VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
PROGRAMŲ SISTEMŲ KATEDRA

Sukčiavimo aptikimas bekontakčiuose elektroniniuose mokėjimuose

Detecting fraud in contactless electronic payments

Magistro baigiamasis darbas

Atliko: Ignas Bobinas (parašas)
Darbo vadovas: Partn. Doc. Vaidas Jusevičius (parašas)
Recenzentas: Prof. Dr. Romas Baronas (parašas)

Vilnius – 2018

Santrauka

Darbe tiriamas sukčiavimo aptikimas atliekant elektroninius bekontakčius mokėjimus terminale, naudojančius artimojo atstumo duomenų perdavimo technologiją. Sukčiavimo aptikimas dažniausiai vykdomas kaip periodinis procesas po transakcijos patvirtinimo. Šio darbo tikslas sukurti sukčiavimo aptikimo būdą bekontakčių NFC technologija atliekamų mokėjimų apdorojimui, kuris galėtų pateikti įvertį realiu laiku bei užtikrintų pakankamą tikslumą. Siekiant šio tikslo buvo identifikuotos galimos bekontakčių mokėjimų saugumo spragos ir sukčiavimo būdai. Ištirtas sukčiavimo aptikimui reikalingas duomenų paruošimo procesas, išanalizuotos viešai prieinamos sukčiavimo aptikimo būdų realizacijos. Sukčiavimo įverčio skaičiavimo įgyvendinimui pasirinkta naudoti Bajeso tinklus, apibrėžtas įverčio skaičiavimas, tinklo struktūra bei transakcijų atributai tinkami sukčiavimo aptikimui. Pasiūlyti būdai įvertinti transakcijos sumos, atlikimo vietos, atlikimo laiko, transakcijų kiekio rizikas. Apibrėžus naudojamą sukčiavimo aptikimo tinklą suprojektuota sukčiavimo aptikimo sistema, pasiūlyta rekomenduojama sistemos diegimo konfigūracija. Atliktas įgyvendintos sukčiavimo aptikimo sistemos tikslumo ir efektyvumo tyrimas. Tyrimo metu nustatyta, kad tinklo grupavimas į nepakankamą kriterijų grupių kiekį mažina sukčiavimo aptikimo jautrumą. Vis dėlto, buvo prieita prie išvados, kad naudojant sukčiavimo aptikimui sukonkretintą Bajeso tinklą įmanoma sukurti sukčiavimo aptikimo sistemą, galinčią užtikrinti pakankamą tikslumą ir pateikiančią sukčiavimo įverčius realiu laiku.

Summary

Research focuses on fraud detection in electronic contactless payments executed through payment terminal using near field communication technology. In most cases, fraud detection is implemented as periodic batch process executed after payment transactions is approved. Objective of research was creation of fraud detection method for contactless payments executed using NFC technology. Created method should be able to evaluate and provide transaction risk estimates real time and ensure sufficient accuracy. In order to fulfill this objective existing security vulnerabilities and existing fraud schemes for contactless cards were analyzed, required data preprocessing processes were identified. After that, publicly accessible fraud detection method implementations were analyzed. During implementation analysis Bayesian networks were selected as suitable method for detection system implementation. Furthermore, fraud risk evaluation in Bayesian network and network structure was defined. Base transaction attributes suitable for fraud detection were selected, methods of evaluating transaction amount risk, location risk, time risk and transaction count risk was suggested. Furthermore fraud detection system was designed and implemented, recommended deployment configuration was defined. After that accuracy and effectiveness of created system was measured. It was concluded that network grouping into small amount of groups negatively affects precision of fraud detection system, however it is possible to create fraud detection system which can ensure sufficient accuracy and efficiency to evaluate transactions real time.

TURINYS

Įvadas

Tyrimo objektas

Bekontakčiai mokėjimai tai – mokėjimai atliekami prie mokėjimo terminalo, naudojantis bekontakte kortele, mobiliąja pinigine išmaniajame telefone ar kitu įrenginiu susietu su jūsų mokėjimo kortele [UCA16b]. Šiame darbe nagrinėjami NFC[1] technologija atliekami bekontakčiai mokėjimai. Mokėjimus atliekamus bekontakčiu būdu pagal mokėjimo sumą galima suskirstyti į dvi kategorijas: mikro-mokėjimai atliekami mažesni nei nustatytasis limitas, makro-mokėjimai, kurių suma viršija nustatytąjį limitą. Atliekant mikro-mokėjimus nereikalinga papildoma autentifikacija, pakanka priglausti mokėjimo įrenginį prie terminalo. Makro-mokėjimų atveju naudojama mokėtojo autentifikacija (mokėjimo kortelės PIN kodas, mobiliojo įrenginio užrakto atrakinimas).

„Sukčiavimas mokėjimuose yra išskirtinė asmenybės vagystės forma, į kurią įeina neautorizuotas kito asmens kredito kortelės informacijos pasisavinimas su tikslu apmokėti pirkinius ar išimti lėšas“ [KMN+15]. Mokėjimų saugumas yra užtikrinamas įvairiais būdais: techniniais sprendimais, dviejų veiksnių mokėtojo autentifikacija, sukčiavimo aptikimu ir kt. Sukčiavimo aptikimas leidžia padidinti mokėjimų saugumą, mėginant identifikuoti ar mokėjimas yra atliktas autorizuoto mokėtojo. Šio darbo tyrimo objektas yra sukčiavimo aptikimas atliekant elektroninius bekontakčius mokėjimus terminale naudojančius artimojo atstumo duomenų perdavimo technologiją.

Problema

Sukčiavimo aptikimas dažniausiai vykdomas kaip periodinis procesas po transakcijos patvirtinimo. Dėl bekontakčių mokėjimų specifikos bei technologinės pažangos, reikalavimai sukčiavimo aptikimui performuluojami. Šiomis dienomis, svarbiausi sukčiavimo aptikimui keliami reikalavimai yra tikslumas, pakankamo, kad atsakas būtų pateikiamas realiu laiku, efektyvumo užtikrinimas [JKK+16].

Atlikus akademinėje literatūroje pateikiamų realizacijų analizę nustatyta, kad tyrimuose didžiausias dėmesys skiriamas atskirų sukčiavimo aptikimo proceso dalių įtakai aptikimo tikslumui. Analizuojamos aptikimo proceso dalys: duomenų apdorojimas, įverčių skaičiavimas, duomenų analizės metodai. Norint įvertinti sukčiavimo aptikimo metodų efektyvumą, lyginti juos tarpusavyje bei įvertinti galimybes pateikti sukčiavimo įvertį prieš patvirtinant transakciją (toliau darbe – realiu laiku), reikia analizuoti visą aptikimo procesą: duomenų paruošimą,

[1] NFC – near field communication. Artimojo ryšio.

sukčiavimo įverčio identifikavimą. Taip pat būtina ištirti sistemos architektūros įtaką sukčiavimo aptikimo efektyvumui, kadangi įgyvendinančios sistemos realizacija daro įtaką gebėjimui pateikti sukčiavimo įvertį realiu laiku.

Tyrimo aktualumas

Atsiskaitymai už prekes ar paslaugas yra neatsiejama mūsų kasdienio gyvenimo dalis. Tačiau grynieji pinigai jau nebėra pagrindinis atsiskaitymo būdas. Didžiojoje Britanijoje 2015 m. mažiau nei pusė vartotojų mokėjimų buvo atlikti grynaisiais pinigais [Col16]. Prie šio pasiekimo prisidėjo ir elektroniniai bekontakčiai mokėjimai. Šis mokėjimų tipas populiarėja visame pasaulyje. Lietuvoje 2016 m. buvo išleistos pirmosios bekontakčius mokėjimus palaikančios kortelės. Didžiojoje Britanijoje per pastaruosius metus bekontakčių mokėjimų apimtys išaugo nuo 89 mln. iki 260 mln. svarų sterlingų ir 2016 m. sudarė 21 proc. visų atliktų mokėjimų kortelėmis [UCA16a].

Bekontakčiai mokėjimai atliekami prie mokėjimo terminalo yra priskiriami kortelė pateikiama (angl. *card present*) mokėjimų grupei ir yra įgyvendinami naudojant tuos pačius saugumo standartus kaip ir įprastų mokėjimo kortelių su mikroschemomis atveju. Nepaisant tobulinamų saugumo standartų yra pastebima, kad sukčiavimo mokėjimuose apimtys bėgant laikui ne mažėja, o didėja. Magnetines korteles pakeitus kortelėmis su mikroschemomis atsiskaitymai kortelėmis tapo žymiai saugesni, tačiau ir tai nepadėjo sumažinti sukčiavimo apimčių. Nepaisant esamų saugumo priemonių sukčiavimas yra reali grėsmė. Egzistuoja techninio pobūdžio atakos išnaudojančios trūkumus įrenginiuose bei protokoluose. Užkirsti kelią tokio tipo sukčiavimui tiesioginėmis prevencijos priemonėmis yra sunku. Rinkoje paplitę įrenginiai su potencialiomis saugumo spragomis: mokėjimo terminalai, kortelės su mikroschemomis.

Žinomų atakų įgyvendinimui reikalingos gilios techninės žinios ir pasiruošimas, tačiau nereikia tikėtis, kad dėl atakų sudėtingumo bekontakčiai mokėjimai yra saugūs. Tobulėjant saugumo technologijoms kartu tobulėja ir sukčiai, todėl būtina atkreipti dėmesį į naują mokėjimų būdą bei užtikrinti jo saugumą.

Užtikrinti saugumą atliekant mokėjimus galima daugeliu būdų, sukčiavimo aptikimas yra vienas iš jų. Vystantis technologijoms siekiama supaprastinti atsiskaitymui reikalingus mokėtojo veiksmus. Atliekant mokėjimus bekontakčiu būdu, būtų galima visais atvejais reikalauti mokėtojo autentifikacijos ar net dviejų veiksnių autentifikacijos, tačiau apmokėjimo procedūra būtų sudėtinga ir ilgai trunkanti. Sukčiavimo aptikimas leistų identifikuoti ir padėti išvengti sukčiavimo nekeičiant protokolų bei įrenginių. Manoma, kad pakankamai ištobulinus sukčiavimo aptikimą, būtų galima mažinti tiesiogines klientą apsunkinančias saugumo priemones [LRL10]. Galimybė pateikti sukčiavimo įvertį prieš bekontakčiu būdu atliktos

transakcijos patvirtinimą potencialiai leistų padidinti mokėjimų limitus ir sumažinti finansinius praradimus.

Sukčiavimo aptikimas jau ilgą laiką taikomas įvairiose srityse. Visi atsiskaitymai atliekami elektroninėse parduotuvėse taip pat mokėjimai atliekami prie mokėjimo terminalų naudojant įprastą mokėjimo kortelę yra tikrinami įvairių tipų sukčiavimo aptikimo būdais. Vieni jų remiasi ekspertų parengtais taisyklių rinkiniais (angl. *rule based*), kiti pasitelkia mašininio mokymosi algoritmus. Mokslinėje literatūroje sutinkami jau aprašyti specializuoti būdai pritaikyti šių tipų mokėjimams. Vis dėlto, bekontakčiai mokėjimai iš šių mokėjimo būdų išsiskiria savo specifiniais autentifikacijos reikalavimais dėl kurių naudingas sukčiavimo įvertinimas realiu laiku. Kadangi šiuo metu nėra viešai prieinamo sukčiavimo aptikimo būdo orientuoto į bekontakčius mokėjimus, galinčio pateikti įvertį realiu laiku, buvo suformuluotas darbo tikslas – sukurti specializuotą sukčiavimo aptikimo būdą bekontakčių, NFC technologija paremtų, mokėjimų apdorojimui.

Darbo tikslai ir uždaviniai

Magistro baigiamojo darbo tikslas yra sukurti sukčiavimo aptikimo būdą specializuotą bekontakčių NFC technologija atliekamų mokėjimų apdorojimui, kuris galėtų pateikti įvertį realiu laiku bei užtikrintų pakankamą tikslumą. Siekiant šio tikslo, turi būti išspręsti šie uždaviniai:

- identifikuoti galimus sukčiavimo būdus;
- identifikuoti galimus bekontakčių mokėjimų sukčiavimo aptikimui tinkamus atributus;
- išanalizuoti esamus sukčiavimo aptikimo būdus;
- apibrėžti sukčiavimo aptikimo būdą;
- apibrėžti sukčiavimo aptikimo būdą įgyvendinančią sistemą;
- įvertinti sistemos efektyvumą ir įgyvendinamumą;
- realizuoti sukčiavimo aptikimo būdą;
- įsitikinti sukurto būdo veiksmingumu.

1. Sukčiavimo aptikimo apžvalga

1.1. Mokėjimų saugumas

Prieš gilinantis į saugumo užtikrinimo metodus yra tikslinga susipažinti su bendru mokėjimų saugumo kontekstu ir potencialiomis grėsmėmis. Mokėjimus atliekamus prie mokėjimų terminalo bekontakčiu būdu naudojantis NFC technologija galima suskirstyti į dvi grupes, pagal įrenginius naudojamus mokėjimui atlikti:

- atliekami mokėjimo kortele;
- atliekami mobiliuoju įrenginiu.

Norint atsiskaityti mokėjimo kortele bekontakčiu būdu reikalinga speciali mokėjimo kortelė, kurioje įdiegta NFC antena. Neskaitant šios antenos, tokia mokėjimo kortelė niekuo nesiskiria nuo įprastos kortelės su mikroschema. Bekontaktėje kortelėje išlieka ir įprastinė mikroschema ir kontaktai įgalinantys atsiskaityti mokėjimo terminaluose nepalaikančiuose bekontakčio atsiskaitymo funkcijos. Taip pat svarbu pabrėžti, kad tiek kontaktinis, įdėjus kortelę į mokėjimo terminalo skaitytuvą, tiek bekontaktis atsiskaitymas, atliekamas naudojant tą patį saugumo standartą. Atliekant bekontaktį mokėjimą terminalas taip pat apsikeičia duomenimis su kortelėje esančia mikroschema. Esminis skirtumas, kad duomenų apsikeitimas nevyksta tiesiogiai per mikroschemos kontaktus, o per kortelėje esančią anteną.

„Mobilusis mokėjimas yra mokėjimo paslauga atliekama iš ar per mobilųjį įrenginį[2]“ [WHS16]. Nors kiekvienas paslaugų teikėjas turi savitų realizacijos skirtumų, tačiau bendrai apmokėjimo procesas yra panašus. Mobiliųjų mokėjimų atveju įrenginys veikia tik kaip komponentas perduodantis informaciją terminalui, todėl norint atlikti mokėjimą vien mobiliojo įrenginio nepakanka. Šio tipo atsiskaitymai atliekami mobiliosios piniginės principu. Pirmiausia mokėjimo kortelė registruojama pas mobiliųjų apmokėjimų paslaugų teikėją[3]. Teikėjas suteikia mobiliąją piniginę. Atliekant mokėjimą prie terminalo yra apsikeičiama tik mobiliosios piniginės duomenimis, todėl nėra atskleidžiami mokėjimo kortelės duomenys. Taip pat reikia pabrėžti, kad atsiskaitymas bekontakčiu būdu NFC pagalba nėra vienintelis būdas atlikti mobilųjį mokėjimą. Siūlo mobiliuosius mokėjimus suskirstyti į grupes [WHS16]:

- atliekami prie terminalo;
- atliekami per mokėjimo platformą;
- atliekami per telefonijos paslaugų operatorių.

[2] Mobile payment is a payment service performed from or via a mobile device

[3] Šiuo metu populiariausi mobiliųjų mokėjimo paslaugų tiekėjai: „Apple pay“, „Android pay“, „Samsung pay“.

Šiuo atveju bekontakčiai NFC mokėjimai priskiriami mokėjimams atliekamiems prie mokėjimų terminalo. Svarbu paminėti, kad egzistuoja kiti bekontakčio atsiskaitymo būdai mobiliesiems įrenginiams, tačiau šiame darbe orientuojamasi tik į mobiliuosius mokėjimus vykdomus prie mokėjimų terminalo naudojant NFC protokolą, atsiskaitant mobiliąja pinigine, susieta su mokėjimo kortele.

1.1.1. Mobiliųjų mokėjimų saugumas

„Android“ operacinę sistemą naudojančiuose mobiliuosiuose įrenginiuose, mokėjimams atlikti skirti duomenys saugomi specialiame saugiajame elemente (angl. *secure element*). Saugusis elementas – fiziškai atskirta įrenginio dalis su ribota prieiga. „Android“ atveju teoriškai saugiojo elemento prieiga suteikta tik sistemos kūrėjų – „Google“ programinei įrangai. Saugiajame elemente saugomi tik duomenys susiejantys įrenginį su mobiliąja pinigine [KMN+15]. Juos pasisavinus galima panaudoti atsiskaitymams, tačiau toks saugumo mechanizmas padeda užtikrinti, kad perėmus mobilųjį įrenginį nepavyktų gauti kortelės duomenų, kuriuos galima panaudoti kitais sukčiavimo metodais. Taip pat svarbu nepamiršti, kad jokia kenkėjiška programinė įranga negali perimti PIN kodo, kadangi jis pateikiamas ir apdorojamas terminale.

Nepaisant šių saugumo mechanizmų teorinės galimybės sukčiavimui išlieka, o viena iš pagrindinių saugumo problemų yra įrenginių daugiafunkciškumas [KMN+15]. Mobilieji įrenginiai, priešingai nei mokėjimo terminalai, yra skirti programėlėms bei pramogoms, todėl nėra apsaugoti nuo papildomos programinės įrangos įdiegimo. Yra tikėtina, kad programinėje įrangoje pasitaikys saugumo spragų, kurias sukčiai mėgins išnaudoti, tačiau tik nuo pačių vartotojų priklauso operacinės sistemos ar programėlių atnaujinimų sudiegimas. Egzistuoja saugumo spragos leidžiančios gauti papildomų prieigos teisių ar perimti jas iš kitų aplikacijų [KMN+15]. Taip pat pabrėžiama, kad prieiga prie NFC įrangos nėra ribojama, todėl programėlės įrenginyje galėtų inicijuoti duomenų apsikeitimą su mokėjimų terminalu naudojantis NFC technologija. Mobilieji įrenginiai nėra atsparūs paketų perėmimo (angl. *sniffing*), užsimaskavimo (angl. *spoofing*), išviliojimo (angl. *phishing*) ir kitoms atakoms [WHS16]. Todėl nors kenkėjiška programinė įranga mobiliajame įrenginyje negali pasisavinti kortelės duomenų bei PIN kodo, yra ir kitų būdų pakenkti mokėjimų saugumui. Egzistuoja kenkėjiška programinė įranga, kuri gali rinkti įvairią informaciją apie įrenginio naudotoją [WHS16]. Tokio tipo programinę įrangą galima naudoti kaip priemonę, galinčią padėti apeiti apgaulės aptikimą, pavyzdžiui, sekant asmens geografinę vietą, atsiskaitymų laikus ir vietas.

Įmanoma duomenų perdavimo ataka apeinanti saugumo sistemas naudojantis kenkėjiška programine įranga [KMN+15]. Šios atakos pagrindinė idėja labai paprasta. Atakos metu

apmokėjimo prašymas buvo perduotas į aukos mobilųjį įrenginį ir patvirtintas naudojant aukos duomenis

Siekiant įgyvendinti šią ataką, pirmiausia reikia užkrėsti aukos įrenginį kenkėjiška programine įranga, kuri iš sukčiaus galėtų priimti prašymą atlikti mokėjimą ir perduoti įrenginyje mokėjimus apdorojantiems procesams. Radęs auką, sukčius savo mobiliuoju įrenginiu inicijuoja mokėjimą prie terminalo, tada per perdavimo serverį susisiekia su aukos mobiliajame įrenginyje įdiegta programine įranga. Ši programinė įranga integruojasi su mokėjimo duomenis valdančiais procesais. Gavusi patvirtinimą bei mokėjimui atlikti reikalingus duomenis, juos per tinklą perduoda atgal į sukčiaus mobilųjį įrenginį. Sukčiaus įrenginys, papildomos programinės įrangos pagalba, gavęs reikiamus duomenis juos perduoda į mokėjimo terminalą, kuris patvirtina transakciją. Tokiu būdu sukčius atsiskaito naudodamas kito asmens sąskaitos duomenis, jam to net nepastebint.

Atliekant tokią transakciją vienintelis pastebimas požymis išskiriantis nuo įprastinio atsiskaitymo - pailgėjęs apdorojimo laikas [KMN+15]. Atsiskaitant tiesiogiai mokėjimo kortele transakcija yra patvirtinama pakankamai greitai, tačiau pridėjus apsikeitimą duomenimis per tinklą, su kitoje pasaulio vietoje esančiais aukų įrenginiais, atsiskaitymo laikas prailgėjo keliomis sekundėmis, o tai galėtų sukelti pardavėjų įtarimą. Vis dėlto, šis požymis nėra pakankamas, kadangi sukčius galėtų paneigti ilgą patvirtinimo laiką kaip techninius nesklandumus. Jei pavyktų tokiu būdu masiškai užkrėsti daugiau mobiliųjų įrenginių, mokėjimus būtų galima paskirstyti kelioms aukoms, taip neatkreipiant dėmesio į sukčiavimą. Nuo šios atakos apsisaugoti padėtų autentifikacija su PIN kodo tikrinimu terminale, tačiau atliekant šio tipo mokėjimus šis autentifikacijos metodas netaikomas, todėl vienintelis saugumo sluoksnis galintis padėti šiuo atveju yra sukčiavimo aptikimas [KMN+15].

1.1.2. Mokėjimo kortelių saugumas

Mokėjimo kortelių atveju mokėjimui atlikti reikalingi duomenys saugomi kortelės mikroschemoje. Mikroschema su terminalu bendrauja remdamasi EMV saugumo standartu. Šio standarto laikosi visos mokėjimo kortelės su mikroschema, nepriklausomai nuo to ar kortelė skirta bekontakčiam mokėjimui ar įprastam mokėjimui naudojantis skaitytuvu. Palaikomi 4 kortelės turėtojo verifikacijos būdai [EMV11]:

- PIN kodas tikrinamas tinkle (angl. *online PIN*);
- PIN kodas tikrinamas terminale (angl. *offline PIN*);
- pateikiamas parašas;
- verifikacija netaikoma.

Verifikacijos metodą pasirenka pardavėjo terminalas priklausomai nuo techninių galimybių. Verifikacijos metodo parinkimas yra svarbus žingsnis, kadangi nuo jo priklauso

teisinė atsakomybė dėl sukčiavimo. Verifikacijos metodai su PIN kodu laikomi tokiais saugiais, kad atsakomybė už finansinius praradimus tenka pačiam kortelės savininkui, kadangi laikoma, kad protokolas saugus ir sukčiavimas galėjo įvykti tik dėl atmestino PIN kodo saugojimo.

Pirmasis verifikavimo metodas laikomas saugiausiu – pateikimas PIN kodas. PIN kodas, naudojantis tinklo prieigą, patikrinamas pas kortelę išdavusią šalį. Kai tinklo prieiga nėra galima, naudojamas antrasis metodas, kai PIN kodas yra pateikiamas ir verifikuojamas kortelėje esančios mikroschemos pagalba. Šioje mikroschemoje saugomas užšifruotas PIN kodas ir kiti reikalingi duomenys. Parašo atveju - pasitikima pardavėju, kortelės naudotojas turi pasirašyti, o pardavėjas turėtų užtikrinti, kad parašas sutampa su parašu ant kortelės nugarėlės. Paskutinysis atvejis - verifikacija išvis nenaudojama. Bekontaktės mokėjimo kortelės išsiskiria tuo, kad mokėjimai iki tam tikros vertės laikomi mikro transakcijomis ir atliekami netaikant verifikacijos. Šiuo atveju, kad kortelės savininkas nepatirtų didelių finansinių praradimų, yra ribojamas atsiskaitymų be verifikacijos kiekis.

1.1.2.1. Tarpininko ataka

Egzistuoja sukčiavimo būdas, kurį įmanoma atlikti naudojantis bet kuria EMV standartu paremta mokėjimo kortele, naudojant verifikacijos metodą PIN kodas tikrinamas terminale, tiek atsiskaitant kontaktiniu būdu, tiek bekontakčiu [MDA+10]. Kortelės mikroschemoje yra saugomi įvairūs duomenys, tarp jų kortelės privatusis raktas, viešasis raktas, PIN kodas [EMV11]. Įdėjus kortelę į terminalą, terminalas kartu su pradiniais duomenimis gauna kortelės viešąjį raktą. Asmuo suveda PIN kodą, terminalas dar karta kviečia kortelės komandas – šį kartą komandą GET CHALLANGE. Ši komanda reiškia, kad pradedama autentifikacijos fazė – kortelė grąžina atsitiktinį sugeneruotą 8 skaitmenų skaičių terminalui. Terminalas apjungia grąžintą atsitiktinį skaičių su kliento įvestu PIN kodu ir užšifruoja mikroschemos viešuoju raktu, perduoda kontaktais duomenis. Mikroschema gavusi autentifikacijos prašymą, savo privačiuoju raktu išsišifruoja siųstą pranešimą ir atlieka reikalingus patikrinimus. Patikrinama ar atsitiktinis sugeneruotas skaičius sutampa su terminalo pateiktu skaičiumi, taip pat patikrinama ar pateiktas PIN kodas sutampa su kortelėje saugomu PIN kodu. Sėkmingos autentifikacijos atveju mikroschema terminalui grąžina statuso kodą 9000.

Šiame procese egzistuoja trūkumas – terminalas grąžiną statinį specifikacijoje aprašytą pranešimą. Su realiomis mokėjimo kortelėmis ir mokėjimo terminalais pakankamai paprasta pateikti fiktyvų statinį atsakymą terminalui [MDA+10]. Norint įgyvendinti ataką mokėjimo kortelė įkišama į fiktyvų terminalą, kuris bendrauja su tikruoju mokėjimo terminalu. Fiktyvus terminalas atlieka tarpininko vaidmenį (angl. *man in the middle*). Pradžioje vykdoma įprastinė procedūra, kortelė siunčia duomenis su viešuoju raktu, tarpininkas perduoda nepakeistus paketus tikrajam terminalui, terminalas tarpininkavimo net nepastebi. Terminale suvedamas betkoks PIN

kodas, taip pat per tarpininką įvyksta apsikeitimas atsitiktiniu sugeneruotu skaičiumi. Terminalas prieš siųsdamas PIN kodą ir atsitiktinio skaičiaus apjungimo rezultatą užkoduoja mikroschemos viešuoju raktu, todėl tarpininkas negali perskaityti žinutės ir gauti joje užšifruoto PIN kodo, tačiau įrenginys tarpininkas to nesiekia, iš perduodamos komandos pakanka identifikuoti, kad vyksta PIN kodo verifikavimas ir šį kartą pats tarpininkas grąžina fiktyvų 9000 kodą terminalui, neperdavęs užšifruotų duomenų su PIN kodu į kortelę patikrinimams. Duomenys nepasiekė kortelės ir verifikavimas kortelėje niekad neįvyko, tačiau terminalas gavo patvirtinimą, kad PIN kodas teisingas ir patvirtino transakciją.

1.1.2.2. Fiktyvaus terminalo ataka

Rinkoje pasirodžius bekontaktėms kortelėms, jos įgalino naujos atakos galimybę. Egzistuoja trūkumai protokole leidžiantys išgauti atsiskaitymui reikalingus kortelės duomenis [Emm16]. Ataka įgyvendinama apjungus kelias protokolo savybes. Bekontaktė sąsaja leidžia bendrauti su mokėjimo kortele, jai esant savininko kišenėje. Taip pat naudojantis bekontakte sąsaja protokole numatytas atsiskaitymas be PIN kodo. Bekontaktės kortelės nereikalauja verifikacijos kai terminalas prašo atsiskaitymo kita valiuta, o pardavėjo duomenys nėra užšifruojami pranešime, gautame iš kortelės.

Atakai reikalingas mobilusis įrenginys su kenkėjiška programine įranga simuliuojančia terminalą. Ataka pradedama įdiegus fiktyvią programinę įrangą į sukčiaus mobilųjį įrenginį. Įrenginyje nurodomas užsienio valiutos kodas, kuriuo bus reikalaujama atlikti mokėjimą bei suma. Atakos metu, mobilųjį įrenginį reikia priglausti kuo arčiau aukos mokėjimo kortelės, kad būtų galimas duomenų apsikeitimas bekontakčiu būdu. Fiktyvus terminalas, identifikavęs kortelę, pradeda duomenų apsikeitimą ir prašo apmokėti sukonfigūruotą sumą. Kortelė pateikia transakcijai atlikti reikalingus duomenis. Šie duomenys išsaugomi ir vėliau panaudojami atlikti mokėjimui. Kadangi grąžinamame duomenų rinkinyje kortelė neužšifruoja duomenų pardavėjo identifikavimui, norimi pardavėjo duomenys pridedami prieš atliekant atsiskaitymą.

Pastebima, kad ši ataka turi ir trūkumų. Iš vienos aukos galime gauti mokėjimo duomenis atlikti vienai transakcijai, o fiktyvų terminalą prie aukos kortelės reikia pridėti labai nedideliu atstumu. Vis dėlto, tokio tipo sukčiavimas yra rimta grėsmė, kadangi šio tipo atakas labai patogu atlikti masinio susibūrimo vietose, kur vienu metu galima nuskaityti daugelio kortelių duomenis [Emm16].

Panašių sukčiavimo būdų, paremtų fiktyvaus terminalo idėja, buvo identifikuota ir daugiau. Egzistuoja analogiškas kortelės duomenų, reikalingų mokėjimui atlikti, pasisavinimo būdas atliekant mokėjimą ta pačia valiuta [Gre12]. Šiuo sukčiavimo būdu taip pat gaunami duomenys leidžiantys atlikti vieną transakciją.

1.1.3. Sukčiavimo elgsenos šablonai

Sukčių elgsena pasižymi tam tikrais elgsenos šablonais, ypač kai jų veiklą riboja galimų atakų kiekis [Mon04]. Dėl šios priežasties, nagrinėjami pagrindiniai, pastebimi elgsenos šablonai.

Vienas iš paprasčiausių būdų, kuriuo sukčiai siekia būti nesusekti – vienu metu sukčiaujama išnaudojant kelis pardavėjus [Mon04]. Sukčius gavęs transakcijai atlikti reikalingus duomenis, apsiperka pas pardavėją, tačiau kitą kartą grįš pas tą patį pardavėją po labai ilgo laiko periodo, jeigu jis turės galimybę dar kartą atsiskaityti kortelės duomenimis, jis tai atliks pas kitą pardavėją ir dažniausiai sieks įsigyti didelės vertės prekes. Pardavėjo pusėje tokį sukčiavimo būdą aptikti sunku. Mokėtojo apdorotojo sukčiavimo aptikimo sistemoje tokius atsiskaitymus galima būtų identifikuoti, jei sukčius atsiskaitinės ta pačia kortele per trumpą laiko periodą, pirkdamas daug didelės vertės pirkinių. Šiuo atveju aptikimo sistema, įmokos apdorotojo pusėje, neturės prieinamų užsakymo duomenų, tačiau tokias transakcijas atskirti galima tiesiog iš jų didelės vertės.

Sutinkama ir priešinga elgsena - kai sukčius išnaudoja tą patį pardavėją daugybę kartų naudodamas skirtingus duomenis. Atakos tipiškai vyksta labai trumpu laikotarpiu, tokias atakas galima mėginti atskirti pagal apsipirkimo dažnį ar vertinant transakcijos duomenų pokyčių retumą. Vis dėlto, jei sukčius turės prieigą prie kelių kortelių, kurias naudos su mažu pasikartojimu, aptikimas bus žymiai sudėtingesnis.

Niekas taip negąsdina pardavėjų, kaip organizuoti sukčiai, jie randa silpnuosius taškus apsaugos nuo sukčiavimo procesuose ir juos išnaudoja [Mon04]. Sukčiai kantrūs. Jie neskuba ir ilgai mokosi saugumo politikos ir saugumo procedūrų. Tipiškai iškart naudoja kelias skirtingas atakas, kad išsiaiškintų kaip į jas reaguojama ir rastų būdą atlikti masyvesnę ataką. Naudodami socialinę inžineriją tyrinėja saugumo šablonus. Dažnai organizuoja atakas per šventinius laikotarpius kai apsipirkimo mastai ypač dideli ir dauguma asmenų leidžia lėšas ne pagal jiems įprastus šablonus. Šio tipo sukčiavimo aptikimas ypač sudėtingas. Organizuotos grupuotės dažnai naudoja tikras, asmenybės vagystės pagalba gautas, sąskaitas ir korteles su aktyviais naudojimo periodais ir tvarkinga kredito istorija, kadangi dažnai juos galima susieti su neteisėtu kortelių duomenų gavyba užsiimančiais asmenimis [Mon04].

Vidinis sukčiavimas vykdomas, kai pardavėjo darbuotojas dirba kartu su sukčiais. Tai gali būti tiesiog darbas įmonės viduje ir naudojamų saugumo bei sukčiavimo aptikimo technikų atskleidimas sukčiams. Taip pat įmanomi atvejai, kai vidinis darbuotojas pats pateikinėja užsakymus. Tokiais atvejais padeda paprasta priemonė – kiekvienos atliktos transakcijos susiejimas su atlikusiu pardavėju, tačiau apgaulės aptikimo sistemoms šie duomenys nėra prieinami ir aptikimo sistema negalės identifikuoti tokių sukčiavimo atvejų.

1.1.4. Mokėjimų saugumo apibendrinimas

Apibendrinant galima teigti, kad nepaisant taikomų saugumo priemonių sukčiavimas yra reali grėsmė. Identifikuotoms atakoms įgyvendinti reikalingos gilios techninės žinios ir pasiruošimas, nes reikalinga specializuota programinė įranga ar kiti techniniai sprendimai. Tačiau nereikia tikėtis, kad dėl to šios mokėjimų sritys yra saugios. „Su pardavėjais aš dirbu jau ilgus metus ir aš vis dar stebiuosi kūrybingumu sukčių ir metodais, kuriuos sugalvoja, kad apgautų pardavėjus, šie žmonės nėra kvaili neišsilavinę vagys, jie išsilavinę, nagingi ir kantrūs“ [Mon04]. Svarbu paminėti, kad sukčiavimas juda mažiausio pasipriešinimo kryptimi, pridėjus naujų saugumo priemonių, sukčiavimas neišnyksta, o persiorientuoja į kitą sritį. Bankai ignoruoja šiuos identifikuotus sukčiavimo būdus kaip sunkiai tikėtinus, o tai galėtų lemti dar didesnį sukčių susidomėjimą [Gre12]. Todėl galima teigti, kad tokios techninėmis žiniomis paremtos atakos yra reali grėsmė bekontakčiams mokėjimams, kadangi investavus į infrastruktūrą, sukčiavimas sunkiai pastebimas ir lengvai įgyvendinamas dideliais mastais.

Kadangi identifikuotos atakos techninio pobūdžio, remiasi trūkumais komponentuose ar protokoluose, gali būti sunku užkirsti kelią šiam sukčiavimui tiesioginėmis prevencijos priemonėmis. Protokolo lygio problemos yra sunkiai išsprendžiamos, kadangi rinkoje jau paplitę mokėjimo terminalai bei kortelės su mikroschemomis. Sukčiavimo aptikimas leistų identifikuoti sukčiavimą ir padėti jo išvengti, nekeičiant protokolo bei įrenginių.

Techninis atakos kompleksiškumas lemia ir tai, kad sukčiavimas yra tikėtinas iš organizuotų grupuočių, kurios potencialiai siekdamos būti nesusektos, neturėtų remtis primityviais elgsenos šablonais. Dėl šios priežasties galima teigti, kad siekiant užtikrinti bekontakčių mokėjimų saugumą, nėra tikslinga mėginti identifikuoti sukčiavimui būdingus elgsenos šablonus, reikėtų orientuotis į priešingo tipo metodikas, paremtas klientų normalios elgsenos šablonais.

1.2. Sukčiavimo aptikimas

Elektroninių mokėjimų saugumo užtikrinimo priemonės yra skirstomos į 3 grupes [ZK17]:

- atgrasymas nuo sukčiavimo (angl. *fraud deterrence*);
- sukčiavimo išvengimas (angl. *fraud prevention*);
- sukčiavimo aptikimas (angl. *fraud detection*).

Atgrasymu laikomos įvairios teisinės ir socialinės priemonės. Pavyzdžiui, kuriami kultūriniai pamatai visuomenėje - žmonės skatinami nesiimti sukčiavimo. Konkrečios priemonės skiriasi priklausomai nuo aplinkos, tačiau bendru atveju, tai apima įkalinimą, pinigines baudas ar tiesiog socialinę netoleranciją iš kitų gyventojų.

Sukčiavimo išvengimo tikslas apibrėžti metodus ir strategijas naudojamas išvengti sukčiavimo atvejų, kurių nepavyko atgrasyti. Tai įprasti saugumo užtikrinimo metodai, paremti techniniais standartais, dviejų faktorių autentifikacija ir kitos priemonės padedančios užkirsti kelią prieš sukčiavimui įvykstant.

Tuo tarpu, sukčiavimo aptikimo tikslas yra identifikuoti įvykusį sukčiavimo atvejį, kad būtų galima imtis veiksmų kurie padėtų išvengti tolimesnių nuostolių.

Vis dėlto, dabar šioje klasifikacijoje galima įžvelgti problemą. Pagal pateiktą sukčiavimo aptikimo apibrėžimą teigiama, kad sukčiavimas aptinkamas po to kai jis įvyksta. Tačiau šiuo metu, pagerėjus techninėms galimybėms, sukčiavimo aptikimas galėtų būti vykdomas ir realiu laiku. Dėl šios priežasties kyla diskusijos: ar sukčiavimo aptikimo nereikėtų imti laikyti išvengimo priemone? Šiame darbe nebus mėginama atsakyti į šį klausimą, o toliau darbe laikoma, kad tiek vykdant sukčiavimo atvejų identifikavimą prieš patvirtinant transakciją realiu laiku, tiek po transakcijos patvirtinimo, toks veiksmas bus vadinamas bendrai – sukčiavimo aptikimu.

Sukčiavimas tobulėja kartu su saugumo priemonėmis. Sukčiavimas evoliucionavo iš kasdieniškos sukčių veiklos į organizuotą nusikalstamumą su sudėtingomis schemomis, naudojančiomis sudėtingus metodus, todėl sukčiavimą aptikti vis sunkiau [JKK+16]. Siekiant aptikti sukčiavimą reikia įveikti įvairius sunkumus. Pirmiausia, sukčiavimas yra retas reiškinys. Sukčiavimu pripažintos transakcijos sudaro mažiau nei 1 proc. visų apdorojamų transakcijų. Transakcijos nepastebimai užmaskuotos, o sukčiavimas gerai apgalvotas organizuotų sukčių elgesys, nėra impulsyvus ir neplanuotas, jei taip būtų sukčiavimo aptikimas būtų labai paprastas [BVV15]. Svarbu atsižvelgti, kad sukčiavimas evoliucionuoja. Sukčiai nuolat tobulina sukčiavimo mechanizmus, stengiasi išlikti aptikimo sistemų priešakyje. Dažniausiai sukčiavimas gerai organizuotas, tai reiškia, kad tuo neužsiima pavieniai sukčiai bei pasireiškia daugybe tipų ir formų.

Vienas svarbiausių šių dienų apgaulės aptikimo sistemų tikslų yra užtikrinti pakankamą efektyvumą, kad atsakas būtų pateikiamas realiu laiku [JKK+16]. Bankai yra suinteresuoti įgyvendinti paprastesnius atsiskaitymo būdus kliento atžvilgiu, kadangi klientai yra linkę išleisti daugiau lėšų, kai atsiskaitymo procesas yra paprastesnis. Pakankamai ištobulinus sukčiavimo aptikimą, būtų galima mažinti tiesiogines klientą apsunkinančias saugumo priemones [LRL10]. Papildomus saugumo mechanizmus galima naudoti, tik tais atvejais, kai sukčiavimo aptikimas įvertina transakciją kaip įtartiną [JKK+16].

1.2.1. Aptikimo metodų klasifikavimas

Sukčiavimo aptikimo metodus galima grupuoti įvairiai, tačiau žvelgiant bendriausia prasme, galime išskirti dvi pagrindines grupes:

- naudojantys taisyklių rinkinius (angl. *rule based*);
- naudojantys duomenų gavybos (angl. *data mining*) metodus.

Svarbu tinkamai atskirti šias dvi grupes, kadangi tam tikruose duomenų gavybos metoduose, pavyzdžiui, sprendimo medžiuose – taip pat naudojama savotiška taisyklių išraiška. Esminis skirtumas, kad taisyklių rinkinių grupei priskiriami metodai kuriuose taisykles kuria ir prižiūri srities ekspertai, tuo tarpu duomenų gavybos metoduose ekspertai analizuoja ir manipuliuoja duomenimis, iš kurių sudaromas taisyklių modelis.

1.2.2. Aptikimui naudojami atributai

„Atributas yra duomenų laukas atspindintis charakteristiką ar savybę duomenų objekte“ [HKP11]. Nepriklausomai nuo to, kokio tipo sukčiavimo aptikimo būdas pasirenkamas įgyvendinimui būtina turėti tinkamai paruoštus duomenis. Duomenų gavybos metodų atveju duomenys yra pagrindinė įeiga, kurios pagalba sudaromas modelis. Tačiau duomenys yra nemažiau svarbūs ir taisyklių rinkiniais paremtuose aptikimo metoduose, kadangi pagal taisykles bus vertinamos skirtingų duomenų atributų reikšmės bei jų kombinacijos.

