VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS
FUNDAMENTINIŲ MOKSLŲ FAKULTETAS
INFORMACINIŲ SISTEMŲ KATEDRA

Dmitrij Kollerov

DDOS ATAKŲ APTIKIMAS NEURONINIU TINKLU
DDOS ATTACKS DETECTION USING NEURAL NETWORK

Baigiamasis magistro darbas

Informacijos ir informacinių technologijų saugos studijų programa, valstybinis kodas 621E14007
Informatikos inžinerijos studijų kryptis

Vadovas prof. dr. Dalius Mažeika

Vilnius, 2015

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS
FUNDAMENTINIŲ MOKSLŲ FAKULTETAS
INFORMACINIŲ SISTEMŲ KATEDRA

TVIRTINU
Katedros vedėjas

_____
(Parašas)

_____
(Vardas, pavardė)

_____
(Data)

Dmitrij Kollerov

DDOS ATAKŲ APTIKIMAS NEURONINIU TINKLU
DDOS ATTACKS DETECTION USING NEURAL NETWORK

Baigiamasis magistro darbas

Informacijos ir informacinių technologijų saugos studijų programa, valstybinis kodas 621E14007
Informatikos inžinerijos studijų kryptis

Vadovas _prof. dr. Dalius Mažeika_ _____ _____
(Moksl. laipsnis, vardas, pavardė) (Parašas) (Data)

Lietuvių kalbos konsultantas _dr. Vaida Buivydienė___ _____ _____
(Moksl. laipsnis, vardas, pavardė) (Parašas) (Data)

Vilnius, 2015

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS
FUNDAMENTINIŲ MOKSLŲ FAKULTETAS
INFORMACINIŲ SISTEMŲ KATEDRA

Informatikos inžinerijos studijų kryptis

Informacijos ir informacinių technologijų saugos studijų programa, valstybinis kodas 621E14007

TVIRTINU
Katedros vedėjas

____
(Parašas)

____
(Vardas, pavardė)

____
(Data)

BAIGIAMOJO MAGISTRO DARBO
UŽDUOTIS

……......................Nr. ....
Vilnius

Studentui (ei) ....……….....…………..........................…...……........…......
(Vardas, pavardė)

Baigiamojo darbo tema: ....………..

....

....

patvirtinta 201…m. …………………..…… d. dekano potvarkiu Nr. ………….

Baigiamojo darbo užbaigimo terminas 201…m. ……………….…… d.

BAIGIAMOJO DARBO UŽDUOTIS:

....

....

....

....

....

....

....

Baigiamojo darbo rengimo konsultantai:

....

....

....
(Moksl. laipsnis/pedag. vardas, vardas, pavardė)

Vadovas ....
(Parašas) (Moksl. laipsnis/pedag.vardas, vardas, pavardė)

Užduotį gavau

…………………………………..
(Parašas)

………………………………….
(Vardas, pavardė)

…………………………….…....
(Data)

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS
FUNDAMENTINIŲ MOKSLŲ FAKULTETAS
INFORMACINIŲ SISTEMŲ KATEDRA

Informatikos inžinerijos studijų kryptis

Informacijos ir informacinių technologijų saugos studijų programa, valstybinis kodas 621E14007

PAŽYMA
APIE BAIGIAMĄJĮ MAGISTRO DARBĄ
…….....Nr. .....
Vilnius

Studentas (ė).....
(Vardas, pavardė)

Studento (ės) studijų svertinis įvertinimų vidurkis.....balo.

Baigiamojo darbo tema: .....

Baigiamasis darbas peržiūrėtas ir studentui (ei) .....
leidžiama ginti šį baigiamąjį darbą magistro laipsnio suteikimo komisijoje.

Katedros vedėjas ....
(Parašas) (Moksl. laipsnis/pedag. vardas, vardas, pavardė)

VADOVO ATSILIEPIMAS
APIE BAIGIAMĄJĮ MAGISTRO DARBĄ
……

.....
.....
.....
.....
.....
.....
.....
.....
.....
……..
.....
……

Vadovas ....
(Parašas) (Moksl. laipsnis/pedag. vardas, vardas, pavardė)

Vilniaus Gedimino technikos universiteto egzaminų sesijų ir baigiamųjų darbų rengimo bei gynimo organizavimo tvarkos aprašo 2011–2012 m. m.
2 priedas

(Baigiamojo darbo sąžiningumo deklaracijos forma)

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

_____
(Studento vardas ir pavardė, studento pažymėjimo Nr.)

_____
(Fakultetas)

_____
(Studijų programa, akademinė grupė)

BAIGIAMOJO DARBO SĄŽININGUMO DEKLARACIJA

20 m. _____ d.
(Data)

Patvirtinu, kad mano baigiamasis darbas (projektas) tema _____

_____

_____

patvirtintas 20 ____ m. _____ d. dekano potvarkiu Nr. _____, yra savarankiškai parašytas. Šiame darbe pateikta medžiaga nėra plagijuota. Tiesiogiai ar netiesiogiai panaudotos kitų šaltinių citatos pažymėtos literatūros nuorodose.

Parenkant ir įvertinant medžiagą bei rengiant baigiamąjį darbą, mane konsultavo

mokslininkai ir specialistai: _____

_____

_____

Mano darbo vadovas _____

Kitų asmenų indėlio į parengtą baigiamąjį darbą nėra. Jokių įstatymų nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs (-usi).

_____ _____
(Parašas) (Vardas ir pavardė)

Vilniaus Gedimino technikos universitetas
Fundamentinių mokslų fakultetas
Informacinių sistemų katedra

ISBN ISSN
Egz. sk.. ……
Data ….-….-….

Antrosios pakopos studijų **Informacijos ir informacinių technologijų saugos** programos magistro baigiamasis darbas.

Pavadinimas: **DDoS atakų aptikimas neuroniniu tinklu**

Autorius **Dmitrij Kollerov** Vadovas **prof. dr. Dalius Mažeika**

Kalba

[X] Lietuvių

[ ] Užsienio

Anotacija

Darbe nagrinėjami *DoS*/*DDoS* atakų tipai, pateikiama esančių rinkoje įsilaužimų aptikimo sistemų apžvalga, pateikiami šių sistemų privalumai ir trūkumai. Atliekama daugiasluoksnio perceptrono, spindulinių bazinių funkcijų, Hopfildo tinklo, dvikryptės asociatyviosios atminties modelių analizė, nagrinėjama jų struktūra ir veikimo principai. Siūlomas *DDoS* atakų aptikimo metodas neuroniniu tinklu. Aprašomas neuroninio tinklo mokymo procesas, įėjimo vektorių formavimas, remiantis *KDD Cup 1999* duomenų rinkinio duomenimis. Sukuriami daugiasluoksnio perceptrono ir spindulinių bazinių funkcijų neuroniniai tinklai, aptinkantys *DDoS* atakas. Atliekami šių neuroninių tinklų skirtingų konfigūracijų tyrimai ir pateikiami *DDoS* atakų aptikimo efektyvumo rezultatai.

Darbą sudaro 7 dalys: įvadas, *DDoS* atakos, neuroniniai tinklai, *DDoS* atakų aptikimas naudojant neuroninius tinklus, siūlomas *DDoS* atakų aptikimo metodo prototipas, išvados, literatūros sąrašas.

Darbo apimtis – 74 psl. teksto be priedų, 30 iliustracijų, 19 lentelių, 41 literatūros šaltinis. Atskirai pridedami darbo priedai.

Prasminiai žodžiai: neuroninis tinklas, daugiasluoksnis perceptronas, spindulinės bazinės funkcijos, tinklo srauto anomalija, paskirstytas atsisakymas aptarnauti.

Vilnius Gediminas Technical University
Faculty of Fundamental Sciences
Department of Information Systems

ISBN ISSN
Copies No. ……
Date ….-….-….

Master Degree Studies **Information and Information Technologies Security** study programme Master Graduation Thesis.

Title: **DDoS Attacks Detection Using Neural Network**

Author **Dmitrij Kollerov** Academic supervisor **Prof Dr Dalius Mažeika**

Thesis language

[x] Lithuanian

[ ] Foreign (English)

Annotation

The paper analyzes types of *DoS*/*DDoS* attacks, presents a review of the intrusion detection systems currently existing in the market, as well as discusses the pros and cons of these systems. The analysis of multilayer perceptron, radial basis functions, Hopfield network, bidirectional associative memory models is carried out, their structure and operating principles are considered. The suggested method for the detection of attacks is the neural network method. The neural network learning process and the formation of input vectors are described based on the *KDD Cup 1999* data set. For the detection of *DDoS* attacks, multilayer perceptron and radial basis functions neural networks are developed. Analysis of different configurations of the aforementioned neural networks is carried out, and the results of their effectiveness for the detection of *DDoS* attacks are presented.

The paper consists of 7 parts: introduction, *DDoS* attacks, neural networks, the detection of the *DDoS* attacks by the neural network method, the suggested prototype of the *DDoS* attacks detection method, conclusions and references.

The thesis consists of 74 pages of text (annexes excluded), 30 illustrations, 19 tables, 41 references. The annexes are attached separately.

Keywords: neural network, multilayer perceptron, radial basis functions, anomaly network traffic, distributed denial of service.

Turinys

Paveikslų sąrašas

Lentelių sąrašas

Santrumpos

AIDS (angl. *Application Based Intrusion Detection System*) – programų pagrindu veikianti įsilaužimų aptikimo sistema

C&C (angl. *Command and Control Center*) – valdantysis mazgas

CPU (angl. *Central Processing Unit*) – centrinis procesorius

DB – duomenų bazė

DBVS – duomenų bazių valdymo sistema

DDOS (angl. *Distributed Denial of Service*) – paskirstytas atsisakymas aptarnauti

DMZ (angl. *Demilitarized Zone*) – demilitarizuota zona

DOS (angl. *Denial of Service*) – atsisakymas aptarnauti

DSP – daugiasluoksnis perceptronas

FP (angl. *False Positive*) – klaidingai teigiamas

HIDS (angl. *Host Based Intrusion Detection System*) – tarnybinės stoties pagrindu veikianti įsilaužimų aptikimo sistema

HTTP (angl. *HyperText Transfer Protocol*) – hiperteksto perdavimo protokolas

ICMP (angl. *Internet Control Message Protocol*) – interneto kontrolės pranešimo protokolas

IDS (angl. *Intrusion Detection System*) – įsilaužimų aptikimo sistema

IP (angl. *Internet Protocol*) – interneto protokolas

IT (angl. *Information Technology*) – informacinės technologijos

MLP (angl. *Multilayer Perceptron*) – daugiasluoksnis perceptronas

NIDS (angl. *Network Based Intrusion Detection System*) – tinklo pagrindu veikianti įsilaužimų aptikimo sistema

NT – neuroninis tinklas

NTP (angl. *Network Time Potocol*) – tinklo laiko protokolas

OS – operacinė sistema

OSI (angl. *Open Systems Interconnection*) – atvirųjų sistemų jungtis

RAM (angl. *Random-Access Memory*) – operatyvioji atmintis

RBF (angl. *Radial Basis Function*) – spindulinė bazinė funkcija

SBF – spindulinė bazinė funkcija

SOM (angl. *Self-Organizing Maps*) – saviorganizuojantys žemėlapiai

TCP (angl. *Transmission Control Protocol*) – perdavimo kontrolės protokolas

UDP (angl. *User Datagram Protocol*) – vartotojo paketų kelio schemos protokolas

Įvadas

DDoS (angl. *Distributed Denial of Service*) atakos tapo viena iš populiariausių ir pavojingiausių šių laikų kibernetinio terorizmo [**1**] priemonių. Pirmas oficialus incidentas, susijęs su *DDoS* ataka, buvo užfiksuotas 1996 metais, kai JAV buvo atakuotas interneto tekėjas *Panix* [**2**], bet rimtai apie šią problemą pradėta kalbėti 1999 metais, kai buvo atakuoti *Amazon, Yahoo, CNN, eBay, E-Trade* ir kt. kompanijų serveriai. 2007 metais *DDoS* atakos buvo panaudotos prieš kai kurias Estijos valstybines, finansines įstaigas ir žiniasklaidos organizacijas.

2011 metais programišių grupė *Anonymous* pasirinko *DDoS* atakas, kaip pagrindinį įrankį savo tikslams įgyvendinti. Įspūdingi *Anonymous* grupės rezultatai padarė šias atakas dažnai naudojamu kibernetinio terorizmo įrankiu, kurį pamėgo kitos programišių grupės ir šis metodas tapo gerai žinomas ir lengvai prieinamas ne tik informacinės saugos specialistams, bet ir paprastiems vartotojams, kurie už tam tikrą kainą gali nusipirkti kenkėjišku programiniu kodu užkrėstą ir per atstumą valdomą kompiuterių tinklą (angl. *botnet*) [**3**], [**4**] ir organizuoti *DDoS* ataką.

Yra daug laisvai prieinamų įrankių, skirtų *DDoS* atakoms įvykdyti, o tai skatina tokio tipo atakas naudoti žmones, gerai neišmanančius tinklo protokolų ir tinklo servisų veikimo principų. Pagrindinę šios žmonių kategorijos dalį sudaro vaikai (angl. *script kiddies*), naudojantys profesionalų sukurtus įrankius. Tokios žmonių kategorijos tikslai daugiausia siejami su saviraiška ir vandalizmu. *DDoS* atakų aukomis dažnai tampa konkurentų įmonės, kurios atakuojamos pagal konkuruojančių įmonių užsakymus, siekiant suteikti finansinę žalą savo konkurentams. Akivaizdu, kad šios atakos tapo vienu iš pavojingiausių politinių ir finansinių ginklu informacinėje visuomenėje, kuri dar nėra gerai pasiruošusi šių atakų atsparumui.

Egzistuojančios dabartinės tinklo anomalijų stebėjimo priemonės, tokios kaip *IDS* (angl. *Intrusion Detection System*) dažniausiai naudoja taisyklėmis grindžiamus metodus, kurių pagrindinis tikslas – gauto tinklo srauto paketų palyginimas su esančiomis DB (duomenų bazė) reikšmėmis. Šis metodas yra efektyvus tada, kai DB nuolat papildoma naujais parašais, tačiau toks sprendimas reikalauja žmogaus įsikišimo, be to, naujo parašo sukūrimas reikalauja laiko, todėl po anomalijos aptikimo ir atitinkamo parašo sukūrimo praeina tam tikras laiko intervalas, per kurį teoriškai gali būti pakartotinai organizuota tokia pati ataka, kuri nebus pastebėta tol, kol DB atsiras naujas parašas. Dar vienas tokios sistemos trūkumas – tam, kad visada būtų aktualiausios ir naujausios parašų DB, reikalingas nuolatinis prisijungimas prie interneto, per kurį bus vykdomi atnaujinimai. Todėl pastaruoju metu mokslinėse publikacijose atsirado straipsnių kuriuose pateikiami tinklo anomalijų aptikimai, naudojant neuroninius tinklus [**5**], [**6**], [**7**], [**8**], [**9**].

Darbo problema

DDoS atakos skiriasi nuo daugelio kitų informacinės visuomenės atakų. Visų pirma, jos techniškai sunkiai aptinkamos, kadangi kiekvienas tinklo paketas, siunčiamas aukos kompiuteriui, atrodo kaip teisėtas, t. y. neturi jokios informacijos, kuri galėtų atskirti šios atakos tinklo paketą nuo paprastos, teisėtos vartotojo užklausos. Antra – tokios paralyžiuojančios įmonių serverius ir teikiamas paslaugas atakos gali būti vykdomos daug dienų ar net savaičių. Rezultatas – beveik sustabdoma įmonės veikla, o kompanija patiria nuostolių. Labai sunku atskirti kompiuterius, siunčiančius teisėtas užklausas, nuo naudojamų *DDoS* atakoms. Įprasti metodai, pavyzdžiui, atakuojamų resursų blokavimas, netinka apsaugai nuo *DDoS* atakų, kadangi dažniausiai to ir siekia šio tipo atakų organizatoriai, kad aukos resursas taptų nepasiekiamas. Egzistuoja nuomonė, kad *DDoS* atakos fakto neįmanoma nepastebėti, t. y., jeigu *DDoS* ataka jau įvyko, tai bus aišku be papildomos *DDoS* atakų aptikimo įrangos. Tačiau kai kuriais atvejais *DDoS* atakos gali būti nepastebėtos iš karto, o apie jų egzistavimą sužinoma tik po labai padidėjusių tinklo srauto paslaugų apmokėjimo sąskaitų. Be to, esant *DDoS* atakai, tinklo srauto apkrovimas didėja tolygiai, t. y. nuo atakos vykdymo pradžios iki momento, kai resursas taps nepasiekiamas, praeina tam tikras laiko intervalas ir laiku nustatytas *DDoS* atakos faktas gali suteikti galimybę operatyviai sureaguoti į aptiktą incidentą ir imtis atitinkamų kontrpriemonių.

Tyrimo objektas

Šio darbo tyrimo objektas – neuroniniai tinklai (NT) ir jų pritaikymas aptikti *DDoS* atakas.

Darbo tikslas

Sukurti neuroninį tinklą *DDoS* atakoms aptikti ir atlikti jo efektyvumo tyrimą.

Uždaviniai

1. Atlikti egzistuojančių *DDoS* atakų aptikimo metodų ir įrankių analizę, išnagrinėti jų veikimo principus, privalumus ir trūkumus.
2. Pasiūlyti neuroninius tinklus efektyviam *DDoS* atakų aptikimui.
3. Atlikti eksperimentus, siekiant nustatyti naudojamų neuroninių tinklų efektyvumą.

Temos naujumas

Vienas iš efektyviausių masinio lygiagretumo ir duomenų apdorojimo ir perdavimo procesų paspartinimo, sprendžiant priklausomybių aptikimo ir duomenų klasifikavimo klausimus, yra NT. Darbe siūlomas *DDoS* atakų aptikimo metodas leidžia prisitaikyti prie įvairių atakų nustatymo bei atlikti galimų tinklo saugos incidentų prognozavimą. Laiku aptiktas *DDoS* atakos vykdymo bandymas gali padėti imtis kontrpriemonių, siekiant išvengti šios atakos pasekmių. Dauguma

šiuolaikinių įsilaužimų aptikimo sistemų naudoja parašais pagrįstą metodą tinklo anomalijoms nustatyti, tačiau toks metodas reikalauja daug atminties išteklių ir neapsaugo nuo naujų grėsmių, kurių negalima pastebėti parašais grindžiamu metodu. *DDoS* atakos tampa vis intelektualesnės, vis dažniau naudojama decentralizuota atakuojančių mazgų (angl. *zombie*) architektūra, o tai apsunkina kenkėjišku programiniu kodu užkrėsto kompiuterių tinklo valdančiųjų centrų (angl. *Command and Control Center, C&C*) aptikimą. Išnagrinėtuose literatūros šaltiniuose [**5**], [**6**], [**7**], [**8**], [**9**], kuriuose atliekami panašūs eksperimentai, pateikiama informacija, naudojant tik tam tikrą *KDD Cup 1999* atributų skaičių, tačiau nėra bandoma parodyti, koks turi būti minimalus atributų skaičius, suteikiantis galimybę aptikti tinklo anomaliją. Taip pat nepateikiama informacija apie tai, kokie *KDD Cup 1999* duomenų rinkinio atributai turi didesnę įtaką galutiniam tinklo anomalijų aptikimo rezultatui. Šiame darbe atliekamas eksperimentas, kurio metu nustatoma *KDD Cup 1999* duomenų rinkinio požymių įtaka galutiniam *DDoS* atakos aptikimo rezultatui bei parodoma, kaip keičiasi *DDoS* atakų atpažinimo tikslumas priklausomai nuo šio duomenų rinkinio požymių skaičiaus.

Temos aktualumas

DDoS atakos, populiariausia kibernetinio terorizmo rūšis, yra viena iš aktualiausių šiuolaikinės informacinės visuomenės problemų. Masiškas IT išpopuliarėjimas ir įterpimas į įvairias veiklos sritis, padarė visuomenę priklausomą nuo šių technologijų, ir bet kuris incidentas, orientuotas į informacinės infrastruktūros sutrikdymą, gali sukelti pavojų privačiam gyvenimui, verslui, politikai ir pasaulio stabilumui [**10**]. XXI amžiuje terorizmas persikėlė į virtualią erdvę, tačiau ši erdvė jau seniai yra mūsų gyvenimo realybė ir sukelia realų pavojų mūsų gyvenimui. Pasak *Kaspersky Lab* ir *B2B International* 2013 metais atliktos apklausos, 19 % dalyvavusių apklausoje įmonių tapo *DoS* arba *DDoS* atakų aukomis [**11**]. Remiantis *Arbor Networks* kompanijos duomenimis, 2013 metais buvo pastebėtas *DDoS* atakų, kurių galingumas viršija 20 GB/s, 350 % padidėjimas palyginus su 2012 metais [**12**].

Tyrimo metodika

Darbe naudojami bibliotekų tyrimai, literatūros analizė, empiriniai tyrimo metodai, t. y. tiriama neuroninio tinklo galimybė aptikti pavojingą tinklo srautą. Lyginami skirtingi *DDoS* atakų ir kitų tinklo anomalijų aptikimo metodai, esamos *DDoS* aptikimo ir prevencijos sistemos. Atliekamas eksperimentas, kurio metu nustatomas sukurto NT efektyvumas.

Mokslinė darbo vertė

NT pritaikymas *DDoS* atakoms aptikti suteiks galimybę naudoti dirbtinio intelekto sistemas tinklo anomalijoms nustatyti, prognozuoti galimas tinklo resursų grėsmes, formuoti ir kaupti

informaciją apie jau įvykdytų *DDoS* atakų incidentus, naudoti intelektinius grėsmingų tinklo užklausų srauto filtravimo metodus.

Darbo struktūra

Pirmoje dalyje aprašoma *DDoS* atakų tipų analizė, atliekama esančių rinkoje *DDoS* atakų aptikimo sprendimų apžvalga, pateikiami jų privalumai ir trūkumai. Antroje dalyje aprašomi NT sąvoka, veikimo principas, klasifikacija ir mokymo algoritmai. Trečioje dalyje pateikiama *DDoS* atakų aptikimo procedūra ir atliekami bandymai, siekiant nustatyti naudojamų eksperimente NT efektyvumą aptikti *DDoS* atakas. Tyrimuose parodoma kaip skiriasi *DDoS* atakų atpažinimo efektyvumas priklausomai nuo *KDD Cup 1999* duomenų rinkinio naudojamų atributų skaičiaus. Pateikiami atliktų bandymų rezultatai. Ketvirtoje dalyje pateikiama įsilaužimų aptikimo sistemos architektūra, kurios pagrindą sudaro NT modulis.

Darbo aprobacija

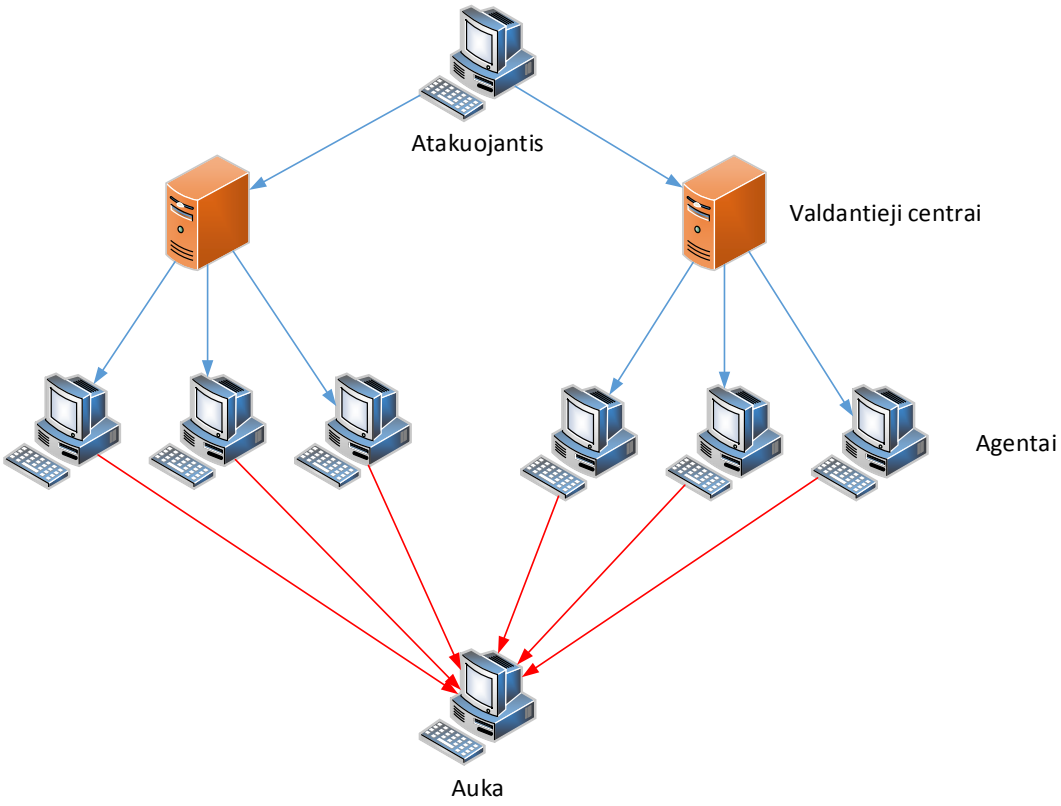
Svarbiausieji darbo rezultatai buvo pristatyti 18-ojoje Lietuvos jaunųjų mokslininkų konferencijoje „Mokslas – Lietuvos ateitis“ (Vilnius, Vilniaus Gedimino technikos universitetas, 2015 m. balandžio 16 d.). Pagal šio darbo tematiką buvo skaitytas pranešimas „DDoS atakų aptikimas neuroniniu tinklu“.

1. DDoS atakos

1.1. Sąvoka ir klasifikacija

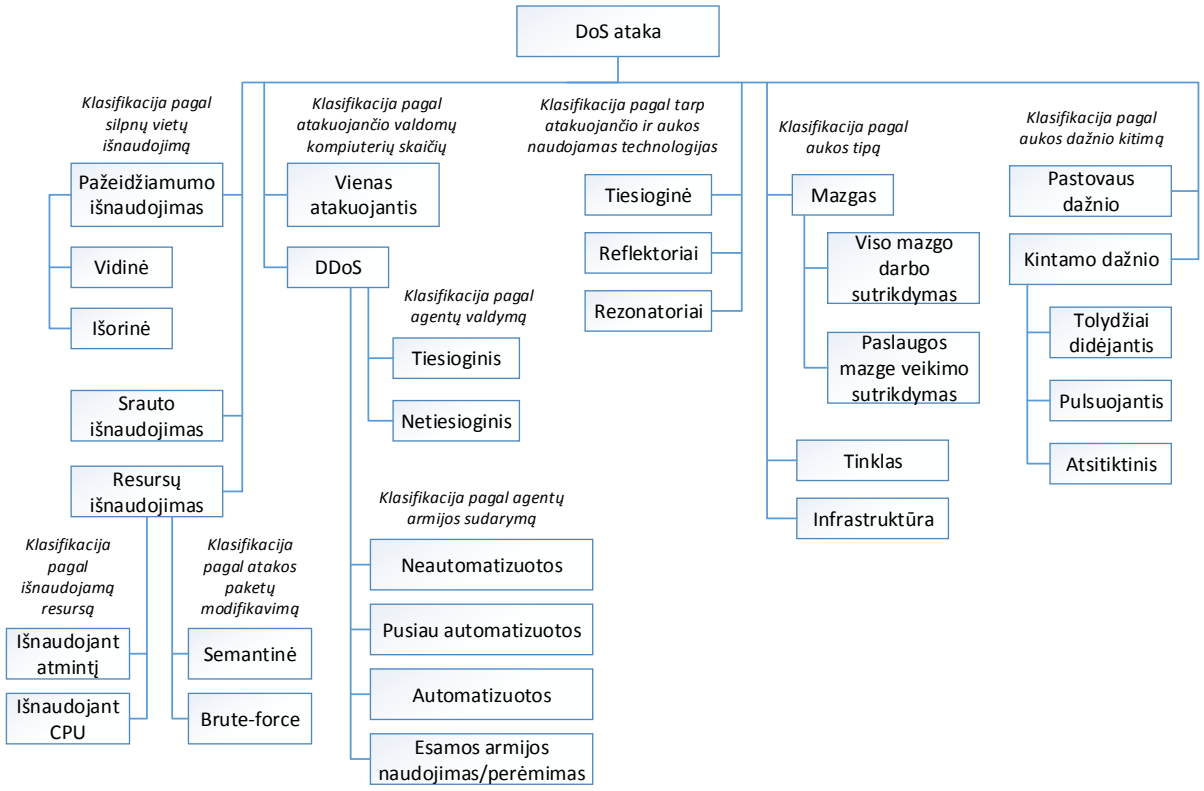
DoS (angl. *Denial of Service*, atsisakymas aptarnauti) ataka – tai ataka prieš informacinę sistemą, skirta sukurti tokias sąlygas, kad sistemos vartotojai negalėtų prieiti prie jos išteklių arba paslaugų. Didelis skaičius vienu metu vykdomų *DoS* atakų vadinamas *DDoS* (angl. *Distributed Denial of Service*, paskirstytas atsisakymas aptarnauti) [**13**]. Šiais laikais *DDoS* atakos dažniausiai organizuojamos kenkėjišku programiniu kodu užkrėstu ir per atstumą valdomu kompiuterių tinklu. Tipinį tokio tipo modelį sudaro du pagrindiniai komponentai (1.1 pav.):

- Valdantysis centras – serveris, prie kurio prisijungę agentai gauna komandas.
- Agentai (atakuojantys mazgai) – programos, galinčios vykdyti nuotoliniu būdu gautas iš valdančiojo centro komandas.



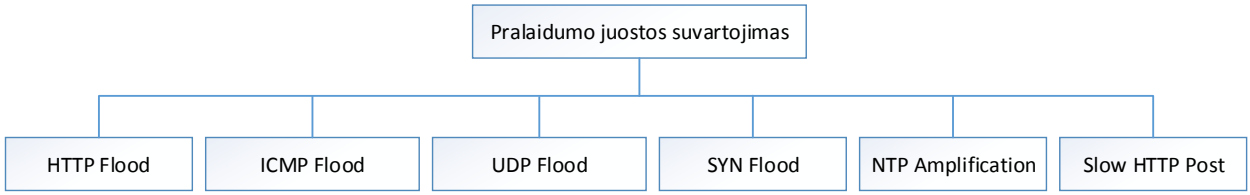
1.1 pav. Kenkėjišku programiniu kodu užkrėsto kompiuterių tinklo modelis

DoS atakų taksonomija pateikta 1.2 pav.



