Kauno technologijos universitetas

Informatikos fakultetas

Belaidžio 802.11 standarto tinklo deautentifikavimo atakų aptikimo tyrimas panaudojant mašininio mokymo algoritmą

Baigiamasis magistro studijų projektas

Saulius Juškevičius

Projekto autorius

Lekt. dr. Dangis Rimkus

Vadovas

Kaunas, 2021

Kauno technologijos universitetas

Informatikos fakultetas

Belaidžio 802.11 standarto tinklo deautentifikavimo atakų aptikimo tyrimas panaudojant mašininio mokymo algoritmą

Baigiamasis magistro studijų projektas

Informacijos ir informacinių technologijų sauga (6211BX008)

Saulius Juškevičius

Projekto autorius

Lekt. dr. Dangis Rimkus

Vadovas

Doc. Gedeiminas Činčikas

Recenzentas

Kaunas, 2021

Kauno technologijos universitetas

Informatikos fakultetas

Saulius Juškevičius

Belaidžio 802.11 standarto tinklo deautentifikavimo atakų aptikimo tyrimas panaudojant mašininio mokymo algoritmą

Akademinio sąžiningumo deklaracija

Patvirtinu, kad:

1. baigiamąjį projektą parengiau savarankiškai ir sąžiningai, nepažeisdama(s) kitų asmenų autoriaus ar kitų teisių, laikydamasi(s) Lietuvos Respublikos autorių teisių ir gretutinių teisių įstatymo nuostatų, Kauno technologijos universiteto (toliau – Universitetas) intelektinės nuosavybės valdymo ir perdavimo nuostatų bei Universiteto akademinės etikos kodekse nustatytų etikos reikalavimų;

2. baigiamajame projekte visi pateikti duomenys ir tyrimų rezultatai yra teisingi ir gauti teisėtai, nei viena šio projekto dalis nėra plagijuota nuo jokių spausdintinių ar elektroninių šaltinių, visos baigiamojo projekto tekste pateiktos citatos ir nuorodos yra nurodytos literatūros sąraše;

3. įstatymų nenumatytų piniginių sumų už baigiamąjį projektą ar jo dalis niekam nesu mokėjęs (-usi);

4. suprantu, kad išaiškėjus nesąžiningumo ar kitų asmenų teisių pažeidimo faktui, man bus taikomos akademinės nuobaudos pagal Universitete galiojančią tvarką ir būsiu pašalinta(s) iš Universiteto, o baigiamasis projektas gali būti pateiktas Akademinės etikos ir procedūrų kontrolieriaus tarnybai nagrinėjant galimą akademinės etikos pažeidimą.

Saulius Juškevičius

Patvirtinta elektroniniu būdu

Juškevičius, S. „Belaidžio 802.11 standarto tinklo deautentifikavimo atakų aptikimo tyrimas panaudojant mašininio mokymo algoritmą“. Magistro baigiamasis projektas / vadovas lekt. dr. Dangis Rimkus; Kauno technologijos universitetas, informatikos fakultetas, kompiuterių katedra.

Studijų kryptis ir sritis (studijų krypčių grupė): technologijos mokslų studijų sritis

Reikšminiai žodžiai: Adam optimizatorius, tinklo klasifikavimas, gilusis mokymasis, IEEE 802.11 standartas

Kaunas, 2021. 53 p.

SANTRAUKA

Šias laikais belaidis tinklas yra populiari technologija. Dauguma nesusimąsto, kad belaidžio tinklo patogumas taip pat sukelia lengvesnę prieigą įsibrovėliams įvykdyti atakas. Galima nepastebėti, kai įsibrovėlis įvykdo ataką ir jau yra perėmęs tinklo srautą tarp naudotojo ir prieigos taško, taip perduodant privačią informaciją įsibrovėliui. Nuo tokių įsibrovimų gali apsaugoti bevielio tinklo įsibrovimų aptikimo sistemos. Tokios sistemos paremtos savybių atpažinimu arba anomalijų aptikimu.

Šiame darbe realizuotas prototipas, kuris aptinka įvykdytą belaidžio tinklo ataką. Naudojamas mašininio mokymo algoritmas ištirtas ir palygintas su kitais mašininio mokymo algoritmais realizuotais palyginimui su surinktu duomenų rinkiniu. Duomenų rinkinys surinktas panaudojant papildomas stebėjimo stoteles. Duomenų rinkinys normalizuotas ir išskirtos savybės mašininio mokymo algoritmui. Apmokytas algoritmas ištirtas ir palygintas.

Juškevičius, Saulius. *Wireless 802.11 Standard Network Deauthentication Attack Detection Research Using Machine Learning Algorithm*: *Master's* thesis in information and information system security / supervisor lect. dr. Dangis Rimkus. The Faculty of Informatics, Kaunas University of Technology.

Study field and area (study field group): study of technological sciences

Key words: Adam optimizer, network classification, deep learning, IEEE 802.11 standard

Kaunas, 2021. 53 p.

SUMMARY

These days, wireless network is a popular technology. Most do not realize that the convenience of a wireless network also leads to easier access for intruders to carry out attacks. It can go unnoticed when an intruder executes an attack and has already intercepted network traffic between the user and the access point, thus transmitting private information to the intruder. Wireless intrusion detection systems can protect against such intrusions. Such systems are based on feature recognition or anomaly detection.

In this work, a prototype that detects a wireless attack is realized. The used machine learning algorithm was researched and compared with other machine learning algorithms implemented for comparison with the collected data set. The data set was collected using additional monitoring stations. The data set is normalized and the properties of the machine learning algorithm are isolated. Algorithm trained, researched and compared.

TURINYS

LENTELIŲ SĄRAŠAS

PAVEIKSLŲ SĄRAŠAS

TERMINŲ IR SANTRUMPŲ ŽODYNAS

Man in the middle (*mitm*)– tinklo atakos tipas, kai tinklo srautas keliauja per įsibrovėlį.

ESSID (angl. *extended service set identifier*) – išplėstinis paslaugų rinkinio identifikatorius.

BSSID (angl. *basic service set identifier*) – pagrindinio paslaugų rinkinio identifikatorius.

DoS (angl. *Denial of Service*) – paslaugų atsisakymo ataka.

WIDS (angl. *wireless intrusion detection system*) – belaidžio tinklo įsibrovimų aptikimo sistema.

U2R (angl. *user to root*) – ataka, kai gaunama prieiga prie vartotojo su aukščiausiomis teisėmis.

R2L (angl. *remote to user*) – ataka, kai po prisijungimo prie kompiuterio gaunamas vartotojo valdymas.

PSO (angl. *particle swarm optimization*) – dalelių spiečiaus optimizavimas.

BPSO (angl. *binary particle swarm optimization*) – dvejetainių dalelių spiečiaus optimizavimas.

pdAPSO (angl. *primal-dual particle swarm optimisation*) – pirminio dvigubo dalelių spiečiaus optimizavimas.

KNN (angl. *k-nearest neighbour*) – k-artimiausias kaimynas.

SVM (angl. *support vector machine*) – pagalbinio vektoriaus mašina.

LSTM (angl. *long short-term memory*) – ilgoji turmpalaikė atmintis.

RNN (angl. *recurrent neural network*) – rekurentinis neuroninis tinklas.

ĮVADAS

Informacijos ir informacinių sistemų saugos studento Sauliaus Juškevičiaus darbas

Darbo problematika ir aktualumas

Belaidis vietinis tinklas (angl. *Wireless local area network*) – populiari technologija, ypatingai IEEE 802.11 standartas. Ši technologija naudojama įvairiose vietose, pvz. namuose, kavinėse, didžiulėse įmonėse ir kitose viešose vietose. Dėl patogumo ir naudojimo paprastumo ši technologija yra labai išplitusi, todėl atsiranda saugumo ir privatumo užtikrinimo problemos. Dėl to kaip ši technologija veikia yra galimos kelios atakos.

Įsilaužėlis gali įdiegti prieigos tašką su tuo pačiu paslaugos rinkinio identifikatoriumi (angl. *service set identifier*) kaip ir tikrasis prieigos taškas ir gali, siųsdamas kadrus į vartotojų įrangą, priversti tuos įrenginius prisijungti prie neautorizuoto prieigos taško. Taip įsilaužėlis gauna prieigą prie vartotojo siunčiamos informacijos ir gali atlikti *man in the middle* atakas. Bandydami būti neaptikti tinkle ir nebūti sugauti, įsilaužėliai periodiškai generuoja vis kitą prieigos taško tinklo plokštės adresą (angl. *media access control address*). Tokias atakas yra paprasta surengti, ypač kai programinę įrangą galima atsisiųsti iš interneto, taip pat užsisakyti reikiamą aparatinę įrangą galima per internetą. Be to, internete yra pamokų, kaip įvykdyti tokias atakas.

Atakų aptikimui naudojamos belaidžio tinklo įsibrovimų aptikimų sistemos, kurios, naudodamos mašininio mokymo algoritmus ar kitokius metodus, aptinka atakas. Tačiau šios sistemos ir metodai pilnai neužtikrina saugumo. Vartotojai gali sumažinti tokių atakų riziką ir padarinius naudodami virtualius privačius tinklus. Kadangi belaidžio tinklo saugumas nėra užtikrintas nuo tokių atakų, todėl yra reikalingas metodas, kuris užtikrintų perimto srauto aptikimą.

Darbo tikslas ir uždaviniai

Darbo tikslas sukurti metodą, kuris aptinka įvykdytą belaidžio tinklo *man in the middle* ataką.

Darbo uždaviniai:

1) atlikti analizę ir ištirti esamas belaidžio tinklo įsibrovimo aptikimo sistemas ir metodus.;
2) suprojektuoti prototipą pagal sudarytą metodą, kuris aptinka įvykdytą belaidžio tinklo *man in the middle* ataką.;
3) realizuoti prototipą pagal sudarytą metodą.;
4) atlikti metodo tyrimą panaudojant prototipą ir įvertinti gautus rezultatus, palyginant su kitais metodais.

Darbo rezultatai ir jų svarba

Realizuotas ir ištirtas metodas, kuris turi privalumų, lyginant su kitais metodais.

Darbo struktūra

Dokumentas išskirtas į analizės, projektavimo, realizacijos, tyrimo dalį, bei išvadas.

1 BELAIDŽIO TINKLO ATAKŲ IR APSAUGOS SISTEMŲ ANALIZĖ

1.1 Analizės tikslas

Šio darbo tikslas yra išanalizuoti belaidžio *IEEE* 802.11 standarto tinklo saugumo problemas bei jų prevencijos būdus ir priemones. Visame darbe bus kalbama apie belaidį *IEEE* 802.11 standarto ryšį.

1.2 Tyrimo objektas, sritis ir problema

1.2.1 Belaidis *IEEE* 802.11 tinklas

Belaidžio *IEEE* 802.11 standarto tinklo [1] technologija naudojama komunikacijų sprendimams realizuoti. Ją naudojant belaidžio ryšio įrenginiai sujungiami į vietinį kompiuterių tinklą, panaudojant prieigos tašką, užtikrinantį prieigą prie interneto resursų. Interneto paslaugų tiekėjai belaidžio ryšio įrenginius naudoja tarptinkliniams ryšiams realizuoti, sujungiant kompiuterių tinklus, esančius atskiruose pastatuose ar gyvenvietėse. Didžiausias patikimo belaidžio ryšio atstumas tarp dviejų belaidžių įrenginių priklauso nuo pastarųjų techninių parametrų ir aplinkos sąlygų – jis gali skirtis nuo keliolikos metrų iki kelių šimtų kilometrų.

Belaidės tinklo technologijos užtikrinama maksimali duomenų perdavimo sparta priklauso nuo naudojamo belaidžio tinklo standarto. Maksimali vartotojo duomenų perdavimo sparta praktikoje dėl perteklinės ryšio ir duomenų perdavimo protokolų informacijos bei belaidį ryšį slopinančios aplinkos poveikio paprastai neviršija 50 % teorinės spartos.

Belaidės tinklo technologijos palaikymas įdiegtas visose populiariausiose operacinėse sistemose. Belaidžio ryšio įranga komplektuojama daugelyje šiuolaikinių nešiojamųjų, planšetinių bei delninių kompiuterių, kai kuriuose mobiliuosiuose telefonuose ir praktiškai visuose išmaniuosiuose telefonuose. Tai suteikia galimybę mobilioms ir stacionarioms darbo stotims prisijungti prie interneto, komfortiškai naudotis internetine balso telefonija. Belaidžio *IEEE* 802.11 standarto tinklo viešosios prieigos taškų tinklas yra sparčiai plečiamas. Interneto prieigos taškai įrengti daugelyje viešųjų vietų – degalinėse, viešbučiuose, restoranuose, oro uostuose, geležinkelio stotyse, parduotuvėse, klubuose, bibliotekose, pėsčiųjų alėjose. Kai kurie telefonai ryšiui su tinklo operatoriumi gali naudoti ne tik *GSM*, bet ir belaidį *IEEE* 802.11 standarto ryšį. Belaidis ryšys taip pat naudojamas duomenų mainams tarp įvairių buitinių elektroninių prietaisų.

Belaidės *IEEE* 802.11 standarto technologijos naudojimas turi privalumų lyginant su laidais sujungtais tinklais:

- Belaidis ryšys suteikia galimybę patogiai (per atstumą) prisijungti prie kompiuterių tinklo aprūpinto prieigos tašku.
- Belaidis ryšys suteikia galimybę vartotojui judėti kartu su kompiuteriu, neprarandant ryšio – belaidis įrenginys esant galimybei automatiškai prisijungs prie kito prieigos taško, jei pirmasis taps nebepasiekiamas.
- Belaidis ryšys leidžia greitai ir nebrangiai išplėsti kompiuterių tinklus, kai technologijos, kurių realizacijai reikalingi kabeliai ar šviesolaidžiai, yra ekonomiškai neefektyvios.

Ši technologija naudojama įvairiose vietose, pvz. namuose, didžiulėse įmonėse ir viešose vietose. Dėl tokio šios paplitusios technologijos prieigos patogumo atsiranda saugumo ir privatumo užtikrinimo problemos, kurias šiame darbe ir aptarsime.

1.3 Belaidžio tinklo *man in the middle* atakos

1.3.1 *Man in the middle* ataka

Man in the middle atakų išpuoliai padaro duomenų saugumą ir privatumą ypač sudėtinga užduotimi, nes atakos gali būti vykdomos iš nuotolinių kompiuterių su suklastotais adresais. Kadangi ryšio saugumas visų pirma buvo užtikrinamas kompiuterinių tinklų šifravimo, todėl saugumo problema taip pat apima aktyvių įsibrovėlių puolimą ir „žmogus viduryje ataka“ yra viena iš galimų atakų. *Man in the middle* ataka pasinaudoja autentifikavimo protokolų tarp bendraujančių šalių trūkumais. Kadangi autentifikavimą įprastai teikia trečiosios šalys, kurios išduoda sertifikatus, sertifikatų generavimo sistema tampa dar vienu galimu pažeidžiamumo šaltiniu.

Man in the middle ataka leidžia įsibrovėliui šnipinėti tinklą per *galines duris* (angl. *backdoor*). Ši intervencija taip pat yra naudojama įmonių šnipinėti savo darbuotojus. Pavyzdžiui, 2015 metų pradžioje, buvo pastebėta, kad *Lenovo* kompiuteriuose buvo jau iš anksto įdiegta reklaminė programa, *Superfish,* kuri įrašo reklaminius įskiepius naršyklėse, pavyzdžiui, *Google Chrome* ir *Internet Explorer*. *Superfish* įdiegia savarankiškai sugeneruotą šakninį sertifikatą į Windows sertifikatų saugyklą ir tada pakeičia visus saugiųjų sujungimų lygmens (*SSL)* sertifikatus, kuriuos pateikia *HTTPS* protokolo svetainės su savo sertifikatu. Tai gali leisti įsilaužėliams potencialiai pavogti jautrius duomenis, pvz., elektroninės bankininkystės prisijungimus arba šnipinėti vartotojų veiklą [2].

Kriptografiniai protokolai, sukurti užtikrinti ryšių saugą per kompiuterių tinklą, yra transportavimo lygmens saugos (*TLS*) dalis. Šie protokolai naudoja *X. 509* tipo sertifikatą, kuris yra *ITU-T* standartas, nurodantis standartinį formatą viešojo rakto sertifikatams, atšauktųjų sertifikatų sąrašams, atributo sertifikatams ir sertifikavimo kelio patvirtinimo algoritmams [3]. *X.509* sertifikatai naudojami autentifikuoti šalis ir derėtis dėl simetrinio rakto. Kaip minėta, sertifikatų institucijos yra silpna saugumo sistemos grandis. Elektroniniame pašte, nors serveriai reikalauja *SSL* šifravimo, turinys yra tvarkomas ir saugomas atviru tekstu serveriuose [2].

1.3.2 *Man in the middle* atakų pavyzdžiai

Paprastas *man in the middle* atakos pavyzdys yra, kai paštininkas perima laišką, tiesiog skaito jo turinį arba net pakeičia jo turinį. Panašiai galima vizualizuoti tinkle vykstančią *man in the middle* ataką viešoje vietoje kaip prekybos centre, kuris suteikia nemokamą belaidį ryšį, yra įdiegtas belaidis maršrutizatorius su įdiegta kenkėjiška programine įranga. Jei naudotojas tuo metu apsilanko banko tinklalapyje iš telefono ar nešiojamojo kompiuterio, jis gali prarasti banko prisijungimo duomenis. Ši ataka gali būti sukelta tokiais populiariausiais būdais:

1) *ARP* podėlio apnuodijimas (angl. *ARP cache poisoning*)
2) *DHCP* klastojimas (angl. *DHCP spoofing*)
3) Domenų vardų serverio klastojimas (angl. *DNS spoofing*)
4) Sesijos perėmimas (angl. *Session hijacking*)
5) *SSL* perėmimas (angl. *SSL hijacking*)
6) *Deautentifikavimo atakos*

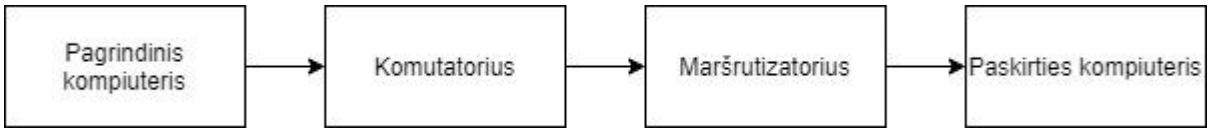
1.3.2.1 Adresų susiejimo protokolo podėlio apnuodijimas

Adresų susiejimo protokolo (*ARP*) veikimo metu kliento kompiuteris atsiųs paketą, kuriame bus šaltinio ir paskirties *IP* adresai pakete ir bus ištransliuoti į visus įrenginius, prijungtus prie to tinklo. Prietaisas, kuris turi paskirties *IP* adresą, išsiųs *ARP* atsakymą su savo tinklo plokštės adresu ir tada

tinkle įvyks bendravimas. *ARP* protokolas nėra apsaugotas protokolas ir *ARP* podėlis neturi patikimo mechanizmo, kas sąlygoja didelę problemą. Sekančiame 1.1 pav. galima matyti įprastą tinklo srautą tarp dviejų įrenginių.

ARP atsakymo paketas gali būti lengvai suklastotas ir gali būti siunčiamas į įrenginį, kuris išsiuntė *ARP* užklausą, nežinodamas, kad tai nėra tikrasis įrenginys, bet ataka gali sukelti duomenų pažeidimus. Taip atsitinka, nes *ARP* talpyklos lentelė bus atnaujinta taip, kaip nusprendė užpuolikas, ir todėl visas tinklo srautas eis per užpuoliką ir jis turės visus duomenis.

Įvairių tipų įrankiai yra prieinami *ARP* talpyklos apnuodijimui, pavydžiui, *Ettercap, Dsniff* ir *Cain and Abel*'s. Mes galime pabandyti kontroliuoti *ARP* talpyklos apnuodijimą naudodami dinamines *ARP* inspekcijas (*DAI*). *DAI* yra saugos funkcija, kuri naudojama patikrinti *ARP* paketus tinkle ir išmesti netinkamus *IP* ir tinklo plokštės adresų susiejimus. Šį patikrinimą reikia atlikti laidinio tinklo jungikliuose, rankiniu būdu juos sukonfigūravus, bet mes tinkle negalime to padaryti jungikliams, kurie nepalaiko šio patikrinimo. [2]



1.1 pav. Įprastas srautas tarp dviejų įrenginių

ARP užklausoms ir atsakymams nereikia autentifikavimo arba patvirtinimo, nes visi klientai tinkle pasitikės *ARP* atsakymais. Mes taip pat galime atnaujinti *ARP* talpyklos lentelę įterpdami statinius šliužo įrašus, kad užpuolikas negalėtų manipuliuoti šliužo įrašu lentele, bet tai nėra tobulas sprendimas, kadangi mes turime pakeisti *ARP* lentelėje esančius šliužo įrašus, jei mes pakeičiame įrangos poziciją. Saugaus lizdo sluoksnis (*SSL*) yra naudojamas *HTTP* protokole arba transportavimo sluoksnyje saugiam ryšiui. Žiniatinklio naršyklė ieškos žiniatinklio serverio sertifikatų ir autentifikuos jo galiojimą. Jei sertifikatas yra autentifikuotas mes turėsime saugų ryšį, bet jei mes turime kokių nors problemų su sertifikatu, tuomet sertifikatas bus laikomas nepatikimu. ARP podėlio apnuodijimo metu užpuolikas patektų į tinklą, kontroliuodamas tinklo komutatorių, kad galėtų stebėti tinklo srautą ir modifikuotų *ARP* paketus tarp bendraujančių kompiuterių, tokiu būdu įgyvendindamas *man in the middle* ataką [2].

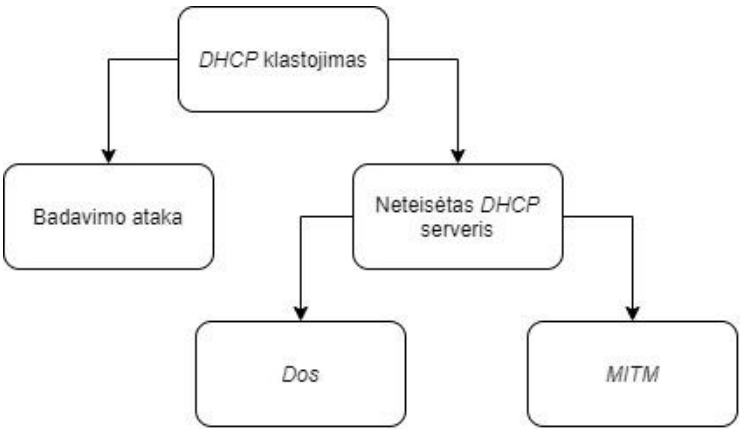
1.3.2.2 DHCP protokolo klastojimas

DHCP yra protokolas, teikia tinklo konfigūracijos parametrus naujai prijungtiems įrenginiams. Į parametrus įeina *IP* adresas, potinklio kaukė, numatytasis šliuzas, duomenų vardų serveris (DNS), serveris ir išnuomotas laikas. *DHCP* teikia kliento / serverio struktūrą, kuria *DHCP* paketais keičiasi tarp *DHCP* serverio ir pagrindinio kompiuterio, norint automatiškai priskirti aukščiau nurodytus parametrus.

DHCP yra ypač svarbus valdant tinklą. Tačiau *DHCP* kelia nemažai žinomų saugumo problemų, visų pirma:

- *DHCP* neturi pranešimų kilmės autentifikavimo. Viena vertus, *DHCP* klientai negali garantuoti, kad yra prisijungę prie patikimo *DHCP* serverio. Kita vertus, *DHCP* serveris negali užtikrinti ryšio su teisėtu klientu.
- Kiekvienas *DHCP* pranešimas perduodamas atviru tekste.

1.2 pav. parodytas *DHCP* klastojimo klasifikavimas.