Siekiant paruošti duomenis aptikimui būtina pasirinkti naudojamus duomenų atributus. Ruošiant modelius tinkamai neatsižvelgiama į sukčiavimo aptikimo sistemos kontekstą, tyrimuose pateikiamuose modeliuose daromos nerealistiškos prielaidos pasirenkant sukčiavimo aptikimui naudojamus atributus [ZYL09]. Pasirenkami atributai, kurie ne visada prieinami sukčiavimo aptikimo sistemoms. Nereikėtų remtis užsakymo ir pardavėjo duomenimis [SKS+08]. Daugumoje metodų transakcijų duomenis mėginama grupuoti pagal pardavėjo tipo kodą, tačiau spėlioti pirkinio tipą iš pardavėjo tipo ne visada yra teisinga, kadangi dauguma pardavėjų siūlo įvairaus pobūdžio prekes, o konkrečios apsipirkimo detalės prieinamos ne visų tipų sukčiavimo aptikimo sistemose. Tokio tipo atributų naudojimas prasmingas tik kuriant aptikimo sistemą, kuri veiks pardavėjo informacinėje sistemoje, tačiau tokio tipo sistemos neturės platesnio transakcijų rinkinio iš kitų pardavėjų, todėl nebus atvaizduojamas pilnas kortelės savininko elgsenos modelis.

Vykdant sukčiavimo aptikimą patartina remtis baziniais EMV standartu atliekamos transakcijos atributais [KMN+15]:

- kortelės identifikatorius;
- sąskaitos identifikatorius;
- geografinė transakcijos vykdymo vieta;
- transakcijos vykdymo laikas;
- transakcijos suma;
- pardavėjo identifikatorius.

Svarbu pastebėti, kad apdorojant kortelių duomenis, nelieka galimybės naudoti tinklo prieigos informacijos, kuri dažnai naudojama apdorojant internete inicijuotas transakcijas. Taip pat daroma prielaida, kad kliento asmeniniai duomenys bei užsakymo duomenys neprieinami.

1.2.2.1. Atributų agregavimas

Pastebima, kad dažnai pristatomi sukčiavimo aptikimo metodai naudojantys neapdorotus bazinius atributus. Naudojant tik bazinius atributus tiksliai nėra įvertinama vartotojo elgsena, todėl reikia naudoti papildomus, iš bazinių atributų agreguotus atributus [BAS+16]. Siekiant tikslumo reikia agreguoti praeities elgseną atspindinčius atributus, pavyzdžiui, laikas nuo praeito atsiskaitymo, praėjusio atsiskaitymo suma ir pan. Tokiu būdu agreguojant pilnesnį istorinės elgsenos modelį. Taip pat reikia agreguoti duomenis remiantis skirtingais laiko intervalais, ne tik apie praėjusį mokėjimą. Svarbu tinkamai pasirinkti intervalus, kadangi bėgant laikui asmens elgsenos šablonai keičiasi ir duomenų vertė mažėja. Taip pat reikia nepamiršti, kad papildomi atributai didina analizės sudėtingumą. Naudojant atributų agregavimą pavyko 200 proc. padidinti finansinę naudą gaunamą iš sukčiavimo aptikimo sistemos [BAS+16].

1.2.2.2. Atributų tipai

Sukčiavimo aptikimui naudojamus atributus reikia interpretuoti pagal atributą nusakantį duomenų tipą. Pagrindiniai naudojami duomenų tipai [BVV15]:

- tęstiniai intervalai;
- nominalios reikšmės (angl. *nominal*);
- išvardijimo reikšmės (angl. *ordinal*);
- dvejetainės reikšmės (angl. *binary*).

Tęstinis intervalas - skaitinė vertė, kuri priskiriama ribotam ar neribotam intervalui. Pavyzdžiui – transakcijos suma. Nominalūs duomenys, išreiškiami iš anksto apibrėžtomis kategorijomis, kurios neturi tarpusavio sąryšių rikiavimo prasme. Pavyzdžiui, lytis. Tuo tarpu, išvardijimo reikšmės apibrėžiamos kaip nominalios reikšmės, kurių tvarka turi prasmę tarpusavyje. Pavyzdžiui, mokėjimo laiką išreiškus nominaliomis reikšmėmis: rytas, diena, vakaras, naktis, turėtume tarpusavyje susietas nominalias reikšmes, kadangi žinome, kad po ryto eina diena, tada vakaras ir t.t. Dvejetainiai atributai - nominalios reikšmės su 2 galimomis vertėmis.

Dažnai aptikimo metoduose pasirenkami netinkami statistiniai metodai atributų įvertinimui, todėl atributų tipų identifikavimas yra svarbus [BVV15]. Aptikimo metu kiekvienam tipui turi būti taikomi tinkami įvertinimo metodai. Pavyzdžiui, nominaliems duomenims negalime skaičiuoti aritmetinio vidurkio ar Euklido atstumo. Dažnai pamirštama, kad laikas yra periodinis vienetas ir atliekami neprasmingi skaičiavimai [BVV15]. Pavyzdžiui, atlikus

mokėjimą po vidurnakčio ir vėlai vakare prieš vidurnaktį, gautume, kad vidutiniškai mokėjimus atliekame vidurdienį, tačiau toks įvertis neprasmingas.

Dėl šios priežasties, prieš atliekant tolimesnį duomenų apdorojimą, būtina identifikuoti atributų tipus ir kiekvienam iš atributų identifikuoti tinkamus įverčių metodus.

1.2.2.3. Atributų normalizavimas

Pasirinkus tinkamus atributus, įvertinus jų tipus bei atributų vertinimo būdus, reikia atlikti duomenų normalizavimą [BVV15]. Skirtingi atributai išreiškiami skirtingais matavimo vienetais, jų vertės kinta skirtinguose intervaluose. Dėl šios priežasties vieno atributo nedidelis pokytis gali turėti didesnę prasminę įtaką nei kito atributo didelis pasikeitimas. Kad išvengi netikslumų, duomenys turi būti normalizuojami arba skirstomi į kategorijas. Tikslios skaitinės vertės įneša tik triukšmą, o suskirsčius duomenis į kategorijas galime sekti pokyčius tarp nominalių reikšmių ir gauti vienareikšmį rezultatą [BVV15].

1.2.2.4. Atributų apibendrinimas

Apibendrinant galima teigti, kad įgyvendinant sukčiavimo aptikimą būtinas tinkamas aptikimui naudojamų duomenų atributų pasirinkimas bei apdorojimas. Patartina orientuotis į bazinius atributus prieinamus visoms aptikimo sistemoms, tokius kaip: kortelės identifikatorius, sąskaitos identifikatorius, geografinė transakcijos vykdymo vieta, transakcijos vykdymo laikas, transakcijos suma, pardavėjo identifikatorius.

Pasirinkus bazinius atributus reikia jais neapsiriboti ir identifikuoti tinkamus išvestinius atributus, kurie išreiškia istorinę kortelės savininko elgseną. Papildomi duomenų atributai neigiamai veikia skaičiavimų efektyvumą, tačiau daro teigiamą įtaką sukčiavimo aptikimo rezultatams. Taip pat, identifikavus atributus, būtina įvertinti naudojamų duomenų tipus bei galimus su jais atlikti veiksmus, normalizuoti reikšmes siekiant tinkamos atributo pokyčio įtakos rezultatams bei suskirstyti reikšmes į kategorijas siekiant sumažinti triukšmą duomenyse.

1.2.3. Aptikimo vertinimas

Siekiant analizuoti bei lyginti sukčiavimo aptikimo būdus, reikia apsibrėžti palyginimo kriterijus. Geras sukčiavimo aptikimo būdas pasižymi šiomis charakteristikomis [BVV15]:

- statistinis tikslumas;
- interpretavimo paprastumas;
- vykdymo efektyvumas;
- ekonominė kaina.

Statistinis tikslumas išreiškiamas ir matuojamas naudojant papildomas statistines metrikas, kurios vertina aptikimo metodo transakcijų įverčių atitikimą realioms transakcijų būsenoms.

Modelio interpretavimo paprastumas apibrėžia sprendimų aiškumą žmonėms. Nepriklausomai nuo naudojamo metodo, turi būti įmanoma atsekti kodėl buvo pateiktas atitinkamas įvertis.

Efektyvumas apibrėžiamas kaip laikas ir resursų kiekis reikalingas įvertinti transakciją.

Ekonominė kaina apibrėžia lėšas reikalingas sukurti sukčiavimo aptikimo sistemą bei ją palaikyti. Finansinės institucijos siekia pelno, todėl kaina dažnai yra vienas svarbiausių faktorių, kadangi finansiškai neapsimoka išleisti sukčiavimo aptikimui daugiau, nei aptikimo pagalba bus atgauta lėšų.

1.2.3.1. Statistinio vertinimo kriterijai

Finansinėse sferose ypač svarbu rezultatų matavimas vertinant sukčiavimo aptikimo būdus, kadangi ir mažas rezultatų pagerėjimas gali atnešti didesnę naudą [WB15]. Teigiama, kad sukčiavimo aptikimo tikslas – maksimalus teisingų spėjimų kiekis ir priimtinas neteisingų spėjimų kiekis. Norint užtikrinti prasmingus palyginimus, reikia turėti vieningas metrikas rezultatų vertinimui.

Pagrindinės statistinės metrikos leidžiančios įvertinti sukčiavimo aptikimo metodą [WB15]:

- tikslumas (angl. *accuracy*);
- precizija (angl. *precision*);
- jautrumas (angl. *sensitivity*);
- specifiškumas (angl. *specifity*);
- netikrų aliarmų santykis (angl. *false positive rate*).

Tikslumas apibrėžiamas, kaip santykis tarp teisingai identifikuotų ir visų analizuotų transakcijų kiekių. Precizija, tai santykis tarp teisingai identifikuotų kaip sukčiavimas ir visų identifikuotų kaip sukčiavimas kiekių. Jautrumas, tai santykis tarp teisingai identifikuotų sukčiavimo atvejų ir visų sukčiavimo atvejų kiekių. Specifiškumas, santykis tarp neteisingai identifikuotų transakcijų ir visų nesukčiavimo atvejų kiekių. Tuo tarpu netikrų aliarmų santykis, tai santykis tarp gerų atvejų identifikuotų kaip sukčiavimas ir visų gerų atvejų kiekių.

1.2.3.2. Finansine nauda paremtas vertinimas

Sukčiavimo aptikimo vertinimą taip pat galima tiesiogiai susieti su kaštų valdymu atliekant statistinį vertinimą. Sukčiavimo aptikimo metodus galima vertinti ne tik pagal būdo tikslumą, tačiau ir pagal finansinę naudą. Aptikimo metodo rezultatai skirstomi į 4 grupes [BAS+16]:

- teisingas teigiamas (angl. *true positive*) – transakcija buvo identifikuota kaip sukčiavimas, įvertis buvo teisingas;

- neteisingas teigiamas (angl. *false positive*) – transakcija buvo identifikuota kaip sukčiavimas, įvertis buvo neteisingas;
- teisingas neigiamas (angl. *true negative*) - transakcija buvo identifikuota kaip nesukčiavimas įvertis buvo teisingas;
- neteisingas neigiamas (angl. *false negative*) - transakcija buvo identifikuota kaip nesukčiavimas, įvertis buvo neteisingas.

Teisingo teigiamo ir neteisingo teigiamo, atveju siūloma kaštus vertinti kaip nekintančius administracinius kaštus. Neteisingo neigiamo atveju laikoma, kad praradimas lygus transakcijos vertei. Tuo tarpu, teisingo neigiamo atveju, praradimo nėra, kadangi geroms transakcijoms nereikalingas papildomas administravimas dėl sukčiavimo apdorojimo.

Realybėje praradimai yra didesni, kadangi kyla netiesioginių nuostolių dėl klientų nepasitenkinimo, tačiau juos sunku objektyviai įvertinti [BAS+16]. Vis dėlto, naudojantis nurodytomis vertėmis, vertindami aptikimo būdą, galime įvertinti tikėtiną įtaką finansiniams rodikliams. Šis vertinimo metodas skatina ignoruoti mažesnės vertės transakcijas, tačiau netinkamai identifikavus daug mažos vertės transakcijų, praradimai taip pat bus dideli. Norint to išvengti galima keisti aptikimo įverčio įtaką galutiniam rezultatui naudojant papildomas įverčių korekcijas priskiriant svarbą nusakantį svorį.

1.2.3.3. Sukčiavimo aptikimo vertinimo apibendrinimas

Apibendrinant galima teigti, kad vertinant ir siekiant palyginti sukčiavimo aptikimą su kitais aptikimo būdais, svarbu naudoti vieningas vertinimo metrikas. Tuo tarpu kuriant konkrečią realizaciją skirtą naudojimui gamybinėje aplinkoje svarbu vertinti ir galutinį rezultatą – finansinę naudą, kadangi tik remiantis aptikimo suteikiamais finansinės naudos įverčiais ir sukčiavimo aptikimo įgyvendinimo kainos įverčiais, galima įvertinti, kuris sukčiavimo aptikimo metodas yra tinkamas konkrečiu atveju. Vis dėlto, nepaisant svarbos taikant gamybinėje aplinkoje, finansinis įvertinimas nėra paplitęs akademinėje visuomenėje, kadangi yra tiesiogiai priklausomas nuo ekonominio konteksto. Todėl kuriant teorinį modelį pakanka apsiriboti tiesioginių rezultatų ir efektyvumo vertinimu.

1.2.4. Taisyklių rinkiniai

Taisyklių rinkiniai apibrėžiami, kaip loginių sakinių rinkinys naudojamas siekiant identifikuoti transakcijas atitinkančias sukčiavimui būdingas transakcijų savybes [Mon04]. Taisyklės rinkinyje remiasi aptikimui naudojamų atributų palyginimu su tikėtinais sukčiavimo profiliais.

Taisyklių rinkiniuose reikėtų atsižvelgti į pardavėjo tipą, kadangi tam tikrų grupių pirkiniai yra susiję su aukštesne sukčiavimo rizika. Atsiskaitymo dažnumas ir transakcijos profilio

pasikeitimo dažnis yra vieni svarbiausių atributų į kuriuos reikia atkreipti dėmesį kuriant šio tipo sistemas [Mon04].

Vertinant atsiskaitymo dažnumą įprasta manyti, kad kuo didesnis naudojimo dažnumas, tuo didesnė rizika susijusi su šia transakcija. Tuo tarpu, pasikeitimo dažnis išreiškia, kaip dažnai keičiasi kortelės naudotojo elgsena. Pokyčius galime vertinti įvairiais būdais: vertindami vietos pasikeitimą, transakcijos vertės pasikeitimą. Taip pat tam tikros geografinės vietos yra rizikingesnės ir jose sukčiavimas vyksta dažniau, todėl reikia realizuoti geografine vieta paremtas taisykles [Mon04]. Siekiant įgyvendinti aptikime naudojamas taisykles nereikėtų naudoti taisyklių paremtų fiksuotomis ribinėmis reikšmėmis, manoma, kad jos neefektyvios. Sukčiai pakankamai greitai identifikuoja šias ribines reikšmes ir pateikinėja taisykles atitinkančias transakcijas [Mon04].

Taisykles rinkiniuose galima apjungti ir interpretuoti įvairiais būdais. Paprasčiausias iš jų, tai taisyklių sąrašas. Jame taisyklės pateikiamos sąrašo pavidalu, taisyklių tikrinimas vykdomas paeiliui, jei netenkinama bent viena taisyklė transakcija laikoma įtariama sukčiavimu. Šis būdas lengvai įgyvendinamas, tačiau nėra tikslus ir kelia aukštą netikrų aliarmų santykį. Geresnių rezultatų padedanti pasiekti taisyklių sąrašo alternatyva yra svorinis sąrašas. Šiame sąraše kiekviena taisyklė turi savo svorį į kurį atsižvelgiama apjungiant taisyklių rezultatus į bendrą skaitinį įvertį, kurį labiau įtakoja turi svarbesnės taisyklės. Dar sudėtingesni taisyklių rinkiniai sudaromi medžio pavidalu. Medžio pavidalo rinkiniuose skirtingais atvejais atliekami skirtingi patikrinimai, kadangi medžiu pereinama tik viena šaka nuo šaknies, tai leidžia skirtingo pobūdžio sukčiavimą vertinti skirtingai.

Taisyklių rinkinius naudojanti sistema, tai klasikinis sukčiavimo aptikimo sprendimas. Tačiau tokio tipo sistemos nelanksčios, sukčiai prisitaiko prie naudojamų sukčiavimo aptikimo taisyklių [Mon04]. Todėl nepaisant paprasto įgyvendinimo jų palaikymas tampa brangus, nuolat reikalingi ekspertai atliekantys taisyklių priežiūrą ir atnaujinimą. Taip pat manoma, kad duomenų gavyba paremti metodai dažnai būna tikslesni, jiems nereikia nurodyti kokius atvejus aptikti, todėl sistema aptinka ir tuos atvejus, apie kuriuos ekspertas nepagalvojo. „Sukčiai nuolat keičia savo strategijas, kad išvengtų aptikimo, dėl to tradicinės taisyklėmis paremtos sistemos yra neadekvačios“ [BAS+16]. Todėl galime teigti, kad duomenų gavyba paremti būdai laikomi pranašesniais siekiant aptikti sukčiavimą.

1.2.5. Duomenų gavybos metodai

„Duomenų gavyba – tai įdomių šablonų ir žinių iš didelių duomenų kiekių atradimo procesas“ [HKP11]. Duomenų rinkiniai sudaryti iš duomenų objektų. Duomenų objektas, tai esybė, kurios būseną apibūdina atributai. Apsimokančių sistemų kontekste atributai dažniausiai vadinami savybėmis (angl. *feature*). Norint darbe išlaikyti vientisumą, nepriklausomai nuo

sukčiavimo aptikimo sistemos tipo toliau duomenis apibūdinančius požymius vadinsime atributais.

Yra daugybė duomenų gavybos būdų, visus šiuos būdus pagal būdo siekiamą tikslą galima suskaidyti į dvi pagrindinės grupės:

- nuspėjantis[4] (angl. *predictive*);
- apibrėžiantis[5] (angl. *descriptive*).

Nuspėjimas naudoja pateiktą suklasifikuotų duomenų rinkinį ir siekia įvertinti tikėtinas atributų vertes, pagal kurias galima klasifikuoti naujus duomenų įrašus [ZK17]. Atitinkamai sukčiavimo aptikime, tokio tipo nuspėjimas naudojamas mėginant suklasifikuoti transakcijas. Nuspėjimą įgyvendinančios sukčiavimo aptikimo sistemos geriau identifikuoja sukčiavimą pridengiamą normalaus elgsenos šablonais [BVV15]. Šių metodų problema, kad jų naudojimui reikalingas pradinis suklasifikuotas duomenų rinkinys.

Apibrėžiančioji duomenų analitika nereikalauja suklasifikuoto duomenų rinkinio. Jos tikslas – charakterizuoti turimą duomenų rinkinį pagal atributus. Dažniausiai naudojami apibrėžiančiosios analitikos būdai: klasterizavimas, anomalijų aptikimas. Apibrėžiančiąją analitiką įgyvendinančios sistemos turi privalumų, kadangi pasikeitus elgsenai jos automatiškai aptinka ir visiškai naujus sukčiavimo būdus, nereikia sudarinėti naujų taisyklių ar laukti kol bus suklasifikuotas atnaujintas duomenų rinkinys, kurį galima naudoti nuspėjančio metodo paruošimui [BVV15].

1.2.5.1. Duomenų gavybos metodų klasifikavimas

Yra daug duomenų gavybos metodų. Toliau pristatomi pagrindiniai metodų tipai:

- klasifikavimas;
- klasterizavimas;
- anomalijų aptikimas;
- socialinių tinklų analizė.

1.2.5.1.1. Klasifikavimas

Duomenų klasifikavimas siekia suskirstyti naujai gaunamus duomenis į kategorijas remdamasis istoriniais duomenimis. Klasifikavimas, priešingai nuo klasterizavimo, priklauso nuspėjančiųjų metodų grupei, todėl norint naudoti klasifikavimą, duomenų klasės turi būti iš anksto žinomos ir apibrėžtos. Taip pat reikalingas pradinis duomenų rinkinys su jau

[4] Nuspėjantis – mašininio apsimokymo kontekste įprasta vadinti prižiūrimu mokymųsi (angl. *supervised learning*)

[5] Apibrėžiantis – mašininio apsimokymo kontekste įprasta vadinti neprižiūrimu mokymųsi (angl. *unsupervised learning*)

suklasifikuotais duomenimis. Šis duomenų rinkinys naudojamas paruošti metodą duomenų klasifikavimui.

1.2.5.1.2. Klasterizavimas

„Klasteris yra duomenų objektų rinkinys, toks, kad objektai klasterio viduje yra panašūs tarpusavyje ir skirtingi nuo objektų už klasterio ribų“ [HKP11]. Duomenų klasterizavimas, kaip ir klasifikavimas, siekia suskirstyti naujai gaunamus duomenis remiantis istoriniais duomenimis. Klasterizavimas priklauso apibrėžiančiųjų metodų grupei, todėl ruošiant metodą, naujų duomenų skirstymui į klasterius, nereikalingas iš anksto sužymėtas duomenų rinkinys. Klasterizavimo metodai automatiškai suskirsto pradinį duomenų rinkinį pagal principinius skirtumus į skirtingus klasterius, į kuriuos bus skirstomi naujai gaunami duomenys [ZK17].

Elektroninių mokėjimų sukčiavimas yra retas reiškinys, todėl turimi duomenų rinkiniai gali būti netinkami klasifikavimo metodo paruošimui dėl sukčiavimo transakcijų trūkumo. Dėl šios priežasties klasterizavimas gali pasirodyti puikiai tinkantis sukčiavimo aptikimui, kadangi nereikalingas iš anksto sužymėtas duomenų rinkinys. Nepaisant gebėjimo apdoroti nebalansuotą duomenų rinkinį, klasterizavimas turi ir trūkumų. Ne visi algoritmai vienodai gerai klasterizuoja daugelio dimensijų duomenis, o dauguma gerai veikia tik su 2 - jų, 3 - jų dimensijų duomenimis [HKP11]. Tačiau siekiant aptikti sukčiavimą transakcijose, vertinami transakcijų baziniai ir išvestiniai atributai, todėl duomenys bus daugiau nei trijų dimensijų. Taip pat klasterizavimo algoritmai gerai veikia su skaitinių intervalų reikšmėmis, tačiau prastai su dvejetainėmis ar nominaliomis reikšmėmis, kadangi atliekant klasterizavimą dažnai naudojami erdvės skaidymo metodai, dažniausiai vertinamas atstumas tarp klasterių ar klasterio tankis. Vertinant atstumą tarp klasterių iškyla papildomų problemų, duomenų klasteriai dažniausiai būna netaisyklingų formų, todėl nesutariama tarp kurių elementų reikia matuoti klasterių atstumą [BVV15]. Siūloma matuoti tarp panašiausių klasterio elementų, ar priešingai, tarp labiausiai besiskiriančių elementų. Vis dėlto, dažniausiai naudojami metodai atstumą vertinantys tarp klasterių centrų.

1.2.5.1.3. Anomalijų aptikimas

„Anomalija yra duomenų objektas, kuris žymiai skiriasi nuo kitų objektu, lyg jis būtų sugeneruotas kito mechanizmo“ [HKP11]. Anomalijų aptikimas, kaip ir duomenų klasterizavimas, priskiriamas apibrėžiančiųjų metodų grupei, todėl metodo paruošimui naudojamas nesuklasifikuotas duomenų rinkinys. Anomalijų aptikimas nuo klasterizavimo skiriasi tuo, kad klasterizavimas ieško daugumai būdingų principinių šablonų ir pagal juos grupuoja duomenis į klasterius, o anomalijos dažnai palaikomos triukšmu duomenyse. Tuo tarpu anomalijų aptikimas mėgina identifikuoti normalų elgsenos šabloną ir rasti tuos atvejus kurie išsiskiria iš šių šablonų.

Anomalijų aptikimui naudojami įvairūs būdai, praktikoje taikomi Euklido atstumo įverčiai tarp vektorių, skaičiuojamas statistinis Z-įvertis, kuris nurodo per kiek standartinių nuokrypių duomenys nutolo nuo vidurkio. Pasirinkus anomalijos identifikavimo būdą, taip pat reikia pasirinkti duomenis, kurių atžvilgių bus atliekamas vertinimas. Paprasčiausias būdas – lyginti su visu duomenų rinkiniu, tačiau tokie palyginimai sudėtingi skaičiavimo prasme. Todėl, prieš lyginant objektą su normalios elgsenos objektų grupe, duomenų rinkinį galima grupuoti ir lyginti tik su tam tikromis pasirinktomis grupėmis. Kaip alternatyvų sprendimą galima naudoti skaidymą pagal laiką – išsiskirti duomenų rinkinį, kuris atitinka numatytus ribojimus laike ir atlikti vertinimą jo atžvilgiu, o einant laikui, kartu keisti ir šį duomenų rinkinį, taip užtikrinant, kad anomalijų aptikimas remiasi naujausiais elgsenos šablonais [BVV15].

Anomalijų identifikavimą apsunkina ir tai, kad žmonės tobulai neatitinka elgsenos šablonų ir kiekvienas kortelės turėtojas generuoja tam tikrą triukšmo kiekį [ZK17]. Triukšmas duomenų prasme, panašus į anomalijas, tačiau nėra įdomus aptinkant sukčiavimą.

1.2.5.1.4. Socialinių tinklų analizė

Socialiniai tinklai – socialinė struktūra sudaryta iš individų bei organizacijų atvaizduojamų grafo mazgais [ZK17]. Sąryšiai atvaizduojami grafo keliais. Socialiniame tinkle gali būti atvaizduojami įvairių tipų sąryšiai: ekonominiai, geografiniai ar kiti, priklausomai nuo socialinių tinklų analizės panaudojimo srities. Tuo tarpu kredito kortelių sukčiavimo aptikimo kontekste, informacija prieinama tik apie ekonominius sąryšius.

Siekiant aptikti sukčiavimą tinkluose vykdoma [ZK17]:

- sąryšių analizė;
- pasikartojimų analizė.

Sąryšius vertinanti socialinių tinklų analizė siekia įvertinti sąryšio stiprumą tinkle tarp mazgų [ZK17]. Mokėjimų atveju galima ieškoti sąryšių tarp sukčių, pavyzdžiui laikyti, kad mokėjimas atliekamas į sąskaitą, į kurią atliekama daug sukčiavimu pripažintų transakcijų, yra rizikingesnis. Realybėje sukčiavimo schemos sudėtingesnės, todėl vertinti sąryšių tarp dviejų mazgų nepakanka, dėl šios priežasties ir naudojamas tinklas, leidžiantis įvertinti sąryšio stiprumą tarp tiesioginio sąryšio neturinčių mazgų.

Pasikartojimų aptikimu paremta socialinių tinklų analizė ieško pasikartojančių sukčiavimo šablonų socialiniame tinkle [ZK17]. Prieš atliekant analizę, identifikavus sukčių bei nukentėjusįjį, išsiaiškinama sukčiavimo schema. Sužinome kokiu grafo keliu lėšos pasiekdavo sukčiaus mazgą. Turėdami šias žinias apie sukčiavimo atvejį visame grafe galime ieškoti grafo poaibių, kurie būtų panašūs į identifikuotą schemą. Akivaizdu, kad šis metodas nepadės apsisaugoti nuo paprastų pavienio neorganizuoto sukčiavimo atvejų, kadangi jie būtų atvaizduojami trivialiais grafo keliais tarp viršūnių, kurie yra dažni pasitaikantys visame grafe.

Galima teigti, kad metodas orientuotas į organizuoto sukčiavimo aptikimą ir labiau tinkamas naudoti kaip papildoma sukčiavimo aptikimo priemonė.

1.2.5.2. Duomenų gavybos procesas

Duomenų gavyba paremta duomenų analize, todėl metodo rezultatų gerumas tiesiogiai priklauso nuo duomenų paruošimo. „Žemos kokybės duomenys ves prie žemos kokybės duomenų gavybos rezultato“ [HKP11]. Dėl šios priežasties būtina atsakingai atlikti visas modelio paruošimo fazes.

Duomenų gavyba susideda iš kelių fazių. Norint paruošti sukčiavimo aptikimui tinkamą metodas, reikia identifikuoti duomenų šaltinius, juos apjungti, identifikuoti modelių paruošimui tinkamus duomenų rinkinius [BVV15].

Identifikuoti duomenų šaltinius sukčiavimo aptikimo kredito kortelių transakcijų atveju nesudėtinga, kadangi prieinamas tik nedidelis duomenų rinkinys - transakcijos duomenys. Papildomai, priklausomai nuo aptikimo įgyvendintojo, gali būti prieinami sutarties duomenys iš banko, o realizuojant aptikimą pardavėjo pusėje – užsakymo duomenys.

Duomenų apjungimo fazėje pirmiausia siekiama suprasti duomenis bei principus, kuriuos duomenys atvaizduoja. Vykdomas išankstinis apdorojimas, skaičiuojami išvestiniai atributai, apdorojamas triukšmas, duomenys normalizuojami.

Duomenų rinkinys modelio paruošimui parenkamas siekiant tikslesnio modelio ir efektyvumo. Pasirenkant rinkinį reikia įvertinti, kad duomenys sensta, atsižvelgti į periodiškumą, įvertinti kokiam laikotarpiui ruošiamas metodas [BVV15]. Todėl siekiama pasirinkti tinkamą duomenų rinkinį, kad jis atspindėtų aktualius elgsenos modelius. Parenkant naudojamą duomenų rinkinį galima duomenų segmentacija. Segmentavimas naudojamas siekiant didesnio tikslumo sukčiavimo aptikime. Pateikiamuose metoduose naudojami duomenys nesuskirstyti į segmentus arba suskirstyti į kortelės/mokėtojo lygio segmentus. Dažniausiai ruošiami asmenine elgsena paremti aptikimo profiliai, tačiau bendru elgsenos profiliu paremtas aptikimas yra tikslesnis [AS12]. Tai galima pagrįsti asmeninio profilio duomenų rinkinio dydžiu, kadangi Europos centrinio banko pateikiamais statistiniais duomenimis, Lietuvos pilietis vidutiniškai per metus atlieka 71 atsiskaitymą kreditine ar debetine kortele [ESD16]. Todėl galima teigti, kad kliento elgsena paremti metodai, dėl naudojamo mažo duomenų rinkinio, yra jautrūs triukšmui ir besikeičiančiai kliento elgsenai, todėl neužtikrina tikslesnio aptikimo.

1.2.5.3. Duomenų gavybos apibendrinimas

Apibendrinant galima teigti, duomenų gavyba įgaliną įvairius sukčiavimo aptikimo būdus, tačiau ne visi šie metodai yra tinkami elektroninių mokėjimų sukčiavimo aptikimui. Anomalijų aptikimas ir klasterizavimas gali atrodyti kaip tinkami metodai sukčiavimo aptikimui. Jie ieško duomenų, kurie neatitinka šablonų, ar mėgina juos sugrupuoti. Aptariamieji metodai taip pat

sprendžia elgsenos pasikeitimo problemą ir nereikalauja duomenų apsimokymui. Tačiau šie metodai yra netinkami, nes pateikia prastus rezultatus su daugelio dimensijų duomenimis bei į normalius elgsenos šablonus įsiliejančiomis transakcijomis. Tuo tarpu socialinių tinklų analizė, nors ir gali suteikti naudingos informacijos, naudingesnis kaip antrinis sukčiavimo aptikimo būdas, kadangi yra netinkamas aptikti neorganizuotą sukčiavimą, kuris nepasižymi sudėtingomis schemomis. Todėl galima teigti, kad sukčiavimo aptikimui verta naudoti klasifikavimu paremtus metodus. Jie gali pasiekti didesnį tikslumą ir geriau aptinka sukčiavimą atitinkantį normalius elgsenos šablonus.

Taip pat svarbu paminėti, kad nepriklausomai nuo aptikimo metodo realizacijos, rezultatas tiesiogiai priklauso nuo duomenų rinkinio naudojamo metodo paruošimui. Todėl yra būtinas tinkamas duomenų rinkinio paruošimas. Tuo tarpu siekiant didesnio tikslumo rinkinio segmentacijos pagalba, svarbu pasirinkti tinkamus rinkinio segmentavimo kriterijus, kadangi per didelė segmentacija veda prie mažo duomenų kiekio segmentuose ir daro neigiamą įtaką rezultatui.

1.2.6. Sukčiavimo aptikimo apibendrinimas

Sukčiavimo aptikimas tampa vis svarbesnis užtikrinant elektroninių mokėjimų saugumą. Anksčiau aptikimas buvo vykdomas kaip periodinis procesas po transakcijos patvirtinimo, dėl technologinės pažangos reikalavimai sukčiavimo aptikimui performuluojami tikintis aptikimo realiu laiku. Realaus laiko aptikimas leistų sumažinti finansinius praradimus ir tiesiogiai klientą veikiančias saugumo priemones.

Yra įvairių būdų realizuoti sukčiavimo aptikimą. Vis dėlto, klasikiniu sprendimu laikomi taisyklių rinkiniai, palyginus su duomenų gavyba, nėra tokie tikslūs, jiems trūksta lankstumo. Galima teigti, kad vienas geriausiai tinkančių sukčiavimo aptikimui būdų - duomenų gavyba paremta klasifikavimu. Toks metodas leis apdoroti daugelio dimensijų duomenis, aptikti sukčiavimą įgyvendintą remiantis normalios elgsenos šablonais. Nepriklausomai nuo to, koks sukčiavimo aptikimo realizacijos bus pasirinktas būtinas išankstinis duomenų apdorojimas. Siekiant gerų rezultatų, reikia identifikuoti aptikimui tinkamus atributus, pasirinkti agreguojamus sudėtinius atributus, atlikti duomenų normalizavimą, duomenų segmentavimą ir pradinio duomenų rinkinio parinkimą.

Taip pat galime teigti, kad vertinant sukčiavimo aptikimo algoritmą reikia naudotis vieningais vertinimo kriterijais, o kuriant realizaciją realiam panaudojimui - įvertinti ne tik statistinį tikslumą, tačiau ir finansinę metodo teikiamą naudą.

1.3. Sukčiavimo aptikimo metodų realizacijos

Toliau pristatomos konkrečios kredito kortelių sukčiavimo aptikimo metodų realizacijos, kurios pateikiamos akademiniuose šaltiniuose ir naudoja skirtingus duomenų gavybos metodus.

1.3.1. Sukčiavimo aptikimas paremtas paslėptaisiais Markovo modeliais

Yra pasiūlyta daug įvairių būdų naudojančių prižiūrimus (angl. *supervised*) apsimokančius algoritmus, tačiau tokios sistemos nėra lanksčios naujų sukčiavimo būdų atžvilgiu, o kiti siūlomi aptikimo metodai pateikia daug netikrų aliarmų [SKS+08]. Sukčiavimo aptikimo sistemos mato labai ribotą atributų kiekį. Šioms sistemoms nėra prieinama užsakymo informacija ir jo turinys. Šis aptikimo metodas vertina tik transakcijos sumą, bet atsižvelgiančią į tai, kad kiekvienas asmuo turi jam būdingus elgsenos šablonus, o atitinkamai ir lėšų leidimo šablonus. Metodas realizuojamas naudojant paslėptuosius Markovo modelius (angl. *hidden Markov models*).

1.3.1.1. Metodo realizacija

Metodo veikimas paremtas tikimybiniais įverčiais. Modelis sudaromas iš būsenų ir stebimų būsenų. Modelyje apsibrėžiama tikimybė, kad su tam tikromis stebėjimų būsenų reikšmėmis bus pereita į tam tikrą kitą būseną. Metodas leidžia įvertinti tikimybę, kad galėjo įvykti tam tikra veiksmų seka. Dėl šios priežasties tinkamai pasirinkus modelio būsenas, galima įvertinti tikimybę, kad buvo atlikti mokėjimai su žinomais atributais. Turėdami tikimybinį įvertį galime juo manipuliuoti ir vertinti transakcijos statusą.

Apsibrėžus modelio būsenas ir stebimas būsenas, modelis paruošiamas naudojant pradinį duomenų rinkinį. Apmokymo algoritmas, pagal būsenų pasikartojimus duomenų rinkinyje, apskaičiuoja tikimybes perėjimams tarp būsenų, todėl paslėptųjų Markovo modelių metodas nereikalauja iš anksto suklasifikuotų duomenų. Transakcijų sekos įvykio tikimybė apskaičiuojama remiantis būsenų pasikeitimų tikimybėmis.

Svarbu pabrėžti, kad gavus apmokėjimo duomenis vertinama ne pavienio įvykio tikimybė, o tikimybė, kad įvyko n paskutinių įvykių seka. Modelis iš esmės vertina, ar tikėtinas apsipirkimo elgsenos būsenų pasikeitimas, atsižvelgus į apsipirkimo sumos pokyčius veiksmų sekoje. Žvelgiant konkrečiai, realizacija vertina, kokia tikimybė mažai išleidžiančiam klientui įsigyti brangų pirkinį, ar net tapti daug išleidžiančiu klientu ir atvirkščiai.

1.3.1.2. Metodo apibendrinimas

Pateikiamas metodas - paprastas ir lengvai įgyvendinamas. Nereikalingas iš anksto suklasifikuotų duomenų rinkinys. Taip pat modelis vertina ne pavienio mokėjimo duomenis, o mokėjimų sekos duomenis. Siekiant aptikti sukčiavimą kitose mokėjimų srityse tai naudinga savybė, kadangi yra sunkiau įvertinti elgseną iš vienos transakcijos duomenų. Tokiu atveju žymiai lengviau pastebėti kelias įtartinas transakcijas atliktas paeiliui. Vis dėlto, buvo identifikuota, kad atliekant bekontakčius mokėjimus, labiausiai tikėtinas vienos transakcijos duomenų pasisavinimas iš vieno kliento, todėl šis aptikimo metodas yra netinkamas.

1.3.2. Anomalijų aptikimas pagal kryptinį duomenų vektorių

Sukčiavimas mokėjimuose sudaro tik nedidelę visų mokėjimų dalį, todėl metodas naudojantis anomalijų aptikimą yra tinkamas sukčiavimo aptikimui [LYW13]. Jam nereikalingas iš anksto suklasifikuotas duomenų rinkinys, išsprendžiama nesubalansuotų duomenų analizės problema.

Anomalijų aptikimui įprastai naudojami atstumo ar duomenų tankio įverčiais paremti metodai. Šie metodai apdoroja daugelio dimensijų duomenis ir yra sudėtingi skaičiavimo atžvilgiu, todėl šie sprendimai tinkami tik paketiniam apdorojimui (angl. *batch processing*) ir netenkina šiuolaikinių realaus laiko reikalavimų [LYW13]. Sukčiavimo aptikimo metodas veikiantis anomalijų aptikimo principu, kuris principinės komponentų analizės pagalba sumažina duomenų rinkinio dimensijų kiekį, o anomalijai identifikuoti naudoja kampą tarp duomenų rinkinio kryptinių vektorių, sprendžia šiuos trūkumus [LYW13].

1.3.2.1. Metodo realizacija

Metodui nereikalingas suklasifikuotas duomenų rinkinys. Krypties vektoriais paremtas metodas remiasi idėja, kad anomalija duomenų rinkinyje žymiai iškreips duomenų rinkinio kryptinį vektorių.

Įgyvendinant aptikimą pirmiausia duomenyse įvertinamas duomenų rinkinys ir apskaičiuojamas kryptinis vektorius. Kaip kryptinis vektorius duomenų rinkinyje pasirenkamas tikrinis duomenų rinkinio vektorius. Vektorių palyginimui galimi keli būdai.

Pakopinio mažininimo duomenų rinkinio atnaujinimo būdo atveju duomenų rinkinys apdorojamas iš anksto, visada saugoma tikrinio vektoriaus išraiška. Norint įvertinti konkrečią transakciją, mokėjimo įrašas išimamas iš duomenų rinkinio ir skaičiuojamas duomenų rinkinio kryptinis vektorius be šio įrašo.

Pakopinis metodas vykdomas priešingai. Modelyje taip pat saugomas iš anksto įvertintas duomenų rinkinys ir jo tikrinis vektorius. Šiuo atveju norint įvertinti ar transakcija yra anomalija, prie esamo duomenų rinkinio pridedamas naujasis įrašas, po to įvertinamas naujojo rinkinio tikrinis vektorius.

Siekiant pakopinio mažinimo būdu įvertinti mokėjimą, jis jau turi būti įtrauktas į duomenų rinkinį su paskaičiuotu kryptiniu vektoriumi. Dėl šios priežasties pakopinio mažinimo būdas yra per daug sudėtingas laiko atžvilgiu, kai siekiama realaus laiko atsako [LYW13]. Pastebima, kad abiem atvejais egzistuoja problema dėl duomenų rinkinių dydžio. Didelės apimties duomenų rinkiniuose vienas anomalijos įrašas darys labai mažą įtaką viso duomenų rinkinio vektoriaus krypties pasikeitimui.

Dėl šių trūkumų galima naudoti pakopinį metodą su pertekliniu duomenų pavyzdžiu (angl. *oversampling*). Prieš skaičiuojant kryptinio vektoriaus pokytį, pridėti ne vieną vertinamo įrašo

egzempliorių, o didesnį jų kiekį. Kiekis turėtų būti įvertinamas pagal duomenų rinkinio dydį. Taip sustiprinamas anomalijos poveikis vektoriaus krypties pokyčiui, nepakenkiant pradiniam duomenų rinkiniui. Naudojant apytikslio kryptinio vektoriaus skaičiavimo algoritmą, transakcijos įvertinimo sudėtingumas skaičiavimų atžvilgiu vertinamas $O(p)$, čia p – duomenų rinkinio dimensijų kiekis.

Siekiant efektyvesnio veikimo naudojamas ne kampo tarp vektorių pokytis, o kosinusų panašumo (angl. *cosine similarity*) įvertis, kadangi tarp dviejų vektorių nesunkiai galima pasiskaičiuoti kampo kosinusą žinant jų koordinates erdvėje [LYW13]. Naudojant kosinusų panašumą taip pat gaunamas normalizuotas panašumo įvertis, kadangi kampo kosinusas kinta intervale [0; 1].