1.2 pav. DoS atakų taksonomija [**14**]

Šiame darbe *DDoS* atakos nagrinėjamos remiantis srauto išnaudojimo (pralaidumo juostos suvartojimo) aspektais, kurie aiškiau pateikti 1.3 pav. [**13**], [**15**].



1.3 pav. Pralaidumo juostos suvartojimas

Pralaidumo juostos suvartojimas (angl. *bandwith consumption*). Tokio tipo ataka apkrauna tinklą, formuodama nekorektiškus paketus.

HTTP Flood ataka naudoja ryšio patikrinimo (angl. *ping*) užklausas. Šios atakos esmė – suformuoti tokį mažą *HTTP* (angl. *HyperText Transfer Protocol*) paketą, siunčiamą serveriui, kad serveris atsakytų į šią užklausą *HTTP* paketu, kurio dydis kelis šimtus kartų didesnis. Tokiu būdu greitai apkraunamas serverio ryšys. Tam, kad atakuojančio vartotojo kompiuteriui nebūtų siunčiami atakuojamo serverio *HTTP* paketai, atakuojantysis kiekvieną kartą keičia savo *IP* (angl. *Internet Protocol*) adresą kitu [**15**].

ICMP Flood ataka naudoja ryšio patikrinimo užklausas siunčiamas plačiajuosčio pranešimo (angl. *broadcast*) režimu. Atakuojantis siunčia aukai padirbtą *ICMP* (angl. *Internet Control Message Protocol*) paketą, paskui atakuojančio adresas keičiamas į aukos adresą, ir visi tinklo mazgai gavę padirbtą plačiajuosčio pranešimo *ICMP* paketą, siunčia atsakymą aukos mazgui. Tokie tinklo mazgai suveikia kaip stiprintuvas, todėl *ICMP* paketas, išsiųstas per stiprinantį tinklą, sudarytą iš 200 mazgų, bus pastiprintas 200 kartų.

UDP Flood – tai *ICMP Flood* atakos analogas, kuriame vietoje *ICMP* paketų, siunčiami *UDP* (angl. *User Datagram Protocol*) paketai. Atakos principas yra toks: 7 prievadai (angl. *port*) siunčiamos aido (angl. *echo*) komandos per transliacinį adresą (angl. *broadcast address*). Paskui atakuojančio *IP* adresas keičiamas aukos *IP* adresu, kurį gauna daugybę atsakymo pranešimų. Pranešimų skaičius priklauso nuo tinklo mazgų skaičiaus. Ši ataka apkrauna tinklo kanalą ir priverčia aukos serverį atsisakyti aptarnauti. Net jeigu atjungtas aido servisas, bus sugeneruoti *ICMP* paketai, o tai irgi nulems tinklo kanalo apkrovimą [**16**].

SYN Flood remiasi 3 rankų paspaudimo protokolo (angl. *3-Way Handshake Protocol*) principais. Atakuojantis siunčia aukai *SYN* paketą ir keičia savo *IP* adresą į neegzistuojantį. Aukos serveris, gavęs *SYN* paketą, siunčia atsakymą *SYN/ACK* neegzistuojančiu adresu ir pereina i *SYN-RECEIVED* būseną [**17**]. Ši būsena nusako, jog sujungimas yra tik pusiau atidarytas, ir laukia atsakymo su *ACK* paketu. Toks pusiau atidarytas sujungimas patenka į eilę. *Windows Server* šeimos sistemose šios eilės dydis pagal nutylėjimą lygus 100, *Unix* sistemose – 256. Kadangi *SYN/ACK* paketas siunčiamas neegzistuojančiam mazgui, sistema niekada negaus iš jo atsakymo ir kiekvienas pusiau atidarytas sujungimas užpildo išskirtą eilę. Kai eilė pasiekia maksimaliai nustatytą dydį, sistema jau nebegali priiminėti naujų užklausų [**18**], [**19**].

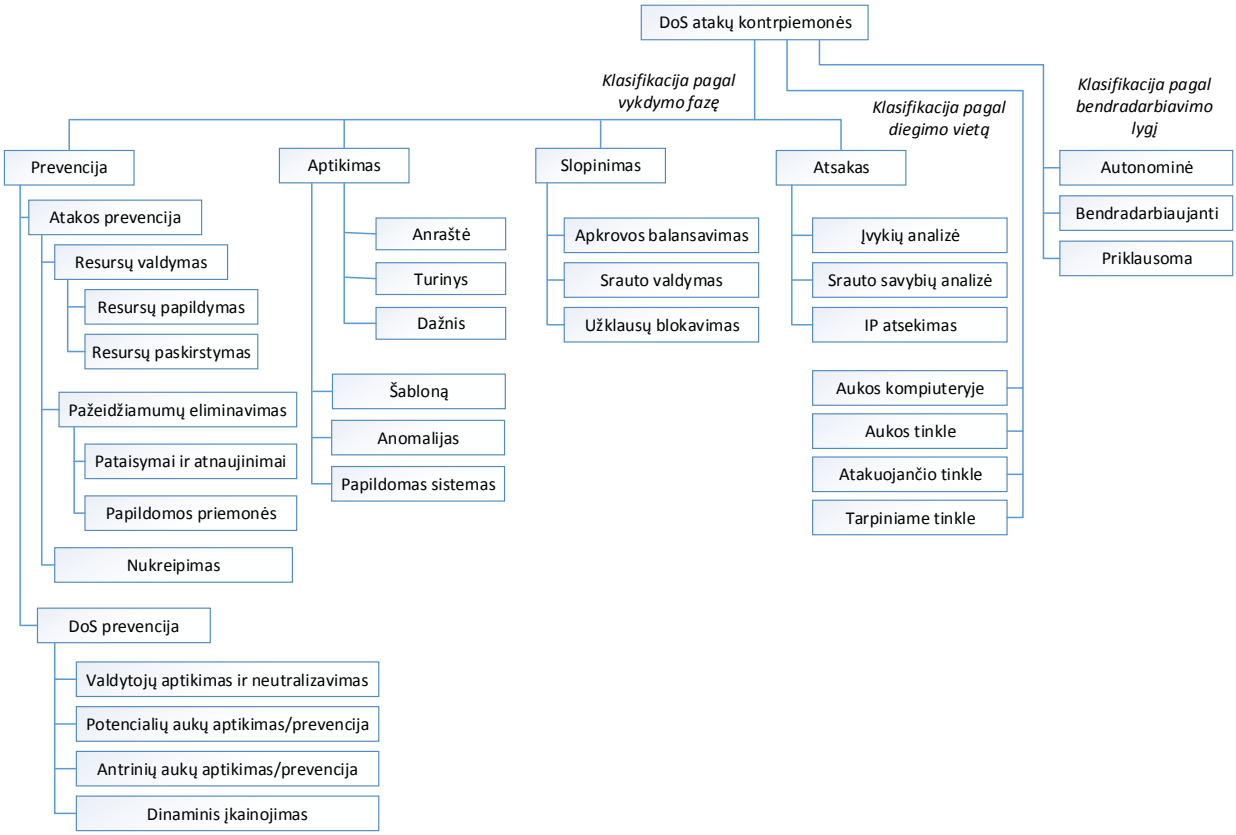
NTP Amplification ataka naudoja *NTP* (angl. *Network Time Protocol*) protokolą. Užkrėsti kompiuteriai siunčia *NTP* serveriui formuojančią sąrašą (angl. *monlist*) užklausą su padirbtu siuntėjo *IP* adresu. Ši užklausa grąžina sąrašą iš 600 paskutinių *NTPD* klientų. Tokiu būdu, suformavus nedidelės apimties užklausą, aukai iš užkrėsto kompiuterio siunčiamas didelis *UDP* srautas. Tai ir yra stiprinimo (angl. *amplification*) esmė. Neapsaugotas *NTP* serveris tampa tarpiniu atakos grandimi. Ši ataka gali būti realizuota *NTPD* kliente, kurio versija senesnė už 4.2.7p26 [**20**].

Slow HTTP Post ataka pasinaudoja *HTTP* protokolo pažeidžiamumu. Atakuojantis siunčia *POST* antraštę su teisėtu lauku *Content-Length*, kuris nusako, kokia duomenų apimtis siunčiama serveriui. Po to, kai antraštė yra persiųsta, pradedama labai lėtu greičiu siųsti *POST* pranešimo turinį, perduodant po 1 baitą. Tam tikru laiko momentu sujungimas nutraukiamas, nesulaukiant užbaigimo, ir pradedama siųsti duomenis iš naujo. Remiantis *HTTP* protokolo standartu [**17**], serveris privalo sulaukti viso persiunčiamų duomenų kiekio (kiek nurodoma lauke *Content-Length*) ir nutraukti sujungimą tik pagal laiko limito išnaudojimą (angl. *time-out*). Serveris

aktyvuoja daug naujų sujungimų, išnaudojami jo resursai, rezultatas – atsisakymas aptarnauti [**19**], [**21**].

1.2. Tinklo srauto anomalijų aptikimo metodai

Egzistuoja skirtingi tinklo anomalijų aptikimo metodai ir būdai, leidžiantys išvengti anomalijas. 1.4 pav. pateikta *DoS* atakų, kaip vienos iš tinklo anomalijų atmainų, kontrpriemonių taksonomija.



1.4 pav. DoS atakų kontrpriemonių taksonomija [**14**]

Vienas iš pagrindinių *DDoS* atakų aptikimo metodų remiasi anomalijų paieška tinklo srauto struktūroje. Tinklo srauto anomalija – tai įvykis arba sąlyga, egzistuojanti tinkle, apibūdinama statistiniu nuokrypiu nuo standartinės srauto struktūros. *DoS* ir *DDoS* atakų metu, bet kuris skirtumas tinklo srauto struktūroje, kuris viršija tam tikrą ribą, traktuojamas kaip ataka. Tradicinės tinklo anomalijų paieškos priemonės, tokios kaip *IDS*, analizuoja tinklo srauto struktūrą ir palygina ją su saugoma DB informacija. Todėl tradiciniai *IDS* mechanizmai remiasi parašais grindžiamais analizės metodais.

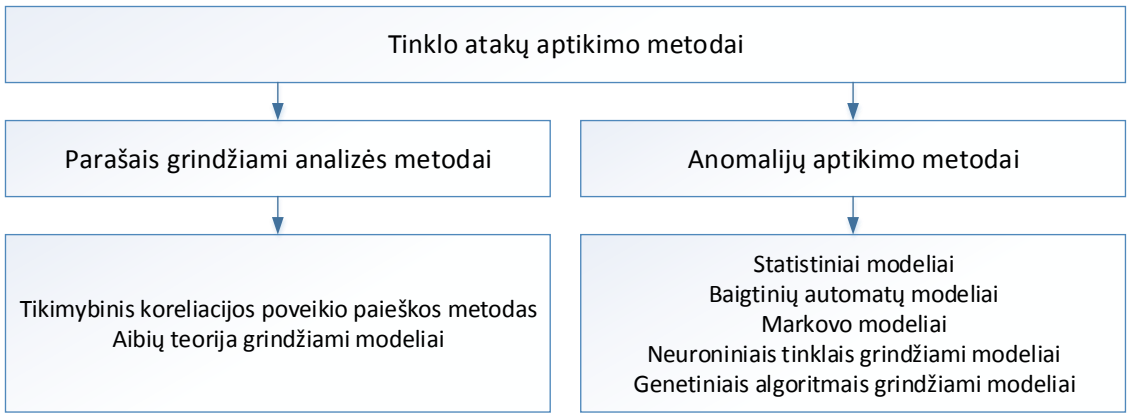
Tinklo atakų aptikimo metodai skaidomi į dvi pagrindines grupes [**8**], [**14**], [**22**], [**23**], [**24**]:

- Parašais grindžiami analizės metodai.
- Anomalijų aptikimo metodai.

Parašais grindžiamas metodas skirtas jau žinomoms atakoms aptikti, remiasi esančiomis DB signatūromis, aprašančiomis anomalijas. Pagrindinis šio metodo privalumas – jis suteikia 100 % anomalijų aptikimą ir tokiu būdu sumažina klaidingai teigiamų (angl. *false-positive*) pranešimų tikimybę iki nulio. Tačiau toks variantas reikalauja nuolatinio signatūrų DB atnaujinimo ir neapsaugo nuo naujų dar nežinomų atakų tipų. Be to, jis netinka *0-day* atakų (atakos, kurios jau yra žinomos, tačiau joms nėra sukurta kontrpriemonių) aptikimui. Dar vienas šio metodo trūkumas yra tas, kad pasaulyje pastoviai atsiranda naujos atakos, todėl teoriškai signatūrų DB dydis turi būti be galo didelis, kad joje būtų galima išsaugoti informaciją apie visas egzistuojančias atakas.

Anomalijų aptikimo metodai skirti dar nežinomoms atakoms nustatyti. Pagrindinis šio metodo principas – nustatoma anomalinė tinklo srauto būsena, kuri skiriasi nuo tipiškos būsenos, ir, remiantis šiuo faktu, priimamas sprendimas dėl įtariamos atakos egzistavimo [**25**].

1.5 pav. pateikiama tinklo atakų aptikimo metodų klasifikacija remiantis naudojamais matematiniais modeliais.



1.5 pav. Tinklo atakų aptikimo metodų klasifikacija [**22**]

Šiame darbe nagrinėjamas anomalijų aptikimo metodas NT pagalba, pateikiama NT klasifikacija ir aprašytos mokymo procedūros.

NT suteikia galimybę aptikti tinklo anomalijas, remiantis anksčiau sukaupta informacija net tuo atveju, kai mokymo rinkinyje nebuvo pavyzdžio, su 100 % tikslumu apibrėžiančio nagrinėjamą tinklo anomaliją. Tai leidžia atsisakyti signatūrų DB, kurioje saugomi tinklo anomalijų pavyzdžiai.

Tačiau NT taip pat turi trūkumų, pavyzdžiui, NT nesugeba teisingai klasifikuoti objektą, jeigu jis ženkliai skiriasi nuo visų pavyzdžių, kurie buvo pateikti NT mokymosi etape. Tai reiškia, kad NT efektyvumas labai priklauso nuo mokymo rinkinio pavyzdžių ir nuo to, kaip tiksliai šie pavyzdžiai apibūdina klasifikuojamą objektą. Be to, NT būdingas ilgas mokymosi laikas. Šis laikas priklauso nuo mokymo duomenų rinkinio apimties, t. y. kuo daugiau pavyzdžių pateikiama NT, tuo ilgiau vykdomas mokymas.

1.3. Įsilaužimų aptikimo sistemos

Pagrindinis komercinių *IDS* (tokių kaip *Cisco IPS, Juniper NetScreen, ISS RealSecure, NFR* ir kt.) analizės sunkumas yra tas, kad viešai prieinama informacija apie šias sistemas dažniausiai pateikiama tik remiantis marketingo tikslais, o ne vidine šių sistemų struktūra. Dėl šios priežasties šiame darbe bus pateikta tik atvirojo kodo *IDS* analizė.

Nagrinėjamų *IDS* sąrašas (1.1 lentelė)

1.1 lentelė. Atvirojo kodo IDS [**26**]

Pavadinimas	**Gamintojas**
Bro	University of California, Lawrence Berkeley National Laboratory
OSSEC	Daniel B. Sid , OSSEC.net
Prelude	Yoann Vandoorselaere Laurent Oudot
Snort	Martin Roesch

IDS palyginimo analizei naudosime tokius kriterijus [**26**]:

Aptinkamų atakų klasė. Šis kriterijus nustato, kokias atakų klases (kategorijas) gali aptikti *IDS*. Tai vienas iš pagrindinių *IDS* vertinimo kriterijų. Atakos klasė susideda iš keturių atributų <L,R,A,D>, kur L – atakuojamo objekto išsidėstymas, R – atakuojamas resursas, A – tikslinis poveikis resursui, D – atakos paskirstymo tipas.

L: atakuojamo objekto išsidėstymas. Gali būti vidinis (*li*) arba išorinis (*le*).

R: atakuojamas resursas. Resursai skirstomi pagal išsidėstymą ir tipą. Pagal išsidėstymą: lokalūs (*rl*) ir tinkliniai (*rn*). Pagal tipą: vartotojo resursai (*ru*), sisteminiai resursai (*rs*), DBVS resursai (*rd*), skaičiavimo resursus (*rc*), apsaugos resursus (*rp*).

A: tikslinis poveikis resursui. Informacijos surinkimas (*as*), resurso vartotojo teisių gavimas (*au*), resurso administratoriaus teisių gavimas (*ar*), resurso vientisumo pažeidimas (*ai*), resurso funkcionavimo pažeidimas (*ad*).

D: atakos pasiskirstymo požymis. Paskirstyta (*dd*), nepaskirstyta (*dn*).

Sistemos monitoringo lygmuo: šis kriterijus nustato įsilaužimų aptikimo sistemos analizuojamos informacijos ribas, t. y. *IDS* tipą:

- *HIDS* (angl. *Host Based Intrusion Detection System*) – tarnybinės stoties pagrindu veikianti įsilaužimų aptikimo sistema.
- *NIDS* (angl. *Network Based Intrusion Detection System*) – tinklo pagrindu veikianti įsilaužimų aptikimo sistema.
- *AIDS* (angl. *Application Based Intrusion Detection System*) – programų pagrindu veikianti įsilaužimų aptikimo sistema.

- *Hybrid* – skirtingų tipų įsilaužimų aptikimo sistemų kombinacija.

Prisitaikymas prie atakos pasikeitimų: galimybė prisitaikyti prie atakos realizacijos pasikeitimų. Šis kriterijus yra vienas iš pagrindinių anomalijomis grindžiamų aptikimo metodų privalumų palyginus su parašais grindžiamais metodais. Galimybės prisitaikyti prie naujų atakos tipų nebuvimas neleidžia operatyviai reaguoti į nežinomas atakas ir reikalauja nuolat atnaujinamų signatūrų.

Aptinkamų atakų klasė. Visos nagrinėjamos sistemos turi galimybę aptikti kelių klasių atakas. Todėl klasių sąjungos ir sankirtos žymėjimui yra atitinkamai naudojami simboliai „∪“ ir „∩“.

Bro sistema – tai tinklo pagrindu veikianti įsilaužimų aptikimo sistema, *NIDS*. Ši sistema sudaryta iš kelių modulių, kiekvienas iš kurių atsakingas už tam tikro *OSI* (angl. *Open Systems Interconnection*) modelio lygio atakų aptikimą. *Bro* remiasi parašais grindžiamu anomalijų aptikimo metodu. *Bro* suteikia galimybę aptikti tokias atakų klases:

$L = \{li \cup le\}$ (išorinės ir vidinės atakos);

$R = \{rn\} \cap \{ru \cup rs\}$ (atakos į tinklo vartotojo resursus ir sisteminius resursus);

$A = \{as \cup au \cup ar \cup ad\}$ (informacijos apie sistemą surinkimas, vartotojo teisių gavimo bandymai, administratoriaus teisių gavimo bandymai, resurso funkcionavimo pažeidimas);

$D = \{dn \cup dd\}$ (nepaskirstytos ir paskirstytos atakos).

OSSEC sistema – vienintelė iš nagrinėjamų darbe sistemų, kurios pagrindinė veikla akcentuojama į OS lygio atakas. *OSSEC* suteikia galimybę aptikti tokias atakų klases:

$L = \{li\}$ (atakuojami objektai yra sistemos viduje);

$R = \{rl\} \cap \{ru \cup rs\}$ (tinklo vartotojo resursai ir sisteminiai resursai);

$A = \{au \cup ar \cup ai\}$ (vartotojo teisių gavimo bandymai, administratoriaus teisių gavimo bandymai, resurso funkcionavimo pažeidimas);

$D = \{dn \cup dd\}$ (nepaskirstytos ir paskirstytos atakos).

Prelude sistema, taip pat kaip *STAT*, tai hibridinė įsilaužimų aptikimo sistema, kuri suteikia galimybę aptikti tokias atakų klases:

$L = \{li \cup le\}$ (išorinės ir vidinės atakos);

$R = \{rl \cup rn\} \cap \{ru \cup rs \cup rp\}$ (atakos į lokalius ir tinklo vartotojo resursus arba sisteminius vartotojo resursus, sisteminius resursus ir apsaugos resursus);

$A = \{as \cup au \cup ar \cup ad\}$ (informacijos apie sistemą surinkimas, vartotojo teisių gavimo bandymai, administratoriaus teisių gavimo bandymai, resurso funkcionavimo pažeidimas);

$D = \{dn\}$ (nepaskirstytos).

Snort – populiariausia šiuo metu nekomercinė *IDS*. *Snort* – tai tinklinė *IDS*, kuri turi papildomai įdiegiamų modulių rinkinį, skirtų specifinių atakų aptikimui. Ši sistema suteikia galimybę aptikti tokias atakų klases:

$L = \{li \cup le\}$ (išorinės ir vidinės atakos);

$R = \{rl \cup rn\} \cap \{ru \cup rs \cup rp\}$ (atakos į lokalius ir tinklo vartotojo resursus arba sisteminius vartotojo resursus, sisteminius resursus ir apsaugos resursus);

$A = \{as \cup au \cup ar \cup ad\}$ (informacijos apie sistemą surinkimas, vartotojo teisių gavimo bandymai, administratoriaus teisių gavimo bandymai, resurso funkcionavimo pažeidimas);

$D = \{dn \cup dd\}$ (nepaskirstytos ir paskirstytos).

Kaip matome, nė viena iš analizuojamų sistemų negali aptikti visų atakų klasių.

Sistemos monitoringo lygmuo. *OSSEC* veikia tik programų lygmenyje (*AIDS* klasė). *Bro* ir *Snort* analizuoja tik tinklo duomenis, t. y. jos priskiriamos *NIDS* klasei. *Prelude* analizuoja duomenis kaip iš lokalių sisteminių šaltinių, taip ir tinklo duomenis (*HIDS*, *NIDS* klasės).

Iš nagrinėjamų sistemų nė viena neapima visų monitoringo lygmenų. Visų aukščiau išvardintų atakų klasių aptikimui reikia analizuoti duomenis programų, tinklo ir tarnybinės stoties lygmenyje.

Prisitaikymas prie naujų atakų tipų. Šiuo metu tokios galimybės nėra nė vienoje iš nagrinėjamų *IDS*.

Žemiau pateikiami aukščiau išnagrinėtų *IDS* palyginimo rezultatai (1.2 lentelė).

1.2 lentelė. IDS palyginimo rezultatai [**26**]

	Bro	**OSSEC**	**Prelude**	**Snort**
Atakų klasės	$(\{li \cup le\}, \{rn\} \cap \{ru \cup rs\}, \{as \cup au \cup ar \cup ad\}, \{dn \cup dd\})$	$(\{li\}, \{rl\} \cap \{ru \cup rs\}, \{au \cup ar \cup ai\}, \{dn \cup dd\})$	$(\{li \cup le\}, \{rl \cup rn\} \cap \{ru \cup rs \cup rp\}, \{as \cup au \cup ar \cup ad\}, \{dn\})$	$(\{li \cup le\}, \{rl \cup rn\} \cap \{ru \cup rs \cup rp\}, \{as \cup au \cup ar \cup ad\}, \{dn \cup dd\})$
Sistemos monitoringo lygmuo	HIDS	HIDS	HIDS, NIDS	NIDS
Aptikimo metodas	Parašais grindžiamas	Parašais grindžiamas	Parašais grindžiamas	Parašais grindžiamas
Prisitaikymas prie naujų atakų tipų	Ne	Ne	Ne	Ne

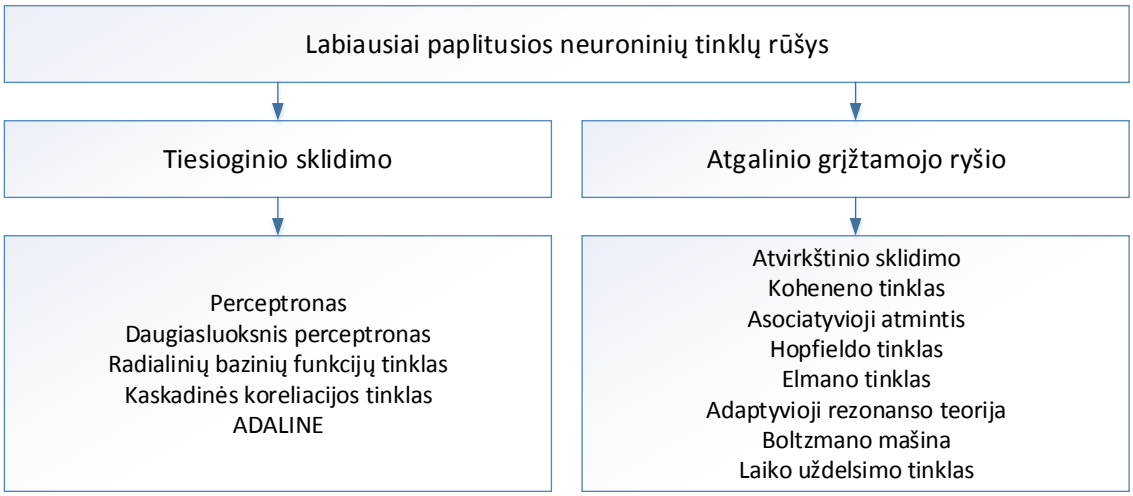
Visos nagrinėjamos sistemos atakų aptikimui naudoja parašais grindžiamą metodą. Skiriasi tik vidinė parašų (signatūrų) analizės realizacija. Tai reiškia, kad šios sistemos nepritaikytos prie naujų, dar nežinomų, atakų aptikimo. Atsižvelgiant į tai, kad kiekvieną diena atsiranda naujos

atakos, ne visada fiziškai įmanoma laiku atnaujinti signatūrų DB. Be to, kol informacija apie naują tinklo anomalijos tipą bus patalpinta į signatūrų DB, praeina tam tikras laiko momentas, reikalingas tam, kad ši anomalija būtų išanalizuota.

2. Neuroniniai tinklai

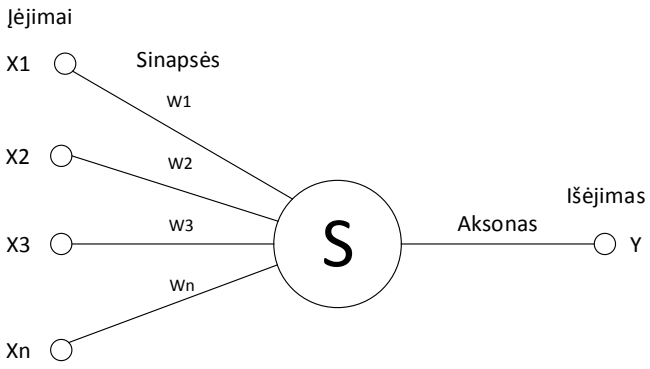
2.1. Apibrėžtis

Dirbtinis neuroninis tinklas (toliau – neuroninis tinklas, NT) – tai supaprastintas smegenų modelis, pateikiantis neuronų, sujungtų tarpusavyje tam tikru būdu, rinkinį [**27**]. Dar viena NT apibrėžtis skamba taip: neuroninis tinklas – tai tinklas su baigtinių sluoksnių skaičiumi, sudarytas iš vienatipių elementų – neuronų analogų su skirtingais sąryšių tipais t/arp sluoksnių. Neuroniniai tinklai leidžia spręsti įvairius praktinius klausimus, susietus daugiausia su atvaizdų (formų) atpažinimu ir klasifikavimu. Neabejotinas neuroninių tinklų privalumas yra tas, kad jie sugeba automatiškai kaupti žinias mokymosi procese ir turi galimybę daryti apibendrinimus. 2.1 pav. pateikta labiausiai paplitusių NT modelių klasifikacija.



2.1 pav. Labiausiai paplitusios NT rūšys [**28**]

Pagrindinis NT elementas yra dirbtinis neuronas [**27**] – matematinis biologinės ląstelės modelis. Dirbtinis neuronas turi sinapsių grupę – vienakrypčių įėjimo ryšių, sujungtų su kitų neuronų išėjimais, taip pat turi aksoną – išėjimo neurono ryšį, iš kurio signalas patenka į kitų neuronų sinapses. Bendras neurono vaizdas pateiktas 2.2 pav.



2.2 pav. Dirbtinio neurono modelis [**27**]

x_i – įėjimo signalai. Visų x_i $(i = 1, 2, \dots, n)$ visuma formuoja vektorių x.

w_i – svorio koeficientai, rodantys stiprumus atskirų įvesčių, aprašytų vektoriumi x.

S – sumavimo blokas. Kiekvienos įvesties signalas dauginamas iš svorio koeficiento. Tokiu būdu gaunama neuroninė jungtis $x_i w_i$. Jei svorio koeficientas teigiamas, xw sužadina signalą išvestyje y, o jei neigiamas – xw slopina išvesties signalą. Sumavimo funkcija aprašoma šia formule:

$$S = \sum_{i=1}^{n} x_i w_i \tag{2.1}$$

Neurono išėjimo signalas Y – tai aktyvavimo funkcija:

$$y = f(x) \tag{2.2}$$

Atsižvelgiant į aktyvavimo funkcijos $f(x)$ tipą, išskiriami keli neuronų tipai. Dažniausiai naudojamas – sigmoidinis neuronas (2.3 pav.) [**28**], kurio funkcija galima išreikšti tokiu pavidalu:

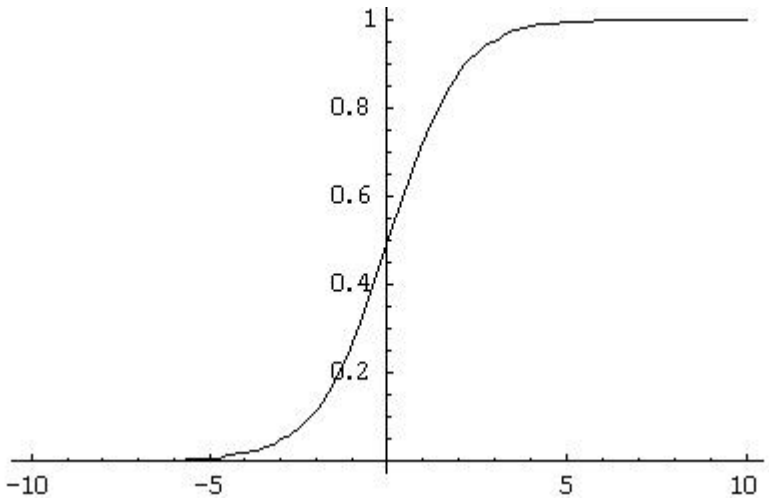
$$f(x) = \frac{1}{1 + e^{-\alpha x}} \tag{2.3}$$

Sigmoidinės funkcijos ypatybės:

$f(x)$ beveik tiesinė, kai $|x|$ yra mažas;

$f(x)$ artėja prie 0, kai $x \to -\infty$;

$f(x)$ artėja prie 1, kai $x \to \infty$.