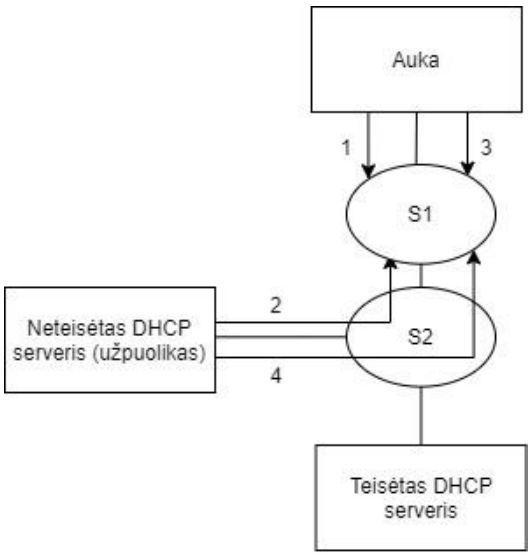


1.2 pav. *DHCP* klastojimo klasifikavimas

Per neautorizuotą *DHCP* serverį galima įvykdyti *DHCP* klastojimu pagrįstą *man in the middle* ataką. Užpuolikas bando atsakyti į *DHCP* užklausą greičiau nei vietinio tinklo teisėtas *DHCP* serveris. Panagrinėkime šį pavyzdį: tinklą sudaro *auka, teisėtas DHCP serveris, neautorizuotas DHCP serveris* ir du komutatoriai S1, S2. Kai *auka* prisijungia prie tinklo, įvyksta kitas ryšys (žr. 1.3 pav):

1) klientas transliuoja *DHCP* atradimą.
2) neautorizuotas serveris siunčia *DHCP* pasiūlymą (*Unicast*).
3) klientas perduoda *DHCP* užklausą.
4) neautorizuotas serveris siunčia DHCP ACK (*Unicast*).

Be to, užpuolikas gali vykdyti *DoS* ataką prieš teisėtą *DHCP* serverį, kad užtikrintų, jog aukos kompiuteris iš jo negaus atsakymo. Kita galimybė yra paleisti *DHCP* badavimo ataką (kai užpuolikas išnaudoja galiojančio *DHCP* serverio siūlomus *IP* adresus, kad naujiems pagrindiniams kompiuteriams nepavyktų jų gauti) [3].



1.3 pav. Grubus DHCP serverio išpuolis

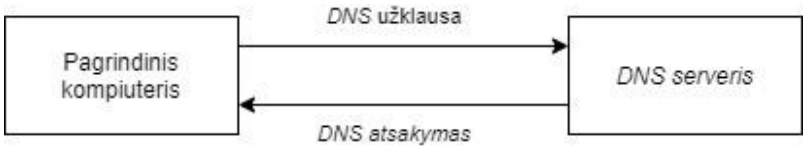
Šiuo metu užpuolikas gali pateikti tris neteisingas konfigūracijas, kur kiekviena sukels *man in the middle* ataką:

1) Neteisingas numatytas maršrutizatorius.

2) Neteisingas domenų vardų serveris.
3) Neteisingas *IP* adresas.

1.3.2.3 Domeno vardų serverio klastojimas

Šiuo atveju aukai bus pateikta netikra informacija, kuri leistų prarasti prisijungimo duomenis. Kaip buvo paaiškinta anksčiau, tai yra *man in the middle* ataka. Pavysdžiui, užpuolikas sukuria netikrą banko svetainę, todėl, jums jungiantis prie jūsų banko svetainės, jūs būsite nukreipti į užpuoliko sukurtą svetainę ir tokiu būdu užpuolikas gaus visus jūsų prisijungimo duomenis. Kai mes įeiname į svetainę savo kompiuteryje, domeno vardų serverio (*DNS)* užklausa siunčiama į domeno vardų serverio serverį ir mes gauname domeno vardų serverio atsakymo pranešimą. Tai parodyta 1.4 pav.



1.4 pav. Komunikavimas tarp kliento ir domeno vardų serverio

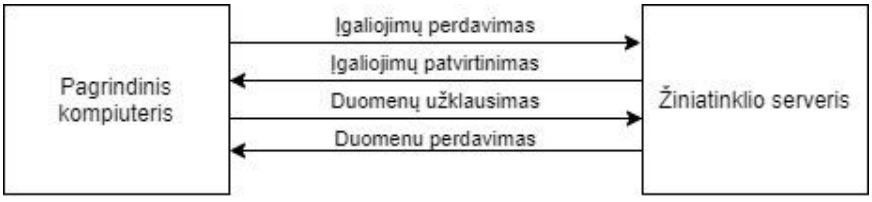
Ši domeno vardų serverio užklausa ir atsakymas susiejami su unikaliu identifikavimo numeriu. Kai užpuolikas gauna unikalų identifikacijos numerį, tada užmaskuoja auką su sugadintu paketu, kuriame yra identifikavimo numeris, tada ataka gali būti pradėta. Užpuolikas nukreipia auką į netikrą svetainę atliekant *ARP* podėlio apnuodijimą, kad nukreiptų domeno vardų serverio užklausos pranešimą, į kurį buvo išsiųstas suklastotas atsakymo paketas 1.5 pav.

1.5 pav. Užpuolikas atlieka *man in the middle* ataką panaudodamas domeno vardų serverio klastojimą

Pagrindinis kompiuteris nori prisijungti prie svetainės, todėl jis siųs domeno vardų serverio užklausos užklausą domeno vardų serverio serveriui, bet dėl *man in the middle* atakos, užpuolikas perims šią domeno vardų serverio užklausą ir išsiųs netikrą domeno vardų serverio atsakymą į pagrindinį kompiuterį. Priimantysis kompiuteris nežinotų, ar atsakymas yra teisėtas, ar ne, ir jis pradės bendrauti su kenkėjiška svetaine, užpuolikas sukels duomenų pažeidimus [2].

1.3.2.4 Sesijos perėmimas

Sesija yra sudaryta, kai mes turime ryšį tarp kliento ir serverio. Duomenų perdavimo valdymo protokolas (*TCP)* sudaro sesiją, nes jis pirmą kartą nustato ryšį, tada perduoda duomenis ir galiausiai nutraukia ryšį. Tai vadinama 3-jų krypčių pasisveikinimu. 1.6 pav. parodo, kaip atrodo tinkama įprasta sesija, sudaryta tarp pagrindinio kompiuterio ir žiniatinklio serverio. Vienas iš populiariausių sesijos perėmimų būdų yra sausainėlių (angl. *cookies*) vogimas su *HTTP* protokolo pagalba.



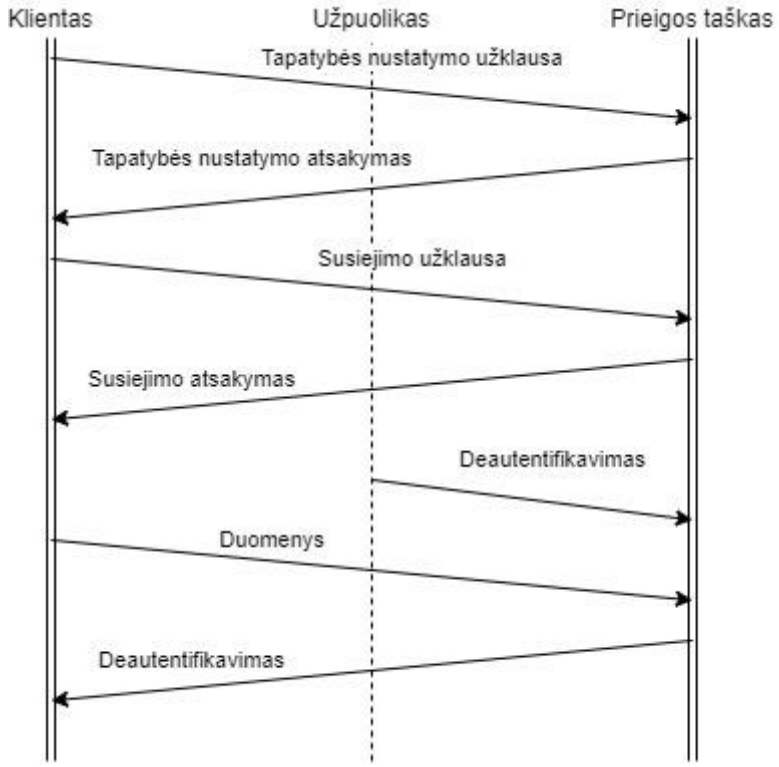
1.6 pav. Sesijos sudarymas tarp kliento kompiuterio ir žiniatinklio serverio

Bet kurioje svetainėje autentifikacijai ir sesijos sudarymui yra reikalingi vartotojo vardas ir slaptažodis. Kai sesija yra sudaryta, nebent jei vienas atsijungia iš sesijos, ji nėra nutraukiama, todėl administruoti sesijos sausainėliai yra naudojami, kad pateiktų informaciją, kad sesija vis dar tęsiasi. Jei užpuolikas gauna šį sausainėlį, tada jis gali turėti sesijos informaciją, kuri gali būti slapta [2].

1.3.2.5 Deautentifikavimo atakos

Deautentifikavimo ataka yra labai paprasta antrojo sluoksnio ataka, kuri naudojasi tuo, kaip 802.11 protokolas tvarko valdymo kadrus, kurių vienas tipas yra deautentifikavimo kadras. Iki 802.11w protokolo išleidimo 2009 m. valdymo kadrai niekada nebuvo užšifruoti ir jokiu būdu neapsaugoti nuo atkūrimo ar apsimetinėjimo, o tai reiškia, kad bet kuri šalis galėjo apgauti ir išsiųsti šiuos kadrus. Šis šifravimo trūkumas reiškia, kad vienintelis galutinio kompiuterio autentifikavimo būdas yra siuntėjo tinklo plokštės adresas, kur užpuolikas gali lengvai suklastoti adresą ir jo siunčiamus kadrus. Nors naujai išleistas 802.11w protokolas buvo skirtas ištaisyti daugeliui autentifikavimo pažeidžiamumų, tačiau sukelia naujų problemų, leidžiančių pasinaudoti kitomis paslaugų atsisakymo atakomis. Kitas susirūpinimą keliantis klausimas yra tai, kad 802.11w naudoja porinį WPA protokolo šifravimo raktą, o tai reiškia, kad tinklai be šifravimo, esantys oro uoste ar kavinėje, vis dar yra labai pažeidžiami atakų. Kai prieigos taškas – ar klientas nori atsiriboti nuo kito, tuomet išsiunčiamas deautentifikavimo kadras, kad klientas nori išeiti iš tinklo. Deja, norint nustatyti, kas yra tikrasis siuntėjas, autentifikavimo ar patvirtinimo nereikia, nes atpažinimo kadras yra identiškas bet kuriam kitam, išskyrus valdymo potipio kodo lauko reikšmę *0x2c* [4].

Kai kita šalis gauna kadrą su šiuo bitų rinkiniu, jie atsako su patvirtinimo atsakymu, patvirtindami pradinę užklausą, o tada ryšys nutrūksta. Dėl to užpuolikui labai lengva suklastoti užklausas, kad kiti vartotojai galėtų nutraukti belaidį tinklą. Vartotojui nėra galimybės palaikyti ryšį su tinklo dalimi, kai gaunama užklausa iš jų tinklo plokštės adreso, net jei jis ir toliau siunčia duomenis po to, kai tariamai paprašė išeiti iš tinklo. Atakos pavaizdavimas sraute 1.7 pav.



1.7 pav. Deautentifkavimo ataka

Ši ataka gali būti nukreipta į vieną vartotoją, siunčiant kadrus su padirbto šaltinio lauku, nukreipiančiu į bet kurį kompiuterį, kurį jie nori atjungti, arba gali nukreipti į visą tinklą. Apgaulingos autentifikavimo užklausos gaunamos iš prieigos taško, siunčiamos tinklo šliuzams. Ataka užtrunka mažiau nei penkias minutes ir susideda iš užpuoliko, turinčio nešiojamąjį kompiuterį, ir belaidžio tinklo plokštės, su kuria jis pirmiausia gali sugauti kadrus, naudodamas programinę įrangą srautui fiksuoti tarp prieigos taško ir tinklo plokštės. Tada, naudodamas programą, jis sukurs suklastotus kadrus su bet kokiu adresu.

Nors deautentifikavimo išpuolis yra riboto naudojimo ir riboto pavojingumo, jis gali būti naudojamas kaip inicijavimas didesnių išpuolių, kai kuriais atvejais leidžiant užpuolikui visiškai kontroliuoti aukos srautą. Tokio tipo atakos bus nagrinėjamos toliau, pateikiant išsamią informaciją apie tai, kaip paprasta jas įvykdyti.

Viena iš tokių atakų yra *Evil Twin* ataka. Užpuolikai pastato nešifruotą kenkėjišką prieigos tašką, kurio *ESSID* (tinklo pavadinimas), *BSSID* (prieigos taško tinklo plokštės adresas) ir kanalas yra identiškas tiems, kurie šiuo metu yra prisijungę prie aukos prieigos taško. Tada užpuolikas atpažins ir nukreips visą srautą per savo prieigos tašką, o naudodamas kitą programinę įrangą realiuoju laiku gali paimti prisijungimo duomenis ir sesijas. Signalo stiprumas ir atsakymo greitis taip pat vaidina svarbų vaidmenį nustatant, su kuriuo prieigos tašku klientas vėl susies. Tačiau, jei užpuolikas vis užtvindo tinklą deautentifikavimo pranešimais iš legalaus prieigos taško, jis suklaidina klientą, jis bus visam laikui atjungtas arba galiausiai bus vėl susietas su nauju kenksmingu prieigos tašku.

1.3.3 Neautorizuotas prieigos taškas

Neautorizuoti prieigos taškai yra neteisėti prieigos taškai [5]. Neautorizuoti prieigos taškai gali kelti didelę grėsmę, sukurdami užpakalines duris (angl. *backdoor*) į įmonių tinklus. Tokia spraga leidžia patekti į apsaugotą tinklą išvengiant visų apsaugos priemonių. Kaip žinoma, belaidžiai signalai keliauja oru ir daugeliu atvejų neturi ribų. Jie gali judėti per sienas ir langus, pasiekdami galimus didelius atstumus toli nuo įmonės pastato perimetro.

Šie radijo signalai, esantys už saugaus perimetro, gali vaizduoti neautorizuotus prieigos taškus arba legalius prieigos taškus. Abu gali turėti tuos pačius duomenis, kai kurie iš jų gali būti neskelbtini ir slapti. Skirtumas tarp šių dviejų belaidžių prieigos taškų yra tas, kad neautorizuotą prieigos tašką įdiegė ribotos apsaugos darbuotojas, dažnai palikdamas jį numatytomis konfigūracijomis, o legalų prieigos tašką įdiegė kvalifikuotas inžinierius.

Esmė ta, kad neautorizuoti prieigos taškai, kuriuos įdiegė darbuotojai, kelia didelę grėsmę, nes jie naudoja silpnas saugumo priemones, tuo pat metu išplečiant korporacijų tinklo galimybes iš išorės, nesiimant jokių saugumo priemonių. Taip pat tokiais neautorizuotais prieigos taškais gali pasinaudoti užpuolikai, kurie nori stebėti tinklą ar net perimti tinklo srautą tarp bendraujančių įrenginių, tai yra vienas iš būdų kaip įvykdyti *man in the middle* ataką.

1.4 Belaidžio tinklo apsaugos sistemos

1.4.1 Įsibrovimo aptikimo sistemos

Įprastas būdas apsaugoti belaidį tinklą yra suprojektuoti arba naudoti saugos mechanizmus, tokius kaip autentifikavimo mechanizmai, virtualūs privatūs tinklai (VPT) ir ugniasienės, sukuriančios apsauginį barjerą visame tinkle. Tačiau tokios saugumo priemonės turi neišvengiamų pažeidžiamumų ir

paprastai jų nepakanka užtikrinti, kad sistemos būtų saugios visą laiką. Kita vertus, užpuolikai visada bando rasti būdų, kaip patekti į sistemas. Dėl šios priežasties reikėjo saugumo technologijų, kurios galėtų stebėti sistemas, nustatyti galimas grėsmes ir bandyti užkirsti kelią jų pasisekimui.

Įsibrovimo aptikimo prevencijos sistema [6] (angl. *Intrusion detection prevention system*) gali būti naudojamas papildyti įprastus saugumo mechanizmus. Tai teikia keturias pagrindines saugos funkcijas. Šios funkcijos apima neįprastos ir neleistinos veiklos stebėjimą, analizę, aptikimą ir prevenciją. Įsibrovimo aptikimo sistemos pagalba siekiama nustatyti kenkėjišką veiklą, kad būtų išvengta didesnės žalos saugomoms sistemoms.

Įsibrovimo aptikimo sistema (angl. *Intrusion detection system*) yra programinė ar aparatinė įranga, kuri automatiškai nustato įsibrovimą į sistemą. Įsibrovimo prevencijos sistema (angl. *Intrusion prevention system*) nustato įsibrovimus ir gali bandyti sustabdyti įsibrovimą. Be to, Įsibrovimo prevencijos sistema taip pat gali palyginti signalus su žinomais parašais iš anksčiau aptiktų įsibrovimų dabartinėje sistemoje ar iš surinktų ir paskelbtų atakų parašų duomenų bazėje.

Yra du pagrindiniai būdai, kaip aptikti įsilaužimus, paremtus parašu, įskaitant parašo ir anomalijų nustatymus. Parašais pagrįstą būdą galima lengvai palyginti su antivirusine programine įranga, analizuojančia ir apibūdinančia išpuolių detales, formuojant parašus. Kai parašai bus apibūdinti, jų galima bus ieškoti ir palyginti su turima kompiuterių sistemose informacija, pavyzdžiui audito duomenų žurnalai. Priešingai anomalijos aptikimas pastebi neįprastą elgesį tinkle ar sistemoje, palyginti su tuo, kas apibūdinama kaip „normalu“. Šiam metodui svarbu sukurti konstrukcijas, skirtas normaliam vartotojui, pagrindinio kompiuterio ir tinklo elgesiui, atliekant sulyginimą su įprastais surinktų duomenų elementais. Įvykio duomenys, kuriuos stebi įsibrovimo aptikimo sistema, yra lyginami su įvairiomis veiklomis, kad būtų galima nustatyti, kas yra normalu, o kas gali būti laikoma nenormaliu ir iššauktų aliarmą. Aptikimo būdai palyginti 1.1 lentelėje.

1.1 lentelė. Įsibrovimų aptikimo sistemų privalumai ir trūkumai

Įsibrovimo aptikimo būdai	**Privalumai**	**Trūkumai**
Parašu grįstos sistemos	Veiksmingas metodas ir didelis aptikimo tikslumas, kai yra žinoma ataka Mažos resursų sąnaudos	Sudėtinga atnaujinti turimą žinių bazę Daug netikrų pavojų pranešimų, nežinant atakos ar spragos
Anomalijų aptikimu grįstos sistemos	Efektyvus aptikti naujas spragas Mažiau priklausomas nuo operacinės sistemos	Laiko reikalaujantis, kad išklasifikuotų atakas Sudėtinga aktyvuoti pranešimu teisingu metu

Skaitmeninė kriminalistika – tai tyrimas, kurio metu siekiama nustatyti, atsekti ir analizuoti neteisėtus ir nesąžiningus įvykius bei pateikti įrodymus, kad įstatymų tvarka būtų sutvarkyti įvykiai. Įsibrovimo aptikimo prevencijos sistema gali būti naudojama teikiant, registruojant ir dokumentuojant informaciją, reikalingą įtartinai ankstyvai veiklai nustatyti, ir tai netgi gali padėti išvengti rimtesnės žalos padarymo. Taigi įsibrovimo aptikimo prevencijos sistema yra ne tik labai naudinga priemonė rinkti ir aiškinti skaitmeninius įrodymus, kurie gali būti naudojami teisme, bet taip pat gali sudaryti

bendrą sistemos veiklos vaizdą ir išbandyti prieštaringos aplinkos efektyvumą, nustatydama strategijas, kurios pažeidžia saugumą ir privatumą.

Be to, įsibrovimo aptikimo prevencijos sistema pateikia vertingos informacijos apie tai, kaip įvyko išpuolis, ko pasiekė įsibrovėlis ir kokius metodus įsibrovėliai naudojo savo tikslams pasiekti, net jei įsibrovimo aptikimo prevencijos sistema nesugeba užkirsti kelio įsibrovimui. Žmogus, organizacija ar verslas gali gauti naudos iš šios papildomos informacijos, kad galėtų greitai reaguoti į neįprastą sistemos veiklą ar taisyti saugumo priemones ir bandyti užkirsti kelią jiems pasiekti tinkamu laiku. Galiausiai jis gali būti naudojamas formuluoti nuolatinius saugumo patobulinimus ateityje.

1.4.2 Esamos įsibrovimo aptikimo sistemos

Buvo išnagrinėtos jau esamos belaidžio tinklo saugos aptikimo sistemos: *Cisco wIPS, Aruba Hybrid WIDS, OpenWISP*.

Cisco wIPS [7] įterpia belaidžio tinklo grėsmių aptikimą ir jų mažinimą į belaidžio tinklo infrastruktūrą. Tai yra vienas iš išsamesnių, tikslesnių ir ekonomiškai efektyvesnių belaidžio tinklo saugumo sprendimų. *Cisco* sistema teikia šias pagrindines funkcijas ir privalumus: neautorizuotų prieigos taškų aptikimas, vietos nustatymas, klasifikavimas, neautorizuotų prieigos taškų atakų įvykdymo rizikos sumažinimas. Ši sistema nustato belaidžio tinklo išpuolius, tokius kaip: neautorizuoto prieigos taškai, *DoS* atakos, *man in the middle* atakos, taip pat įvairių naujų ir nežinomų atakų atpažinimas. Sistema automatiškai nuolatos stebi ir įvertina belaidį tinklą dėl saugumo spragų, bei konfigūravimo klaidų. Ši sistema integruota į *Cisco Prime* infrastuktūrą, kurios pagalba galima valdyti ir stebėti visą tinklą.

Aruba Hybrid WIDS [8] sistema (kitaip negu *Cisco*) naudoja hibridinę stebėjimo metodą. Įsibrovimų aptikimui ir apsaugai naudoja tą patį įrenginį kaip prieigos tašką ir belaidžio tinklo stebėjimo stotelę, tokiu atveju nebereikia atskiro įrenginio apsaugai. Dėl šios priežasties įrenginių dislokavimas, lankstumas padidėjimas sąlygoja mažiau problemų. Visas valdymas bei analizavimas vyksta centriniame valdiklyje. Kaip ir *Cisco* sistema, *Aruba* taip pat efektyviai apsaugo belaidį tinklą.

OpenWISP-NG [9] yra atvirojo kodo programinė įranga, skirta belaidžio tinklo įsibrovimų aptikimui, ir prevencijos sistema, kuri sudaryta iš jutiklių, serverio ir sąsajos. Ši sistema gali veikti ant įvairios įrangos, sistema veikia ant *OpenWRT* [10] aparatinės programinės įrangos. Sistemą sudaro: jutikliai, kurie fiksuoją srautą ir siunčia informaciją į serverį, serveris, kuris kaupia visų jutiklių duomenis, analizuoja juos ir reaguoja į išpuolius bei viską registruoja žurnaluose, ir sąsaja, per kurią leidžiama valdyti serverį ir kurioje pateikiama informaciją vartotojui.

Dar viena atvirojo kodo sistema pateikta kitame darbe [11], kuriame pasiūlytas sprendimas naudoja tą patį prieigos tašką belaidžiui tinklui bei informacijos fiksavimui apie aplinkinį tinklą. Ši sistema ir recenzuotas straipsnis buvo pristatyta *Inovatyvios (eko)technologijos, verslumas ir regionų plėtra* (angl. *Innovative (Eco-)Technology, Entrepreneurship and Regional Development, IECOTERD*) konferencijoje [12]. Tačiau ši sistema nepasižymi plačiu atakų aptikimo spektru.

1.4.3 Belaidžio tinklo srauto analizavimo metodai

1.4.3.1 Mašininio mokymo ir giliojo mokymo skirtumai

Yra daugybė galvosūkių apie mašininį mokymą, gilųjį mokymą ir dirbtinio intelekto ryšį. Tai techninis mokslas, tiriantis ir plėtojantis teorijas, metodus ir programas, imituojančias, praplečiančias ir

pagilinančias žmogaus intelektą. Tai kompiuterių mokslo šaka, kuria siekiama suprasti intelekto esmę ir pagaminti naujo tipo intelektualųjį aparatą, kuris reaguoja panašiai kaip žmogaus intelektas. Tyrimai šioje srityje apima robotiką, kompiuterių mokslą, kalbos apdorojimo sistemas.

Mašininis mokymas yra DI šaka, kuri glaudžiai susijusi (ir dažnai sutampa) su skaičiavimo statistika, kuri daugiausia sutelkiama į prognozavimą, naudojant kompiuterius. Tai yra stipriai susiję su matematiniais optimizavimais, nes yra panaudojami metodai, teorijos ir pritaikymai srityje. Mašininis mokymas taip pat gali būti naudojamas mokantis ir nustatant pradinius elgesio profilius įvairiems subjektams ir panaudotas aptikti reikšmingas anomalijas. Mašininis mokymas daugiausia dėmesio skiria klasifikavimui ir regresijai pagal žinomas savybes, anksčiau išmoktas iš mokymo duomenų [13].