Šis metodas neaptiks anomalijos, jei ji yra toli nuo normalių duomenų grupės, tačiau išlaiko tinkamą vektoriaus kryptį, tačiau tokias anomalijas nesunku identifikuoti kitais metodais [LYW13]. Tokia transakcija turėtų turėti anomalias visų atributų reikšmes, todėl šio atvejo apdorojimui galima pasitelkti kitus metodus arba papildyti kampu paremtą metodą, kad būtų atsižvelgiama ne tik į vektoriaus krypties pokytį, tačiau ir į vektoriaus ilgio pokytį.

1.3.2.2. Metodo apibendrinimas

Apibendrinant galima teigti, kad metodas išsprendžia dalį anomalijų aptikimo metodų problemų kylančių dėl duomenų balansuotumo. Taip pat teigiama, kad metodas yra tinkamas apdoroti daugelio dimensijų duomenų rinkinius bei yra pakankamai efektyvus realaus laiko transakcijų apdorojimui.

1.3.3. Elgsenos modelis paremtas savaiminiu susiejimu

Kasdien yra apdorojama milijonai transakcijų, todėl yra apsunkinamas sukčiavimo aptikimo uždavinys. Dėl šios priežasties bendro modelio atveju duomenų apdorojimas sudėtingas ir reikalaujantis daug resursų.

Nelogiška transakcijos vertinimui naudoti kitų asmenų transakcijų duomenis [ZYL09]. Siūloma atsižvelgti į kiekvienos unikalios kortelės elgsenos modelį. Taip pat pristatant sukčiavimo aptikimo metodus dažnai naudojami atributai, kurie ne visada prieinami sukčiavimo aptikimo sistemoms. Aptariama sukčiavimo aptikimo metodo realizacija sprendžia šias problemas. Įgyvendinimas naudoja savaiminio susiejimo metodą (angl. *self organising mapping*). Šis algoritmas veikia anomalijų aptikimo principu, todėl jam nereikalingas iš anksto suklasifikuotas duomenų rinkinys.

Šis metodas orientuotas į sukčiavimo aptikimą bankomatuose inicijuojamose transakcijose. Metodas vadinamas elgsena paremtu, nes aptikimui naudojami tik transakcijos atributai ir jų istoriniai duomenys [ZYL09].

1.3.3.1. Metodo realizacija

Naudojami baziniai atributai yra: transakcijos data, transakcijos suma ir transakcijos vieta. Norint įgalinti tikslesnį aptikimą iš šių bazinių atributų agreguojami išvestiniai atributai. Papildomai agreguojamas laikas praėjęs nuo paskutinės transakcijos, bendras transakcijų dažnis ir t.t. Prieš naudojant duomenis aptikimui jie normalizuojami į reikšmes intervale [0;1].

Svarbu paminėti, kad metodo paruošimas vykdomas tik gavus transakcijos patvirtinimo prašymą. Prieš patvirtinant transakciją gaunami naujos transakcijos duomenys bei istorinių transakcijų duomenys. Sugeneruojami išvestiniai atributai, visi atributai apdorojami pagal apibrėžtas transformacijas. Apdorojus duomenis yra apmokomas aptikimo modelis, jis panaudojamas įvertinti transakciją. Kadangi apdorojimui parenkami tik vieno asmens duomenys, duomenų rinkiniai labai maži, todėl net ir vykdant šias operacijas prieš kiekvieną įvertinimą, modelis yra pakankamai efektyvus, kad būtų tinkamas realaus laiko panaudos atvejams [ZYL09].

1.3.3.2. Metodo apibendrinimas

Galima teigti, kad metodas tinkamai atsižvelgia į duomenų paruošimo reikalavimus. Siekiant didesnio tikslumo skaičiuojami išvestiniai atributai, visi duomenys normalizuojami. Tačiau pasirinkti duomenų segmentai metodo paruošimui labai maži, sudaryti iš asmens transakcijų istorijos. Tai gali daryti neigiamą įtaką tikslumui, kadangi vieno asmens duomenų imtis yra jautri triukšmui bei elgsenos pokyčiams.

Svarbu pastebėti, kad paruošiamieji veiksmai ir apsimokymas atliekamas prieš patvirtinant transakciją, o metodo efektyvumas argumentuojamas mažo duomenų rinkinio idėja, todėl be papildomų efektyvumo įverčių neįmanoma įvertinti ar šis metodas tinkamas naudoti gamybinėje aplinkoje.

1.3.4. Aptikimas paremtas dirbtiniais neuroniniais tinklais

Taisyklėmis paremti aptikimo modeliai yra klasikinis apgaulės aptikimo sprendimas, kuris techniškai lengvai realizuojamas, tačiau reikalauja daug ekspertinių žinių. Dėl šios priežasties buvo sukurtas dirbtiniais neuroniniais tinklais paremtas modelis. Naudojant neuroniniais tinklais paremtą realizaciją pasiekiamas iki 40 proc. geresnis rezultatas nei naudojant taisyklėmis paremtą modelį [GR94].

Tokių tyrimo rezultatų negalima vienareikšmiškai priimti kaip pripažinimo, kad taisyklėmis paremti metodai yra neefektyvus. Taisyklių rinkinio efektyvumas tiesiogiai priklauso nuo taisyklių parinkimo, taip pat kaip ir apsimokančios sistemos atveju – tikslumas tiesiogiai priklausomas nuo tinkamo duomenų rinkinio paruošimo.

Pateikiamoje metodo realizacijoje, kaip analizuojami duomenų atributai naudojami: sąskaitos numeris, mokėjimo suma, pardavėjo kodas ir laikas. Iš šių pagrindinių atributų buvo

išvesti 20 papildomų sudėtinių atributų. Tačiau neuroninį tinklą įgyvendinanti sukčiavimo aptikimo realizacija, taip pat nėra optimali. Nebuvo atliekamas atributų normalizavimas, reikšmių skirstymas į kategorijas ar segmentavimas siekiant rasti tikslinį duomenų rinkinį.

1.3.5. Bajeso pasitikėjimo tinklais paremtas aptikimas

Sukčiavimo aptikimo sistema turi gebėti tiksliai įvertinti transakcijas nepaisant nebalansuoto duomenų rinkinio ir duomenų triukšmo [MTV+02].

1.3.5.1. Metodo realizacija

Bajeso pasitikėjimo tinklai naudoja Bajeso teoremą, kuri apibrėžia sąlyginės tikimybės išraišką. Bajeso tinklas išreiškiamas orientuotu acikliniu grafu. Kiekvienas grafo mazgas apibūdinamas keliomis būsenomis. Mazgai siejami tarpusavyje naudojant grafo kelius. Kiekvienam grafo mazgui priskiriama tikimybinių įverčių lentelė, kuri nusako tikimybę, kad mazgas bus kiekvienoje iš jam apibrėžtų būsenų. Naudojant tokį metodą galima identifikuoti įvykio tikimybę, jeigu žinomos pradinės atributų reikšmes.

Turėdami paruoštą Bajeso modelį, jam pateikiame transakcijos duomenis, pagal sąlyginės tikimybės formulę gauname įvertį, nusakanti kokia tikimybė, kad mokėjimas su tam tikrais atributais yra sukčiavimas. Šio metodo trūkumas tas, kad grafo struktūra nėra identifikuojama automatiškai, tai turi atlikti ekspertas.

Šis metodas yra panašus į Markovo modelius, tačiau Markovo modeliai vertina tikimybę, kad įvyks būsenos pasikeitimas, tuo tarpu Bajeso tinklas, vertina vieno įvykio būsenos parinkimo tikimybę. Dėl šios priežasties Bajeso tikimybinių tinklų paruošimas yra paprastesnis ir efektyvesnis procesas. Pakanka įvertinti kiekvieno mazgo būvimo kiekvienoje būsenoje tikimybes remiantis elementų kiekiu ir elementų, tam tikroje būsenoje, kiekiu. Egzistuoja šio metodo išvestinis metodas – naivieji Bajeso tinklai (angl. *naive bayes*), kurie turi tik vieną esminį skirtumą – atskiriami įeigos mazgai, kurie grafe negali turėti tarpusavio kelių, todėl negali daryti įtakos kitų įeigų būsenoms. Tokiu būdu gaunamas paprastesnis grafo pavidalas,

1.3.5.2. Apibendrinimas

Palyginami Bajeso pasitikėjimo tinklus naudojantys metodai su dirbtinius neuroninius tinklus naudojančiu metodais. Abejais atvejais šie metodai priskiriami nuspėjančiajai analitikai, abiem paruošti reikalingi duomenų rinkiniai su iš anksto klasifikuotais duomenimis. Galima teigti, kad realiuose panaudos atvejuose dirbtinių neuroninių tinklų panaudojimas būtų paprastesnis. Dirbtinio neuroninio tinklo apmokymui reikia paruošti duomenų rinkinį su tinkamai parinktais atributais. Tuo tarpu pasitikėjimo tinklo atveju reikalingas toks pats duomenų rinkinio paruošimas, tačiau papildomai reikia sudaryti būsenų modelį. Vis dėlto Bajeso tinklai užtikrina geresnį tikslumą nei neuroniniai tinklai, yra greičiau apmokomi [MTV+02].

Taip pat verta pastebėti, kad neautomatizuotas būsenų modelio sudarymas suteikia didesnes konfigūravimo galimybes, tačiau sukūrus netinkamą modelį, bus neigiamai įtakojamas aptikimo rezultatas.

1.3.6. Imunine sistema paremtas aptikimas

Sukčiavimo aptikimo metodas turėtų prisitaikyti prie naujų apgaulės būdų. Šiomis dienomis sukčiavimo aptikimo sistemos dažniausiai vis dar veikia ne realaus laiko principu, o sukčiavimo atveju įverčiai yra perduodami sukčiavimo vertinimo skyriams, atkreipiamas dėmesys, kad sukčiavimo aptikimas dažnai vertinamas tikslumo įverčiais [HA14]. Metodas įgyvendinamas naudojant imuninės sistemos algoritmą mėgina spręsti šias problemas.

Imuninės sistemos metodas išnaudoja žmogaus imuninės sistemos veikimo idėją. Imuninė sistema tiesiogiai sprendžia visas problemas, kurias turėtų išspręsti sukčiavimo aptikimas [HA14]. Žmogaus imuninės sistemos tikslas - identifikuoti svetimkūnius. Imuninės sistemos dirba su nebalansuotais duomenų rinkiniais, kadangi dauguma ląstelių gerosios. Be to imuninė sistema geba atsižvelgti į pasislėpusius svetimkūnius bei išmoksta identifikuoti naujus svetimkūnius.

1.3.6.1. Metodo realizacija

Imuninės sistemos metodo paruošimas vyksta neigiamos atrankos principu. Erdvė atsitiktinėse vietose užpildoma ląstelėmis detektoriais, įterpiami nesukčiavimo duomenys. Detektoriai, kurie persidengia su gerosiomis duomenų ląstelėmis sunaikinami. Po paruošimo erdvė yra užpildyta detektoriais erdvės vietose, kuriose nebuvo gerųjų duomenų objektų.

Tokio metodo privalumas, kad jis apmokomas naudojant tik gerųjų duomenų atvejus, tai didelis privalumas sukčiavimo aptikime, kadangi tarp sukčiavimo ir gerų transakcijų vyrauja didelis disbalansas.

Neigiama atranka yra neefektyvi sukčiavimo aptikimo atveju [HA14]. Vietoje neigiamos atrankos metode naudojamas apsimokymas naudojantis detektorių reitingavimą. Reitinguojant detektoriui persidengus su gerąja duomenų ląstele jis nėra šalinamas. Sumažinamas detektoriaus reitingas, o duomenų objektas įtraukiamas į ląstelės atmintį, tam kad nebebūtų identifikuojami panašūs objektai.

Vykstant naujos transakcijos vertinimui detektoriai esantys arčiausiai transakcijos objekto gražina savo įverčius. Kiekvienas įvertis prieš apjungimą koreguojamas pagal detektoriui priskirtą reitingą. Detektorius duomenų objektą vertina, pagal savo atmintyje išsaugotus duomenų objektus.

Siekiant įvertinti sukčiavimo aptikimo rezultatus naudojamas kainų modelis. Apsibrėžiama, kad neidentifikuotas sukčiavimas kainuoja 100 valiutos vienetų, neteisingas aliarmas – 10 valiutos vienetų. O identifikuoto sukčiavimo apdorojimo administraciniai kaštai –

1 valiutos vienetas. Vertinant rezultatus naudojantis šiomis transakcijų vertėmis skaičiuojamas finansinės naudos įvertis.

1.3.6.2. Metodo apibendrinimas

Reikia atkreipti dėmesį, kad metodas nenaudoja į duomenų paruošimo. Taip pat kaina paremto skaičiavimo modelis remiasi konstantomis išreikštomis transakcijų vertėmis. Kainos įverčio duomenys yra iškreipiami, o mėginimas įvertinti neteisingo aliarmo finansinę vertę yra subjektyvus, kadangi realūs praradimai sudaromi iš administracinių kaštų ir pelno prarasto dėl kliento nepasitenkinimo.

1.3.7. Hibridinis sukčiavimo vertinimas

Nereikia vienareikšmiškai pasitikėti vienu sukčiavimo aptikimo metodu [KC16]. Hibridinis metodas apjungia 6 sukčiavimo aptikimo metodus į vieną bendrą vykdomą balsavimo principu. Šio jungtinio metodas įgyvendinamas naudojant sprendimų medžius (angl. *decision tree*), atsitiktinius miškus (angl. *random forest*), Bajeso tikimybinius tinklus bei naiviuosius Bajeso tikimybinius tinklus, palaikančiųjų vektorių metodą, k modelių klasterizavimo metodą.

1.3.7.1. Sprendimų medžiai

Sprendimų medžio (angl. *decision tree*) metodai atlieka nuspėjančiąją analizę, todėl jiems paruošti reikalingas suklasifikuotas duomenų rinkinys. Apsimokymo proceso metu sudaromas medžio pavidalo grafas, kurio kiekvienas mazgas išreiškiamas funkcija. Pagal šios funkcijos rezultatą nusprendžiama kuria grafo šaka leistis žemyn. Duomenų įrašo vertinimo procesas pradedamas medžio šaknyje ir baigiamas viename iš medžio lapų. Teigiama, kad sprendimo medžiai gerai apdoroja triukšmą, taip pat yra lengvai suprantami žmonėms ir sudarymui reikalauja sąlyginai mažai duomenų.

Savo ruožtu atsitiktinis miškas (angl. *random forest*) nėra visiškai naujas metodas. Miškas sudaromas iš tam tikro kieko sprendimo medžių. Duomenų rinkinys, skirtas metodo apmokymui, atsitiktinai su grąžinimu skaidomas į kelis duomenų rinkinius, tada kiekvienas iš poaibių naudojamas apmokyti atskirą sprendimo medį. Po miško apmokymo, vertinant transakciją, miške transakcija perduodama visiems sprendimų medžiams, o gauti rezultatai apjungiami naudojant daugumos principą.

1.3.7.2. Atraminiai vektoriai

Atraminių vektorių metodas (angl. *support vector machines*) atlieka duomenų klasterizavimą. Duomenys pildomi daugiamatėje erdvėje vektorių pavidalu, užpildžius duomenis siekiama rasti tokią hiperplokštumą, kuri būtų maksimaliu atstumu tarp duomenų grupių erdvėje. Atitinkamai gavus naują transakciją galima įvertinti, į kurią erdvės dalį ji patenka, taip nustatysime klasterį, kuriam ji priklauso.

1.3.7.3. K-vidurkių klasterizavimas

K-vidurkių metodas remiasi klasterizavimu (angl. *K-means clustering*), todėl jam nereikalingas iš anksto suklasifikuotas duomenų rinkinys. Metodas atlieka pakopinį taškų perkėlimą link klasterių centrų. Prieš pradedant klasterizavimą, nurodoma į kelias duomenų grupes turi būti suskirstyti duomenis. Tada daugiamatėje erdvėje atsitiktinai parenkama tiek erdvės taškų, kiek buvo pasirinkta grupių, šie taškai laikomi būsimų klasterių centrais. Tuomet pirmosios iteracijos metu, kiekvienas duomenų įrašas priskiriamas arčiausiai esančiam klasterio centrui. Suskirsčius visus erdvės taškus, grupės pagrindinis taškas perstatomas į jam priskirtų taškų geometrinį centrą. Toliau procesas kartojamas, vykdoma antra iteracija. Visi erdvės taškai vėl laikomi niekam nepriskirtais, kiekvienas taškas iš naujo priskiriamas perstatytam artimiausiam klasterio centrui. Procesas kartojamas tol kol nusistovi klasterių centrų pozicija erdvėje.

1.3.7.4. Hibridinio metodo realizacija

Visi metodai naudojami kartu. Gavus užklausą įvertinti transakciją, ji pateikiama visiems aptikimo metodams vienu metu ir gauti rezultatai apjungiami. Rezultatų apjungimui siūlomos 4 strategijos:

- daugumos principu;
- optimistinio balsavimo;
- pesimistinio balsavimo;
- paremta svoriais.

Pirmosiose trijose strategijose daroma prielaida, kad būdai pateikia dvejetainį atsaką.

Daugumos principas trivialus. Laikoma, kad transakcija yra sukčiavimas, jeigu įverčių nurodančių, kad tai yra sukčiavimas, daugiau nei nurodančių priešingai.

Optimistinis įvertis teigia, kad transakcija nėra sukčiavimas, jei bent vienas iš algoritmų grąžiną įvertį, nurodantį kad transakcija nėra sukčiavimas.

Pesimistinio atveju laikoma, kad transakcija yra sukčiavimas jeigu bent vienas algoritmas gražina atsaką, kad tai yra sukčiavimas.

Svoriais paremto įverčio skaičiavimo atveju, pirmiausia visų metodų grąžinami rezultatai normalizuojami į intervalą [0;1], dauginami iš svorio priskirto metodui ir sudedami. Svoriai metodams paskirstomi taip, kad galutinis metodo rezultatas taip pat priklausytų intervalui [0;1]. Svorius rekomenduojama parinkti po algoritmo apmokymo atlikus testavimą su žinomais duomenimis [KC16]. Svoris priskiriamą metodui apibrėžiamas kaip algoritmo rezultato santykis su visų algoritmų rezultatų suma. Gaunama, kad jeigu algoritmas aptiks tai, ko nepavyko aptikti kitiems algoritmams, jis gaus didesnį santykinį svorį, o mažiau tikslus algoritmas gaus mažesnį svorį.

Šios 4 jungtinės strategijos buvo palygintos su pavieniais naudojamais metodais. Nustatyta, kad prieš apjungimą geriausiai pasirodė Bajeso pasitikėjimo tinklų metodas. Siūlomos 4 sukčiavimo aptikimo strategijos išties leidžia pasirinkti sukčiavimo aptikimo metodą pagal organizacijai reikalingas savybes. Optimistinis metodas turėjo tik 0,1 proc. netikrų aliarmų kiekį, tačiau pasižymėjo tik 30 proc. jautrumo įverčiu. Pesimistinis, priešingai, aptiko apie 94 proc. sukčiavimo, tačiau turėjo net apie 14 proc. netikrų aliarmų santykį. Tuo tarpu svoriais paremtas balsavimo metodas pateikė pakankamai gerą vidutinį aptikimo variantą, su 64 proc. jautrumu ir mažesniu nei 1 proc. netikrų aliarmų kiekiu. Naivieji Bajeso tinklai užtikrina 92 proc. jautrumą, tačiau sukėlia mažiau nei 5 proc. netikrų aliarmų, o įprasti Bajeso tinklai užtikrina 50 proc. jautrumą tačiau netikrų aliarmų kiekis išliko mažesnis nei 1 proc [KC16].

1.3.7.5. Apibendrinimas

Galima teigti, kad hibridinio sukčiavimo aptikimo idėja yra verta dėmesio. Svorinis modelis iš tiesų užtikrino aukštą tikslumą išlaikydamas minimalų netikrų aliarmų skaičių. Vis dėlto norint tokį algoritmą įgyvendinti gamybinėje aplinkoje reikėtų sudaryti ir prižiūrėti 6 sukčiavimo aptikimo sistemas. Tai reikalautų žymiai daugiau resursų, o siekiant užtikrinti realaus laiko atsaką, spręsti skirtingų metodų efektyvumo problemas.

Taip pat neatsižvelgiama į tai, kad skirtingi algoritmai skirti spręsti skirtingoms problemoms, todėl naudoti tą patį duomenų rinkinį gali būti netikslinga. Galima teigti, kad butų galima pasiekti dar geresnių rezultatų, jei duomenų rinkinys būtų laikomas baziniu, tačiau išvestiniai atributai būtų pritaikomi individualiai kiekvienam metodui, taip siekiant pasiekti maksimalų rezultatą.

Apibendrinant galima teigti, kad nors pristatomas hibridinis aptikimo modelis ir pasiekė geriausius tikslumo rodiklius, tačiau jų pritaikymas brangus ir reikalaujantis daug resursų. Tačiau matoma, kad Bajeso tinklų realizacijos pateikė rezultatus artimus hibridiniam modeliui. Naivieji Bajeso tinklai, palyginus su pesimistine strategija, pasiekė panašų jautrumo rodiklį, tačiau sukėlė daugiau nei dvigubai mažesnį netikrų aliarmų kiekį. Tuo tarpu įprastieji Bajeso tikimybiniai tinklai vieninteliai buvo sulyginami su svoriniu modeliu, sukėlė nežymiai didesnį netikrų aliarmų kiekį ir sulyginamą jautrumą.

1.3.8. Sukčiavimo aptikimo metodų realizacijų apibendrinimas

Atlikus akademinėje literatūroje pateikiamų realizacijų analizę galima daryti išvadą, kad sukčiavimo aptikimas nėra tiriamas ir analizuojamas, kaip programų sistemos kūrimo uždavinys. Tyrimai koncentruojasi tik į tam tikrą realizacijos dalį: duomenų apdorojimą, įverčių skaičiavimą ar naudojamą duomenų gavybos metodą. Sukčiavimo efektyvumo vertinimai metodų palyginimuose buvo nereikalingi dėl aptikimo vykdymo paketinio apdorojimo būdu, tačiau dabartinėms sukčiavimo aptikimo sistemoms keliamas reikalavimas pateikti atsaką realiu laiku,

prieš transakcijos patvirtinimą. Dauguma pritaria realaus laiko sukčiavimo aptikimo idėjai ir mėgina ją vystyti, tačiau norint užtikrinti šio reikalavimo įgyvendinimą, reikalingi vienareikšmiški sukčiavimo aptikimo efektyvumo įverčiai. Metodą vertinant kaip duomenų analizės uždavinį, galima išmatuoti statistines tikslumo metrikas, tačiau sukčiavimo aptikimas galiausiai yra įgyvendinamas programų sistemos pavidalu. Efektyvumui ir įgyvendinamumui įtaką daro ne tik naudojamas algoritmas, bet ir pasirinkta sistemos architektūra. Dėl šios priežasties norint vertinti sukčiavimo aptikimo metodus ir lyginti juos tarpusavyje, reikia tyrimą atlikti pilnai apimant duomenų paruošimą, sukčiavimo įverčių gavimą bei sistemos kontekstą.

Remiantis atliktų tyrimų rezultatais, buvo patvirtinta, kad nors taisyklių rinkiniais paremtas sukčiavimo aptikimas yra nesudėtingai įgyvendinamas, tačiau duomenų gavybos metodai lankstesni ir įgalina tikslesnį aptikimą. Apibendrinant gavyba paremtus metodus, buvo identifikuota, kad Bajeso tikimybiniai tinklai yra tinkamas metodas sukčiavimo aptikimui, kadangi skirtinguose tyrimuose buvo prieita prie išvadų, kad jie sukčiavimą leido aptikti tiksliau nei dažniausiai naudojami pavieniai metodai ir pateikė rezultatus artimus hibridiniam modeliui apjungiančiam 6 pavienius metodus. Bajeso tikimybiniai tinklai taip pat leidžia pakankamai greitą modelio paruošimą, o metodo rezultatai, nors ir nėra taip paprastai interpretuojami kaip sprendimo medžių atveju, tačiau lengvai suprantami turint tikimybių teorijos žinių. Dėl šių priežasčių tolimesni sukčiavimo aptikimo projektavimo darbai bus atliekami naudojant Bajeso tinklus, kaip kuriamos sukčiavimo aptikimo sistemos pagrindą.

2. Siūlomas sukčiavimo aptikimo metodas

2.1. Bajeso tinklo analizė

„Bajeso tinklai yra grafinės struktūros atvaizduojančios tikimybinius sąryšius tarp didelio kiekio kintamųjų ir išvadų darymas iš šių kintamųjų“ [Nea03]. Tinklai įgyvendinami naudojant Bajeso teoremą, aprašytą Thomas Bayes 1763 metais. Šis tikimybinio samprotavimo būdas yra plačiai paplitęs ir naudojamas įvairiose srityse: klientų segmentavime, veido atpažinime, diagnozavime.

Bajeso tinklas – aciklinis, orientuotas grafas turintis ribotą skaičių mazgų. Kiekvienas mazgas yra apibrėžiamas baigtiniu skaičiumi būsenų. Jei tinkle turime mazgus A ir E, o iš mazgo E eina briauna į mazgą A, laikome, kad E yra mazgo A tėvinis mazgas. Kiekvienam iš tinklo mazgų yra apibrėžiama įvykių jungties tikimybių lentelė, kurioje nurodomos mazgo būsenų ir jo tėvinių mazgų būsenų jungties tikimybės. Tinkle briaunomis sujungiami tik tie įvykiai, kurių tarpusavio priklausomybė laikoma reikšminga. Taip mažinamas tinklo kompleksiškumas. Galimi skirtingi būdai apibrėžti tinklo struktūrą:

- naudojantis ekspertinėmis žiniomis;
- naudojantis istorinių duomenų rinkiniu.

Sudarant tinklą naudojantis ekspertinėmis žiniomis, tinklo mazgai ir reikšmingos įvykių tarpusavio priklausomybės identifikuojamos eksperto. Sudarant tinklą iš istorinio duomenų rinkinio ekspertai identifikuoja tinklo mazgus, tačiau įvykių tarpusavio priklausomybes tinkle pagal transakcijų duomenis identifikuoja algoritmai.

2.1.1. Įverčio gavimas

Siekiant manipuliuoti tinklo efektyvumu bei vertinti įverčio gavimo sudėtingumą būtina suprasti tinklų veikimo principus bei Bajeso teoremą.

Sąlyginė tikimybė apibrėžia įvykio tikimybę su sąlyga, kad žinomas kitas įvykis. Tegu A, E – įvykiai. Tada įvykio A tikimybė su sąlyga E vadiname sąlygine tikimybe ir žymime $P(A|E)$. Todėl $P(SUK|ATRIBUTAI)$ žymi tikimybę, kad transakcija su žinoma atributų kombinacija yra sukčiavimas, čia įvykis SUK nurodo, kad transakcija yra sukčiavimas. Įvykis $ATRIBUTAI$ nurodo, kad transakcija sudaryta iš mums žinomos transakcijos atributų kombinacijos.

Jeigu $P(A|E) \neq P(A)$ įvykiai A, E yra priklausomi. Priklausomų įvykių jungties tikimybė yra lygi pirmojo įvykio tikimybei padaugintai iš antrojo įvykio tikimybės, su sąlyga, kad pirmasis įvykis įvyko: $P(A \cap E) = P(A) \cdot P(E|A) = P(E) \cdot P(A|E)$. Bendru atveju jungties tikimybės išraiška:

$$P(A_1 \cap A_2 \cap \dots \cap A_n) = P(A_1) \cdot P(A_2|A_1) \cdot \dots \cdot P(A_n|A_{n-1} \cap A_{n-2} \cap \dots \cap A_2 \cap A_1)$$

Čia $A_1, A_2, \ldots, A_n$ tarpusavyje priklausomi įvykiai, o n – bet koks natūralusis skaičius. Iš lygybės aišku, kad siekiant bendru atveju apskaičiuoti įvykių jungties tikimybę reikia atsižvelgti į visų įvykių tarpusavio priklausomybes. Tai apsunkina įvykių jungties tikimybinio įverčio gavimą.

Įrodyti, kad įvykiai yra visiškai tarpusavyje nepriklausomi yra sudėtinga. Dažnai įvykiai yra tarpusavyje priklausomi, tačiau jų tarpusavio poveikis labai mažas. Naudojant Bajeso tinklą iš anksto apsibrėžiama, kurių įvykių tarpusavio priklausomybes laikome reikšmingomis. Tai leidžia supaprastinti skaičiavimus ir Bajeso tinklus efektyviai naudoti su didesniais tinklo mazgų kiekiais.

Iš įvykių jungties apskaičiavimo formulės išvedama supaprastinta Bajeso teoremos išraiška - $P(A|E) = \frac{P(A)\cdot P(E|A)}{P(E)}$. Ši Bajeso teoremos išraiška parodo kaip remiantis istorinėmis žiniomis įvertinti sąlyginę tikimybę[6]. Vietoje įvykių A ir B įstatę įvykius SUK ir $ATRIBUTAI$ gauname - $P(SUK|ATRIBUTAI) = \frac{P(SUK)\cdot P(ATRIBUTAI|SUK)}{P(ATRIBUTAI)}$. Iš to galima daryti prielaidą, kad siekiant įvertinti sukčiavimo tikimybę pakanka žinoti:

- bendrą tikimybę sutikti sukčiavimo transakciją;
- tikimybę sutikti transakciją su šiuo konkrečiu atributų rinkiniu tarp sukčiavimo transakcijų;
- tikimybę sutikti transakciją su šiuo atributų rinkiniu tarp visų transakcijų.

Dedamąsias galima apskaičiuoti iš istorinio duomenų rinkinio pagal klasikinį įvykio tikimybės apibrėžimą: $P(SUK) = \frac{n_{suk}}{n}$, $P(ATRIBUTAI|SUK) = \frac{n_{suk\_atributai}}{n_{suk}}$, $P(ATRIBUTAI) = \frac{n_{atributai}}{n}$, čia n – transakcijų kiekis istoriniame duomenų rinkinyje, n_{suk} – sukčiavimo transakcijų kiekis istoriniame duomenų rinkinyje, $n_{atributai}$ – transakcijų su tokia pat atributų kombinacija kiekis istoriniame duomenų rinkinyje, $n_{suk\_atributai}$ – sukčiavimo transakcijų su tokia pat atributų kombinacija kiekis istoriniame duomenų rinkinyje.

Ši Bajeso teoremos išraiška puikiai demonstruoja Bajeso teoremos taikymą ir Bajeso tinklų veikimo principą, tačiau toks sukčiavimo tikimybių įvertinimas būtų nekorektiškas. Sukčiavimas yra retas reiškinys, o skirtingų transakcijos atributų kombinacijų kiekis yra labai didelis. Dėl šios priežasties duomenų rinkinyje dažnai nepavyktų identifikuoti sukčiavimo transakcijų su ta pačia atributų kombinacija. Ši problema bus sprendžiama parenkant tinkamą tinklo struktūrą vykdant kriterijų agregavimą, duomenų normalizavimą, reikšmių skirstymą į kategorijas.

[6] Bendroji Bajeso teoremos išraiška pateikiama [Kub96] šaltinyje

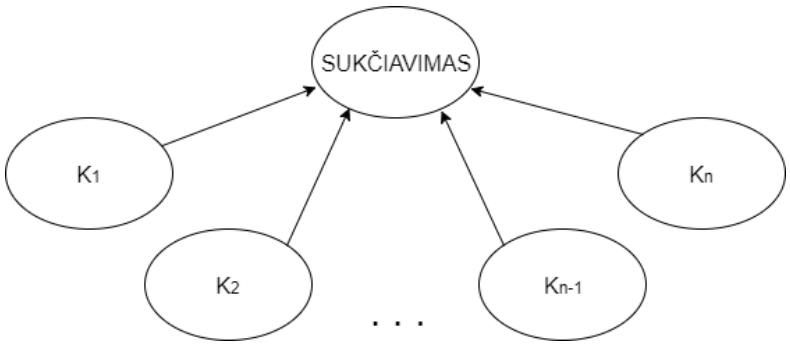
2.1.2. Pavidalas

Tinklo pavidalas tiesiogiai veikia įverčio gavimo efektyvumą, saugomų tikimybinių lentelių kiekį ir dydį. Siekiant sudaryti tinklą galintį pateikti sukčiavimo įvertį realiu laiku, būtina galimybė įvertinti ir pagrįsti tinklo efektyvumą. Todėl sudarant tinklo pavidalą priklausomybių identifikavimo algoritmai nenaudojami. Tinklo struktūra bus apibrėžiama naudojant ekspertines žinias.

Apibrėžus tinklo struktūrą ir tikimybines lenteles gaunamas bendrojo pavidalo Bajeso tinklas. Naudojantis šiuo tinklu pagal skirtingas žinomų mazgų aibes galima įvertinti, bet kurio tinkle esančio mazgo būsenos tikimybę. Dėl šios priežasties tinklai naudojami tokiose srityse kaip vaizdo atpažinimas. Sudarant atpažinimo tinklą iš anksto nėra žinoma kokius objektus reikės identifikuoti ir kokias vaizdo savybes, galinčias padėti vaizdo atpažinimui, identifikuosime. Transakcijos yra atliekamos pagal griežtus protokolus. Transakcijos atributai yra iš anksto žinomi ir nekintantys. Visais atvejais skaičiuojamas sukčiavimo įvykio įvertis. Iš to galima daryti prielaidą, kad sukčiavimo aptikimui bendro pavidalo tinklas nėra būtinas. Kadangi tinklo pavidalas gali daryti įtaką įverčio gavimo efektyvumui darbe bus atliktas skirtingo pavidalo tinklų efektyvumo tyrimas.

2.1.3. Vieno lygio tinklas

Vieno lygio tinklu darbe laikomas tinklas, kuriame siekiamo rezultato mazgas tiesiogiai



priklausomas nuo visų stebimų būsenų (žr. 1 pav.).

1 pav. Tiesioginės priklausomybės pavidalas

Čia mazgas „SUKČIAVIMAS“ žymi įvykį SUK, $K = \{K_1, \ldots, K_n\}$ – stebimų kriterijų mazgų aibė, čia n – stebimų kriterijų kiekis išreiškiamas natūraliuoju skaičiumi. Kiekvienas kriterijus apibrėžiamas kriterijaus būsenų aibe. Kiekvieno kriterijaus būsenos ir būsenų kiekis gali skirtis.

Neatsiejama tinklo dalis yra įvykių jungties tikimybių lentelės. Lentelėse apibrėžiamos tikimybės sutikti mazgą ir jo tėvinius mazgus kiekvienoje iš įmanomų būsenų kombinacijų. Šio

pavidalo tinklui reikalingos dviejų tipų tikimybių lentelės: kriterijaus mazgų lentelės, sukčiavimo mazgo lentelė.

Vieno lygio tinkle daroma prielaida, kad kriterijų mazgai neturi tėvinių mazgų ir yra tarpusavyje nepriklausomi. Dėl šios priežasties kriterijų mazgų lentelėse saugomos tikimybės sutikti kiekvieną mazgo būseną. Tegu B_i yra i - tojo kriterijaus būsenų aibė, b_{ij} yra i – tojo kriterijaus j – toji būsena, $|B_i|$ i – tojo mazgo būsenų kiekis išreiškiamas būsenų aibės dydžiu. Tuomet tarkime, kad p_{ij} yra i - tojo kriterijaus, j – tosios būsenos tikimybė. $p_{ij} \in [0;1]$, čia i, j – natūralieji skaičiai ir $i \leq n$, $j \leq |B_i|$.Tada kiekvienam, kriterijui K_i, egzistuoja lentelė (žr. 1 lentelė).

1 lentelė. i - tojo kriterijaus tikimybių lentelė

Būsena	Tikimybė
b_{i1}	p_{i1}
b_{i2}	p_{i2}
...	...
$b_{i\|B_i\|}$	$p_{i\|B_i\|}$

Akivaizdu, kad reikės n lentelių, o kiekviena lentelė turės $|B_i|$ duomenų įrašų. Todėl šioms lentelėms išsaugoti reikės $\sum_{i=1}^{n} |B_i|$ duomenų įrašų. Tegu $b_{max} = \max_{i \leq n} |B_i|$, tada $\sum_{i=1}^{n} |B_i| \leq n \cdot b_{max}$. Lygybė tenkinama kai visi kriterijai turi vienodą būsenų kiekį. Iš to gauname, kad maksimalus duomenų įrašų skaičius yra $n \cdot b_{max}$.

Sukčiavimo mazgo lentelėje reikia pateikti sukčiavimo mazgo ir kriterijų mazgų būsenų jungties kombinacijų tikimybes. Tegu įvykis SUK apibrėžiamas kaip įvykis, nurodantis ar įvyko sukčiavimas. Šis įvykis apibrėžiamas būsenomis $B_{SUK} = \{tiesa,\ netiesa\}$ Naudojant šį tinklo pavidalą sukčiavimo įvykis priklausomas nuo visų kriterijų, todėl lentelėje nurodomos tikimybės įvykti visoms galimoms kriterijų būsenų kombinacijoms: (žr. 2 lentelė)

2 lentelė. Sukčiavimo įvykio tikimybių lentelė

Būsenos				Tikimybė
$SUKČIAVIMAS$	K_1	...	K_n	
$tiesa$	b_{11}	...	b_{n1}	p_{suk1}
$netiesa$	b_{11}	...	b_{n1}	p_{suk2}
$tiesa$	b_{12}	...	b_{n1}	p_{suk3}
$netiesa$	b_{12}	...	b_{n1}	p_{suk4}
...	...	...	...	...

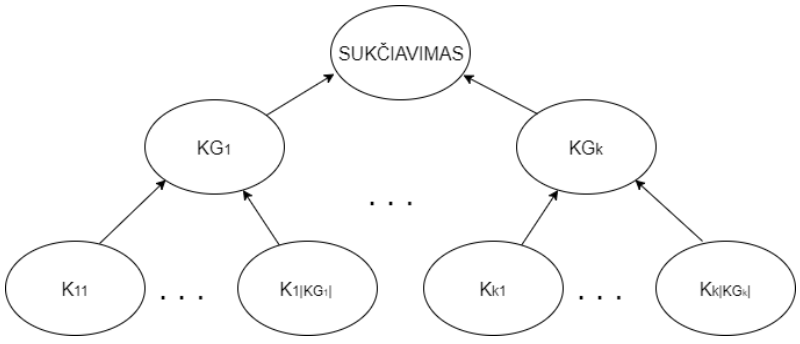
Duomenų įrašų kiekis lentelėje būtų lygus skirtingų galimų būsenų kombinacijų kiekiui: $|B_{SUKČIAVIMAS}| \cdot \prod_{i=1}^{n} |B_i|$. Žinoma, kad $|B_{SUK}| = 2$, o $b_{max} = \max_{i \leq n} |B_i|$, tada $2 \cdot \prod_{i=1}^{n} |B_i| \leq 2 \cdot {b_{max}}^n$. Iš nelygybės aišku, kad blogiausiu atveju duomenų įrašų skaičius yra $2 \cdot {b_{max}}^n$.

Kadangi n apibrėžia kriterijų skaičių tinkle, galima teigti, kad duomenų įrašų kiekis auga eksponentiškai, priklausomai nuo kriterijų skaičiaus. Apibrėžus 30 kriterijų, iš kurių kiekvienas turi 5 būsenas, gautumėme apie $1{,}8 \cdot 10^{21}$ duomenų įrašų. Jeigu 1 duomenų įrašą sugebėtumėme išsaugoti 1 baite atminties, mums reiktų apie $1{,}7 \cdot 10^{9}$ TB. Realiose duomenų saugyklose įrašai užima daugiau nei 1 baitą, todėl darbo rašymo metu nėra praktinių galimybių tokio duomenų kiekio išsaugojimui. Dėl didelio skirtingų kombinacijų kiekio tikėtina, kad duomenų rinkinyje nebus statistiškai reikšmingo transakcijų kiekio su tokia pačia kriterijų kombinacija, todėl įverčiai bus nekorektiški.

Taip pat, vieno lygio pavidalas gali apriboti ekspertus primityvioje tinklo struktūroje. Tikėtina, kad ekspertai gali identifikuoti tarpusavyje susijusių kriterijų ar kriterijų grupių, kurių neįmanoma išreikšti šio pavidalo tinklu. Todėl apibendrinant galima teigti, kad nepriklausomai nuo to, kokį atsako laiką ar tikslumą gali užtikrinti, toks tinklas yra neįgyvendinamas realioje aplinkoje.

2.1.3.1. Dviejų lygių tinklas

Įvertinus vieno lygio tinklo pavidalą identifikuotos problemos kylančios dėl didelio galimų kriterijų kombinacijų kiekio, didelių atminties poreikių, primityvios struktūros. Kelių lygių tinklas įveda papildomus mazgus leidžiančius apsibrėžti kriterijų grupavimą, identifikuoti tarpusavyje susijusius kriterijus (žr. 2 pav.)



2 pav. Netiesioginės priklausomybės tinklo pavidalas

Tegu KG_i yra i - toji kriterijų grupė, čia k – kriterijų grupių kiekis ir $i <= k$. K_{ij} yra i - tosios grupės j - tasis kriterijus. Siekiant prasmingo palyginimo tarkime, kad naudojame kriterijus iš vieno lygio tinklo sugrupuotus į kriterijų grupes. Todėl $\sum_{i=1}^{k} |KG_i| = n$. Bendrinei šio pavidalo tinklo realizacijai reikia trijų tipų lentelių:

- nusakanti K_{ij} būsenų tikimybes, su visais i, j;
- nusakanti KG_i būsenų tikimybes, su visais i;

- nusakanti SUK būsenų tikimybes.

Kriterijų K_{ij} lentelėse duomenų įrašų kiekis bus lygus visų kriterijų grupių kiekvieno kriterijaus būsenų sumai: $\sum_{i=1}^{k}\sum_{j=1}^{|KG_i|}|B_{ij}|$. Žinant, kad $b_{max} = \max_{i\leq n}|B_i|$, $\sum_{i=1}^{k}|KG_i| = n$, gauname: $\sum_{i=1}^{k}\sum_{j=1}^{|KG_i|}|B_{ij}| \leq \sum_{i=1}^{k}\sum_{j=1}^{|KG_i|} b_{max} = b_{max}\cdot\sum_{i=1}^{k}|KG_i| = b_{max}\cdot n$. Iš to aišku, kad blogiausiu atveju reikės saugoti $b_{max}\cdot n$ duomenų įrašų.