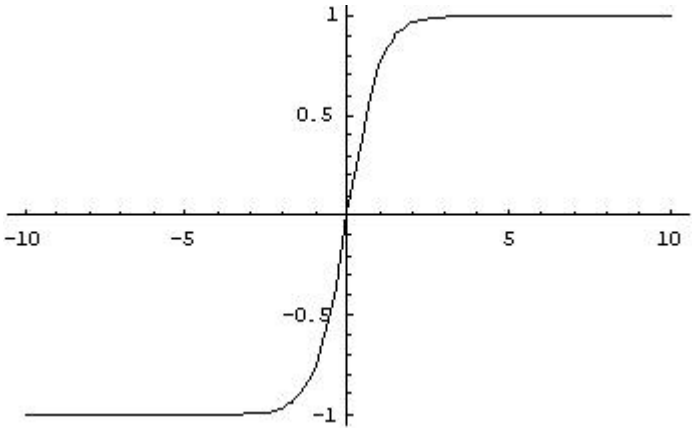
0.4

2.3 pav. Sigmoidinė aktyvavimo funkcija [**28**]

Sigmoidinės arba dar kitaip vadinamos logistinės funkcijos [**29**] rezultatas priklauso intervalui [0, 1], todėl neurono išėjimo vektoriaus reikšmė, naudojant sigmoidinę funkciją, visada priklausys intervalui [0, 1].

Kita dažnai naudojama NT yra hiperbolinio tangento funkcija (2.4 pav.) [**29**], kuri skirtingai nuo sigmoidinės funkcijos priklauso intervalui [-1, 1], kas gali būti naudinga kai kuriuose NT.

$$f(x) = \frac{e^{\alpha} - e^{-\alpha}}{e^{\alpha} + e^{-\alpha}} \tag{2.5}$$



2.4 pav. Hiperbolinio tangento funkcija [**28**]

Aktyvavimo funkcijos pasirinkimas priklauso nuo:

1. užduoties specifikos;
2. realizacijos kompiuteryje patogumu;
3. mokymo algoritmu. Kai kurių algoritmų naudojimas ribojamas tam tikromis aktyvavimo funkcijomis [**29**].

2.2. Dirbtinių neuroninių tinklų mokymas

Galimybė mokytis yra esminė intelekto savybė. Nors tikslų mokymo apibrėžimą sunku suformuluoti, dirbtinio neuroninio tinklo mokymo procesas apibrėžiamas kaip tinklo struktūros ir jungčių svorių keitimo uždavinys, siekiant, kad tinklas galėtų atlikti jam skirtą užduotį. Kiekviename neuroninio tinklo mokymo žingsnyje keičiamos svorių reikšmės atsižvelgiant į įėjimo ir išėjimo reikšmes, gautas ankstesniame mokymo žingsnyje. Procesas kartojamas, kol pasiekiamas norimas rezultatas. Skirtingos tinklų architektūros reikalauja skirtingų jų mokymo algoritmų. Yra trys pagrindinės neuronų mokymo paradigmos [**30**]:

1. mokymo su mokytoju algoritmai (angl. *supervised learning*);
2. mokymo be mokytojo algoritmai (angl. *unsupervised learning*);
3. hibridinis mokymas (angl. *hybrid learning*).

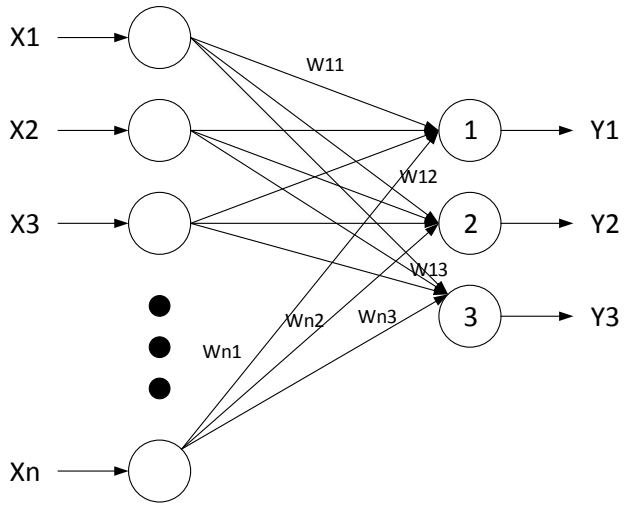
Kalbant apie mokymo su mokytoju algoritmus, yra vartojama norimos išėjimo reikšmės sąvoka (*target, desired output*). Tai iš anksto žinomos reikšmės, pavyzdžiui, klasių numeriai, prognozuojamos reikšmės ir pan.

Mokymo su mokytoju atveju tinklo išėjimų reikšmės, skaičiuojamos kiekvienam įėjimo vektoriui $X_i = (x_{i1}, x_{i2}, \dots, x_{in})$, $i \in \{1, \dots, m\}$, yra tiesiogiai susijusios su norimomis tų išėjimų reikšmėmis. Tinklas koreguojamas keičiant svorių vektorių reikšmes ir siekiant gauti kiek galima mažesnę paklaidą, t. y. ieškoma tokių svorių, kad skirtumas tarp norimų išėjimo reikšmių ir reikšmių, gautų išmokius neuroninį tinklą, būtų kiek galima mažesnis.

Kartais norimos gauti tinklo išėjimo reikšmės nėra žinomos. Tada naudojami mokymo be mokytojo algoritmai. Šio tipo metoduose tinklas mokomas ieškoti koreliacijų ar panašumų tarp mokymo aibės įėjimų. Čia nėra grįžtamojo ryšio, pasakančio, kuris atsakymas bus arba yra teisingas. Mokymo be mokytojo algoritmuose nėra mokytojo signalo. Turima tik mokymo aibė X, kuri sudaryta iš vektorių $X_i, i = 1, \dots, m$, priklausančių erdvei R^n, t. y. $\{X_1, X_2, \dots, X_m\}$. Metodų tikslas yra suskirstyti mokymo duomenis (vektorius) į tam tikras klases arba rasti juose kokius nors reguliarumus ar ypatumus. Gali būti sprendžiamas ir toks uždavinys: vektorių $X_i, i = 1, \dots, m$, reikia atvaizduoti į mažesnio matmenų skaičiaus erdvės vektorių rinkinį taip, kad X_i savybės būtų išlaikytos ir naujos aibės vektoriams. Mokymo be mokytojo sėkmę garantuoja nuo mokytojo nepriklausomas kriterijus, kuris mokymo metu optimizuojamas parenkant tinkamas tinklo jungčių svorių reikšmes [**30**].

2.3. Perceptronas

Nors vienas neuronas gali vykdyti paprasčiausias atpažinimo procedūras, NT skaičiavimų privalumas yra tas, kad jie susideda iš daugelio tarpusavyje sujungtų ir apsikeičiančių signalais neuronų. Atskiri neuronai sujungiami į tinklus su įvairia struktūra. NT struktūros pasirinkimas priimamas atsižvelgiant į užduoties specifiką ir sudėtingumą. Šiuo metu plačiai naudojami daugiasluoksniai perceptronai (2.5 pav.) [**27**], [**28**]. Šiuose tinkluose vieno sluoksnio neuronų išėjimai yra kito sluoksnio įėjimai.



2.5 pav. Vienasluoksnis perceptronas [**28**]

Kiekvienas perceptrono išėjimas y_j, j=1,...d, yra įėjimų $x_1, x_2, \dots x_n$ funkcija, kuri skaičiuojama pagal šią formulę:

$$y_j = f(a_j) = f\left[\sum_{k=0}^{n} w_{jk} x_k\right] \tag{2.6}$$

kur j = 1,...,d, w_{jk} yra jungties iš k-ojo įėjimo į i-ąjį neuroną svoris, x_0 – papildomas įėjimas, įprastai x_0 = 1.

Tarkime, yra m mokymo aibės vektorių $X_i = (x_{i1}, x_{i2}, \dots, x_{in})$, i = 1,…,m. Vektorius X_i dar vadinamas įėjimo vektoriumi, nes jo komponenčių reikšmės $x_{i1}, x_{i1}, \dots, x_{in}$ yra pateikiamos į NT kaip įėjimai.

Įėjimo vektoriai $X_i = (x_{i1}, x_{i2}, \dots, x_{in})$, i = 1,…,m, yra susieti su norimų reikšmių vektoriais $T_i = (t_{i1}, t_{i2}, \dots, t_{id})$, čia t_{ij} – norimo j-ojo išėjimo reakcija į X_i.

Perceptrono mokymo procese svoriai w_{jk} keičiami taip, kad tinklo išėjimo reikšmių vektorius $Y_i = (y_{i1}, y_{i2}, \dots, y_{id})$, gautas į įėjimą pateikus vektorių X_i, būtų kiek galima artimesnis norimų reikšmių vektoriui T_i, t. y. perceptrono paklaida būtų kiek galima mažesnė.

Paklaidos matas $E(W)$ apibrėžiamas kaip svorių matricos $W = \{w_{jk},\ j = 1, \dots, d,\ k = 0, \dots, n\}$ funkcija. Jeigu paklaidos funkcija $E(W)$ yra diferencijuojama pagal svorius w_{jk}, jos minimumą galima rasti gradientiniais optimizavimo metodais. Geriausiai žinomas yra gradientinio nusileidimo algoritmas. Dažniausiai naudojama paklaidos funkcija yra kvadratinių paklaidų suma:

$$E(W) = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{d}(y_{ij} - t_{ij})^2 = \sum_{i=1}^{m} E_i(W) \tag{2.7}$$

$$E_i(W) = \frac{1}{2}\sum_{j=1}^{d}(y_{ij} - t_{ij})^2 \tag{2.8}$$

Iš pradžių generuojamos atsitiktinės svorių w_{jk} reikšmės. Tada gradientinio nusileidimo algoritmu judama antigradiento kryptimi, svorių reikšmes keičiant pagal iteracinę formulę:

$$w_{jk}(t+1) = w_{jk}(t) + \Delta w_{jk}(t) \tag{2.9}$$

čia

$$\Delta w_{jk}(t) = -\eta \frac{\partial E(t)}{\partial w_{jk}} = -\eta \sum_{i=1}^{m} \frac{\partial E_i(t)}{\partial W_{jk}} = \sum_{i=1}^{m} \Delta w^i{}_{jk}(t) \tag{2.10}$$

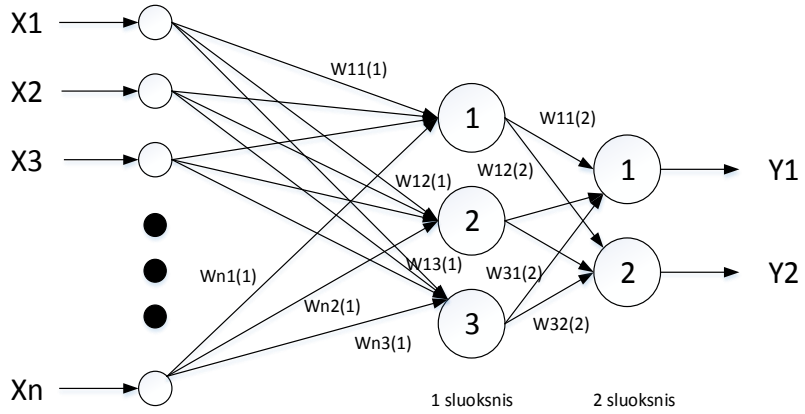
$$\Delta w^i{}_{jk}(t) = -\eta \frac{\partial E_t(t)}{\partial w_{jk}} \tag{2.11}$$

kur η yra teigiamas daugiklis, kuris vadinamas mokymo greičiu (angl. *learning rate*) ir kuriuo reguliuojamas gradientinio optimizavimo žingsnio ilgis, t – iteracijos numeris.

Galimos dvi svorių keitimo kategorijos. Vienu atveju svoriai w_{jk} pakeičiami pagal (2.9) formulę pateikus į tinklą visus mokymo aibės vektorius. Kitoje strategijoje svoriai w_{jk} pakeičiami pagal (2.6) formulę po kiekvieno mokymo aibės vektoriaus pateikimą į tinklą.

Vienasluoksniai dirbtinių neuronų tinklai netinka sudėtingiems uždaviniams spręsti dėl jų skiriamųjų paviršių paprastumo. Šiai problemai išspręsti naudojami tinklai, turintys keletą neuronų sluoksnių. Jų pagalba galima sudaryti žymiai sudėtingesnius skiriamuosius paviršius, kas leidžia spręsti daug sudėtingesnius uždavinius [**31**].

Turintys daugiau nei vieną neuronų sluoksnį tinklai, kuriuose galimi tik ryšiai į priekį iš įėjimų į išėjimus, yra vadinami daugiasluoksniais perceptronais (angl. *multilayer perceptrons*) arba tiesioginio sklidimo neuroniniais tinklais (angl. *multilayer feedforward neural networks*) (2.6 pav.). Kiekvienas toks tinklas sudarytas iš įėjimų aibės, išėjimų neuronų sluoksnio ir paslėptųjų neuronų sluoksnių tarp įėjimų ir išėjimų [**30**].



2.6 pav. Daugiasluoksnis perceptronas [**27**]

Įprastai daugiasluoksnis NT daromas su vienu ar keliais paslėptais sluoksniais. Tačiau reikalingas sluoksnių skaičius ir neuronų skaičius juose nėra nusakomas vienareikšmiškai. Neuroninis tinklas bendru atveju gali būti kokio tik norima dydžio. Didinant sluoksnių skaičių juose, didėja tinklo sudėtingumas. Jau dviejų sluoksnių neuroninis tinklas gali aproksimuoti bet kokią tolydinę funkciją. Tačiau dažnai didinant sluoksnių skaičių tinklo apimtys mažėja, t. y. neuronų skaičius mažėja. Dažniausiai praktiškai naudojamų neuroninių tinklų architektūra susideda iš nuosekliai vienas po kito einančių neuronų sluoksnių, kur kiekvieno sluoksnio neuronas savo išėjimais sujungtas su kito sluoksnio visais neuronais ir neturi jokių kitų jungčių. Čia iš žemesnio sluoksnio signalai keliauja į aukštesnį sluoksnį, kol galiausiai pasiekia išėjimo sluoksnį.

NT neuronų ir sluoksnių skaičiaus parinkimas nustatomas priklausomai nuo sprendžiamos užduoties. Kuo didesnis neuronų ir sluoksnių skaičius, tuo didesnės NT galimybės ir tuo daugiau laiko užima jo apmokymo procesas.

Neuronų ir sluoksnių skaičius susietas su:

1. uždavinio sunkumu;
2. duomenų kiekiu, reikalingu NT apmokymui;
3. reikalaujamu NT įėjimų ir išėjimų skaičiumi;
4. aparatiniais resursais, tokiais kaip: atmintis, procesoriaus našumas [**29**].

Tegul turime daugiasluoksnį neuroninį tinklą, kuriame yra L sluoksnių, pažymėtų l = 0,1,...,L, čia sluoksnis l = 0 žymi įėjimus, o $l = L$ – paskutinį (išėjimų) sluoksnį. Kiekviename sluoksnyje l yra n_l neuronų. Pirmojo sluoksnio j-ojo neurono išėjimo reikšmė y_j yra apskaičiuojama pagal (2.6) formulę. Išėjimai į neuronus l-ajame sluoksnyje yra neuronų išėjimai ($l-1$)-ajame sluoksnyje. Todėl kiekvieno j-ojo neurono išėjimo reikšmė y_j l-ajame sluoksnyje yra apskaičiuojama taip:

$$y_j = f(a_j) = f\left[\sum_{k=0}^{n_{l-1}} w_{jk} y_k\right] \tag{2.12}$$

kur j = 1,...n_l, f – neuronų aktyvacijos funkcija, w_{jk} – svoriai jungčių, kurios jungia k-ąjį neuroną ($l-1$)-ajame sluoksnyje su j-uoju neuronu l-ajame sluoksnyje, n_{l-1} – neuronų skaičius ($l-1$)-ajame sluoksnyje, $y_0 = 1$. Kairioje (2.12) lygybės pusėje y_j yra l-ojo neurono išėjimo reikšmė, o dešiniojoje – y_k yra ($l-1$)-ojo sluoksnio k-ojo neurono išėjimo reikšmė. Indeksai l ir ($l-1$) į (2.12) formulę neįtraukiami siekiant, kad būtų paprasčiau aiškinti tinklo mokymo algoritmą [**30**].

Vienas iš dažniausiai naudojamų daugiasluoksnio perceptrono apmokymo metodų – atvirkštinio klaidos skleidimo algoritmas. Daugiasluoksnio perceptrono apmokymas remiasi tinklo funkcijos klaidos minimizavimu $E(w)$. Klaida nustato tikimų (laukiamų) NT išėjimų t^n nukreipimą nuo gautų reikšmių y^n. Dažniausiai funkcija $E(w)$ nustatoma remiantis mažiausių kvadratų metodu:

$$E(w) = \frac{1}{2}\sum_{j,p}\left(y_{j,p}{}^{(N)} - d_{j,p}\right)^2 \tag{2.13}$$

kur $y_{j,p}{}^{(N)}$ – reali N-ojo išėjimo sluoksnio j-ojo neurono būsena, kai į NT įėjimus paduodamas p atvaizdas;

$d_{j,p}$ – ideali (norima) neurono būsena.

Minimizavimas vykdomas gradientinio nusileidimo metodo pagalba, kas reiškia visų svorio koeficientų koregavimą (paderinimą) tokiu būdu:

$$\Delta w_{ij}{}^{(n)} = -\eta \frac{\partial E}{\partial w_{ij}} \tag{2.14}$$

čia w_{ij} – sinapsių sąryšio svorio koeficientas, sujungiantis *(n-1)*-ojo sluoksnio *i*-tąjį neuroną su *j*-tuoju *n* sluoksnio neuronu, η – apmokymo greičio koeficientas, $0 < \eta < 1$.

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial y_j}\frac{\partial y_j}{\partial s_j}\frac{\partial s_j}{\partial w_{ij}} \tag{2.15}$$

kur y_j – *j*-ojo neurono išėjimas, s_j – įėjimo signalų svorinė suma, t. y. aktyvavimo funkcijos argumentas. Kadangi daugiklis $\frac{\partial y_j}{\partial s_j}$ yra šios funkcijos išvestinė pagal jos argumentą, tai reiškia, kad aktyvavimo funkcijos išvestinė turi būti apibrėžiama visoje abscisės ašyje. Todėl ir pavienio šoktelėjimo funkcija, ir kitos aktyvavimo nevienarūšės funkcijos netinka apžvelgiant NT. Jose taikomos tokios sklandžios funkcijos, kaip hiperbolinis tangentas arba klasikinė sigmoidė su eksponente. Hiperbolinio tangento atveju:

$$\frac{\partial y}{\partial s} = 1 - s^2 \tag{2.16}$$

Trečias daugiklis $\frac{\partial s_j}{\partial w_{ij}}$ akivaizdu, yra lygus neuronų išėjimui prieš tai esančiame sluoksnyje $y_j{}^{(n-1)}$. Kas liečia pirmojo (2.15) formulės daugiklio, jis yra lengvai išskaidomas šiuo būdu:

$$\frac{\partial E}{\partial y_j} = \sum_k \frac{\partial E}{\partial y_k}\frac{\partial y_k}{\partial s_k}\frac{\partial s_k}{\partial y_j} = \sum_k \frac{\partial E}{\partial y_k}\frac{\partial y_k}{\partial s_k} w_{jk}{}^{(n+1)} \tag{2.17}$$

Čia vykdomas sumavimas pagal *k* tarp *(n+1)*-ojo sluoksnio neuronų. Įvedus naują kintamąjį:

$$\delta_j{}^{(n)} = \frac{\partial E}{\partial E_j}\frac{\partial s_j}{\partial s_j} \tag{2.18}$$

gauname rekursinę formulę, kuri skirta *n*-ojo sluoksnio $\delta_j{}^{(n)}$ reikšmių skaičiavimui remiantis (n+1)-ojo sluoksnio reikšmėmis $\delta_j{}^{(n+1)}$.

$$\delta_j{}^{(n)} \left[\sum_k \delta_k{}^{(n+1)} w_{jk}{}^{(n+1)} \right] \frac{\partial y_j}{\partial s_j} \tag{2.19}$$

Išėjimo sluoksnio $\delta_j{}^{(n)}$ reikšmių skaičiavimui naudojama formulė:

$$\delta_j{}^{(N)} = (y_j{}^{(N)} - d_j)\frac{\partial y_j}{\partial s_j} \tag{2.20}$$

Dabar formulę (2.14) galima užrašyti atskleistame pavidale:

$$\Delta w_{ij}{}^{(n)} = -\eta \delta_j{}^{(n)} y_i{}^{(n-1)} \tag{2.21}$$

Kartais suteikiant svorių korekcijos procesui truputį inertiškumo, sulyginančio ryškius šoktelėjimus, veikiančius pagal tikslinę funkciją, formulę (2.21) papildoma svorio pateikimo reikšme pirmesnėje iteracijoje:

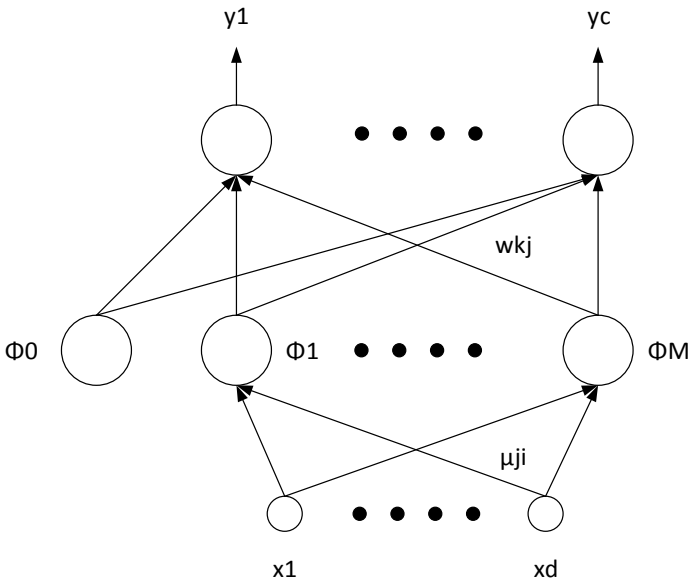
$$\Delta w_{ij}{}^{(n)}(t) = -\eta(\mu \Delta w_{ij}{}^{(n)}(t-1) + (1-\mu)\delta_j{}^{(n)} y_i{}^{(n-1)}) \quad (2.22)$$

kur μ – inertiškumo koeficientas, t – esančios iteracijos numeris.

Iš formulės (2.21) seka: kai $y_j{}^{(n-1)}$ išėjimo reikšmė siekia nulį, apmokymo efektyvumas ryškiai mažėja. Esant dvejetainiams vektoriams vidutiniškai pusė svorinių koeficientų nebus koreguojama, todėl neuronų išėjimo galimų reikšmių sritis [0,1] geriausiai pasirinkti [-0.5,+0.5], to galima pasiekti paprastai modifikuojant logines funkcijas. Pavyzdžiui, sigmoidė su eksponente galima paversti į tokią formulę:

$$f(x) = -0.5 + \frac{1}{1+e^{-\alpha x}} \quad (2.23)$$

2.4. Spindulinės bazinės funkcijos



2.7 pav. Spindulinių bazinių funkcijų tinklas [**28**]

Spindulinių bazinių funkcijų (SBF) (angl. *Radial Basis Functions, RBF*) tinklai (2.7 pav.) kardinaliai skiriasi nuo daugiasluoksnio perceptrono. Daugiasluoksnis perceptronas sprendžia uždavinį neuronų, atliekančių nelinijinius savo svorinių įėjimų transformavimus, pagalba. SBF tinkluose neuronų aktyvacija nustatoma remiantis tarp įėjimo vektoriaus ir vektoriaus prototipo, nustatyto mokymo procese, distancija. SBF mokymo procesas yra greitesnis nei daugiasluoksnio perceptrono ir šio tipo tinkluose gali būti naudojamas kaip mokymas su mokytoju, taip ir mokymas be mokytojo metodas. SBF tinklų teorijos pagrindas yra tikslus funkcijų artėjimas (aproksimacija), kurį 1987 m. pasiūlė Michael J. D. Powell [**28**].

Tegul yra užduotas rinkinys, sudarytas iš *N* įėjimo vektorių x^n su atitinkamais t^n išėjimais. Tikslaus funkcijų artėjimo (aproksimacijos) uždavinys – rasti tokią funkciją h, kad $h(x^n) =$

$t^n, n = 1, \dots, N$. Šiam tikslui naudojamas bazinių funkcijų $\Phi(|x - x^n|)$ pavidalo rinkinys. Tada gauname:

$$h(x) = \sum_n w_n \Phi(|x - x^n|) = t^n, \;\; n = 1, \dots, N \tag{2.24}$$

čia w_n – laisvai pasirenkami parametrai. Dažniausiai bazinė funkcija – tai eksponentinė funkcija:

$$\Phi(x) = e^{-x^2/(2\sigma^2)} \tag{2.25}$$

kur σ – tai reguliuojamas parametras. Tokia funkcija turi reikalingą bazinės funkcijos savybę:

$$\lim_{x \to \infty} (\Phi(x)) \to 0 \tag{2.26}$$

SBF tinklo mokymas:

Tarkime, turime mokymo rinkinį: $\{x^n\} \to \{t^n\}$. Mokymas sudarytas iš dviejų etapų:

1. Nustatomi bazinių funkcijų parametrai: μ_j, σ_j. Be to, kaip taisyklė, naudojami tik įėjimo vektoriai $\{x^n\}$, t. y. naudojamas mokymas be mokytojo metodas. σ_j nustatymui naudojamas k-artimiausio kaimyno (angl. *k-nearest neighbor*) algoritmas. Pasirenkamas centrų (bazių) skaičius *M*. Algoritmo pagalba ieškomas $\{x^n\}$ aibės skaidymas į *M* gretimų poaibių S_j tokiu būdu, kad minimizuoti funkciją *J*:

   $$J = \sum_{j=1}^{M} \sum_{n \in S_j} \left| x^n - \mu_j \right|^2 \tag{2.27}$$

   kur $\mu_j = \frac{1}{N_j} \sum_{n \in S_j} x^n$, čia μ_j yra faktiškai vidurinis S_j taškas. Pirminis skaidymas atliekamas atsitiktiniu būdu. Apskaičiuojamas μ_j. Paskui skaidymas keičiamas tol, kol egzistuoja *J* mažinimo galimybė. Parametras σ_j pasirenkamas euristiniu būdu. Kaip taisyklė, σ_j priskiriama reikšmė, kuri šiek tiek viršija atstumą tarp atitinkamų bazinių funkcijų μ_j centrų.
2. Fiksuojamos bazinės funkcijos. SBF tinklas tampa lygus vienasluoksniam NT. Paskui mokymas vykdomas remiantis mokymo su mokytoju metodu. Minimizuojama klaida:

   $$E(w) = \frac{1}{2} \sum_n \sum_k (w_{kj} \Phi_j - {t_k}^n)^2 \tag{2.28}$$

   Kadangi *E* yra kvadratinė *w* svorių funkcija, todėl minimumas *E* randamas sprendžiant tiesinių lygčių sistemą:

   $$\frac{\partial E}{\partial w} = \sum_n \left\{ \sum_k (w_{kj} \Phi_j - {t_k}^n) \right\} \Phi_j = 0 \tag{2.29}$$

2.5. Mokymas be mokytojo

Mokymas be mokytojo, arba dar kitaip vadinamas neprižiūrimas mokymas, neturi išorinio mokytojo. Remdamasi vidiniais kriterijais ir tinklo informacija, sistema pati save turi suderinti. Mokymas be mokytojo metodas yra arčiau prie biologinės sistemos mokymo modelio. Šiame modelyje nėra apmokymo duomenų, o egzistuoja tik įėjimo reikšmės. Mokymosi procese nėra neuronų išėjimo reikšmių palyginimo su iš anksto apibrėžtu etalonu. Mokymo procesas, kaip ir mokymosi su mokytoju atveju, yra tame, kad jame koreguojami sinapsių svoriniai koeficientai. Kai kurie algoritmai numato ir NT struktūros pasikeitimą, t. y. neuronų kiekį ir jų sąryšius, bet tokius pertvarkymus tiksliau būtų pavadinti saviorganizacija. Akivaizdu, kad sinapsų koregavimas gali būti vykdomas tik remiantis ta informacija, kuri prieinama neuronui, t. y. jo informacija ir turimus svorinius koeficientus. Remiantis aukščiau išvardintais suvokiamais buvo sukurti mokymo be mokytojo algoritmai.

Galimos trys mokymo be mokytojo strategijos [**30**]:

1. Hebbo mokymas (angl. *Hebbian learning*);
2. varžytinių mokymas (angl. *competitive learning*);
3. saviorganizuojantis mokymas (angl. *self-organizing learning*).

Šios strategijos yra realizuojamos skirtingos struktūros neuroniniais tinklais ir todėl leidžia spręsti skirtingus duomenų analizės uždavinius. Hibridinis mokymas apima mokymo su mokytoju ir be mokytojo algoritmus: dalis tinklo svorių nustatomi pagal mokymą su mokytoju, kita dalis gaunama iš mokymo be mokytojo [**30**].

Hebbo taisyklė teigia, kad, jeigu neuronas priima įėjimo signalą nuo kito neurono ir abu neuronai yra aktyvūs (matematiškai turi tą patį ženklą), tai svoris tarp neuronų turi būti sustiprintas [**32**]. Hebbo taisyklė gali būti naudojama tiesioginio skleidimo NT mokymui, skirtam klasifikavimo uždaviniams spręsti. Paprasčiausia Hebbo taisyklės realizacija vadinama signaliniu metodu [**28**].