Gilusis mokymas yra naujas mašininio mokymosi tyrimų laukas. Pagrindinė savybė yra neuroninio tinklo sudarymas, kuris imituoja žmogaus smegenis analitiniam mokymuisi, ir sukūrimas. Tai imituoja žmogiškuosius smegenų mechanizmus, skirtus aiškinti paveikslėlius, garsus ir tekstus.

Gilusis mokymasis yra mašininio mokymosi metodas, pagrįstas duomenų mokymosi apibūdinimu. Vaizdas gali būti išreikštas įvairiais būdais, pavyzdžiui, kiekvienos pikselio intensyvumo vertės vektoriumi, arba abstrakčiau kaip kraštų serija, tam tikros formos sritis ar pan. Naudojant specialias reprezentacijas, užduotis lengviau išmokti iš egzempliorių. Panašiai kaip mašininio mokymosi metodai giliojo mokymosi metodai taip pat apima prižiūrimą ir neprižiūrimą mokymąsi. Giliojo mokymosi pranašumas yra neprižiūrimų ar pusiau prižiūrimų funkcijų mokymasis ir hierarchinis ypatybių ištraukimas, kad būtų galima veiksmingai pakeisti savybes rankiniu būdu. Mašininio mokymosi ir giliojo mokymosi skirtumai yra šie [13]:

- Duomenų priklausomybės. Pagrindinis skirtumas tarp gilaus ir mašininio mokymosi yra jo našumas didėjant duomenų kiekiui. Giluminiai mokymosi algoritmai neveikia taip gerai, kai duomenų kiekis yra mažas, nes norint gerai suprasti duomenis, norint giliai išmokti algoritmus reikia daug duomenų. Ir atvirkščiai, kai mašininio mokymosi algoritmas naudoja nustatytas taisykles, našumas bus geresnis.
- Aparatinės įrangos priklausomybės. Giliojo mokymosi algoritmas reikalauja daug matricų operacijų. Norint, kad matricos operacija būtų veiksmingai optimizuota, GP turi būti naudojama tik labai daug. Todėl vaizdo plokštė yra aparatinė įranga, reikalinga giliajam mokymuisi tinkamai veikti.
- Savybių apdorojimas. Tai pagrindinių žinių įtraukimas į savybių ištraukiklį, siekiant sumažinti duomenų sudėtingumą ir generuoti modelius, leidžiančius geriau naudoti algoritmus. Savybių apdorojimas reikalauja laiko ir specialių žinių. Mašininio mokymosi metu daugumą programos ypatybių turi nustatyti ekspertas, o tada užkoduojamas kaip duomenų tipas. Bandymas išgauti aukšto lygio savybes tiesiogiai iš duomenų sudaro didelį skirtumą tarp giliojo ir mašininio mokymosi algoritmų. Taigi giliojo mokymosi nereikia pritaikyti skirtingai problemai spręsti.
- Problemų sprendimo metodai. Taikydamas tradicinius mašininio mokymosi algoritmus problemoms spręsti, mašininis mokymasis paprastai išskiria problemą į daug problemų ir išsprendžia papildomas problemas, galiausiai gaudamas galutinį rezultatą. Priešingai, gilusis mokymasis randa problemos sprendimą tiesiogiai.
- Vykdymo laikas. Apskritai giliojo mokymosi algoritmo mokymas užtrunka ilgai, nes giliojo mokymosi algoritme yra daug parametrų, todėl apmokymo žingsnis užtrunka ilgiau. Pažangiausias giliojo mokymosi algoritmas trunka lygiai dvi savaites, o mašininio mokymosi algoritmo apmokymas užima palyginti nedaug laiko – tik kelias sekundes ar kelias valandas.

Tačiau testavimo laikas yra visiškai priešingas. Giliųjų mokymosi algoritmų testavimui atlikti reikia labai mažai laiko.

– Aiškinamumas. Aiškumas yra svarbus veiksnys lyginant mašininį mokymą su giliuoju mokymu. Giliojo mokymo ranka rašytų numerių atpažinimas gali prilygti žmonių standartams. Tačiau giliojo mokymosi algoritmas neparodys kaip gaunamas šis rezultatas. Žinoma, matematikos požiūriu yra aktyvuotas gilaus nervinio tinklo mazgas. Taigi sunku paaiškinti, kaip rezultatas buvo sugeneruotas. Priešingai giliajam mokymui, mašininio mokymosi algoritmas pateikia aiškias taisykles, kodėl algoritmas pasirenka jį, todėl nesunku paaiškinti sprendimo pagrindimą.

Mašininio mokymo metodus pirmiausia apima šie keturi etapai:

– Savybių inžinerija. Pasirinkimas kaip numatymo pagrindas (požymiai, savybės).
– Pasirinkti tinkamą mašininio mokymosi algoritmą.
– Apmokyti ir įvertinti modelio veikimą.
– Naudoti išmoktą modelį, kad klasifikuoti ar numatyti nežinomus duomenis.

Giliojo mokymosi metodika yra panaši į tradicinio mašininio mokymo, tačiau, kaip minėta aukščiau, skirtingai nei mašininio mokymosi metodai, savybių ištraukimas yra automatizuotas, o ne rankinis. Modelio pasirinkimas yra nuolatinis bandymų ir klaidų procesas, kuriam reikia tinkamų mašininio mokymo ir giliojo mokymosi tipų. Yra trys mašininio mokymosi / giliojo mokymosi metodų tipai: prižiūrimas, neprižiūrimas ir pusiau prižiūrimas. Prižiūrimo mokymosi metu kiekvieną pavyzdį sudaro įvesties pavyzdys ir etiketė. Prižiūrimas mokymosi algoritmas analizuoja mokymo duomenis ir analizės rezultatus, kuriuos naudoja naujų atvejų sutapatinimui. Neprižiūrimas mokymasis yra mašininio mokymosi užduotis, kai iš nepaženklintų duomenų savarankiškai aptinka naujus modelius ir savybes. Kadangi pavyzdys nepaženklintas, algoritmo išvesties tikslumo įvertinti negalima, o apibendrinti ir paaiškinti galima tik pagrindines duomenų savybes. Pusiau prižiūrimas mokymasis – tai būdas sujungti prižiūrimą mokymąsi su neprižiūrimu mokymu. Pusiau prižiūrimas mokymasis naudoja daug nepaženklintų duomenų, kai šablonų atpažinimui naudojami paženklinti duomenys. Dažniausiai naudojami mašininio mokymosi algoritmai apima k-artimiausią kaimyną (angl. *k-nearest neighbors*), pagalbinio vektoriaus mašiną (angl. *support-vector machine),* sprendimų medį ir naiviojo Bajeso klasifikatorių. Į giliojo mokymosi modelį įeina gilioji Boltzmano mašina (angl. *deep Boltzmann machine),* konvoliuciniai neuroniniai tinklai (angl. c*onvolutional neural network)* ir Ilgoji trumpalaikė atmintis (angl. l*ong short-term memory).* Yra tokių parametrų, kaip pasirinktų sluoksnių ir mazgų skaičius, taip pat patobulinti modelį ir integraciją. Baigus apmokymą, alternatyvus modelis turėtų būti įvertintas skirtingais aspektais. Vertinimo modelis yra labai svarbi mašininio mokymosi misijos dalis.

1.4.4 Belaidžio tinklo atakų aptikimo metodų apžvalga

1.4.4.1 Pagalbinio vektoriaus mašina

Pagalbinio vektoriaus mašina (angl. *support-vector machine)* yra vienas patikimiausių ir tiksliausių metodų iš visų mašininio mokymosi algoritmų. Dažniausiai tai apima pagalbinio vektoriaus klasifikavimą (angl. *support vector classification*) ir pagalbinio vektoriaus regresiją (angl. s*upport vector regression*). Pagalbinio vektoriaus klasifikavimas remiasi sprendimų ribų koncepcija. Sprendimo riba atskiria egzempliorių, turinčių skirtingas klasės reikšmes, rinkinį tarp dviejų grupių. Pagalbinio vektoriaus klasifikavimas palaiko tiek dvejetainių, tiek kelių klasių klasifikavimą. Klasifikavimo procese įvesties vektoriai, esantys atskyrimo hiper plokštumos pusėje, patenka į vieną

klasę, o kita pusė patenka į kitą klasę kitoje plokštumos pusėje. Jei duomenų taškai nėra linijiškai atskirti, pagalbinio vektoriaus mašina naudoja atitinkamas branduolio funkcijas, kad suskirstytų juos į aukštesnių matmenų tarpus, kad jie tose erdvėse taptų atskirtini.

Nagrinėtame tyrime [14] autoriai pasirinko du tipinius duomenų rinkinius: mišrus ir *10% KDD cup 99*. Pagalbinio vektoriaus mašina naudojama klasifikuoti *DoS, Probe, U2R* ir *R2L* duomenų rinkinius. Tyrime apskaičiuojamos parametrų vertės, susijusios su įsibrovimų aptikimų sistemos veikimo įvertinimu. Mišraus duomenų rinkinio patvirtinimo tikslumas ir *10 % KDD cup 99* duomenų klasifikavimo tikslumas buvo įvertintas atitinkamai 89,85 % ir 99,9 %.

Kitame tyrime [15] autoriai pasiūlė hibridinį *PSO-SVM* metodą, kuriant įsibrovimų aptikimų sistemą. Tyrime buvo naudojami du savybių mažinimo būdai: informacijos gavimas ir *BPSO* (angl. *binary particle swarm optimization*). 41 atributas sumažintas iki 18 atributų. Klasifikavimo efektyvumas buvo nurodytas kaip 99,4 % naudojant *DoS* ataką, 99,3 % – zondavimo ar skenavimo ataką, 98,7 % – *R2L* ir 98,5 % naudojant *U2R* ataką. Šis metodas pateikia gerą paslaugų atsisakymo (*DoS*) atakų nustatymo rezultatą ir pasiekia gerą aptikimo greitį *U2R* ir *R2L* atakų atveju. Tačiau zondavimo, *U2R* ir *R2L* tikslumas yra atitinkamai 84,2 %, 25,0 % ir 89,4 %. Šis metodas yra linkes į didesnį klaidingo aliarmo dažnį.

1.4.4.2 K-arčiausias kaimynas

K-arčiausias kaimynas (angl. *k-nearest neighbors*) klasifikatorius yra pagrįstas Euklido atstumo funkcija, matuojančia skirtumą ar panašumą tarp dviejų egzempliorių. Standartinis Euklido atstumas tarp dviejų *x* ir *y* atvejų yra apibrėžiamas formule (1).

$$\boldsymbol{d(x,y) = \sqrt{\sum_{k=1}^{n}(x_k - y_k)^2};} \qquad (1)$$

čia *d(x,y)* yra standartinis Euklido atstumas tarp dviejų *x* ir *y* atvejų; x_k yra *k*-asis *x* egzemplioriaus elementas; y_k yra k-asis duomenų rinkinio elementas; *n* visų objektų skaičius.

Tarkime, kad k-arčiausio kaimyno klasifikatoriaus duomenų rinkinys yra *U*. Bendras pavyzdžių skaičius rinkinyje yra *S*. Tegul $C = \{C_1, C_2, \ldots C_L\}$ yra atskiros *L* klasės etiketės, kurios yra *S*. Tegul *x* įvesties vektorius, kuriam reikia numatyti klasės etiketę. Tegul y_k žymi *k*-ąjį vektorių projektiniame rinkinyje *S*. K-arčiausio kaimyno klasifikatoriaus algoritmas turi surasti k artimiausius vektorius, suprojektuotuose elementuose *S* pagal įvesties vektorių *x*.

Nagrinėtame darbe [16] pateikiama k-vidurkių ir k-arčiausio kaimyno algoritmo kombinuoto įsilaužimo aptikimo sistema. Pirma, įvestiniai invazijos duomenys (*NSL-KDD*) iš anksto apdorojami atliekant pagrindinių komponentų analizę, kad būtų parinkta 10 svarbių savybių. Tada šie iš anksto apdoroti duomenys yra padalijami į tris dalis ir įtraukiami į k-vidurkių algoritmą, norint gauti grupavimo centrus ir etiketes. Šis procesas vykdomas 20 kartų, kad būtų galima pasirinkti geriausią grupavimo schemą. Tuomet šie grupių centrai ir etiketės naudojami klasifikuojant įvestų duomenų panaudojant k-arčiausio kaimyno algoritmą. Eksperimento metu buvo naudojami du metodai, kad būtų galima palyginti siūlomą metodą ir k-arčiausio kaimyno algoritmo rezultatus. Įgyvendintos programos, kad ištirtų rezultatus. Pirmuoju atveju bandymo duomenys yra atskirti nuo mokymo duomenų, tuo tarpu antruoju atveju kai kurie bandymo duomenys nėra pakeisti mokymo duomenimis. Tačiau bet kuriuo atveju vidutinis eksperimento tikslumas buvo maždaug 90 % ir jis neatsižvelgė į tikslumą bei atmetimo dažnį.

Kitame tyrime [17] k-arčiausio kaimyno algoritmas buvo naudojamas įsilaužimui aptikti tame pačiame *KDD Cup 99* duomenų rinkinyje. Pagrindinis skirtumas yra tas, kad *KNN*, *SVM* ir *pdAPSO* algoritmai yra sumaišomi, norint aptikti įsibrovimus. Eksperimento rezultatai rodo, kad maišant skirtingus klasifikatorius galima pagerinti klasifikavimo tikslumą. Statistiniai rezultatai rodo, kad klasifikavimo tikslumas yra 98,55 %. Išskyrus tikslumą, tyrime nebuvo įvertinti kiti rodikliai.

1.4.4.3 Sprendimų medis

Sprendimų medis yra medžio struktūra, kurioje vidinis mazgas parodo vienos savybės testą, o kiekviena šaka – bandymo išvestį, o kiekvienas lapo mazgas reiškia kategoriją. Mašininiame mokyme sprendimų medis yra numatomasis modelis; jis parodo objekto atributų ir objekto verčių tapatumą. Kiekvienas mazgas medyje atspindi objektą, kiekvienas išsiskyrimo kelias pateikia galimą atributo vertę ir kiekvienas lapo mazgas atitinka objekto vertę, nurodytą keliu nuo šaknies mazgo iki lapo mazgo. Sprendimų medis turi tik vieną išvestį; jei yra sudėtinga išvestis, galima sukurti nepriklausomą sprendimų medį, skirtą įvairiems išvestims tvarkyti.

Sprendimų medis klasifikuoja pavyzdžius pagal mokymo sąlygas ir turi geresnį žinomų įsibrovimo metodų aptikimo tikslumą, tačiau jis nėra tinkamas nežinomo įsibrovimo aptikimui.

Nagrinėtame darbe [18] ištirta sprendimų medžiu pagrįsta įsibrovimų aptikimo sistema, *NSL-KDD* duomenų rinkinys. Savybių pasirinkimas, naudojant koreliacijos savybių pasirinkimo (*CFS*) metodą, iš kiekvieno duomenų imties pasirenkant 14 savybių, pagerina sprendimų medžiu pagrįstų įsibrovimų aptikimo sistemos numatymo našumą. Našumas buvo vertinamas atskirai penkioms ir dviem kategorijoms; bendras tikslumas buvo atitinkamai 83,7 % ir 90,3 %. Iš eksperimento rezultatų pateiktas metodas nebuvo pakankamai tikslus.

Sekančiame tyrime [19] autoriai siūlo du savybių pasirinkimo metodus: *C4.5* sprendimo medžio algoritmą ir *C4.5* sprendimo medį (su genėjimu). Apmokymo ir testavimo klasifikatoriai naudoja *KDD Cup 99* ir *NSL-KDD* duomenų rinkinius. Klasifikacijos procese laikomos tik diskretinės vertės *protocol_type, Service, flag, land, logged_in, is_host_login, is_guest_login* ir *class*. Eksperimento rezultatai rodo, kad *C4.5* (su genėjimu) yra tikslesnis 98,45 %, 1,55 % daugiau nei *C4.5* sprendimo medis.

1.4.4.4 Giliojo tikėjimo tinklas

Giliojo tikėjimo tinklas *(angl. Deep belief network*) yra tikimybinis generacinis modelis, susidedantis iš kelių stochastinių ir paslėptų kintamųjų sluoksnių. Apribota Boltzmann‘o mašina (*angl. Restricted Boltzmann machine*) ir giliojo tikėjimo tinklas yra tarpusavyje susijusios, nes daugelio apribotų Boltzmano mašinų sudarymas ir sudėliojimas įgalina daugelį paslėptų sluoksnių efektyviai mokyti duomenis, aktyvinant vieną apribotą Boltzmano mašiną tolimesniam treniravimo dydžiui. Apribotos Boltzmano mašinos principas kilo iš statistinės fizikos kaip modeliavimo metodo, pagrįsto energijos funkcija, galinčia apibūdinti aukšto laipsnio kintamųjų sąveiką. Boltzmano mašina yra simetriškai susietas atsitiktinio grįžtamojo ryšio dvejetainis neuroninis tinklas, sudarytas iš matomo sluoksnio ir daugybės paslėptų sluoksnių. Tinklo mazgas yra padalintas į matomą vienetą ir paslėptą vienetą, o matomasis ir paslėptasis vienetai naudojami atsitiktiniam tinklui ir atsitiktiniai aplinkai išreikšti. Mokymosi modelis išreiškia koreliaciją tarp vienetų pagal svorius.

Nagrinėto darbo autoriai [20] palygino skirtingas giliojo tikėjimo tinklų struktūras, pakoregavo tinklo modelio sluoksnių skaičių ir paslėptų neuronų skaičių bei gavo keturių sluoksnių giliojo tikėjimo tinklo

modelį. *KDD Cup 99* duomenų rinkinys buvo naudojamas testavimui. Modelio tikslumas, preciziškumas buvo 93,49 % ir 92,33 %.

Kitame nagrinėtame darbe [21] įgyvendinamas metodas grindžiamas giliojo tikėjimo tinklu, naudojant logistinės regresijos *soft-max funkciją* giliojo tinklo sureguliavimui. Kelių klasių logistinės regresijos sluoksnis buvo mokomas 10 epochų, atsižvelgiant į iš anksto patobulintus apmokymo duomenis, siekiant pagerinti bendrą tinklo veikimą. Šiuo metodu buvo pasiektas 97,9 % aptikimo rodiklis visame *10 % KDD Cup 99* bandymo duomenų rinkinyje.

1.4.4.5 Rekurentiniai neuroniniai tinklai

Sekos duomenims apdoroti naudojamas rekurentinis neuroninis tinklas (angl. *recurrent neural network*). Sluoksniai yra visiškai sujungti ir tarp kiekvieno sluoksnio mazgų nėra ryšio. Yra daug problemų, kurių šis įprastas neuroninis tinklas negali išspręsti. Rekurentinio neuroninio tinklo dabartinis sekos išėjimas taip pat yra susijęs su išėjimu prieš jį. Konkreti išraiška yra ta, kad tinklas gali atsiminti informaciją apie ankstesnį momentą ir ją pritaikyti skaičiuojant dabartinę išvestį; tai yra, mazgai tarp paslėptų sluoksnių sujungiami, o paslėpto sluoksnio įvestis apima tik pradinio sluoksnio išvestį ir paskutinės akimirkos paslėpto sluoksnio išvestį. Teoriškai bet koks rekurentinio neuroninio tinklo sekos duomenų ilgis gali būti apdorojamas. Tačiau praktikoje, siekiant sumažinti sudėtingumą, dažnai manoma, kad dabartinė būklė yra susijusi tik su ankstesnėmis būsenomis.

Nagrinėto darbo autoriai [22] siūlo įsilaužimo aptikimą, remiantis cikliniu neuroniniu tinklu. Duomenų rinkinys *NSL-KDD* buvo naudojamas įvertinti modelio veikimą dvejetainėje klasifikacijoje ir daugiaklasėje klasifikacijoje, taip pat neuronų skaičiaus ir skirtingų mokymosi greičių įtaką modelio veikimui. Treniravimo tikslumas ir testavimo tikslumas (gauti modeliai dvejetainėje klasifikacijoje) yra atitinkamai 99,81 % ir 83,28 %.

Darbo [23] autoriai lygina šešių dažniausiai naudojamų optimizatorių poveikį įsibrovimo aptikimo modeliui (angl. *long short-term memory*, *LSTM*). Eksperimentuodamas su *KDD Cup 99* duomenų rinkiniu, *LSTM RNN* modelis su *Nadam* optimizatoriumi [23] lenkia ankstesnius darbus. Įsibrovimo tikslumas yra 97,54 %, preciziškumas yra 98,95 %.

1.4.4.6 Konvoliucinis neuroninis tinklas

Konvoliuciniai neuroniniai tinklai (angl. C*onvolutional neural network*) yra tam tikros rūšies dirbtinis neuroninis tinklas, tapęs pagrindiniu metodu kalbos analizės ir vaizdo atpažinimo srityje. Dėl svorių paskirstymo tinklo struktūroje jis tampa panašesnis į biologinį neuroninį tinklą, todėl tinklo modelio sudėtingumas sumažėja, nes mažėja svorių skaičius. Šis pranašumas yra akivaizdesnis, kai įvestis yra kelių dimensijų nuotrauka, kuri gali būti panaudota tiesiogiai, kad būtų išvengta sudėtingų savybių išsiskyrimo ir duomenų rekonstravimo kaip tradiciniame atpažinimo algoritme. Konvoliucinis tinklas yra daugiasluoksnis jutiklis, specialiai suprojektuotas taip, kad atpažintų dviejų matmenų formas, kurie yra labiausiai nepakitę vertimui, mastelio keitimui, pakreipimui ar kitoms deformacijų formoms. Konvoliucinis neuroninis tinklas yra pirmasis tikrai sėkmingas mokymosi algoritmas daugiasluoksnėms tinklo struktūroms mokyti. Tai sumažina parametrų, kuriuos reikia išmokti, norint pagerinti algoritmo mokymo efektyvumą, erdvinių ryšių skaičių. Kaip giliojo mokymosi architektūra, konvoliucinis neuroninis tinklas yra siūlomas siekiant sumažinti duomenų išankstinio apdorojimo reikalavimus. Yra trys pagrindinės konvoliucinio neuroninio tinklo priemonės, skirtos sumažinti tinklo treniruočių parametrus: vietinis jautrumas, svorio pasidalijimas ir sutelkimas. Pati galingiausia

algoritmo dalis yra mokymosi savybių hierarchijos iš didelių nepaženklintų duomenų kiekių. Todėl konvoliucinis neuroninis tinklas yra perspektyvus naudojimui tinklo įsibrovimo aptikimo srityje.

Nagrinėtame darbe [24] autoriai perkelia našumo patobulinimus neuroninių tinklų srityje į išardytų kenksmingų dvejetainių failų vykdymo sekos modeliavimą. Įdiegtas neuroninis tinklas, susidedantis iš konvoliucijos ir tiesioginio sklidimo neuroninių struktūrų. Ši architektūra įkūnija hierarchinį bruožų ištraukimo metodą, kuris ypatybes sujungia su paprastu konvoliucijos vektorizavimu. Straipsnyje ypatybės ištraukiamos iš nešiojamųjų vykdomųjų failų antraščių tik vertinimui. Rezultatai rodo, kad siūlomas metodas pralenkia lyginamuosius metodus, tokius kaip tiesioginio sklidimo neuroninį tinklą ir pagalbinio vektoriaus mašiną. *F1* balas yra 92 %, tikslumas siekia 93 %.

1.5 Darbo tikslas, uždaviniai, planas ir siekiami privalumai

Darbo tikslas sukurti metodą, kuris, ne kaip ištirti metodai, aptinka ne bandomą įvykdyti ataką, o jau įvykdytą ataką. Darbo uždaviniai būtų: algoritmo projektavimas, realizavimas, testavimas, tyrimas. Įvykus nesklandumams ar blogai realizacijai, pakartoti šiuos žingsnius. Šio principo privalumai būtų tokie, kad esamų metodų neaptiktas *man in the middle* atakas pavyktų aptikti, kai jos jau būna sėkmingai įvykdytos.

1.6 Siekiamo sprendimo apibrėžimas

Siekiamas sprendimas skirsis nuo analizuotų taip, kad algoritmas ne ataką bandys atpažinti, bet kad ataka jau bus įvykdyta belaidžiame tinkle. Papildomai šitam sprendimui būtų reikalingi prieigos taškai arba stebėjimo taškai, kurie rinktų belaidžio ryšio kadrus erdvėje. Surinkus belaidžio tinklo kadrus, galima būtų ieškoti *man in the middle* atakos, panaudojant mašininio mokymo algoritmus.