Kriterijų grupavimo mazgams KG_i bendruoju atveju reikia paruošti kriterijų grupės ir tėvinių kriterijų būsenų jungties tikimybes: $P\left(KG_i \cap K_{i1} \cap \ldots \cap K_{i|KGi|}\right)\forall i \leq k$. Kiekvienai kriterijų grupei paruošiamos jos tėvinių kriterijų jungties būsenų kombinacijos. Kiekvienos lentelės duomenų įrašų kiekis priklauso nuo mazgo KG_i būsenų kiekio ir tėvinių kriterijų būsenų kiekio. Todėl yra $\left|B_{KG_i}\right|\cdot\prod_{j=1}^{|KG_i|}\left|B_{ij}\right|$ skirtingų būsenų kombinacijų vienai kriterijų grupei. Sudėjus visų kriterijų grupių kombinacijas gaunamas visų kombinacijų kiekis: $\sum_{i=1}^{k}\left|B_{KG_i}\right|\cdot\prod_{j=1}^{|KG_i|}\left|B_{ij}\right|$. Žinant, kad $b_{max} = \max_{i\leq k}\max_{j\leq|KG_i|}\left|B_{ij}\right|$, $b_{KGmax} = \max_{i\leq k}\left|B_{KG_i}\right|$, $KG_{max} = \max_{i\leq k}|KG_i|$, gaunama:

$$\sum_{i=1}^{k}\left|B_{KG_i}\right|\cdot\prod_{j=1}^{|KG_i|}\left|B_{ij}\right| \leq \sum_{i=1}^{k}\left|B_{KG_i}\right|\cdot {b_{max}}^{|KG_i|} \leq {b_{max}}^{KG_{max}}\cdot\sum_{i=1}^{k}\left|B_{KG_i}\right|$$

$$\leq {b_{max}}^{KG_{max}}\cdot b_{KGmax}\cdot k$$

Iš to aišku, kad blogiausiu atveju galimas duomenų įrašų kiekis bus ${b_{max}}^{KG_{max}}\cdot b_{KGmax}\cdot k$.

SUK mazgui reikia apibrėžti mazgo ir kriterijų grupių būsenų jungties tikimybes $P(SUK \cap KG_1 \cap \ldots \cap KG_k)$. Šiuo atveju reikalinga vienintelė lentelė. Skirtingų būsenų kombinacijų kiekis priklauso nuo sukčiavimo mazgo būsenų kiekio ir kriterijų grupių būsenų kiekių: $|B_{SUK}|\cdot\prod_{i=1}^{k}|B_{KG_i}|$. Žinant, kad $b_{KGmax} = \max_{i\leq k}\left|B_{KG_i}\right|$, $|B_{SUK}| = 2$. Gaunama: $|B_{SUK}|\cdot\prod_{i=1}^{k}\left|B_{KG_i}\right| \leq 2\cdot {b_{KGmax}}^{k}$. Iš to aišku, kad blogiausiu atveju galimas duomenų įrašų kiekis: $2\cdot {b_{KGmax}}^{k}$.

Sudėjus visų lentelių duomenų įrašų poreikius įvertintus didžiausiam galimam kombinacijų kiekiui gaunama, kad iš viso didžiausias galimas duomenų įrašų kiekis: $b_{max}\cdot n + {b_{max}}^{KG_{max}}\cdot b_{KGmax}\cdot k + 2\cdot {b_{KGmax}}^{k}$. Siekiant papildyti tinklą galima pridėti naują kriterijų grupę. Pridėjus naują grupę didės kriterijų grupių kiekis k. Taip pat galima pridėti kriterijus į esamas grupes, šiuo atveju didės maksimalus kriterijų grupėje kiekis - KG_{max}. Akivaizdu, kad abiem atvejais didėja išraiškos laipsnis. Todėl galime teigti, kad tai eksponentiškai auganti priklausomybė.

Priešingai nei vieno lygio tinklo atveju, eksponentė priklauso nuo kriterijų grupių kiekio, o ne nuo visų kriterijų kiekio. Darome prielaidą, kad tinklas suskirstytas į k balansuotų grupių -

$KG_{max} \approx \frac{n}{k}$. Todėl nepaisant to, kad duomenų įrašų kiekis auga eksponentiškai, šios eksponentės laipsnis yra $\approx k$ kartų mažesnis lyginant su vieno lygio tinklu. Pakartotinai įvertinamas praeitame skyrelyje pateiktas pavyzdys. Tegu turima 30 kriterijų, kiekvienas apibrėžiamas 5 būsenomis. Kriterijus padalinti į 5 grupes, po 6 kriterijus. Daroma prielaida, kad kriterijų grupės taip pat apibrėžiamos 5 būsenomis. Įstačius reikšmes į didžiausio kombinacijų kiekio išraišką gaunama: $5 \cdot 5 + 5^6 \cdot 5 \cdot 5 + 2 \cdot 5^5 = 396900$. Vieno lygio tinklo pavidalo atveju nebuvo praktinių galimybių išsaugoti duomenų įrašus, tačiau šis tinklo skaidymas į potinklius sumažino eksponentės laipsnį, žymiai sumažindamas reikalingų duomenų įrašų kiekį.

Kriterijų grupavimas visiškai problemos neišsprendžia, sudėtingumas vietos atžvilgiu išlieka eksponentiškai augantis, todėl nėra pritaikomas sukčiavimo aptikimo tinkluose naudojančiuose šimtus kriterijų. Tačiau $\approx k$ kartų sumažinus eksponentės laipsnį, šis būdas yra pritaikomas naudojant iki 100 kriterijų. Todėl šis būdas įgyvendinamas.

Atliekamas įverčio gavimo tyrimas. Kriterijų grupę galima apibrėžti 2 būdais:

- kriterijų grupės būsena iš anksto žinoma;
- kriterijų grupės būsena iš anksto nėra žinoma.

Darant prielaidą, kad kriterijų grupės iš anksto žinomos, įvykiai SUK ir bet kuris kriterijus K yra sąlyginai nepriklausomi. Jeigu tinkle yra 3 mazgai: A, B, C. Ir B yra mazgo A tėvas, o C yra mazgo B tėvas, tai A ir C yra sąlyginai nepriklausomi, su sąlyga kad žinomas B. Sąlyginai nepriklausomiems įvykiams galioja lygybė: $P(A|B \cap C) = P(A|B)$. Dviejų lygių tinkle ši savybė reiškia, kad žinant KG_i būsenas, čia $i \leq k$, galioja lygybė (žr. 1 priedas):

$$P(SUK|KG_1 \cap KG_2 \cap \ldots \cap KG_k \cap K_{11} \cap \ldots \cap K_{1|KG_{1|}} \cap \ldots \cap K_{k1} \cap \ldots \cap K_{k|KG_k|})$$
$$= P(SUK|KG_1 \cap KG_2 \cap \ldots \cap KG_k)$$

Todėl norint įvertinti įvykio SUK tikimybę, pakanka rasti $P(SUK|KG_1 \cap KG_2 \cap \ldots \cap KG_k)$. Iš sąlyginės tikimybės apibrėžimo žinoma, kad ją galima apskaičiuoti pasinaudojus lentelių duomenimis:

$$P(SUK|KG_1 \cap KG_2 \cap \ldots \cap KG_k) = \frac{P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)}{P(KG_1 \cap KG_2 \cap \ldots \cap KG_k)}$$
$$= \frac{P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)}{P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k) + P(\overline{SUK} \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)}$$

Visas reikalingas reikšmes galima rasti jungtinėje įvykio SUK lentelėje, taigi aišku, kad turint iš anksto paruoštas duomenų lenteles įverčio gavimas yra labai greita operacija. Vis dėlto, tinklui sudaryti buvo ruošiamos trijų tipų lentelės nusakančios kriterijų mazgų, kriterijų grupių mazgų bei sukčiavimo tikimybes.

Darant prielaidą, kad tam tikros kriterijų grupės iš anksto nėra žinomos, kiekvieno kriterijaus grupę galima įvertinti skaičiuojant tikimybę:

$$P(KG_i|K_{i1} \cap \ \ldots \cap \ K_{i|KGi|}) = \frac{P(KG_i \cap K_{i1} \cap \ \ldots \cap \ K_{i|KGi|})}{P(K_{i1} \cap \ \ldots \cap \ K_{i|KGi|})}$$

$$= \frac{P(KG_i \cap K_{i1} \cap \ \ldots \cap \ K_{i|KGi|})}{P(\overline{KG_i} \cap K_{i1} \cap \ \ldots \cap \ K_{i|KGi|}) + P(KG_i \cap K_{i1} \cap \ \ldots \cap \ K_{i|KGi|})}$$

Šiam papildomam veiksmui visas reikalingas reikšmes galima rasti kriterijų grupių lentelėse. Todėl šiuo atveju būtų panaudotos dvi iš trijų paruoštų lentelių grupių.

Šie pavyzdžiai demonstruoja, kaip naudojant bendrinį Bajeso tinklą tik vieno tipo įverčių gavimui yra švaistomi resursai nenaudojamų lentelių paruošimui bei saugojimui. Buvo nustatyta, kad reikalingų duomenų įrašų kiekis priklausomas nuo apibrėžtų kriterijų kiekio. Todėl švaistant duomenų saugyklų resursus neigiamai veikiamas aptikimui galimų naudoti kriterijų skaičius. Taip apsunkinamos aptikimo panaudojimo galimybės. Saugomų duomenų įrašų kiekis nedaro įtakos įverčio pateikimo efektyvumui jį vertinant algoritmo abstrakcijos lygmenyje. Vis dėlto, atsižvelgiant būsimos sistemos realizaciją žinoma, kad norint rasti reikiamą duomenų įrašą didesniuose duomenų rinkiniuose reikia atlikti daugiau palyginimo operacijų. Taip pat reikia nepamiršti, kad pasiekus duomenų kiekį, dėl kurio būtinas duomenų klasterizavimas, kyla potencialios rizikos susidurti su CAP teoremos ribojimais sisteminio įgyvendinimo lygmenyje.

2.1.4. Apibendrinimas

Siekiant pateikti prasmingą tinklo įgyvendinamumo ir efektyvumo tyrimą buvo išanalizuoti Bajeso tinklų veikimo principai. Ištyrus du skirtingus tinklo pavidalus: vieno lygio tinklą, dviejų lygių tinklą, buvo nustatyta, kad vieno lygio Bajeso tinklo duomenų saugojimo poreikiai auga eksponentiškai priklausomai nuo kriterijų kiekio, todėl realistišką kriterijų kiekį turintys tinklai yra neįgyvendinami dėl technologinių ribojimų dabartinėse duomenų saugyklose. Taip pat buvo identifikuota, kad vieno lygio tinklo primityvus pavidalas riboja tikslumą, trukdo išreikšti realistiškas struktūras. Tiriant dviejų lygių tinklą buvo nustatyta, kad tinklo duomenų saugyklos resursų poreikis taip pat auga eksponentiškai. Tačiau kriterijų grupavimas eksponentės laipsnį sumažina iki k kartų, čia k – kriterijų grupių skaičius. Todėl galima teigti, kad naudojant kriterijų grupavimą į potinklius reikšmingai sumažinamas poreikis duomenų saugyklų resursams. Taip pat buvo identifikuota, kad dėl sumažėjusio skirtingų kriterijų būsenų kombinacijų kiekio, šio pavidalo tinklas gali padėti užtikrinti didesnį tikslumą. Kadangi šios struktūros tinklas leidžia apibrėžti tarpusavio sąsajų turinčius kriterijus bei įvesti papildomas kriterijų abstrakcijas, jį lengviau pritaikyti realistiškuose panaudos atvejuose. Todėl apibendrinant tinklo pavidalų tyrimą galima teigti, kad dviejų lygių tinklo struktūra yra pranašesnė ir pritaikoma realistiškose sukčiavimo aptikimo sistemose.

Tiriant tinklo pavidalą buvo parodyta, kad sukčiavimo aptikimui pilnas Bajeso tinklas nėra būtinas dėl transakcijų vykdymo protokolų griežtumo. Išanalizavus įverčių skaičiavimą abiejų

pavidalų atveju, nustatyta, kad siekiant sukčiavimo įverčio gavimo, naudoti pilnąjį Bajeso tinklą yra neefektyvu. Skaičiuojant sukčiavimo įvertį nėra išnaudojamos visos paruoštos duomenų lentelės. Kadangi duomenų saugyklos resursai riboja galimų apsibrėžti kriterijų kiekį, galima teigti, kad toks resursų išnaudojimas riboja tinklo panaudojimo galimybes bei tikslumą.

2.2. Įverčio skaičiavimo analizė

Išanalizavus įverčių apskaičiavimą nustatyta, kad siekiant konkretaus įverčio gavimo pagal statinę kriterijų aibę naudoti bendrąjį Bajeso tinklą yra neefektyvu. Todėl darbe bus siekiama sukurti konkretų optimizuotą sukčiavimo įverčio skaičiavimo būdą Šio būdo pagrindu naudojant dviejų lygių Bajeso tinklą.

Buvo identifikuota, kad kriterijų grupes galima apibrėžti dviem būdais: teigiant, kad grupės būsenos gavus transakciją yra iš anksto žinomos bei teigiant kad būsenos nežinomos. Toliau darbe bus analizuojamas įverčio skaičiavimas sudarant kriterijų grupės abiem būdais.

Skyriuje siekiama įvertinti aptikimo efektyvumą priklausomai nuo tinklo struktūros. Todėl reikalingų atlikti operacijų kiekis ir duomenų saugojimo poreikiai vertinami nesigilinant į realizacijos detales. Operacijos šiame abstrakcijos lygmenyje laikomos elementariomis ir lygiavertėmis. Tuo tarpu duomenų saugojimo poreikis vertinamas duomenų įrašų kiekiu nesigilinat į duomenų tipus.

Darome prielaidą, kad visų kriterijų būsenos lengvai identifikuojamos elementariais matematiniais veiksmais palyginant transakcijos atributus su istoriniu duomenų rinkiniu. Dėl to, vertinant įverčio apskaičiavimą, į kriterijų būsenų nustatymą neatsižvelgiama, atskirai nevertinami duomenų rinkiniai reikalingi išsaugoti transakcijas bei išvestinius transakcijų duomenis reikalingus identifikuoti kriterijų būsenas.

Prieš skaičiuojant transakcijų sukčiavimo įverčius reikia apmokyti tinklą pakankamu istorinių transakcijų kiekiu. Vykdyti aptikimą be pradinio duomenų rinkinio teoriškai įmanoma, tačiau įverčių skaičiavimas pagal statistiškai nežymų duomenų rinkinį pateiktų nekorektiškus įverčius. Taip pat dažnai būtų pateikiami nulinės vertės įverčiai. Todėl sukčiavimo aptikimo sistemą turi būti galima apmokyti iš einamųjų transakcijų duomenų, importuojant istorines transakcijas. Siekiant išvengti aptikimo būdo kompleksiškumo, daroma prielaida, kad esant duomenų importavimo poreikiui bus naudojamas apmokymo procesas apdorojantis einamąsias transakcijas. Tinklo apmokymas neturi tiesiogiai paveikti įverčio pateikimo laiko, tačiau norint užtikrinti metodo įgyvendinamumą būtina ištirti ar apmokymą įmanoma įvykdyti per realistišką laiką su realistišku resursų kiekiu.

2.2.1. Žinomos kriterijų grupės būsenos

Analizė pradedama nuo tinklo, kuriame kriterijų grupės apibrėžimas leidžia identifikuoti kriterijų grupės būseną gavus transakciją įvertinimui. Norint įvertinti šio tinklo efektyvumą reikia atsižvelgti į kriterijų grupės apibrėžimą.

Yra įvairių būdų apibrėžti kriterijų grupės būseną: pagal transakcijų atributus, naudojant palyginimus su istoriniais duomenimis, naudojant ekspertines žinias. Naudojant šiuos būdus įvykis SUK ir apibrėžtos grupės kriterijai tampa sąlyginai nepriklausomi, tokiu atveju grupės kriterijai tiesiogiai nebenaudojami įverčio skaičiavimui. Todėl prasminga apibrėžti kriterijų grupės būsenas taip, kad jos apibendrintų grupės kriterijus. Toks grupės apibrėžimas nėra trivialus, daroma prielaida, kad užtikrinti būdo efektyvumą galima tik įvertinus grupės būsenos identifikavimo efektyvumą.

2.2.1.1. Kriterijų grupės apibrėžimas

Norint įvertinti būsenos identifikavimo efektyvumą būtina apibrėžti kriterijų grupę.

Tarkime, kad kriterijų grupę apibrėžiame kaip įvykį, nurodantį tikėtinumą sutikti šios grupės kriterijų būsenas tarp sukčiavimo transakcijų: $P(SUK \cap K_{i1} \cap \ldots \cap K_{i|KG_i|})$ kiekvienam $i \leq k$. Toks grupės apibrėžimas įvertins grupės kriterijų būsenų rizikingumą. Tikimybinio įverčio panaudoti kaip būsenos negalima, todėl įverčiai skirstomi į 5 kategorijas:

- LM (labai maža rizika);
- M (maža rizika);
- V (vidutinė rizika);
- D (didelė rizika);
- LD (labai didelė rizika).

Tegu P_{ivid} – vidutinis transakcijų rizikos įvertis i – tajai kriterijų grupei tarp istorinių kriterijų grupių kombinacijų, σ_i – standartinis nuokrypis i – tajai kriterijų grupei. Tegu $p_i = P(SUK \cap K_{i1} \cap \ldots \cap K_{i|KG_i|})$. Tada apibrėžtas kategorijas parenkame pagal tai per kiek vidutinių standartinių reikšmė p_i yra nutolusi nuo vidutinės kriterijų grupės kriterijų kombinacijų reikšmių vidurkio.

$$b_{KG_i} = \begin{cases} LM, \ p_i - P_{vid} \leq -2 \cdot \sigma_i \\ M, \ -2 \cdot \sigma_i < p_i - P_{vid} \leq -\sigma_i \\ V, \ -\sigma_i < p_i - P_{vid} < \sigma_i \\ D, \ \sigma_i \leq p_i - P_{vid} < 2 \cdot \sigma_i \\ LD, \ 2\sigma_i \leq p_i - P_{vid} \end{cases}$$

Atlikus tinklo pavidalo analizę (žr. 2.1 poskyris) buvo identifikuota, kad labai didelis skirtingų kriterijų būsenų kombinacijų kiekis neigiamai veikia tikslumą. Pasiūlytas grupavimas leidžia atskirai vertinti kriterijų grupių kombinacijas mažinant bendrą kombinacijų kiekį.

2.2.1.2. Įverčio skaičiavimas

Kadangi daroma prielaidą, kad apibrėžtame tinkle pagal transakcijos atributus galima identifikuoti kriterijų būsenas ir kriterijų grupių būsenas, norint gauti sukčiavimo įvertį reikia apskaičiuoti:

$$P(SUK|K_{11} \cap \ldots \cap K_{1|KG_{1|}} \cap \ldots \cap K_{k1} \cap \ldots \cap K_{k|KG_k|} \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)$$

Kai yra žinomos visos kriterijų grupių būsenos, SUK ir bet kuris kriterijus K_{ij} yra sąlyginai nepriklausomi dėl tinklo struktūros. Pasinaudojus sąlyginės nepriklausomybės apibrėžimu gauname (žr. 1 priedas):

$$P(SUK|KG_1 \cap KG_2 \cap \ldots \cap KG_k \cap K_{11} \cap \ \ldots \cap K_{1|KG_{1|}} \cap \ldots \cap K_{k1} \cap \ldots \cap K_{k|KG_k|})$$
$$= P(SUK|KG_1 \cap KG_2 \cap \ldots \cap KG_k)$$

Todėl pakanka skaičiuoti: $P(SUK|KG_1 \cap KG_2 \cap \ldots \cap KG_k)$. Šią įvykių jungties tikimybę galima išreikšti pagal sąlyginės tikimybės apibrėžimą:

$$P(SUK|KG_1 \cap KG_2 \cap \ldots \cap KG_k) = \frac{P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)}{P(KG_1 \cap KG_2 \cap \ldots \cap KG_k)}$$
$$= \frac{P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)}{P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k) + P(\overline{SUK} \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)}$$

Kadangi galimybė atlikti papildomus samprotavimus tinkle nereikalinga, įmanoma kiekvienai kriterijų grupių būsenų kombinacijai iš anksto skaičiuoti tikimybinius įverčius: $P(SUK|KG_1 \cap KG_2 \cap \ldots \cap KG_k)$. Tačiau norint gauti šį įverti iš bendro pavidalo lentelės pakanka atlikti kelias aritmetines operacijas. Dėl šios priežasties saugosime iš anksto paruoštą standartinio pavidalo lentelę su įverčių $P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)$ kombinacijomis.

Prieš skaičiuojant šį įvertį reikia identifikuoti kriterijų grupių būsenas, todėl kiekvienai transakcijos kriterijų grupei reikia apskaičiuoti įvertį $P\left(SUK \cap K_{i1} \cap \ldots \cap K_{i|KG_{i|}}\right)\ \ i \leq k$ apibrėžiantį kriterijų grupes:

- $P\left(SUK \cap K_{i1} \cap \ldots \cap K_{i|KG_{i|}}\right) = \frac{t_{SUK\_KRIT}}{t}$;
- $P\left(SUK \cap K_{i1} \cap \ldots \cap K_{i|KG_{i|}}\right) = P(SUK) \cdot \prod_{j=1}^{|KG_{i|}} P\left(K_{ij}|SUK\right)$ (žr. 2 priedas).

Pirmuoju būdu įvertis skaičiuojamas pagal tikimybės apbrėžimą, čia t_{SUK_KRIT} – transakcijų kiekis su ta pačia įvykių būsenų reikšmių kombinacija duomenų rinkinyje, t – transakcijų kiekis duomenų rinkinyje. Antruoju būdu įvertis skaičiuojamas naudojant įvykių jungties apskaičiavimo išraišką darant prielaidą, kad kriterijai yra tarpusavyje nepriklausomi. Pirmoji išraiška tikslesnė, kadangi netaikome papildomų prielaidų, tačiau šiai išraiškai gauti reikėtų kiekvienai kriterijų grupėje esančių kriterijų būsenų kombinacijai skaičiuoti pasitaikiusių transakcijų su šia kombinacija kiekį. Tokių kombinacijų kiekis būtų

eksponentiškai augantis priklausomai nuo kriterijų grupės dydžio, todėl dėl efektyvumo naudojamas antrasis skaičiavimo būdas.

Norint įvertinti kriterijų grupės būseną antruoju būdu reikia dviejų dedamųjų: $P(SUK), P(K_{ij}|SUK)$. Daroma prielaida, kad šios dedamosios saugomos iš anksto paruoštos, kadangi tai leistų užtikrinti greitesnį įverčio apskaičiavimą.

2.2.1.2.1. Sudėtingumas operacijų atžvilgiu

Analizuojant įverčio skaičiavimą nustatyta, kad reikia identifikuoti kriterijų grupių būsenas skaičiuojant tikimybę $P\left(SUK \cap K_{i1} \cap \ldots \cap K_{i|KG_i|}\right)$. Tada pagal identifikuotas būsenas apskaičiuoti $P(SUK|KG_1 \cap KG_2 \cap \ldots \cap KG_k)$.

Skaičiuojant $P\left(SUK \cap K_{i1} \cap \ldots \cap K_{i|KG_i|}\right)$, kiekvienai kriterijų grupei KG_i reikės atlikti $|KG_i|$ daugybos veiksmų. Sudėjus kiekvienai grupei reikiamą atlikti veiksmų kiekį gaunama, kad iš viso reikės atlikti $\sum_{i=1}^{k}|KG_i|$ veiksmų. Tegu, $KG_{max} = \max_{i \le k}|KG_i|$, gaunama: $\sum_{i=1}^{k}|KG_i| \le KG_{max} \cdot k$. Iš to aišku, kad blogiausiu atveju atliekamų veiksmų kiekis bus $KG_{max} \cdot k$.

Apskaičiavus tikimybinį įvertį pagal kriterijų grupės apibrėžimą reikia identifikuoti būsenos kategoriją. Iš apibrėžimo aišku, kad kiekvienai kriterijų grupei reikia atlikti 1 atimties veiksmą nuokrypio įvertinimui ir iki 4 palyginimo operacijų kategorijos identifikavimui. Todėl, blogiausiu atveju reikės atlikti $5 \cdot k$ elementarių operacijų.

Identifikavus kriterijų grupių būsenas, apskaičiuoti $P(SUK|KG_1 \cap KG_2 \cap \ldots \cap KG_k)$ galima iš lentelės gavus įverčius $P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)$, $P(\overline{SUK} \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)$ ir atlikus vieną sudėties ir vieną dalybos operaciją.

Susumavus visus reikalingus atlikti veiksmus gaunama, kad blogiausiu atveju reikės: $KG_{max} \cdot k + 5 \cdot k + 2 = (KG_{max} + 5) \cdot k + 2$ elementarių operacijų. Didinant tinklą būtų didinamas kriterijų grupių kiekis k ir kriterijų grupėje kiekis KG_{max}. Abi dedamosios išraiškoje tėra daugikliai. Todėl galima teigti, kad didėjant tinklui reikalingas operacijų kiekis auga polinomiškai. Todėl galima teigti, kad operacijų atžvilgiu šis tinklas yra labai efektyvus ir pritaikomas siekiant pateikti įvertį realiu laiku.

2.2.1.2.2. Sudėtingumas vietos atžvilgiu

Buvo nustatyta, kad siekiant efektyvaus įverčio pateikimo yra prasminga iš anksto paruošti įverčius: $P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)$, $P(SUK)$, $P\left(K_{ij}\middle|SUK\right)$.

Įvykio SUK bei kriterijų grupių būsenų jungties tikimybės $P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)$ turi $|B_{SUK}| \cdot \prod_{i=1}^{k}|B_{KG_i}|$ skirtingų kombinacijų. Žinant, kad

$b_{KGmax} = \max_{i \le k} |B_{KG_i}|$, o $|B_{SUK}| = 2$, gaunama: $|B_{SUK}| \cdot \prod_{i=1}^{k} |B_{KG_i}| \le 2 \cdot {b_{KGmax}}^{k}$. Tada blogiausiu atveju reikalingas išsaugoti duomenų įrašų kiekis: $2 \cdot {b_{KGmax}}^{k}$.

Siekiant išsaugoti įvykio sukčiavimo tikimybę $P(SUK)$ pakanka vieno duomenų įrašo, kadangi saugoma tik vienos būsenos nepriklausančios nuo kitų įvykių tikimybė.

Reikia išsaugoti kriterijaus būsenos pasitaikymo tarp sukčiavimo transakcijų tikimybę $P(K_{ij}|SUK)$. Ši tikimybė išreiškiama kiekvieno kriterijaus kiekvienai būsenai, todėl iš viso reikia $\sum_{i=1}^{k} \sum_{j=1}^{|KG_i|} |B_{ij}|$ įverčių. Žinant, kad $b_{max} = \max_{i \le k} \max_{j \le |KG_i|} |B_{ij}|$, o $\sum_{i=1}^{k} |KG_i| = n$ gaunama: $\sum_{i=1}^{k} \sum_{j=1}^{|KG_i|} |B_{ij}| \le \sum_{i=1}^{k} \sum_{j=1}^{|KG_i|} b_{max} = \sum_{i=1}^{k} |KG_i| \cdot b_{max} = b_{max} \cdot \sum_{i=1}^{k} |KG_i| = b_{max} \cdot n$, čia n – visų kriterijų kiekis. Iš to aišku, kad blogiausiu atveju duomenų įrašų kiekis: $b_{max} \cdot n$.

Sudėjus visus kiekius gaunama, kad blogiausiu atveju bus $2 \cdot {b_{KGmax}}^{k} + 1 + b_{max} \cdot n$ duomenų įrašų. Prie tinklo pridedant naujus kriterijus didės visas kriterijų kiekis tinkle – n. Šiuo atveju duomenų įrašų kiekis auga polinomiškai. Tačiau pridedant naujas kriterijų grupes didės kriterijų grupių kiekis – k. Todėl galima teigti, kad didinant kriterijų grupių kiekį didėja išraiškos laipsnis k, o duomenų įrašų kiekis didėja eksponentiškai. Kadangi duomenų įrašų kiekio augimas priklauso ne nuo viso kriterijų kiekio, o nuo kriterijų grupių kiekio, duomenų įrašų kiekis žymiai yra mažesnis nei bendrojo pavidalo tinklo atveju.

2.2.1.3. Tinklo apmokymas

Buvo identifikuota, kad norint apskaičiuoti sukčiavimo įvertį reikalingos dedamosios: $P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)$, $P(SUK)$, $P(K_{ij}|SUK)$. Tinklo apmokymu laikomas procesas, kuris iš transakcijų duomenų geba paruošti šias dedamąsias įverčio gavimo procesui. Norint įvertinti apmokymo efektyvumą apibrėžiami dedamųjų gavimo procesai.

Reikia paruošti tikimybes sutikti sukčiavimą su specifinėmis kriterijų grupių būsenomis $P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)$. Įvertį galima apskaičiuoti pagal tikimybės apibrėžimą, skaičiuojant transakcijų su kiekviena būsenų kombinacija kiekį ir visų transakcijų kiekio santykį. Tačiau tokiu atveju kiekvienai kriterijų grupių būsenų kombinacijai reikia sekti sukčiavimo transakcijų kiekius. Šių kombinacijų kiekis auga eksponentiškai, todėl įvertis skaičiuojamas pagal: $P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k) = P(SUK) \cdot \prod_{i=1}^{k} P(KG_i|SUK)$ (žr. 2 priedas).

Tikimybę sutikti sukčiavimo transakciją duomenų rinkinyje galime apskaičiuoti pagal klasikinį tikimybės apibrėžimą $P(SUK) = \frac{t_{SUK}}{t}$, čia t – transakcijų kiekis duomenų rinkinyje. t_{SUK} – sukčiavimo transakcijų kiekis duomenų rinkinyje.

Tikimybę sutikti sukčiavimo transakciją su specifinė kriterijų grupės būsena galima apskaičiuoti pagal tikimybės apibrėžimą $P(KG_i|SUK) = \frac{t_{KG_i}}{t_{SUK}}$, čia t_{SUK} – sukčiavimo transakcijų

kiekis duomenų rinkinyje. t_{KG_i} – transakcijų su specifine kriterijų grupės KG_i būsena kiekis sukčiavimo transakcijų rinkinyje. Šiuo atveju bus $|B_{SUK}| \cdot \sum_{i=1}^{k} |B_{KG_i}|$ skirtingų kriterijų grupių būsenų kurioms reikės skaičiuoti šiuos įverčius. Tegu $b_{KGmax} = \max_{i \le k} |B_{KG_i}|$, $|B_{SUK}| = 2$, gaunama: $|B_{SUK}| \cdot \sum_{i=1}^{k} |B_{KG_i}| \le 2 \cdot k \cdot b_{KGmax}$. Iš to aišku, kad blogiausiu atveju skirtingų būsenų bus: $2 \cdot k \cdot b_{KGmax}$.

Tikimybė sutikti transakcijos kriterijų duomenų rinkinyje renkantis iš visų sukčiavimo transakcijų: $P(K_{ij}|SUK) = \frac{t_{K_{ij}}}{t_{SUK}}$, čia t_{SUK} – sukčiavimo transakcijų kiekis duomenų rinkinyje. $t_{K_{ij}}$ – transakcijų su specifine kriterijaus K_{ij} būsena kiekis sukčiavimo transakcijų rinkinyje. Aišku, kad bus $\sum_{i=1}^{k} \sum_{j=1}^{|KG_i|} |B_{ij}|$ skirtingų būsenų. Žinant, kad $b_{max} = \max_{i \le k} \max_{j \le |KG_i|} |B_{ij}|$, o $\sum_{i=1}^{k} |KG_i| = n$ gaunama:

$$\sum_{i=1}^{k} \sum_{j=1}^{|KG_i|} |B_{ij}| \le \sum_{i=1}^{k} \sum_{j=1}^{|KG_i|} b_{max} = \sum_{i=1}^{k} |KG_i| \cdot b_{max} = b_{max} \cdot \sum_{i=1}^{k} |KG_i| = b_{max} \cdot n$$

Čia n – visų kriterijų skaičius. Iš to aišku, kad blogiausiu atveju kombinacijų kurioms reikės skaičiuoti įverčius kiekis: $b_{max} \cdot n$.

2.2.1.3.1. Sudėtingumas operacijų atžvilgiu

Buvo identifikuota, kad apmokymo metu reikės perskaičiuoti ir atnaujinti šias išraiškas: $P(SUK) = \frac{t_{SUK}}{t}$, $P(K_{ij}|SUK) = \frac{t_{K_{ij}}}{t_{SUK}}$, $P(KG_i|SUK) = \frac{t_{KG_i}}{t_{SUK}}$,

$P(SUK \cap KG_1 \cap KG_2 \cap \dots \cap KG_k) = P(SUK) \cdot \prod_{i=1}^{k} P(KG_i|SUK)$

Sukčiavimo tikimybė $P(SUK) = \frac{t_{SUK}}{t}$, ši tikimybė nepriklausoma nuo skirtingų būsenų kombinacijų, todėl norint ją atnaujinti pakanka vieno dalybos veiksmo.

Kriterijaus būsenos tikimybė sukčiavime – $P(K_{ij}|SUK) = \frac{t_{K_{ij}}}{t_{SUK}}$. Įvertinta, kad blogiausiu atveju bus $b_{max} \cdot n$ skirtingų būsenų, todėl blogiausiu atveju reikės atlikti $b_{max} \cdot n$ dalybos operacijų, kad perskaičiuoti visas tikimybes.

Kriterijų grupės būsenos tikimybė sukčiavime – $P(KG_i|SUK) = \frac{t_{KG_i}}{t_{SUK}}$. Įvertinta, kad blogiausiu atveju bus $2 \cdot k \cdot b_{KGmax}$ skirtingų būsenų, todėl blogiausiu atveju reikės atlikti $2 \cdot k \cdot b_{KGmax}$ dalybos operacijų.

Reikia įvertinti $P(SUK \cap KG_1 \cap KG_2 \cap \dots \cap KG_k) = P(SUK) \cdot \prod_{i=1}^{k} P(KG_i|SUK)$ atnaujinimą. Įvertinta, kad jungtinė tikimybė turi iki $2 \cdot {b_{KGmax}}^{k}$ skirtingų būsenų kombinacijų. Vienai kombinacijai reikia atlikti k daugybos veiksmų. Iš viso reikės atlikti $2 \cdot k \cdot {b_{KGmax}}^{k}$ daugybos veiksmų.

Sudėjus operacijų kiekius gaunama, kad blogiausiu atveju reikės atlikti: $1 + b_{max} \cdot n + 2 \cdot k \cdot b_{KGmax} + 2 \cdot k \cdot {b_{KGmax}}^{k}$ elementarių operacijų. Prie tinklo pridedant papildomas kriterijų grupes didės reikšmė k. Iš to galima teigti plečiant tinklą priklausomai nuo kriterijų grupių kiekio eksponentiškai auga reikalingas atlikti operacijų kiekis. Kadangi tinklo apmokymas tiesiogiai nedaro įtakos sukčiavimo įverčio pateikimo laikui, galima teigti, kad nors apmokymas ir nėra efektyvus, tačiau jis įgyvendinamas turint realistišką resursų kiekį.

2.2.1.3.2. Sudėtingumas vietos atžvilgiu

Buvo identifikuota, kad apmokymo metu reikės išsaugoti duomenis reikalingus tikimybių perskaičiavimui: $P(SUK) = \frac{t_{SUK}}{t}$, $P(K_{ij}|SUK) = \frac{t_{K_{ij}}}{t_{SUK}}$, $P(KG_i|SUK) = \frac{t_{KG_i}}{t_{SUK}}$,

$P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k) = P(SUK) \cdot \prod_{i=1}^{k} P(KG_i|SUK)$. Kadangi bet kurį įvertį galima gauti atlikus vieną dalybos operaciją su dviem skaitliukais, nesaugome kiekvienos įverčių kombinacijos. Darome prielaidą, kad pakanka saugoti bazinius skaitliukus.

Reikia išsaugoti skaitliukus: $t, t_{SUK}, t_{K_{ij}}, t_{KG_i}$. t - nuo būsenų nepriklausomas skaitliukas, todėl reikalingas 1 duomenų įrašas. t_{SUK} - priklausomas nuo įvykio SUK kuris turi 2 būsenas, todėl reikalingi 2 duomenų įrašai. $t_{K_{ij}}$ – priklausomas nuo kriterijų būsenų kiekio, todėl iš viso reikės $n \cdot b_{max}$ duomenų įrašų. t_{KG_i} - priklausomas nuo kriterijų grupių būsenų kiekio, todėl reikės $k \cdot b_{KGmax}$ duomenų įrašų.

Sudėjus duomenų įrašų kiekius gaunama, kad blogiausiu atveju reikės išsaugoti: $1 + 2 + b_{max} \cdot n + k \cdot b_{KGmax}$ duomenų įrašų. Todėl nepriklausomai nuo transakcijų kiekio reikės išsaugoti polinomiškai kintantį duomenų įrašų kiekį. Todėl galima teigti, kad apmokymas yra įgyvendinamas turint realistiškus duomenų saugyklų resursus.

2.2.1.4. Apibendrinimas

Buvo pateiktas pilnas sukonkretinto tinklo apibrėžimas: kriterijų grupės apibrėžimas, pasiūlytas būdas apskaičiuoti sukčiavimo įvertį, pasiūlytas būdas įgyvendinti apmokymą. Kadangi šis aptikimo būdas naudoja iš anksto apibrėžta kriterijų grupės apibrėžimą, jis yra sunkiau pritaikomas prie specifinių sukčiavimo aptikimo poreikių. Pakeitus kriterijų grupės apibrėžimą efektyvumas nebūtų užtikrinamas.

Ištyrus sukonkretintą tinklą buvo nustatyta, kad naudojant kriterijų grupės apibrėžimą, leidžiantį nustatyti būseną prieš skaičiuojant įvertį, pakanka atlikti polinomiškai augantį operacijų kiekį skaičiuojant sukčiavimo tikimybę. Vis dėlto, iš anksto paruošiamų duomenų įrašų kiekis auga eksponentiškai priklausomai nuo kriterijų grupių skaičiaus. Todėl šis aptikimo būdas netinkamas naudoti su dideliais kriterijų kiekiais. Ištyrus tinklo apmokymą buvo nustatyta, kad blogiausiu atveju reikalingų atlikti operacijų kiekis auga eksponentiškai, tačiau papildomų apmokymo procesui reikalingų duomenų įrašų kiekis saugykloje didėja polinomiškai.

Skaičiuojant įvertį vienai transakcijai yra atsižvelgiama tik į nedidelę dalį duomenų rinkinio, todėl po kiekvienos transakcijos atnaujinti visą duomenų rinkinį yra neefektyvu. Dėl greitai didėjančio reikalingų operacijų kiekio, apmokymą reikėtų vykdyti grupuojant transakcijas. Kadangi apmokymas neturėtų tiesiogiai paveikti įverčio gavimo laiko, galima teigti, kad šio tinklo apmokymas yra įvykdomas per realistišką laiką turint realistišką resursų kiekį.

Apibendrinant galima teigti, kad dėl nedidelio reikalingų operacijų kiekio apibrėžtas sukčiavimo aptikimo būdas yra tinkamas sukčiavimo aptikimo sistemos realizacijai kuri turėtų realiu laiku pateikti sukčiavimo įvertį. Vis dėlto, šio būdo efektyvumą leido užtikrinti išankstinis kriterijų grupės apibrėžimas, todėl šio būdo pritaikomumas yra ribotas.

2.2.2. Bendras skaičiavimo būdas

Kadangi išankstinis kriterijų grupių apibrėžimas riboja panaudojimo galimybes, bus analizuojama galimybė sukurti bendresnį aptikimo būdą, kuris būtų lengviau pritaikomas realistiškose situacijose. Todėl siekiant platesnių pritaikymo galimybių vengiama naudoti konkretų kriterijų grupės apibrėžimą.

2.2.2.1. Įverčio skaičiavimas

Kadangi siekiama nenaudoti grupės mazgo apibrėžimo daroma prielaida, kad iš transakcijos galima sužinoti tik kriterijų reikšmes. Todėl norint įvertinti sukčiavimą reikia apskaičiuoti (žr. 3 priedas):

$$P\left(SUK\middle|K_{11}\cap\ \dots\cap K_{1|KG_{1|}}\cap\dots\cap K_{k1}\cap\dots\cap K_{k|KG_k|}\right)$$

$$=\sum_{\substack{KG_1\in KG_1^\Omega\\ \dots\\ KG_k\in KG_k^\Omega}}\left(P(SUK|KG_1\cap\dots\cap KG_k)\cdot\prod_{i=1}^{k}P(KG_i|K_{i1}\cap K_{i2}\cap\dots\cap K_{ik})\right)$$

Įverčio gavimui reikalingos dedamosios iš anksto apskaičiuojamos: $P(SUK\cap KG_1\cap KG_2\cap\dots\cap KG_k)$, $P\left(KG_i\cap K_{i1}\cap\ \dots\cap\ K_{i|KGi|}\right)$.

Kiekvienai kriterijų grupių būsenų kombinacijai įvertinama tikimybė su šia kombinacija sutikti sukčiavimą. Kadangi grupių būsenos nežinomos papildomai atsižvelgiama į tikimybę sutikti šią kombinaciją. Gaunamas maksimaliai lankstus būdas. Naudojant šią bendro pavidalo išraišką galima rasti įvertį ne tik tada kai nežinoma nei viena kriterijų grupės būsena, tačiau ir tada kai nežinomos tik kelios grupių būsenos, kadangi į tą pačią išraišką galima įstatyti iš anksto žinomas būsenas. Taip būtų sumažintas skirtingų kombinacijų kiekis, tai efektyvumo neigiamai nepaveiks. Toliau analizuojamas atvejis kai visos būsenos nežinomos, taip efektyvumo prasme įvertinamas blogiausias galimas atvejis.