Signalinis Hebbo mokymo metodas remiasi svorių koregavimu pagal tokią taisyklę:

$$w_{ij}(t) = w_{ij}(t-1) + \alpha {y_i}^{(n-1)} {y_j}^{(n)} \tag{2.30}$$

kur ${y_i}^{(n-1)}$ – tai (n+1)-ojo sluoksnio i-ojo neurono išėjimo reikšmė, ${y_j}^{(n)}$ – n-ojo sluoksnio j-ojo neurono išėjimo reikšmė; $w_{ij}(t)$ ir $w_{ij}(t-1)$ – sinapsės, sujungiančios šiuos neuronus, svorinis koeficientas t ir t – 1 iteracijose; α – mokymosi greičio koeficientas. Vykdant šį apmokymo metodą, stiprėja sąryšiai tarp sužadintų neuronų [**27**].

Taip pat egzistuoja diferencialinis Hebbo mokymo metodas:

$$w_{ij}(t) = w_{ij}(t-1) + \alpha\left[{y_i}^{(n-1)}(t) - {y_i}^{(n-1)}(t-1)\right]\left[{y_j}^{(n)}(t) - {y_j}^{(n)}(t-1)\right] \quad (2.31)$$

čia ${y_i}^{(n-1)}(t)$ ir ${y_i}^{(n-1)}(t-1)$ – tai ($n-1$)-ojo sluoksnio i-ojo neurono išėjimo reikšmė t ir ($t-1$)-ose iteracijose; ${y_j}^{(n)}(t)$ ir ${y_j}^{(n)}(t-1)$ – tai n-ojo sluoksnio j-ojo neurono išėjimo reikšmė t ir ($t-1$)-ose iteracijose.

Hebbo mokymo algoritmą galima suskaidyti į tokius žingsnius:

1. Inicializacijos stadijoje visiems svorio koeficientams priskiriamos nedidelės atsitiktinės reikšmės.
2. Į NT įėjimus paduodamas įėjimo atvaizdas (vektorius) ir sujaudinimo signalai perduodami per visus sluoksnius pagal klasikinių tiesioginio skleidimo NT principus, t. y. kiekvienam neuronui apskaičiuojama svorinė jo įėjimų suma, kuriai paskui pritaikoma aktyvavimo neurono funkcija. Rezultatas – gaunama neurono išėjimo reikšmė ${y_j}^{(n)}$, $i = 0 \ldots (M_i - 1)$, kur M_i – neuronų skaičius i sluoksnyje; $n = 0 \ldots (N - 1)$, kur N – NT sluoksnių skaičius.
3. Remiantis gautais išėjimo rezultatais, apskaičiuotais pagal (2.30) arba (2.31) formulę, vykdomas svorių koeficientų koregavimas.
4. Jeigu išėjimo reikšmės nesusistabilizuoja su nurodytu tikslumu, atliekamas ciklo kartojimas pradedant nuo 2 žingsnio. Kadangi taikant Hebbo algoritmą, naudojamas tiesioginio skleidimo NT, jis anksčiau ar vėliau pereis į stabilią būseną (susistabilizuos).

NT apmokytas pagal Hebbo taisyklę sugeba apibendrinti panašius vaizdus, priskiriant juos prie vienos klasės [**27**].

Kitas apmokymo be mokytojo algoritmas – Kohoneno algoritmas. Kohoneno tinklai – vienas iš pirmųjų duomenų vaizdavimo metodų, naudojančių neuroninius tinklus, sumodeliuotas pagal biologinius mechanizmus.

Kohoneno tinklai – tai saviorganizuojantys NT (žemėlapiai) (angl. *self-organizing maps, SOM*) dar vadinami Kohoneno saviorganizuojančiais žemėlapiais. Šio tipo neuroninių tinklų pavadinimas kilo iš to, kad save organizuojantis žemėlapis, naudodamas mokymo aibę, pats save sukuria (organizuoja). *SOM* tinklo esmė – duomenų topografijos išlaikymas. Taškai, esantys arti vieni kitų įėjimo vektorių erdvėje, yra atvaizduojami arti vieni kitų ir *SOM* žemėlapyje. *SOM* žemėlapiai gali būti naudojami siekiant vizualiai pateikti duomenų klasterius bei ieškant daugiamačių duomenų projekcijas į mažesnės dimensijos erdvę, įprastai į plokštumą [**30**].

Kohoneno algoritmas numato sinapsių koregavimą remiantis jų praeitos iteracijos reikšmėmis:

$$w_{ij}(t) = w_{ij}(t-1) + \alpha\left[{y_i}^{(n-1)} - w_{ij}(t-1)\right] \quad (2.32)$$

Iš aukščiau pateiktos formulės akivaizdu, kad mokymas suvedamas prie skirtumo tarp neurono įėjimo signalų, ateinančių iš ankstesnio neuronų sluoksnio ${y_i}^{(n-1)}$ išėjimų ir jo sinapsių svorinių koeficientų, minimizavimo [**27**].

Kitas variantas – atstumo tarp vektorių *p*-matėje erdvėje skaičiavimas:

$$D_j = \sqrt{\sum_{i=0}^{p-1} ({y_i}^{(n-1)} - w_{ij})^2} \quad (2.33)$$

kur p – vektorių dimensija, j – neurono indeksas n-tajame sluoksnyje, i – sumavimo pagal $(n-1)$-ojo sluoksnio neuronus indeksas, w_{ij} – sinapsės, sujungiančios neuronus, svoris; $(n-1)$-ojo sluoksnio neuronai yra n-ojo sluoksnio įėjimo reikšmės. Šiuo atveju nugalėtoju tampa neuronas su mažiausiu atstumu tarp svorinių koeficientų vektoriaus ir įėjimo reikšmių vektoriaus.

Praktinėje Kohoneno mokymo algoritmo realizacijoje, kaip taisyklė, atliekama įėjimo atvaizdų normalizacija. Inicializacijos etape normalizuojamos pradinės svorinių koeficientų reikšmės:

$$x_i = \frac{x_i}{\sqrt{\sum_{j=0}^{n-1} {x_j}^2}} \quad (2.34)$$

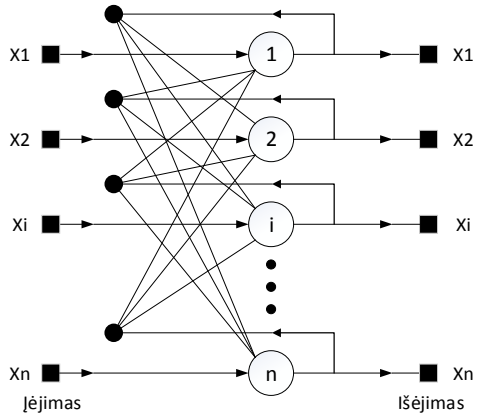
kur x_i – i-oji vektoriaus svorinių koeficientų komponentė, n – šio vektoriaus dimensija. Toks sprendimas leidžia sumažinti mokymo proceso laiką [**27**].

Kitas dažnai naudojamas mokymo be mokytojo metodas – varžytinių tipo metodas. Skirtingai nuo Hebbo mokymo taisyklės, kurioje daugybė išėjimo neuronų gali susižadinti vienu metu, varžytinių tipo metode išėjimo neuronai konkuruoja tarpusavyje už aktyvavimą. Šis reiškinys žinomas kaip „nugalėtojas pasiima viską“ (angl. *winner takes all*) taisyklė. Varžytinių metodas leidžia klasifikuoti duomenis. Mokymo procese modifikuojami tik neurono nugalėtojo svoriai [**27**].

2.6. Atvirkštinio skleidimo neuroniniai tinklai

Aukščiau buvo pateikti tiesioginio skleidimo NT. Juose visi signalai skleidžiami tik viena kryptimi – nuo įėjimo į išėjimą, bet ne atvirkščiai. Toks apribojimas nebūdingas biologiniams NT ir susiaurina NT modelio galimybes. Atvirkštinių sąryšių nebuvimas garantuoja tinklų stabilumą. Tačiau tokia pageidaujama savybė apriboja tiesioginio skleidimo NT galimybes palyginus su atvirkštinio skleidimo NT. Svarbų įnašą į atvirkštinio skleidimo NT teoriją ir panaudojamą padarė Džonas Hopfildas (angl. *John Hopfield*), todėl kai kurios NT konfigūracijos pavadintos jo vardu [**27**].

Struktūrinė Hopfildo tinklo schema pavaizduota 2.8 pav. Jis sudarytas iš vienintelio neuronų, kurių skaičius tuo pačiu metu yra tinklo įėjimai ir išėjimai, sluoksnio.



2.8 pav. Struktūrinė Hopfildo tinklo schema

Kiekvienas neuronas surištas sinapsėmis su visais kitais neuronais, o taip pat turi vieną įėjimo sinapsę, per kurią vykdomas signalo įėjimas. Išėjimo signalai formuojami aksonuose. Hopfildo tinklai gali funkcionuoti kaip asociatyvioji atmintis. Tai reiškia, kad priklausomai nuo vektoriaus, paduoto į NT įėjimą, išėjime bus suformuotas vienas iš įsimintų anksčiau vektorių, kuris labiausiai atitinka šiam paduotam į NT įėjimui vektoriui [**29**].

Pavyzdžiui, žinomas tam tikras dvejetainių signalų rinkinys (vaizdų, garso takelių, kitų, apibrėžiančių tam tikrus objektus arba procesų charakteristikas, duomenys), kurie priimami už etaloninius. Tinklas privalo iš bet kokio neidealaus (triukšmingo) signalo, paduoto į jo įėjimą, išryškinti („atsiminti“ pagal dalinę informaciją) atitinkamą originalų pavyzdį (jeigu toks egzistuoja) arba pateikti informaciją apie tai, kad įėjimo duomenys neatitinka nė vienam iš paduotų NT signalų. Bendru atveju, kiekvienas signalas gali būti apibrėžtas vektoriumi $X = \{x_i : i = 0 \dots (n-1)\}$, kur n – NT neuronų skaičius ir įėjimo ir išėjimo vektorių dimensija. Kiekvienas x_i elementas lygus arba +1 arba -1. Pažymėkime vektorių, apibrėžiantį k-tąjį objektą, kaip X^k, o jo komponentus atitinkamai kaip ${x_i}^k, k = 0 \dots (m-1)$, kur m – objektų (nagrinėjamų pavyzdžių) skaičius. Kai NT atpažins kokį nors iš pateiktų jai pavyzdžių, šio NT išėjimuose bus būtent pateiktas jam pavyzdys, t. y. $Y = X^k$, kur Y – NT išėjimo reikšmių vektorius: $Y = \{y_i : i = 0 \dots (n-1)\}$. Priešingu atveju, išėjimo vektorius nesutaps nė su vienų iš etaloninių vektorių.

NT inicializacijos stadijoje sinapsių svoriniai koeficientai nustatomi tokiu būdu:

$$w_{ij} = \begin{cases} \sum_{k=0}^{m-1} {x_i}^k {x_j}^k, i \neq j \\ 0, \ \ i \neq j \end{cases} \tag{2.35}$$

čia *i* ir *j* – priešsinapsinio ir postsinapsinio neurono indeksai; $x_i{}^k$, $x_j{}^k$ – *k*-ojo vektoriaus *i*-asis ir *j*-asis elementai.

1. Hopfildo NT funkcionavimo algoritmas (p – iteracijos numeris):

Į NT įėjimus paduodamas nežinomas signalas. Faktiškai jo įvedimas realizuojamas tiesioginiu aksonų reikšmių nustatymu:

$$y_i(0) = x_i, \;\; i = 0 \ldots (n-1) \tag{2.36}$$

Nulis skliaustuose dešinėje nuo y_i reiškia NT veikimo nulinę iteraciją.

2. Apskaičiuojama nauja neuronų būsena:

$$s_j(p+1) = \sum_{i=0}^{n-1} w_{ij} y_i(p), \;\; j = 0 \ldots (n-1) \tag{2.37}$$

ir naujos aksonų reikšmės:

$$y_j(p+1) = f\left[s_j(p+1)\right] \tag{2.38}$$

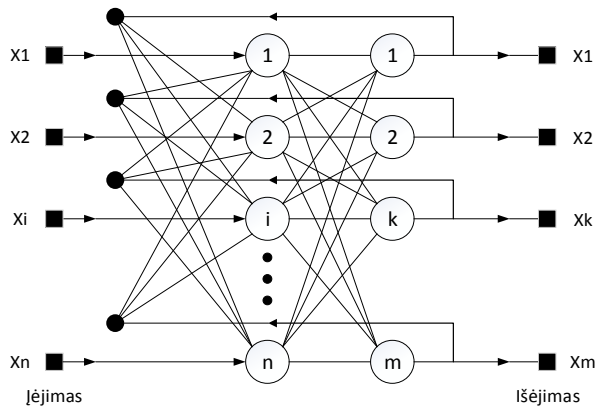
kur f – šuolio pavidalo aktyvavimo funkcija

3. Patikrinimas, ar pasikeitė aksonų išėjimo reikšmės paskutinėje iteracijoje. Jeigu taip – pereiname prie 2 punkto, priešingu atveju (jeigu išėjimai susistabilizavo) – pabaiga.

Išėjimo vektorius apibrėžia pavyzdį, geriausiai atitinkantį įėjimo duomenis.

Hopfildo tinklo įsimenamų klasių skaičius *m* negali viršyti 0,15*n*, kur *n* – tinklo neuronų skaičius. Be to, jeigu abi klasės *A* ir *B* labai panašios, gali atsitikti situacijų, kai paduodant į NT įėjimus vektorių *A*, jis bus traktuotas kaip vektorius *B* ir atvirkščiai.

Kitas atvirkštinio skleidimo NT pavyzdys – dvikryptė asociatyvioji atmintis (DAA). Faktiškai DAA – tai dviejų Hopfildo tinklo sluoksnių apjungimas (2.9 pav.) [**28**].



2.9 pav. DAA struktūrinė schema

DAA sugeba atlikti apibendrinimo uždavinius. Šis atvirkštinio skleidimo NT modelis sugeba įsiminti asocijuotų tarpusavyje klasių poras. Tegul klasių poros užrašomos vektorių pavidalu: $X^k = \{x_i{}^k : i = 0 \dots (n-1)\}$ ir $Y^k = \{y_j{}^k : j = 0 \dots (m-1)\}$, $k = 0 \dots (r-1)$, kur r – porų skaičius. Paduodant į pirmąjį NT sluoksnį tam tikrą vektorių $P = \{p_i : i = 0 \dots (n-1)\}$, antrojo sluoksnio išėjime generuojamas kitas vektorius $Q = \{q_j : j = 0 \dots (m-1)\}$, kuris po to vėl paduodamas į pirmojo sluoksnio įėjimą. Kiekvienoje tokioje iteracijos stadijoje, abejų sluoksnių vektoriai artėja prie etaloninių vektorių poros, pirmas iš kurių – X – labiau atitinka P, kuris buvo paduotas į NT įėjimą pačioje pradžioje, o kitas – Y – asocijuotas su juo. Asociacijos tarp vektorių koduojamos pirmojo sluoksnio svorių matricoje $W^{(1)}$. Antrojo sluoksnio svorinė matrica $W^{(2)}$ lygi pirmajai transponuotai matricai $(W^{(1)})^T$. Mokymo procesas, taip pat kaip ir Hopfildo tinklo atveju, remiasi išankstiniu W (ir atitinkamai W^T) matricos elementų apskaičiavimu pagal formulę:

$$w_{ij} = \sum_k x_i y_j, \;\; i = 0 \dots (n-1), \;\; j = 0 \dots (m-1) \tag{2.39}$$

Ši formulė – tai išplėsta matricinės lygties forma:

$$W = \sum_k X^T Y \tag{2.40}$$

Kaip ir Hopfildo NT, DAA turi maksimalių asocijuojamų klasių, kurias ji gali tiksliai reprodukuoti, skaičiaus limitą. Jeigu viršytas šis limitas, DAA gali sugeneruoti nekorektišką išėjimo signalą, atkurdama asociacijas, kurioms nebuvo apmokyta [**27**].

Turint N neuronų mažiausiame sluoksnyje, DAA atminties talpą galima apskaičiuoti pagal formulę:

$$L < \frac{N}{log_2 N} \tag{2.41}$$

kur L – DAA talpa.

DAA talpa nėra didelė. Turint $N = 1024$ neuronus, NT gali atsiminti $L < 25$ klasių [**28**].

3. DDoS atakų aptikimas naudojant neuroninius tinklus

3.1. Analizuojamų duomenų formavimas

Tinklo anomalijų aptikimas – tai klasifikavimo uždavinys. Klasifikavimo uždaviniui spręsti naudojamas mokymas su mokytoju, todėl reikia pasirinkti tokį NT tipą, kuris skirtas būtent tokiam mokymo variantui. Dažniausiai šiam tikslui naudojamas daugiasluoksnis perceptronas. Klasifikavimo tikslas – nustatyti, kuriai iš anksto apibrėžtų klasių priklauso įėjimo objektas (pvz., tinklo paketas arba sesija), apibūdintas požymių vektoriumi. Sprendžiant klasifikavimo uždavinį, kiekviena tinklo sesija apibrėžiama kaip vektorius:

$$x = \begin{pmatrix} x_1 \\ \dots \\ x_N \end{pmatrix} \tag{3.1}$$

kur $x_1..x_N$ – *KDD Cup 1999* duomenų rinkinio parametrai, apibrėžiantys tinklo sesiją. Keičiant įėjimo vektoriaus *x* dimensiją (keičiant *KDD Cup 1999* duomenų rinkinio atributų skaičių), galima nustatyti kaip NT išėjimo duomenys priklauso nuo įėjimo duomenų. Originaliame *KDD Cup 1999* duomenų rinkinyje kiekvienai sesijai atitinka vienas įrašas, sudarytas iš 41 parametro ir 1 papildomo parametro, nustatančio, kuriai iš tinklo anomalijų klasių priklauso duotoji sesija. Tinklo sesija gali būti klasifikuota kaip teisėta arba anomalinė (priklausyti vienai iš kelių *DoS* atakų klasių), todėl tinklo sesiją sudaro $Y_1..Y_M$ klasių aibė, kur *M – DoS* atakų klasių skaičius.

Remiantis *x* vektoriumi reikia nuspręsti, kuriai klasei priskirti tinklo sesiją, t. y. iš klasių aibės $Y_1..Y_M$ išrinkti tokį Y_i, kuriam priklauso tinklo sesija, apibrėžta parametrų rinkiniu *x*. Užduoties sprendimą galima pateikti kito vektoriaus pavidalu:

$$y = \begin{pmatrix} y_1 \\ \dots \\ y_M \end{pmatrix} \tag{3.2}$$

Tikslas: apmokyti NT generuoti savo išėjime vektorių *y*, kai į įėjimus paduodamas vektorius *x*, t. y. spręsti funkciją $y = f(x)$.

Kalbant apie mokymą su mokytoju metodą, NT naudojimas bet kokio uždavinio sprendimui sudarytas iš dviejų etapų: mokymosi etapas ir atpažinimo etapas [**5**], [**25**], [**33**]. Mokymosi etape NT įėjimui paduodamas mokymo išrankiojimas, sudarytas iš aibės iš anksto atrinktų ir paruoštų vektorių. NT išėjime formuojamas išėjimo vektorius ir tikrinami svorio koeficientai. Jeigu svorio koeficientų paklaida yra priimtina, NT yra apmokytas, priešingu atveju pereinama prie kitos mokymo iteracijos, kurios metu atliekamas svorio koeficientų koregavimas.

Atpažinimo etape į neuroninį tinklą patenka iš anksto nežinomas įėjimo vektorius. Išėjime atsiranda kitas vektorius – atpažinimo rezultatas, pagal kurį įėjimo vektorius priskiriamas vienai iš žinomų klasių.

Tokiu būdu, naudojant NT aptikti *DDoS* atakas, kiekviena tinklo sesija turi būti pateikta kaip žymių vektorius, kuris paduodamas į neuroninio tinklo įėjimą. Po signalo praėjimo per tinklą išėjime formuojamas vektorius, nustatantis ar tinklo sesija yra *DDoS* atakos dalis.

Dažniausiai NT mokymo ir testavimo tikslams eksperimentiniai duomenys išskaidomi į du duomenų rinkinius (angl. *dataset*) – mokymo ir testavimo [**5**].

- Mokymo duomenų rinkinys skirtas NT mokymui.
- Testavimo duomenų rinkinys suteikia galimybę patikrinti, kiek gerai yra apmokytas NT.

Duomenų rinkinį sudaro tinklo parametrų sąrašas, kuriame egzistuoja „normalaus“ ir „kenksmingo“ tinklo srauto paketų pavyzdžiai. Šie rinkiniai konvertuojami į įėjimo vektorius (angl. *feature vectors*). Šiame darbe įėjimo vektorius – tai duomenys, aprašantys tinklo sesiją (kaip teisėtą taip ir anomalinę).

Žodyno (apmokymo duomenų rinkinio) ir įėjimo vektoriaus sukūrimo pavyzdys [**34**].

Teisėta užklausa:

```
0.0.0.0 - - [20/Dec/2011:15:00:03 +0400] "GET /forum/rss.php?topic=347425
HTTP/1.0" 200 1685 "-" "Mozilla/5.0 (Windows; U; Windows NT 5.1; pl; rv:1.9)
Gecko/2008052906 Firefox/3.0"
```

Anomalinė užklausa:

```
0.0.0.0 - - [20/Dec/2011:20:00:08 +0400] "POST /forum/index.php HTTP/1.1" 503
107 "http://www.mozilla-europe.org/" "-"
```

Duomenų rinkinys:

```
['__UA___OS_U', '__UA_EMPTY', '__REQ___METHOD_POST',
'__REQ___HTTP_VER_HTTP/1.0', '__REQ___URL___NETLOC_',
'__REQ___URL___PATH_/forum/rss.php', '__REQ___URL___PATH_/forum/index.php',
'__REQ___URL___SCHEME_', '__REQ___HTTP_VER_HTTP/1.1',
'__UA___VER_Firefox/3.0', '__REFER___NETLOC_www.mozilla-europe.org',
'__UA___OS_Windows', '__UA___BASE_Mozilla/5.0', '__CODE_503', '__UA___OS_pl',
'__REFER___PATH_/', '__REFER___SCHEME_http', '__NO_REFER__',
'__REQ___METHOD_GET', '__UA___OS_Windows NT 5.1', '__UA___OS_rv:1.9',
'__REQ___URL___QS_topic', '__UA___VER_Gecko/2008052906']
```

Testinė užklausa:

```
0.0.0.0 - - [20/Dec/2011:20:00:01 +0400] "GET /forum/viewtopic.php?t=425550
HTTP/1.1" 502 107 "-" "BTWebClient/3000(25824)"
```

Testinės užklausos vektorius:

```
[0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0]
```

Pagrindinis principas suteikiantis galimybę aptikti anomaliją, tai kelių faktorių įvertinimas, tokių kaip: laiko intervalas tarp vienodų užklausų; tinklo paketų, siunčiamų į vieną prievadą, skaičius; tinklo paketų, turinčių nekorektišką antraštę, skaičius; bendras tinklo paketų skaičius, būdingas įprastiniam tam tikro tinklo serviso veikimui [**35**].

Testavimo duomenų rinkinys gali būti suformuotas iš įsilaužimų aptikimo sistemos žurnalų failų arba iš specialiai perengtos duomenų bazės *KDD Cup 1999* [**36**] ir *NSL-KDD 2012* [**37**] (3.1 lentelė). *KDD Cup 1999* duomenų rinkinys pirmą kartą buvo pademonstruotas V Tarptautinėje konferencijoje „Knowledge Discovery and Data Mining“. Masačiusetso technologijos instituto Linkolno laboratorija (angl. *Massachusetts Institute of Technology Lincoln Labs*) sukūrė infrastruktūrą, imituojančią JAV karinių oro pajėgų (angl. *USA Air Force*) lokalų tinklą. Duomenys buvo surenkami taip, lyg tai iš tikrųjų būtų vykdomos įvairios atakos į JAV karinių pajėgų lokalų tinklą. Šis duomenų rinkinys buvo sugeneruotas naudojant *tcpdump* duomenis iš *1998 DARPA IDS* žurnalų failų, kuriuose saugoma 9 savaičių laikotarpio tinklo srauto informacija. Gauti duomenys buvo konvertuoti į apytiksliai 5000000 įrašus, saugančius tinklo sujungimų informaciją [**38**].

3.1 lentelė. KDD Cup 1999 (NSL-KDD 2012) DB atributai

Nr.	Atributas	Nr.	Atributas	Nr.	Atributas
1	duration	15	su_attempted	29	same_srv_rate
2	protocol_type	16	num_root	30	diff_srv_rate
3	service	17	num_file_creations	31	srv_diff_host_rate
4	flag	18	num_shells	32	dst_host_count
5	src_bytes	19	num_access_files	33	dst_host_srv_count
6	dst_bytes	20	num_outbound_cmds	34	dst_host_same_srv_rate
7	land	21	is_host_login	35	dst_host_diff_srv_rate
8	wrong_fragment	22	is_guest_login	36	dst_host_same_src_port_rate
9	urgent	23	count	37	dst_host_srv_diff_host_rate
10	hot	24	srv_count	38	dst_host_serror_rate
11	num_failed_logins	25	serror_rate	39	dst_host_srv_serror_rate
12	logged_in	26	srv_serror_rate	40	dst_host_rerror_rate
13	num_compromised	27	rerror_rate	41	dst_host_srv_rerror_rate
14	root_shell	28	srv_rerror_rate		

Šios dvi DB turi vienodą atributų skaičių. *NSL-KDD 2012* tai patobulintas duomenų rinkinys, kuriame buvo išspręsta dalis problemų, būdingų *KDD Cup 1999* duomenų rinkiniui. Kiekviena iš šių DB tai tekstinis failas, kuriame saugomas atakų ir normalaus tinklo srauto parametrų rinkiniai. Kiekvienas šių DB įrašas – tai tinklo sujungimo pavyzdys, kurį sudaro 41 tinklo srauto parametras [**8**], ir kuris pažymėtas kaip „ataka“ arba „ne ataka“. *KDD Cup 1999* ir *NSL-KDD 2012* atakos skirstomos į 4 kategorijas (3.2 lentelė):

- *DoS* – atsisakymas aptarnauti.
- *R2L* – nesankcionuota prieiga iš nutolusio kompiuterio.
- *U2R* – nesankcionuota prieiga siekiant gauti lokalaus *root* vartotojo teises.
- *Probing* – prievadų (angl. *port*) skenavimas.

3.2 lentelė. KDD Cup 1999 (NSL-KDD 2012) atakų klasifikacija [**36**]

Atakos tipas	Probing	DoS	U2R	R2L
Atakos pavadinimas	ipsweep	back	rootkit	ftp-write
	nmap	land	perl	spy
	portsweep	neptune	loadmodule	phf
	satan	pod	buffer-overflow	guess-passwd
		smurf		imap
		teardrop		warezclient
				warezmaster
				multihop

Originaliame *KDD Cup 1999* duomenų rinkinyje egzistuoja tokie įrašų tipai kaip: „protocol type“ su reikšmėmis „UDP“, „TCP“ ir požymis „Flag“ su reikšmėmis „SF“, „REJ“, „RSTR“ ir t.t. Jeigu planuojama naudoti šiuos laukus NT, reikia konvertuoti atitinkamas tekstines reikšmes į skaičius. Duomenų rinkinio atributų konvertavimas į skaičius atliekamas pagal tam tikras taisykles. Pirmam testavimo atvejui tekstinės atributo reikšmės keičiamos remiantis stulpeliu „Skaitinė vertė 1“. Antram atvejui atributo reikšmės keičiamos remiantis stulpeliu „Skaitinė vertė 2“ (A.1 lentelė).

Kadangi šio darbo tikslas – *DDoS* atakų aptikimas, lentelėje pateikti tik tų anomalijų tipai, kurie būdingi šioms atakoms, o kiti tipai yra ignoruojami. Prieš pradedant NT efektyvumo nustatymą, iš *KDD Cup 1999* duomenų rinkinio yra naikinami įrašai (tinklo sesijos), kurie neapibrėžia *DoS* atakų (3.2 lentelė), t. y. *KDD Cup 1999* variantas, kuris naudojamas NT *DDoS* atakų aptikimo efektyvumo nustatymui, bus sudarytas tik iš įrašų, apibrėžiančių šešis tinklo srauto anomalijų tipus: *back, land, neptune, pod, smurf, teardrop*.

Tyrimuose, kaip pagrindas, naudojamas 10 % nuo bendro *KDD Cup 1999* duomenų rinkinio įrašų sudarantis variantas (3.3 lentelė).

3.3 lentelė. 10 % KDD Cup 1999 duomenų rinkinio įrašų skaičius

Tinklo srauto klasė	**Įrašų skaičius**
normal	97278
back	2203
land	21
neptune	107201
pod	264

Tinklo srauto klasė	Įrašų skaičius
smurf	280790
teardrop	979
Iš viso	488736

Tam, kad sumažintume tyrimų vykdymo laiką ir aparatinės įrangos apkrovimą, iš šio duomenų rinkinio yra atrinkti tik 33467 įrašai. Rinkinys suformuotas atrenkant po 10000 kiekvienos klasės įrašus. Skaičius tų klasių, kurių yra mažiau už 10000, nesikeitė. Šiame darbe naudojami du *DDoS* atakų aptikimo variantai (A.1 lentelė) analizuojant *KDD Cup 1999* duomenų rinkinį:

1. Įrašai suskaidyti į septynias klases (3.4 lentelė).
2. Šeši anomalijų tipai priskiriami vienai klasei, apibrėžiančiai ataką (3.5 lentelė). Įrašai suskaidyti į dvi klases.

3.4 lentelė. Atakos suskaidytos į 6 klases

Tinklo srauto klasė	Įrašų skaičius
normal	10000
back	2203
land	21
neptune	10000
pod	264
smurf	10000
teardrop	979
Iš viso	33467

3.5 lentelė. Atakos priskirtos vienai anomalijų klasei

Tinklo srauto klasė	Įrašų skaičius
normal	10000
attack	23467
Iš viso	33467

Kiekvieno *KDD Cup 1999* (*NSL-KDD 2012*) duomenų rinkinio įrašo (vektoriaus), elementai konvertuojami į atitinkamas skaitines tam, kad NT galėtų juos apdoroti. Pavyzdžiui, turime vektorių:

{0,icmp,ecr_i,SF,1032,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,510,510,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,255,1.00,0.00,1.00,0.00,0.00,0.00,0.00,0.00,smurf}.

