esant autentifikacijos scenarijui;

Remiantis minėtomis ypatybėmis, galime konstatuoti, kad sudarytasis EFRR algoritmas yra pranašesnis autentifikacijos tvindymo ir analogiškų atakų malšinime.

5. EFRR algoritmas ir kiti tirtieji algoritmai pasižymi *O(n)* sudėtingumu - didinant apdorojamų kadrų kiekį, algoritmo vykdymo laikas didėja tiesiškai. Taikant optimizacijas, algoritmų vykdymo greitį galima padidinti (sumažinti tiesės statumą), tačiau pakeisti į efektyvesnią funkciją nepavyks, nes algoritmai privalo analizuoti kiekvieną kadrą ir negali eliminuoti dalies įvesties ties tam tikru vykdymo etapu.

LITERATŪRA

[1] **Anjum F., Das S., Gopalakrishnan P., Kant L., Kim B.** Security in an Insecure WLAN Network// Proceedings of the International Conference on Wireless Networks, Communications and Mobile Computing. ISBN 0-7803-9305-8. Wuhan, 2005, p. 292-297.

[2] **Anthony D., Stankovic J.** A Taxonomy for Denial-of-Service Attacks in Wireless Sensor Networks// EC2ND. - 2005, Nr. 2, p. 97-105.

[3] **Arbaugh W. A., Shankar N., Wan J.** Your 802.11 Wireless Network has no Clothes// Wireless Communications. ISSN 1536-1284. 2002, Nr. 6, p. 44-51.

[4] **Aslam B., Islam M. H., Khan S. A.** Pseudo Randomize Sequence Number Based Solution to 802.11 Disassociation Denial of Service Attack// Proceedings of the First Mobile Computing and Wireless Communication International Conference. ISBN 978-9957-486-00-6. Amman, 2006, p. 215-220.

[5] **Bansal R., Bansal D., Tiwari S.** Non-Cryptographic Methods of MAC Spoof Detection in Wireless LAN// Proceedings of the 16th IEEE International Conference on Networks. ISBN 978-1-4244-3805-1. New Delhi, 2008, p. 1-6.

[6] **Bellardo J., Savage S.** 802.11 Denial of Service Attacks: Real Vulnerabilities and Practical Solutions// Proceedings of the 12th conference on USENIX Security Symposium. Washington, 2003, p. 2.

[7] **Bernaschi M., Ferreri F., Valcamonici L.** Access points vulnerabilities to DoS attacks in 802.11 networks// Wireless Networks. ISSN 1022-0038. 2008, Nr. 2, p. 159-169.

[8] **Bicakci K., Tavli B.** Denial of Service attacks and countermeasures in IEEE 802.11 wireless networks// Computer Standarts & Interfaces. ISSN 0920-5489. 2009, Nr. 5, p. 931-941.

[9] **Cacheand J., Liu V.** Hacking Exposed Wireless: Wireless Security Secrets & Solutions. ISBN 978-0-072-26258-2. Osborne: McGraw, 2007. 225-234 p.

[10] **Chiueh T., Guo F.** Sequence Number-Based MAC Address Spoof Detection// Proceedings of 8th International Symposium on Recent Advances in Intrusion Detection. ISBN 3-540-31778-3. Seattle, 2005, p. 309-329.

[11] **Faria D. B., Cheriton D. R.** Detecting Identity-Based Attacks in Wireless Networks Using Signalprins// Proceedings of the 5th ACM workshop on Wireless security. ISBN 1-59593-557-6. Los Angeles, 2006, p. 43-52.

[12] **Geng X., Huang Y., Whinston A. B.** Defending Wireless Infrastructure Against the Challenge of DDoS Attacks// Mobile Networks and Applications. ISSN 1383-4694. 2002, Nr. 3, p. 213-223.

[13] **Goel S., Kumar S.** An Improved Method of Detecting Spoofed Attack in Wireless LAN// Proceedings of the First International Conference on Networks and Communications. ISBN 978-1-4244-5364-1. Chennai, 2009, p. 104-108.