2.2.2.1.1. Sudėtingumas operacijų atžvilgiu

Kad apskaičiuoti sandaugą: $P(SUK|KG_1 \cap KG_2 \cap ... \cap KG_k) \cdot \prod_{i=1}^{k} P(KG_i|K_{i1} \cap K_{i2} \cap ... \cap K_{ik})$ reikia atlikti k daugybos veiksmų. Skaičiuojant įvertį ši sandauga bus skaičiuojama kiekvienai nežinomų būsenų kombinacijai. Iš viso kombinacijų: $\prod_{i=1}^{k}|B_{KG_i}|$. Žinant, kad $b_{KGmax} = \max_{i \le k}|B_{KG_i}|$, gaunama: $\prod_{i=1}^{k}|B_{KG_i}| \le {b_{KGmax}}^{k}$. Iš to aišku, kad blogiausiu atveju kombinacijų bus ${b_{KGmax}}^{k}$. Kiekvienos kombinacijos sandaugą reikia sudėti į bendrą sumą. Taigi, iš viso blogiausiu atveju reikės $k \cdot {b_{KGmax}}^{k}$ operacijų.

Galima teigti, kad skaičiavimo būdas, kai visos kriterijų grupių būsenos žinomos, yra palankesnis siekiant greito atsako laiko, kadangi juo naudojantis pakaktų atlikti polinomiškai augantį operacijų kiekį. Šiuo atveju gauta eksponentinė operacijų kiekio priklausomybė nuo kriterijų grupių skaičiaus. Vis dėlto, eksponentė auga priklausomai nuo kriterijų grupių skaičiaus, o ne nuo kriterijų skaičiaus, todėl šis būdas turėtų būti pritaikomas realioje aplinkoje.

2.2.2.1.2. Sudėtingumas vietos atžvilgiu

Saugomos iš anksto paruoštos įvykių jungties tikimybės: $P(SUK \cap KG_1 \cap KG_2 \cap ... \cap KG_k)$, $P(KG_i \cap K_{i1} \cap \ ... \cap \ K_{i|KGi|}) \ \forall i \le k$.

Sukčiavimo ir kriterijų grupių įvykių jungties tikimybė $P(SUK \cap KG_1 \cap KG_2 \cap ... \cap KG_k)$ turi $|B_{SUK}| \cdot \prod_{i=1}^{k}|B_{KG_i}|$ skirtingų įvykių kombinacijų. Žinant, kad $b_{KGmax} = \max_{i \le k}|B_{KG_i}|$, o $|B_{SUK}| = 2$, gauname: $|B_{SUK}| \cdot \prod_{i=1}^{k}|B_{KG_i}| \le 2 \cdot {b_{KGmax}}^{k}$. Iš to aišku, kad blogiausiu atveju reikalingas duomenų įrašų kiekis: $2 \cdot {b_{KGmax}}^{k}$.

Kriterijų grupės ir kriterijų įvykių jungties tikimybė $P(KG_i \cap K_{i1} \cap \ ... \cap \ K_{i|KGi|})$ turi $|B_{KG_i}| \cdot \prod_{j=1}^{|KG_i|}|B_{ij}|$ skirtingų būsenų kombinacijų kiekvienai kriterijų grupei. Susumavus visų kriterijų grupių lenteles gaunama: $\sum_{i=1}^{k}|B_{KG_i}| \cdot \prod_{j=1}^{|KG_i|}|B_{ij}|$. Tegu $b_{max} = \max_{i \le k} \max_{j \le |KG_i|}|B_{ij}|$, $b_{KGmax} = \max_{i \le k}|B_{KG_i}|$, $KG_{max} = \max_{i \le k}|KG_i|$, tada: $\sum_{i=1}^{k}|B_{KG_i}| \cdot \prod_{j=1}^{|KG_i|}|B_{ij}| \le {b_{max}}^{KG_{max}} \cdot b_{KGmax} \cdot k$. Todėl blogiausiu atveju reikalingas duomenų įrašų kiekis: ${b_{max}}^{KG_{max}} \cdot b_{KGmax} \cdot k$

Sudėjus rezultatus gaunama, kad blogiausiu atveju reikalingas išsaugoti duomenų įrašų kiekis: $2 \cdot {b_{KGmax}}^{k} + {b_{max}}^{KG_{max}} \cdot b_{KGmax} \cdot k$. Kadangi didinant tinklą bus pridedami nauji kriterijai ir didės reikšmė k, arba bus pridedami kriterijai į esamas kriterijų grupes ir didės KG_{max}. Todėl prie tinklo pridedant naujus kriterijus duomenų įrašų kiekis didės eksponentiškai.

2.2.2.2. Tinklo apmokymas

Norint apskaičiuoti sukčiavimo įvertį reikalingos paruoštos įvykių jungties tikimybės: $P(SUK \cap KG_1 \cap KG_2 \cap ... \cap KG_k)$, $P(KG_i \cap K_{i1} \cap \ ... \cap \ K_{i|KGi|})$.

Sukčiavimo ir kriterijų grupių įvykių jungties tikimybės bus skaičiuojamos pagal (žr. 1 priedas): $P(SUK \cap KG_1 \cap KG_2 \cap ... \cap KG_k) = P(SUK) \cdot \prod_{i=1}^{k} P(KG_i|SUK)$. Jungties

tikimybė turi $|B_{SUK}| \cdot \prod_{i=1}^{k} |B_{KG_i}|$ skirtingų kombinacijų. Žinant, kad $b_{KGmax} = \max_{i \le k} |B_{KG_i}|$, o $|B_{SUK}| = 2$, gaunama: $|B_{SUK}| \cdot \prod_{i=1}^{k} |B_{KG_i}| \le 2 \cdot {b_{KGmax}}^{k}$. Iš to aišku, kad blogiausiu atveju kombinacijų kiekis: $2 \cdot {b_{KGmax}}^{k}$.

Kriterijų grupės ir kriterijų įvykių jungties tikimybės bus skaičiuojamos pagal: $P(KG_i \cap K_{i1} \cap \ldots \cap K_{i|KGi|}) = P(KG_i) \cdot \prod_{j=1}^{k} P(K_{ij}|KG_i)$ Ši jungties tikimybė turi $|B_{KG_i}| \cdot \prod_{j=1}^{|KG_i|} |B_{ij}|$ skirtingų būsenų kombinacijų kiekvienai kriterijų grupei. Susumavus visų kriterijų grupių kombinacijas gaunama: $\sum_{i=1}^{k} |B_{KG_i}| \cdot \prod_{j=1}^{|KG_i|} |B_{ij}|$. Žinant, kad: $b_{max} = \max_{i \le k} \max_{j \le |KG_i|} |B_{ij}|$, $b_{KGmax} = \max_{i \le k} |B_{KG_i}|$, $KG_{max} = \max_{i \le k} |KG_i|$, gaunama: $\sum_{i=1}^{k} |B_{KG_i}| \cdot \prod_{j=1}^{|KG_i|} |B_{ij}| \le {b_{max}}^{KG_{max}} \cdot b_{KGmax} \cdot k$. Tada blogiausiu atveju kombinacijų: ${b_{max}}^{KG_{max}} \cdot b_{KGmax} \cdot kn$

Šioms įvykių jungties tikimybėms apskaičiuoti reikalingos dedamosios: $P(SUK)$, $P(KG_i|SUK)$, $P(KG_i)$, $P(K_{ij}|KG_i)$.

Tikimybė sutikti sukčiavimo transakciją duomenų rinkinyje apskaičiuojama pagal: $P(SUK) = \frac{t_{SUK}}{t}$, čia t – transakcijų kiekis duomenų rinkinyje, t_{SUK} – sukčiavimo transakcijų kiekis duomenų rinkinyje.

Tikimybė sutikti transakcijos kriterijų grupę duomenų rinkinyje renkantis iš visų sukčiavimo transakcijų apskaičiuojama pagal: $P(KG_i|SUK) = \frac{t_{KG_i}}{t_{SUK}}$, čia t_{SUK} – sukčiavimo transakcijų kiekis duomenų rinkinyje, t_{KG_i} – transakcijų atitinkančių specifinę kriterijaus grupės KG_i būseną kiekis. Iš viso bus $|B_{SUK}| \cdot \sum_{i=1}^{k} |B_{KG_i}|$ skirtingų būsenų. Žinant, kad $b_{KGmax} = \max_{i \le k} |B_{KG_i}|$, o $|B_{SUK}| = 2$, gaunama: $|B_{SUK}| \cdot \sum_{i=1}^{k} |B_{KG_i}| \le 2 \cdot k \cdot b_{KGmax}$. Iš to aišku, kad blogiausiu atveju bus $k \cdot b_{KGmax}$ skirtingų būsenų kombinacijų.

Tikimybė sutikti transakcijos grupės būseną duomenų rinkinyje renkantis iš visų transakcijų: $P(KG_i) = \frac{t_{KG_i}}{t}$, čia t – transakcijų kiekis duomenų rinkinyje. t_{KG_i} – transakcijų kiekis kurių kriterijus KG_i atitinka specifinę būseną. Egzistuoja $\sum_{i=1}^{k} |B_{KG_i}|$ skirtingų būsenų. Žinant, kad $b_{KGmax} = \max_{i \le k} |B_{KG_i}|$, gaunama: $\sum_{i=1}^{k} |B_{KG_i}| \le k \cdot b_{KGmax}$. Iš to aišku, kad blogiausiu atveju bus $k \cdot b_{KGmax}$ skirtingų kriterijų grupių būsenų kombinacijų.

Tikimybė sutikti transakcijos kriterijaus būseną renkantis iš transakcijų su specifine transakcijų kriterijų grupės būsena: $P(K_{ij}|KG_i) = \frac{t_{K_{ij}}}{t_{KG_i}}$, čia $t_{K_{ij}}$ - transakcijų su i – tosios kriterijų grupės, j - tojo kriterijaus specifine būsena, kiekis. t_{KG_i} - transakcijų su i – tosios

kriterijų grupės specifine būsena, kiekis. Kiekvienai kriterijų grupei galima sudaryti $\left|B_{KG_i}\right| \cdot \prod_{j=1}^{|KG_i|}\left|B_{ij}\right|$ skirtingų būsenų kombinacijų. Susumavus visų kriterijų grupių kombinacijų kiekius: $\sum_{i=1}^{k}\left|B_{KG_i}\right| \cdot \prod_{j=1}^{|KG_i|}\left|B_{ij}\right|$. Tegu $b_{max} = \max_{i \le k} \max_{j \le |KG_i|} \left|B_{ij}\right|$, $b_{KGmax} = \max_{i \le k}\left|B_{KG_i}\right|$, $KG_{max} = \max_{i \le k} |KG_i|$. Tada: $\sum_{i=1}^{k}\left|B_{KG_i}\right| \cdot \prod_{j=1}^{|KG_i|}\left|B_{ij}\right| \le {b_{max}}^{KG_{max}} \cdot b_{KGmax} \cdot k$. Iš to aišku, kad blogiausiu atveju skirtingų būsenų kiekis: ${b_{max}}^{KG_{max}} \cdot b_{KGmax} \cdot k$.

2.2.2.2.1. Sudėtingumas operacijų atžvilgiu

Apmokymo metu reikės perskaičiuoti ir atnaujinti šias dedamąsias: $P(SUK) = \frac{t_{SUK}}{t}$, $P(KG_i|SUK) = \frac{t_{KG_i}}{t_{SUK}}, P(KG_i) = \frac{t_{KG_i}}{t}, P\left(K_{ij}\middle|KG_i\right) = \frac{t_{K_{ij}}}{t_{KG_i}}$.

Sukčiavimo tikimybė $P(SUK) = \frac{t_{SUK}}{t}$ priklauso tik nuo įvykio SUK būsenų kiekio. Įvykis turi 2 būsenas, todėl galimos 2 kombinacijos. Norint atnaujinti visas kombinacijas pakanka 2 dalybos veiksmų.

Kriterijaus grupės būsenos tikimybė sukčiavime $P(KG_i|SUK) = \frac{t_{KG_i}}{t_{SUK}}$. Buvo identifikuota, kad blogiausiu atveju bus $2 \cdot k \cdot b_{KGmax}$ skirtingų būsenų kombinacijų, todėl atitinkamai blogiausiu atveju reikės atlikti $2 \cdot k \cdot b_{KGmax}$ dalybos operacijų, kad perskaičiuoti visas tikimybes.

Kriterijaus grupės pasirinkimo tarp visų transakcijų tikimybė $P(KG_i) = \frac{t_{KG_i}}{t}$. Buvo identifikuota, kad blogiausiu atveju bus $k \cdot b_{KGmax}$ skirtingų būsenų[7], todėl atitinkamai blogiausiu atveju reikės atlikti $k \cdot b_{KGmax}$ dalybos operacijų, kad perskaičiuoti visas tikimybes.

Kriterijaus tikimybė tarp transakcijų su specifine grupės būsena $P\left(K_{ij}\middle|KG_i\right) = \frac{t_{K_{ij}}}{t_{KG_i}}$. Buvo identifikuota, kad blogiausiu atveju bus ${b_{max}}^{KG_{max}} \cdot b_{KGmax} \cdot k$ skirtingų būsenų kombinacijų[3], todėl blogiausiu atveju reikės atlikti ${b_{max}}^{KG_{max}} \cdot b_{KGmax} \cdot k$ operacijų.

Sukčiavimo ir kriterijų grupių įvykių jungties tikimybė apskaičiuojama pagal: $P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k) = P(SUK) \cdot \prod_{i=1}^{k} P(KG_i|SUK)$. Buvo identifikuota, kad blogiausiu atveju bus $2 \cdot {b_{KGmax}}^{k}$ skirtingų kombinacijų. Kiekvienai būsenų kombinacijai reikės atlikti k sandaugų, todėl iš viso reikės atlikti $2 \cdot k \cdot {b_{KGmax}}^{k}$ veiksmų.

Kriterijaus grupės ir kriterijų jungties tikimybė apskaičiuojama pagal: $P\left(KG_i \cap K_{i1} \cap \ldots \cap K_{i|KGi|}\right) = P(KG_i) \cdot \prod_{j=1}^{k} P(K_{ij}|KG_i)$. Buvo identifikuota, kad blogiausiu

[7] Analizė pateikiama poskyryje 2.2.2.2 Tinklo apmokymas.

atveju bus $b_{max}{}^{KG_{max}} \cdot b_{KGmax} \cdot k$ skirtingų kombinacijų[3]. Kiekvienai būsenų kombinacijai reikia atlikti k sandaugų. Gaunama, kad iš viso reikės atlikti $b_{max}{}^{KG_{max}} \cdot b_{KGmax} \cdot k^2$ veiksmų.

Susumavus dedamųjų apskaičiavimui reikalingas operacijas gaunama: $2 + 2 \cdot k \cdot b_{KGmax} + k \cdot b_{KGmax} + b_{max}{}^{KG_{max}} \cdot b_{KGmax} \cdot k + 2 \cdot b_{max}{}^{KG_{max}} \cdot b_{KGmax} \cdot k^2$. Šiuo atveju didinant kriterijų kiekį tinkle, tačiau juos pridedant prie esamų kriterijų grupių didėtų reikšmė KG_{max}, todėl reikalingų atlikti operacijų kiekis didėtų eksponentiškai. Pridedant naujas kriterijų grupes tinkle didėtų reikšmė k, tokiu atveju reikalingų atlikti operacijų kiekis didėja polinomiškai. Matome, kad operacijų kiekis reikalingas apmokymui auga eksponentiškai priklausant nuo kriterijų grupėje skaičiaus ir polinomiškai nuo kriterijų grupių skaičiaus, tačiau kadangi apmokymas nedaro įtakos įverčio apskaičiavimo laikui, galime teigti, kad apmokymas yra įgyvendinamas.

2.2.2.2.2. Sudėtingumas vietos atžvilgiu

Buvo identifikuota, kad apmokymo metu reikės perskaičiuoti ir atnaujinti šias sudedamąsias išraiškas: $P(SUK) = \frac{t_{SUK}}{t}$, $P(KG_i|SUK) = \frac{t_{KG_i}}{t_{SUK}}$, $P(KG_i) = \frac{t_{KG_i}}{t}$, $P(K_{ij}|KG_i) = \frac{t_{K_{ij}}}{t_{KG_i}}$. Kad įvertinti šias tikimybes pakanka saugoti transakcijų kiekio skaitliukus: $t, t_{SUK,} t_{K_{ij}}, t_{KG_i}$. Šie skaitliukai sutampa su skaitliukais reikalingais žinomų kriterijų grupės tinklo atveju. Buvo identifikuota, kad blogiausiu atveju duomenų įrašų kiekis bus: $1 + 2 + b_{max} \cdot n + k \cdot b_{KGmax}$. Aišku, kad nepriklausomai nuo transakcijų kiekio reikės išsaugoti polinomiškai kintantį duomenų įrašų kiekį. Galima teigti, kad apmokymas yra įgyvendinamas turint realistiškus duomenų saugyklos resursus.

2.2.2.3. Apibendrinimas

Buvo pateiktas pilnas bendrinio tinklo apibrėžimas: pasiūlytas būdas apskaičiuoti sukčiavimo įvertį iš anksto nežinant kriterijų grupių būsenų, pasiūlytas būdas įgyvendinti apmokymą.

Ištyrus aptikimo tinklą naudojantį iš anksto neapibrėžtas kriterijų grupių būsenas buvo nustatyta, kad ne tik duomenų įrašų kiekis, bet ir reikalingų atlikti operacijų kiekis, norint apskaičiuoti sukčiavimo įvertį, auga eksponentiškai. Vis dėlto, abiem atvejais eksponentės augimas priklauso ne tiesiogiai nuo kriterijų kiekio, o nuo kriterijų grupių ir kriterijų grupės būsenų kiekio. Ištyrus tinklo apmokymo procesą nustatyta, kad blogiausiu atveju reikalingų atlikti operacijų kiekis auga eksponentiškai, tačiau duomenų įrašų kiekis auga polinomiškai. Kadangi apmokymo metu vienai transakcijai vykdomų operacijų kiekis auga eksponentiškai priklausomai nuo tinklo struktūros, siekiant efektyvaus resursų išnaudojimo vykdant apmokymą reikia optimizuoti procesą grupuojant transakcijas prieš apdorojimą.

Apibendrinant galima teigti, kad šis būdas yra netinkamas jei siekiama įgyvendinti aptikimą pagal šimtus skirtingų kriterijų, tačiau pakankamas jei sukčiavimo aptikimas apribojamas nedideliu kriterijų kiekiu. Iš anksto nežinant transakcijų grupių būsenų apmokymas yra įvykdomas per realistišką laiką turint realistišką resursų kiekį. Vis dėlto, norint pateikti vienareikšmį atsakymą ar šis aptikimo būdas pakankamai efektyvus naudoti kuriant realaus laiko sukčiavimo aptikimo sistemą, reikia sisteminio lygio efektyvumo analizės.

2.2.3. Apibendrinimas

Naudojantis dviejų lygių Bajeso tinklo įgyvendinimo principais buvo pasiūlyti du tinklai tinkami sukčiavimo aptikimui: tinklas su konkrečiomis kriterijų grupėmis, bendresnio pavidalo tinklas nepriklausomas nuo kriterijų grupių apibrėžimų. Abiems tinklams buvo pateiktas pilnas apibrėžimas: pasiūlytas būdas apskaičiuoti sukčiavimo įvertį, pasiūlytas būdas įgyvendinti apmokymą. Sukonkretintam tinklui buvo pateiktas kriterijų grupės apibrėžimas.

Ištyrus abiejų tinklų įgyvendinamumą, įverčių skaičiavimą bei tinklo apmokymą nustatyta, kad sudarant bendresnio pavidalo aptikimo tinklą, gaunamas eksponentiškai augantis operacijų ir duomenų įrašų kiekis. Tinklus su mažu kriterijų kiekiu galima perstruktūrizuoti taip, kad būtų pakankamai mažas reikalingų atlikti operacijų kiekis, pakaktų išsaugoti realistišką duomenų kiekį. Tačiau šis būdas netinkamas siekiant realaus laiko atsako užtikrinimo naudojant didesnius, sudarytus iš šimtų kriterijų, tinklus. Tuo tarpu, tinklo apibrėžime naudojant konkrečią kriterijų grupę, įverčio apskaičiavimui pakankamas polinomiškai augantis operacijų kiekis. Dėl to, galimas realaus laiko atsako pateikimas su žymiai didesniais tinklais. Vis dėlto, šio tinklo atveju taip pat reikalingas eksponentiškai augantis duomenų įrašų kiekis. Taip pat, kadangi negalima numatyti efektyvumo pakeitus kriterijų grupės apibrėžimą, šio būdo pritaikymas sudėtingesnis.

Apibendrinant galima teigti, kad tinklo sukonkretinimas leidžia sukurti efektyvesnį sukčiavimo aptikimo būdą. Sukonkretinus aptikimo tinklo kriterijų grupes gaunamas efektyvus sukčiavimo aptikimo būdas, kuriam pakanka polinominio kiekio operacijų. Tuo tarpu apibendrinto aptikimo būdo poreikis operacijų kiekiui auga eksponentiškai. Todėl siekiant pateikti sukčiavimo įverti realiu laiku sukonkretintas tinklas yra pranašesnis.

2.3. Aptikimo kriterijai

Siekiama pateikti pilną aptikimo būdo apibrėžimą, todėl būtina identifikuoti kriterijų sudaryme galimus naudoti transakcijos atributus, apibrėžti aptikimui naudojamus kriterijus. Kriterijai atspindės tinklo mazgus ir jų būsenas stebimas tinkle, todėl tinkamai parinktas kriterijų rinkinys teigiamai veikia aptikimo tikslumą. Apibrėžiant kriterijus reikia atsižvelgti į praktikoje taikomus duomenų paruošimo procesus. Tačiau vien šių procesų taikymas negarantuoja aukštų tikslumo rodiklių. Siekiant tikslumo yra būtinos ekspertinės srities žinios bei duomenų rinkiniai

kalibravimui. Duomenų paruošimo proceso įgyvendinimas leis sukurti efektyvų, realioje aplinkoje panaudojamą būdą, kurį galima plėtoti siekiant didesnio tikslumo.

2.3.1. Transakcijos atributai

„Atributas yra duomenų laukas atspindintis charakteristiką ar savybę duomenų objekte“ [HKP11]. Literatūros analizės metu buvo išanalizuotas transakcijų atributų parinkimo procesas, identifikuotos atributų parinkimo gairės. Priklausomai nuo aptikimo vykdytojo skirtingos aptikimo sistemos turi prieigą prie skirtingų transakcijos atributų. Ruošiant metodus tinkamai neatsižvelgiama į sukčiavimo aptikimo sistemos kontekstą [ZYL09]. Norint išvengti netinkamų atributų parinkimo naudojami baziniai EMV standartu atliekamos transakcijos atributai (žr. 1.2.2 poskyris):

- kortelės identifikatorius;
- sąskaitos identifikatorius;
- pardavėjo identifikatorius;
- geografinė transakcijos vieta;
- transakcijos laikas;
- transakcijos suma.

2.3.2. Kriterijų apibrėžimas

Literatūros analizės metu buvo identifikuoti duomenų paruošimo procesai naudojami užtikrinti aukštam tikslumui: duomenų grupavimas, duomenų normalizavimas ir kategorizavimas, išvestinių kriterijų agregavimas.

Duomenų grupavimas apibrėžia duomenų rinkinio dalijimą į grupes siekiant didesnio aptikimo tikslumo. Pagrindiniai grupavimo būdai: asmeninis duomenų rinkinys, bendras duomenų rinkinys. Naudojant asmeninį duomenų rinkinį kiekvienam asmeniui aptikimas vykdomas pagal jo istorinių transakcijų duomenis. Naudojant bendrą duomenų rinkinį vertinimai ir skaičiavimai atliekami visų istorinių transakcijų rinkinyje, jo negrupuojant. Asmeninis duomenų rinkinys leidžia atsižvelgti į asmeniui unikalius elgsenos šablonus, tačiau tyrimuose nustatyta, kad aptikimas pagal bendrą duomenų rinkinį tikslesnis [AS12]. Asmeniniai duomenų rinkiniai neturi statistiškai reikšmingų transakcijų kiekių. Lietuvos pilietis vidutiniškai per metus atlieka 71 atsiskaitymą kreditine ar debetine kortele [ESD16]. Todėl sukčiavimo aptikimui kriterijai bus apibrėžiami bendrame duomenų rinkinyje.

Duomenų normalizavimas apibrėžia transakcijos atributų transformavimą į iš anksto apibrėžiamą reikšmių intervalą. Duomenų kategorizavimas apibrėžia intervalo reikšmių suskirstymą į iš anksto apibrėžtas kategorijas, kurios bus naudojamos kaip kriterijų būsenos.

Tikslios skaitinės vertės įneša tik triukšmą, nedidina tikslumo. Suskirsčius duomenis į kategorijas galima sekti pokyčius tarp nominalių reikšmių ir gauti vienareikšmį rezultatą.

Išvestinių kriterijų agregavimas apjungia bazinių atributų reikšmes ir istorinius duomenis. Išvestiniai kriterijai leidžia atsižvelgti į papildomus elgsenos aspektus. Taip pat siekiant didesnio tikslumo įprasta agreguoti duomenis iš skirtingų laiko intervalų. Kadangi transakcija turi tik 6 bazinius atributus, todėl išvestiniai kriterijai turėtų padidinti aptikimo tikslumą.

Remiantis šiais metodais bus apibrėžti aptikimo kriterijai, įvertinti kriterijų duomenų tipai, parinkti metodai kriterijų reikšmių apskaičiavimui. Naudojantis baziniais transakcijos atributais galime įvertinti šias elgsenos savybes:

- transakcijos sumą;
- transakcijos laiką;
- transakcijos vietą;
- transakcijų kiekį.

2.3.3. Sumos rizikingumas

Poskyryje aptariami išvestiniai transakcijų kriterijai leidžiantys įvertinti sumos rizikingumą.

2.3.3.1. Periodo išlaidų suma

Pateikiamas kriterijus palyginantis pasirinkto laiko periodo išlaidų sumą su praeities periodų vidutinėmis išlaidomis. Kriterijus įvertina ar per šį periodą išleistų lėšų kiekis yra tikėtinas. Dažniausiai asmenys gauna fiksuotas periodines pajamas, todėl išlaidos tam tikrų periodų ribose turėtų būti nusistovėjusios.

Kadangi skaičiuojant sumą kalendoriniam mėnesiui ar savaitei, įmokos pirmoje mėnesio dalyje sudarytų tik mažą sumos dalį, suma būtų mažesnė nei viso periodo suma. Dėl to skaičiuojama slenkanti apibrėžto periodo suma. Savaitės trukmės periodo atveju – praėjusių 7 dienų įmokos. Mėnesio atveju – praėjusių 30 dienų įmokos.

Kriterijus apibrėžia transakcijų vertės sumą, tačiau tiesiogiai lyginti sumas neprasminga. Palyginimai naudojantys statines reikšmes nelankstūs, reikalauja ekspertinių žinių norint parinkti tinkamas palyginimo sąlygas, jas parinkus reikšmes reikia nuolat kalibruoti. Pranašesnis sprendimas būtų naudotis statistinius kriterijus priklausančius nuo duomenų rinkinio. Standartinis nuokrypis įvertina tikėtiną duomenų nuokrypį nuo vidurkio. Galima įvertį apibrėžti kaip periodo sumos įverčio ir periodų vidurkių skirtumo bei praeities periodų standartinio sumos nuokrypio santykį: $\frac{sum_{vid}-sum}{\sigma}$, čia, σ – standartinis nuokrypis, sum – periodo suma, sum_{vid} – vidutinė periodo suma.

Vieno asmens transakcijų kiekio gali nepakakti norint turėti statistiškai prasmingą transakcijų kiekį vienos dienos periode, todėl sumos turėtų būti skaičiuojamos ilgesniems nei diena periodams: savaitė, mėnuo.

2.3.3.2. Periodo išlaidų sumos dalis

Periodo išlaidų sumos dalies kriterijus įvertintų procentinę periodo sumos dalį, kurią sudaro transakcijos vertė periode. Šis įvertis gali būti naudingas, kadangi padėtų įvertinti pavienės transakcijos vertės tikėtinumą. Pavyzdžiui, lyg šiol periode visos transakcijų vertės buvo mažesnės nei tikėtina, tačiau paskutinės transakcijos vertė yra žymiai didesnė nei tikėtina. Tokiu atveju šio periodo išlaidų suma gali atrodyti normali, kadangi sudėjus mažesnes nei įprastai transakcijų vertes ir vieną didesnę gausime įprastą sumą. Tačiau sumos dalies kriterijus leistų identifikuoti papildomą riziką, nes viena transakcija sudaro didesnę nei įprastai periodo sumos dalį.

Kriterijaus reikšmė – procentinė išraiška pateikiama realiuoju skaičiumi. Galima teigti, kad procentinė išraiška yra pakankamai normalizuota, kadangi rezultatas visada priklauso reikšmių intervalui $[0; 100]$. Kaip ir prieš tai aptartų kriterijų atveju, statinės aprašytos kategorijos yra vengtinos, todėl siūloma skaičiuoti transakcijos procentinės išraiškos reikšmės ir periodo reikšmių vidurkio skirtumo ir standartinio nuokrypio nuo vidurkio santykį. $\frac{d_{vid}-d}{\sigma}$, Čia σ – standartinis nuokrypis, d – periodo išlaidų sumos dalis, d_{vid} – vidutinį periodo išlaidų dalis. Sumos dalis vertinama periodams: savaitė, mėnuo.

2.3.3.3. Didžiausia transakcijos vertė

Prieš tai aptarti sumos kriterijai vertina periodines išlaidas, tačiau nebūtina naudoti periodinius kriterijus. Apibrėžiamas kriterijus identifikuojantis transakcijas, kurios yra didesnės vertės nei kortelės savininkas yra atlikęs per visą naudojimosi kortele istoriją.

Kriterijus iliustruoja, kad galima naudoti ne tik skaitines reikšmes žyminčius kriterijus, tačiau ir dvejetainius kriterijus, kurių vertė – tiesa/melas. Tokie dvejetainiai kriterijai gali padėti identifikuoti išskirtinę ir asmeniui nebūdingą elgseną.

2.3.3.4. Praradimo vertė

Prieš tai apibrėžti transakcijos vertės kriterijai skirti identifikuoti nukrypimams nuo tikėtinos elgsenos. Sudarinėjant kriterijų rinkinį nebūtina rinktis kriterijų apibūdinančių kortelės savininko elgseną. Galima atsižvelgti į bet kokią potencialiai naudingą informaciją. Todėl siūlomas kriterijus leidžiantis atsižvelgti į transakcijos praradimo kainą, kurią reikės padengti. Tai gali būti naudinga, nes galimos situacijos, kai pagal asmens elgsenos kriterijus įmoka atrodo įtartina, tačiau įmoka yra mažos vertės ir neverta rizikuoti dėl neteisingų spėjimų.

Apibrėžiame fiksuotą skalę įvertinančią realią vertę. Transakcijas suskirstome į 4 grupes: labai mažos vertės transakcijos, mažos vertės transakcijos, didelės vertės transakcijos, labai didelės vertės transakcijos. Siūloma vengti statinių apibrėžtų kriterijų grupių, tačiau šio kriterijaus atveju sunku identifikuoti tinkamą atskaitos tašką, todėl apibrėžiamos statinės grupių ribos. Transakcija laikoma labai mažos vertės jei jos vertė mažesnė už 25 EUR. Transakciją laikome mažos vertės jei jos vertė didesnė arba lygi 25 EUR, tačiau mažesnė už 100 EUR. Transakciją laikome didelės vertės jei jos vertė didesnė arba lygi 100 EUR, tačiau mažesnė už 500 EUR. Transakciją laikome labai didelės vertės jei jos vertė didesnė arba lygi 500 EUR. Šios reikšmės realiame modelyje gali būti parenkamos ar keičiamos priklausomai nuo poreikių.

2.3.4. Kiekio rizikingumas

Apibrėžiami kriterijai susiję su transakcijų kiekiu. Vertės atveju buvo mėginama atsižvelgti į mokėjimų sumos amplitudę, skirstyti vertes į kategorijas. Transakcijų kiekis yra mažiau kintantis rodiklis, naudoti tuos pačius kriterijus neprasminga. Taip pat nėra prasmės skaičiuoti kiekio procentinės dalies, kadangi kiekis yra statinis – vertinimas visada atliekamas vienai transakcijai.

2.3.4.1. Periodo transakcijų kiekis

Kaip ir transakcijos vertės atveju, taip ir kiekiui galima įvertinti tikėtiną transakcijų kiekį per periodą. Kriterijus galėtų įvertinti ar praeityje, per tą patį laiko periodą, buvo atliktas panašus transakcijų kiekis. Transakcijų kiekis per periodą išreiškiamas natūraliuoju skaičiumi. Kadangi siekiama išvengti statinių palyginimų, vertinamas periodo transakcijų kiekio įverčio ir vidutinio praeities periodų transakcijų kiekio skirtumo ir standartinio nuokrypio santykis $\frac{kiek_{vid}-kiek}{\sigma}$, čia σ – standartinis nuokrypis, $kiek$ – periodo transakcijų kiekis, $kiek_{vid}$ – vidutinis periodo transakcijų kiekis.

Kiekis, taip pat kaip ir suma, skaičiuojamas kaip slenkančio periodo transakcijų kiekis. Kadangi transakcijų kiekis turi žymiai mažiau variacijos nei transakcijos suma, o asmuo per dieną atlieka greičiausiai iki kelių transakcijų, kiekio atveju taip pat vertinami ilgesni periodai: savaitė, mėnuo.

2.3.5. Laiko rizikingumas

Transakcijos laikas yra vienas iš bazinių atributų. Norint apibrėžti kriterijus, kaip pagrindą naudojančius transakcijos laiko atributą, aritmetinio vidurkio ar standartinio nuokrypio skaičiavimai netinkami, nes laikas matuojamas ciklinėje erdvėje. Pavyzdžiui, atlikus dvi transakcijas viena po kitos 4h laiko tarpu: viena atlikta 22h, kita 2h nakties, tuomet galima teigti, kad dažniausiai apsipirkinėjama naktį. Tačiau apskaičiavus aritmetinį vidurkį būtų gautas

vidurdienis. Šiuo atveju aritmetinis vidurkis neteikia vertingos informacijos ir klaidina. Todėl laiko kriterijų skaičiavimui bus naudojami alternatyvūs metodai.

2.3.5.1. Paros metas

Kadangi laikas matuojamas ciklinėje erdvėje ir kiekviena para turi baigtinį kiekį valandų, laikas yra savaime normalizuotas. Galima paprastai įvertinti paros metą suskirstant parą į baigtinį kiekį kategorijų. Laiką Suskirstome į 6 kategorijas trunkančias po 4h: $[0;4), [4;8), [8;12), [12;16), [16;20), [20;24)$. Toks laiko skirstymas padėtų identifikuoti paros metą kuriuo tikėtina sutikti sukčiavimą.

2.3.5.2. Laikas tarp transakcijų

Daroma prielaida, kad dažnesni nei įprastai apsipirkimai gali būti sukčiavimo identifikatorius. Todėl apibrėžiamas kriterijus vertinantis laiką praėjusį nuo paskutinės transakcijos. Tarkime, kad laikas tarp apsipirkimų matuojamas minutėmis. Kaip ir sumos bei kiekio atveju tiesiogiai lyginti minučių reikšmes sunku. Todėl siūloma vertinti laiką praėjusį nuo paskutinės transakcijos ir vidutinio laiko tarp transakcijų skirtumą ir standartinio nuokrypio santykį $\frac{t_{vid}-t}{\sigma}$, čia σ – standartinis nuokrypis, t – laikas praėjęs po paskutinės transakcijos, t_{vid} – vidutinis laikas tarp transakcijų. Tikėtina, kad turint pakankamą duomenų rinkinį vidurkis bus panašus skaičiuojant įvairiems periodams, todėl nėra prasmės atskirai vertinti šį kriterijų skirtingiems periodams.

2.3.5.3. Tikėtinas laikas tarp transakcijų

Buvo apibrėžtas kriterijus palyginantis vidutinį laiką tarp transakcijų praeityje, tačiau kaip ir kiekio bei vertės atvejais galima įvertinti tikėtino laiko tarp transakcijų kitimą. Atributo apibrėžimas analogiškas tikėtino kiekio kriterijui. Tiesinės regresijos pagalba randama regresijos lygtis pagal kurią identifikuojamas tikėtinas laikas. Įvertis išreiškiamas kaip laiko nuo paskutinės transakcijos ir tikėtino laiko skirtumas apskaičiuojant santykį su standartinės klaidos įverčiu: $\frac{t_{tik}-t}{\sigma}$, čia σ – standartinis nuokrypis, t – laikas praėjęs po paskutinės transakcijos, t_{tik} – tikėtinas vidutinis laikas tarp transakcijų.

2.3.6. Vietos rizikingumas

Transakcijos atlikimo vieta yra labai svarbus atributas galintis suteikti papildomų įžvalgų apie transakcijos riziką. Transakcijų vieta drastiškai keičiasi rečiau nei kiti atributai: atostogų metu, pakeitus darbą, persikrausčius. Vieta iš kitų atributų išsiskiria tuo, kad reikšmės kinta dvimatėje erdvėje. Todėl transakcijos vietos įvertinimas gali būti sudėtingas uždavinys. Reikia atsargiai parinkti stebimus kriterijus ir jų vertinimo metodus.

Skaičiavimų prasme vieta žemėlapyje ar šalyje aptikimo būdui nėra svarbi. Nėra prasmės naudoti žemėlapių abstrakcijas. Apytikslį atstumą galima apskaičiuoti pagal koordinačių pokytį

darant prielaidą, kad žemės rutulys yra ideali sfera su žinomu spinduliu. Taip pat galima pastebėti kad atstumas išreikštas ilgio vienetais taip pat nėra būtinas. Siekiant didesnio efektyvumo pakanka vertinti koordinačių pokytį. Iš geometrijos pagrindų žinoma, kad apskritimo lanko ilgis prieš kampą yra tiesiogiai proporcingas lanko kampui. Dėl to prasmingai galime lyginti atstumą be konvertavimo į ilgio vienetus.

2.3.6.1. Transakcijos vieta

Daroma prielaida, kad egzistuoja teritorijos, kuriose vyrauja aukštesni nusikalstamumo rodikliai. Todėl vertinama geografinė transakcijos atlikimo vieta tikintis nustatyti ar teritorija istoriškai rizikinga. Vertinti vietą pagal administracinius teritorinius vienetus pakankamai sudėtinga, o kaip buvo parodyta[8], aptikimo tikslams pakanka atstumą skaičiuoti koordinačių pokyčiu. Naudojant tokį atstumo matavimo metodą, vietoje administracinio skirstymo, galima žemės rutulį suskirstyti į taisyklingas teritorijas kurių kiekviena yra apribojama 4 erdvės taškais. Toks transakcijos teritorijos identifikavimas pakankamai efektyvus. Pakanka patikrinti kurios teritorijos koordinačių ribos apima transakcijos koordinates.

Toks kriterijaus apibrėžimas sukeltų efektyvumo problemų atliekant skaičiavimus tinkle. Žemės teritorija labai didelė. Skirstymas reikštų, kad tinkle kiekviena teritorija būtų prilyginama vienai kriterijaus būsenai. Iš efektyvumo tyrimo žinoma, kad didelis kriterijaus būsenų kiekis neigiamai įtakoja efektyvumą. Dėl to apibrėžiamas kriterijus įvertinantis teritorijos rizikingumą. Kiekvienai teritorijai skaičiuojamas tikimybinis įvertis sutikti sukčiavimą $P(SUK|TER)$. Šį įvertį apskaičiuoti galima kiekvienai teritorijai skaičiuojant teritorijoje atliktų sukčiavimo transakcijų ir visų transakcijų santykį. Kaip ir anksčiau aptartų kriterijų atveju šį santykį lyginti su statinėmis reikšmėmis nėra prasminga. Kiekvienai teritorijai galima priskirti įvertį, kuris apskaičiuojamas teritorijos tikimybės ir vidutinės tikimybės skirtumo santykiu su standartiniu nuokrypiu $\frac{p_{vid}-p}{\sigma}$, čia σ – standartinis nuokrypis, p – transakcijos teritorijos rizikingumo tikimybinis įvertis, p_{vid} – vidutinis teritorijų rizikingumo įvertis.

2.3.6.2. Atstumas nuo įprastinės teritorijos

Darant prielaidą, kad asmenys dažniausiai atlieka transakcijas toje pačioje teritorijoje, galima atsižvelgti į transakcijas atliktas už šios teritorijos ribų. Naudojantis transakcijos vietos atributu apskaičiuojamas atstumas nuo teritorijos kurioje dažniausiai vykdomos transakcijos. Kadangi pakanka išlaikyti dimensijų vientisumą tarp lyginamų įverčių, dėl efektyvumo atstumas pateikiamas skaičiuojant koordinačių pokytį koordinačių plokštumoje[4].

[8] Žiūrėti poskyrį 2.3.6 Vietos rizikingumas

Norint išvengti statinių palyginimų kriterijus vertinamas nuokrypio nuo atstumų vidurkio ir standartinio nuokrypio santykiu $\frac{s_{vid}-s}{\sigma}$, čia σ – standartinis nuokrypis, s – transakcijos atlikimo vietos atstumas iki įprastos teritorijos, s_{vid} – vidutinis transakcijos atlikimo vietos atstumas iki įprastos teritorijos, kai transakcija už įprastinės teritorijos ribų.