Kiekvienas neskaitinis šio vektoriaus atributas paverčiamas skaitiniu (A.1 lentelė):

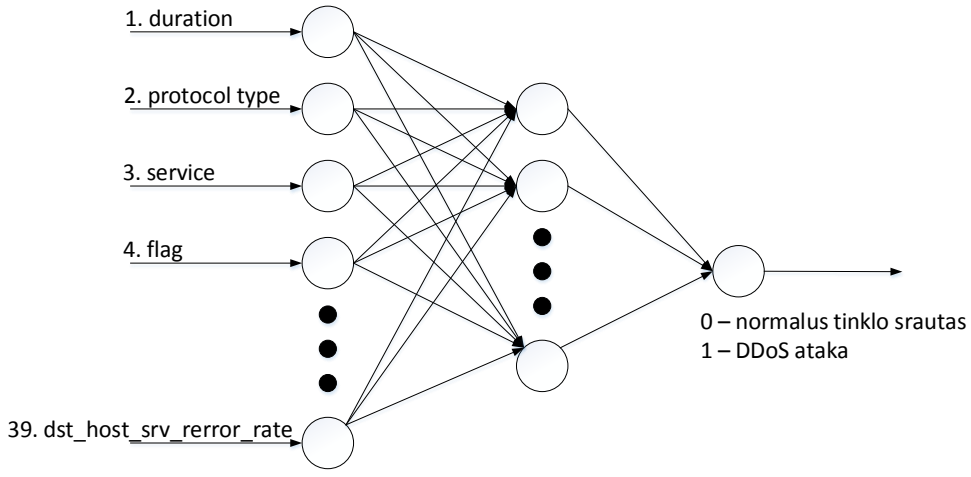
{0,4,27,14,1032,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,510,510,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,255,1.00,0.00,1.00,0.00,0.00,0.00,0.00,0.00,1}.

Sprendžiant pirmą klasifikavimo uždavinį, tinklo anomalijos nustatymo vektorius (visi galimi *DoS* atakos aptikimo užduoties sprendimo variantai) sudarytas iš septynių elementų: {0,1,2,3,4,5,6}.

Sprendžiant antrą klasifikavimo uždavinį, tinklo anomalijos nustatymo vektorius sudarytas iš dviejų elementų: {0,1}. Kadangi kiekvienas iš aukščiau išvardintų anomalijų tipų apibrėžia *DoS* ataką, priskiriame visus šešis anomalijų tipus vienai klasei.

DDoS atakų aptikimui reikalingi tik tie įrašai, kurie priklauso vienam iš 6 aukščiau išvardintų anomalijų tipų: *back, land, neptune, pod, smurf, teardrop*, o taip pat pažymėti kaip *normal*. Kiti anomalijų tipai apibrėžia kitus atakų tipus ir neturi įtakos pagrindinio šio darbo tikslo nustatymui, todėl juos galima panaikinti. Nereikšminių įrašų panaikinimas sumažins bendrą duomenų rinkinio įrašų skaičių ir pagreitins NT apmokymo procesą.

Išnagrinėjus šaltinius [**23**], [**28**], [**29**], [**33**], [**34**], [**39**], kad būtų galima įgyvendinti pagrindinį šio darbo tikslą, yra naudojami 2 skirtingi NT modeliai, skirti klasifikacijos uždaviniams spręsti: DSP ir SBF. Abu NT modeliai tiriami, naudojant skirtingus įėjimo parametrus. 3.1 pav. pavaizduotas vienas iš naudojamų NT variantų – daugiasluoksnis (2 sluoksniai) perceptronas. Šis NT modelis yra gerai aprašytas literatūroje ir jo programinė realizacija yra sąlyginai paprasta palyginus su kitų NT modeliais. NT turi 39 įėjimus ir 1 išėjimą. Kiekvienam NT įėjimui atitinka vienas *KDD Cup 1999* duomenų rinkinio požymių. Analizuojamo tinklo srauto rezultatas gali turėti 2 atsakymus: anomalinis tinklo srautas (*DDoS* ataka) arba teisėtas tinklo srautas.



3.1 pav. NT, skirto aptikti DDoS atakas, schema

DDoS atakos nustatymui, analizuojant *KDD Cup 1999* arba *NSL-KDD 2012* duomenų rinkinį, reikia apibrėžti kriterijus, remiantis kuriais analizuojamas duomenų rinkinio įrašas (sesija) bus traktuojamas kaip ataka (3.6 lentelė). Sesija – *TCP* paketų seka, išsiųsta per tam tikrą baigtinį laiko momentą, kurio pradžios ir pabaigos intervalai yra griežtai apibrėžti, ir kurio metu duomenys perduodami iš vieno *IP* adreso į kitą, naudojant nustatytą protokolą [**8**].

3.6 lentelė. NSL-KDD duomenų rinkinio įrašai

label	**service**	**flag**	**host_count**	**srv_count**	**duration**
normal	ecr_i	SF	1	1	0
smurf	ecr_i	SF	350	350	0

Kriterijai, pagal kuriuos nustatoma ataka, yra aprašomi taip:

```
smurf: service = ecr_i, host_count >= 5, host_srv_count >= 5
```

Jeigu serviso tipas yra *ICMP echo request*, prisijungimų per pastarąsias 2 sekundes prie to paties tinklo mazgo skaičius yra nemažiau 5 ir prisijungimų prie to paties serviso skaičius yra ne mažiau 5, tai tokia sesija yra traktuojama kaip *smurf* (*DDoS*) atakos dalis.

3.2. Tyrimų aplinka

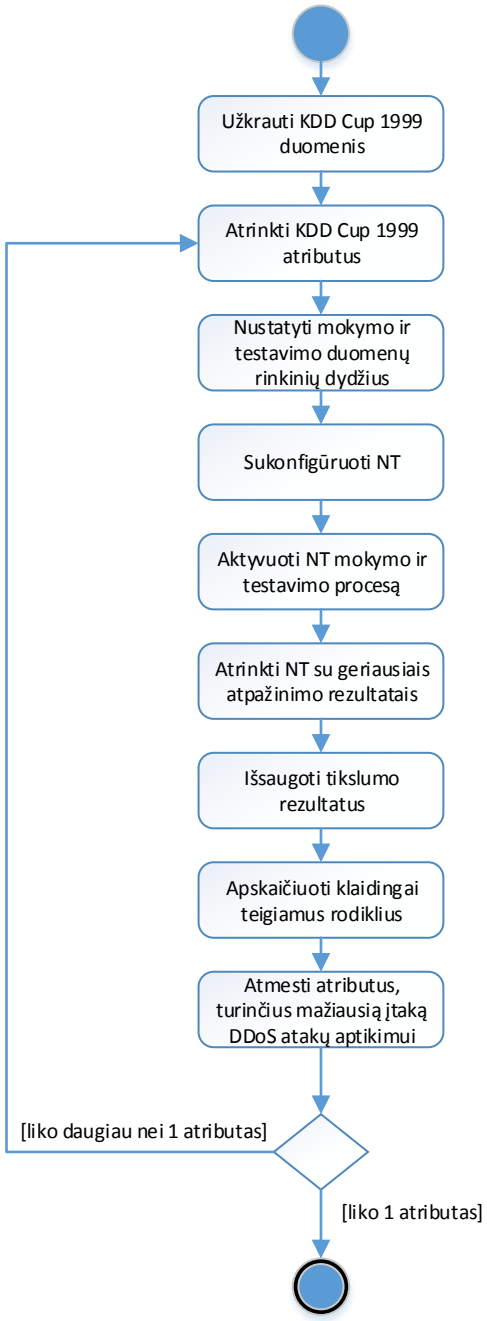
Šiame darbe, nustatant NT efektyvumą aptikti *DDoS* atakas, naudojamas *StatSoft Statistica 8.0.360.0* programinis paketas. Šis programinis paketas klasifikavimo užduotims spręsti leidžia naudoti DSP ir SBF. Pasinaudosime *Automated Network Search (ANS)* modulio funkcionalumu, kuris suteikia galimybę automatiškai parinkti optimalias NT konfigūracijas duotam uždaviniui spręsti, remiantis nurodytais nustatymais ir leidžia išvengti daug atskirų konfigūracijų perrinkimo rankiniu būdu, siekiant parinkti optimalią NT konfigūraciją.

Atliekant tyrimus siekiama parodyti kaip skiriasi *DDoS* atakų atpažinimo efektyvumas priklausomai nuo *KDD Cup 1999* duomenų rinkinio naudojamų atributų skaičiaus. Pagrindinis tikslas – rasti tokį atributų kiekį, kurį naudojant bus pasiektas optimalus *DDoS* atakų atpažinimo ir NT apmokymo laiko santykis. Bandymai atliekami naudojant skirtingus NT modelius. Tai suteiks informacijos apie tai, kuris NT modelis geriau sprendžia *DDoS* atakų aptikimo problemą. Žemiau pateikiami *ANS* įrankio nustatymai (3.7 lentelė).

3.7 lentelė. ANS įrankio nustatymai, naudojami tyrimuose

Parametras	**Reikšmė**
Analizės tipas	klasifikavimas
DSP paslėpto sluoksnio neuronų skaičius	nuo 10 iki 30
SBF tinklo paslėpto sluoksnio neuronų skaičius	nuo 20 iki 60
Paslėpto sluoksnio aktyvavimo funkcijos	logistinė, hiperbolinis tangentas
Išėjimo sluoksnio aktyvavimo funkcijos	logistinė, hiperbolinis tangentas
Paslėpto sluoksnio neuronų svorių reguliarizacija	nuo 0,001 iki 0,01
Išėjimo sluoksnio neuronų svorių reguliarizacija	nuo 0,001 iki 0,01

3.2 pav. pavaizduotas tyrimų procesas.



3.2 pav. Tyrimų procesas

Eksperimentiniai duomenys kraunami į *StatSoft Statistica* programinio paketo *ANS* modulį iš *MS SQL Server 2012* DB. 70 % visų eksperimentinio duomenų rinkinio įrašų naudojami mokymui, 30 % įrašų naudojami apmokyto NT testavimui. Mokymo ir testavimo įrašai pasirenkami atsitiktiniu būdu. To reikia tam, kad mokymo ir testavimo įrašai nesikartotų, nes NT gebėjimą teisingai klasifikuoti duomenis būtina tikrinti paduodant jam nežinomus (nedalyvavusius mokyme) pavyzdžius.

Kiekvieno tyrimo etape kiekvienam NT modeliui (DSP ir SBF) automatiškai generuojami penki tinklai, iš kurių atrenkamas vienas, turintis geriausius atpažinimo rezultatus. Jeigu yra du ar daugiau NT, kurių atpažinimo rezultatai vienodi, atrenkamas mažiausiai neuronų tarpiniame sluoksnyje turintis NT. Tyrimuose naudojamos 8 skirtingos požymių vektorių dimensijos, kurių dydis kiekviename tyrimo etape mažėja: 39, 30, 20, 10, 5, 3, 2, 1. Požymių vektorius sudaromas iš *KDD Cup 1999* duomenų rinkinio atributų aibės. Pagal šiuos požymius atliekama tinklo srauto analizė. Kiekviename etape atmetami tie požymiai, kurie turi mažiausią įtaką galutiniam tinklo anomalijos atpažinimo rezultatui. Požymių įtaka ir tinklo anomalijų atpažinimo tikslumas nustatomi *ANS* modulio pagalba. Tokiu būdu tyrimų eigoje nustatomas minimalus *KDD Cup 1999* duomenų rinkinio atributų skaičius, suteikiantis priimtiną *DDoS* atakų aptikimo lygį. Remiantis gautais rezultatais, kuriami grafikai, kuriuose pateikiami NT tikslumo ir klaidingai teigiami tinklo anomalijų aptikimo rodikliai.

3.3. Neuroninių tinklų efektyvumas aptinkant tinklo anomalijas

DSP (3.8 lentelė) ir SBF (3.9 lentelė) testavimo rezultatai, naudojant 7 klasifikavimo kategorijas.

3.8 lentelė. DSP tikslumas, 7 klasės

Atributų skaičius	NT Pavadinimas	Mokymo tikslumas, %	Testavimo tikslumas, %	Mokymo algoritmas	Klaidos funkcija	Paslėptų neuronų aktyvavimo funkcija	Išėjimo neuronų aktyvavimo funkcija
39	MLP 39-13-7	100,0000	99,9701	BFGS 47	Entropy	Logistic	Softmax
30	MLP 30-16-7	99,9872	99,9602	BFGS 62	Entropy	Tanh	Softmax
20	MLP 20-22-7	99,9957	99,9602	BFGS 86	Entropy	Tanh	Softmax
10	MLP 10-23-7	99,9872	99,9203	BFGS 58	Entropy	Logistic	Softmax
5	MLP 5-16-7	94,5362	94,9004	BFGS 102	Entropy	Tanh	Softmax
3	MLP 3-16-7	94,2203	94,4920	BFGS 118	Entropy	Logistic	Softmax
2	MLP 2-12-7	92,4147	92,8984	BFGS 96	Entropy	Logistic	Softmax
1	MLP 1-14-7	62,5347	63,1574	BFGS 66	Entropy	Logistic	Softmax

DSP atpažinimo lygis viršija 90 % net tada, kai įėjimo sluoksnį sudaro tik 2 neuronai. Kai įėjimo sluoksnis turi tik 1 neuroną, DSP atpažinimo lygis sumažėja iki 62,5347 %.

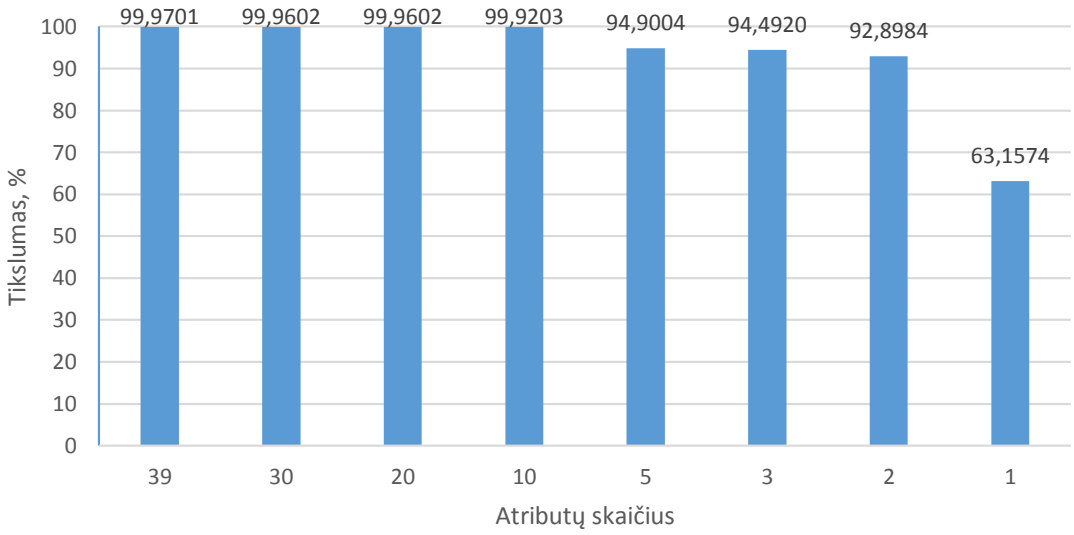
3.9 lentelė. SBF tikslumas, 7 klasės

Atributų skaičius	NT Pavadinimas	Mokymo tikslumas, %	Testavimo tikslumas, %	Mokymo algoritmas	Klaidos funkcija	Paslėptų neuronų aktyvavimo funkcija	Išėjimo neuronų aktyvavimo funkcija
39	RBF 39-26-7	87,7108	88,3466	RBFT	Entropy	Gaussian	Softmax
30	RBF 30-30-7	85,3332	85,6574	RBFT	Entropy	Gaussian	Softmax
20	RBF 20-25-7	77,4961	77,6594	RBFT	Entropy	Gaussian	Softmax
10	RBF 10-31-7	77,0820	77,2311	RBFT	Entropy	Gaussian	Softmax
5	RBF 5-50-7	53,5152	54,5319	RBFT	Entropy	Gaussian	Softmax

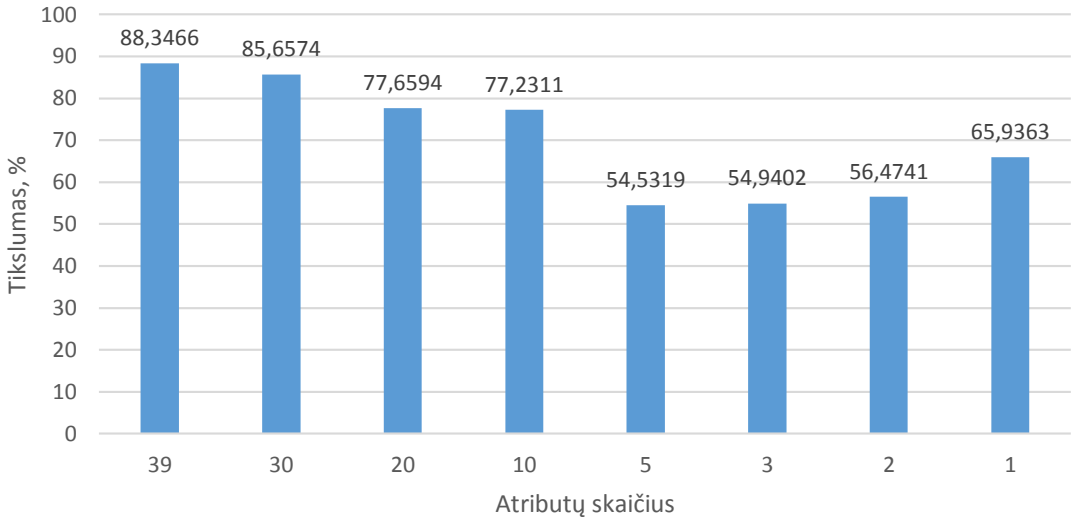
Atributų skaičius	NT Pavadinimas	Mokymo tikslumas, %	Testavimo tikslumas, %	Mokymo algoritmas	Klaidos funkcija	Paslėptų neuronų aktyvavimo funkcija	Išėjimo neuronų aktyvavimo funkcija
3	RBF 3-56-7	53,9292	54,9402	RBFT	Entropy	Gaussian	Softmax
2	RBF 2-54-7	55,5427	56,4741	RBFT	Entropy	Gaussian	Softmax
1	RBF 1-44-7	66,0819	65,9363	RBFT	Entropy	Gaussian	Softmax

SBF suteikia žemesnį atpažinimo lygį palyginus su DSP. Naudojant 39 įėjimo parametrus, SBF tinklo tikslumas sudaro 88,3466 %, palyginus su 99,9701 % DSP rezultatu. Sumažinus įėjimo sluoksnio neuronų skaičių iki 20, šio tinklo atpažinimo lygis krenta iki 77,6594 %. Kai paduodamų parametrų skaičius sumažėja iki 5, SBF tikslumas sudaro tik 54,5319 % ir beveik nesikeičia mažinant paduodamų parametrų skaičių iki 1.

3.3 pav. ir 3.4 pav. pavaizduotas DSP ir SBF tikslumas.

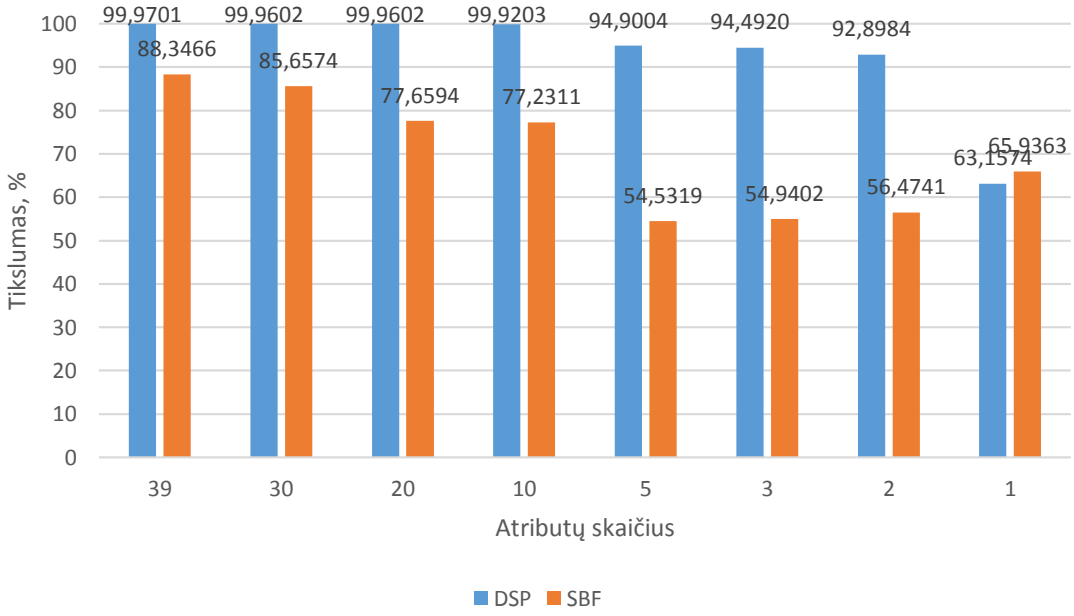


3.3 pav. DSP tikslumas. 7 klasės



3.4 pav. SBF tikslumas, 7 klasės

3.5 pav. pateikiamas DSP ir SBF tikslumo palyginimas, naudojant 7 klases.



3.5 pav. DSP ir MLP tikslumo palyginimas, 7 klasės

Klaidingai teigiami DSP (3.10 lentelė) ir SBF (3.11 lentelė) klasifikavimo rodikliai.

3.10 lentelė. DSP klaidingai teigiami DDoS atakų aptikimo rezultatai, 7 klasės

		normal	**back**	**land**	**neptune**	**pod**	**smurf**	**teardrop**
39	MLP 39-13-7	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
30	MLP 30-16-7	0,0143	0,1264	0,0000	0,0000	0,0000	0,0000	0,0000
20	MLP 20-22-7	0,0000	0,0632	0,0000	0,0000	0,0000	0,0000	0,0000
10	MLP 10-23-7	0,0000	0,1264	8,3333	0,0000	0,0000	0,0000	0,0000
5	MLP 5-16-7	2,5604	69,5954	0,0000	0,0000	0,0000	0,0000	0,0000

		normal	back	land	neptune	pod	smurf	teardrop
3	MLP 3-16-7	2,4889	73,0088	75,0000	0,0000	2,8571	0,1562	0,0000
2	MLP 2-12-7	0,5149	98,5461	58,3333	0,0000	100,0000	0,0000	0,0000
1	MLP 1-14-7	90,6308	100,0000	75,0000	0,0000	100,0000	0,0000	100,0000

Naudojant 39 įėjimo parametrus, DSP negeneruoja nė vieno klaidingai teigiamo rezultato. Sumažinus DSP įėjimų skaičių iki 20, klaidingai teigiami *DDoS* atakų aptikimo rezultatai neviršija 1 %. Kai įėjimo sluoksnio neuronų skaičius sumažėja iki 5, DSP klaidingai teigiamų *back DDoS* atakų tipo aptikimo rezultatų rodiklis sudaro 69,5954 %, tačiau kitų *DDoS* atakos klasių klaidingai teigiamų aptikimo rezultatų tikimybė sudaro 0 %, o normalaus tinklo srauto aptikimo rezultatai sudaro tik 2,5604 %. *Neptune* – vienintelė *DDoS* atakų klasė, kurią DSP visada traktuoja teisingai (0 % klaidingai teigiamų rezultatų), nepriklausomai nuo įėjimo parametrų skaičiaus. *Smurf DDoS* atakų klasė taip pat beveik visada nustatoma teisingai, išskyrus atvejį, kai DSP pateikiami 3 įėjimo parametrai, tačiau net tada šios atakos klaidingai teigiamų rezultatų rodiklis sudaro tik 0,1562 %.

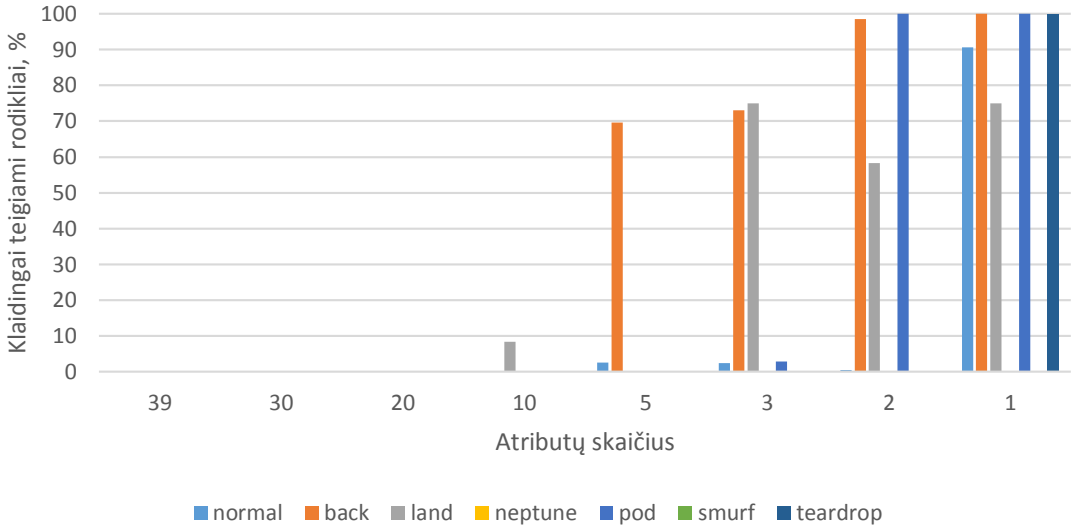
3.11 lentelė. SBF klaidingai teigiami DDoS atakų aptikimo rezultatai, 7 klasės

		normal	back	land	neptune	pod	smurf	teardrop
39	RBF 39-26-7	15,9920	96,2705	100,0000	0,0288	100,0000	0,6959	0,0000
30	RBF 30-30-7	16,5212	86,3464	100,0000	0,0575	100,0000	0,6959	100,0000
20	RBF 20-25-7	39,5079	99,6839	100,0000	1,0214	100,0000	0,0000	100,0000
10	RBF 10-31-7	38,8643	96,4602	100,0000	3,7117	100,0000	0,0852	100,0000
5	RBF 5-50-7	21,8567	100,0000	100,0000	0,0144	29,1429	100,0000	100,0000
3	RBF 3-56-7	27,6355	100,0000	100,0000	0,0432	41,1429	99,9432	22,9630
2	RBF 2-54-7	13,2742	100,0000	100,0000	0,0288	100,0000	100,0000	100,0000
1	RBF 1-44-7	80,6036	100,0000	100,0000	0,0863	100,0000	0,2272	77,0370

SBF klaidingai teigiami rodikliai kardinaliai skiriasi nuo DSP. Net kai naudojami 39 įėjimo parametrai, šis NT modelis nesugeba teisingai nustatyti nė vienos *land* ir *pod* tipų *DDoS* atakų. Geriausiai aptinkama *DDoS* atakų klasė, kaip ir DSP atveju, yra *neptune*. Šios atakos klasės klaidingai teigiami aptikimo rezultatai neviršija 3,7117 %.

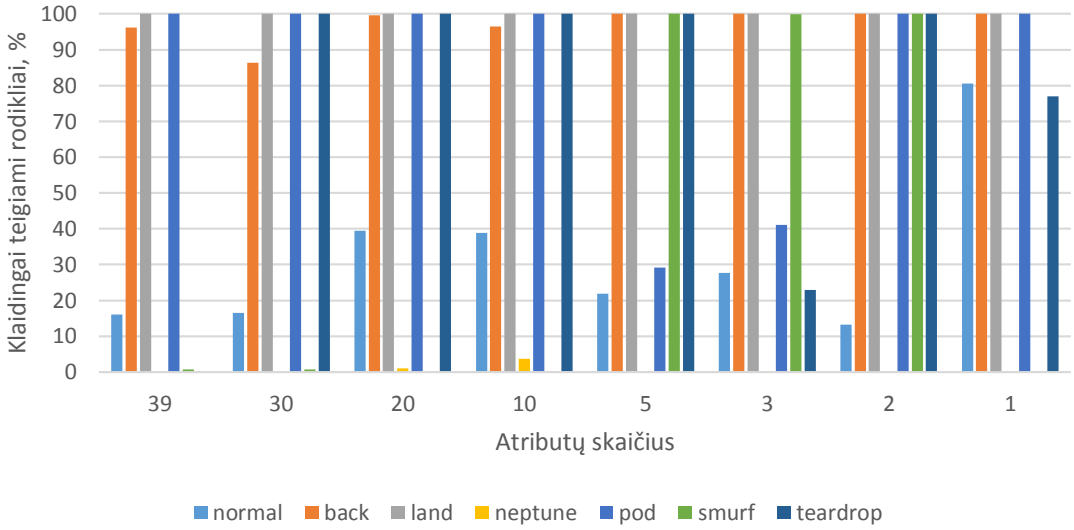
Akivaizdu, kad patiektos SBF tinklų konfigūracijos netinka *DDoS* atakų aptikimui dėl labai didelių klaidingai teigiamų aptikimo rodiklių.

3.6 pav. pateikiami DSP klaidingai teigiami *DDoS* atakų aptikimo rezultatai.



3.6 pav. DSP. Klaidingai teigiami DDoS atakų aptikimo rezultatai, 7 klasės

3.7 pav. pateikiami SBF klaidingai teigiami *DDoS* atakų aptikimo rezultatai.



3.7 pav. SBF. Klaidingai teigiami DDoS atakų aptikimo rezultatai, 7 klasės

Žemiau pateikiami DSP (3.12 lentelė) ir SBF (3.13 lentelė) testavimo rezultatai, naudojant 2 klasifikavimo kategorijas, t. y. kai visi *KDD Cup 1999* duomenų rinkinio *DDoS* atakų tipai yra priskirti vienai *attack* klasei.