1.7 Analizės išvados

Analizės metu buvo išanalizuotos belaidžio tinklo įvairios saugumo problemos. Paaiškėjo, kad šios problemos nėra patikimai išspręstos. Tokių problemų sprendimui yra naudojamos belaidžio tinklo įsibrovimo aptikimo sistemos, kurios aptinka atakas ir įspėja administratorių arba pačios pradeda imtis veiksmų prieš atakas. Yra skirtingų sistemų veikimo principų, tokių kaip pagrįstų taisyklėmis, žinomomis atakomis arba anomalijų aptikimu. Tokios sistemos naudoja įvairius algoritmus, tokius kaip mašininio mokymo ar giliojo mokymo. Tokio tipo algoritmai buvo išanalizuoti, aprašyti bei išnagrinėti tyrimų duomenys.

2 BELAIDŽIO TINKLO ATAKŲ APTIKIMO PROTOTIPO PROJEKTAS

Šio darbo tikslas yra suprojektuoti belaidžio tinklo *man in the middle* atakos atpažinimo metodą, kuris iš surinktų duomenų, panaudojant suprojektuotą giliojo mokymo metodą, aptiktų ataką.

2.1.1 Sprendimo bendra idėja

Viena senesnių, tačiau vis dar populiarių atakų prieš kompiuterinius belaidžius tinklus yra deautentifikavimo *man in ihe middle* ataka. Ataką sudaro užpuolikas, kuris perima esamą ryšį tarp dviejų kompiuterių. Šias atakas naudoja kenkėjiški vartotojai rinkti konfidencialią informaciją apie kitus tinklo vartotojus. Šioje atakoje užpuolikas save laiko žmogumi viduryje – dviejų kompiuterių ryšio viduryje. Kadangi užpuolikas yra tarp abiejų mašinų, jų tinklo srautas pirmiausia pasiekia užpuoliko mašiną, kol ji nepatenka į aukos kompiuterius. Tada užpuolikas gali nuspręsti pavogti neskelbtiną informaciją, pasirinktinai ją modifikuoti ar atlikti bet kokią kitą kenkėjišką veiklą prieš informacijai pasiekiant numatytą tikslą.

Nors *MITM* atakai įvykdyti egzistuoja skirtingi mechanizmai, tačiau jų elgesys yra bendras. Paprasčiausia ataka kurią galima įvykdyti yra deautentifikavimo ataka. Užpuolikas perduoda srautą, kurį gauna iš bet kurios aukos, kitai šaliai. Mes tai vadiname persiuntimo elgesiu. Tai leidžia iš aukos mašinos sugeneruotiems kadrams patekti į tikrąją paskirties vietą. Todėl tikslinė mašina neįtaria, kad puolėjo mašina yra kelio viduryje. Tiek gautas, tiek atitinkamai persiųstas kadras bus rodomas tinklo sraute kaip du beveik panašūs kadrai. Tai sukuria pusiau pasikartojančius kadrus tinklo sraute. Priežastis, kodėl mes juos vadiname pusiau pasikartojančiais kadrais, yra ta, kad užpuolikas gali pakeisti dalį gautos informacijos apie kadrus, norėdamas ją persiųsti į tikrąją paskirties vietą. Be to, gali pakisti kai kurie kadrų laukai. Belaidžio tinklo kadrus galima palyginti kartu, kad būtų galima rasti šiuos pusiau pasikartojančius kadrus. Vykdomą deautentifikavimo ataką rodo tinklo sraute esantys deautentifikavimo kadrai, kai jie yra nuolatos pakartotinai siunčiami tam tikru adresu.

2.1.2 Sprendimo reikalingumo pagrindimas

Vartotojai nuo m*an in the middle* atakos galėtų apsisaugoti naudodamiesi virtualiais privačiais tinklais (VPT). Tačiau pagal *Global Web Index* statistiką [25] 26 % interneto vartotojų naudoja VPT ir pagal *DataProt* statistiką [26] daugiau kaip pusė visų vartotojų VPT naudoja pasiekti tam tikrus pramogų išteklius, tokius kaip transliacijos paslaugos, kurių dėl regioninių apribojimų nepavyktų pasiekti nesinaudojant VPT. Dėl šios priežasties galima teigti, kad vartotojai naudoja VPT ne dėl asmeninių duomenų apsaugojimo. Dėl tokios situacijos yra reikalinga sistema kuri padėtų suvaldyti tokias atakas.

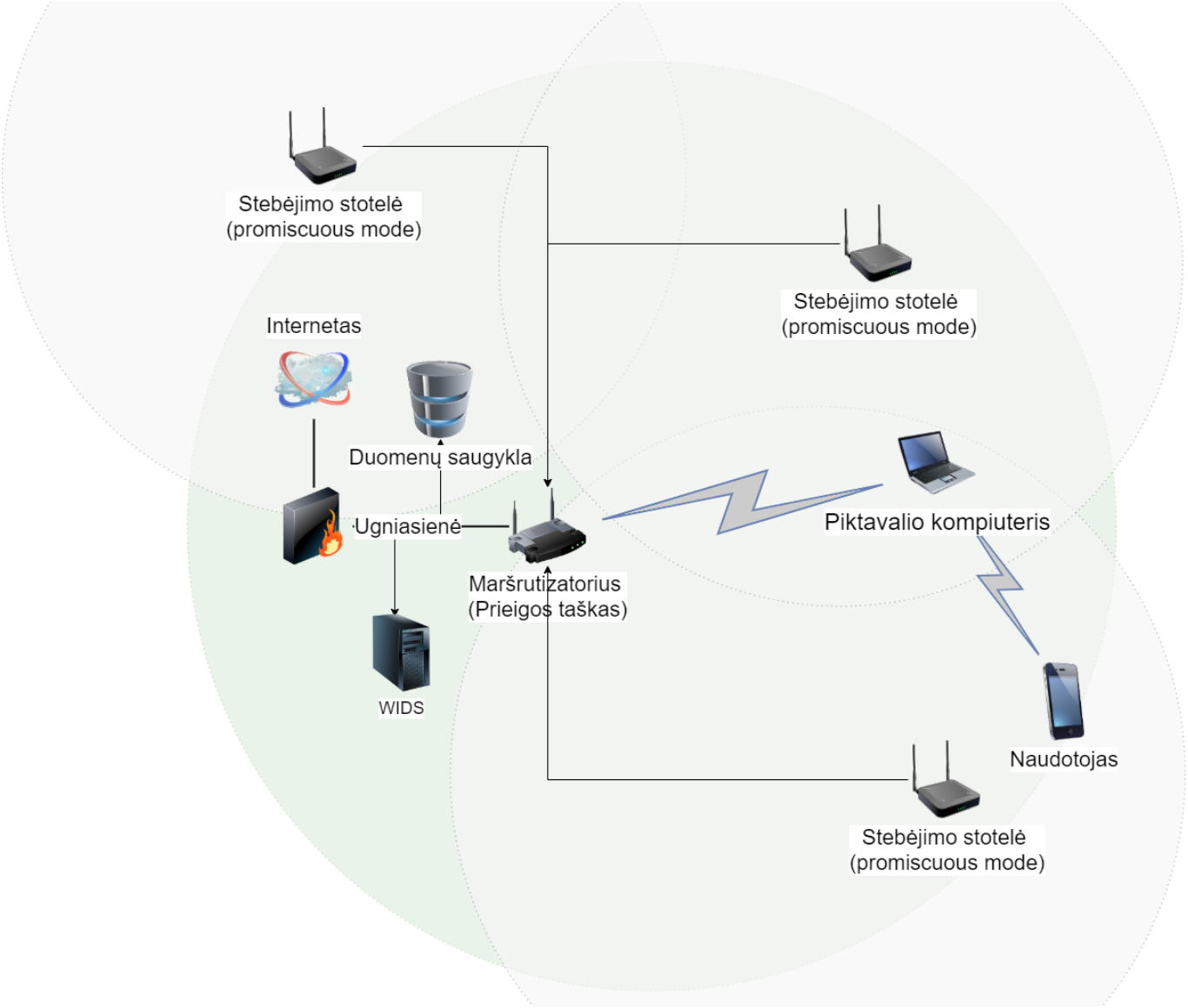
2.1.3 Sprendimo panaudojimas sistemoje

Sprendimas bus panaudotas aprašytoje bakalaurinio baigiamojo darbo *Neautorizuotų prieigos taškų (Rogue AP) belaidžiame tinkle kontrolės sistema* sistemoje [11], kurioje naudojama atvirojo kodo programinė ir aparatinė įranga. Į tai bus atsižvelgta projektuojant ir realizuojant šitą metodą. Viena iš sistemos funkcijų yra esamų prieigos taškų aptikimas erdvėje, bet nustatyti kuriuos prieigos taškus blokuoti turi pats administratorius. Taigi šis kuriamas metodas padėtų administratoriui nustatyti, kuriuos reikalingus prieigos taškus būtina blokuoti.

2.2 Sistemos projektas

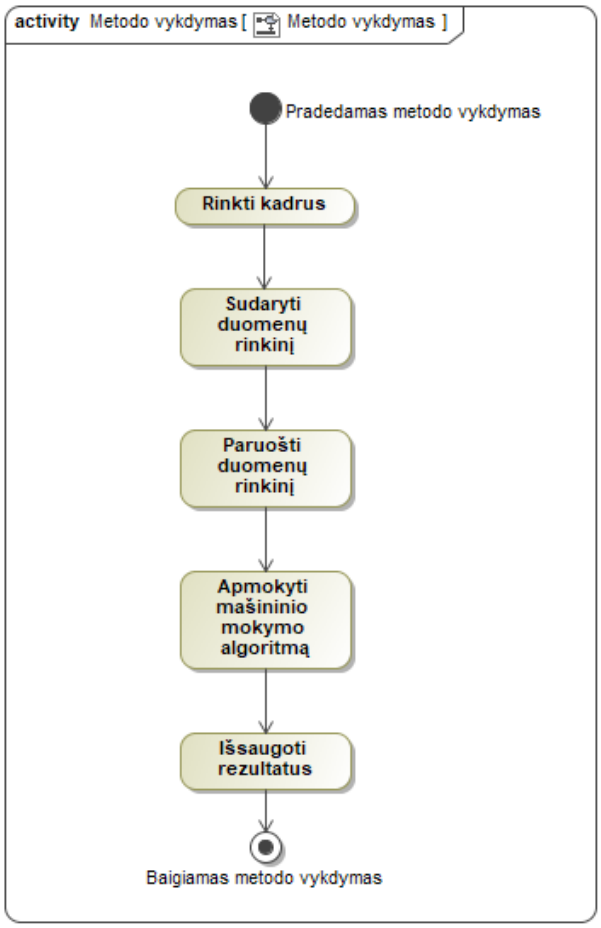
2.2.1 Sistemos architektūra

Sistema susideda iš tinklo, aptikimo metodo ir įrenginio kuriame veiktų metodo. 2.1 pav. pavaizduota tinklo struktūra, kurioje galima matyti ugniasienę, duomenų saugyklą, maršrutizatorių, stebėjimo stoteles, kurios aptinka kadrus, esančius erdvėje, ir *WIDS*, kuriame veiktų aptikimo metodas.



2.1 pav. Prototipo tinklo struktūra

Metodo vykdymo seką galima matyti 2.2 pav. Iš pradžių metodas renka kadrus iš skirtingų stotelių, tada sudaro bendrą duomenų rinkinį. Metodas yra apmokomas paruoštais duomenimis ir gauti rezultatai – ar aptikta deautentifikavimo *mitm* ataka – išsaugomi. Mokymo duomenų rinkinys bei tyrimo duomenų rinkinys bus sudaromas pagal duomenų specifikaciją, kadangi pagal duomenų rinkimo principą nėra viešų sudarytų duomenų rinkinių.



2.2 pav. Metodo vykdymo seka

2.2.2 Funkciniai ir nefunkciniai reikalavimai

Funkciniai reikalavimai:

- Aptikti deautentifikavimo *man in the middle* atakos srautą.
- Surinkti belaidžio tinklo srauto duomenis.

Nefunkciniai reikalavimai:

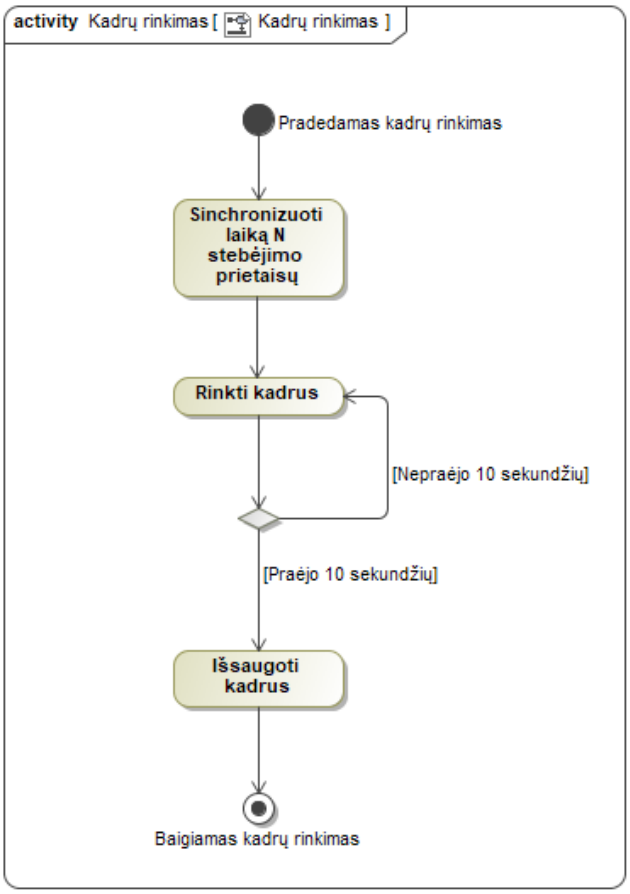
- Metodas turi įsivykdyti per 60 sekundžių.

2.2.3 Rezultato kokybės kriterijai

Metodo kokybės įvertinimas priklausys nuo tinkamai aptinkamų atakų skaičiaus, metodo tikslumo, kuris bus lyginimas su kitų metodų tikslumu.

2.2.4 Duomenų modelio specifikacija

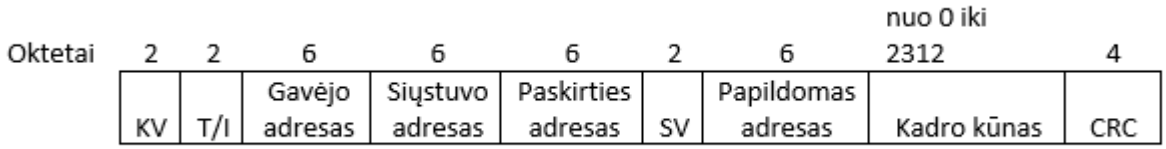
Norint sudaryti bendrą duomenų rinkinį, reikia imtis papildomų veiksmų. Kaip 2.3 pav. schemoje pavaizduota, visi N stebėjimo įrenginiai sinchronizuoja savo laiką ir pradeda *IEEE* 802.11 standarto kadrų erdvėje rinkimą. Yra duotas laiko tarpas, kurio metu renkami kadrai, kad būtų galima analizuoti mažesniais kiekiais. Surinkti kadrai išsaugomi. Tokia pačia seka duomenys renkami apmokymo duomenų rinkiniui, tik laiko tarpas yra didesnis.



2.3 pav. Kadrų rinkimo schema

Toks duomenų rinkimo algoritmas pasirinktas, kad būtų galima aptikti tiek kliento, tiek trečiosios šalies srautą, bei trečiosios šalies su prieigos tašku tinklo srautą.

Toliau 2.4 pav. paaiškintas *IEEE* 802.11 standarto duomenų kadras. Kadro laukai paaiškinti 2.1 lentelėje.



2.4 pav. IEEE 802.11 standarto duomenų kadras

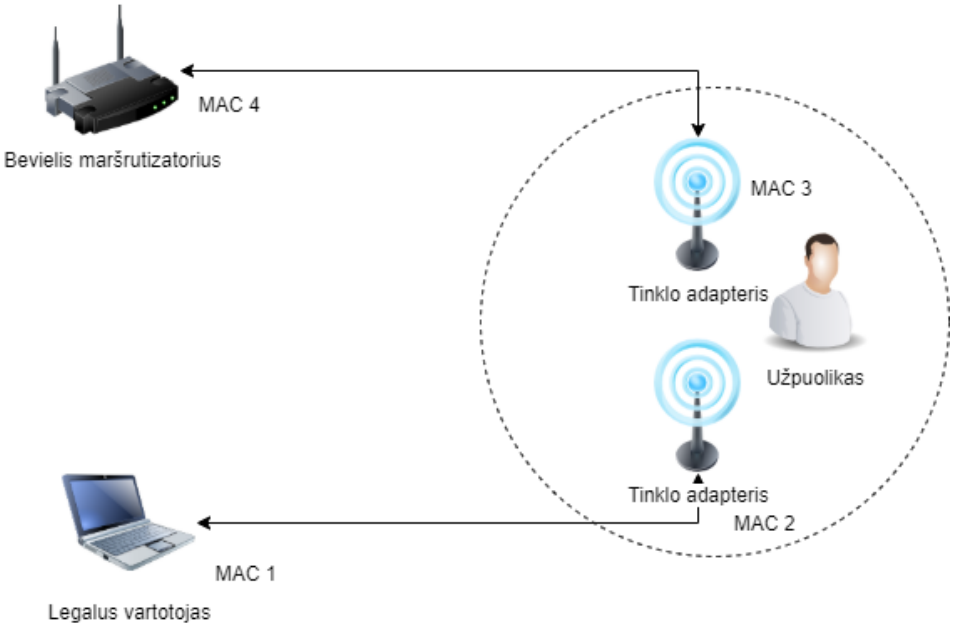
2.1 lentelė. IEEE 802.11 standarto duomenų kadro laukų paaiškinimai

Lauko pavadinimas	**Aprašymas**
KV – kadro valdymas	Nurodo kadro tipą (valdymo, tvarkymo arba duomenų) ir teikia valdymo informaciją. Valdymo informacija apima tai, ar kadras yra skirtas DS, ar ne, fragmentacijos informaciją ir privatumo informaciją.
T\I – trukmė / ryšio ID	Jei naudojamas trukmės laukas, nurodomas laikas (mikrosekundėmis), per kurį kanalas bus skirtas sėkmingam tinklo plokštės kadro perdavimui. Kai kuriuose valdymo kadruose šiame lauke yra asociacijos arba ryšio identifikatorius.
SV – sekos valdymas	Jame yra 4 bitų fragmentų skaičiaus polaukis, naudojamas fragmentavimui ir pakartotiniam surinkimui, ir 12 bitų eilės numeris, naudojamas kadrų, siunčiamų tarp nurodyto siųstuvo ir imtuvo, numeravimui.

Adresai	48 bitų adreso laukų skaičius ir reikšmė priklauso nuo konteksto. Siųstuvo adresas ir imtuvo adresas yra prie prisijungusių stočių, perduodančių ir gaunančių kadrus belaidžiu vietiniu tinklu, tinklo plokščių adresų. Paslaugų rinkinio ID (*SSID*) identifikuoja belaidį vietinį tinklą, per kurį perduodamas kadras. Belaidžiame vietiniame tinkle, kuris yra didesnės konfigūracijos dalis, paslaugų rinkinio ID identifikuoja stotelę, per kurią perduodamas kadras; paslaugų rinkinio ID yra šios stotelės tinklo plokštės lygio adresas. Galiausiai šaltinio ir paskirties adresas yra belaidžių ar kitokių stočių tinklo plokščių adresai, kurie yra pagrindinis šio kadro šaltinis ir paskirtis. Šaltinio adresas gali būti tapatus siųstuvo adresui, o paskirties adresas gali būti tapatus gavėjo adresui.
Rėmo korpusas	Tinklo plokštės turi adreso paslaugų duomenų fragmentą arba MAC valdymo informaciją.
Kadrų tikrinimo seka (CRC)	32 bitų ciklinis tikrinimas

2.2.5 Reikalavimai duomenims

Metodui įvesties duomenys turi būti surenkami naudojant papildomas belaidžio tinklo stoteles (angl. *beacons*), kadangi renkant duomenis tik prieigos taške, prie kurio prisijungęs yra vartotojas, neaptiktų tarpinių kadrų, per kuriuos vykdoma *man in the middle* ataka. Tinklo struktūrą, kurioje bus renkami duomenys, galima matyti 2.1 pav. Naudojamos stebėjimo stotelės, kurių pagalba galima sugauti kadrus, keliaujančius tarp naudotojo ir piktavalio, tarp piktavalio ir maršrutizatoriaus. Paveiksle galima matyti ir likusią tinklo struktūrą: ugniasienė, duomenų saugykla ir pati aptikimo sistema (*WIDS*). Sekančiame paveiksle 2.5 pavaizduotas įvykdytos *man in the middle* atakos srautas su pateiktais tinklo plokštės adresais.



2.5 pav. Sudaryto srauto su žmogumi viduryje schema

2.2.5.1 Duomenų rinkinio išskirtos savybės

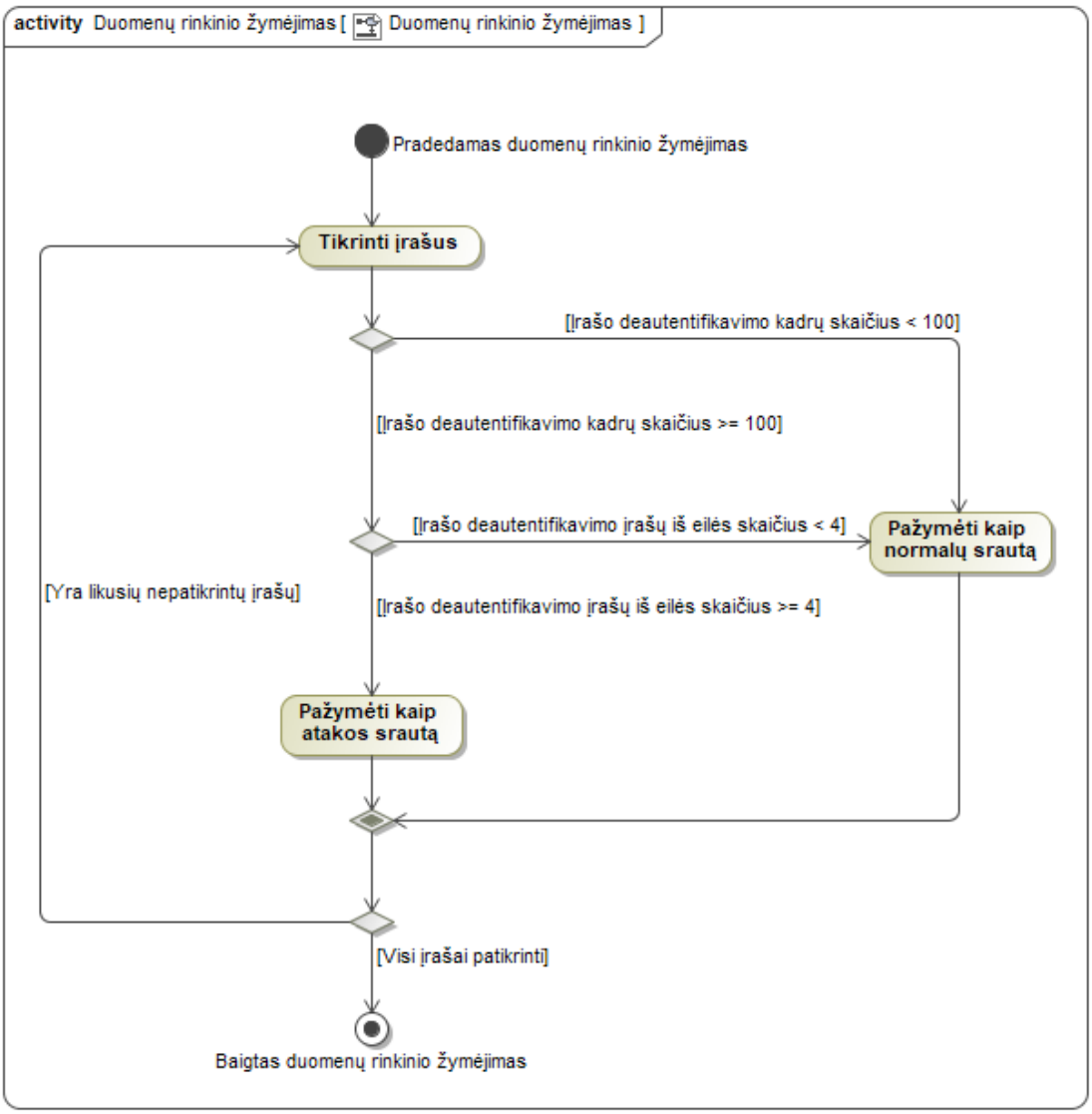
Apmokymo duomenų rinkinys bei tiriamasis duomenų rinkinys bus sudaryti pagal aprašytą specifikaciją, kadangi taip surinkto duomenų rinkinio nepavyko rasti. Išskirtos duomenų rinkinio savybės, kurios bus naudojamos mašininio mokymo metodui, pavaizduotos 2.2 lentelėje.