2.3.6.3. Atstumas nuo paskutinės transakcijos vietos

Daroma prielaidą, kad per trumpą laiko tarpą žymiai pasikeitusi transakcijos vieta gali būti rizikos identifikatorius. Apibrėžiamas kriterijus vertinantis atstumą iki paskutinės transakcijos vietos. Keliaujant atstumas nuo dažniausios vietos gali drastiškai pasikeisti, tačiau šis įvertis leis atsižvelgti į laipsnišką vietos kitimą. Kadangi vieta kinta dviejų dimensijų erdvėje identifikuoti tikėtiną reikšmę sunku, todėl tikėtinos vietos įvertinimas pakeičiamas atstumu iki paskutinės transakcijos vietos. Atstumas išreiškiamas koordinačių pokyčiu koordinačių plokštumoje. Įvertis išreiškiamas atstumo nuokrypio nuo atstumų vidurkio ir standartinio nuokrypio santykiu $\frac{s_{vid}-s}{\sigma}$, čia σ – standartinis nuokrypis, s – atstumas nuo transakcijos atlikimo vietos iki paskutinės transakcijos atlikimo vietos, s_{vid} – vidutinis atstumas nuo transakcijos atlikimo vietos iki paskutinės transakcijos atlikimo vietos.

2.3.6.4. Pardavėjas

Daroma prielaida, kad egzistuoja pardavėjai, kurie dėl parduodamų prekių ar kitų veiksnių yra patrauklesni sukčiams. Apibrėžti kriterijaus, kuris nurodo pardavėją negalima, kadangi kiekvienas pardavėjas būtų išreikštas atskira tinklo kriterijaus būsena. Iš efektyvumo tyrimo žinoma, kad didelis būsenų kiekis neigiamai veikia aptikimo efektyvumą.

Pardavėją identifikuojantis kriterijus pakeičiamas kriterijumi nurodančiu pardavėjo rizikingumą. Kiekvienam pardavėjui apskaičiuojama tikimybė sutikti sukčiavimo transakciją, tarp pardavėjo apdorotų transakcijų. Tikimybė išreiškiama sukčiavimo transakcijų skaičiaus ir visų transakcijų skaičiaus santykiu. Kriterijaus įvertis skaičiuojamas įvertinant nuokrypį nuo skirtingų pardavėjų sukčiavimo tikimybės vidurkio ir apskaičiuojant santykį su standartiniu nuokrypiu $\frac{p_{vid}-p}{\sigma}$, čia σ – standartinis nuokrypis, p – pardavėjo rizikingumo tikimybinis įvertis, p_{vid} – vidutinis pardavėjų rizikingumo įvertis.

2.3.7. Įverčių kategorijos

Daugelyje įverčių kriterijai buvo apibrėžti kaip nuokrypis nuo vidutinės reikšmės ir standartinio nuokrypio santykis. Šis santykis yra realusis skaičius, todėl reikia suskirstyti reikšmes į kategorijas siekiant baigtinio būsenų kiekio tinkle. Išskiriamos 5 kategorijos:

- daug mažesnė nei tikėtina reikšmė,
- mažesnė nei tikėtina reikšmė,
- tikėtina reikšmė,

- didesnė nei tikėtina reikšmė,
- daug didesnė nei tikėtina reikšmė.

Tegu santykis tarp nuokrypio nuo vidurkio ir standartinio nuokrypio žymimas: z. Tada reikšmė laikoma tikėtina, jeigu: $z \in [-1; 1]$. Reikšmė laikoma didesne nei tikėtina, jeigu $z \in (1; 2]$. Reikšmė laikoma daug didesne nei tikėtina, jeigu $z \in (2; +\infty)$. Analogiškai, reikšmė laikoma mažesne nei tikėtina, jeigu $z \in (-2; -1]$. Reikšmė laikoma daug mažesne nei tikėtina jeigu $z \in (-\infty; -2)$.

2.3.8. Konkretus aptikimo tinklas

Sugrupavus pateikiamus kriterijus ir jais užpildžius ištirtą dviejų lygių tinklo struktūrą gaunamas konkretus aptikimo tinklas (žr. 4 priedas). Tinkle siūloma apibrėžti 4 kriterijų grupes: transakcijos vertės rizikingumas, transakcijų kiekio rizikingumas, transakcijos laiko rizikingumas, transakcijos vietos rizikingumas. Toliau darbe bus analizuojamas tinklo įgyvendinimas sistemos pavidalu.

2.3.9. Apibendrinimas

Tiriant aptikimo kriterijus buvo pasiūlyti būdai tinkle įvertinti transakcijos sumos, transakcijos laiko, transakcijų kiekio bei transakcijų vietos rizikingumą. Atlikus bazinių transakcijos atributų analizę, naudojant duomenų paruošimo procesus, buvo apibrėžta 14 išvestinių transakcijos kriterijų padedančių įvertinti asmens elgseną. Kiekvienam iš kriterijų buvo apibrėžtas būdas normalizuoti reikšmes bei suskirstyti jas į kategorijas tinkamas išreikšti tinklo būsenomis. Skirstymas į kategorijas buvo apibrėžtas naudojantis baziniais statistikos metodais siekiant sumažinti tinklo priklausomybę nuo ekspertinių žinių. Norint užtikrinti didesnį tikslumą ir būdo panaudojamumą, dalis kriterijų apibrėžta skirtingiems transakcijos periodams, taip padidinant aptikimo kriterijų kiekį. Naudojant apibrėžtus transakcijų kriterijus pasiūlytas sukčiavimo aptikimo tinklas naudojantis identifikuotus kriterijus pateikiamus dviejų lygių Bajeso tinklo pavidale.

Transakcijų duomenų paruošimo procesų pritaikymas negarantuoja maksimalaus sukčiavimo aptikimo tikslumo. Vis dėlto, tai leidžia sudaryti sukčiavimo aptikimo būdą, palyginamą su realioje aplinkoje naudojamais būdais siekiant ištirti siūlomo būdo efektyvumą bei užtikrinti tinklo panaudojamumą. Apibendrinant sukčiavimo aptikimo kriterijų analizę galima teigti, kad nepaisant nedidelio bazinių transakcijos atributų skaičiaus, naudojant statistinius metodus, duomenų agregavimą į išvestinius kriterijus, reikšmių normalizavimą bei skirstymą į kategorijas, galima apibrėžti kriterijų rinkinį galintį įvertinti transakciją atlikusio žmogaus elgseną bei transakcijos rizikingumą.

2.4. Sukčiavimo aptikimo sistema

Buvo ištirti tinklo įgyvendinimo būdai, identifikuoti aptikimui tinkami kriterijai. Atlikus tinklo efektyvumo tyrimą buvo identifikuotas tikimybinis tinklas, kurio efektyvumas leistų įgyvendinti sukčiavimo aptikimo būdą galintį pateikti atsaką realiu laiku. Vis dėlto, įgyvendinamumą bei efektyvumą įtakoja ne tik algoritmo kompleksiškumas. Siekiant paprastumo efektyvumo analizėje visos operacijos buvo laikomos lygiavertėmis. Pateikti efektyvumo vertinimai padeda įsitikinti ar algoritmas yra įgyvendinamas teorine prasme, įvertinti algoritmo panaudojimo galimybes didėjant transakcijų ar kriterijų kiekiui. Tačiau skirtingų matematinių operacijų, duomenų manipuliacijų sudėtingumas skiriasi. Kadangi sistemos naudoja daugybę abstrakcijos sluoksnių, įvertinti kelių sistemų kontekstą dar sudėtingiau. Abstrakcijos slepia ne tik įprastus aritmetinius veiksmus, tačiau ir manipuliavimą mašinos ar tinklo resursais. Dėl bendravimo tinkle problemų sudėtingumas ypač sunkiai vertinamas. Tinklas yra nepatikimas, neturi begalinio pralaidumo. Dėl to, norint užtikrinti metodo įgyvendinamumą ir efektyvumą, šiame skyriuje aptariami aptikimo sistemos reikalavimai, siūlomas techninis sprendimas sistemos įgyvendinimui.

2.4.1. Reikalavimai sistemai

Norint atlikti programų sistemos architektūrinių sprendimų analizę reikia apibrėžti reikalavimus sistemai. Sistema turi 3 esminius funkcinius reikalavimus:

- apskaičiuoti ir pateikti įvertį nurodantį tikimybę, kad transakcija yra sukčiavimas, pagal transakcijų statistinius duomenis ir tikimybinį tinklą;
- atnaujinti įverčio skaičiavimui reikalingus duomenis naudojant įvertintų transakcijų duomenis;
- atnaujinti įverčio skaičiavimui reikalingus duomenis naudojant istorinius transakcijų duomenis.

Tokio tipo sistemai veikiančiai bankinėje aplinkoje reikėtų apibrėžti daugybę nefunkcinių reikalavimų susijusių su sistemos saugumu, privačių duomenų saugą ir kt. Kadangi siekiama ištirti sukčiavimo aptikimo efektyvumą ir pasiūlyti efektyvų aptikimo būdą, atsižvelgiama tik į nefunkcinius reikalavimus įtakojančius galimybę pateikti atsaką realiu laiku. Apibrėžiami esminiai nefunkciniai reikalavimai:

- gali pateikti sukčiavimo tikimybinį įvertį greičiau nei per 1 sekundę;
- gali įvertinti daugiau nei 10 000 transakcijų per sekundę[9];
- apmokymas neturi neigiamai veikti sukčiavimo įverčio pateikimo atsako laiko.

[9] Visa transakcijų apdorojimas 2011 gruodžio 23d pasiekė 11000 transakcijų per sekundę [Tril11]

2.4.2. Sistemos funkcijos

Atsižvelgiant į sistemos funkcinius reikalavimus apibrėžiamos sukčiavimo aptikimo sistemos funkcijos:

- FN – 1 priimti transakcijos atributus apdorojimui;
- FN – 2 identifikuoti kriterijų būsenas pagal transakcijos atributus;
- FN – 3 apskaičiuoti sukčiavimo tikimybinį įvertį;
- FN – 4 pateikti įvertį HTTP protokolu;
- FN – 5 atnaujinti tikimybinį tinklą;
- FN – 6 atnaujinti transakcijų statistinius duomenis;
- FN – 7 priimti istorines transakcijas apdorojimui;
- FN – 8 pažymėti apdorotą transakciją kaip sukčiavimas.

Pirmoji funkcija užtikrina, kad sistema galės priimti transakcijos duomenis apdorojimui iš kitų sistemų taip užtikrinant galimybę integruoti sistemą su transakcijas apdorojančiomis sistemomis. Antroji funkcija užtikrina, kad naudojant vertinamos transakcijos atributus ir statistinius transakcijų duomenis bus įvertintos apsibrėžtų sukčiavimo kriterijų reikšmės. Trečioji funkcija apibrėžia, kad iš antrojoje funkcijoje gautų kriterijų reikšmių tikimybinio tinklo pagalba bus apskaičiuojamas sukčiavimo tikimybinis įvertis. Ketvirtoji funkcija užtikrina, kad transakciją įvertinimui pateikusi sistema galės gauti sukčiavimo įverti HTTP protokolu. Penktoji funkcija apibrėžia, kad transakcijų sukčiavimo kriterijų reikšmės bus naudojamos atnaujinti tikimybinį tinklą, nuolatos užtikrinant sistemos prisitaikymą prie elgsenos pokyčių. Šeštoji funkcija nurodo, kad įvertintų transakcijų duomenys bus naudojami atnaujinti transakcijų statistiniams duomenims. Septintoji ir aštuntoji funkcijos užtikrina sistemos pritaikymo realioje aplinkoje galimybes. Sukčiavimo aptikimo sistema pateiks pakankamo tikslumo rezultatus tik apdorojusi statistiškai reikšmingą transakcijų kiekį, todėl po pradinio sistemos paleidimo sistemą būtina užpildyti transakcijų duomenimis. Tokiu būdu atnaujinami statistiniai transakcijų bei tikimybinio tinklo duomenys. Tačiau sukčiavimo transakcijos yra tiksliai identifikuojamos tik tada, kai kortelės savininkai praneša apie įvykusius sukčiavimo atvejus. Sukčiavimo aptikimo sistema naudoja apdorotas transakcijas statistinių duomenų atnaujinimui. Šio atnaujinimo metu daroma prielaida, kad transakcijos nėra sukčiavimas. Todėl aštuntoji funkcija apibrėžia galimybę praeityje apdorotas transakcijas pažymėti kaip sukčiavimą. Jau apdorotų transakcijų žymėjimas sukčiavimu turėtų inicijuoti transakcijų statistikos atnaujinimo ir tikimybinio tinklo duomenų atnaujinimo procesus.

Apibrėžtos sistemos funkcijos skirstomos į sistemos loginius komponentus:

- K – 1 duomenų apsikeitimo komponentas;

- K – 2 įverčio skaičiavimo komponentas;
- K – 3 tikimybinio tinklo saugykla;
- K – 4 transakcijų duomenų saugykla.

Siekiant užtikrinti visų funkcijų įgyvendinimą bei apibrėžti komponentų funkcines atsakomybes sudaryta funkcijų įgyvendinimo atsekamumo matrica (žr. 3 lentelė).

3 lentelė. Funkcijų įgyvendinimo atsekamumo matrica

	FN – 1	FN – 2	FN – 3	FN – 4	FN – 5	FN – 6	FN – 7	FN – 8
K – 1	X			X			X	X
K – 2		X	X					X
K – 3					X			X
K – 4						X		X

Atsekamumo matricoje apibrėžiama, kad duomenų apsikeitimo komponentas atsakingas už integracijas su išorinėmis sistemomis. Komponentas realizuos programinę duomenų apsikeitimo sąsają HTTP protokolu, užtikrins protokolo sesijos valdymą. Įgyvendinamos sąsajos: transakcijos pateiktos vertinimui priėmimas, įverčio gražinimas, istorinių duomenų priėjimas, nustatytų sukčiavimo atvejų priėmimas. Gavęs užklausą komponentas konvertuoja duomenis iš išorinėms sąsajoms naudojamo protokolo duomenų formato į vidinių sistemos komponentų integracijai naudojamą duomenų apsikeitimo formatą. Perduoda konvertuotus duomenis tolimesniam apdorojimui kituose komponentuose.

Įverčio skaičiavimo komponentas gauna transakcijos duomenis ir įvertina sukčiavimo kriterijų vertes atsižvelgdamas į statistinius transakcijų duomenis gautus iš transakcijų duomenų saugyklos komponento. Šių kriterijų vertės naudojamos gauti įvykių tikimybes iš tikimybinio tinklo saugyklos komponento. Gautos tikimybės naudojamos įvertinti kriterijų kombinacijos rizikingumą lyginant su visomis galimomis kombinacijomis. Naudojant kriterijų grupių rizikingumo rodiklius apskaičiuojamas sukčiavimo įvertis ir gražinamas duomenų apsikeitimo komponentui. Tarpiniai skaičiavimų rezultatai perduodami tikimybinio tinklo saugyklos ir transakcijų duomenų saugyklos komponentams. Gavus prašymą apdoroti istorinę transakciją taip pat reikia įvertinti transakcijos kriterijus, kriterijų grupių rizikos rodiklius. Istorinės transakcijos apdorojimo rezultatai taip pat perduodami tikimybinio tinklo saugyklos ir transakcijų duomenų saugyklos komponentams. Gavus prašymą pažymėti praeityje apdorotą transakciją sukčiavimu, reikia identifikuoti pirminio apdorojimo tarpinius rezultatus ir kartu su papildomu sukčiavimo požymiu perduoti juos saugyklų komponentams.

Tikimybinio tinklo saugyklos komponentas įgyvendina įverčio skaičiavimui reikalingų duomenų paruošimo procesą. Šis procesas aprašytas įverčio skaičiavimo analizės poskyryje (žr. 2.2.1.3 poskyris). Komponentas taip pat užtikrina saugomų duomenų pateikimą įverčio

skaičiavimo komponentui. Transakcijų duomenų saugyklos komponentas renka ir agreguoja statistinius duomenis reikalingus sukčiavimo kriterijų įvertinimui, užtikrina šių duomenų pateikimą sukčiavimo įverčio komponentui. Saugyklų komponentų požiūriu istorinės transakcijos ir einamosios transakcijos yra identiškos. Abejais atvejais vykdomas tas pats duomenų paruošimo procesas. Gavę užklausą pažymėti praeityje jau apdorotą transakciją sukčiavimu šie komponentai turi gebėti perskaičiuoti reikiamus statistinius rodiklius.

2.4.3. Komponentų sąsajos

Reikalingų sąsajų kiekis tarp komponentų nėra didelis. Tačiau netinkamai įgyvendintos sąsajos apsunkina nefunkcinių sistemos reikalavimų įgyvendinimą. Todėl aptariami galimi komponentų integracijos variantai.

Sąsajas tarp komponentų galima suskirstyti į dvi pagrindines grupes: sinchroninė sąsaja, asinchroninė sąsaja. Sinchroninės sąsajos atveju perdavus žinutę pradedamas atsako iš žinutę gavusio komponento laukimas. Asinchroninė sąsaja turi esminį skirtumą, kad rezultatai nėra sinchronizuojami, žinutės perdavimą inicijavęs komponentas tęsia proceso vykdymą.

2.4.3.1. Sinchroninis įgyvendinimas

Integruojant sistemų komponentus dažniausiai naudojama sinchroninė sąsaja. Ši sąsaja lengvai įgyvendinama, tačiau turi trūkumų. Žinutės perdavimą inicijavęs komponentas blokuoja proceso vykdymą iki kol bus sulaukta atsako. Siekiant vykdyti daug operacijų vienu metu tokia sistemos realizacija laikoma neefektyvia. Kiekvienam blokuojamam procesui yra išskiriami sisteminiai resursai, reikalingas papildomas laikas procesų kontekstų keitimui.

Sukčiavimo aptikimo įgyvendinimas naudojant sinchroninę sąsają tarp komponentų, atvaizduojamas sekų diagrama (žr. 5 priedas). Naudojant šį komponentų integracijos būdą egzistuoja tiesioginė sukčiavimo įverčio skaičiavimo laiko priklausomybė nuo transakcijų duomenų paruošimo. Kiekvienai transakcijai turi būti atliktas tinklo tikimybių bei transakcijų statistikos atnaujinimas saugykloje. Bajeso tinklo analizės metu nustatyta, kad duomenų paruošimo procesas, lyginant su įverčio skaičiavimu, yra daug sudėtingesnis operacijų atžvilgiu. Todėl duomenų atnaujinimo laikas neigiamai veikia sukčiavimo įverčio pateikimo laiką. Komponentui perduodančiam pranešimą į mažesnį pralaidumą užtikrinantį komponentą tektų laukti kol užklausos bus apdorotos. Galima teigti, kad įgyvendinus sukčiavimo aptikimą naudojant sinchroninę sąsają tarp komponentų, visos integruotų komponentų grandinės pralaidumas būtų lygus mažiausią pralaidumą užtikrinančio komponento pralaidumui.

Sinchroninė sąsaja tarp komponentų taip pat didina sistemos nestabilumo riziką. Jeigu naudojant sinchroninę sąsają komponentui nuolat perduodama daugiau užklausų nei komponentas gali apdoroti, tai gali iššaukti sistemos neveiksnumą. Tokiu atveju užklausas inicijuojančiame komponente blokuojasi laukiantys procesai išnaudodami papildomus

komponento resursus. Žymiai peržengus sistemos pralaidumo ribas kyla rizika, kad užklausas inicijuojanti sistema taip pat taps neveiksni. Galima teigti, kad viršijus sistemos pralaidumą kyla didelė incidentų tikimybė silpniausioje sistemos grandyje.

Apibendrinant galima teigti, kad sinchroninė sąsaja tarp komponentų yra lengvai įgyvendinama ir dažnai pakankama. Tačiau griežtus nefunkcinius reikalavimus turinčiose sistemose toks įgyvendinimas ne visada tinkamas. Sukčiavimo aptikimo sistemos atveju šio tipo sąsajos negalima naudoti inicijuojant duomenų atnaujinimo procesus, kadangi tai neigiamai paveiktų įverčio skaičiavimo atsako laiką.

2.4.3.2. Asinchroninis įgyvendinimas

Sąsajos asinchroniškumas gali būti įgyvendinamas įvairiuose architektūros hierarchijos lygiuose. Loginių komponentų integracijos lygmenyje asinchroniškumą galima užtikrinti įgyvendinant skelbėjo-prenumeruotojo (angl. *publish-subscribe*) principą. Pavyzdžiui įgyvendinant duomenų perdavimą eile. Naudojant eilės mechanizmą žinutės padėjimo į eilę sąsaja ir išėmimo iš eilės sąsaja gali būti sinchroninė, tačiau loginių komponentų atžvilgiu ši integracija vis tiek bus asinchroninė. Šiuo atveju sinchroninio atsako laukiama tik iš sąlyginai nesudėtingos operacijos.

Asinchroninę sąsają taip pat galima įgyvendinti, naudojant programavimo kalbų kalbinius primityvus. Šiuo atveju sąsaja loginių komponentų atžvilgiu atrodo sinchroninė, kadangi žinutės perdavimą iniciavęs komponentas atsako laukia. Vis dėlto, kalbinių primityvų pagalba šis laukimas gali būti iškeltas į atskirą procesą neblokuojantį pagrindinio proceso vykdymo, ar net išvengti proceso blokavimo. Tokiu atveju gavus atsaką vykdymas pratęsiamas jau naujame procese nenaudojant būsenos sinchronizavimo tarp procesų.

Vis dėlto, asinchroniškumo įgyvendinimas kalbiniais primityvais neišsprendžia sistemos stabilumo rizikų. Komponentui pateikus didesnį žinučių kiekį nei žinutes gaunantis komponentas gali apdoroti, išlieka rizika, kad žinutes gaunantis komponentas taps neveiksnus ir tai sukels klaidas inicijuojant žinučių perdavimą. Šiuo atveju žinučių perdavimą iniciuojanti sistema yra atspari visiškam neveiksnumui. Blogiausiu atveju klaidos bus gaunamos konkrečios žinutės iniciavimo kontekste. Nesikaups procesai laukiantys rezultato. Todėl žinutes inicijuojančios sistemos stabilumas neturėtų būti pažeistas.

Sistemos nelankstumo didėjančiam žinučių kiekiui rizika sistemose sprendžiama dviem pagrindiniais būdais:

- saugikliais (angl. *circuit breakers*);
- integracija naudojant eiles.

Saugikliai apibrėžia apsaugines taisykles, pagal kurias yra nutraukiama integracijos grandinė užtikrinant likusios sistemos gyvybingumą. Gavus klaidas iš sistemos ar nepavykus

pateikti užklausos sistemai per apsibrėžtą laiką (angl. *timeout*) žinutės perdavimas nutraukiamas. Šis mechanizmas leidžia išsaugoti sistemų gyvybingumą. Tačiau kadangi žinutės perduoti nepavyksta, sukčiavimo aptikimo atveju gauti transakcijos duomenys būtų prarasti. Norint sumažinti riziką galima įgyvendinti pakartotinių komandos perdavimo bandymų kiekį. Žinutės pakartojimas sumažina riziką, tačiau nesuteikia garantijos. Siekiant neprarasti transakcijų ir išlaikyti sistemos gyvybingumą reikalingas atsarginis mechanizmas saugantis transakcijas. Tokiu atveju šias transakcijas būtų galima perduoti po neveiksnaus proceso atstatymo. Tačiau tokiu atveju reikalingas papildomo žinučių saugojimo proceso įgyvendinimas.

Dėl šių priežasčių integracija naudojant žinučių eiles (angl. *message queue*) yra pranašesnis sprendimas. Eilės mechanizmas šias problemas išsprendžia savaime. Žinutės dedamos į eilę, o apdorojantis komponentas iš eilės ima po tiek žinučių, kiek gali apdoroti. Staiga padidėjus transakcijų kiekiui, padidėja eilėje laukiančiųjų transakcijų kiekis. Šiuo atveju žinutes apdorojantis komponentas apdoroja transakcijas įprastu tempu. Dėl to išauga ilgiau eilėje laukusių žinučių apdorojimo laikas, tačiau išvengiama sistemos stabilumo rizikų. Taip pat žinučių saugojimas eilėje yra efektyvesnis nei procesų blokavimas. Žinučių eilė yra duomenų saugykla optimizuota dideliam įrašų kiekių saugojimui, užtikrinanti išliekamumą (angl. *durability*). Vis dėlto, naudojant eiles svarbu tinkamai apskaičiuoti apdorojamų žinučių kiekį per laiko vienetą. Paskyrus nepakankamą resursų kiekį žinučių apdorojimui eilė niekada nesumažės, o transakcijų apdorojimo laikas nuolatos augs.

2.4.3.3. Siūloma komponentų integracija

Buvo trumpai pristatyti galimi sistemos komponentų integracijos metodai. Negalima teigti, kad vienas metodas yra pranašesnis už kitus ir turėtų būti naudojamas visoms sistemos sąsajoms. Visi sistemų integravimo būdai turi savo privalumus ir trūkumus. Sukčiavimo aptikimo sistemos realizacijoje siūloma apjungti skirtingus integracijos būdus tarp komponentų (žr. 6 priedas).

Buvo identifikuota, kad sukčiavimo aptikimo sistemos kontekste duomenų atnaujinimo funkcijos sudėtingiausios operacijų atžvilgiu, todėl sinchroninis duomenų perdavimas apdorojimui netinkamas. Duomenų atnaujinimo komponentai turėtų būti inicijuojami asinchroniškai, naudojant žinučių eiles. Šis sprendimas leistų išvengti visų sistemos stabilumo rizikų ir užtikrinti, kad duomenų atnaujinimas nepaveiktų įverčio pateikimo laiko ar įverčio skaičiavimo komponento pralaidumo. Šis duomenų perdavimo būdas atnaujinimo komponentams turi ir neigiamą pusę. Inicijavus atnaujinimo funkcijas nėra laukiama atnaujinimo rezultato todėl sukčiavimo įvertis bus skaičiuojamas pagal nenaujausią duomenų rinkinį. Vis dėlto, siekiant tikslaus sukčiavimo aptikimo apdorotų transakcijų rinkinys turi būti pakankamai

didelis. Todėl galima teigti, kad sukčiavimo įverčio skaičiavimui naudojamame duomenų rinkinyje kelios trūkstamos transakcijos neturės esminės įtakos įverčio tikslumui.

Likusios komponentų sąsajos reikalingos sukčiavimo įverčio skaičiavimo metu. Iš šių komponentų atsaką gauti būtina, kadangi rezultatai tiesiogiai naudojami skaičiavimuose. Loginių komponentų sąsajų prasme šios integracijos sinchroninės. Tačiau, sinchroninės integracijos atveju nėra efektyviai išnaudojami sistemos resursai. Taip ribojamas sistemos pralaidumas. Dėl to likusias sąsajas tarp sistemos komponentų sistemoje reikėtų įgyvendinti naudojant asinchroninę sąsają realizuotą kalbiniais primityvais.

2.4.4. Komponentai

Buvo aptarti komponentų integracijos variantai, tačiau norint įgyvendinti nefunkcinius reikalavimus svarbu atsižvelgti ir į komponentų realizaciją. Pateikiama detalizuota komponentų diagrama (žr. 7 priedas).

2.4.4.1. Duomenų apsikeitimo komponentas

Duomenų apsikeitimo komponentas įgyvendina protokolo lygio integraciją su išorinėmis sistemomis. Gavęs užklausas komponentas apdoroja gautus duomenis. Iš gauto duomenų formato konvertuoja duomenis į vidinėms sistemų integracijoms naudojamą duomenų formatą. Reikiamu formatu duomenis perduoda įverčio skaičiavimo komponentui. Dėl populiarumo bei paplitimo išorinėms sistemos integracijoms pasirinktas HTTP protokolas, kuris leistų užtikrinti maksimalias pritaikymo galimybes. Todėl papildoma šio komponento atsakomybė – HTTP protokolo sesijos valdymas.

Iš komponentų diagramos (žr. 7 priedas) matoma, kad duomenų apsikeitimo komponentas pateikia 2 programines sąsajas HTTP protokolu: sąsają sukčiavimo įvertinimo prašymui bei sąsają transakcijos pažymėjimui sukčiavimu. Sistemos funkcijose buvo apibrėžta, kad sistema turi gebėti apdoroti istorines transakcijas. Joms atskiras transakcijų apdorojimo procesas nebuvo įgyvendinamas siekiant išlaikyti sistemos paprastumą. Istorines transakcijas perduoti sistemai galima naudojant įprastinį įverčio sukčiavimo procesą. Šis procesas taip pat inicijuoja visas reikiamas duomenų paruošimo funkcijas.

Duomenų apsikeitimo komponentą siūloma realizuoti jį sudarant iš 3 vidinių komponentų: prašymų valdymo komponento, sukčiavimo įvertinimo prašymo valdymo komponento bei sukčiavimo žymėjimo prašymo valdymo komponento. Prašymų valdymo komponentas užtikrintų HTTP protokolo sąsajos bei sesijos valdymą. Šis komponentas būtų agreguojamas iš dviejų papildomų komponentų užtikrinančių skirtingų protokolo žinučių konvertavimą į įverčio skaičiavimo komponento pateikiamas sąsajas.

Duomenų apsikeitimo komponento atskyrimas sistemai suteikia atskirtą sistemos fasadą (angl. *facade design pattern*). Šis sprendimas pagerina sukčiavimo aptikimo sistemos pritaikymo

realioje aplinkoje galimybes. Kilus poreikiui sistemą integruoti kitais protokolais pakaktų vietoje duomenų apsikeitimo komponento parašyti naują protokolo adapterį. Tai leistų protokolo pakeitimus izoliuoti viename komponente nepaveikiant likusių sistemos funkcinių komponentų.

HTTP protokolas reikalauja sesijos saugojimo, yra sinchroninis. Dėl to, sunku užtikrinti komponento efektyvumą ir stabilumą apdorojant didelį vienu metu pateikiamų užklausų kiekį. Duomenų apsikeitimo komponentas nėra atsparus rizikoms aptartoms komponentų sąsajos poskyryje. Vis dėlto, komponento atskyrimas padeda dalinai sumažinti šias rizikas. Realiame sistemos diegime (angl. *deployment*) atskyrus šį komponentą į atskirą fizinę mašiną sistemos neveiksnumas būtų izoliuojamas išsaugant likusios sistemos gyvybingumą. Duomenų apsikeitimo komponentas kitų sistemų atžvilgiu yra HTTP serveris koordinuojantis veiksmus tarp kitų funkcinių komponentų. Vietoje įprasto sinchroninio HTTP serverio galima naudoti asinchronines HTTP serverio realizacijas. Pavyzdžiui realizacijas įgyvendinančias aktorių modelį (angl. *actor model*) ar kitą reaktyvių sistemų įgyvendinimo būdą. Toks sistemos įgyvendinimas leistų sukurti komponentą kuris gali užtikrinti didesnį pralaidumą. Svarbu paminėti, kad palyginus su įverčio skaičiavimo komponentu šis komponentas atlieka labai ribotą funkcijų kiekį. Galima teigti, kad problemos dėl efektyvumo pirmiausia turėtų kilti įverčio skaičiavimo komponente.

2.4.4.2. Įverčio skaičiavimo komponentas

Įverčio skaičiavimo komponento pagrindinė atsakomybė apskaičiuoti sukčiavimo tikimybę. Siekiant galutinio rezultato reikia įvertinti kriterijų vertes, kriterijų grupių vertes ir kt.

2.4.4.2.1. Įgyvendinami skaičiavimai

Analizuojant tinklo pavidalą buvo apibrėžti du būdai sukčiavimo įverčio apskaičiavimui. Buvo prieita prie išvados, kad sukonkretinus kriterijų grupes gaunamas žymiai efektyvesnis tinklas. Todėl toliau analizuojama šio sukčiavimo aptikimo tinklo (žr. 2.2.1.2 poskyris) realizacija.

Norint įgyvendinti sukčiavimo įverčio gavimą pagal sukonkretintą tinklą reikia užtikrinti algoritmo iš keturių žingsnių įgyvendinimą:

1. Įvertinti kriterijų apibrėžtų šeštajame skyriuje vertes.
2. Naudojant gautas kriterijų vertes apskaičiuoti išraiškos: $p_i = P\left(SUK \cap K_{i1} \cap \ldots \cap K_{i|KG_i|}\right) = P(SUK) \cap P(K_{i1}) \cap \ldots \cap P(K_{i|KG_i|})$ reikšmę kiekvienai kriterijų grupei ir pagal 5.1.1 poskyryje pasiūlytą kriterijų grupės apibrėžimą identifikuoti kriterijų grupės rizikingumą.
3. Pagal lygybę $P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k) = P(SUK) \cdot \prod_{i=1}^{k} P(KG_i|SUK)$ (žr. 2 priedas) apskaičiuoti dedamąsias: $P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)$, $P(\overline{SUK} \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)$.

4. Naudojant gautas dedamąsias apskaičiuoti galutinį rezultatą, sukčiavimo tikimybę: $P(SUK|KG_1 \cap KG_2 \cap \ldots \cap KG_k) = \frac{P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)}{P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k) + P(\overline{SUK} \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)}$

2.4.4.2.2. Siūloma realizacija

Pirmasis algoritmo žingsnis apibrėžia kriterijų reikšmių įvertinimą pagal transakcijos duomenis. Šis procesas vyksta kriterijų vertinimo komponente (žr. 7 priedas). Siekiant užtikrinti galimybes lengvai pridėti naujus sukčiavimo kriterijus pateikiama detali kriterijų vertinimo posistemės realizacija (žr. 8-12 priedai).

Kriterijus sistemoje siūloma realizuoti kaip abstrakciją galinčią pateikti įvertinimo rezultatą pagal transakcija ir istorinius duomenis. Kadangi dalis kriterijų periodiniai, papildomai kriterijų hierarchijoje įvedamas periodinis kriterijus Jis apibrėžiamas nurodytam periodo ilgiui dienomis. Pagrindinė kriterijaus realizacija – pavadinamas kriterijus komponuojamas iš dviejų esminių dalių: operacijos duomenų konverterio bei kriterijų vertinimo realizacijos (žr. 8 priedas). Operacijos duomenų konverterio atsakomybė iš transakcijos duomenų bei istorinių duomenų išgauti duomenis reikalingus konkrečiai kriterijų vertinimo realizacijai. Apibrėžiamos 3 konkrečios duomenų vertinimui naudojamos klasės: nuokrypio duomenys, palyginama statistika, transakcija (žr. 11 priedas). Kiekviena konkreti kriterijaus vertinimo realizacija geba apdoroti vieną iš šių konkrečių tipų ir pateikia spausdinamą rezultatą. Konkreti rezultato realizacija taip pat priklauso nuo kriterijaus skaičiavimo būdo (žr. 10 priedas). Kriterijai apibrėžiami naudojant standartinių nuokrypių santykį, gražina nuokrypio tikėtinumo rezultato realizaciją. Kriterijai vertinantys sąlyginius sakinius ir gražinantys dvejetainius rezultatus aibėje tiesa, netiesa gražina dvejetainio rezultato realizaciją. Ši kompozicija leidžia nesunkiai apibrėžti papildomus kriterijus ir išvengti pasikartojančių skaičiavimų dubliavimo. Apibrėžiant kiekvieną kriterijų nurodoma duomenų konverterio realizacija. Ji iš transakcijos ir istorinių duomenų išgauna reikiamus duomenis ir perduoda juos pasirinktai vertinimo realizacijai.

Ši kompozicija apibrėžta, nes analizuojant kriterijus pastebėta, kad daugumai kriterijų atliekami tie patys skaičiavimai: dviejų reikšmių dydžio palyginimas, standartinio nuokrypio santykio skaičiavimas. Vis dėlto, kiekvienu atveju buvo lyginami vis kiti duomenys. Dėl šios priežasties kiekvienam kriterijui yra apibrėžiama po unikalų duomenų konverterį (žr. 11 – 12 priedai). Todėl norint pridėti naują kriterijų sistemoje pakanka užtikrinti, kad transakcijų duomenų saugykloje saugomi kriterijui apskaičiuoti reikalingi duomenys bei aprašytas konverteris iš šių duomenų gaunantis kriterijaus įvertinimui svarbius atributus. Tokiu principu kriterijų konfigūracijos komponente apibrėžiami visi kriterijai.

Prieš skaičiuojant kriterijų vertes iš transakcijų duomenų saugyklos reikia gauti istorinius duomenis apie transakcijas. Šiuos duomenis pateikia transakcijų duomenų saugykla per „gauti transakcijų duomenis“ programinę sąsają. Gaunami trijų tipų statistiniai duomenys: asmeninė

praeities transakcijų statistika, globali transakcijų statistika, išorinių veiksnių (laiko, vietos, pardavėjo) rizikos statistika. Naudojant šiuos duomenis įvertinamas kiekvienas kriterijus.

Sekantis proceso žingsnis – kriterijų grupių įvertinimas. Tam, kad apskaičiuoti išraišką: $P(SUK) \cap P(K_{i1}) \cap \ldots \cap P(K_{i|KG_i|})$ reikalinga tikimybė sutikti kiekvieną iš kriterijų bei bendra sukčiavimo tikimybė. Įverčio skaičiavimo komponentas kreipiasi į tikimybinių duomenų saugyklą naudojant programinę sąsają „gauti tikimybinius duomenis“. Gavus atsaką kriterijų grupių vertinimo komponente atliekami daugybos veiksmai, įvertinama kiekvienos kriterijų grupės rizika.

Pagal gautas rizikas apskaičiuojamos dedamosios: $P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)$, $P(\overline{SUK} \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)$. Dedamosioms apskaičiuoti reikalingos kiekvienos kriterijų grupės pasitaikymo sukčiavimo transakcijoje ir nesukčiavimo transakcijoje tikimybes. Kadangi iš viso buvo apibrėžta 4 kriterijų grupės ir kiekviena gali turėti 5 galimas vertes, iš viso yra $2 \cdot 4 \cdot 5 = 40$ galimų tikimybinių reikšmių. Šių reikšmių reikia kiekvieną kartą skaičiuojant sukčiavimo įvertį, o reikšmių kiekis labai nedidelis. Todėl reikšmes galima saugoti periodiškai atnaujinamame podėlyje. Podėlio naudojimas padeda išvengti dalies kreipinių į duomenų saugyklos komponentus. Naudojant šias reikšmes gaunamas galutinis sukčiavimo įvertis. Jis gražinamas per programinę sąsają „įvertinti sukčiavimą“.

Įvertinus sukčiavimą procesas nesibaigia. Visi tarpiniai rezultatai (kriterijų vertės, kriterijų grupių vertės, statistiniai duomenys iš kurių buvo gautos vertės) perduodami įverčių archyvavimo komponentui. Šis komponentas visus tarpinius duomenis saugo duomenų bazėje. Šis veiksmas leidžia optimizuoti transakcijų žymėjimą sukčiavimu bei esant reikalui gauti šiuos tarpinius rezultatus ir naudoti sistemos rezultatų interpretavimui. Šis sistemos atvirumas tarpiniams rezultatams padeda užtikrinti aukštas sukčiavimo aptikimo sistemos rezultatų interpretavimo galimybes žmonėms. Pateikiamas tarpinių rezultatų pavyzdys (žr. 13 priedas). Išsaugojus tarpinius rezultatus archyve šie rezultatai perduodami į transakcijų statistikos saugyklos komponentą bei tikimybių saugyklos komponentą naudojant programines sąsajas „atnaujinti duomenis“.

Kai įverčio skaičiavimo komponentas inicijuojamas per programinę sąsają „pažymėti sukčiavimą“, komponentas pakartotinai turi persiųsti tarpinius rezultatus transakcijų statistinių duomenų komponentui bei tikimybinių duomenų saugyklos komponentui. Kadangi naudojami tie patys tarpiniai rezultatai pakartotinai reikėtų skaičiuoti transakcijos kriterijų reikšmių vertes bei kriterijų grupių vertes. Dėl to yra išnaudojamas įverčių archyvas. Pagal transakcijos identifikatorių iš archyvo gaunami visi duomenys ir persiunčiami kitiems komponentams. Realioje aplinkoje sukčiavimo faktas išaiškėja vėliau nei vertinama transakcija. Todėl šie tarpiniai duomenys sukčiavimo žymėjimo momentu archyve visada bus prieinami.

2.4.4.3. Tikimybinio tinklo saugykla

Tikimybinio tinklo saugyklos komponentas turi dvi pagrindines atsakomybes. Atnaujinti ir kaupti tikimybinio tinklo rezultatus bei pateikti šiuos rezultatus įverčio skaičiavimo komponentui.

Analizuojant duomenų saugyklos komponentus svarbu nepamiršti, kad siekiama didelio nuolatos apdorojamų transakcijų kiekio. Įverčio skaičiavimo komponento atveju išlaikyti didelį pralaidumą sąlyginai paprasta. Vykdomos tik duomenų nuskaitymo operacijos, todėl kiekvienai transakcijai įvertį galima skaičiuoti nepriklausomai nuo kitų transakcijų. Esant poreikiui užtikrinti didesnį duomenų nuskaitymo iš duomenų bazės pralaidumą, galima naudoti nuskaitymo replikas (angl. *read replica*). Tuo tarpu saugyklos komponentas atlieka duomenų atnaujinimo funkcijas.

Analizuojant nefunkcinius reikalavimus buvo identifikuota, kad pasiekiamumas ir duomenų skirstymas į particijas yra būtini realioje aplinkoje veikiančioje sukčiavimo aptikimo sistemoje. Todėl visiškas duomenų vientisumo užtikrinimas tarp duomenų patricijų neįmanomas dėl CAP teoremos ribojimų. Vis dėlto, siekiant tikslaus sukčiavimo aptikimo, sistemos transakcijų statistiniai duomenys bei tikimybiniai duomenys turi būti paruošti iš didelio kiekio transakcijų. Daroma prielaida, kad skaičiuojant galutinį įvertį skaičiavimo rezultato paklaida dėl galimo duomenų nevientisumo bus minimali.