[14] **Hsieh W. C., Lo C. C., Lee J. C., Huang L. T.** The Implementation of a Proactive Wireless Intrusion Detection System// Proceedings of the 4th international conference on Computer and Information Technology. ISBN 0-7695-2216-5. Washington, 2004, p. 581-586.

[15] **Institute of Electrical and Electronics Engineers.** IEEE Standart 802.11-2007// IEEE Computer Society. ISBN 0-7381-5655-8. New York, 2007, p. 23-32.

[16] **Institute of Electrical and Electronics Engineers.** IEEE Standart 802.11w-2009// IEEE Computer Society. ISBN 978-0-7381-6049-8. New York, 2009, p. 8-19.

[17] **Kamal S, Issac B.** Analysis of Network Communication Attacks// Proceedings of the 5th Student Conference on Research and Development. ISBN 978-1-4244-1469-7. Selangor, 2007, p. 11-12.

[18] **Khan S., Loo K. K., Naeem T., Khan M. A.** Denial of Service Attacks and Challenges in Broadband Wireless Networks// International Journal of Computer Science and Network Security. ISSN 1738-7906. 2008, Nr. 7, p. 1-6.

[19] **Lau F., Rubin S. H., Smith M. H, Trajkovic L.** Distributed Denial of Service Attacks// Proceedings of the IEEE conference on Systems, Man, and Cybernetics. ISSN 1062-9222. Nashville, 2000, p. 2275-2280.

[20] **Malekzadeh M., Azim A., Ghani A, Desa J., Subramaniam S.** An Experimental Evaluation of DoS Attack and Its Impact on Throughput of IEEE 802.11 Wireless Networks// International Journal of Computer Science and Network Security. ISSN 1738-7906. 2008, Nr. 8, p. 1-5.

[21] **Mohammed L. A, Issac B.** DoS Attacks and Defense Mechanisms in Wireless Networks// Proceedings of the 2nd International Conference on Mobile Technology, Applications and Systems. ISBN 981-05-4573-8. Guangzhou, 2005, p. 8.

[22] **Nummenmaa J.** On constructive research in computer science. Viešos prezentacijos medžiaga. [žiūrėta 2011-02-11]. Prieiga per internetą: www.cs.uta.fi/~TKOPS407/sd-seminar-19-9-2007.pdf

[23] **Pahwa P., Tiwari G., Chabra R.** Spoofing Media Access Control (MAC) and its Counter Measures// An International Journal of Advanced Engineering & Applications. ISSN 0975-7783. 2010, Nr. 1, p. 190-197.

[24] **Sarafinienė N., Plėštys R., Rimkus D., Kavaliūnas R., Lagzdinytė I.** Kompiuterinių tinklų sauga. Kaunas: Technologija, 2008, 85-94 p.

[25] **Trappe W., Chandrasekaran G., Francisco J. A., Ganapathy V.** Detecting Identity Spoofs in IEEE 802.11e Wireless Networks// Proceedings of the IEEE Global Telecommunications Conference. ISBN 978-1-4244-4148-8. Honolulu, 2005, p. 97-105.

[26] **Trappe W., Li Q.** Detecting Spoofing and Anomalous Traffic in Wireless Networks via Forge-Resistant Relationships// Information Forensics and Security. ISSN 1556-6013. 2007, Nr. 6, p. 793-808.

[27] **Ureten O., Serinken N.** Wireless security through RF fingerprinting// Canadian Journal of Electrical and Computer Engineering. ISSN 0840-8688. 2007, Nr. 1, p. 27-33.

[28] **Xiao Y., Shen X., Du D. Z.** Wireless Network Security. ISBN-13 978-0-387-28040-0. New York: Springer, 2007. 159-172 p.

[29] **Xu W., Ma K., Trappe W., Zhang Y.** Jamming Sensor Networks: Attack and Defense Strategies// IEEE Network. ISSN 0890-8044. 2006, Nr. 3, p. 41-47.