2.2 lentelė. Duomenų rinkinio savybės

Nr.	Pavadinimas	Aprašymas
1.	Protocol	Protokolo tipas
2.	Length	Kadro ilgis
3.	Arrival Time	Kadro atvykimo laikas
4.	Time delta	Laiko skirtumas tarp kadrų
5.	Frame Number	Kadro numeris
6.	Frame Length on the wire	Kadro ilgis laide
7.	Version	Kadro valdymo lauko versija
8.	Receiver Address	Imtuvo tinklo plokštės adresas
9.	Destination Address	Paskirties tinklo plokštės adresas
10.	Source Address	Siųstuvo tinklo plokštės adresas
11.	Flag	Vėliava
12.	Frame Control Field	Kadro valdymo laukas
13.	Reason code	Priežasties kodas
14.	Fixed parameters	Fiksuoti parametrai
15.	Data	Duomenys
16.	Data Length	Duomenų ilgis
17.	QoS control	Kokybės valdymas
18.	Sequence number	Eilės numeris
19.	Deauthentication Frame Count	Deautentifikavimo kadrų skaičius iš to paties siųstuvo
20.	Count of Deauthentication Frames in Row	Deautentifikavimo kadrų skaičius iš eilės iš to paties siųstuvo

2.2.5.2 Duomenų klasifikavimas

Mašininio apmokymo algoritmui yra reikalingi sužymėti duomenys, kadangi tinklo srautas gali būti didelis duomenų rinkinys. Pats algoritmas pavaizduotas 2.6 pav. Duomenų rinkinys sužymėtas į dvi grupes: *normal* ir *mitm*, grupės aprašytos 2.3 lentelėje. Sužymėti duomenys bus panaudoti apmokyti klasifikavimo algoritmą.



2.6 pav. Duomenų rinkinio žymėjimas

2.3 lentelė. Paženklinti atributai

Savybė	**Aprašymas**
normal	Įprastas tinklo srautas
mitm	Įvykdyta *mitm* ataka

2.2.6 Naudojamas metodas

Kadangi duomenų rinkinys yra sudarytas, o ne paimtas viešas rinkinys, bus galima objektyviai palyginti ir ištirti pasirinktą metodą, taip pat bus įgyvendinti analitinėje dalyje aprašyti mašininio mokymo metodai.

2.2.6.1 Pasirinkto metodo aprašymas

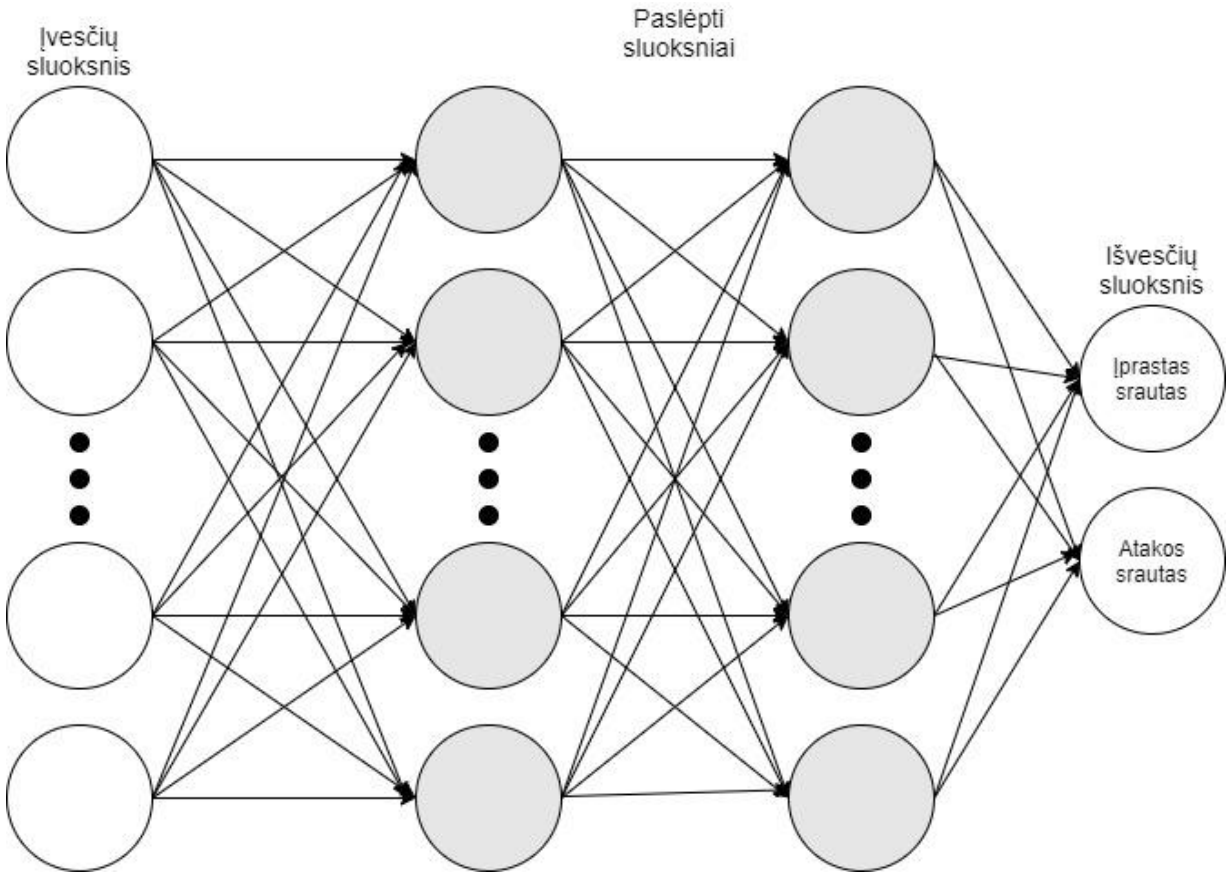
Pasirinktas mašininio mokymo algoritmas yra *Adam*. *Adam* yra optimizavimo algoritmas, kuris gali būti naudojamas vietoj klasikinės stochastinės gradiento nusileidimo procedūros atnaujinant tinklo svorius, iteratyviai pagrįstus mokymo duomenimis. 2015 m. *Adam* pristatė Diederik Kingma iš *OpenAI* ir Jimmy Ba iš Toronto universiteto – *Stochastic Optimization Method*. Algoritmas vadinamas *Adam*.

Pristatydami algoritmą, autoriai išvardija patraukliausius *Adam* naudojimo pranašumus, neišgaubtų optimizavimo problemų atveju:

- Skaičiavimo požiūriu efektyvus.
- Maži atminties reikalavimai.
- Įstrižainės gradientų skalės kitimas.
- Puikiai tinka problemoms, kurios yra didelės duomenų ir (arba) parametrų atžvilgiu.
- Tinka nestacionariems tikslams.
- Tinka problemoms, susijusioms su labai triukšmingais arba retais nuolydžiais.
- *Hyper* parametrai yra intuityviai interpretuojami ir paprastai reikalauja mažai derinimo.

2.2.6.2 Adam optimizatoriaus neuroninio tinklo architektūra

Pasirinktas algoritmas yra giliojo mokymosi algoritmas. Panaudotas neuroninis tinklas, kuris yra pavaizduotas 2.7 pav. Tinklas susideda iš įvesčių sluoksnio, kurį sudaro 20 neuronų, dviejų paslėptų sluoksnių (kiekviename po 300 neuronų) ir išvesčių sluoksnio, kuris susideda iš 2 neuronų. Kiekvieno sluoksnio neuronai sujungti su kiekvienu sekančio sluoksnio neuronu.



2.7 pav. Adam optimizatoriaus neuroninio tinklo architektūra

2.3 Projektinės dalies išvados

Buvo suprojektuotas prototipas, metodo veikimo seka, kadrų rinkimas erdvėje ir duomenų rinkinio sudarymas. Išskirtos duomenų rinkinio savybės, pagal kurias bus apmokomas mašininio mokymo algoritmas. Taip pat parodyta pagal kokius laukus buvo sužymėtas duomenų rinkinys į normalų srautą ir įvykdytos atakos srautą. Pasirinktas Adam giliojo mokymo algoritmas.

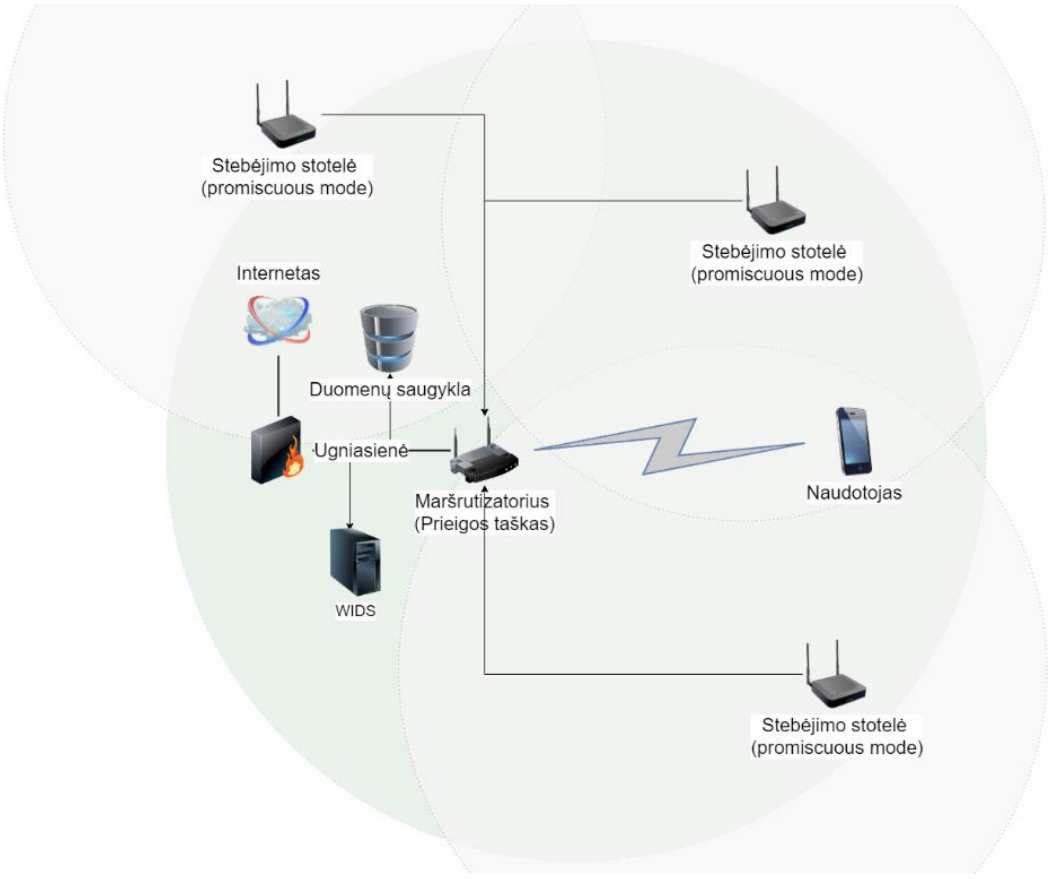
3 BELAIDŽIO TINKLO ATAKŲ APTIKIMO PROTOTIPO REALIZACIJA

3.1 Belaidžio tinklo aptikimo prototipo realizavimo priemonės

1) Metodui reikalinga aparatinė įranga:
    a. *Raspberry Pi 3B* arba *Raspberry Pi 4B* mikrokompiuteris, kuris turi 2.4 GHz integruotą belaidžio tinklo plokštę, 1 GB darbinės atminties bei bent 1 GB atminties.
    b. Belaidžio tinklo *USB* adapteris, palaikantis stebėjimo režimą (angl. *monitor mode*).
2) Metodui reikalinga programinė aparatinė įranga:
    a. *OpenWrt* programinė aparatinė įranga, palaikanti *Raspberry Pi 3B/4B* mikrokompiuterį, pritaikyta *OpenWrt* programinei aparatinei įrangai.
3) Metodui reikalinga papildoma programinė įranga:
    a. *Python 2.7* ar naujesnė versija, palaikanti *Scapy* ir *Tensorflow* bibliotekas.
    b. *Aircrack-ng* programinė įranga.
4) Duomenų rinkinio sudarymui reikalinga papildoma įranga:
    a. Piktavalio kompiuteris su papildomu belaidžio tinklo adapteriu.
    b. Naudotojo įrenginys.

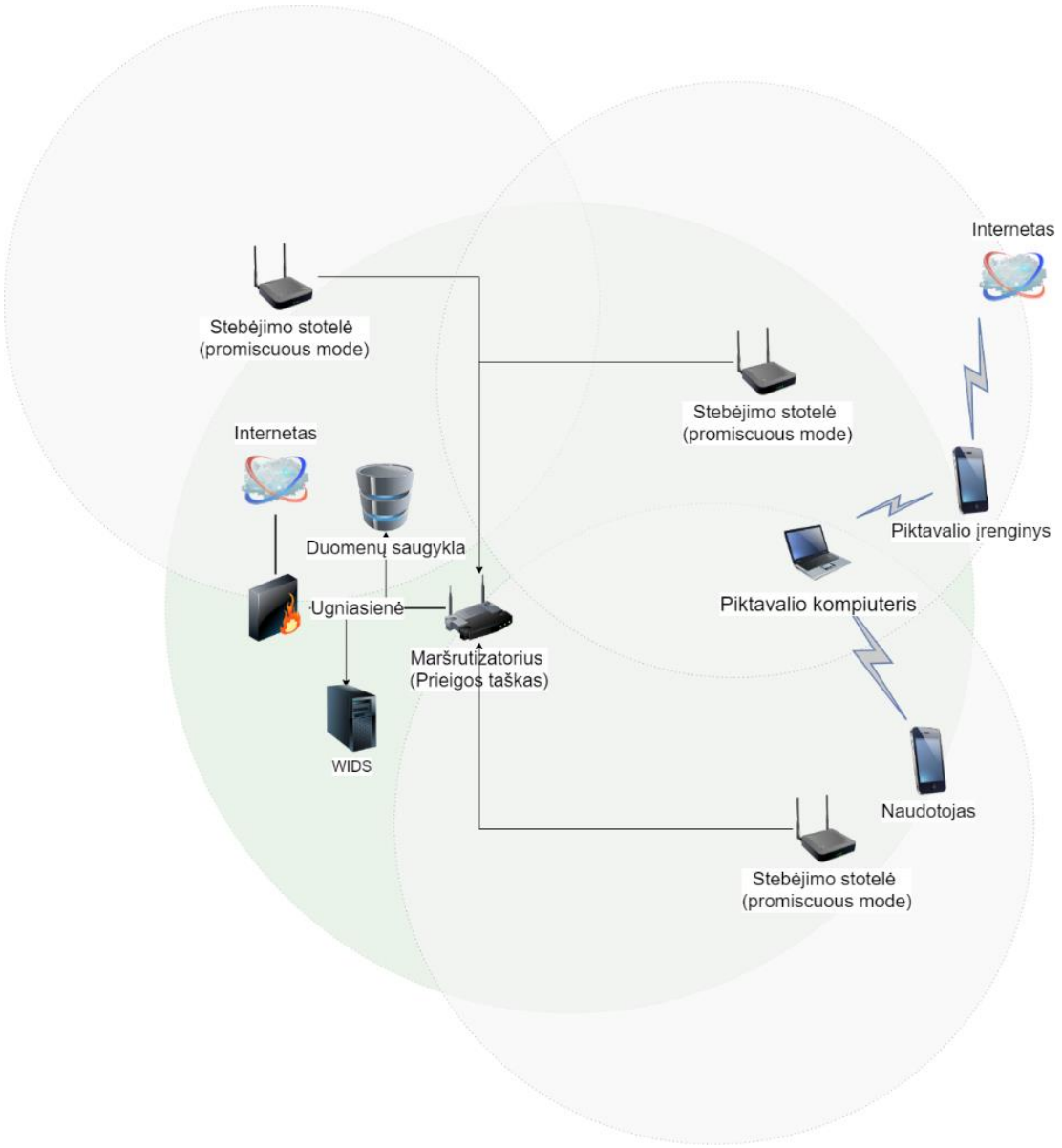
3.2 Belaidžio tinklo atakų aptikimo duomenų rinkinio sudarymas

Sudaromas duomenų rinkinys skirsis nuo jau sudarytų ir viešai prieinamų duomenų rinkinių tuo, kad kadrai bus renkami erdvėje su papildomomis stotelėmis, nes esami duomenų rinkiniai surinkti galiniame taške. Įprastą vartotojo prisijungimą galima matyti 3.1 pav.



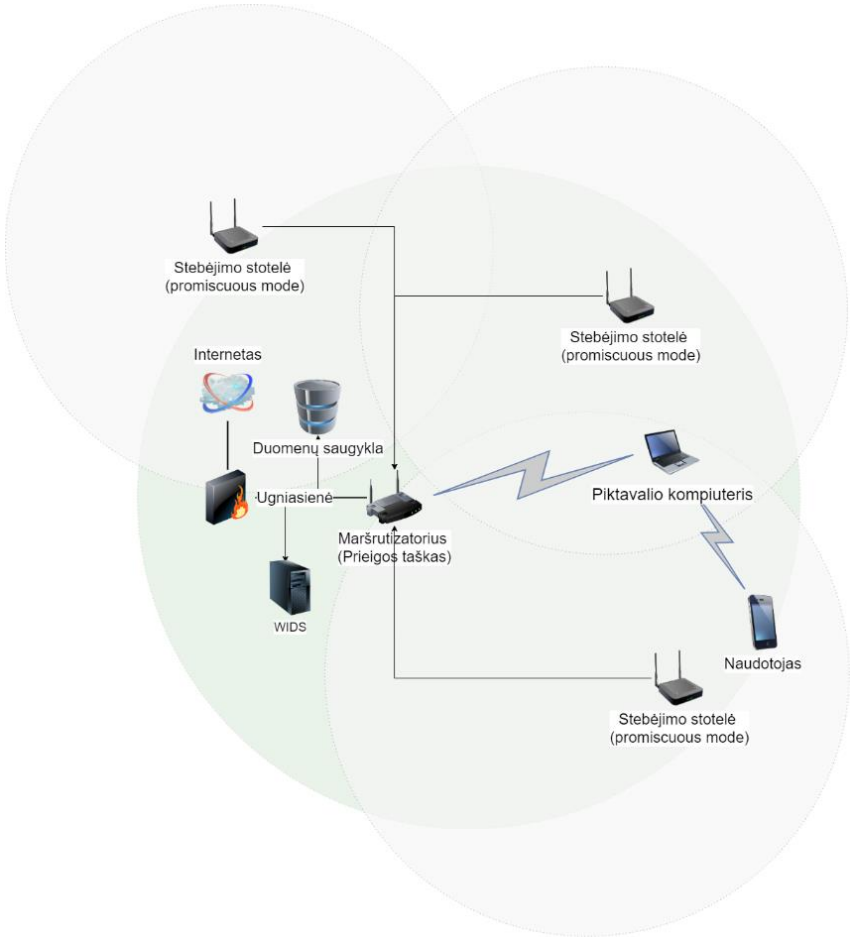
3.1 pav. Įprastas vartotojo prisijungimas prie belaidžio tinklo

Po įvykdytos *mitm* atakos schema atrodytų taip – 3.2 pav. Kai srautas perimamas ir naudojamasi savo prieiga prie interneto.



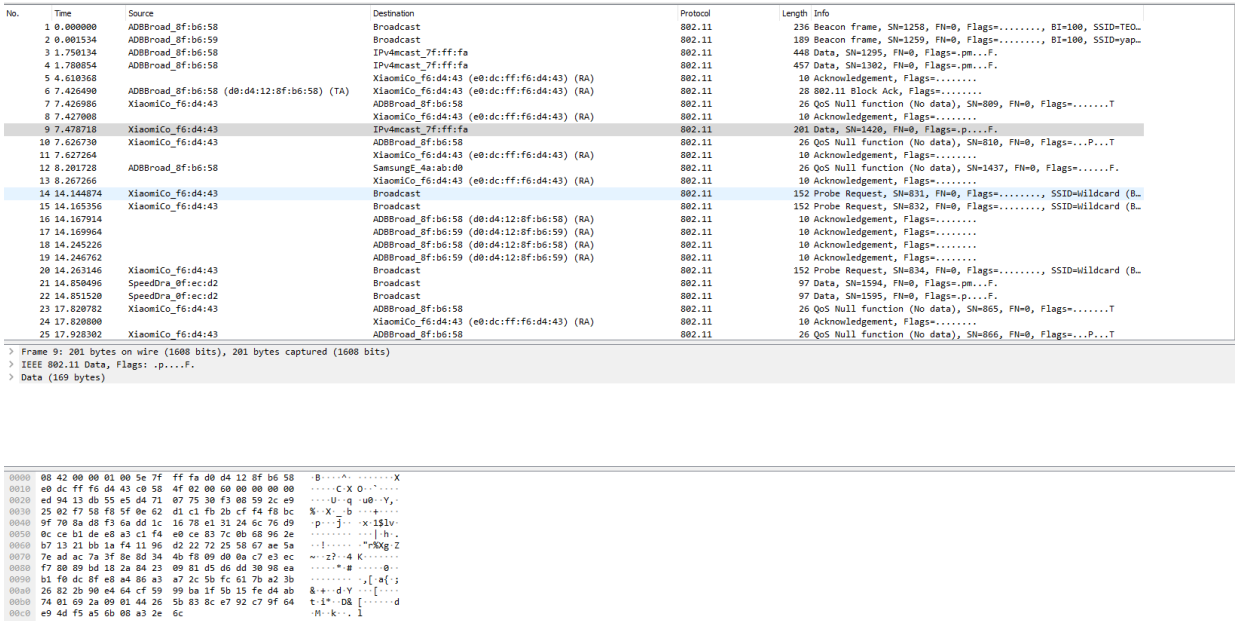
3.2 pav. Piktavalio įvykdyta ataka, kai piktavalis turi savo interneto šaltinį

Dar galimas atakos įvykdymo rezultatas, kai sėkmingai įvykdoma *mitm* ataka, piktavalis prisijungia prie mūsų tinklo, pateikiamas 3.3 pav.



3.3 pav. Piktavalio įvykdyta ataka tarp legalaus tinklo ir naudotojo

Wirespy įrankio pagalba pavyko automatizuoti *EvilTwin* ataką, tokiu būdu perimti srautą ir esamomis stebėjimo stotelėmis sugauti tuos kadrus. Surinktų kadrų vaizdą galima matyti 3.4 pav.