3.12 lentelė. DSP tikslumas, 2 klasės

Atributų skaičius	NT Pavadinimas	Mokymo tikslumas, %	Testavimo tikslumas, %	Mokymo algoritmas	Klaidos funkcija	Paslėptų neuronų aktyvavimo funkcija	Išėjimo neuronų aktyvavimo funkcija
39	MLP 39-23-2	100,0000	99,9900	BFGS 48	Entropy	Logistic	Softmax
30	MLP 30-29-2	100,0000	99,9900	BFGS 43	Entropy	Tanh	Softmax
20	MLP 20-24-2	100,0000	99,9900	BFGS 44	Entropy	Tanh	Softmax
10	MLP 10-22-2	99,9659	99,9402	BFGS 21	Entropy	Logistic	Softmax
5	MLP 5-10-2	94,7326	95,1295	BFGS 99	SOS	Logistic	Logistic
3	MLP 3-20-2	93,2471	93,8048	BFGS 2	SOS	Tanh	Logistic
2	MLP 2-6-2	93,2471	93,8048	BFGS 2	Entropy	Tanh	Softmax
1	MLP 1-5-2	93,2471	93,8048	BFGS 4	SOS	Logistic	Tanh

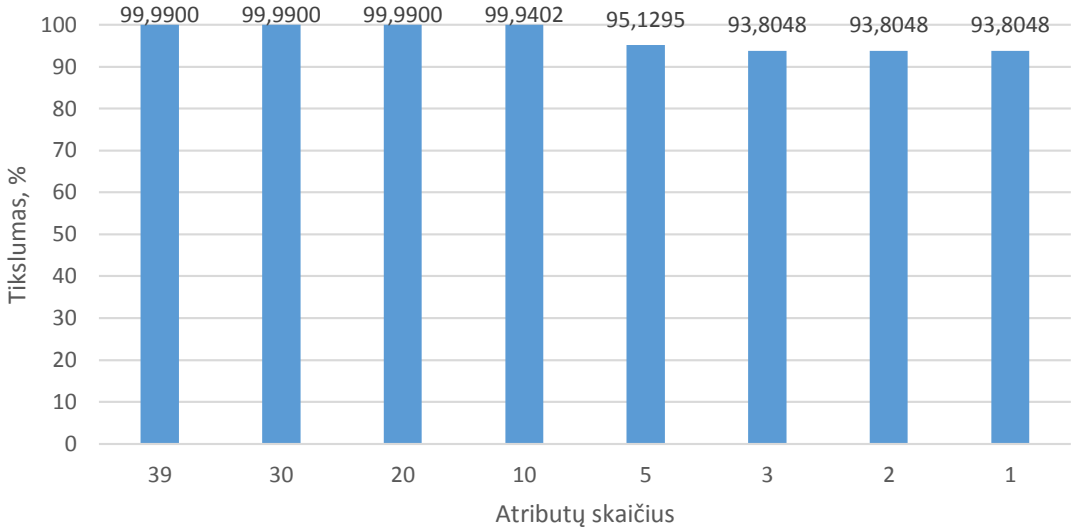
DSP atpažinimo lygis, teikiant 39 parametrus, suteikia 99,99 % tikslumo. Beveik toks pat tikslumas pasiekiamas sumažinus parametrų skaičių iki 10. Toliau mažinant įėjimų skaičių, tikslumas išlieka virš 93 %. Pradedant nuo 3 įėjimo parametrų naudojimo, DSP tikslumas lygus 93,8048 % ir nesikeičia, mažinant parametrų skaičių iki 1.

3.13 lentelė. SBF tikslumas, 2 klasės

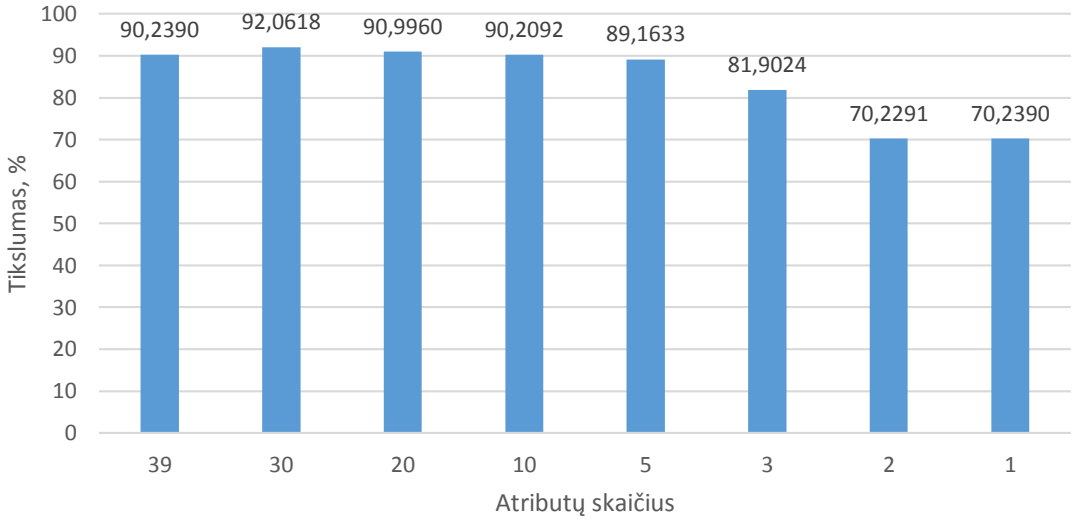
Atributų skaičius	NT Pavadinimas	Mokymo tikslumas, %	Testavimo tikslumas, %	Mokymo algoritmas	Klaidos funkcija	Paslėptų neuronų aktyvavimo funkcija	Išėjimo neuronų aktyvavimo funkcija
39	RBF 39-41-2	89,4182	90,2390	RBFT	Entropy	Gaussian	Softmax
30	RBF 30-41-2	91,2964	92,0618	RBFT	Entropy	Gaussian	Softmax
20	RBF 20-46-2	91,3006	90,9960	RBFT	Entropy	Gaussian	Softmax
10	RBF 10-49-2	89,3840	90,2092	RBFT	Entropy	Gaussian	Softmax
5	RBF 5-49-2	88,5517	89,1633	RBFT	Entropy	Gaussian	Softmax
3	RBF 3-60-2	81,8159	81,9024	RBFT	Entropy	Gaussian	Softmax
2	RBF 2-43-2	70,3803	70,2291	RBFT	Entropy	Gaussian	Softmax
1	RBF 1-56-2	70,4145	70,2390	RBFT	Entropy	Gaussian	Softmax

Apjungus visas *KDD Cup 1999* duomenų rinkinio *DDoS* atakų klases į vieną, t. y. naudojant 2 klasifikavimo kategorijas, šis NT modelis suteikia geresnių rezultatų, lyginant su 7 klasifikavimo kategorijų testavimo rezultatais. Kai naudojami 39 įėjimo parametrai, SBF tikslumas sudaro 90,239 %. Sumažinus įėjimo parametrų skaičių iki 10, tikslumas sumažėja neryškiai, ir sudaro 90,2092 %. Esant 3 įėjimo parametrams, SBF tikslumas sudaro 81,9024 %, o esant 1 įėjimo parametrui, tikslumas sudaro 70,239 %.

3.8 pav. ir 3.9 pav. pavaizduoti DSP ir SBF tikslumo rezultatai, naudojant 2 klases.

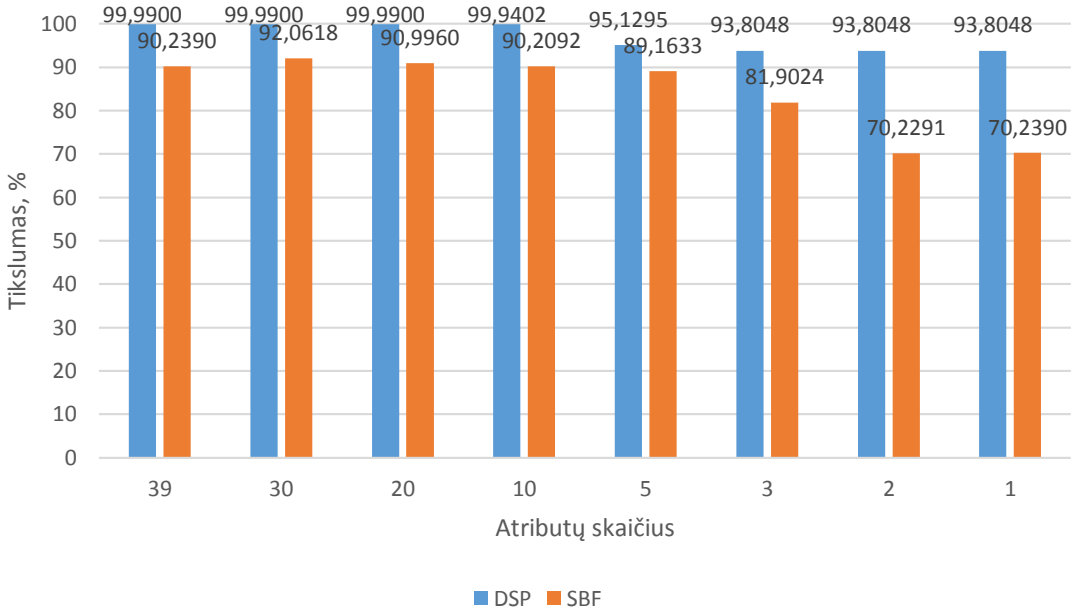


3.8 pav. DSP tikslumas, 2 klasės



3.9 pav. SBF tikslumas, 2 klasės

3.10 pav. pateikiamas DSP ir SBF tikslumo palyginimas, naudojant 2 klases.



3.10 pav. DSP ir SBF tikslumo palyginimas, 2 klasės

Žemiau pateikiami klaidingai teigiami DSP (3.14 lentelė) ir SBF (3.15 lentelė) rodikliai, naudojant 2 klasifikavimo kategorijas.

3.14 lentelė. DSP. Klaidingai teigiami DDoS atakų aptikimo rezultatai, 2 klasės

Atributų skaičius	**NT pavadinimas**	**normal**	**attack**
39	MLP 39-23-2	10,8859	0,0000
30	MLP 30-29-2	0,0000	0,0000
20	MLP 20-24-2	6,8374	9,4974
10	MLP 10-22-2	0,0429	0,0304
5	MLP 5-10-2	1,6736	6,7961
3	MLP 3-20-2	0,0000	9,6252
2	MLP 2-6-2	0,0000	9,6252
1	MLP 1-5-2	0,0000	9,6252

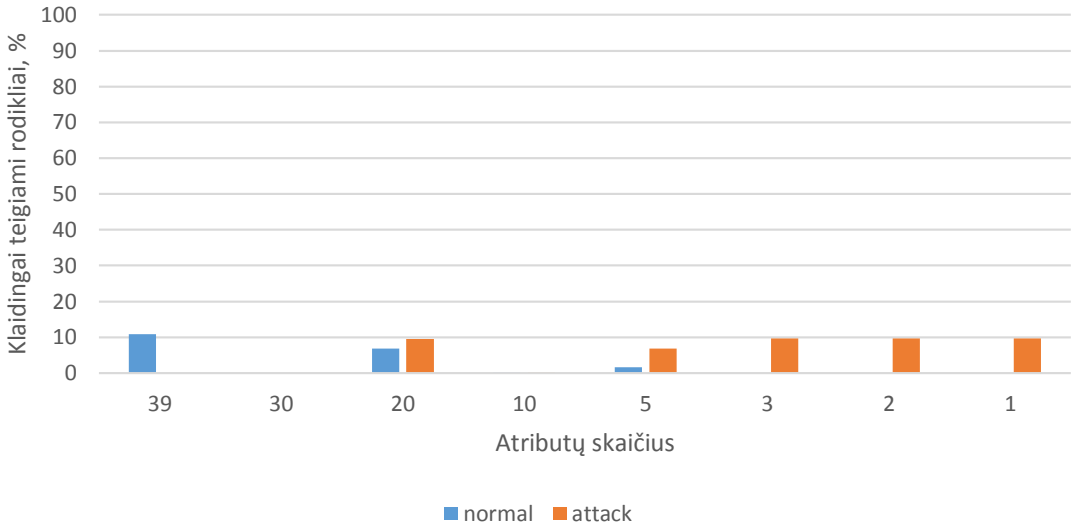
Apjungus visus *KDD Cup 1999 DDoS* atakų tipus į vieną bendrą, klaidingai teigiami rodikliai sumažėjo. Esant 39 įėjimo parametrams, DSP negeneruoja nė vieno klaidingai teigiamo pranešimo aptinkant anomalinį tinklo srautą. Aptinkant teisėtą tinklo srautą, klaidingai teigiamų pranešimų rodiklis sudaro 10,8859 %. Esant 30 įėjimo parametrams, teisėto ir anomalinio tinklo srauto aptikimas negeneruoja nė vieno klaidingai teigiamo rezultato. Naudojant 1 įėjimo parametrą, teisėto tinklo srauto klaidingai teigiami atpažinimo rezultatai lygus 0 %, o anomalinio tinklo srauto rezultatai lygus 9,6252 %.

3.15 lentelė. SBF. Klaidingai teigiami DDoS atakų aptikimo rezultatai, 2 klasės

Atributų skaičius	**NT pavadinimas**	**normal**	**attack**
39	RBF 39-41-2	19,5861	7,4260
30	RBF 30-41-2	6,8374	9,4974
20	RBF 20-46-2	13,5746	6,6257
10	RBF 10-49-2	17,1077	7,8547
5	RBF 5-49-2	7,2522	13,2331
3	RBF 3-60-2	54,4128	2,7744
2	RBF 2-43-2	98,9129	0,1460
1	RBF 1-56-2	98,9987	0,0608

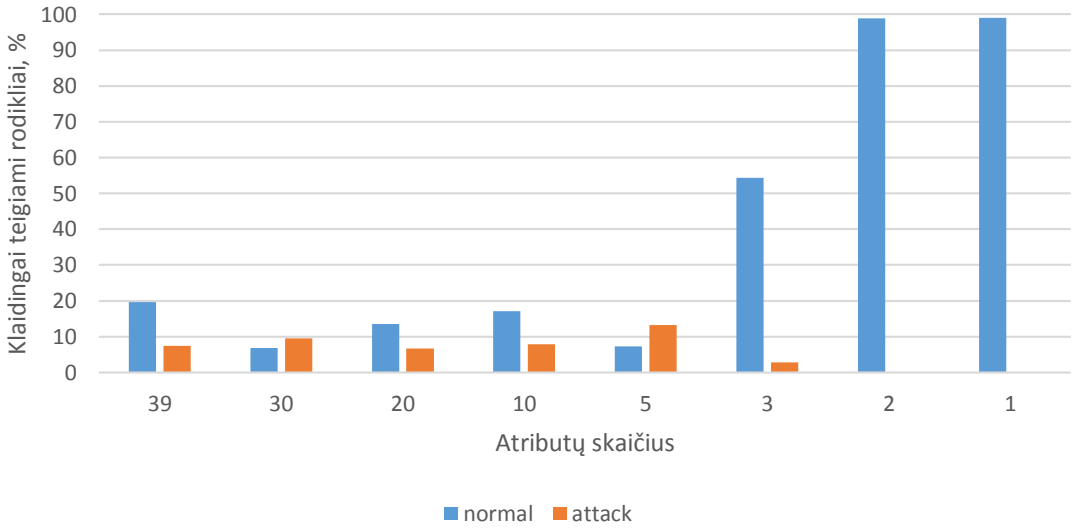
SBF tinklo klaidingai teigiami rodikliai, naudojant 2 klasifikavimo kategorijas, yra mažesni, palyginus su 7 klasifikavimo kategorijų rodikliais. Esant 39 įėjimo parametrams, SBF tinklo klaidingai teigiamų rezultatų skaičius sudaro 19,5861 % aptinkant teisėtą tinklo srautą, ir 7,426 %, aptinkant anomalinį tinklo srautą. Kaip ir DSP atveju, naudojant 30 įėjimo parametrų, SBF tinklo rodikliai yra geresni, lyginant su 39 parametrais – teisėto tinklo srauto rodikliai lygūs 6,8347 %, o anomalinio tinklo srauto rodikliai lygūs 9,4974 %. Sumažinus įėjimo parametrų skaičių iki 3, tinklo anomalijų klaidingai teigiami rodikliai sumažėja iki 2,7744 %, bet ženkliai padidėja teisėto tinklo srauto aptikimo klaidingai teigiamų rodikliai – iki 54,4128 %. Tokia pati tendencija išlieka mažinant parametrų skaičių iki 1 – anomalinio tinklo srauto klaidingai teigiami rodikliai mažėja mažinant parametrų skaičių, o teisėto tinklo srauto klaidingai teigiami rodikliai atvirkščiai, didėja. Esant 1 įėjimo parametrui, klaidingai teigiamų teisėto tinklo srauto aptikimo rodikliai sudaro 98,9987 %, t. y. beveik visos tinklo sesijos, priklausančios teisėtam tinklo srautui, traktuojamos kaip anomalinės. Visiškai priešinga situacija yra su anomalinių tinklo srautu, kurio klaidingai teigiami rodikliai sudaro vos 0,0608 %. Išsamūs tinklo srauto klasifikavimo rezultatai pateikiami B priede.

3.11 pav. pateikiami DSP klaidingai teigiami *DDoS* atakų aptikimo rezultatai, naudojant 2 klases.



3.11 pav. DSP. Klaidingai teigiami DDoS atakų aptikimo rezultatai, 2 klasės

3.12 pav. pateikiami SBF klaidingai teigiami *DDoS* atakų aptikimo rezultatai, naudojant 2 klases.



3.12 pav. SBF. Klaidingai teigiami DDoS atakų aptikimo rezultatai, 2 klasės

Kalbant apie NT mokymą, reikia paminėti tokį svarbų aspektą, kaip mokymo laikas. Akivaizdu, kad kuo daugiau parametrų pateikiama NT, tuo geresnis atpažinimo tikslumas, tačiau tai reikalauja didesnių laiko sąnaudų. Todėl būtinas balansas tarp rezultato tikslumo ir laiko, reikalingo mokymui. NT mokymo laiko matavimai buvo atliekami naudojant žemiau pateiktą konfigūraciją:

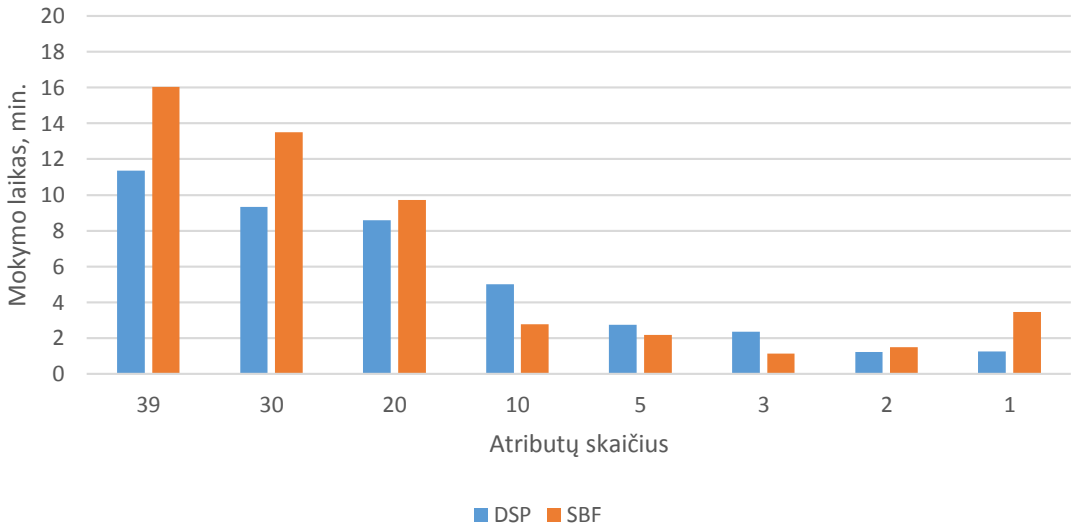
- CPU: Intel Core i5-4200U, 2300 MHz.
- RAM: DDR3 2x4 GB, 1600 MHz.
- OS: Windows 8.1 Enterprise x64.

Žemiau pateikiama NT mokymo laiko priklausomybė nuo įėjimo parametrų skaičiaus (3.16 lentelė).

3.16 lentelė. DSP ir SBF mokymo laiko priklausomybė nuo įėjimo parametrų skaičiaus

Atributų skaičius	**DSP laikas, min.**	**SBF laikas, min.**
39	11,3667	16,0333
30	9,3333	13,5000
20	8,6000	9,7167
10	5,0000	2,7833
5	2,7333	2,1833
3	2,3667	1,1333
2	1,2167	1,5000
1	1,2500	3,4667

3.13 pav. pavaizduota DSP ir SBF tinklų mokymo laiko priklausomybė nuo įėjimo parametrų skaičiaus.



3.13 pav. DSP ir SBF mokymo laiko priklausomybė nuo įėjimo parametrų skaičiaus

Naudojant 39 įėjimo parametrus, DSP mokymo laikas sudaro 11,3667 min. Toliau mažinant įėjimo parametrų skaičių, mažėja ir laikas, skirtas DSP mokymui. Esant 10 parametrams, DSP mokymo laikas sumažėja apytiksliai dvigubai, ir lygus 5 min., o esant 2 parametrams, DSP

mokymo laikas lygus 1,2167 min. Sumažinus įėjimo parametrų skaičių iki 1, mokymo laikas padidėja palyginus su 2 įėjimo parametrais, ir sudaro 1,25 min.

SBF tinklų mokymo laikas esant 39 įėjimo parametrams sudaro 16,0333 min. Naudojant 10 įėjimo parametrų, RBF mokymo laikas sudaro 2,7833 min. Kai įėjimo parametrų skaičius sumažėja iki 3, SBF mokymo laikas sudaro 1,1333 min., tačiau toliau mažinant įėjimo parametrų skaičių, SBF mokymo laikas pradeda didėti. Esant 2 įėjimo parametrams, SBF mokymo laikas sudaro 1,5 min., o esant 1 įėjimo parametrui, SBF mokymo laikas sudaro 3,4667 min.

3.4. Rezultatų aptarimas

Remiantis gautais rezultatais, akivaizdu, kad DSP efektyvumas aptinkant *DDoS* atakas, naudojant 39 *KDD Cup 1999* duomenų rinkinio požymių, sudaro 99,97 %. Naudojant net tik 2 šio duomenų rinkinio požymius, tikslumas išlieka didesnis nei 90 %, ir tik sumažinus požymių skaičių iki 1, tikslumas siekė 63,1574 %.

SBF tinklai, palyginus su DSP, suteikia mažesnį efektyvumo lygį – naudojant 39 atributus, šių tinklų efektyvumas sudaro 88,3466 %, o sumažinus atributų skaičių iki 5, jų efektyvumas sumažėja iki 54,5319 %. Toliau mažinant atributų skaičių, SBF efektyvumas beveik nesikeičia, o naudojant 1 įėjimo parametrą, efektyvumo rezultatai buvo geresni, nei naudojant 5 įėjimo parametrus – 65,9363 % prieš 54,5319 %.

DSP klaidingai teigiami rodikliai, naudojant 39 įėjimo parametrus, sudaro 0 % aptinkant visas *KDD Cup 1999* duomenų rinkinio klases. Panašūs rodikliai išlieka sumažinus įėjimo parametrų skaičių iki 10.

SBF tinklo klaidingai teigiami rodikliai, analizuojant visas *KDD Cup 1999* duomenų rinkinio klases, viršija DSP rodiklius. Esant 39 įėjimo parametrams, šis tinklas nesugeba teisingai aptikti nė vienos *land* ir *pod* atakos. Įėjimo parametrų rinkiniui sumažėjus iki 30, SBF tinklas jau nesugeba aptikti nė vienos *land*, *pod* ir *teardrop* atakos. *Back* tipo atakų aptikimas šio NT irgi esant 39 įėjimo parametrams sudaro 96,2705 %, o sumažėjus parametrų rinkiniui iki 5, SBF nesugeba aptikti nė vienos *back* tipo atakos. Teisėto tinklo srauto klaidingai teigiami rezultatai esant 39 įėjimo parametrams sudaro 15,992 %. Sumažinus įėjimų skaičių iki 10, teisėto tinklo srauto klaidingai teigiami rodikliai sudaro 38,8643 %. Vienintelė *DDoS* atakos klasė, kurią SBF tinklas atpažįsta generuojant ne daugiau 3,7117 % esant bet kuriam įėjimų parametrų skaičiui, tai *neptune* tipo ataka. Savaime suprantama, kad SBF tinklo modelis netinka *DDoS* atakų aptikimui naudojant net 39 *KDD Cup 1999* duomenų rinkinio atributus dėl labai aukštų klaidingai teigiamų rodiklių.

Atliekant tyrimus apjungus *KDD Cup 1999* duomenų rinkinio *DDoS* atakų klases į vieną bendrą klasę, DSP ir SBF aptikimo rodikliai padidėja. Akivaizdu, kad šiuo atveju NT reikia klasifikuoti pateikiamą į jo įėjimus objektą remiantis ne 7 klasėmis, o tik 2.

DSP efektyvumo rodikliai esant 39 įėjimo parametrams sudaro 99,99 %. Lygiai toks pat tikslumas pasiekiamas naudojant 20 įėjimo parametrus. DSP efektyvumas išlieka efektyvus esant net 1 įėjimo parametrui ir sudaro 93,8048 %.

SBF efektyvumas, aptinkant *DDoS* atakas esant 2 klasifikavimo kriterijams, yra žemesnis, palyginus su DSP, kaip ir 7 klasifikavimo kriterijų atveju. Pateikiant 39 įėjimo parametrus, šio tinklo efektyvumas sudaro 90,239 % ir išlieka virš 90 % ribose sumažinus įėjimo parametrų skaičių iki 10. Esant 1 įėjimo parametrui, SBF efektyvumas sudaro 70,239 %.

DSP teisėto ir anomalinio tinklo srauto klaidingai teigiamų rezultatų rodikliai esant 39 įėjimo parametrams sudaro atitinkamai 10,8859 % ir 0 %. Sumažinus įėjimo parametrų skaičių iki 30, DSP negeneruoja nė vieno klaidingai teigiamo pranešimo aptinkant kaip teisėtą taip ir anomalinį tinklo srautą. Panašūs rezultatai gaunami ir su mažesniu parametru skaičiumi. Naudojant 1 įėjimo parametrą, šis tinklas negeneruoja nė vieno klaidingai teigiamo pranešimo, analizuojant teisėtą tinklo srautą, anomalinio tinklo srauto klaidingai teigiamų rezultatų rodiklis sudaro tik 9,6252 %.

SBF klaidingai teigiamų aptikimų rezultatai naudojant 39 įėjimo parametrus sudaro 19,5861 % aptinkant teisėtą tinklo srautą ir 7,426 % aptinkant anomalinį tinklo srautą. Sumažinus atributų skaičių iki 5, klaidingai teigiami teisėto tinklo srauto rodikliai sudaro 7,2522 %, o anomalinio tinklo srauto rodikliai sudaro 13,2331 %. Sumažinus atributų skaičių iki3, klaidingai teigiami teisėto tinklo srauto rezultatai padidėja iki 54,4128 %, tačiau anomalinio tinklo klaidingai teigiami rezultatai sumažėja iki 2,7744 %. Esant 1 atributui, teisėto tinklo srauto rezultatai lygūs 98,9987 %, o anomalinio tinklo srauto rezultatai lygūs 0,0608 %. Tai reiškia, kad mažėjant atributų skaičiui, didėja anomalinio tinklo srauto aptikimo rezultatai.

Akivaizdu, kad *KDD Cup 1999* duomenų rinkinyje dalis atributų turi mažesnę įtaką bendram atpažinimo rezultatui, todėl ne visais atvejais apsimoka naudoti didesnį atributų skaičių. Be to, kuo daugiau atributų paduodama NT (kuo ilgesnis įėjimo vektorius), tuo daugiau laiko užima NT apmokymo procesas. Todėl būtina pasirinkti optimalią NT konfigūraciją.

Prieš pateikdami galutinius darbo rezultatus, apibrėžkime, kas yra optimali NT konfigūracija. Šiame darbe optimali konfigūracija – tai tokia konfigūracija, kuri suteikia bent 99 % atpažinimo tikslumą su mažiausiai naudojamų įėjimo neuronų skaičiumi.

Remiantis gautais tyrimų rezultatais, galima teigti, kad optimali NT konfigūracija, skirta *DDoS* atakų aptikimui, naudojant 7 tinklo srauto klasifikavimo grupes, yra DSP 10-23-7. Jeigu nėra poreikio klasifikuoti *DDoS* ataką, o reikia nustatyti tik šios atakos faktą, užtenka naudoti DSP 10-22-2 konfigūraciją. Žemiau pateikiamos optimalios NT konfigūracijos, naudojant 7 ir 2 tinklo

srauto klasifikavimo grupes (3.17 lentelė). NT su šiomis konfigūracijomis suteikia virš 99 % tikslumo rezultatus.

3.17 lentelė. Optimalios NT konfigūracijos, siekiant aptikti DDoS ataką

Analizuojamo tinklo srauto klasių skaičius	**NT modelis**	**Įėjimo sluoksnio neuronų skaičius**	**Vidinio sluoksnio neuronų skaičius**
7	DSP	10	23
2	DSP	10	22

Atributų įtaka galutiniam *DDoS* atakų atpažinimo rezultatui pateikta C priede.