[30] **Xu W., Trappe W., Zhang Y.** Anti-jamming Timing Channels for Wireless Networks// Proceedings of the first ACM Conference on Wireless Network Security. ISBN 978-1-59593-814-5. New York, 2008, p. 203-213.

[31] **Xu W., Wood T., Trappe W., Zhang Y.** Channel Surfing and Spatial Retreats: Defenses against Wireless Denial of Service// Proceedings of the 3rd ACM workshop on Wireless security. ISBN 1-58113-925-X. Philadelphia, 2004, p. 80-89.

[32] **Zhang Y., Zheng J., Ma M.** Handbook of Research on Wireless Security. ISBN 978-1-59904-899-4. New York: Information Science Reference, 2008. 78-95 p.

DEVELOPMENT AND RESEARCH OF 802.11 WIRELESS NETWORKS FRAME SEQUENCE NUMBER ANALYSIS BASED SECURITY ALGORITHM

SUMMARY

Insufficient IEEE 802.11 standard security mechanisms against denial of service (DoS) attacks has caused a lot of field research on wireless networks vulnerabilities, attacks and security methods. Our work follow this trend with the main object to propose a frame sequence number analysis based security algorithm against DoS attacks in 802.11 standard networks.

In the analytical part of this work we have described most common DoS mechanisms and countermeasures in 802.11 wireless networks and detailed on present security solutions based on frame sequence number analysis. We concluded that physical layer attacks based on wlan signal manipulations are most severe, but link layer attacks are also popular due less power consumption and simpler link layer frame forgery based attack mechanisms, such as fake deauthentication or massive probe request flood. Majority link layer DoS attacks methods could potentially be blocked by analysing frames sequence number and actively destroying frames with anomaliuos sequence characteristics.

In the research part of this work we have proposed a frame sequence number analysis based security algorithm against link layer DoS attacks and developed a model for simulating and evaluating DoS attacks and security algorithms. We have compared our proposed algorithm with present solutions by simulating model in terms of blocking ratio dependency on quantity of attack frames, number of network adapters, length of frame window and frame drop ratio. Results show that our proposed algorithm generates up to 56 percent lower false blocks and 63 percent higher true blocks than present similar solutions, but due possibility driven anomaly identification, both ratios is highly dependent on traffic characteristics.

SANTRUMPŲ IR TERMINŲ ŽODYNAS

ATIM	Ad-Hoc režimu veikiančio belaidžio tinklo kadro antraštės laukas, nurodantis: sugeneruotų kadrų su tinklo parametrų skelbimu (Beacon) kiekį ir kadrų kaupimo stotelės buferyje identifikaciją (angl. *Ad-Hoc Traffic Indication Map*)
CCA	Belaidžio tinklo neužimto eterio identifikacijos algoritmas (angl. *Clear Channel Assesment*)
CF	Belaidžio tinklo kontrolės kadras kadras su antraštėje deklaruojamu laisvos terpės prieigos režimu (angl. *Contention Free*)
CSMA/CA	Bendros terpės prieigos valdymo metodas belaidžiame tinkle (angl. *Carrier Sense Multiple Access with Colission Avoidence*)
CW	Laiko tarpas, kurį tinklo plokštė turi laukti, prieš mėgindama siųsti duomenis (angl. *Contention Window*)
DCF	Paskirstytosios architektūros belaidžio tinklo mazgų komunikacijos valdymo mechanizmas (angl. *Distributed Cooordination Function*)
EFRR	Kadrų sekos numerio analize paremtas apsaugos nuo atsisakymo aptarnauti atakų algoritmas – išplėstoji FRR algoritmo versija (angl. *Extended Forge Resilient Relationship*)
FEC	Perduodamų duomenų apsaugojimo nuo klaidų mechanizmas (angl. *Forward Error Correction*)
FRR	Kadrų sekos numerio analize paremtas apsaugos nuo atsisakymo aptarnauti atakų algoritmas, kurio identifikacija paremta didžiausio sekos numerio skirtumo tarp tam tikro ilgio kadrų eilės paieška (angl. *Extended Forge Resilient Relationship*)
GAP	Kadrų sekos numerio analize paremtas apsaugos nuo atsisakymo aptarnauti atakų algoritmas, kurio identifikacija paremta sekos numerių skirtumo įvertinimu
MIC	Kriptografiniais metodais suformuota kadro santrauka, kuri siunčiama kartu su kadru jo autentifikacijai (angl. *Message Integrity Code*)
NAV	Vieno tinklo komunikacijai išskirto eterio laiko trukmė (angl. *Network Allocation Vector*)
PLCP	Belaidžio tinklo fizinio lygmens kadro antraštė (angl. *Physical Layer Convergence Procedure*)
RUA	Fizinio lygmens atsisakymo aptarnauti ataka belaidžiame tinkle, kuomet naudojant didelį enegijos kieki pastoviai generuojamas triukšmas tinklo naudojamame spektre (angl. *Resource Unlimited Attack*)
SNRA	Kadrų sekos numerio analize paremtas apsaugos nuo atsisakymo aptarnauti atakų algoritmas, kurio identifikacija paremta sekos numerio kitimo spartos įvertinimu (angl. *Sequence Number Rate Analysis*)
TIM	Kadro antraštės laukas, saugantis kadrų kaupimo stotelės buferyje identifikacijos žymą (angl. *Traffic Identification Map*)