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	ADBBroad_8f:b6:58	Broadcast	802.11	236	Beacon frame, SN=1258, FN=0, Flags=........, BI=100, SSID=TEO…
2	0.001534	ADBBroad_8f:b6:59	Broadcast	802.11	189	Beacon frame, SN=1259, FN=0, Flags=........, BI=100, SSID=yap…
3	1.750134	ADBBroad_8f:b6:58	IPv4mcast_7f:ff:fa	802.11	448	Data, SN=1295, FN=0, Flags=.pm...F.
4	1.780854	ADBBroad_8f:b6:58	IPv4mcast_7f:ff:fa	802.11	457	Data, SN=1302, FN=0, Flags=.pm...F.
5	4.610368		XiaomiCo_f6:d4:43 (e0:dc:ff:f6:d4:43) (RA)	802.11	10	Acknowledgement, Flags=........
6	7.426490	ADBBroad_8f:b6:58 (d0:d4:12:8f:b6:58) (TA)	XiaomiCo_f6:d4:43 (e0:dc:ff:f6:d4:43) (RA)	802.11	28	802.11 Block Ack, Flags=........
7	7.426986	XiaomiCo_f6:d4:43	ADBBroad_8f:b6:58	802.11	26	QoS Null function (No data), SN=809, FN=0, Flags=.......T
8	7.427008		XiaomiCo_f6:d4:43 (e0:dc:ff:f6:d4:43) (RA)	802.11	10	Acknowledgement, Flags=........
9	7.478718	XiaomiCo_f6:d4:43	IPv4mcast_7f:ff:fa	802.11	201	Data, SN=1420, FN=0, Flags=.p....F.
10	7.626730	XiaomiCo_f6:d4:43	ADBBroad_8f:b6:58	802.11	26	QoS Null function (No data), SN=810, FN=0, Flags=...P...T
11	7.627264		XiaomiCo_f6:d4:43 (e0:dc:ff:f6:d4:43) (RA)	802.11	10	Acknowledgement, Flags=........
12	8.201728	ADBBroad_8f:b6:58	SamsungE_4a:ab:d0	802.11	26	QoS Null function (No data), SN=1437, FN=0, Flags=......F.
13	8.267266		XiaomiCo_f6:d4:43 (e0:dc:ff:f6:d4:43) (RA)	802.11	10	Acknowledgement, Flags=........
14	14.144874	XiaomiCo_f6:d4:43	Broadcast	802.11	152	Probe Request, SN=831, FN=0, Flags=........, SSID=Wildcard (B…
15	14.165356	XiaomiCo_f6:d4:43	Broadcast	802.11	152	Probe Request, SN=832, FN=0, Flags=........, SSID=Wildcard (B…
16	14.167914		ADBBroad_8f:b6:58 (d0:d4:12:8f:b6:58) (RA)	802.11	10	Acknowledgement, Flags=........
17	14.169964		ADBBroad_8f:b6:59 (d0:d4:12:8f:b6:59) (RA)	802.11	10	Acknowledgement, Flags=........
18	14.245226		ADBBroad_8f:b6:58 (d0:d4:12:8f:b6:58) (RA)	802.11	10	Acknowledgement, Flags=........
19	14.246762		ADBBroad_8f:b6:59 (d0:d4:12:8f:b6:59) (RA)	802.11	10	Acknowledgement, Flags=........
20	14.263146	XiaomiCo_f6:d4:43	Broadcast	802.11	152	Probe Request, SN=834, FN=0, Flags=........, SSID=Wildcard (B…
21	14.850496	SpeedDra_0f:ec:d2	Broadcast	802.11	97	Data, SN=1594, FN=0, Flags=.pm...F.
22	14.851520	SpeedDra_0f:ec:d2	Broadcast	802.11	97	Data, SN=1595, FN=0, Flags=.p....F.
23	17.820782	XiaomiCo_f6:d4:43	ADBBroad_8f:b6:58	802.11	26	QoS Null function (No data), SN=865, FN=0, Flags=.......T
24	17.820800		XiaomiCo_f6:d4:43 (e0:dc:ff:f6:d4:43) (RA)	802.11	10	Acknowledgement, Flags=........
25	17.928302	XiaomiCo_f6:d4:43	ADBBroad_8f:b6:58	802.11	26	QoS Null function (No data), SN=866, FN=0, Flags=...P...T

```
> Frame 9: 201 bytes on wire (1608 bits), 201 bytes captured (1608 bits)
> IEEE 802.11 Data, Flags: .p....F.
> Data (169 bytes)
```

```
0000  08 42 00 00 01 00 5e 7f  ff fa d0 d4 12 8f b6 58   ·B····^· ·······X
0010  e0 dc ff f6 d4 43 c0 58  4f 02 00 60 00 00 00 00   ·····C·X O··`····
0020  ed 94 13 db 55 e5 d4 71  07 75 30 f3 08 59 2c e9   ····U··q ·u0··Y,·
0030  25 02 f7 58 f8 5f 0e 62  d1 c1 fb 2b cf f4 f8 bc   %··X·_·b ···+····
0040  9f 70 8a d8 f3 6a dd 1c  16 78 e1 31 24 6c 76 d9   ·p···j·· ·x·1$lv·
0050  0c ce b1 de e8 a3 c1 f4  e0 ce 83 7c 0b 68 96 2e   ········ ···|·h·.
0060  b7 13 21 bb 1a f4 11 96  d2 22 72 25 58 67 ae 5a   ··!····· ·"r%Xg·Z
0070  7e ad ac 7a 3f 8e 8d 34  4b f8 09 d0 0a c7 e3 ec   ~··z?··4 K·······
0080  f7 80 89 bd 18 2a 84 23  09 81 d5 d6 dd 30 98 ea   ·····*·# ·····0··
0090  b1 f0 dc 8f e8 a4 86 a3  a7 2c 5b fc 61 7b a2 3b   ········ ·,[·a{·;
00a0  26 82 2b 90 e4 64 cf 59  99 ba 1f 5b 15 fe d4 ab   &·+··d·Y ···[····
00b0  74 01 69 2a 09 01 44 26  5b 83 8c e7 92 c7 9f 64   t·i*··D& [·····d
00c0  e9 4d f5 a5 6b 08 a3 2e  6c                       ·M··k···. l
```

3.4 pav. Surinktų kadrų pavyzdys programoje *Wireshark*

Kadrai surinkti naudojant 3.1 lentelėje aprašyta įrangą, kuri realizuota kaip pavaizduota 3.2 pav.

3.1 lentelė. Kadrų rinkimui naudota įranga

Pavadinimas	**Aprašymas**
Acer nešiojamas kompiuteris	Piktavalio kompiuteris su Kali Linux ir papildomu bevielio tinklo adapteriu, su *Wirespy* programine įranga buvo vykdomos atakos
Xiaomi Mi 9T	Mobilus telefonas su 4G ryšiu, padedantis piktavaliui palaikyti ryšį
Lenovo T6400	Naudotojo nešiojamas kompiuteris
Raspberry Pi 4B	Prieigos taškas, kuris renka kadrus

3.3 Prototipo realizacija

3.3.1 Duomenų rinkinio savybių išskyrimas

Savybių išskyrimo metodais siekiama sumažinti įvesties kintamųjų skaičių iki tų, kurie, manoma, yra naudingiausi modeliui, norint numatyti tikslą. Filtro pasirinkimo metodai naudoja statistinius metodus, kad būtų galima įvertinti kiekvieno įvesties kintamojo ir tikslinio kintamojo santykį, ir šie balai naudojami kaip pagrindas pasirinkti (filtruoti) tuos įvesties kintamuosius, kurie bus naudojami modelyje. Tai regresijos prognozavimo modeliavimo problema su skaitiniais įvesties kintamaisiais.

Pagrindinė normalizavimo idėja visada yra ta pati. Kintamieji, kurie matuojami skirtingomis skalėmis, nevienodai prisideda prie modelio pritaikymo ir gali sukelti šališkumą. Taigi, norint išspręsti šią potencialią problemą, paprastai prieš modelio pritaikymą naudojamas normalizavimas.

Panaudotas įvesties kintamųjų normalizavimo būdas yra *standard scaler*. Tokiu būdu visas savybes standartizuoja pašalinant vidurkį ir išskaido pagal vieneto dispersiją pagal (2) formulę.

$$z = \frac{x-u}{s}; \qquad (2)$$

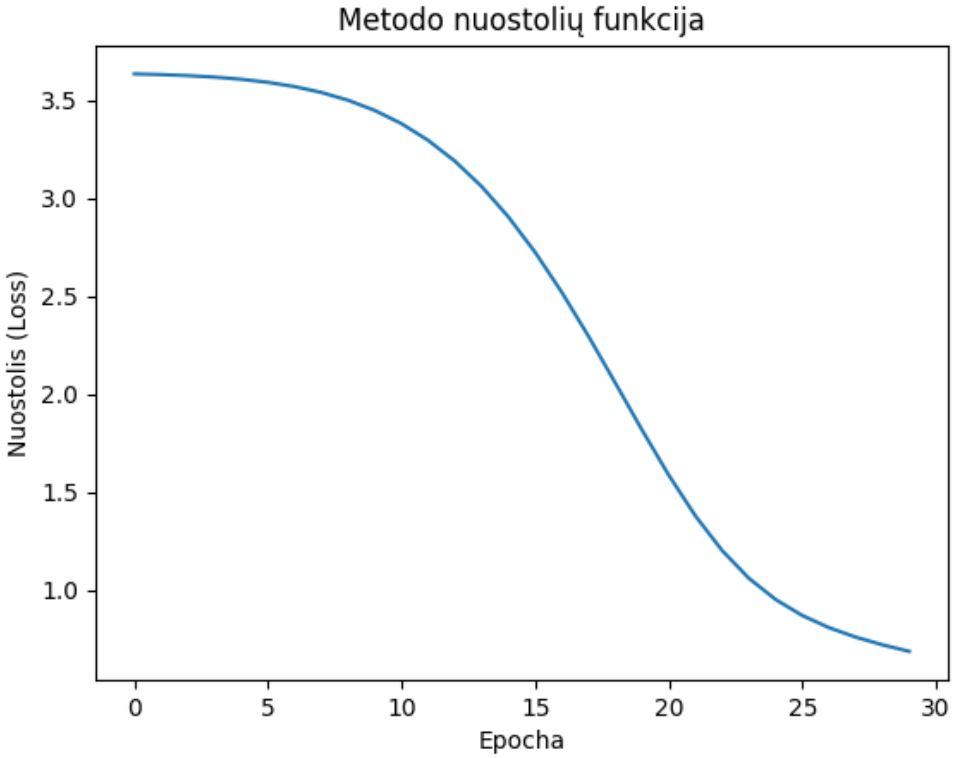
čia *z* yra paskaičiuotas įvertis, *x* yra vertinama reikšmė; *u* yra mokymo rinkinio vidurkis, *s* yra standartinis mokymo rinkinio nuokrypis.

3.3.2 Metodo realizavimas

Mašininis mokymas yra realizuotas Python bibliotekomis *TensorFlow* ir *sklearn*. Duomenų rinkinys apdorotas, panaudojant *StandardScaler* normalizavimo funkciją išskaidyti duomenų savybių reikšmes. Duomenų rinkinys išskaidytas į 70 % apmokymui ir 30 % testavimui.

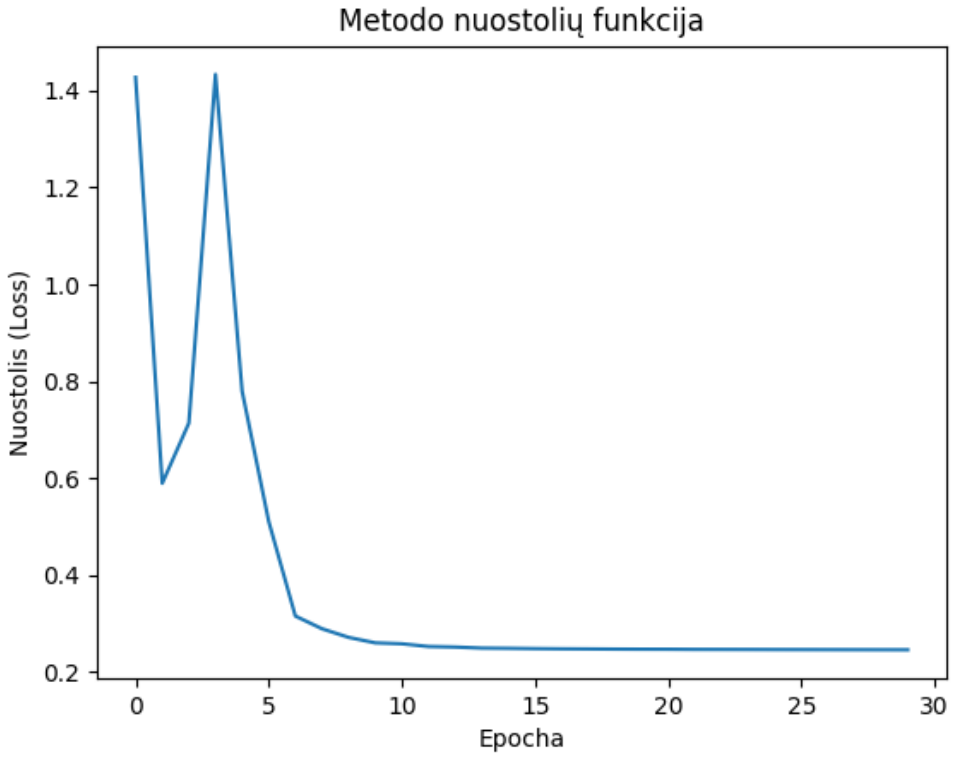
Buvo realizuoti analitinės dalies algoritmai: logistinės regresijos, linijinė diskriminantinė analizė, k-artimiausias kaimynas, sprendimų medžio klasifikatorius, naivusis Bajeso klasifikatorius. Algoritmai realizuoti su rekomenduojamais parametrais.

Realizuotas pasirinktas mašininio mokymo algoritmas. *Adam* optimizatorius realizuotas rekomenduojamais parametrais, išskyrus mokymosi greičio parametrą. Rekomenduojama parametro reikšmė yra 0,001. Panaudojant nuostolių funkciją galime matyti, kaip gerai yra parinkti parametrai. Kai parinktas 0,0005 mokymosi greitis, 3.5 pav. galima matyti, kad žingsnis yra per mažas ir nuostolis lėtai mažėja.



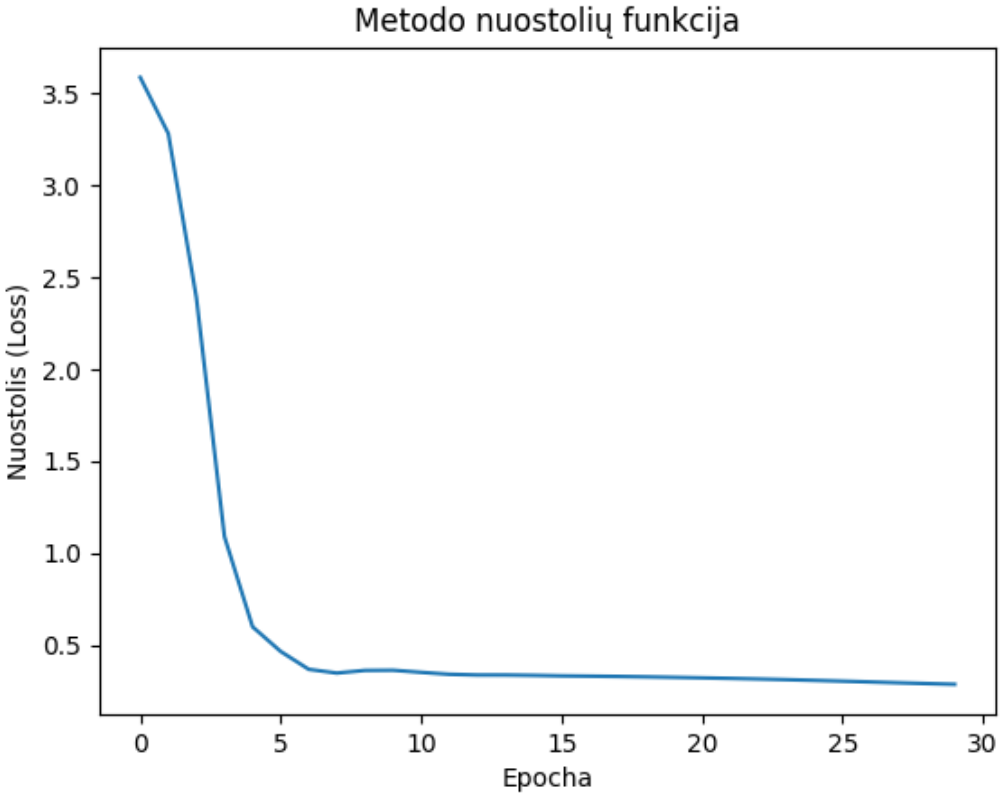
3.5 pav. Nuostolių funkcija, kai mokymosi greitis yra 0,0005

Kai nustatome mokymosi greičio parametrą 0,005, 3.6 pav. matome funkcijos nepastovumą.



3.6 pav. Nuostolių funkcija, kai mokymosi greitis yra 0,005

Nustačius mokymosi greitį 0,001, nuostolio funkcija greičiau ir tolygiau leidžiasi negu su anksčiau pasirinktais parametrais (3.7 pav.).

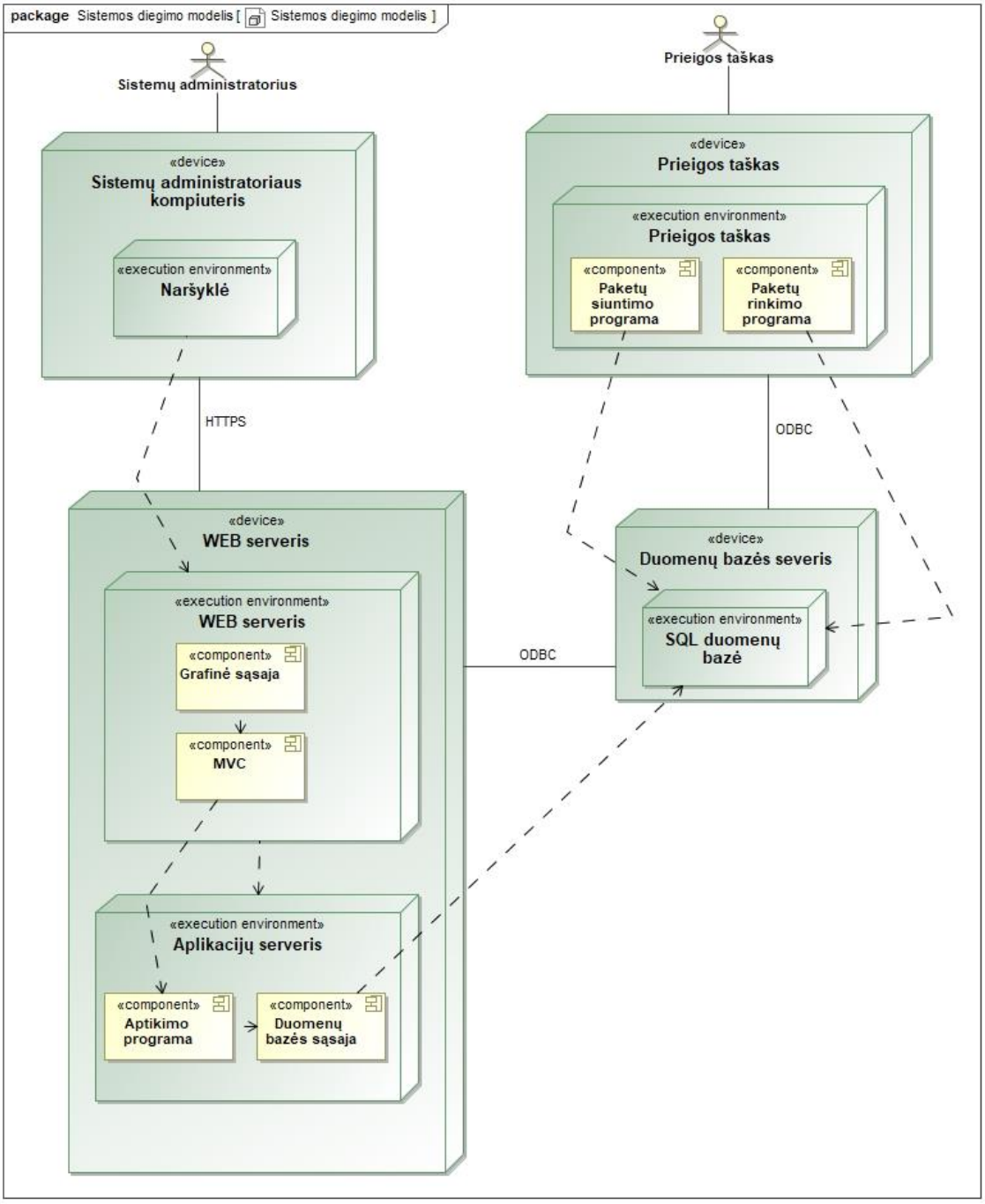


3.7 pav. Nuostolių funkcija, kai mokymosi greitis yra 0,001

Viso tyrimo metu bus naudojamas 0,001 mokymosi greičio parametras.

3.4 Belaidžio tinklo saugos metodo diegimas

Metodas būtų įdiegtas į esamą sistemą, kuri *Paketų rinkimo* komponentę jau turi ir duomenis išsaugo duomenų bazėje. Taip pat šis metodas įeina į *Aptikimo* komponentą, kur ir vyksta atakos ir prieigos taškų aptikimas. Sistema, į kurią būtų diegiamas metodas, atrodytų taip, kaip pavaizduota 3.8 pav.



3.8 pav. Sistemos diegimo modelis

3.5 Išvados

Surinktas duomenų rinkinys, panaudojant programinę ir aparatinę įrangą. Duomenų rinkinyje išskirtos savybės ir rinkinys buvo normalizuotas. Realizuoti analitinėje dalyje aprašyti mašininio mokymo algoritmai, kad būtų galima objektyviai palyginti algoritmus. Realizuotas belaidžio tinklo saugos metodas, kuris remiasi mašininio mokymo *Adam* algoritmu.

4 BELAIDŽIO TINKLO ATAKŲ APTIKIMO TYRIMAS

4.1 Tyrimo tipas

Tyrimo metu bus naudojamas *taikomojo tyrimo tipas,* kuris šiame darbe yra surinkto belaidžio tinklo duomenų rinkinio panaudojimas, kurio pagalba bus galima sudaryti metodą, mašininio mokymo algoritmų pagrindu. Tyrimų metu vadovaujamasi *konstruktyviu tyrimo metodu.*

4.2 Tyrimo metodika

Tyrimo metu bus įvertinami analitinėje dalyje aprašyti mašininio mokymosi (angl. *machine learning*) algoritmai:

1) Logistinės regresijos (LR)
2) Linijinės diskriminantinės analizės (LDA)
3) K-artimiausias kaimynas (KNN)
4) Sprendimų medžio (CART)
5) Naivusis Bajeso klasifikatorius (NB)
6) Adam optimizatorius

Paskutinis Adam algoritmas pasirinktas dėl savo skaičiavimo paprastumo ir mažų atminties reikalavimų. O likusieji panaudoti kaip atskaitos taškas Adam algoritmo palyginimui.

Sekančiuose skyreliuose pateikiami algoritmų parametrų įvertinimai.

4.2.1 Tyrimo aplinka

Testavimas ir eksperimentiniai tyrimai buvo atliekami *Intel Core I7 3770k 3.5 GHz,* 16 *GB* operatyviosios atminties *Windows 10 x64* operacine sistema. Duomenų rinkinys yra iš anksto apdorojamas, kad būtų galima apmokyti mašininio mokymo algoritmus su tuo pačiu tvarkingu duomenų rinkiniu. Duomenų rinkinys su pasirinktomis 20 savybių buvo padalytas į 70 % mokymo ir 30 % bandymų rinkinio.

4.2.2 Algoritmų įvertinimai

Neabejotinas ir išsamus būdas pateikti mašininio mokymosi modelio prognozavimo rezultatus yra naudoti sumaišymo matricą. Tačiau tikslumas gali būti klaidinantis, kai etikečių skaičius nesubalansuotas.

4.1 lentelė. Išmaišymo matricos struktūra

		Spėjimas	
		Teigiamas	**Neigiamas**
Tikras	**Teigiamas**	Tikrai teigiamas *True Positive*: TP	Klaidingai neigiamas *False Negative*: FN
	Neigiamas	Klaidingai teigiamas *False Positive*: FP	Tikrai neigiamas *True Negative*: TN

4.1 lentelėje aprašytoje dvejetainėje išmaišymo matricoje Tikrai teigiamas (TP) nurodo atvejus, kai tikroji etiketė yra teigiama, o modelio spėjimas taip pat teisingai nustatytas kaip teigiamas. Klaidingai neigiamas (FN) nurodo atvejus, kai tikroji etiketė yra teigiama, tačiau modelio spėjimas neteisingai nustatytas. Klaidingai teigiamas (FP) nurodo atvejus, kai tikroji etiketė yra neigiama, tačiau modelio spėjimas neteisingai nustatytas kaip teigiamas. Galiausiai tikrai neigiamas (TN) nurodo atvejus, kai tikroji etiketė yra neigiama, o modelio spėjimas taip pat teisingai nustatytas kaip neigiamas.

Įveikti tikslumo matavimo problemą galima apskaičiuojant papildomus matavimus: formulėje (3) atšaukimo koeficientą $Recall$, formulėje (4) preciziškumą $Precision$, formulėje (5) $f1$ balą ir formulėje (6) tikslumą $Accuracy$, kad įvertintume modelius. Jie apibrėžiami naudojant dvejetainę išmaišymo matricą taip:

$$Recall = \frac{TP}{TP+FP} \quad (3)$$

$$Precision = \frac{TP}{TP+FN} \quad (4)$$

$$f1 = \frac{2*Precision*Recall}{Precision+Recall} \quad (5)$$

$$Accuracy = \frac{TN+TP}{TP+FP+TN+FN}; \quad (6)$$

F1 balas rodo preciziškumo ir aptikimo koeficiento harmoninį vidurkį ir santykinai tiksliai nurodo mašininio mokymosi modelio klasifikavimo rezultatus. Jis išreiškiamas intervalu nuo 0 iki 1, kur geriausia vertė yra 1. Apskaičiuojame visų taikomųjų etikečių aptikimo koeficientus, preciziškumus ir f1 balus, o paskui gauname vieną bendrą f1 balo matavimą, tai yra visų modelių našumo palyginimas. Tikslumas parodo prognozių dalį, kurią modelis nuspėjo teisingai.