4. Siūlomas DDoS atakų aptikimo metodo prototipas

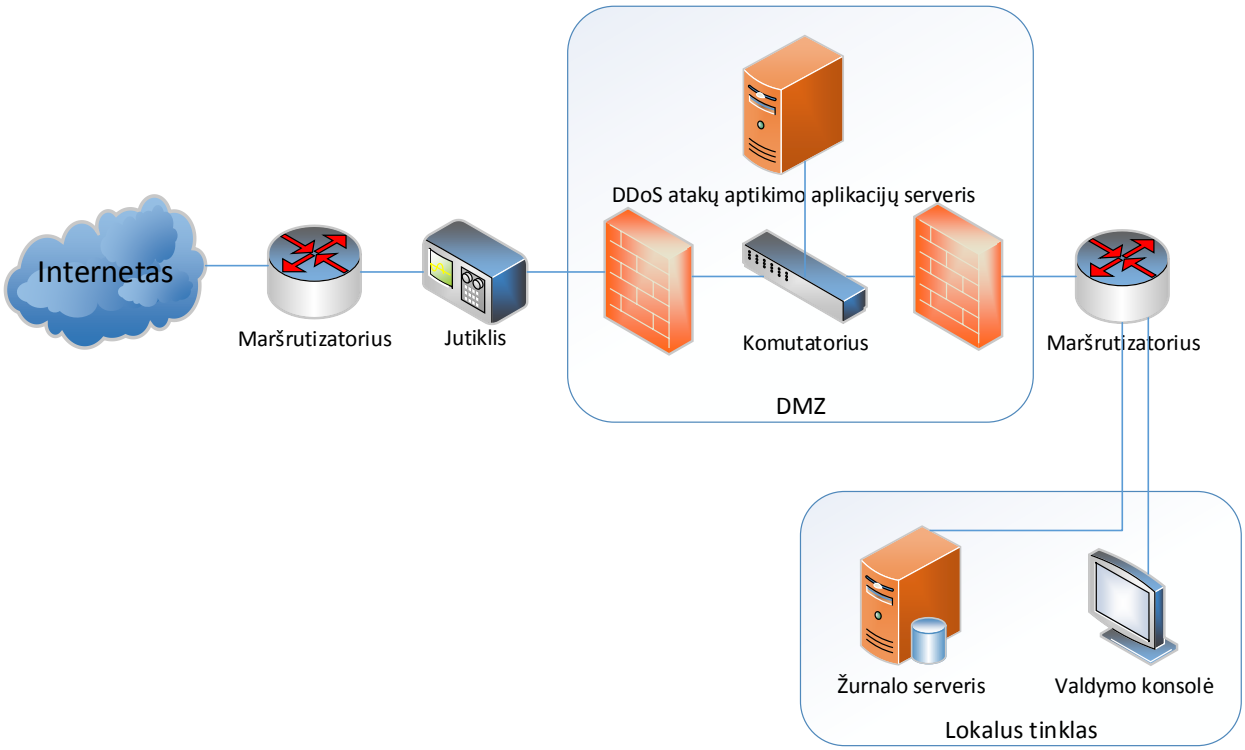
4.1. Siūlomo prototipo architektūra

Siūlomas šiame darbe *DDoS* atakų aptikimo metodo prototipas sudarytas iš keturių pagrindinių dalių (4.1 pav.):

1. Tinklo srauto analizės jutiklių, stebinančių tinklo paketus.
2. Neuroninio tinklo, kuris remdamasis jam paduodama iš jutiklių informacija, galės nuspręsti, ar paketas priklauso teisėtam tinklo srautui, ar šis paketas yra *DDoS* atakos dalis.
3. Statistikos DB, kurioje saugoma išanalizuotų tinklo paketų informacija su nustatytais požymiais (teisėtas/neteisėtas).
4. Valdymo konsolės, kurios pagalba suteikiama galimybė konfigūruoti analizuojamų tinklo paketų parametrus, vykdyti NT apmokymą ir testavimą, stebėti tinklo srautą realiuoju laiku, žiūrėti jau išanalizuotų paketų statistiką.

Prototipas realizuotas pagal 3 lygmenų (angl. *tier*) architektūrą:

- DBVS – DB serveris, kuriame saugoma tinklo srauto statistika.
- Aplikacijų serveris – pagrindinis elementas, kurio branduolį sudaro NT.
- Plonas klientas – interneto naršyklė, kurios pagalba galima valdyti ir stebėti visos sistemos veiklos procesą.



4.1 pav. DDoS atakų aptikimo sistemos architektūra

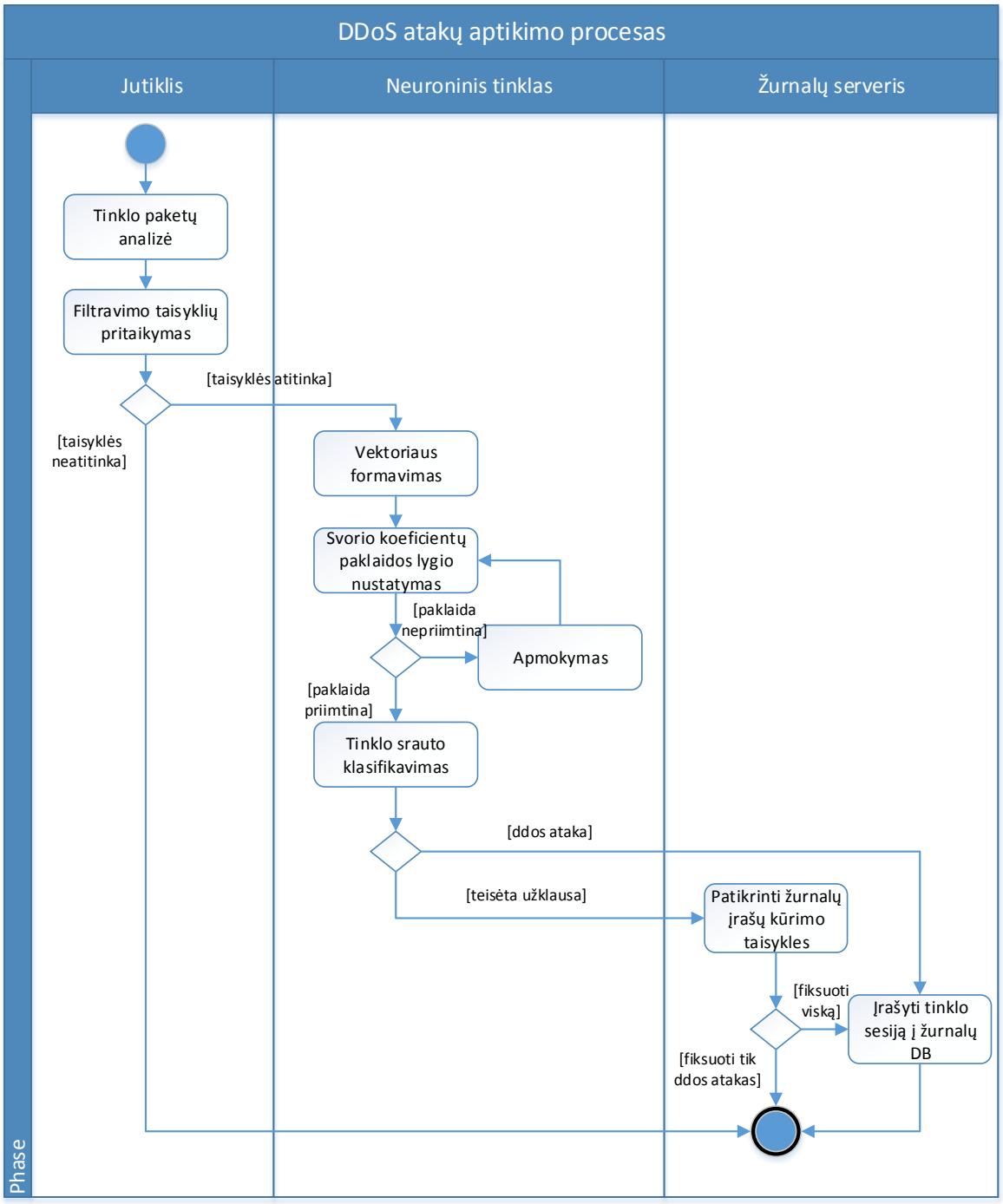
Jutiklio vaidmenį atlieka programa *tcpdump* [**40**], [**41**], kuri analizuoja tinklo paketus ir išanalizuotus duomenis perduoda aplikacijų serveriui.

Tcpdump – tai tinklo paketų analizatorius. Viena iš šios programos savybių yra ta, kad *tcpdump* gali perjungti tinklo adapterį į mišrų režimą (angl. *promiscuous mode*), kai tinklo adapteris perima visus, cirkuliuojančius tinkle paketus, o ne tik tuos, kurie buvo adresuoti jam vienam. Ši savybė labai naudinga, norint panaudoti *tcpdump* viso įeinančio į tinklą paketų srauto stabėjimui. Kuriant *DDoS* atakų aptikimo prototipą, *tcpdump* programa yra įdiegta pačiame maršrutizatoriuje su *DD-WRT* mikroprograma (angl. *firmware*), todėl 4.1 pav. jutiklis pavaizduotas kaip loginis tinklo elementas, o ne kaip konkretus fizinis įrenginys. *Tcpdump* turi galimybę stebėti tik tam tikro protokolo paketus, tam naudojami specialūs filtrai, tai labai naudinga tais atvejais, kai reikia analizuoti tik konkretaus protokolo paketus, kas sumažina analizės laiką ir mažiau įtakoja bendram tinklo srauto greičiui.

Jutiklis yra už lokalaus tinklo ir *DMZ* ribų, išoriniame maršrutizatoriuje, sujungiančiame bendrą įmonės tinklą su interneto tiekėju. Toks sprendimas priimtas dėl to, kad *DoS/DDoS* atakos atveju dalis paketų gali būti atmesta pagal išorinės ugniasienės taisykles, o tai sumažins bendrą paketų aibę, pagal kurią yra nustatoma *DoS/DDoS* atakos tikimybė.

Iš jutiklio duomenys perduodami aplikacijų serveriui. Aplikacijų serveryje įdiegta *DDoS* atakų aptikimo sistema, naudojanti aptikimo mechanizmui NT. Remiantis sukaupta informacija, gauta NT apmokymo metu, atliekama gautų iš jutiklio tinklo paketų analizė. NT išėjimo duomenys

(atsakymas) siunčiami žurnalo serveriui ir įrašomi į DB. Tinklo srauto analizės procesas stebimas per valdymo konsolę. Valdymo konsolės pagalba taip pat atliekamas jutiklių ir NT ir žurnalų įrašų parametrų nustatymai. Aplikacijų serveris patalpintas *DMZ* už išorinės ugniasienės, nes tai apsaugo jį nuo galimų atakų iš interneto. Siekiant apsaugoti žurnalų serverį nuo įvairių išorinių atakų, jis patalpintas lokaliame tinkle už ugniasienės. 4.2 pav. pavaizduotas *DDoS* atakų aptikimo procesas sukurto prototipo pagalba.

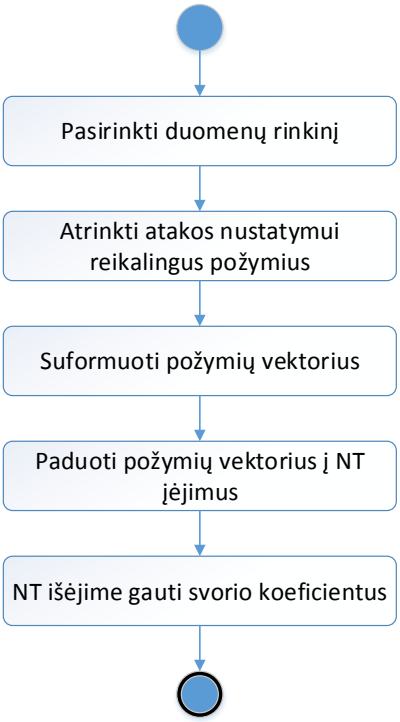


4.2 pav. DDoS atakų aptikimo proceso UML veiklos diagrama

Iš pradžių atliekama tinklo paketų analizė. Priklausomai nuo naudojamų tinklo analizatoriaus filtravimo taisyklių, atrenkami reikalingi tolimesniam tinklo srauto anomalijų aptikimui paketai. Jeigu paketas atitinka nustatytoms filtravimo taisykles, tada jis pateikiamas į NT modulį. NT modulyje tinklo paketai apjungiami į sesijas, kur kiekvienai sesijai sukuriamas įėjimo vektorius. Šis vektorius pateikiamas į NT įėjimus. Prieš pradedant naudoti NT *DDoS* atakų aptikimui, jį reikia išmokyti korektiškai klasifikuoti pateiktą informaciją. Tam reikia nustatyti svorio koeficientų paklaidos lygį. NT mokymas – tai iteratyvus procesas, kurio metu tikrinamas svorio koeficientų paklaidos lygis. Jeigu šis rodiklis yra priimtinas (priimtinumo lygis nustatomas iš anksto), reikia manyti, kad NT yra apmokytas, priešingu atveju, pereinama prie kitos iteracijos tol, kol bus pasiektas priimtinas paklaidos lygis. Kai NT yra apmokytas, jį galima taikyti tinklo srauto klasifikavimui. Remiantis sukaupta informacija, NT klasifikuoja pateiktą informaciją, kuri aprašo tinklo sesiją. Priklausomai nuo žurnalų įrašų kūrimo taisyklių, analizuojamos tinklo sesijos informacija gali būti įrašyta į žurnalų DB.

4.2. DDoS atakų aptikimo procesas

Siekiant mokyti NT aptikti *DDoS* tipo atakas reikia atlikti žemiau išvardintus žingsnius (4.3 pav.):

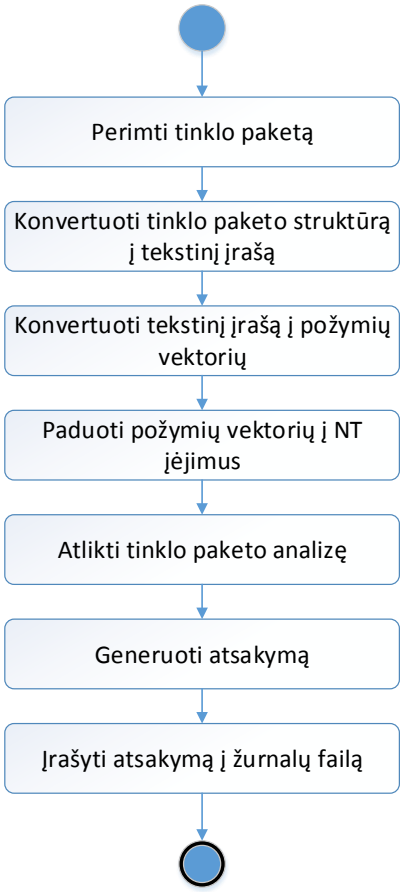


4.3 pav. NT mokymo procesas

1. Pasirinkti duomenų rinkinį, kuriame aprašyti teisėto ir anomalinio tinklo srauto pavyzdžiai.

2. Atrinkti iš pasirinkto duomenų rinkinio atakos nustatymui reikalingus požymius.
3. Konvertuoti atrinktus požymius į skaitinius ekvivalentus ir suformuoti vektorius.
4. Paduoti vektorius į NT įėjimus.
5. NT išėjime gauti svorio koeficientus.

NT mokymo procese tikrinami išėjimo vektorių rezultatai (svorio koeficientai). Jeigu jų paklaida sąlyginai maža, NT įvardinamas kaip apmokytas, priešingu atveju atliekamas svorio koeficientų koregavimas ir apmokymo procesas yra kartojamas [**23**], [**33**]. Kai NT yra apmokytas, jį galima naudoti *DDoS* atakų aptikimui. *DDoS* atakų aptikimo procesas sudarytas iš veiksmų, parodytų 4.4 pav.:



4.4 pav. DDoS atakų aptikimo procesas

1. Tinklo srauto analizatorius (angl. *sniffer*) perima paketus ir siunčia tekstinių duomenų analizatoriui (angl. *parser*).
2. Tekstinių duomenų analizatorius iš gauto tinklo paketo struktūros formuoja tam tikro formato tekstinį įrašą.
3. Tekstinis įrašas konvertuojamas į vektorių.
4. Vektoriai paduodami į NT įėjimus.

5. NT remiantis sukauptomis jame žiniomis atlieka atitinkamo tinklo paketo analizę.
6. NT išėjime generuojamas atsakymas, nusakantis išanalizuotų paketų požymį (teisėtas, *DDoS* ataka).
7. Atsakymas įrašomas į žurnalų failą.

Tinklo paketai, perimti tinklo srauto analizatoriaus pagalba, konvertuojami į tam tikrą struktūrą, iš kurios gali būti suformuotas požymių vektorius. Šios struktūros sudarymas – tai duomenų gavybos uždavinys. Nuo to, kaip yra sudaryta ši struktūrą, priklauso NT *DDoS* atakų atpažinimo tikslumas.

Išvados

1. Atlikus skirtingų atvirojo kodo įsilaužimų aptikimo sistemų analizę, buvo nustatyta, kad šios sistemos netinka naujų, dar nežinomų, tinklo anomalijų aptikimui. Visos išnagrinėtos sistemos atakų aptikimui naudoja paršais grindžiamą metodą. Tai reiškia, kad nė viena iš pateiktų įsilaužimų aptikimo sistemų nesugeba aptikti tų tinklo anomalijų tipų, kurie nėra aprašyti parašų DB.
2. Išnagrinėjus įsilaužimų aptikimo sistemų ir parašais grindžiamų metodų trūkumus, buvo priimtas sprendimas naudoti anomalijų aptikimo metodą *DDoS* atakoms nustatyti. Šiam metodui įgyvendinti buvo panaudoti DSP ir SBF tinklai. NT suteikia galimybę aptikti dar nežinomas atakas ir atsisakyti signatūrų DB.
3. Atliktas eksperimentas, kurio metu nustatytas nagrinėjamų NT efektyvumas sprendžiant *DDoS* atakų aptikimo uždavinį, pateikti šio eksperimento rezultatai. Eksperimento metu buvo nustatyta, kad sėkmingam *DDoS* atakų aptikimui užtenka naudoti tik dalį *KDD Cup 1999* duomenų rinkinio požymių. Palyginus DSP ir SBF tinklų rezultatus, akivaizdu, kad tik DSP suteikia priimtiną *DDoS* atakų atpažinimo lygį. Tyrimų rezultatai parodo, kad NT gali konkuruoti su tradiciniais tinklo anomalijų aptikimo sprendimais, tokiais kaip įsilaužimų aptikimo sistemos.

Literatūros sąrašas

[1] SCHNEIER, B. Schneier on Security. [interaktyvus]. [žiūrėta 2015-04-26]. Prieiga per internetą: <https://www.schneier.com/crypto-gram-0501.html>

[2] ACM Committee on Computers and Public Policy. The Risks Digest. [interaktyvus]. [žiūrėta 2015-04-26]. Prieiga per internetą: <http://catless.ncl.ac.uk/Risks/18.45.html>

[3] SILVA, S.; SILVA, R.; PINTO, R.; SALLES, R. Botnets: A survey, *Computer Networks*, 2013, 57, p. 378–403.

[4] KAMLUK, V. The botnet business. [interaktyvus]. [žiūrėta 2015-04-26]. <Prieiga per internetą: http://securelist.com/analysis/36209/the-botnet-business/>

[5] ТИМОФЕЕВ, А.; БРАНИЦКИЙ, А. Исследование и моделирование нейросетевого метода обнаружения и классификации сетевых атак, *Information Technologies & Knowledge*, 2012, 6, p. 257–265.

[6] XUEYING, J.; KEAN, L.; JIEGOU, Y.; WENHUI, C. Application of Improved SOM Neural Network in Anomaly Detection, *Physics Procedia*, 2012, 33, p. 1093–1099.

[7] DERMATAS, E.; GAVRILIS, D., Real-time detection of distributed denial-of-service attacks using RBF networks and statistical features, *Computer Networks and ISDN Systems*, 2005, 48(2), p. 235–245.

[8] ЕМЕЛЬЯНОВА, Ю.; ТАЛАЛАЕВ, А.; ТИЩЕНКО, И.; ФРАЛЕНКО, В. Нейросетевая технология обнаружения сетевых атак на информационные ресурсы, *Программные системы: теория и приложения*, 2011, 3(7), p. 3–15.

[9] ALFANTOOKH, A. A. DoS Attacks Intelligent Detection using Neural Networks, *Journal of King Saud University - Computer and Information Sciences*, 2006, 18, p. 31–51.

[10] LABOVITZ, C., The Internet Goes to War. [interaktyvus]. [žiūrėta 2015-04-26]. Prieiga per internetą: <http://www.arbornetworks.com/asert/2010/12/the-internet-goes-to-war/>

[11] Kaspersky Lab, Global Corporate IT Security Risks. [interaktyvus]. [žiūrėta 2015-04-26]. Prieiga per internetą: <http://media.kaspersky.com/en/business-security/Kaspersky_Global_IT_Security_Risks_Survey_report_Eng_final.pdf>

[12] Arbor Networks, Inc., Arbor Networks Releases Q3 Global DDoS Attack Trends Data. [interaktyvus]. [žiūrėta 2015-04-26]. Prieiga per internetą: <http://www.arbornetworks.com/news-and-events/press-releases/recent-press-releases/5026-arbor-networks-releases-q3-global-ddos-attack-trends-data>

[13] PATRIKAKIS, C.; MASIKOS, M.; ZOURARAKI, O. Distributed Denial of Service Attacks, *The Internet Protocol Journal*, 2004, 7(4), p. 13–35.

[14] RAMANAUSKAITĖ, S; ČENYS, A. Taxonomy of DoS Attacks and Their Countermeasures, *Central European Journal of Computer Science*, 2011, 1(3), p. 355–366.

[15] WESLEY, M. Defenses Against TCP SYN Flooding Attacks, *The Internet Protocol Journal*, 2006, 9(4), p. 2–31.

[16] HASHMI, J; SAXENA, M; SAINI, R. Classification of DDoS Attacks and their Defense Techniques using Intrusion Prevention System, *International Journal of Computer Science & Communication Networks*, 2012, 2(5), p. 607–614.

[17] ПАРКЕР, Т.; СИЯН, К. *TCP/IP*. Санкт-Петербург: ЗАО Издательский дом "Питер", 2004. ISBN 5-8046-0041-9

[18] XIAO, B.; CHEN, W.; HE, Y. An autonomous defense against SYN flooding attacks: Detect and throttle attacks at the victim side independently, *J. Parallel Distrib. Comput.*, 2007, 68, p. 456–470.

[19] KOSTADINOV, D. Layer Seven DDoS Attacks. [interaktyvus]. [žiūrėta 2015-04-26]. Prieiga per internetą: <http://resources.infosecinstitute.com/layer-seven-ddos-attacks/>

[20] CERT. NTP Amplification Attacks Using CVE-2013-5211. [interaktyvus]. [žiūrėta 2015-04-26]. Prieiga per internetą: <https://www.us-cert.gov/ncas/alerts/TA14-013A>

[21] ЩЕРБА М. Обнаружение низкоактивных распределенных атак типа «отказ в обслуживании» в компьютерных сетях, 2012.

[22] КЛИМОВ, С.; СЫЧЕВ, М.; АСТРАХОВ, А. Противодействие компьютерным атакам. Методические основы, 2013.

[23] PATCHA, A.; PARK, J.M. An overview of anomaly detection techniques: Existing solutions and latest technological trends, *Computer Networks*, 2007, 51, p. 3448–3470.

[24] CATANIA, C.A.; GARINO, C.G. Automatic network intrusion detection: Current techniques and open issues, *Computers and Electrical Engineering*, 2012, 38, p. 1062–1072.

[25] GARCIA-TEODORO, P.; DIAZ-VERDEJO, J.; MACIA-FERNANDEZ, G.; VAZQUEZ, E. Anomaly-based network intrusion detection: Techniques, systems and challenges, *Computers & Security*, 2009, 28, p. 18–28.

[26] ГАМАЮНОВ, Д. Обнаружение компьютерных атак на основе анализа поведения сетевых объектов, Московский государственный университет им. Ломоносова, 2007.

[27] КОМАШИНСКИЙ, В.; СМИРНОВ, Д. *Нейронные сети и их применение в системах управления и связи*. Москва: Горячая Линия - Телеком, 2003. ISBN 5-93517-094-9.

[28] АКСЕНОВ, С.; НОВОСЕЛЬЦЕВ, В. *Организация и использование нейронных сетей (методы и технологии)*. Томск: НТЛ, 2006. ISBN 5-89503-285-0.

[29] ЗАЕНЦЕВ, И. Нейронные сети: основные модели, 1999.

[30] DZEMYDA, G.; KURASOVA, O.; ŽILINSKAS, J. *Daugiamačių duomenų vizualizavimo metodai*. Vilnius: Matematikos ir informatikos institutas, 2008. ISBN 978-9986-680-42-0.

[31] UNGURYTĖ, J. Neuroninių tinklų taikymas daugiakriteriniame optimizavime, Lietuvos edukologijos universitetas, Vilnius, 2012.

[32] STUDENKIN, O. Atvirkštinio skleidimo neuroniniai tinklai: vaizdų atpažinimas, Kauno technologijos universitetas, Kaunas, 2005.

[33] NG. A. Stanford Machine Learning. [interaktyvus]. [žiūrėta 2015-04-26]. Prieiga per internetą: <http://www.holehouse.org/mlclass/>

[34] Хабрахабр. Нейронная сеть против DDoS'a. [interaktyvus]. [žiūrėta 2015-04-26]. Prieiga per internetą: <http://habrahabr.ru/post/136237/>

[35] СЛЕПОВИЧЕВ, И.; ИРМАТОВ, П.; КОМАРОВА, М.; БЕЖИН, А. Обнаружение DDoS атак нечеткой нейронной сетью, *Известия Саратовского университета. Математика. Механика. Информатика*, 2009, 3, p. 84–89.

[36] Irvine The UCI KDD Archive Information and Computer Science University of California. KDD Cup 1999 Data. [interaktyvus]. [žiūrėta 2015-04-26]. Prieiga per internetą: <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>

[37] SANG-HYUN, C.; HEE-SU, C. Feature Selection for Intrusion Detection using NSL-KDD, *International Conference Data Mining, Civil and Mechanical Engineering (ICDMCME'2014)*, 2014, p. 90–92.

[38] КАРАЙЧЕВ, Г. Выявление аномальной активности методом адаптивных сеток, *Известия ЮФУ. Технические науки*, 2009, 100(11), p. 84–92.

[39] SRINIVASU, P.; AVADHANI, P.S. Genetic Algorithm based Weight Extraction Algorithm for Artificial Neural Network Classifier in Intrusion Detection, *Procedia Engineering*, 2012, 38, p. 144–153.