PRIEDAI

Priedai Nr. 2, Nr. 3 ir Nr. 4 pateikti prie šio darbo aiškinamojo rašto galinio viršelio prėdėtame CD.

1. EFRR algoritmo C išeities kodas

```
int dropEFRR(struct ieee80211_frame_parsed *fp, int size, int dif, double speed, int win, int maxdur)
      {
          int window = 0, i, j, k, seqh, seql, d, step = -1;

          double timel, timeh, s, param[size][7], id = 0.0, maxDR = 0.0, maxATK = 0.0, maxIR = 0.0,
          sumdr = 0.0, sumatk = 0.0, sumir = 0.0, IDdr=0.0, IDatk=0.0, IDir=0.0;

          for (i = 0; i < size; i++)
          {
              window += ((fp+(i+1))->timestamp - (fp+i)->timestamp);

              if (((win-window) < maxdur) || (i == size-1))
              {
                  param[0][0] = 0;
                  param[0][1] = 0;
                  param[0][2] = 0;
                  param[0][3] = 0;

                  if (((fp+(step+1))->fc[0] == 76) || ((fp+(step+1))->fc[0] == 74) )
                  {
                      param[0][4]=1;
                  }
                  else
                  {
                      param[0][4]=0;
                  }

                  if (((fp+(step+1))->fc[0] == 68) || ((fp+(step+1))->fc[0] == 75 ) ||
                  ((fp+(step+1))->fc[0] == 64 ))
                  {
                      param[0][5]=1;
                  }
                  else
                  {
                      param[0][5]=0;
                  }

                  param[0][6] = 0;

                  for (j = 1; j < (i - step); j++)
                  {
                      seql = seqatov((fp+(step + j))->seq);
                      seqh = seqatov((fp+(step + 1 + j))->seq);
                      timel = (double)(fp+(step + j))->timestamp;
                      timeh = (double)(fp+(step + 1 + j))->timestamp;
                      d = (seqh - seql) % 4095;
                      s = ((double)abs(d))/(timeh-timel);

                      for (k = (j-1); k >=0; k--)
                      {
                          if ((seqatov((fp+(step + 1 + k))->seq)+1) == seqh)
                          {
                              param[j][0] = param[k][0];
                              break;
                          }
                          else if (k == 0)
                          {
                              id = id + 1.0;
                              param[j][0] = id;
                          }
                          else
                          {
                              continue;
```

```
            }
        }

        param[j][1] = (double)d;
        param[j][2] = s;

        if ((maccmp((fp+(step + 1 + j))->addr2, (fp+(step + j))->addr2) == 0))
        {
            param[j][3] = 1.0;
        }
        else
        {
            param[j][3] = 0.0;
        }

        if (((fp+(step+1+j))->fc[0] == 76) || ((fp+(step+1+j))->fc[0] == 74) )
        {
            param[j][4]=1.0;
        }
        else
        {
            param[j][4]=0.0;
        }

        if (((fp+(step+1+j))->fc[0] == 68) || ((fp+(step+1+j))->fc[0] == 75 ) ||
        ((fp+(step+1+j))->fc[0] == 64 ))