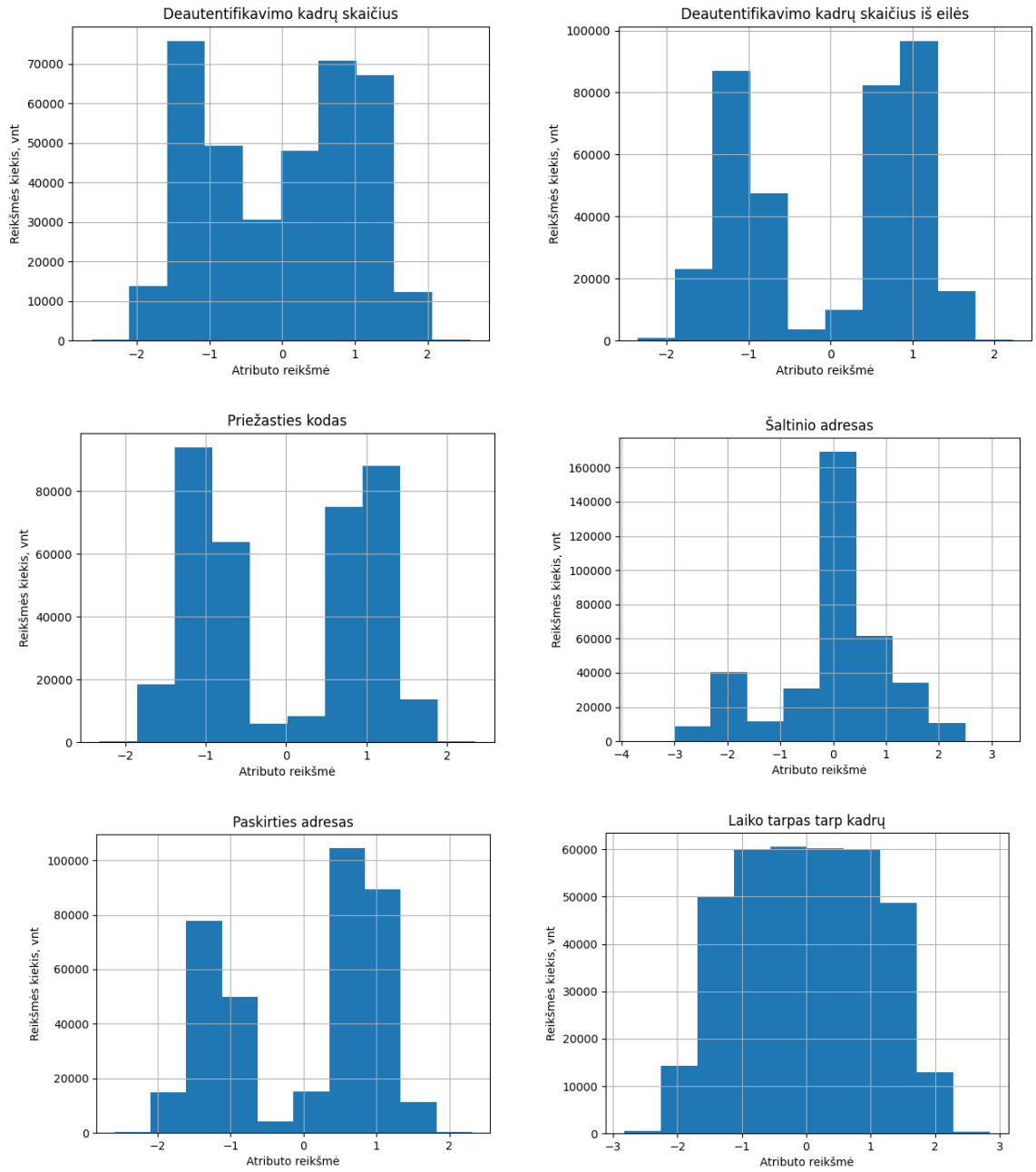
4.2.3 Tyrimo eiga

Tyrimo eigoje nustatomi tiriamų algoritmų įvertinimai ir tarpusavyje palyginami.

4.2.3.1 Išanalizuotų algoritmų palyginimas su surinktu duomenų rinkiniu

Šiame skyrelyje bus analizuojamas surinktas duomenų rinkinys ir pritaikomi algoritmai išanalizuoti analitinėje dalyje. Toliau iš analizuotų algoritmų pagal įvykdymo greitį ir tikslumą bus atrinkti pora algoritmų detalizavimui.

Sekančiame 4.1 paveiksle galima matyti, kaip duomenų rinkinio laukuose yra pasiskirstę duomenys histogramų pavidalu. 4.1Pavaizduotos esminių laukų histogramos. Iš laiko tarp kadrų histogramos galima matyti, kad pasiskirstymas panašus į normalųjį skirstinį. Šaltinio adresą histogramoje galima matyti labai aiškiai, kadangi viena reikšmė viršija kitą beveik 3 kartus. Tikėtina, kad iš to adreso buvo siunčiami deautentifikavimo kadrai, todėl bendras kadrų skaičius yra didesnis.



4.1 pav. Duomenų rinkinio laukų pasiskirstymo histogramos

Duomenų rinkinio histogramų su pasikliautinių intervalų grafiniu vaizdu, sklaidos matricos diagramos ir likusių laukų histogramos yra pridėtos prieduose.

Sekančioje lentelėje 4.2 galima matyti duomenų rinkinio įrašų kiekį bei pasiskirstymą tarp normalaus srauto (57,47 % visų įrašų) ir jau įvykdytos atakos srauto (42,53 % visų įrašų). Visi algoritmai buvo apmokyti 50000 vienetų partijomis.

4.2 lentelė. Duomenų rinkinio įrašų suvestinė

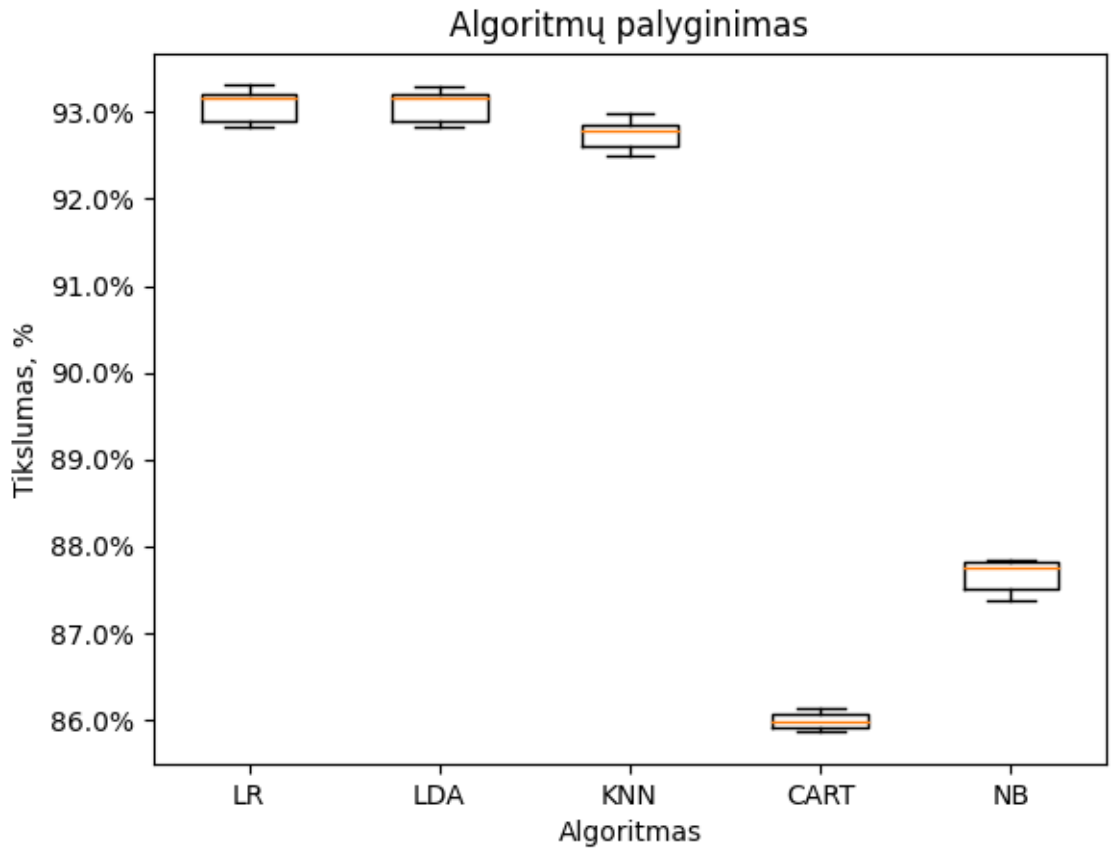
Pavadinimas	**Žymėjimas (etiketė)**	**Kiekis, vnt.**
Normalus srautas	1	211319
Įvykdytos atakos srautas (mitm)	0	156367
Iš viso		367686

Mašininio mokymo algoritmai apmokyti, panaudojant duomenų rinkinį, įvertintas jų tikslumas (angl. *accuracy*) ir algoritmo įvykdymo laikas (4.3 lentelė).

4.3 lentelė. Analitinės dalies algoritmų įvertinimai

Algoritmo pavadinimas	**Tikslumas, %**	**Įvykdymo laikas, s**
Logistinė regresija (LR)	93,05	11,77
Linijinė diskriminantinė analizė (LDA)	93,06	9,72
K-artimiausias kaimynas (KNN)	92,72	1335,06
Sprendimų medis (CART)	85,82	290,21
Naivusis Bajeso klasifikatorius (NB)	87,69	2,19

Didžiausius tikslumus turi logistinės regresijos ir linijinės diskriminantinės analizės algoritmai, kurie viršija 93 %. Ilgiausią algoritmo įvykdymo laiką turi k-artimiausio kaimyno algoritmas, kuris užtruko 1335 sekundžių arba 22,25 minutes. Šie duomenys taip pat atvaizduoti 4.2 pav.



4.2 pav. Algoritmų palyginimo histograma su pasikliautinių intervalų grafiniu vaizdu

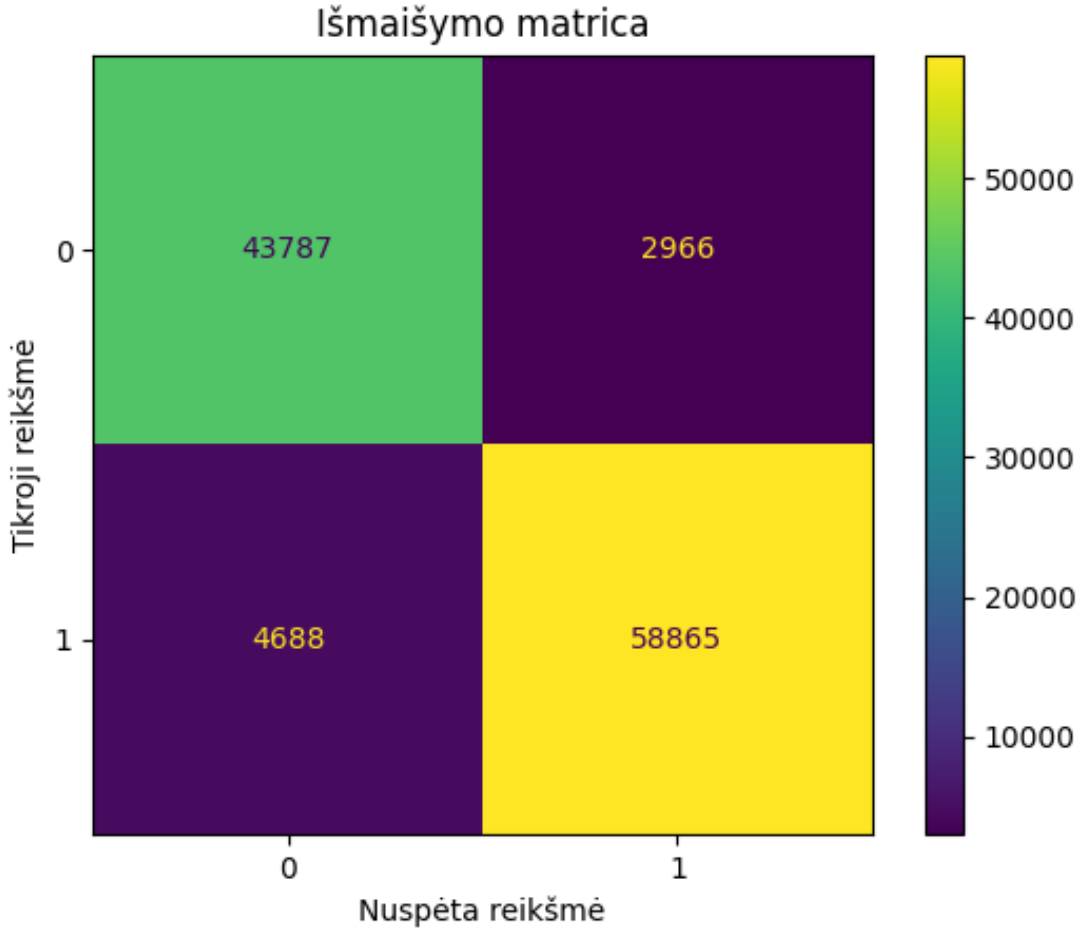
Pagal šiuos duomenis detalizacijai dėl savo mažo įvykdymo laiko ir didesnio tikslumo buvo atitinkamai pasirinkti naivusis Bajeso klasifikatorius ir linijinės diskriminantinės analizės algoritmas.

Sekančioje 4.4 lentelėje pavaizduoti linijinės diskriminantinės analizės algoritmo matavimų įvertinimai ir jų vidurkiai. Matoma, kad skirtingų reikšmių preciziškumas skiriasi 4,87 %, kiti įvertinimai tarp reikšmių tokio skirtumo neturi.

4.4 lentelė. Linijinės diskriminantinės analizės algoritmo įvertinimas

	Tikslumas, %	**Preciziškumas, %**	**Aptikimo koeficientas, %**	**F1 reikšmė, %**
0	93,06	90,33	93,66	91,96
1	93,06	95,20	92,62	93,90
Vidurkis		92,77	93,14	92,93
Svertinis vidurkis		93,14	93,06	93,08

Toliau 4.3 pav. pavaizduota algoritmo išmaišymo matrica, kurioje matosi, kiek kurios reikšmės buvo teisingai nuspėta algoritmo. Galima matyti, kad reikšmės 1, tai yra normalaus tinklo srauto, buvo dažniau klaidingai nuspėtas negu įvykdytos *mitm* atakos tinklo srauto.



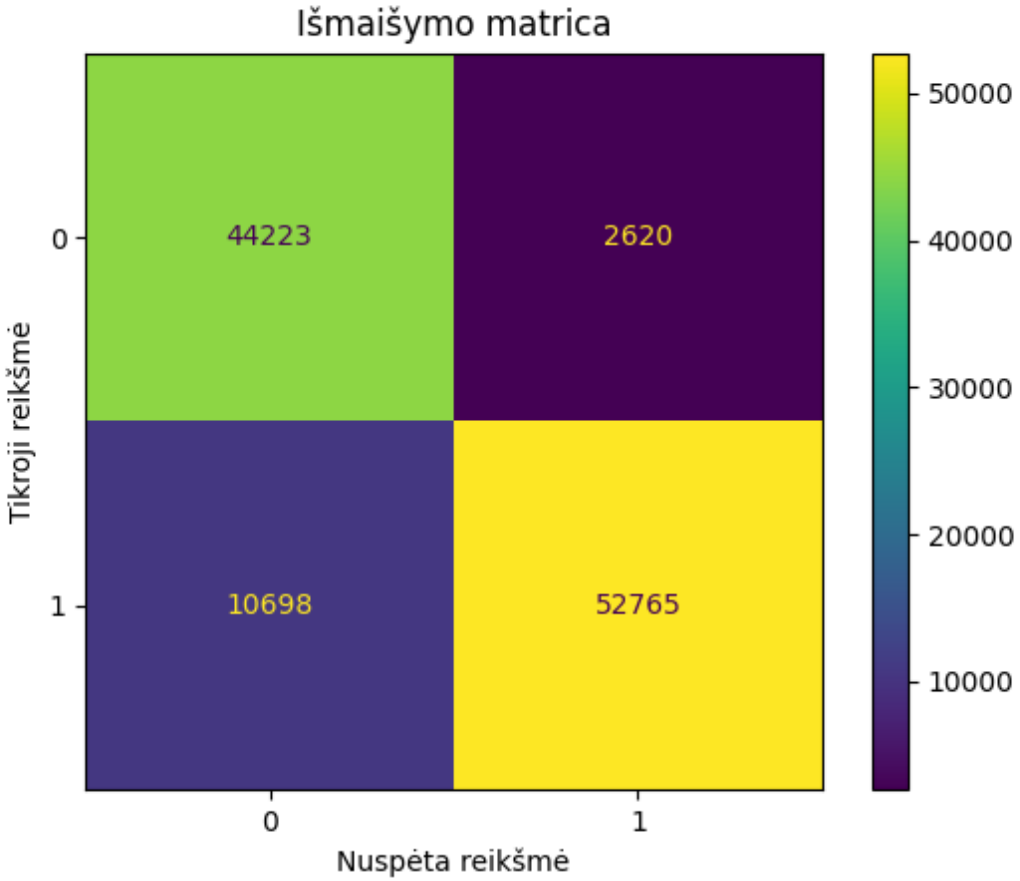
4.3 pav. Linijinės diskriminantinės analizės algoritmo išmaišymo matrica

Toliau pateiktas naiviojo Bajeso klasifikatoriaus matavimų įvertinimas 4.5 lentelėje. Šio algoritmo preciziškumo skirtumas tarp reikšmių nuspėjimo yra 14,75 %. Skirtumas yra 3 kartus didesnis negu linijinės diskriminantinės analizės algoritmo nuspėtų reikšmių. Dar galima matyti, kad aptikimo koeficiento skirtumas tarp nuspėtų reikšmių yra 11,27 %, kai linijinės diskriminantinės analizės algoritmo skirtumas yra tik 1,04 %.

4.5 lentelė. Naiviojo Bajeso klasifikatoriaus įvertinimas

	Tikslumas, %	**Preciziškumas, %**	**Aptikimo koeficientas, %**	**F1 reikšmė, %**
0	87,69	80,52	94,41	86,91
1	87,69	95,27	83,14	88,79
Vidurkis		87,90	88,77	87,85
Svertinis vidurkis		89,01	87,93	88,00

Toliau pateikta naiviojo Bajeso klasifikatoriaus išmaišymo matrica (4.4 pav.), kurioje matosi, kad normalus tinklo srautas buvo 2,28 karto klaidingai nuspėtas negu linijinės diskriminantinės analizės algoritmo.



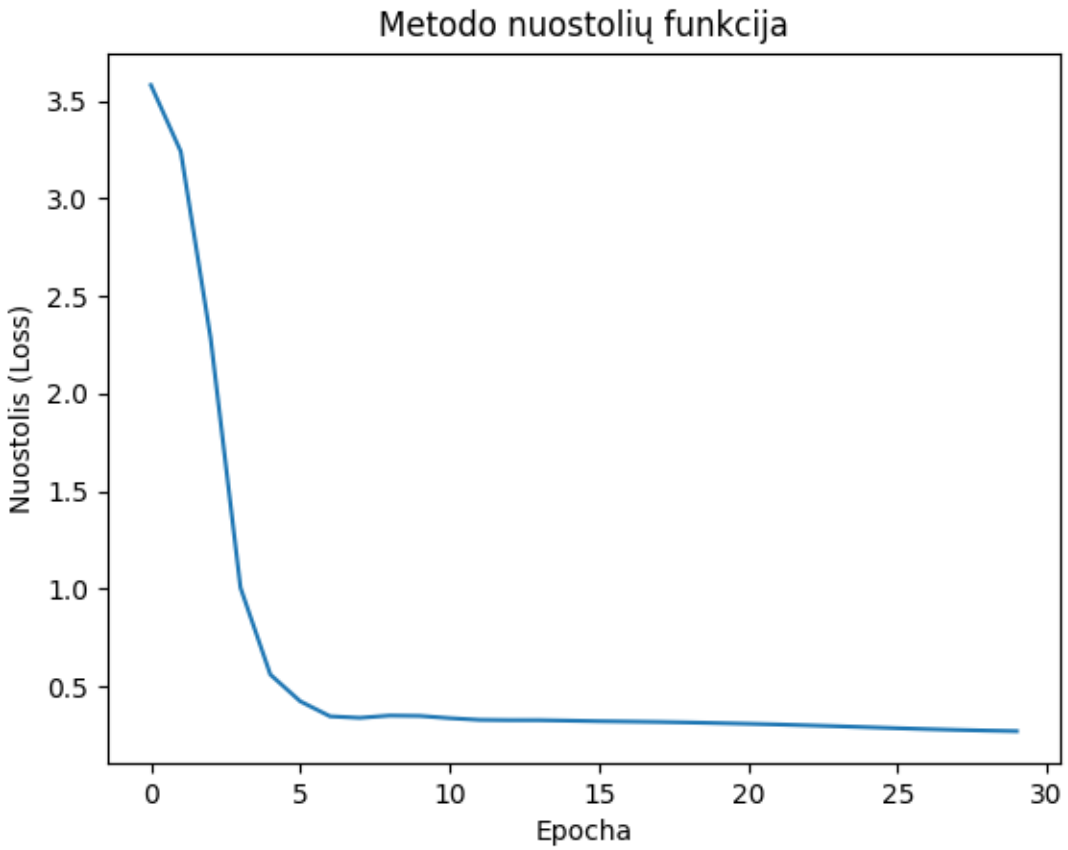
4.4 pav. Naiviojo Bajeso klasifikatoriaus algoritmo išmaišymo matrica

Palyginus algoritmus, linijinės diskriminantinės analizės algoritmas yra vidutiniškai 5 % tikslesnis, bet naivusis Bajeso klasifikatorius yra 4,43 karto greitesnis. Galima priežastis naiviojo Bajeso klasifikatoriaus preciziškumo skirtumo tarp srautų nuspėjimo yra ta, kad buvo naudojamos algoritmo parametrų rekomenduojamos reikšmės.

4.2.3.2 Adam optimizatoriaus tyrimas

Tas pats duomenų rinkinys panaudotas Adam algoritmo apmokymui ir tyrimui. Įvertinti, kaip gerai algoritmas modeliuoja duomenų rinkinį, yra panaudojama nuostolių funkcija. Keičiant mašininio mokymo parametrus, pakitimai bus atvaizduoti nuostolių funkcijoje. Adam algoritmo mokymosi

greitis nustatytas 0,001, kiti parametrai nekeisti, paliktos numatytos reikšmės. 4.5 pav. galima matyti, kad nuostolių funkcijos rezultatas sustoja gerėti nuo 6 epochos.