Papildomą duomenų atnaujinimo įgyvendinimo sudėtingumą įneša duomenų atnaujinimas iš kelių lygiagrečių procesų. Tokiu atveju reiktų užtikrinti duomenų vientisumą kritinėse sekcijose. Skirtingos duomenų bazių valdymo sistemos užtikrina skirtingus duomenų bazės operacijų transakcijų lygmenis. Naudojami skirtingi vientisumo užtikrinimo mechanizmai. Pavyzdžiui tradicinės releacinės duomenų bazių valdymo sistemos gali užtikrinti visišką duomenų vientisumą atnaujinant duomenis iš kelių procesų vienu metu. Tuo tarpu dokumentinės duomenų bazių valdymo sistemos dažniausiai užtikrina vientisumą vieno dokumento ribose. Vis dėlto nepriklausomai nuo naudojamo mechanizmo, kritinių sekcijų valdymas dažniausiai įneša papildomą sudėtingumą kuris neigiamai veikia proceso efektyvumą. Todėl norint padidinti efektyvumą ir pralaidumą siūloma skaičiavimus išskirstyti išvengiant kritinių sekcijų egzistavimo. Išvengti kritinių sekcijų galima užtikrinant, kad to paties duomenų rinkinio niekad nemėgintų atnaujinti keli procesai vienu metu. Tai galima pasiekti naudojant pakopinį proceso vykdymą.

Pakopinis vykdymas įgyvendinamas išskaidant procesą į tarpusavyje nepriklausomas pakopas. Kiekviena iš pakopų atnaujintų skirtingus duomenų rinkinius. Skirtingi procesai vienu metu vykdytų skirtingų transakcijų atnaujinimo procesų skirtingas pakopas. Kiekvienai proceso pakopai bet kuriuo laiko momentu būtų skiriamas tik vienas procesas. Pakopinis proceso

vykdymas prasideda nuo pirmosios pakopos. Pirmasis procesas vykdo pirmosios pakopos operacijas. Jas įvykdęs perduoda vykdymo rezultatus antrajam procesui kuris pradės vykdyti antrosios pakopos operacijas. Vykdant pakopos operacijas vykdomi reikalingi duomenų atnaujinimai duomenų bazėje. Baigus pakopos vykdymą rezultatai perduodami sekančiam procesui. Jeigu pakopos bus vykdomos lėčiau nei gaunami duomenys apdorojimui, susidarys situacija kai pirmasis procesas apdoros pavyzdžiui transakcijos, numeriu $n + 2$, pirmąją pakopą. Tuo tarpu paskutinis procesas vis dar apdoroja n - tosios transakcijos paskutinę pakopą. Tokiu būdu skirtingos pakopos bus išskirstytos per skirtingus procesus be papildomo vykdymo koordinavimo mechanizmo. Pakopų vykdymą vienu metu galima išskirstyti į tiek procesų kiek nepriklausomų proceso pakopų pavyks apsibrėžti. Šio metodo trūkumas, kad lygiagrečiai galinčių veikti procesų kiekis ribojamas priklausomai nuo to konkretaus sprendžiamo uždavinio. Nuo jo priklauso nesusijusių veiksmų duomenų atnaujinimo procese identifikavimo sudėtingumas.

Galimi alternatyvus įgyvendinimo būdai. Siekiant papildomai optimizuoti vykdymą galima būtų leisti vienu metu vykdyti skirtingas tos pačios transakcijos pakopas. Tačiau tais atvejais, kai sekančioms pakopoms reikalingi prieš tai vykusių pakopų darbo rezultatai toks įgyvendinimas reikalauja papildomo procesų koordinavimo tarpusavyje. Todėl šis įgyvendinimo būdas nėra rekomenduojamas. Atnaujinimo procesą taip pat galima įgyvendinti atsižvelgiant į funkcinę programavimo paradigmą. Pagal funkcinę paradigmą pakopas reikėtų išreikšti grynomis funkcijomis (angl. *pure function*). Tai reikštų, kad pakopos operacijos turėtų gauti visus reikiamus duomenis vykdymo inicijavimo metu, o visus rezultatus gražinti inicijavusiam komponentui arba sekančiam komponentui grandinėje. Šiuo įgyvendinimo atveju tiesioginis duomenų nuskaitymas ar atnaujinimas duomenų bazėje būtų ribojamas. Toks įgyvendinimo būdas užtikrina, kad skirtingi procesai neturėtų bendrų resursų kuriuos reikia sinchronizuoti tarpusavyje. Taip įgyvendintus pakopų komponentus būtų galima horizontaliai plėsti beveik neribotai. Tačiau dėl sistemos specifikos kiekvienai transformacijai reikalingas didžiulis duomenų rinkinys, kurio kopijos perdavimas kiekvienam komponentui būtų neefektyvus. Taip pat svarbu paminėti, kad duomenų atnaujinimo funkcijose lėčiausioji proceso grandis greičiausiai bus būtent operacijos su duomenų baze, o ne duomenų atnaujinimo operacijų paruošimas. Todėl šiuo atveju yra efektyviau kiekvienos pakopos vykdymo metu atlikti duomenų nuskaitymą ir duomenų modifikacijas duomenų bazėje. Toks įgyvendinimas per kelis procesus išskirsto ne tik skaičiavimus, tačiau ir duomenų bazės operacijų valdymą.

2.4.4.3.1. Agreguojami duomenys

Tinklo saugykloje agreguojamų duomenų poreikis tiesiogiai priklauso nuo įgyvendinamo algoritmo. Toliau analizuojamas sukonkretinto tinklo duomenų paruošimo procesas. Įverčio

skaičiavimui reikalingi iš anksto paruošti duomenų įrašai: $P(SUK \cap KG_1 \cap KG_2 \cap ... \cap KG_k)$, $P(SUK)$, $P(K_{ij}|SUK)$. Buvo identifikuota, kad $P(SUK \cap KG_1 \cap KG_2 \cap ... \cap KG_k)$ yra išvestinė dedamoji, kurią galima apskaičiuoti pagal: $P(SUK \cap KG_1 \cap KG_2 \cap ... \cap KG_k) = P(SUK) \cdot \prod_{i=1}^{k} P(KG_i|SUK)$ (žr. 2 priedas). Todėl papildomai reikia paruošti dedamąsias: $P(KG_i|SUK)$. Taip pat, norint įvertinti kriterijaus grupės riziką reikia žinoti vidutinę grupės kriterijų kombinacijų rizikos tikimybę bei standartinį nuokrypį nuo šios vidutinės tikimybės.

Žinoma, kad $P(SUK) = \frac{t_{SUK}}{t}$, $P(KG_i|SUK) = \frac{t_{KG_i}}{t_{SUK}}$, $P(K_{ij}|SUK) = \frac{t_{K_{ij}}}{t_{SUK}}$. Todėl norint atnaujinti duomenis pakanka atnaujinti skaitliukus: $t, t_{SUK}, t_{K_{ij}}, t_{KG_i}$. Tam kad apskaičiuoti vidutinę riziką bei standartinį nuokrypį, reikia sekti kiekvienos kriterijų grupės kriterijų reikšmių kombinacijos riziką. Vien tik sekti šias rizikas nepakanka, kadangi skaičiavimuose naudojamas vidurkis ir standartinis nuokrypis. Tokiu atveju tiek vidurkį tiek standartinį nuokrypį reikėtų perskaičiuoti kaskart sumuojant visų dedamųjų rezultatus. Duomenų atnaujinimo funkcija bus vykdoma dažnai, todėl tikslinga įgyvendinti pakopinį vidurkio bei standartinio nuokrypio atnaujinimą. Norint atnaujinti vidurkį iš naujo nesumuojant visų dedamųjų pakanka papildomai saugoti jau apdorotų kombinacijų rizikų sumą ir ją nuolatos atnaujinti. Siekiant standartinio nuokrypio apskaičiavimo reikia papildomai saugoti rizikų reikšmių kvadratų sumą.

Pateikiama tinklo esybių diagrama (žr. 14 priedas) atspindinti visus reikalingus išsaugoti duomenis. Tinklo esybių diagramoje pateikiamos 4 pagrindinės esybės: kriterijų tikimybės, kriterijų grupių tikimybės, grupių statistika, bendra statistika. Nesunku pastebėti, kad kriterijų tikimybės bei kriterijų grupių tikimybės praktiškai identiškos struktūros prasme. Gali pasirodyti logiška jas apjungti į vieną esybę. Taip pat verčių ir kombinacijų esybės yra praktiškai identiškos. Tačiau nepaisant panašumo šias esybes siūloma įgyvendinti atskirai. Kadangi duomenų atnaujinimo procesą siūloma įgyvendinti išskaidant procesą į izoliuotas proceso pakopas, nepaisant esybių panašumo jų denormalizavimas ir atskyrimas padės lengviau apibrėžti izoliuotas duomenų atnaujinimo proceso dalis. Atskyrus visas šias esybes galima leisti jas atnaujinti iš skirtingų procesų. Dėl to siūloma visas šias esybes atskirti, o esybių saugojimą įgyvendinti atskirose duomenų bazės struktūrose. Duomenų struktūrų izoliaciją įgyvendinti galima įvairiais būdais priklausomai nuo naudojamos duomenų bazių valdymo sistemos realizacijos: atskiros duomenų bazės, atskiros duomenų bazės schemos ir kt.

Įgyvendinus šių 4 esybių atskyrimą galima apibrėžti 4 aiškias pakopas tikimybinių duomenų atnaujinimui: bendros statistikos atnaujinimas, grupių statistikos atnaujinimas, kriterijų grupių statistikos atnaujinimas, kriterijų statistikos atnaujinimas. Apjungus kriterijų ir kriterijų grupių esybes apribotumėme galimybes išskirstyti darbus, kadangi bendros struktūros taptų kritinėmis sekcijomis norint šias esybes atnaujinti iš skirtingų procesų. Analogiška situacija su

verčių kombinacijų esybėmis. Normalizavus struktūras ir apjungus dalį esybių taip pat susidarytų situacija kai tektų iš kelių skirtingų procesų atlikti duomenų atnaujinimus tose pačiose duomenų struktūrose. Svarbu paminėti, kad dauguma duomenų bazių valdymo sistemų pilnai užtikrina ACID transakcijas. Tai reiškia, kad atnaujinant duomenis iš skirtingų procesų, problemų dėl duomenų vientisumo nekils, tačiau tokiu atveju priklausomai nuo duomenų bazės realizacijos vis tiek potencialiai prarandamas efektyvumas. Norint užtikrinti duomenų vientisumą dažniausiai pasitelkiami lentelės lygio ar duomenų įrašo lygio užraktai (angl. *locks*). Atliekant daug užklausų tame pačiame duomenų rinkinyje kyla didelė rizika, kad atnaujinimo užklausų vykdymo metu užklausoms teks dažnai laukti resursų.

2.4.4.3.2. Siūloma realizacija

Tinklo duomenų saugykla pateikia dvi programines sąsajas: duomenų atnaujinimo bei duomenų gavimo.

Duomenų atnaujinimas realizuojamas prašymus atnaujinti duomenis priimant tikimybinių duomenų paruošimo inicijavimo komponente. Šis komponentas grupuoja transakcijas ir tolimesnes atnaujinimo operacijas vykdo ne kiekvienai transakcijai atskirai, o sukaupus iš anksto apibrėžtą kiekį transakcijų. Duomenų atnaujinimo komponentams perduodama sukauptų atnaujinimo prašymų grupė. Atnaujinant duomenis ne po kiekvienos transakcijos sutaupoma resursų. Sumažinamas reikalingų atlikti pavienių operacijų su duomenų baze kiekis. Pavyzdžiui jeigu po kiekvienos transakcijos būtų atnaujinamas įvykusių transakcijų skaitliukas, apdorojus 50 transakcijų būtų atlikta 50 duomenų bazės užklausų vykdančių atnaujinimo operaciją padidinančią skaitliuko reikšmę per vieną. Sugrupavus šias užklausas, o jų kiekį suskaičiavus prieš inicijuojant užklausą pakaktų vienos užklausos padidinančios skaitliuką per 50 vienetų.

Tikimybinių duomenų paruošimo inicijavimo komponentas surinktus atnaujinimo prašymus perduoda bendrinių tikimybių atnaujinimo komponentui. Šis komponentas iš gautų prašymų apskaičiuoja bendrinės statistikos pokytį, atlieka vieną duomenų atnaujinimo užklausą ir perduoda nepakeistus duomenų atnaujinimo prašymus kriterijų grupių tikimybių atnaujinimo komponentui.

Kriterijų grupių tikimybių atnaujinimo komponentas taip pat apskaičiuoja statistikos pokytį ir atnaujina kriterijų grupių tikimybių esybę. Atlikus visus reikiamus veiksmus atnaujinimo prašymai toliau perduodami sekančiai pakopai. Procesas kartojamas kol visi 4 duomenų atnaujinimo komponentai atnaujina kiekvienam iš jų priskirtas esybes. Komponentų atnaujinančių esybes grandinė pateikiama detalizuotoje komponentų diagramoje (žr. 7 priedas).

Antroji tinklo statistikos saugyklos realizuojama funkcija yra tikimybinių duomenų pateikimas. Gavus užklausą tikimybinių duomenų pateikimo komponentas iš visų komponento duomenų bazių nuskaito reikalingus duomenis ir juos gražina. Šiuo atveju duomenų pateikimo

komponentas iš visų duomenų struktūrų atlieka tik nuskaitymo operacijas. Duomenų nuskaitymas iš duomenų bazės, kurioje duomenis atnaujina kitas procesas, efektyvumui pakenkti neturėtų. Pagal apibrėžimą struktūroje susidaro kritinė sekcija. Vienas procesas duomenis įrašo, kitas juos nuskaito. Todėl tarp procesų reikia užtikrinti duomenų sinchronizaciją. Vis dėlto, dauguma duomenų bazių valdymo sistemų, pavyzdžiui „PostgreSQL“, šalia užraktų mechanizmo naudoja vidinį įrašų versijavimą. Šis versijavimas leidžia įgyvendinti mechanizmą, kurio pagalba duomenis nuskaitantis procesas gauna duomenų įrašą iš versijų žurnalo, ir duomenis įrašantis procesas nėra blokuojamas. Todėl šiuo atveju vykdymas efektyvesnis nei naudojant kelis duomenis atnaujinančius procesus vykdančius atnaujinimo užklausas toje pačioje duomenų struktūroje.

Svarbu paminėti, kad visiškas duomenų vientisumas tarp skirtingų duomenų struktūrų nėra užtikrinamas. Nėra garantuojama, kad duomenų nuskaitymo metu bus įvykdytos visos duomenų atnaujinimo pakopos, o naujai gauto transakcijų rinkinio pakopų apdorojimas dar nebus pradėtas. Tai reikštų, kad gali kilti situacija kai pavyzdžiui tik dvi pirmosios duomenų struktūros buvo atnaujintos pagal naujausią transakcijų rinkinį. Vis dėlto nutraukus duomenų perdavimą duomenų atnaujinimo komponentams, galiausiai visos pakopos būtų įvykdytos ir tuo laiko momentu duomenys būtų vientisi. Todėl galima teigti, kad užtikrinamas galutinis vientisumas (angl. *eventual consistency*). Dėl didelio duomenų kiekio saugyklose tikslumo rezultatų paklaida turėtų būti minimali.

Norint papildomai padidinti efektyvumą ir išvengti didelio kiekio kreipinių į duomenų bazę galima dalį duomenų saugoti podėlyje (angl. *cache*), kuris išsaugotų duomenis operatyviojoje atmintyje. Dauguma iš saugomų tikimybinio tinklo esybių (žr. 14 priedas) turi pakankamai mažai skirtingų kombinacijų išsaugojimui. Sukčiavimo aptikimui naudojami iš anksto apibrėžti kriterijai (žr. 2.3 poskyris). Su šiais konkrečiais kriterijais kriterijų tikimybės bei kriterijų grupių tikimybės esybės turės kelias dešimtis skirtingų kriterijų bei jų verčių. Tokio dydžio duomenų rinkinys nesunkiai išsaugomas operatyviojoje atmintyje. Bendros statistikos esybė visada turės vieną esybės objektą, kadangi saugo tik du skaitliukus reikalingus apskaičiuoti bendrą sukčiavimo tikimybę. Tačiau grupių statistikos esybė turės didelį kiekį skirtingų kombinacijų, todėl jos podėlyje saugoti nereikėtų. Žvelgiant trumpuoju laikotarpiu tikimybė, kad nuolatos bus naudojamos tos pačios grupių kombinacijos yra nedidelė. Todėl išsaugojus kombinacijų poaibį podėlyje reikiamos reikšmės nebūtų ir tektų atnaujinti visą podėlį.

2.4.4.4. Transakcijų duomenų saugykla

Transakcijų duomenų saugykla, kaip ir tikimybinių duomenų saugykla pateikia programines sąsajas duomenų atnaujinimui ir duomenų gavimui.

Aptarinėjant tinklo tikimybių saugyklos realizaciją buvo pasiūlyta duomenų atnaujinimo išskirstymo per kelis procesus strategija. Transakcijos duomenų saugyklos tikslas yra analogiškas, tik duomenys skiriasi. Todėl siekiant išskirstyti duomenų atnaujinimą vadovaujamasi tais pačiais principais.

Transakcijų duomenys išskaidomi į 3 esybes: asmeninė statistika, bendra periodinė statistika, išorinių sąlygų statistika. Asmeninės statistikos esybė apima tokius kriterijus kaip: paskutinės transakcijos laikas, paskutinės transakcijos vieta, mažiausias tarpas tarp dviejų transakcijų. Bendra periodinė statistika apibrėžia tokius duomenis kaip: vidutiniškai per savaitę išleidžiama pinigų suma, vidutiniškai per mėnesį atliekamų transakcijų kiekis ir kt. Išorinių sąlygų statistika apibrėžia vietos, laiko bei pardavėjo rizikingumą. Šių trijų esybių atnaujinimas išskaidytas į tris izoliuotus komponentus (žr. 7 priedas).

Transakcijų duomenų saugykloje prašymų atnaujinti duomenis apdorojimas vykdomas per papildomą paruošimo inicijavimo komponentą. Šis komponentas grupuoja atnaujinimo prašymus ir tolimesniems duomenų atnaujinimo procesams perduoda atnaujinimo prašymų masyvus. Tinklo duomenų saugykla ir transakcijų duomenų saugykla sąmoningai turi po atskirą prašymus grupuojantį komponentą. Tai leidžia priklausomai nuo komponentų efektyvumo kalibruoti ir apibrėžti skirtingus grupių dydžius, esant poreikiui įvesti papildomas koordinavimo taisykles. Taip užtikrinamas transakcijų duomenų komponento bei tinklo duomenų komponento tarpusavio autonomiškumas. Kadangi nėra bendrų dalinamų resursų tarp komponentų juos galima diegti atskiroje infrastruktūroje, horizontaliai plėsti nepriklausomai nuo kitų komponentų.

Transakcijų duomenų gavimo funkcijos įgyvendinimas analogiškas tinklo saugyklos duomenų gavimo funkcijos realizacijai. Transakcijų duomenų statistikos pateikimo komponentas surenka reikiamus duomenis iš esybių duomenų bazių ir pateikia gautus duomenis.

Transakcijos duomenų saugyklos atveju taip pat įgyvendinamas duomenų saugojimas podėlyje. Išorinių sąlygų esybių laiko rizika vertinama valandomis todėl bus saugoma tik 24 skirtingų rizikų skaitliukai. Unikalių pardavėjų bei vietų kiekis gerokai didesnis, todėl juos saugoti podėlyje gali būti neefektyvu. Taip pat podėlyje turi būti saugoma bendra periodinė statistika, kadangi ši statistika reikalinga kiekvienos transakcijos vertinimui. Podėlyje neturi būti saugoma asmeninė statistika, kadangi skirtingų pirkėjų kiekiai yra labai dideli. Transakcijas apdorojanti įstaiga gali turėti ir milijonus klientų, tačiau tikimybė du kartus, mažu laiko skirtumu aptarnauti tą patį pirkėja labai maža.

2.4.4.5. Komponentų apibendrinimas

Buvo suprojektuoti 4 pagrindiniai loginiai sistemos komponentai, kurie kartu sudaro sukčiavimo aptikimo sistemą. Buvo identifikuotos kiekvieno komponento atsakomybės. Atsižvelgiant į komponentų įgyvendinamas funkcijas bei reikalavimus buvo apibrėžtos

sukčiavimo aptikimo sistemos architektūrinio įgyvendinimo detalės agnostiškos naudojamoms technologijoms. Realizacija apibrėžiama agnostiška technologijoms, kadangi konkretūs karkasai, įrankiai ar net programavimo kalbos tobulėja labai greitai. Sukčiavimo aptikimo sistemos įgyvendinimo apribojimas konkrečiomis technologijomis sutrumpintų sukčiavimo aptikimo sistemos pritaikomumą realioje aplinkoje. Visi aprašyti sprendimai gali būti įgyvendinti naudojant bet kurią bendrosios paskirties programavimo kalbą kartu su duomenų bazių valdymo sistema.

Buvo pasiūlyta įgyvendinti duomenų apsikeitimo komponentą kaip fasadą įgyvendinanti duomenų apsikeitimo protokolui reikalingas funkcijas. Apibrėžta sistemos kriterijų konfigūracijos bei vertinimo realizacija. Apibrėžtas tarpinių rezultatų archyvo naudojimas siekiant išvengti pakartotinio rezultatų skaičiavimo. Pasiūlyta tinklo duomenų saugyklos komponento bei transakcijų statistikos komponento realizacija tinkama naudoti kartu su išskirstytaisiais skaičiavimais. Identifikuotos esybės tinkamos saugoti sistemos podėlyje.

2.4.5. Techninės realizacijos detalės

Praeituose skyriuose buvo išnagrinėti sistemos reikalavimai, architektūrinis sistemos vaizdas. Sistema buvo įgyvendinta laikantis apsibrėžtos architektūrinės vizijos. Architektūra buvo apibrėžta agnostiška technologijoms, tačiau įgyvendinant sistemą konkrečių technologinių pasirinkimų išvengti neįmanoma. Sistemos įgyvendinimui panaudotos pagrindinės technologijos:

- „Java“ programavimo kalba;
- „Vertx“ reaktyvių sistemų kūrimo įrankių rinkinys;
- „MongoDB“ duomenų bazių valdymo sistema.

Įgyvendinant sukčiavimo aptikimo sistemą kaip pagrindinė programavimo kalba buvo pasirinkta Java, naudojant kartu su „Vertx“ reaktyvių sistemų kūrimo įrankių rinkiniu. Iš šio rinkinio buvo naudojamas asinchroninis HTTP serveris įgyvendinti duomenų apsikeitimo komponento programines sąsajas bei asinchroninė įvykiu jungtis (angl. *event bus*). „Vertx“ asinchroninė įvykių jungtis leido integruoti sistemos komponentus skelbėjo – prenumeruotojo principu (angl. *publish-subscribe*). Ši jungtis sistemoje taip pat naudojama vietoje atskiros eilės realizacijos, kad užtikrinti asinchroninę integraciją tarp duomenų saugyklų komponentų ir įverčio skaičiavimo komponento.

Įgyvendinant sukčiavimo aptikimo sistemą buvo pasirinkta duomenų bazių valdymo sistema „MongoDB“. Ši duomenų bazių valdymo sistemos realizacija pasirinkta dėl to, kadangi tai yra vieną iš duomenų bazių valdymo sistemų, kuriai yra pateikiama oficiali asinchroninės „Java“ programavimo kalbai tinkamos tvarkyklės versija.

Sistemos asinchroniškumas įgyvendintas visuose sistemos sluoksniuose: asinchroninė HTTP serverio realizacija, asinchroninė tarpusavio komponentų sąsaja per „Vertx“ įvykių jungtį

bei asinchroninė sąsaja su duomenų baze. Šis platus asinchroniškumo naudojimas turėtų padėti užtikrinti optimalų resursų išnaudojimą. Ši optimizacija turėtų leisti pasiekti didelį sistemos pralaidumą su mažesniu sistemai reikalingų resursų kiekiu.

2.4.6. Rekomenduojama sistemos diegimo konfigūracija

Pateikiama siūloma sistemos diegimo konfigūracija tinkama pritaikymui realioje gamybinėje aplinkoje (žr. 15 priedas). Naudojant tokio tipo sistemos diegimo konfigūraciją sistema turėtų galėti patenkinti sistemos nefunkcinius reikalavimus: įvertinti transakcijas greičiau nei per 1 sekundę bei užtikrinti stabilų 10 000 transakcijų per sekundę apdorojimą.

Sistema suprojektuota atsižvelgiant į galimybes duomenų paruošimą vykdyti išnaudojant išskirstytuosius skaičiavimus. Saugyklos esybės apibrėžtos izoliuotos tarpusavyje, kad esant poreikiui būtų galima jas valdyti iš atskirų vienu metu veikiančių procesų, kiekvieną iš esybių perkelti į atskirą duomenų bazių valdymo sistemą.

Pirmiausia siekiant didesnio pralaidumo kiekvieną saugyklos komponento esybę galima iškelti į atskirą duomenų bazę. Taip pat kiekvieną bazės esybę replikuoti į bent kelias nuskaitymo replikas. Tai padėtų paskirstyti duomenų nuskaitymo apkrovas. Taip pat į atskirus serverius siūloma atskirti kiekvieną duomenų saugyklos atnaujinimo procesą. Iškėlus duomenų paruošimo procesų komponentus į atskirus serverius būtų atlaisvinti resursai įverčio skaičiavimo komponentui, jis galėtų užtikrinti didesnį pralaidumą. Kadangi klientinių sistemų atžvilgiu įverčių skaičiavimo komponentas neturi vidinės būsenos, be jokių papildomų sesijos sinchronizavimo mechanizmų komponentas gali būti horizontaliai plečiamas per kelis serverius. Taip pat įverčių skaičiavimo komponente vykdomas įverčių archyvo valdymas. Tačiau archyvo komponentas neatlieka jokių duomenų modifikacijų, o tik naujų įrašų įterpimą ir jų nuskaitymą. Todėl jį galima vykdyti iš kelių procesų vienu metu.

Visi šie pakeitimai įmanomi, kadangi projektuojant sistemą buvo apibrėžta izoliaciją tarp duomenų bazėje saugomų esybių. Buvo aprašytas išskirstytiesiems skaičiavimams tinkamas duomenų atnaujinimo procesas. Įgyvendinant sąsają tarp komponentų buvo panaudota „Vertx“ įrankio įvykių eilė naudojanti TCP protokolą žinutėms tarp komponentų perduoti. Todėl skirtingus sistemos komponentus galima sudiegti tiek toje pačioje Java virtualioje mašinoje, tiek skirtingose fizinėse mašinose turinčiose tarpusavio tinklo prieigą.

2.4.7. Apibendrinimas

Prieš tiriant sukčiavimo aptikimo sistemos efektyvumą bei įgyvendinamumą, buvo apibrėžti reikalavimai sukčiavimo aptikimo sistemai. Atsižvelgiant į aptikimo sistemos reikalavimus buvo identifikuotos reikalingos sukčiavimo aptikimo sistemos funkcijos bei loginiais sistemos komponentai įgyvendinantys šias funkcijas. Siekiant įgyvendinti sistemą

atitinkančią reikalavimus ir pritaikomą realioje aplinkoje, buvo išanalizuotos techninių sprendimų alternatyvos, parinkti komponentų integracijos, realizacijos sprendimai.

Išanalizavus komponentų integracijos būdus buvo pasiūlyta sukčiavimo aptikimo sistemos komponentų integracijai naudoti duomenų perdavimą eilėmis. Išanalizavus komponentų funkcijas bei reikalavimus buvo apibrėžtos komponentų įgyvendinimo strategijos. Pasiūlyta išskirstytųjų skaičiavimų strategija leidžianti išskirstyti duomenų atnaujinimo procesus. Apibrėžtas podėlio naudojimas sistemoje, identifikuotos tinklo saugyklos esybės, pasiūlytas būdas apibrėžti lengvai pildomą kriterijų konfigūraciją sistemoje.

Atsižvelgiant į apibrėžtą sistemos architektūrą, buvo įgyvendinta sukčiavimo aptikimo sistema. Sistemos įgyvendinimui naudojama „Java“ programavimo kalbą, „Vertx“ reaktyvaus programavimo įrankių rinkinys bei „MongoDB“ duomenų bazių valdymo sistema. Taip pat atsižvelgiant į pasirinktą sistemos architektūrą bei nefunkcinius reikalavimus pasiūlyta sistemos diegimo konfigūracija.

3. Sukčiavimo aptikimo metodo vertinimas

Atlikus literatūros analizę buvo identifikuota, kad svarbiausi sukčiavimo aptikimo sistemos vertinimo kriterijai yra (žr. 1.2.3 poskyris):

- statistinis tikslumas;
- interpretavimo paprastumas;
- vykdymo efektyvumas;
- ekonominė kaina.

Ekonominė sistemos įgyvendinimo kaina priklauso nuo įvairių veiksnių, todėl vertinama nebus. Įgyvendinus sukčiavimo aptikimo sistemą buvo atliekamas sistemos efektyvumo, tikslumo bei interpretavimo paprastumo vertinimas. Tikslumo ir efektyvumo bandymai buvo atliekami atskirai. Norint įvertinti tikslumą svarbu užtikrinti griežtą sistemos veikimo kontekstą apdorojamų transakcijų prasme. Kitu atveju būtų sunku nuspėti kokių sukčiavimo įverčių tikėtis. Efektyvumo vertinimo metu svarbu užtikrinti griežtą sistemos infrastruktūros aplinką. Efektyvumo tyrimo metu reikalingi dideli nuolatos apdorojamų transakcijų kiekiai. Vykdant efektyvumo bandymus sukčiavimo įverčių reikšmės ignoruojamos. Šis bandymų atskyrimas padeda izoliuoti sistemą nuo aplinkos poveikio rezultatams, padeda korektiškai įvertinti sistemą.

3.1. Tikslumo vertinimas

Tikslumo vertinimo tikslas įsitikinti sistemos gebėjimu apsimokyti pagal pateiktą duomenų rinkinį ir identifikuoti sukčiavimą dominuojantį apmokymui pateikiamame istoriniame duomenų rinkinyje. Vertinant sistemos tikslumą įprasta naudoti standartizuotus statistinio tikslumo kriterijus, kurie buvo identifikuoti literatūroje (žr. 1.2.3.1 darbo poskyris). Vis dėlto, norint išmatuoti šių kriterijų vertes reikalingas realus transakcijų duomenų rinkinys. Būtina turėti transakcijų atributus bei požymius ar transakcija yra sukčiavimas. Kadangi nėra viešai prieinamų realių transakcijų rinkinių su reikiamais transakcijų atributais, sistemos paruošimui naudojami transakcijų duomenys sugeneruoti pagal apsibrėžtą elgsenos šabloną. Mėginimas dirbtinai generuojant duomenis sužymėti vertinamas transakcijas sukčiavimu ar nesukčiavimu iškraipytų bandymų rezultatus. Tokiu atveju vertinant sistemos tikslumą, bandymų rezultatams didesnį poveikį turėtų duomenų generavimo algoritmas, o ne pati sukčiavimo aptikimo sistema.

Siekiant įvertinti sistemos tikslumą kontroliuojamoje aplinkoje buvo atlikti 5 bandymai pagal skirtingus scenarijus su skirtingą elgseną atitinkančiomis transakcijomis. Svarbu nepamiršti, kad įgyvendinta sukčiavimo aptikimo sistema pateikia ne dvejetainį rezultatą, o skaitinę reikšmę nusakančią tikimybę, kad transakcija su šiais atributais yra sukčiavimas. Šių bandymų metu efektyvumo rodikliai buvo ignoruojami.

3.1.1. Bandymo aplinka

Siekiant prasmingai įvertinti sukčiavimo aptikimo sistemos tikslumą pagal apsibrėžtus elgsenos šablonus buvo sugeneruotas didesnis nei milijono transakcijų rinkinys. Rinkinyje sugeneruota fiktyvi pastarųjų 10 metų transakcijų istorija 4000 unikalių kortelės turėtojų. Šis transakcijų rinkinys panaudotas sistemos paruošimui.

Paruošus sistemą istoriniu duomenų rinkiniu buvo išjungtos duomenų atnaujinimo funkcijos. Taip siekiama užtikrinti, kad kiekvienas iš scenarijų būtų vertinamas identiškoje sistemos būsenoje. Dėl to, galima teigti kad skirtingų scenarijų ar transakcijų įverčiai yra prasmingai palyginami tarpusavyje.

Istorinių transakcijų generavimui buvo parašytas duomenų generatorius generuojantis transakcijų duomenis pagal apsibrėžtą elgsenos šablono konfigūraciją (žr. 21 priedas). Duomenų pateikimui bei įverčio gavimui buvo pasirašyta sugeneruotų duomenų pateikimo programa, pateikianti duomenis ir gaunanti sukčiavimo įvertį per duomenų apsikeitimo komponento programines sąsajas (žr. 17 priedas).

Tikslumo vertinimui buvo apibrėžti 5 sistemos tikslumo testavimo scenarijai. Kiekvienam scenarijui buvo paruoštas transakcijų rinkinys iš 50 transakcijų skirtingiems kortelių savininkams. Kiekviename scenarijuje buvo vertinamos skirtingo rizikingumo transakcijos. Šiame kontekste transakcija ar elgsenos šablonas laikomas rizikingu jeigu tarp šį šabloną atitinkančių transakcijų sukčiavimas pasitaiko dažniau nei įprasta kitiems šablonams.

Pirmojo scenarijaus atveju vertinamos aukščiausios rizikos transakcijos. Modeliuojama situacija, kai transakcija visiškai atitinka įprastus sukčiavimui elgsenos šablonus. Šiame scenarijuje buvo vertinamos transakcijos pasižyminčios šiomis savybėmis: dažnai sukčiavime pasitaikantis paros laikas, dažnai sukčiavime pasitaikanti vieta, dažnai sukčiavime pasitaikantis pardavėjas, daug didesnė nei įprasta transakcijos vertė, daug trumpesnis nei įprasta laiko skirtumas tarp transakcijų, daug mažesnis nei įprasta atstumas tarp transakcijų.

Antrojo scenarijaus atveju buvo vertinamos aukštos rizikos transakcijos. Scenarijus modeliuoja transakcijas dauguma atributų atitinkančias įprastus sukčiavimui elgsenos šablonus. Šiame scenarijuje buvo vertinamos transakcijos pasižyminčios šiomis savybėmis: dažnai sukčiavime pasitaikantis paros laikas, dažnai sukčiavime pasitaikanti vieta, retai sukčiavime pasitaikantis pardavėjas, daug didesnė nei įprasta transakcijos vertė, daug trumpesnis nei įprasta laiko skirtumas tarp transakcijų, daug mažesnis nei įprasta atstumas tarp transakcijų.

Trečiojo scenarijaus atveju buvo vertinamos vidutiniškai rizikingos transakcijos. Scenarijus modeliuoja transakcijas keliais atributais atitinkančias įprastus sukčiavimui elgsenos šablonus. Šiame scenarijuje buvo vertinamos transakcijos pasižyminčios šiomis savybėmis: retai sukčiavime pasitaikantis paros laikas, retai sukčiavime pasitaikanti vieta, retai sukčiavime

pasitaikantis pardavėjas, daug didesnė nei įprasta transakcijos vertė, daug trumpesnis nei įprasta laiko skirtumas tarp transakcijų, daug mažesnis nei įprasta atstumas tarp transakcijų.

Ketvirtojo scenarijaus atveju buvo vertinamos mažai rizikingos transakcijos. Modeliuojamos transakcijos dauguma atributų atitinkančios įprastus elgsenos šablonus. Scenarijuje vertinamos transakcijos pasižyminčios šiomis savybėmis: retai sukčiavime pasitaikantis paros laikas, retai sukčiavime pasitaikanti vieta, dažnai sukčiavime pasitaikantis pardavėjas, įprasta transakcijos vertė, įprastas laiko skirtumas tarp transakcijų, įprastas atstumas tarp transakcijų.

Penktojo scenarijaus atveju buvo vertinamos nerizikingos transakcijos. Šis scenarijus laikomas kontroliniu, kadangi modeliuojamos transakcijos visais atributais atitinkančios įprastus elgsenos šablonus.

Svarbu paminėti, kad realioje aplinkoje sukčiavimas gali pasitaikyti tarp bet kokio tipo transakcijų. Tiriama sukčiavimo aptikimo sistema veikia apsimokančios sistemos principu. Atliekant tikslumo vertinimus sistema buvo apmokyta generuotais duomenimis. Generuojant duomenis buvo modeliuojama sukčiavimo elgsena atitinkanti aprašytus aukšto rizikingumo scenarijus. Apmokius sistemą su realiu duomenų rinkiniu, tinklas būtų užpildytas tikimybėmis atitinkančiomis realius duomenis. Todėl apmokius sistemą realiais duomenimis šiame tyrime apibrėžti scenarijai nebūtinai būtų identifikuojami kaip sukčiavimas. Realiame rinkinyje sukčiavimo šablonai gali būti kitokie nei apibrėžtieji sistemos vertinimui. Apmokius sistemą realiu transakcijų rinkiniu, reikėtų realiame rinkinyje identifikuoti skirtingo rizikingumo transakcijas ir atlikti vertinimus šioms transakcijomis.

3.1.2. Bandymo eiga

Naudojant iš anksto paruoštus scenarijų duomenis buvo paeiliui prašoma sukčiavimo aptikimo sistemos pateikti sukčiavimo įvertį vertinamoms transakcijoms iš skirtingų scenarijų.

Pirmiausia įvertinimas atliktas kiekvienai iš pirmojo scenarijaus transakcijų. Aritmetinis gautų pirmojo scenarijaus įverčių vidurkis lygus - 0,79. Standartinis nuokrypis nuo vidurkio – 0,3. Minimalus gautas sukčiavimo įvertis tarp šio scenarijaus transakcijų – 0,33, maksimalus įvertis - 1,00 (žr. 18 priedas). Sukčiavimo aptikimo sistemoje peržiūrėjus pirmojo scenarijaus transakcijų vertinimo tarpinius rezultatus nustatyta, kad žymus skirtumas tarp minimalaus ir maksimalaus įvertinimo atsiranda dėl sugeneruotos elgsenos skirtumų. Palyginus transakciją įvertinta sukčiavimo tikimybe 0,99 ir transakciją įvertinta sukčiavimo tikimybe 0,33 nustatyta, kad tikimybių skirtumas atsirado dėl to, kad mažesnį įvertį gavusio asmens atveju atstumai tarp transakcijų jo istoriniame duomenų rinkinyje buvo mažesni. Todėl skaičiuojant įvertį šio duomenų rinkinio kontekste transakcijos vieta atrodė nerizikinga. Pateikiami transakcijų apdorojimo tarpiniai rezultatai (žr. 16 priedas).

Gavus antrojo scenarijaus transakcijų įvertinimus nustatyta, kad vidutinis antrojo scenarijaus sukčiavimo įvertis - 0,65. Standartinis nuokrypis nuo vidurkio – 0,33. Minimalus gautas sukčiavimo įvertis tarp antrojo scenarijaus transakcijų taip pat buvo – 0,33, maksimalus įvertis – 1,0.

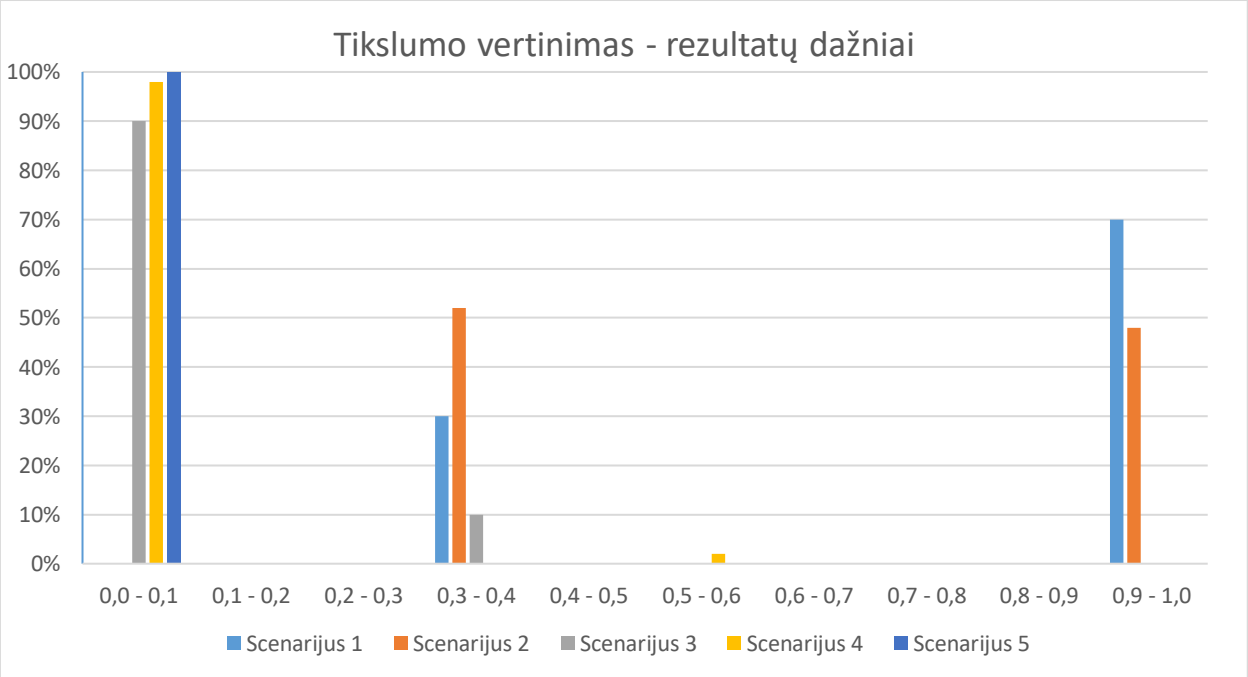
Atlikus trečiojo scenarijaus transakcijų vertinimą nustatyta, kad vidutinis trečiojo scenarijaus transakcijos sukčiavimo įvertis – 0,05. Standartinis nuokrypis nuo vidurkio – 0,09. Minimalus gautas sukčiavimo įvertis tarp trečiojo scenarijaus transakcijų – 0,01, maksimalus įvertis – 0,33.