        {
            param[j][5]=1.0;
        }
        else
        {
            param[j][5]=0.0;
        }

        if ((param[j][3] == 1.0) && ((((d == 0) || (abs(d) > dif)) && (s > speed)) ||
        (d == -1)) && (param[j-1][6] != 1.0))
        {
            param[j][6] = 1.0;
        }
        else if ((param[j][3] == 0.0) && (d == 1))
        {
            param[j][6] = 1.0;
        }
        else if ((param[j][3] == 1.0) && (param[j][0] != param[j-1][0]))
        {
            param[j][6] = 1.0;
        }
        else if ((param[j][3] == 0.0) && (param[j][0] == param[j-1][0]))
        {
            param[j][6] = 1.0;
        }
        else
        {
            param[j][6] = 0.0;
        }
    }

    for (j=0; j<((int)id+1); j++)
    {
        sumdr = 0.0;

        for (k=0; k<(i-step); k++)
        {
            if ((param[k][0] == (double)j) && (param[k][4] == 1.0))
            {
                sumdr+=1.0;
            }
        }

        if ((maxDR == sumdr) && (maxDR != 0.0))
        {
            IDdr = -1.0;
```

```
    }
    else if (maxDR<sumdr)
    {
        maxDR = sumdr;
        IDdr = (double)j;
    }
    else
    {
        continue;
    }
}

if (maxDR == 0.0)
{
    IDdr = -1.0;
}

for (j=0; j<((int)id+1); j++)
{
    sumatk = 0.0;

    for (k=0; k<(i-step); k++)
    {
        if ((param[k][0] == (double)j) && (param[k][6] == 1.0))
        {
            sumatk+=1.0;
        }
    }
    if ((maxATK == sumatk) && (maxATK != 0.0))
    {
        IDatk = -1.0;
    }
    else if (maxATK<sumatk)
    {
        maxATK = sumatk;
        IDatk = (double)j;
    }
    else
    {
        continue;
    }
}

if (maxATK == 0.0)
{
    IDatk = -1.0;
}

for (j=0; j<((int)id+1); j++)
{
    sumir = 0;

    for (k=0; k<(i-step); k++)
    {
        if ((param[k][0] == (double)j) && (param[k][5] == 1.0))
        {
            sumir+=1.0;
        }
    }
    if ((maxIR == sumir) && (maxIR != 0.0))
    {
        IDir = -1.0;
    }
    else if (maxIR<sumir)
    {
        maxIR = sumir;
        IDir = (double)j;
    }
    else
    {
        continue;
    }
```

```
            }
            if (maxIR == 0.0)
            {
                IDir = -1.0;
            }
            if (((IDatk == IDdr) && (IDdr != -1)) || ((IDatk == IDir)&&(IDir !=-1)))
            {
                for (j=0; j<(i-step); j++)
                {
                    if (param[j][0] == IDatk)
                    {
                        destroy(fp, (step+1+j));
                    }
                }
            }
            maxATK = 0.0;
            maxDR = 0.0;
            maxIR = 0.0;
            id = 0.0;
            window = 0;
            step = i;
        }
        else
        {
            continue;
        }
    }
   return 0;
}
```

2. Modelio C išeities kodas ir vykdomoji byla

Vykdomoji byla sukompiliuota *x86* architektūros kompiuteriui, *LINUX* operacinei sistemai.

3. Tyrimo rezultatų duomenys

Tyrimo rezultatai pateikti *Microsoft Excel 2007* bylų rinkinyje.

4. Darbo tema konferencijoje pristatytas pranešimas