[40] Tcpdump/Libpcap. Documentation. [interaktyvus]. [žiūrėta 2015-04-26]. Prieiga per internetą: <http://www.tcpdump.org/>

[41] Хабрахабр. Фильтры захвата для сетевых анализаторов (tcpdump, Wireshark, Paketyzer). [interaktyvus]. [žiūrėta 2015-04-26]. Prieiga per internetą: <http://habrahabr.ru/post/211042/>

PRIEDAI

A priedas

A.1 lentelė. KDD Cup 1999 duomenų rinkinio atributų konvertavimo taisyklės

Tipas	**Atributo pavadinimas**	**Skaitinė vertė 1**	**Skaitinė vertė 2**
Teisėtas tinklo srautas	normal	0	0
Tinklo srauto anomalija	back	1	1
	land	2	
	neptune	3	
	pod	4	
	smurf	5	
	teardrop	6	
Protokolo tipas	TCP	7	7
	UDP	8	8
	ICMP	9	9
Žymė	OTH	10	10
	REJ	11	11
	RSTO	12	12
	RSTOS0	13	13
	RSTR	14	14
	S0	15	15
	S1	16	16
	S2	17	17
	S3	18	18
	SF	19	19
	SH	20	20
Servisas	auth	21	21
	bgp	22	22
	courier	23	23
	csnet_ns	24	24
	ctf	25	25
	daytime	26	26
	discard	27	27
	domain	28	28
	domain_u	29	29
	echo	30	30
	eco_i	31	31
	ecr_i	32	32
	efs	33	33
	exec	34	34
	finger	35	35
	ftp	36	36
	ftp_data	37	37
	gopher	38	38
	hostnames	39	39
	http	40	40

Tipas	Atributo pavadinimas	Skaitinė vertė 1	Skaitinė vertė 2
	http_443	41	41
	imap4	42	42
	IRC	43	43
	iso_tsap	44	44
	klogin	45	45
	kshell	46	46
	ldap	47	47
	link	48	48
	login	49	49
	mtp	50	50
	name	51	51
	netbios_dgm	52	52
	netbios_ns	53	53
	netbios_ssn	54	54
	netstat	55	55
	nnsp	56	56
	nntp	57	57
	ntp_u	58	58
	other	59	59
	pm_dump	60	60
	pop_2	61	61
	pop_3	62	62
	printer	63	63
	private	64	64
	red_i	65	65
	remote_job	66	66
	rje	67	67
	shell	68	68
	systat	69	69
	smtp	70	70
	sql_net	71	71
	ssh	72	72
	sunrpc	73	73
	supdup	74	74
	telnet	75	75
	tftp_u	76	76
	tim_i	77	77
	time	78	78
	urh_i	79	79
	urp_i	80	80
	uucp	81	81
	uucp_path	82	82
	vmnet	83	83
	whois	84	84

Tipas	Atributo pavadinimas	Skaitinė vertė 1	Skaitinė vertė 2
	X11	85	85
	Z39_50	86	86

B priedas

DDoS atakų klasifikacijos rezultatai

B.1 lentelė. 7 klasės, 39 atributai

		normal	**back**	**land**	**neptune**	**pod**	**smurf**	**teardrop**
MLP 39-13-7	Iš viso	6991,0000	1582,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Korektiškai	6991,0000	1582,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Nekorektiškai	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
	Korektiškai(%)	100,0000	100,0000	100,0000	100,0000	100,0000	100,0000	100,0000
	Nekorektiškai(%)	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
RBF 39-26-7	Iš viso	6991,0000	1582,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Korektiškai	5873,0000	59,0000	0,0000	6949,0000	0,0000	6992,0000	675,0000
	Nekorektiškai	1118,0000	1523,0000	12,0000	2,0000	175,0000	49,0000	0,0000
	Korektiškai(%)	84,0080	3,7295	0,0000	99,9712	0,0000	99,3041	100,0000
	Nekorektiškai(%)	15,9920	96,2705	100,0000	0,0288	100,0000	0,6959	0,0000

B.2 lentelė. 7 klasės, 30 atributų

		normal	**back**	**land**	**neptune**	**pod**	**smurf**	**teardrop**
MLP 30-16-7	Iš viso	6991,0000	1582,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Korektiškai	6990,0000	1580,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Nekorektiškai	1,0000	2,0000	0,0000	0,0000	0,0000	0,0000	0,0000
	Korektiškai(%)	99,9857	99,8736	100,0000	100,0000	100,0000	100,0000	100,0000
	Nekorektiškai(%)	0,0143	0,1264	0,0000	0,0000	0,0000	0,0000	0,0000
RBF 30-30-7	Iš viso	6991,0000	1582,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Korektiškai	5836,0000	216,0000	0,0000	6947,0000	0,0000	6992,0000	0,0000
	Nekorektiškai	1155,0000	1366,0000	12,0000	4,0000	175,0000	49,0000	675,0000
	Korektiškai(%)	83,4788	13,6536	0,0000	99,9425	0,0000	99,3041	0,0000
	Nekorektiškai(%)	16,5212	86,3464	100,0000	0,0575	100,0000	0,6959	100,0000

B.3 lentelė. 7 klasės, 20 atributų

		normal	**back**	**land**	**neptune**	**pod**	**smurf**	**teardrop**
MLP 20-22-7	Iš viso	6991,0000	1582,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Korektiškai	6991,0000	1581,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Nekorektiškai	0,0000	1,0000	0,0000	0,0000	0,0000	0,0000	0,0000
	Korektiškai(%)	100,0000	99,9368	100,0000	100,0000	100,0000	100,0000	100,0000
	Nekorektiškai(%)	0,0000	0,0632	0,0000	0,0000	0,0000	0,0000	0,0000
RBF 20-25-7	Iš viso	6991,0000	1582,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Korektiškai	4229,0000	5,0000	0,0000	6880,0000	0,0000	7041,0000	0,0000
	Nekorektiškai	2762,0000	1577,0000	12,0000	71,0000	175,0000	0,0000	675,0000
	Korektiškai(%)	60,4921	0,3161	0,0000	98,9786	0,0000	100,0000	0,0000
	Nekorektiškai(%)	39,5079	99,6839	100,0000	1,0214	100,0000	0,0000	100,0000

B.4 lentelė. 7 klasės, 10 atributų

		normal	**back**	**land**	**neptune**	**pod**	**smurf**	**teardrop**
MLP 10-23-7	Iš viso	6991,0000	1582,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Korektiškai	6991,0000	1580,0000	11,0000	6951,0000	175,0000	7041,0000	675,0000
	Nekorektiškai	0,0000	2,0000	1,0000	0,0000	0,0000	0,0000	0,0000
	Korektiškai(%)	100,0000	99,8736	91,6667	100,0000	100,0000	100,0000	100,0000
	Nekorektiškai(%)	0,0000	0,1264	8,3333	0,0000	0,0000	0,0000	0,0000
RBF 10-31-7	Iš viso	6991,0000	1582,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Korektiškai	4274,0000	56,0000	0,0000	6693,0000	0,0000	7035,0000	0,0000
	Nekorektiškai	2717,0000	1526,0000	12,0000	258,0000	175,0000	6,0000	675,0000
	Korektiškai(%)	61,1357	3,5398	0,0000	96,2883	0,0000	99,9148	0,0000
	Nekorektiškai(%)	38,8643	96,4602	100,0000	3,7117	100,0000	0,0852	100,0000

B.5 lentelė. 7 klasės, 5 atributai

		normal	**back**	**land**	**neptune**	**pod**	**smurf**	**teardrop**
MLP 5-16-7	Iš viso	6991,0000	1582,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Korektiškai	6812,0000	481,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Nekorektiškai	179,0000	1101,0000	0,0000	0,0000	0,0000	0,0000	0,0000
	Korektiškai(%)	97,4396	30,4046	100,0000	100,0000	100,0000	100,0000	100,0000
	Nekorektiškai(%)	2,5604	69,5954	0,0000	0,0000	0,0000	0,0000	0,0000
RBF 5-50-7	Iš viso	6991,0000	1582,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Korektiškai	5463,0000	0,0000	0,0000	6950,0000	124,0000	0,0000	0,0000
	Nekorektiškai	1528,0000	1582,0000	12,0000	1,0000	51,0000	7041,0000	675,0000
	Korektiškai(%)	78,1433	0,0000	0,0000	99,9856	70,8571	0,0000	0,0000
	Nekorektiškai(%)	21,8567	100,0000	100,0000	0,0144	29,1429	100,0000	100,0000

B.6 lentelė. 7 klasės, 3 atributai

		normal	**back**	**land**	**neptune**	**pod**	**smurf**	**teardrop**
MLP 3-16-7	Iš viso	6991,0000	1582,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Korektiškai	6817,0000	427,0000	3,0000	6951,0000	170,0000	7030,0000	675,0000
	Nekorektiškai	174,0000	1155,0000	9,0000	0,0000	5,0000	11,0000	0,0000
	Korektiškai(%)	97,5111	26,9912	25,0000	100,0000	97,1429	99,8438	100,0000
	Nekorektiškai(%)	2,4889	73,0088	75,0000	0,0000	2,8571	0,1562	0,0000
RBF 3-56-7	Iš viso	6991,0000	1582,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Korektiškai	5059,0000	0,0000	0,0000	6948,0000	103,0000	4,0000	520,0000
	Nekorektiškai	1932,0000	1582,0000	12,0000	3,0000	72,0000	7037,0000	155,0000
	Korektiškai(%)	72,3645	0,0000	0,0000	99,9568	58,8571	0,0568	77,0370
	Nekorektiškai(%)	27,6355	100,0000	100,0000	0,0432	41,1429	99,9432	22,9630

B.7 lentelė. 7 klasės, 2 atributai

		normal	back	land	neptune	pod	smurf	teardrop
MLP 2-12-7	Iš viso	6991,0000	1582,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Korektiškai	6955,0000	23,0000	5,0000	6951,0000	0,0000	7041,0000	675,0000
	Nekorektiškai	36,0000	1559,0000	7,0000	0,0000	175,0000	0,0000	0,0000
	Korektiškai(%)	99,4851	1,4539	41,6667	100,0000	0,0000	100,0000	100,0000
	Nekorektiškai(%)	0,5149	98,5461	58,3333	0,0000	100,0000	0,0000	0,0000
RBF 2-54-7	Iš viso	6991,0000	1582,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Korektiškai	6063,0000	0,0000	0,0000	6949,0000	0,0000	0,0000	0,0000
	Nekorektiškai	928,0000	1582,0000	12,0000	2,0000	175,0000	7041,0000	675,0000
	Korektiškai(%)	86,7258	0,0000	0,0000	99,9712	0,0000	0,0000	0,0000
	Nekorektiškai(%)	13,2742	100,0000	100,0000	0,0288	100,0000	100,0000	100,0000

B.8 lentelė. 7 klasės, 1 atributas

		normal	back	land	neptune	pod	smurf	teardrop
MLP 1-14-7	Iš viso	6991,0000	1582,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Korektiškai	655,0000	0,0000	3,0000	6951,0000	0,0000	7041,0000	0,0000
	Nekorektiškai	6336,0000	1582,0000	9,0000	0,0000	175,0000	0,0000	675,0000
	Korektiškai(%)	9,3692	0,0000	25,0000	100,0000	0,0000	100,0000	0,0000
	Nekorektiškai(%)	90,6308	100,0000	75,0000	0,0000	100,0000	0,0000	100,0000
RBF 1-44-7	Iš viso	6991,0000	1582,0000	12,0000	6951,0000	175,0000	7041,0000	675,0000
	Korektiškai	1356,0000	0,0000	0,0000	6945,0000	0,0000	7025,0000	155,0000
	Nekorektiškai	5635,0000	1582,0000	12,0000	6,0000	175,0000	16,0000	520,0000
	Korektiškai(%)	19,3964	0,0000	0,0000	99,9137	0,0000	99,7728	22,9630
	Nekorektiškai(%)	80,6036	100,0000	100,0000	0,0863	100,0000	0,2272	77,0370

B.9 lentelė. 2 klasės, 39 atributai

		normal	attack
MLP 39-23-2	Iš viso	7845,0000	16497,0000
	Korektiškai	6991,0000	16497,0000
	Nekorektiškai	854,0000	0,0000
	Korektiškai(%)	89,1141	100,0000
	Nekorektiškai(%)	10,8859	0,0000
RBF 39-41-2	Iš viso	6137,0000	16375,0000
	Korektiškai	4935,0000	15159,0000
	Nekorektiškai	1202,0000	1216,0000
	Korektiškai(%)	80,4139	92,5740
	Nekorektiškai(%)	19,5861	7,4260

B.10 lentelė. 2 klasės, 30 atributų

		normal	**attack**
MLP 30-29-2	Iš viso	6991,0000	16436,0000
	Korektiškai	6991,0000	16436,0000
	Nekorektiškai	0,0000	0,0000
	Korektiškai(%)	100,0000	100,0000
	Nekorektiškai(%)	0,0000	0,0000
RBF 30-41-2	Iš viso	6991,0000	16436,0000
	Korektiškai	6513,0000	14875,0000
	Nekorektiškai	478,0000	1561,0000
	Korektiškai(%)	93,1626	90,5026
	Nekorektiškai(%)	6,8374	9,4974

B.11 lentelė. 2 klasės, 20 atributų

		normal	**attack**
MLP 20-24-2	Iš viso	6991,0000	16436,0000
	Korektiškai	6991,0000	16436,0000
	Nekorektiškai	0,0000	0,0000
	Korektiškai(%)	100,0000	100,0000
	Nekorektiškai(%)	0,0000	0,0000
RBF 20-46-2	Iš viso	6991,0000	16436,0000
	Korektiškai	6042,0000	15347,0000
	Nekorektiškai	949,0000	1089,0000
	Korektiškai(%)	86,4254	93,3743
	Nekorektiškai(%)	13,5746	6,6257

B.12 lentelė. 2 klasės, 10 atributų

		normal	**attack**
MLP 10-22-2	Iš viso	6991,0000	16436,0000
	Korektiškai	6988,0000	16431,0000
	Nekorektiškai	3,0000	5,0000
	Korektiškai(%)	99,9571	99,9696
	Nekorektiškai(%)	0,0429	0,0304
RBF 10-49-2	Iš viso	6991,0000	16436,0000
	Korektiškai	5795,0000	15145,0000
	Nekorektiškai	1196,0000	1291,0000
	Korektiškai(%)	82,8923	92,1453
	Nekorektiškai(%)	17,1077	7,8547

B.13 lentelė. 2 klasės, 5 atributai

		normal	**attack**
MLP 5-10-2	Iš viso	6991,0000	16436,0000
	Korektiškai	6874,0000	15319,0000
	Nekorektiškai	117,0000	1117,0000
	Korektiškai(%)	98,3264	93,2039
	Nekorektiškai(%)	1,6736	6,7961
RBF 5-49-2	Iš viso	6991,0000	16436,0000
	Korektiškai	6484,0000	14261,0000
	Nekorektiškai	507,0000	2175,0000
	Korektiškai(%)	92,7478	86,7669
	Nekorektiškai(%)	7,2522	13,2331

B.14 lentelė. 2 klasės, 3 atributai

		normal	**attack**
MLP 3-20-2	Iš viso	6991,0000	16436,0000
	Korektiškai	6991,0000	14854,0000
	Nekorektiškai	0,0000	1582,0000
	Korektiškai(%)	100,0000	90,3748
	Nekorektiškai(%)	0,0000	9,6252
RBF 3-60-2	Iš viso	6991,0000	16436,0000
	Korektiškai	3187,0000	15980,0000
	Nekorektiškai	3804,0000	456,0000
	Korektiškai(%)	45,5872	97,2256
	Nekorektiškai(%)	54,4128	2,7744

B.15 lentelė. 2 klasės, 2 atributai

		normal	**attack**
MLP 2-6-2	Iš viso	6991,0000	16436,0000
	Korektiškai	6991,0000	14854,0000
	Nekorektiškai	0,0000	1582,0000
	Korektiškai(%)	100,0000	90,3748
	Nekorektiškai(%)	0,0000	9,6252
RBF 2-43-2	Iš viso	6991,0000	16436,0000
	Korektiškai	76,0000	16412,0000
	Nekorektiškai	6915,0000	24,0000
	Korektiškai(%)	1,0871	99,8540
	Nekorektiškai(%)	98,9129	0,1460

B.16 lentelė. 2 klasės, 1 atributas

		normal	**attack**
MLP 1-5-2	Iš viso	6991,0000	16436,0000
	Korektiškai	6991,0000	14854,0000
	Nekorektiškai	0,0000	1582,0000
	Korektiškai(%)	100,0000	90,3748
	Nekorektiškai(%)	0,0000	9,6252
RBF 1-56-2	Iš viso	6991,0000	16436,0000
	Korektiškai	70,0000	16426,0000
	Nekorektiškai	6921,0000	10,0000
	Korektiškai(%)	1,0013	99,9392
	Nekorektiškai(%)	98,9987	0,0608

C priedas

Atributų įtaka galutiniam *DDoS* atakų atpažinimo rezultatui

C.1 lentelė. 7 klasės, 39 atributai

Nr.	Atributo pavadinimas	MLP 39-13-7	Atributo pavadinimas	RBF 39-26-7
1	dst_host_count	1,117240	logged_in	16270,160198
2	is_guest_login	1,043420	dst_host_same_srv_rate	6766,279198
3	service	1,030209	dst_host_srv_count	6731,778169
4	dst_host_srv_count	1,027603	count	1107,012818
5	srv_rerror_rate	1,022129	dst_host_count	971,948300
6	srv_diff_host_rate	1,021131	wrong_fragment	517,243526
7	flag	1,017706	srv_count	199,684502
8	rerror_rate	1,010792	protocol_type	171,602810
9	same_srv_rate	1,007041	dst_host_serror_rate	107,240226
10	diff_srv_rate	1,005384	hot	52,931275
11	serror_rate	1,004192	land	16,024034
12	dst_host_same_srv_rate	1,003796	src_bytes	11,408699
13	dst_host_srv_serror_rate	1,003715	flag	6,724384
14	dst_host_serror_rate	1,003640	serror_rate	5,763287
15	srv_serror_rate	1,002625	service	4,851567
16	num_shells	1,001073	same_srv_rate	4,224898
17	dst_host_rerror_rate	1,001058	srv_diff_host_rate	3,632448
18	num_file_creations	1,001017	srv_serror_rate	3,229011
19	dst_host_diff_srv_rate	1,000580	dst_host_srv_rerror_rate	2,997799
20	hot	1,000539	srv_rerror_rate	2,970325
21	wrong_fragment	1,000012	dst_host_srv_serror_rate	2,513840
22	dst_host_same_src_port_rate	1,000006	dst_host_same_src_port_rate	1,820629
23	num_root	1,000001	dst_host_rerror_rate	1,703246
24	urgent	1,000000	dst_host_srv_diff_host_rate	1,285470
25	num_failed_logins	1,000000	dst_host_diff_srv_rate	1,081177
26	root_shell	1,000000	is_guest_login	1,077872
27	num_compromised	1,000000	diff_srv_rate	1,013216
28	duration	1,000000	dst_bytes	1,012154
29	land	1,000000	num_compromised	1,009794
30	dst_bytes	1,000000	num_access_files	1,003026
31	num_access_files	0,999999	num_shells	1,000028
32	su_attempted	0,999999	urgent	1,000000
33	dst_host_srv_diff_host_rate	0,999465	num_failed_logins	1,000000
34	src_bytes	0,999459	root_shell	0,999996
35	dst_host_srv_rerror_rate	0,999438	su_attempted	0,999988
36	protocol_type	0,994777	num_root	0,999968
37	count	0,970479	duration	0,999924
38	srv_count	0,961813	num_file_creations	0,999791
39	logged_in	0,941419	rerror_rate	0,997035

C.2 lentelė. 7 klasės, 30 atributų

Nr.	Atributo pavadinimas	MLP 30-16-7	Atributo pavadinimas	RBF 30-30-7
1	dst_host_same_srv_rate	11846,801899	dst_host_srv_count	1,100542
2	dst_host_srv_count	5456,459511	service	1,099282
3	dst_host_same_src_port_rate	4256,507058	flag	1,067017
4	service	3159,068661	srv_rerror_rate	1,056354
5	dst_host_count	1138,167014	srv_diff_host_rate	1,021794
6	same_srv_rate	1035,414807	dst_host_same_srv_rate	1,018991
7	dst_host_srv_serror_rate	895,857795	dst_host_same_src_port_rate	1,002248
8	wrong_fragment	752,058969	hot	1,001311
9	is_guest_login	297,956628	diff_srv_rate	1,001109
10	hot	256,416762	is_guest_login	1,000676
11	srv_rerror_rate	50,944798	src_bytes	1,000017
12	serror_rate	22,633711	dst_bytes	1,000000
13	srv_serror_rate	19,345385	num_compromised	1,000000
14	flag	12,120812	land	1,000000
15	srv_diff_host_rate	4,011033	num_access_files	0,999426
16	land	2,569313	dst_host_srv_diff_host_rate	0,999234
17	dst_host_serror_rate	2,152453	wrong_fragment	0,998870
18	dst_host_rerror_rate	2,030091	dst_host_rerror_rate	0,998731
19	dst_host_diff_srv_rate	1,416574	dst_host_diff_srv_rate	0,994855
20	diff_srv_rate	1,194656	dst_host_srv_rerror_rate	0,993667
21	dst_bytes	1,012207	same_srv_rate	0,988204
22	num_compromised	1,008963	serror_rate	0,984529
23	rerror_rate	1,006935	srv_count	0,984141
24	num_root	1,001293	srv_serror_rate	0,982684
25	num_shells	1,000009	dst_host_srv_serror_rate	0,981700
26	urgent	1,000000	protocol_type	0,980696
27	num_failed_logins	0,999999	dst_host_serror_rate	0,980341
28	duration	0,999993	count	0,969724
29	root_shell	0,999992	dst_host_count	0,931027
30	num_file_creations	0,999958	logged_in	0,913112

C.3 lentelė. 7 klasės, 20 atributų

Nr.	Atributo pavadinimas	MLP 20-22-7	Atributo pavadinimas	RBF 20-25-7
1	dst_host_srv_count	35524,986279	dst_host_same_src_port_rate	5,461640
2	dst_host_same_srv_rate	32154,936109	dst_host_srv_count	3,602330
3	dst_host_count	9641,921119	dst_host_same_srv_rate	3,376167
4	dst_host_srv_serror_rate	8224,000643	service	1,141493
5	service	3838,882679	flag	1,071037
6	dst_host_serror_rate	2309,028931	srv_rerror_rate	1,011518
7	wrong_fragment	1601,594086	dst_host_rerror_rate	1,004368
8	hot	432,879837	dst_host_srv_diff_host_rate	1,003149
9	srv_rerror_rate	364,973611	wrong_fragment	1,002844
10	flag	337,500857	dst_host_diff_srv_rate	1,001960

Nr.	Atributo pavadinimas	MLP 20-22-7	Atributo pavadinimas	RBF 20-25-7
11	is_guest_login	291,814911	hot	1,001008
12	dst_host_same_src_port_rate	145,401808	diff_srv_rate	1,000456
13	same_srv_rate	53,350451	src_bytes	1,000397
14	serror_rate	46,395973	dst_bytes	1,000001
15	land	13,983641	land	1,000000
16	srv_diff_host_rate	13,172634	num_compromised	0,999996
17	dst_host_diff_srv_rate	7,940853	is_guest_login	0,999931
18	srv_serror_rate	7,598163	dst_host_srv_rerror_rate	0,997993
19	dst_host_rerror_rate	3,165474	num_access_files	0,995744
20	diff_srv_rate	1,003525	srv_diff_host_rate	0,948050

C.4 lentelė. 7 klasės, 10 atributų

Nr.	Atributo pavadinimas	MLP 10-23-7	Atributo pavadinimas	RBF 10-31-7
1	dst_host_same_srv_rate	5058,164777	service	6,957535
2	service	4182,990184	dst_host_same_src_port_rate	4,402640
3	dst_host_srv_count	1518,666181	flag	3,332913
4	wrong_fragment	461,469941	srv_rerror_rate	1,186606
5	dst_host_srv_serror_rate	403,954068	dst_host_diff_srv_rate	1,035916
6	dst_host_count	324,565852	wrong_fragment	1,030139
7	hot	260,473703	dst_host_rerror_rate	1,001743
8	dst_host_serror_rate	141,490950	dst_host_srv_diff_host_rate	0,998834
9	flag	18,795188	dst_host_same_srv_rate	0,243022
10	srv_rerror_rate	1,166838	dst_host_srv_count	0,238913

C.5 lentelė. 7 klasės, 5 atributai

Nr.	Atributo pavadinimas	MLP 5-16-7	Atributo pavadinimas	RBF 5-50-7
1	dst_host_srv_serror_rate	199,232347	srv_rerror_rate	1,334146
2	service	89,402999	dst_host_diff_srv_rate	0,997144
3	dst_host_srv_count	62,774262	dst_host_same_src_port_rate	0,074530
4	dst_host_same_srv_rate	26,672402	flag	0,062809
5	wrong_fragment	8,654985	service	0,054984

C.6 lentelė. 7 klasės, 3 atributai

Nr.	Atributo pavadinimas	MLP 3-16-7	Atributo pavadinimas	RBF 3-56-7
1	dst_host_srv_serror_rate	89,653012	dst_host_diff_srv_rate	2,917083
2	service	72,842068	srv_rerror_rate	1,162356
3	dst_host_srv_count	12,541953	dst_host_same_src_port_rate	0,115430

C.7 lentelė. 7 klasės, 2 atributai

Nr.	Atributo pavadinimas	MLP 2-12-7	Atributo pavadinimas	RBF 2-54-7
1	dst_host_srv_serror_rate	92,614770	dst_host_diff_srv_rate	10,774204
2	service	41,982530	srv_rerror_rate	0,907824

C.8 lentelė. 7 klasės, 1 atributas

Nr.	Atributo pavadinimas	MLP 1-14-7	Atributo pavadinimas	RBF 1-44-7
1	dst_host_srv_serror_rate	30,563979	dst_host_diff_srv_rate	5,812928

C.9 lentelė. 2 klasės, 39 atributai

Nr.	Atributo pavadinimas	MLP 39-23-2	Atributo pavadinimas	RBF 39-41-2
1	dst_host_same_srv_rate	3796,337876	logged_in	1,368114
2	dst_host_srv_count	1323,031308	dst_host_count	1,280247
3	count	453,102122	dst_host_same_srv_rate	1,102286
4	dst_host_count	215,787246	dst_host_srv_count	1,081402
5	logged_in	140,173133	srv_rerror_rate	1,032609
6	protocol_type	128,088073	is_guest_login	1,030047
7	is_guest_login	71,535024	dst_host_diff_srv_rate	1,016089
8	srv_count	71,441148	service	1,015056
9	hot	50,560238	hot	1,012036
10	src_bytes	30,155181	rerror_rate	1,006022
11	wrong_fragment	7,560742	flag	1,005823
12	dst_host_same_src_port_rate	7,405975	diff_srv_rate	1,001785
13	srv_serror_rate	5,928692	num_file_creations	1,000381
14	srv_rerror_rate	4,010168	su_attempted	1,000216
15	dst_host_serror_rate	3,550192	wrong_fragment	1,000016
16	dst_host_srv_serror_rate	3,524785	num_root	1,000000
17	flag	1,513524	dst_bytes	1,000000
18	same_srv_rate	1,364597	duration	1,000000
19	dst_host_rerror_rate	1,304304	num_compromised	1,000000
20	srv_diff_host_rate	1,287363	land	1,000000
21	dst_host_srv_diff_host_rate	1,246507	num_failed_logins	1,000000
22	serror_rate	1,233518	urgent	0,999999
23	dst_host_srv_rerror_rate	1,227986	num_shells	0,999999
24	rerror_rate	1,030177	src_bytes	0,999999
25	diff_srv_rate	1,015673	dst_host_srv_diff_host_rate	0,999752
26	num_compromised	1,011373	root_shell	0,999530
27	dst_bytes	1,002112	num_access_files	0,999196
28	land	1,000521	dst_host_srv_rerror_rate	0,996499
29	num_access_files	1,000386	dst_host_rerror_rate	0,994687
30	num_file_creations	1,000078	dst_host_same_src_port_rate	0,982920
31	root_shell	1,000000	count	0,976701
32	urgent	1,000000	same_srv_rate	0,969575

Nr.	Atributo pavadinimas	MLP 39-23-2	Atributo pavadinimas	RBF 39-41-2
33	num_failed_logins	1,000000	protocol_type	0,965415
34	num_root	1,000000	srv_count	0,965073
35	su_attempted	0,999999	dst_host_serror_rate	0,962919
36	num_shells	0,999999	srv_serror_rate	0,962855
37	duration	0,999905	serror_rate	0,962794
38	dst_host_diff_srv_rate	0,987558	dst_host_srv_serror_rate	0,962072
39	service	0,970693	srv_diff_host_rate	0,832089

C.10 lentelė. 2 klasės, 30 atributų

Nr.	Atributo pavadinimas	MLP 30-29-2	Atributo pavadinimas	RBF 30-41-2
1	logged_in	42503,663545	dst_host_same_src_port_rate	5,769867
2	dst_host_srv_count	30874,601097	dst_host_srv_count	1,065985
3	dst_host_same_srv_rate	21215,465734	dst_host_count	1,059269
4	dst_host_count	5637,555569	srv_rerror_rate	1,007521
5	srv_count	2396,918320	rerror_rate	1,005219
6	protocol_type	2050,049296	flag	1,004097
7	count	1952,494905	dst_host_srv_rerror_rate	1,002183
8	is_guest_login	562,342247	dst_host_rerror_rate	1,002178
9	hot	193,526214	hot	1,000701
10	srv_rerror_rate	35,750718	diff_srv_rate	1,000700
11	dst_host_serror_rate	21,672537	src_bytes	1,000034
12	srv_diff_host_rate	18,170304	dst_host_srv_diff_host_rate	1,000007
13	src_bytes	13,809897	num_compromised	1,000001
14	land	7,637444	urgent	1,000000
15	serror_rate	3,690098	num_failed_logins	1,000000
16	flag	2,473012	root_shell	1,000000
17	same_srv_rate	2,183124	su_attempted	1,000000
18	dst_host_srv_rerror_rate	1,883476	duration	1,000000
19	dst_host_srv_diff_host_rate	1,556584	land	1,000000
20	dst_host_same_src_port_rate	1,413534	num_root	1,000000
21	dst_host_rerror_rate	1,358037	dst_bytes	1,000000
22	srv_serror_rate	1,240455	num_access_files	0,999999
23	dst_host_srv_serror_rate	1,061961	num_shells	0,999997
24	dst_bytes	1,011885	num_file_creations	0,999997
25	rerror_rate	1,004626	is_guest_login	0,999905
26	num_access_files	1,004085	wrong_fragment	0,999478
27	diff_srv_rate	1,003133	dst_host_diff_srv_rate	0,999462
28	num_compromised	1,002038	service	0,997385
29	wrong_fragment	1,001094	dst_host_same_srv_rate	0,981807
30	num_file_creations	1,000140	logged_in	0,929971

C.11 lentelė. 2 klasės, 20 atributų

Nr.	Atributo pavadinimas	MLP 20-24-2	Atributo pavadinimas	RBF 20-46-2
1	logged_in	72427,195137	flag	7,229581
2	dst_host_srv_count	40054,819529	dst_host_same_src_port_rate	3,155138
3	dst_host_same_srv_rate	22781,257179	dst_host_srv_count	2,851154
4	dst_host_count	8735,326060	dst_host_count	2,295353
5	protocol_type	2722,643901	dst_host_rerror_rate	2,116635
6	srv_count	2257,109943	diff_srv_rate	2,113338
7	is_guest_login	1180,421720	dst_host_srv_diff_host_rate	2,108032
8	count	768,915841	rerror_rate	2,107714
9	hot	344,643738	hot	2,107667
10	dst_host_serror_rate	123,722590	dst_host_srv_rerror_rate	2,107158
11	srv_rerror_rate	80,350276	src_bytes	2,105920
12	flag	24,354371	srv_rerror_rate	2,103561
13	src_bytes	17,939092	su_attempted	1,000702
14	serror_rate	7,595800	num_compromised	1,000004
15	srv_diff_host_rate	6,970226	num_failed_logins	1,000000
16	same_srv_rate	5,578367	urgent	1,000000
17	dst_host_srv_diff_host_rate	4,488250	num_root	1,000000
18	dst_host_same_src_port_rate	3,829972	duration	1,000000
19	dst_host_srv_rerror_rate	3,626090	land	1,000000
20	land	0,997376	root_shell	0,999999

C.12 lentelė. 2 klasės, 10 atributų

Nr.	Atributo pavadinimas	MLP 10-22-2	Atributo pavadinimas	RBF 10-49-2
1	logged_in	1858,336590	dst_host_same_src_port_rate	7,056780
2	dst_host_same_srv_rate	796,055354	rerror_rate	1,018476
3	dst_host_srv_count	469,711141	dst_host_count	1,004799
4	count	219,488309	hot	1,002516
5	srv_count	198,174944	diff_srv_rate	1,001583
6	hot	110,249759	dst_host_srv_rerror_rate	1,000753
7	dst_host_count	91,095188	dst_host_rerror_rate	1,000697
8	is_guest_login	14,679386	dst_host_srv_diff_host_rate	1,000505
9	protocol_type	6,163865	flag	0,996478
10	dst_host_serror_rate	3,285550	dst_host_srv_count	0,132982

C.13 lentelė. 2 klasės, 5 atributai

Nr.	Atributo pavadinimas	MLP 5-10-2	Atributo pavadinimas	RBF 5-49-2
1	logged_in	14,272470	dst_host_count	2,421307
2	dst_host_same_srv_rate	2,073793	diff_srv_rate	2,277379
3	count	2,072260	rerror_rate	1,100838
4	srv_count	2,072260	hot	1,072527
5	dst_host_srv_count	1,301872	dst_host_same_src_port_rate	0,082468

C.14 lentelė. 2 klasės, 3 atributai

Nr.	Atributo pavadinimas	MLP 3-20-2	Atributo pavadinimas	RBF 3-60-2
1	logged_in	1,856322	diff_srv_rate	7,006465
2	count	1,173918	rerror_rate	1,057069
3	dst_host_same_srv_rate	1,080818	dst_host_count	0,270995

C.15 lentelė. 2 klasės, 2 atributai

Nr.	Atributo pavadinimas	MLP 2-6-2	Atributo pavadinimas	RBF 2-43-2
1	logged_in	1,861246	diff_srv_rate	49,975022
2	count	1,049910	rerror_rate	1,007756

C.16 lentelė. 2 klasės, 1 atributas

Nr.	Atributo pavadinimas	MLP 1-5-2	Atributo pavadinimas	RBF 1-56-2
1	logged_in	3,036937	diff_srv_rate	1,376777