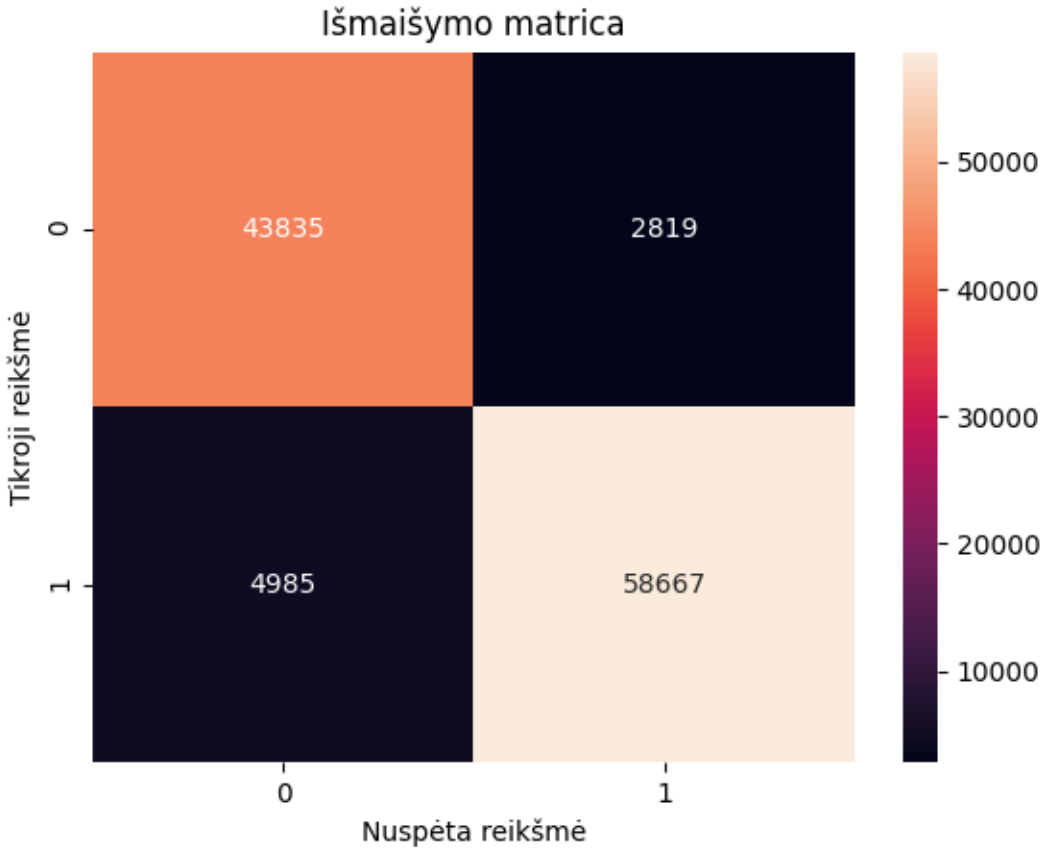
4.5 pav. Adam optimizatoriaus nuostolių funkcija

Vidutinis epochos apsimokymo laikas yra 3,27 sekundės. Adam algoritmo 6-ių epochų įvykdymo laikas yra 19,88 sekundės, o matavimų įvertinimai matosi 4.6 lentelėje. Galima teigti, kad skirtumas tarp preciziškumo nuspėjimo reikšmių yra minimalus – 0,44 %. Tokį patį minimalų skirtumą galima pastebėti ir kituose algoritmų įvertinimų įverčiuose.

4.6 lentelė. Adam algoritmo įvertinimai su didelėmis partijomis

	Tikslumas, %	**Preciziškumas, %**	**Aptikimo koeficientas, %**	**F1 reikšmė, %**
0	92,96	92,60	93,06	92,79
1	92,96	93,04	92,93	92,94
Vidurkis		92,82	92,99	92,87
Svertinis vidurkis		92,04	93,00	92,83

4.6 pav. išmaišymo matricoje galima matyti Adam optimizatoriaus nuspėtų reikšmių pasiskirstymą.



4.6 pav. Adam algoritmo išmaišymo matrica

Adam algoritmo efektyvumas sutampa su linijinės diskriminantinės analizės ir logistinės regresijos algoritmų efektyvumais, tačiau įvykdymo laikas yra atitinkamai 0,48 ir 0,59 karto lėtesnis.

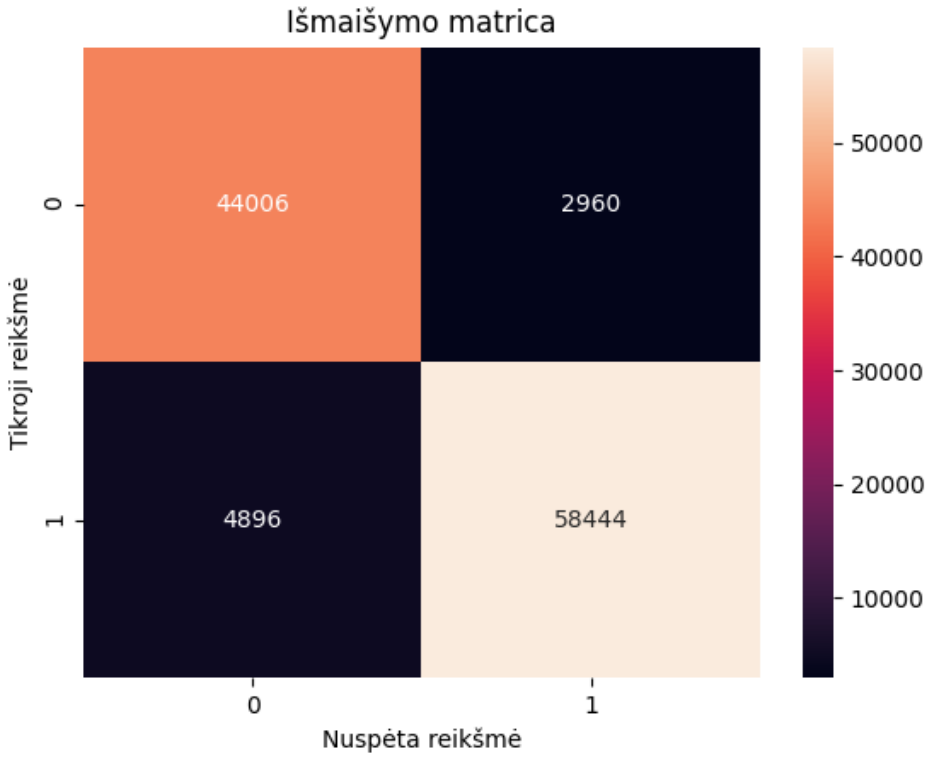
Toliau pateikta, kai Adam algoritmui yra pateikiamas duomenų rinkinys mažomis partijomis, lyg būtų realioje sistemoje. Kiekvieno partijoje yra 20 įrašų.

4.7 lentelė. Adam algoritmo įvertinimai su mažomis partijomis

	Tikslumas, %	**Preciziškumas, %**	**Aptikimo koeficientas, %**	**F1 reikšmė, %**
0	92,67	92,37	92,80	92,55
1	92,67	92,78	92,67	92,69
Vidurkis		92,58	92,74	92,62
Svertinis vidurkis		92.55	92,77	92,60

Iš 4.7 lentelės galima matyti, kad šiuo atveju tikslumas yra 0,29 % mažesnis negu tuomet, kai algoritmui pateikiamos 50000 vienetų partijos.

Toliau 4.7 pav. pateikiama išmaišymo matrica, kai algoritmui pateikiamos mažos partijos. Ir galima teigti, kad nuspėjimo reikšmės minimaliai skiriasi, lyginant su tuo atveju, kai yra pateikiamos didelės partijos.



4.7 pav. Adam algoritmo išmaišymo matrica su mažomis partijomis

4.8 lentelėje visi ištirti algoritmai išrikiuoti tikslumo mažėjimo tvarka.

4.8 lentelė. Visų ištirtų algoritmų tikslumai

Algoritmo pavadinimas	**Tikslumas, %**	**Įvykdymo laikas, s**
Linijinė diskriminantinė analizė (LDA)	93,06	9,72
Logistinė regresija (LR)	93,05	11,77
Adam optimizatorius (didelė partija)	92,96	19,88
K-artimiausias kaimynas (KNN)	92,72	1335,06
Adam optimizatorius (maža partija)	92,67	19,02
Naivusis Bajeso klasifikatorius (NB)	87,69	2,19
Sprendimų medis (CART)	85,82	290,21

4.3 Tyrimo išvados

Buvo ištirti 6 mašininio mokymo algoritmai ir 3 algoritmai išdetalizuoti. Minėti logistinės regresijos, linijinės diskriminantinės analizės, k-artimiausio kaimyno, sprendimų medžio ir naiviojo Bajeso klasifikatoriaus algoritmai buvo apmokyti surinktu duomenų rinkiniu, kad būtų galima įvertinti Adam optimizatoriaus veikimą.

Paaiškėjo, kad Adam optimizatoriaus tikslumas yra labai panašus į linijinės diskriminantinės analizės ir logistinės regresijos algoritmų įvertinimus, kurie yra apie 93 %, tačiau įvykdymo laikas yra lėtesnis. Taip pat, Adam optimizatoriui pateikus didelę 50000 vienetų partiją ir mažą 20 vienetų partiją, algoritmo tikslumas pakito minimaliai 0,44 %.

Galima teigti, kad Adam optimizatorius tiktų *mitm* deautentifikavimo atakų aptikimui.

GALUTINĖS DARBO IŠVADOS

1) Atlikus analizę, paaiškėjo, kad belaidžiai IEEE 802.11 standarto tinklai yra pažeidžiami atakų, po kurių sėkmingo įvykdymo yra perimamas belaidžio tinklo srautas. Esami aptikimo metodai neužtikrina sėkmingo atakų aptikimo. Pagerinti atakų aptikimo rezultatus galima papildomai panaudojus mašininio mokymo algoritmus pagal įvykdytos atakos savybes.

2) Sukurtas metodas, kuris susideda iš kadrų rinkimo erdvėje, siekiant sudaryti duomenų rinkinį. Tas duomenų rinkinys apdorojamas ir normalizuojamas su išskirtomis savybėmis. Apmokytas Adam optimizatoriaus algoritmas suskirsto duomenų rinkinį į normalų srautą arba įvykdytos atakos srautą.

3) Pagal sukurtą metodą realizuotas prototipas. Duomenų rinkinio sudarymas realizuotas sudarytame belaidžiame tinkle su papildoma aparatine ir programine įranga atakų vykdymui. Išskirtos duomenų rinkinio savybės ir rinkinys normalizuotas. Be Adam optimizatoriaus realizuoti ir išanalizuoti mašininio mokymo algoritmai: logistinės regresijos, linijinės diskriminantinės analizės, k-artimiausias kaimynas, sprendimų medžio klasifikatorius, naivusis Bajeso klasifikatorius. Kiti mašininio mokymo algoritmai realizuoti, kad būtų galima objektyviai palyginti Adam optimizatoriaus algoritmą su surinktu duomenų rinkiniu.

4) Realizuoti ir ištirti mašininio mokymo algoritmai.

   a) Logistinės regresijos algoritmo tikslumas yra 93,05 % ir įvykdymo laikas 11,77 s.

   b) Linijinės diskriminantinės analizės algoritmo tikslumas yra 93,06 % ir įvykdymo laikas 9,72 s.

   c) K-artimiausio kaimyno algoritmo tikslumas yra 92,72 % ir įvykdymo laikas 1335,06 s.

   d) Sprendimų medžio klasifikatoriaus tikslumas yra 85,82 % ir įvykdymo laikas 290,21 s.

   e) Naiviojo Bajeso klasifikatoriaus tikslumas yra 87,69 % ir įvykdymo laikas 2,19 s.

   f) Adam optimizatoriaus algoritmo tikslumas yra 92,96 %, o įvykdymo laikas 19,88 s.

5) Adam optimizatoriaus tikslumas yra labai panašus į logistinės regresijos ir linijinės diskriminantinės analizės algoritmų įvertinimus, kurie yra apie 93 %, tačiau įvykdymo laikas yra lėtesnis. Nepriklausomai nuo partijos dydžio algoritmo aptikimo tikslumas panašus. Adam optimizatorius tiktų įvykdytų *man in the middle* deautentifikavimo atakų aptikimui.

LITERATŪRA

1. WI-FI ALLIANCE: *Who We Are* [interaktyvus]. [žiūrėta 2019-12-12]. Prieiga per internetą: <https://www.wi-fi.org/who-we-are>.

2. ARXIV: *A Review of Man-in-the-Middle Attacks* [interaktyvus]. 2015. [žiūrėta 2019-11-25]. Prieiga per internetą: <https://arxiv.org/abs/1504.02115>.

3. CONTI, M. ir kt. A Survey of Man In The Middle Attacks. In *IEEE Communications Surveys Tutorials* . 2016. Vol. 18, no. 3, p. 2027–2051.

4. SYACH, W. The Dangers of Deauthentication Attacks in an Increasingly Wireless World. In [interaktyvus]. [žiūrėta 2021-04-06]. Prieiga per internetą: <https://www.academia.edu/6206658/The_Dangers_of_Deauthentication_Attacks_in_an_Increasingly_Wireless_World>.

5. SHIMEALL, T. SPRING, J. Science Direct Topics. In *Rogue Access Point - an overview* [interaktyvus]. 2014. [žiūrėta 2019-12-07]. Prieiga per internetą: <https://www.sciencedirect.com/topics/computer-science/rogue-access-point>.

6. AL-SHOURBAJI, I. AL-JANABI, S. Intrusion Detection and Prevention Systems in Wireless Networks. In *Kurdistan Journal of Applied Research* . 2017. Vol. 2, no. 3, p. 267–272.

7. CISCO Cisco Wireless Intrusion Prevention System [Interaktyvus]. [žiūrėta 2020-01-22]. Prieiga per internetą: <https://www.cisco.com/c/en/us/products/wireless/index.html>.

8. ARUBA: *Hybrid WIDS* [Interaktyvus]. [žiūrėta 2020-01-22]. Prieiga per internetą: <https://www.arubanetworks.com/pdf/technology/whitepapers/wp_Hybrid_WIDS.pdf>.

9. AIRCRACK-NG aircrack-ng/OpenWIPS-ng. [interaktyvus]. 2020. [žiūrėta 2020-01-24]. Prieiga per internetą: <https://github.com/aircrack-ng/OpenWIPS-ng>.

10. OPENWRT OpenWrt Project. In *Welcome to the OpenWrt Project* [interaktyvus]. [žiūrėta 2020-01-22]. Prieiga per internetą: <https://openwrt.org/>.

11. JUŠKEVIČIUS, S. *Neautorizuotų prieigos taškų (Rogue AP) bevieliame tinkle kontrolės sistema*. : Kauno technologijos universitetas Prieiga per eLABa – nacionalinė Lietuvos akademinė elektroninė biblioteka, 2019.

12. JUŠKEVIČIUS, S. RIMKUS, D. Unauthorized access points (rogue ap) in wireless network control system. In *Entrepreneurship and Regional Development, IECOTERD* [interaktyvus]. 2019. no. 3. [žiūrėta 2021-05-12]. Prieiga per internetą: <http://ojs.kaunokolegija.lt/index.php/ITE/article/view/338>.

13. XIN, Y. ir kt. Machine Learning and Deep Learning Methods for Cybersecurity. In *IEEE Access* . 2018. Vol. 6, p. 35365–35381.

14. KOTPALLIWAR, M.V. WAJGI, R. Classification of Attacks Using Support Vector Machine (SVM) on KDDCUP'99 IDS Database. In *2015 Fifth International Conference on Communication Systems and Network Technologies* . 2015. p. 987–990.

15. SAXENA, H. RICHARIYA, V. Intrusion Detection in KDD99 Dataset using SVM-PSO and Feature Reduction with Information Gain. In *International Journal of Computer Applications (0975*

– *8887)* [interaktyvus]. 2014. Vol. 98. Prieiga per internetą: <https://research.ijcaonline.org/volume98/number6/pxc3897369.pdf>.

16. SHARIFI, A. ir kt. Intrusion Detection Based on Joint of K-Means and KNN. In *Journal of Convergence Information Technology(JCIT)* . 2015. Vol. 10, p. 42–51.

17. MALIK, A.J. KHAN, F.A. A hybrid technique using binary particle swarm optimization and decision tree pruning for network intrusion detection. In *Cluster Computing* . 2018. Vol. 21, no. 1, p. 667–680.

18. INGRE, B. ir kt. Decision Tree Based Intrusion Detection System for NSL-KDD Dataset. In *Smart Innovation* . 2017.

19. RELAN, N. PATIL, D. Implementation of network intrusion detection system using variant of decision tree algorithm. In *2015 International Conference on Nascent Technologies in the Engineering Field, ICNTE 2015 - Proceedings* . 2015.

20. GAO, N. ir kt. An Intrusion Detection Model Based on Deep Belief Networks. In *2014 Second International Conference on Advanced Cloud and Big Data* . 2014. p. 247–252.

21. ALRAWASHDEH, K. PURDY, C. Toward an Online Anomaly Intrusion Detection System Based on Deep Learning. In *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)* . 2016. p. 195–200.

22. A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks - IEEE Journals & Magazine. In [interaktyvus]. [žiūrėta 2020-01-13]. Prieiga per internetą: <https://ieeexplore.ieee.org/document/8066291>.

23. LE, T.-T.-H. - KIM, J. An Effective Intrusion Detection Classifier Using Long Short-Term Memory with Gradient Descent Optimization. In . 2017. p. 1–6.

24. KOLOSNJAJI, B. ir kt. Empowering convolutional networks for malware classification and analysis. In *2017 International Joint Conference on Neural Networks (IJCNN)* . 2017. p. 3838–3845.

25. CALVELLO, M. 50 VPN Statistics That Will Make You Reconsider Your Security. In [interaktyvus]. [žiūrėta 2020-11-24]. Prieiga per internetą: <https://learn.g2.com/vpn-statistics>.

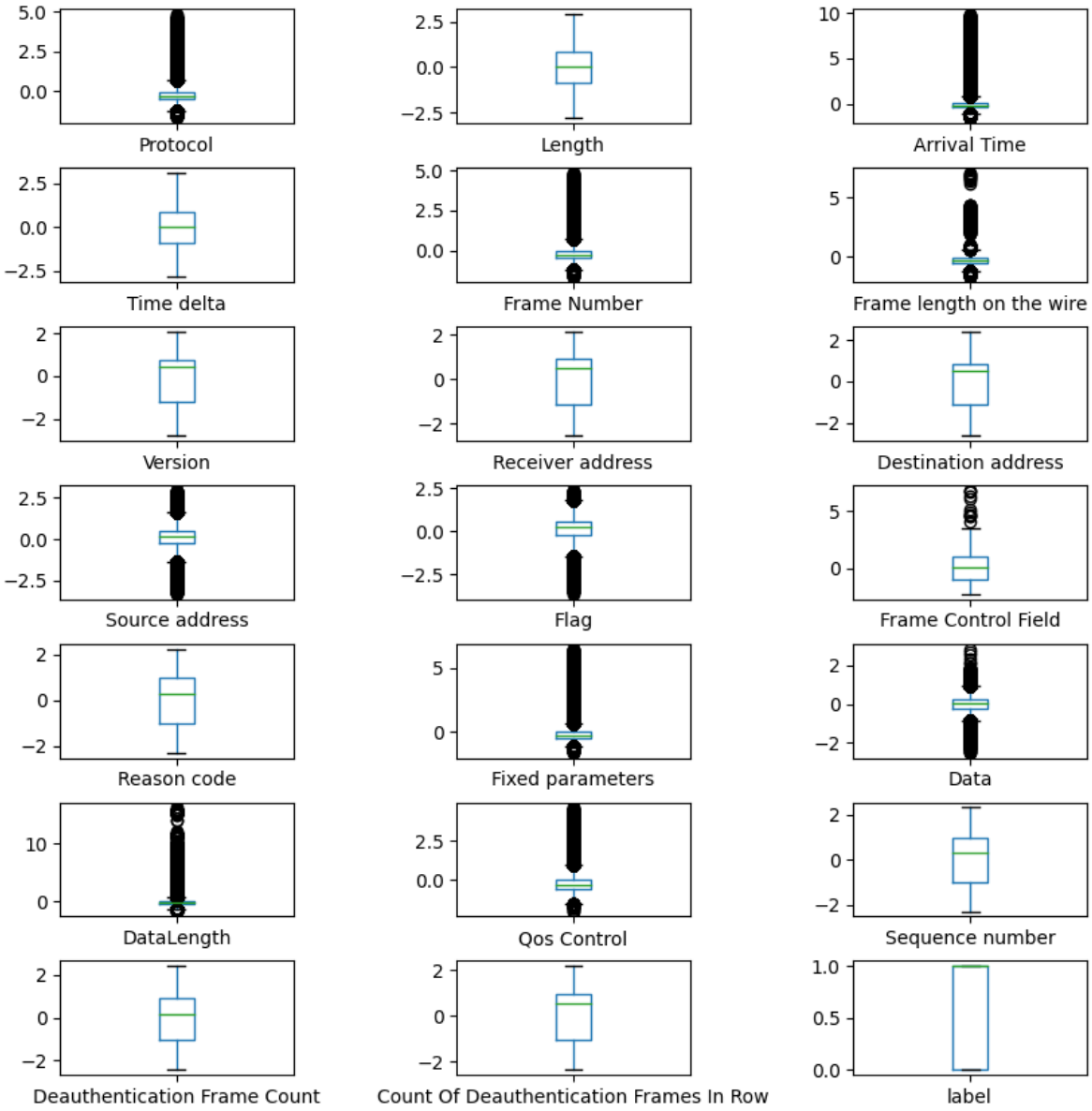
26. DATAPROT VPN statistics for 2020 - Keeping internet privacy alive. In *DataProt* [interaktyvus]. 2019. [žiūrėta 2020-11-24]. Prieiga per internetą: <https://dataprot.net/statistics/vpn-statistics/>.

PRIEDAI

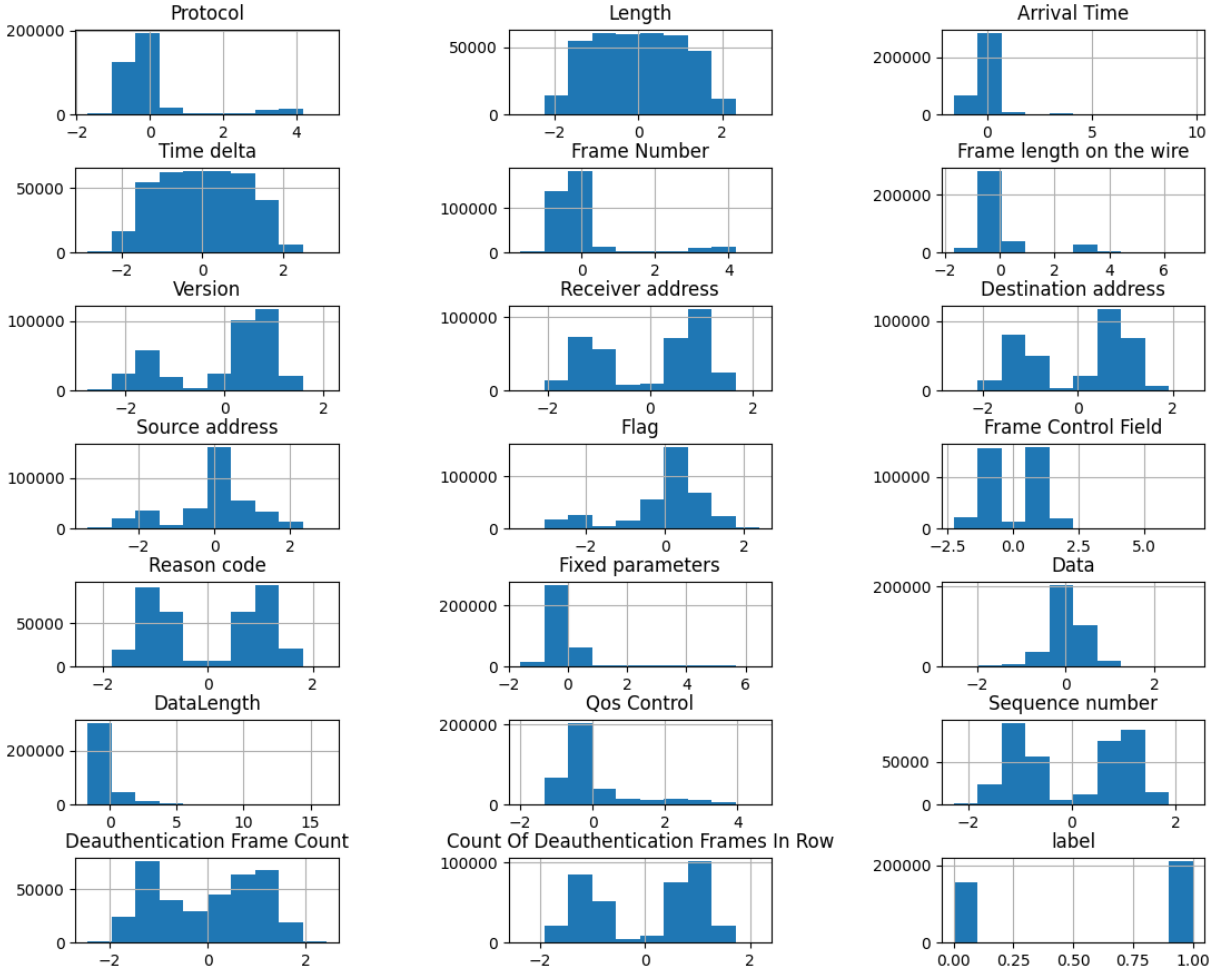
1 priedas. IECOTERD pristatytas recenzuotas straipsnis apie sistemą kurioje realizuotas prototipas

Dėl magistro baigiamojo darbo sutapties patikros straipsnis nepridėtas. Prieiga per internetą: <http://ojs.kaunokolegija.lt/index.php/ITE/article/view/338>.

2 priedas. Duomenų rinkinio palyginimo histogramos su pasikliautinių intervalų grafiniu vaizdu



3 priedas. Duomenų rinkinio laukų pasiskirstymo histogramos



4 priedas. Duomenų rinkinio išmėtymo matricos