Atlikus ketvirtojo scenarijaus transakcijų vertiną nustatyta, kad vidutinis ketvirtojo scenarijaus transakcijos sukčiavimo įvertis – 0,01, standartinis nuokrypis nuo vidurkio – 0,07. Minimalus gautas sukčiavimo įvertis tarp ketvirtojo scenarijaus transakcijų – 0,00003, maksimalus – 0,52.

Taip pat buvo atliktas penktojo scenarijaus transakcijų vertinimas. Nustatyta, kad vidutinis penktojo scenarijaus transakcijos įvertis – 0,002, standartinis nuokrypis nuo vidurkio – 0,002. Minimalus gautas penktojo scenarijaus transakcijų sukčiavimo įvertis – 0,00003, maksimalus – 0,005.

3.1.3. Bandymo rezultatai

Pateikiami apibendrinti tikslumo vertinimo rezultatai. (žr. 3 pav.). Detalūs tikslumo vertinimo rezultatai pateikiami prieduose (žr. 18 priedas).



3 pav. Tikslumo vertinimo rezultatų dažniai

Pirmojo ir antrojo scenarijaus transakcijos buvo apibrėžtos kaip rizikingiausių transakcijų grupės. Iš scenarijų įverčių dažnių pasiskirstymo matoma, kad pirmojo scenarijaus atveju 70 %

transakcijų pateiktas didesnis nei 0,9 tikimybinis įvertis. Likusiems 30 % pateiktas įvertis intervale 0,3 – 0,4. Vidutinė sukčiavimo tikimybė šiam scenarijui buvo 0,79. Antrojo scenarijaus atveju iš rezultatų dažnių matoma, kad sistema 48 % transakcijų vertino įverčiu didesniu nei 0,9. Likusius 52 % transakcijų įvertino intervale 0,3 – 0,4. Tuo tarpu kontrolinio scenarijaus transakcijos 100 % pateko į intervalą kuriame sukčiavimo tikimybė mažesnė nei 0,1. Taip pat kontrolinio scenarijaus maksimalus sukčiavimo įvertis buvo 0,0053, o vidutinė reikšmė 0,0017. Todėl galima teigti, kad vertinimui apibrėžti rizikingiausi scenarijai buvo sėkmingai įvertinti kaip sukčiavimas. Pateikti sukčiavimo įverčiai buvo daug kartų didesni nei kontrolinio scenarijaus įverčiai ir buvo artimi būtinojo įvykio tikimybei.

Pirmasis scenarijus buvo apibrėžtas kaip rizikingiausias, atitinkantis visus sukčiavimui būdingus šablonus. Antrasis scenarijus buvo sudarytas iš mažiau rizikingų transakcijų. Analizuojant įverčių rezultatus galima pastebėti, kad pirmojo scenarijaus atveju žymiai didesnis procentas transakcijų buvo įvertintas sukčiavimo tikimybe didesne nei 0,9 lyginant su antruoju scenarijumi. Taip pat pirmojo scenarijaus vertinimų vidurkis yra aukštesnis už antrojo scenarijaus vertinimų vidurkį, o mažiausios scenarijų įverčių aibių reikšmės buvo identiškos. Todėl galima teigti, kad sukčiavimo aptikimo sistema teisingai gebėjo tarpusavyje diferencijuoti transakcijų rizikingumą ir rizikingesnio scenarijaus atveju dažniau gražino aukštesnius sukčiavimo įverčius.

Toliau apžvelgiami trečiojo ir ketvirtojo scenarijų rezultatai. Šie scenarijai buvo apibrėžti su mažiau rizikinga elgsena nei pirmasis ir antrasis, tačiau rizikingesni nei kontrolinis scenarijus. Iš scenarijų rezultatų dažnių matoma, kad trečiojo scenarijaus atveju 90 % transakcijų buvo įvertinta sukčiavimo įverčio intervale nuo 0 iki 0,1. Vidutinis trečiojo scenarijaus sukčiavimo įvertis buvo 0,05. Analogiškai ketvirtojo scenarijaus atveju net 98 % transakcijų buvo įvertinta intervale nuo 0 iki 0,1. Vidutinis ketvirtojo scenarijaus transakcijų įvertis – 0,01. Iš šių rezultatų matoma, kad sukčiavimo aptikimo sistema trečiajam – vidutinio rizikingumo scenarijui ir ketvirtajam – mažo rizikingumo scenarijui, pateikė žymiai mažesnius sukčiavimo įverčius nei aukšto rizikingumo scenarijams, tačiau aukštesnius nei kontrolinio scenarijaus. Taip pat svarbu pastebėti, kad pirmojo – aukščiausio rizikingumo scenarijaus įverčiai didžiausi, antrojo scenarijaus įverčiai statistiškai mažesni už pirmojo scenarijaus ir didesni už trečiojo. Trečiojo scenarijaus įverčiai statistiškai mažesni už antrojo scenarijaus ir didesni už ketvirtojo. Ketvirtojo scenarijaus įverčiai statistiškai mažesni už trečiojo ir didesni už kontrolinio scenarijaus. Todėl galima teigti, kad sistema teisingai diferencijuoja skirtingo rizikingumo transakcijas.

Iš pirmo žvilgsnio gali pasirodyti, kad toks rezultatas yra patenkinamas ir sukčiavimo aptikimo sistema užtikrina aukštą tikslumą. Tačiau žvelgiant detaliau tampa aišku, kad sistemos jautrumas vidutiniškai rizikingoms ir mažiau rizikingoms transakcijoms yra netinkamas.

Sukčiavimo aptikimo sistema trečiojo scenarijaus transakcijoms pateikė aukštesnius įverčius nei kontroliniam scenarijui, tačiau šie rezultatai sunkiai atskiriami. Vidutinio rizikingumo scenarijaus atveju 90 % transakcijų įverčių pateko į tą pačią grupę kaip ir kontrolinės transakcijos. Šių transakcijų rizikingumas buvo įvertinas < 0,1. Todėl nepaisant to, kad statistiškai įmanoma įžvelgti skirtumą tarp scenarijų grupių. Gavus transakcijos įvertį mažesnį nei 0,1 būtų beveik neįmanoma nustatyti transakcijos realų rizikingumą, kadangi vidutinio rizikingumo transakcijos menkai skiriasi nuo mažo rizikingumo ir nerizikingų transakcijų. Todėl galima teigti, sukčiavimo aptikimo sistemos jautrumas yra prastas ir naudojant šią aptikimo sistemą, būtų sunku identifikuoti subtilesnius sukčiavimo atvejus, geriau atitinkančius normalius elgsenos šablonus.

Toliau analizuojamas transakcijų įverčių pasiskirstymas jį lyginant ne tarp scenarijų, o bendrai - tarp visų transakcijų. Matoma, kad didžioji dalis rezultatų susikaupė trijuose įverčių intervaluose. Rizikingoms transakcijoms buvo pateikiami sukčiavimo įverčiai tikimybių intervaluose 0,3 – 0,4 ir 0,9 – 1,0. Tuo tarpu vidutinio rizikingumo, mažo rizikingumo bei kontrolinio scenarijaus transakcijų daugiausia pateko į įverčių intervalą 0,0 - 0,1. Analizuojant detalią rezultatų išklotinę (žr. 18 priedas) matoma, kad kartojasi ne tik reikšmių intervalai, tačiau dažniausiai kartojasi ir konkrečios įverčių reikšmės. Todėl galima teigti, kad sistemos jautrumas yra prastas ne tik vidutinio ir mažesnio rizikingumo transakcijoms.

Peržiūrėjus tarpinius transakcijų vertinimo rezultatus nustatyta, kad tokį jautrumo trūkumą lėmė projektinis sprendimas suskirstyti kriterijus tik į 4 kriterijų grupes.

Dėl mažo kriterijų grupių skaičiaus labai rizikingoms transakcijoms 3 – 4 kriterijų grupės identifikuojamos kaip rizikingos. Tuo atveju kai 4 kriterijų grupės įvertinamos aukšta rizika vyrauja sukčiavimo tikimybė didesnė nei 0,95. Tuo atveju kai 3 kriterijų grupės įvertinamos rizikingomis, vyrauja tikimybės intervale 0,3 – 0,6. Tačiau vertinant vidutinio ar mažo rizikingumo transakcijas tik 1 – 2 kriterijų grupės vertinamos kaip rizikingos ir tokiu atveju dauguma įverčių pateikiama intervale 0,0 – 0,1.

Priimant sprendimą naudoti 4 kriterijų grupes buvo daroma prielaida, kad 4 grupės suteiks pakankamą diferenciaciją tarp skirtingo rizikingumo transakcijų. Kiekvienai grupei buvo priskirti 5 rizikingumo įverčiai: „labai mažas rizikingumas“, „mažas rizikingumas“, „tikėtinas rizikingumas“, „didelis rizikingumas“, „labai didelis rizikingumas“. Žvelgiant iš teorinės pusės, turint 4 grupes kuriai reikšmę galime parinkti iš 5 variantų iš viso galima sudaryti 625 skirtingas grupių verčių kombinacijas. Tačiau iš transakcijų vertinimų buvo pastebėta, kad sukčiavimo atvejai koncentruojasi tarp 2 – 3 kriterijų grupių. Dėl to geriausiu atveju lieka 16 – 81 skirtingų kriterijų grupių kombinacijų ir todėl sistema panašaus tipo transakcijoms pateikia tą patį įvertį ir

sunkiai geba apskaičiuoti prasmingus įverčius tais atvejais kai transakcijos rizikingumas nėra akivaizdus.

Todėl galima teigti, kad kriterijų grupavimas buvo tinkamai pasirinkta optimizacija. Nepaisant prasto jautrumo sistema statistiškai teisingai įvertino scenarijų rizikingumą ir labai gerai identifikavo akivaizdžius sukčiavimo atvejus. Vis dėlto siekiant geresnio sistemos jautrumo reikia peržiūrėti sistemos kriterijus bei kriterijų skirstymą į grupes. Norint pagerinti rezultatus reikia suskirstyti kriterijus į daugiau grupių turinčių po mažiau kriterijų ir tai turėtų pagerinti sistemos jautrumą.

3.1.4. Apibendrinimas

Apibendrinant galima teigti, kad sukurta sukčiavimo aptikimo sistema labai gerai geba aptikti akivaizdžius sukčiavimo bei akivaizdžius nesukčiavimo atvejus. Taip pat sistema pateikia lengvai interpretuojamus tarpinius rezultatus. Atlikus tyrimą buvo nustatyta, kad sistema teisingai diferencijuoja transakcijas pagal jų rizikingumą. Rizikingesnių scenarijų įverčiai statistiškai buvo pateikiami didesni už mažiau rizikingų scenarijų įverčius. Todėl galima teigti, kad iš esmės sukčiavimo aptikimo būdas veikia korektiškai.

Tačiau dėl sistemos optimizavimo apibrėžtas nedidelis kriterijų grupių kiekis Bajeso tinkle neigiamai įtakoja sistemos jautrumą. Dėl to iš transakcijai gražinamo įverčio sunku nustatyti ar transakcija yra sukčiavimas, kadangi vidutinio ir mažesnio nei vidutinis rizikingumas transakcijų atveju sukčiavimo įverčiai labai panašūs. Todėl siekiant pagerinti sistemos jautrumą reikėtų pertvarkyti kriterijų grupes Bajeso tinkle ir apibrėžti papildomų kriterijų.

3.2. Rezultatų interpretavimo paprastumas

Sistemos rezultatų interpretavimo paprastumas apibrėžia kaip paprasta sistemos naudotojams nustatyti kodėl sistema pateikė konkretų sukčiavimo įvertį. Rezultatų interpretavimo paprastumas būtinas sukčiavimo aptikimo sistemoje, kadangi kitu atveju be sistemą prižiūrinčių specialistų analizuoti sistemos rezultatų būtų beveik neįmanoma. Bajeso tinklo struktūrą turi apibrėžti srities ekspertai. Todėl galimybė žmonėms suprasti ir interpretuoti tarpinius rezultatus būtina. Suprojektuotoje sukčiavimo aptikimo sistemoje kaip tarpinius rezultatus galima traktuoti apibrėžtų kriterijų bei kriterijų grupių įverčius (žr. 13 priedas).

Iš pateikiamų pavyzdinių tarpinių rezultatų matoma, kad vietos (angl. *location*) kriterijų grupė buvo įvertinta aukštos rizikos verte. Norint nustatyti, kodėl buvo gautas toks rizikos įvertis, interpretuojami grupės kriterijų rezultatai. Šiuo konkrečiu atveju rizikos įvertis gautas, nes pagal transakcijos duomenis sistema identifikavo, kad šioje teritorijoje dažniau nei įprasta

vyksta sukčiavimas[10]. Taip pat atstumas nuo paskutinės transakcijos atlikimo vietos yra didesnis nei tikėtina, o atstumas nuo dažniausiai naudojamos teritorijos yra mažesnis nei tikėtina. Sukčiavimo tikimybė yra skaičiuojama pagal tai kaip dažnai kiekviena konkreti kriterijaus reikšmė sutinkama sukčiavimo atvejuose. Todėl turint šias vertes ir iš tinklo duomenų saugyklos, analitinės užklausos pagalba, gavus kriterijų verčių tikimybinius rodiklius, analitikas pats be sistemos pagalbos galėtų pertikrinti skaičiavimus ir išanalizuoti situaciją. Iš šio pavyzdžio matoma, kad pateikiami tarpiniai rezultatai galbūt ir nėra trivialūs tačiau pakankamai paprastai interpretuojami susipažinus su naudojamo Bajeso tinklo struktūra.

3.3. Efektyvumo vertinimas

Sukčiavimo aptikimo sistemos efektyvumo vertinimo tikslas – nustatyti ar sistema yra tinkama naudoti realių transakcijų apdorojimui. Sukčiavimo aptikimo sistemai buvo apibrėžti nefunkciniai reikalavimai maksimaliam atsako laikui bei sistemos pralaidumui. Sistema turi pateikti sukčiavimo įvertį greičiau nei per 1 sekundę. Tuo tarpu orientacinis aptikimo sistemos pralaidumas buvo pasirinktas 10 000 įvertinamų transakcijų per sekundę.

Norint atlikti bandymus rekomenduojamoje sistemos diegimo konfigūracijoje reikėtų daug brangių infrastruktūrinių resursų, todėl siekiant įvertinti sistemos efektyvumą buvo atlikti du bandymai skirtingomis diegimo konfigūracijomis su skirtingais resursų kiekiais. Šie bandymai padės įvertinti sistemos atitikimą reikalavimams bei bus naudingi norint įvertinti sistemos horizontalaus ir vertikalaus plėtimo galimybes.

3.3.1. Bandymų aplinka

Duomenų paieška, atnaujinimai labai mažuose duomenų rinkiniuose atliekami daug greičiau nei duomenimis užpildytoje sistemoje. Todėl tikėtina, kad sistema darbo pradžioje neužpildžius jos pakankamu duomenų kiekiu veiktų žymiai sparčiau ir greičiausiai galėtų užtikrinti aukštesnį pralaidumą. Siekiant išvengti tyrimo rezultatų iškraipymo prieš atliekant bandymus sistema užpildyta transakcijų rinkiniu iš 10 000 000 istorinių transakcijų. Priešingai nei tikslumo bandymų atveju visos šios transakcijos buvo sugeneruotos neatsižvelgiant į elgsenos šablonus.

Siekiant išvengti efektyvumo tyrimo rezultatų iškraipymo bandymų metu įverčiai buvo validuojami. Kiekvienos vertinamos transakcijos atveju buvo įsitikinama, kad naudojama sukčiavimo aptikimo sistemos programinė sąsaja gražina sėkmės statusą. Taip pat transakcija buvo laikoma sėkmingai įvertinta tik tuo atveju, kai pateikiamas sukčiavimo įvertis yra realusis skaičius. Atliekant efektyvumo bandymus konkrečios įverčio reikšmės analizuojamos nebuvo.

[10] LOCATION_RISK kriterijus apibrėžia kaip dažnai teritorijoje sutinkamas sukčiavimas.

Vykdant bandymus buvo užtikrinta maksimali bandymo aplinkos izoliacija nuo bet kokių išorinių veiksnių. Siekiant sumažinti naudojamos infrastruktūros įtaką bandymų rezultatams, bandymai atlikti naudojant dedikuotus serverius. Atliekant šiuos didelio resursų kiekio reikalaujančius bandymus asmeninio kompiuterio pagalba, rezultatus gali iškraipyti įvairūs kompiuteryje veikiantys foniniai procesai: antivirusinė programa, operacinės sistemos naujinimų tikrinimas ir kt. Taip pat visus bandymus atliekant viename fiziniame įrenginyje kyla rizika, kad naudojamas apkrovos generatorius, kuriam reikalinga žymi sistemos resursų dalis, iškreips tyrimo rezultatus. Dėl to, visi bandymai buvo atliekami dedikuotuose serveriuose imituojančiuose realios sistemos aplinką. Tyrimams atlikti buvo naudojami serveriai su šviežiai įdiegta „Ubuntu 16.04“ operacine sistema. Prieš atliekant bandymus serveriuose buvo įdiegta tik bandymų atlikimui reikalinga programinė įranga. Taip užtikrinta, kad bandymo aplinkoje būtų minimalus pašalinės sukčiavimo aptikimo sistemos ar operacinės sistemos darbui užtikrinti nebūtinos programinės įrangos kiekis. Norint izoliuoti naudojamą apkrovos generatorių ir išvengti rezultatų iškraipymo, generatorius visais atvejais veikė fiziškai atskirame serveryje. Apkrovos generavimo metu buvo vykdomas serverio resursų stebėjimas. Taip užtikrinant, kad apkrovos generatoriui pakanka sisteminių resursų. Taip pat vykdant skirtingus bandymus buvo siekiama užtikrinti kuo panašesnes komponentų bendravimo per tinklą sąlygas. Todėl buvo naudojamas virtualių serverių tiekėjo ypač spartus vidinis tinklas veikiantis tarp šio tiekėjo komponentų. Visi dedikuoti virtualūs serveriai bandymams buvo nuomojami iš tiekėjo „Digital Ocean“.

Atliekant efektyvumo bandymus apkrovai generuoti buvo naudojamas įrankis „Gatling“. Kiekvieno bandymo metu buvo sudaromas apkrovos generavimo scenarijus palaipsniui didinantis apkrovą, tam kad surasti maksimalų sistemos užtikrinamą pralaidumą.

Svarbu atsižvelgti, kad apkrovos generavimui naudojant nedidelį unikalių transakcijų kiekį išlieka tikimybė, kad sistemos efektyvumas bus iškreiptas. Nuolatos vertinimus atliekant toms pačioms transakcijoms tarpiniai rezultatai gali išlikti išsaugoti įvairių lygių podėliuose. Dėl to, siekiant realų sistemos naudojimą atitinkančių rezultatų buvo sugeneruotas duomenų rinkinys iš 50 000 skirtingų transakcijų. Šį duomenų rinkinį apkrovos generatorius naudojo kaip įeigą sukčiavimo aptikimo įverčio skaičiavimo užklausoms generuoti.

3.3.2. Efektyvumas vertinimas su mažu resursų kiekiu

3.3.2.1. Bandymo aplinka

Pirmasis bandymas buvo atliekamas išnaudojant pakankamai nedidelį resursų kiekį. Bandymo aplinka sudaryta iš 3 serverių turinčių 4 procesoriaus branduolius bei po 8 GB operatyviosios atminties. Vienas iš serverių buvo naudojamas apkrovos generatoriui izoliuoti, todėl realiai sukčiavimo aptikimo sistema veikė 2 serverių aplinkoje. Vienas serveris buvo

skirtas sistemos programinio įgyvendinimo daliai, kitas duomenų bazei (žr. 19 priedas). Tyrimo metu buvo naudojamas iš anksto sugeneruotų transakcijų duomenų rinkinys (žr. 21 priedas).

3.3.2.2. Bandymo eiga

Bandymo metu buvo matuojamas sukčiavimo įverčių pateikimo laikas, per sekundę pateikiamų užklausų kiekis, per sekundę apdorojamų užklausų kiekis bei aktyvių sesijų laukiančių įverčio apskaičiavimo kiekis. Svarbu pastebėti, kad šio tyrimo kontekste, aktyvių sesijų kiekis apibrėžia užklausų kiekį, kuris einamuoju momentu jau buvo pateiktas įvertinimui, tačiau dar nebuvo įvertintas. Idealiu atveju aktyvių sesijų kiekis per sekundę turėtų sutapti su pateikiamų užklausų įvertinimui kiekiu. Aktyvių sesijų kiekis ima augti kai užklausas apdorojančioji sistema nebespėja pakankamai greitai apdoroti užklausų. Galiausiai transakcijos apdorojimui ima kauptis

Prieš pradedant bandymą kurio metu atliekami matavimai, buvo atliekamas sistemos streso testas (angl. *stress test*). Šis testas padėjo identifikuoti sistemos lūžio tašką ir rasti sistemos pralaidumo ribas prieš sistemos lūžimą kuriose prasminga atlikti bandymą.

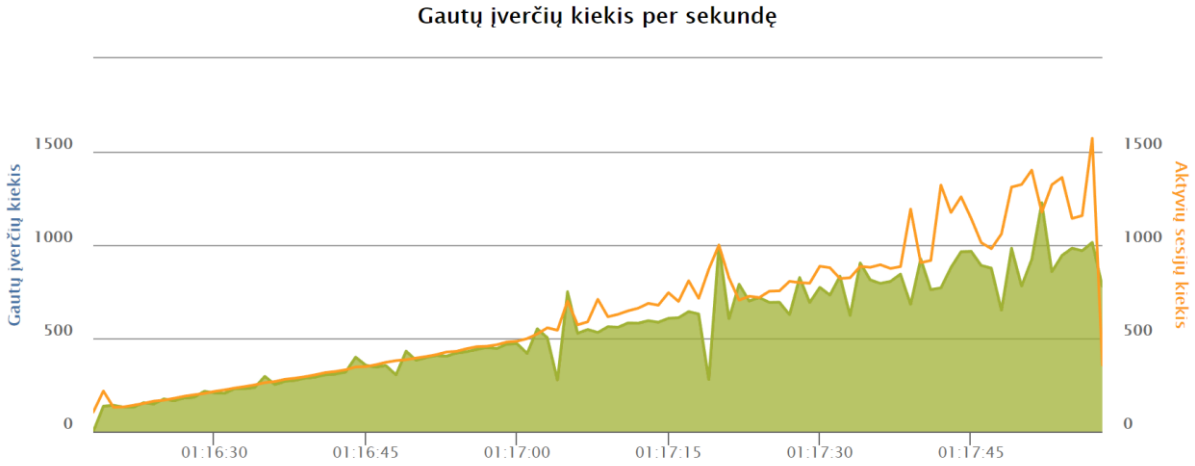
Bandymo metu sistemos apkrova buvo didinama nuo 100 užklausų per sekundę iki 1000 užklausų per sekundę. Iš viso bandymo metu sistema turėjo apdoroti 55 000 užklausų. Siekiant išvengti nenumatytų situacijų buvo apsibrėžtas 5 sekundžių laukimo laiko intervalas. Tai reiškia, kad apkrovos generatorius laukia atsako, jeigu atsakas per 5 sekundes nėra pateikiamas, tokią transakcija rezultatuose žymima kaip neįvertinta.

3.3.2.3. Bandymo rezultatai

Vykdant sistemos apkrovos testavimą tarp 100 ir 1000 aktyvių užklausų per sekundę visoms užklausoms buvo sėkmingai pateiktas sukčiavimo įvertis. Nei viena iš užklausų nebuvo klasifikuojama kaip neįgyvendinta dėl sisteminės klaidos ar ilgesnio nei 5 sekundės apdorojimo laiko. Iš atsako laiko pasiskirstymo (žr. 20 priedas) matoma, kad iš pateiktų 55 000 transakcijų, 54 039 transakcijos, kurios sudaro 98,3 % visų transakcijų, buvo įvertintos greičiau nei per 0,8 sekundės. Taip pat nustatyta, kad tik 1,1 % visų transakcijų atvejų apdorojimas truko tarp 0,8 sekundės ir 1,2 sekundės. Likusių 0,6% transakcijų apdorojimas truko ilgiau nei 1,2 sekundės. Projektuojant sistemą buvo apibrėžtas nefunkcinis reikalavimas teigiantis, kad sistema turi gebėti įvertinti transakcijas greičiau nei per 1 sekundę. Formaliai sistema reikalavimo neįgyvendino. Tačiau buvo tiriamas ypač primityvus sistemos diegimo variantas. Tuo tarpu pasiektas rezultatas, kad daugiau nei 98 % transakcijų įvertis buvo pateiktas pakankamai greitai ir atitiko formalų reikalavimą, o 99% transakcijų įvertis buvo pateikiamas per labai artimą laiko nuokrypį formaliam reikalavimui. Todėl galima teigti, kad nors teoriškai formalus reikalavimas nėra įgyvendinamas, praktiškai sistemos atsako laikas yra labai artimas reikalavimui ir pakankamas pritaikymui realioje aplinkoje.

Toliau apžvelgiami pateiktų apdorojimui užklausų kiekio bei gautų įverčių kiekio per sekundę rezultatai. Pateiktų apdorojimui užklausų kiekis tiesiškai kilo nuo 100 pateikiamų užklausų per sekundę iki 1000 pateikiamų užklausų per sekundę.

Sistemos veikimą bandymo metu galima suskirstyti į 3 fazes (žr. 4 pav.): sistemos veikimas apdorojant mažiau nei 550 užklausų per sekundę, sistemos veikimas apdorojant tarp 550 ir 900 užklausų per sekundę, sistemos veikimas apdorojant tarp 900 ir 1000 užklausų per sekundę.



4 pav. Įvertintų transakcijų kiekis per sekundę nedidelėje konfigūracijoje

Iš bandymo rezultatų matoma, kad sistemai apdorojant iki maždaug 550 užklausų per sekundę tiek gaunamų įverčių kiekis kyla beveik tiesiškai. Taip pat kartu su pateiktų užklausų kiekiu tiesiškai kyla ir aktyvių sesijų kiekis. Todėl galima teigti, kad šiuo atveju sistema pilnai spėja apdoroti visas pateiktas užklausas. Kadangi nėra apdorojimo laukiančių transakcijų, galima teigti, kad greičiausiai apdorojant mažiau nei 550 transakcijų per sekundę visos transakcijos buvo apdorotos greičiau nei per 0,8 sekundės.

Peržengus 550 pateikiamų transakcijų per sekundę ribą matoma, kad sistema ima artėti prie savo pralaidumo galimybių ribos. Intervale tarp 550 – 900 pateikiamų transakcijų per sekundę kiekvienu laiko momentu aktyvių sesijų kiekis buvo lygus apie 110 % įvertinimui pateikiamų užklausų kiekio. Tai reiškia, kad laukiančių transakcijų kiekis bet kuriuo laiko momentu apie 10 %. Nepaisant padidėjusio laukiančiųjų transakcijų kiekio sistema su tokiu pralaidumu tinkamai susitvarkė. Sistemos veikimas nebuvo pažeistas, visos transakcijos buvo apdorotos pakankamai greitai. Svarbu pastebėti, kad laukiančiųjų transakcijų kiekis daugiau nedidėjo, o stabilizavosi ties 10 % riba. Tai reiškia kad sistema yra įgali susidoroti su tokiu transakcijų kiekiu.

Žvelgiant į paskutinę vykdymo fazę, kai apdorojimui pateikiama 900 – 1000 transakcijų per sekundę matoma, kad aktyvių sesijų kiekis dar labiau išauga. Taip pat aktyvių sesijų kiekio

stabilumas mažėja. Laukiančių transakcijų kiekis ima svyruoti nuo 10 % - 30 % intervale. Taip pat iš sistemos gautų įverčių kiekis nebekyla tiesiškai, o ima svyruoti ribose tarp 900-1200 pateiktų įverčių per sekundę. Kadangi šioje tyrimo fazėje nemažai transakcijų laukė eilėje iki kol buvo apdorotos, tikėtina, kad būtent šioms laukiančioms transakcijoms sukčiavimo įvertinimas truko ilgiau nei 0,8 sekundės. Todėl galima teigti, kad paskutinėje bandymo fazėje buvo priartėta prie sistemos galimybių ribos šioje konkrečioje sistemos konfigūracijoje. Sistema gali sėkmingai užtikrinti 900 – 1000 apdorojamų užklausų per sekundę, tačiau tokiu atveju dalies užklausų apdorojimas gali trukti ilgiau nei sekundę. Kadangi šioje tyrimo fazėje nebuvo požymių, kad transakcijos kaupiasi ir jų nespėjama apdoroti. Galima teigti, kad sistema gali užtikrinti pakankamą stabilumą apdorojant iki 1000 transakcijų per sekundę. Tuo tarpu, sistema turėtų būti visiškai stabili ir laiku pateikti visus sukčiavimo įverčius apdorojant iki 900 užklausų per sekunde.

3.3.2.4. Apibendrinimas

Apibendrinant galima teigti, kad buvo atliktas efektyvumo tyrimas sistemai veikiant labai primityvioje konfigūracijoje, kurią galima sulyginti su tipinio galingo asmeninio kompiuterio aplinka. Šioje aplinkoje sukčiavimo aptikimo sistema užtikrino pakankamai trumpą atsako laiką, kadangi daugiau nei 98 % įvertintų transakcijų atveju atsako laikas buvo trumpesnis nei 0,8 sekundės. Taip pat šios aplinkos resursų pakako, kad sistema stabiliai veikdama užtikrintų iki 1000 transakcijų per sekundę įverčio skaičiavimo pralaidumą.

3.3.3. Efektyvumas su vidutiniu resursų kiekiu

3.3.3.1. Bandymo aplinka

Bandymas buvo atliekamas naudojant kiek didesnį resursų kiekį bei sudėtingesnę sistemos konfigūraciją. Bandymo aplinka buvo užtikrinama 5 virtualių serverių turinčių po 16 procesoriaus branduolių bei po 64GB operatyviosios atminties. Vienas iš serverių buvo naudojamas apkrovos generatoriui izoliuoti, todėl realiai sukčiavimo aptikimo sistema veikė išnaudodama 4 serverių resursus. Vienas serveris buvo skirtas suprogramuotai sukčiavimo aptikimo sistemos daliai. Likę 3 serveriai skirti duomenų bazėms (žr. 19 priedas). Pirmasis serveris skirtas tinklo duomenų saugyklos komponento duomenų bazei. Antrasis serveris skirtas transakcijų saugyklos komponento duomenų bazei. Trečiasis serveris skirtas transakcijų saugyklos, bendrosios periodinės statistikos esybės saugojimui. Pirmojo bandymo metu, buvo stebimi serveriuose naudojami resursai kiekvienos esybės operacijoms. Buvo nustatyta, kad bendrosios periodinės statistikos esybės tarpinių rezultatų atnaujinimas reikalauja ypač daug sisteminių resursų. Todėl buvo nuspręsta bendrąją periodinę statistiką iškelti į atskirą fizine mašiną.

Šio bandymo metu taip pat buvo naudojamas iš anksto sugeneruotų transakcijų duomenų rinkinys (žr. 21 priedas).

3.3.3.2. Bandymo eiga

Kaip ir pirmojo bandymo atveju, buvo matuojamas sukčiavimo įverčių pateikimo laikas, pateikiamų užklausų per sekundę kiekis, apdorojamų užklausų per sekundę kiekis bei aktyvių sesijų laukiančių įverčio apskaičiavimo kiekis. Prieš atliekant bandymą atliktas sistemos streso testas (angl. *stress test*). Šis testas padėjo identifikuoti sistemos lūžio tašką ir rasti sistemos pralaidumo ribas, kuriose prasminga atlikti bandymą.

Atliekant bandymą buvo įvykdyti 2 apkrovos testai. Pirmajame apkrova buvo didinama nuo 1000 užklausų per sekundę iki 2000 užklausų per sekundę. Antrajame apkrova didinama nuo 2000 užklausų per sekundę iki 3000 užklausų per sekundę. Iš viso bandymų metu sistema turėjo apdoroti atitinkamai 135 000 ir 300 000 užklausų.

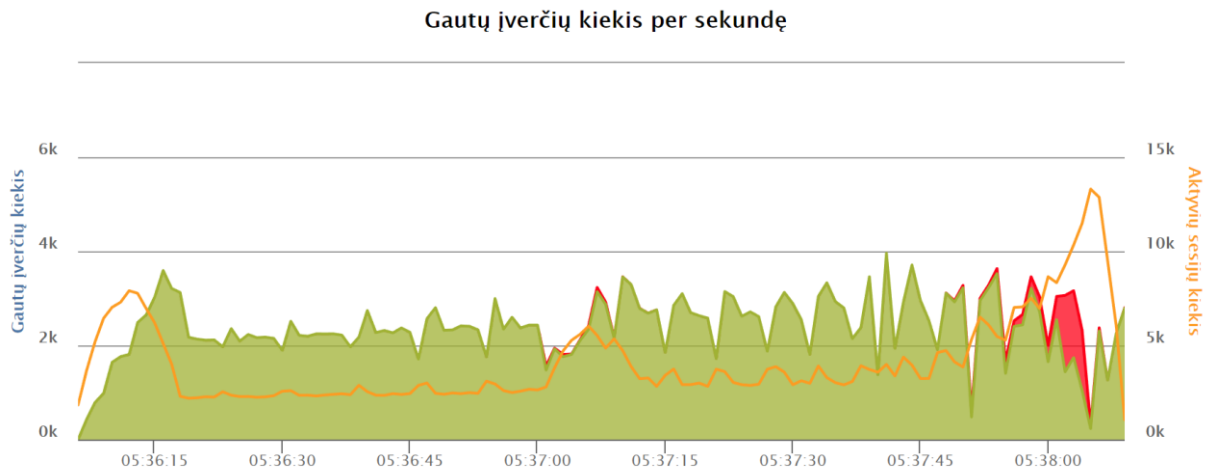
Siekiant išvengti nenumatytų situacijų apklausos generatoriuje buvo apibrėžtas maksimalus 5 sekundžių laukimo laikas. Jeigu sistema nepateikia atsako per 5 sekundes, transakcija laikoma neįvertinta.

3.3.3.3. Bandymo rezultatai

Pirmiausia atliktas bandymas apdorojimui pateikiant iki 2000 užklausų per sekundę. Iš viso šio bandymo metu apdorota 135 000 transakcijų. Iš atsako laiko pasiskirstymo (žr. 20 priedas) matoma, kad 129 960 atvejų sukčiavimo įvertis pateiktas greičiau nei per 0,8 sekundės. Tai sudaro 96,1% visų įvertintų transakcijų. Greičiau nei per 1,2 sekundės sukčiavimo įvertis buvo pateiktas 130 323 atvejų. Tai sudaro 98,7 % visų įvertintų transakcijų. Matoma, kad kaip ir pirmojo efektyvumo bandymo atveju, reikalavimas visas transakcijas įvertinti per 1 sekundę, nėra formaliai įgyvendinamas. Tačiau ši sistemos konfigūracija leido pasiekti rezultatą, kai beveik 99 % transakcijų buvo įvertina su 0,2 sekundės nuokrypiu nuo formalaus reikalavimo. Taip pat tik apie 1 % transakcijų buvo įvertinta žymiai lėčiau nei 1,2 sekundės. Todėl nors sistema formaliai reikalavimo ir neįgyvendino galima teigti, kad sistema 99 % atvejų gali pateikti realioje aplinkoje pakankamą atsako laiką.

Toliau buvo vykdomas bandymas vertinimui pateikiant 2000 – 3000 transakcijų per sekundę. Iš viso šio bandymo metu vertinimui buvo pateikta 300 000 transakcijų. Šis bandymas peržengė sistemos pralaidumo ribas. Iš 300 000 transakcijų, 7300 arba 2,4 % visų transakcijų buvo neįvertintos sistemoje arba jų įvertinimo laikas truko ilgiau nei 5 sekundes ir atsako laukimas buvo nutrauktas. Net 46 081 arba 24% visų transakcijų apdorojimas truko ilgiau nei 1,2 sekundės. Iš to akivaizdu, kad šio bandymo metu sistemos pralaidumo ribos buvo peržengtos. Todėl šio bandymo atsako laiko rezultatai nebus detaliau analizuojami.

Analizuojant bandymų pralaidumo rezultatus, galima teigti, kad pirmojo bandymo metu pralaidumo ribos nebuvo pasiektos, todėl pralaidumo kontekste šis bandymas nebus analizuojamas. Antrojo bandymo metu sistemai pateikiamas užklausų kiekis buvo didinamas nuo 2000 per sekundę iki 3000 per sekundę. Šį bandymą galima suskirstyti į 3 bandymo fazes pagal pateikiamų užklausų kiekį (žr. 5 pav.): pateikiama 2000 – 2500 užklausų per sekundę, pateikiama 2500 – 2750 užklausų per sekundę, pateikiama 2750 – 3000 užklausų per sekundę.



5 pav. Įvertintų transakcijų kiekis per sekundę vidutinėje konfigūracijoje

Pirmojoje bandymo fazėje pateikiama 2000 - 2500 užklausų per sekundę. Aktyvių užklausų kiekis didėja tiesiškai, kartu su beveik tiesiškai juo didėja aktyvių sesijų kiekis. Aktyvių sesijų kiekis sudaro apie 110 % pateikiamų per sekundę užklausų. Tai reiškia, kad bet kuriuo laiko momentu apie 10 % pateiktų užklausų laukia apdorojimo. Užklausos laukiančios apdorojimo nesikaupia, sistemos neveiksnumo požymių nesimato. Galima teigti, kad šiuo rėžimu sistema veikia stabiliai.

Antrojoje bandymo fazėje pateikiama 2500 – 2750 užklausų per sekundę. Pirmiausia verta atkreipti dėmesį, kad vos peržengus 2500 pateikiamų užklausų per sekundę ribą, bandymo metu apdorojamų įverčių kiekis nukrenta iki 1500 įverčių per sekundę. Sistemoje trumpam pakyla apdorojimo laukiančių transakcijų kiekis. Taip pat nedidelis kiekis transakcijų dėl laukimo identifikuojamos kaip nepavykusios. Peržengus šią ribą sistemos darbas stabilizuojasi ir sistema toliau intervale iki pat 2750 užklausų per sekundę sėkmingai pateikia įverčius. Tačiau svarbu pastebėti, kad šioje fazėje apdorojimo laukiančių transakcijų kiekis išauga nuo 10 % procentų iki maždaug 30 %. Vis dėlto, nepaisant padidėjusio laukiančiųjų kiekio, kiekis daugiau nedidėja, pateikiamos transakcijos sėkmingai apdorojamos per numatytą laiką. Šiame intervale sistema buvo trumpam praradusi stabilumą, tačiau sistemos darbas stabilizavosi. Todėl galima teigti, kad sistema negali apdoroti daugiau nei 2500 transakcijų per sekundę išlaikydama visišką stabilumą.

Tačiau esant poreikiui sistema galėtų prisitaikyti netikėtai padidėjusio užklausų kiekio ir užklausų kiekiui sumažėjus stabilizuoti situaciją.

Trečioje bandymo fazėje pateikiama 2750 – 3000 transakcijų per sekundę. Šioje fazėje ima sparčiai augti laukiančių apdorojimo transakcijų kiekis. Pasiekus 2800 transakcijų per sekundę ribą laukiančiųjų transakcijų kiekis sistemoje padidėja iki maždaug 4000 transakcijų. Atėmus 2800 šviežiai pateiktų transakcijų gaunama, kad susikaupė 1200 neapdorotų transakcijų. Pateikiamų transakcijų kiekiui toliau didėjant iki 2900 pateikiamų transakcijų per sekundę, laukiančių apdorojimo transakcijų kiekis pasiekia 6000. Tai reiškia, kad tuo laiko momentu jau 3100 transakcijos laukė apdorojimo. Pasiekus 3000 per sekundę pateikiamų apdorojimui transakcijų ribą visų transakcijų rodiklis išauga iki 10 000 transakcijų, arba 7000 laukiančių neapdorotų transakcijų iš praeities. Akivaizdu kad peržengus 2750 ribą sistema nebegali apdoroti nuolat didėjančio transakcijų kiekio sistemoje. Todėl sistemoje kaupiasi apdorojimo laukiančios transakcijos. Šių transakcijų kiekis nebesistabilizuoja, tik didėja. Žvelgiant į gautų įverčių rezultatus matoma, kad tuo pat metu apkrovos generatorius identifikavo pirmąsias neapdorotas transakcijas. Sistema dalinai pateikinėjo įverčius kol galiausiai pasiekus 2900 pateikiamų transakcijų per sekundę sistemos pateikiamų įverčių kiekis krito. Dauguma aktyvių transakcijų buvo įvertintos kaip klaidingos dėl sisteminių klaidų. Iš šios fazės bandymo rezultatų aišku, kad sukčiavimo aptikimo sistema šioje infrastruktūroje pasiekusi didesnę nei 2750 apdorojamų transakcijų per sekundę ribą prarado stabilumą. Todėl sistema negali būti naudojama įvertinti siekiant įvertinti daugiau nei 2750 transakcijų per sekundę.

3.3.3.4. Apibendrinimas

Apibendrinant bandymo rezultatus galima teigti, kad suteikus papildomų resursų sistemos serveriams ir išskirsčius duomenų bazę į kelis papildomus serverius, pavyko pasiekti 2500 apdorojamų transakcijų per sekundę ir išlaikyti sistemos stabilumą. Buvo nustatyta, kad padidėjus sistemos pralaidumui, sistema vis dar gali 99 % atvejų pateikti atsako laiką greičiau nei per 1,2 sekundės. Tuo tarpu 98 % atvejų yra pateikiami greičiau nei per 0,8 sekundės. Todėl nors sistema formaliai ir netenkina apibrėžtų nefunkcinių reikalavimų, sistemos užtikrinamas atsako laikas pakankamas, kad būtų galima 99 % transakcijų įvertinti realiu laiku.

3.3.4. Apibendrinimas

Apibendrinant galima teigti, kad sukurta sukčiavimo aptikimo sistema gali užtikrinti pakankamai trumpą atsako laiką daugumai apdorojamų transakcijų. Bandymo metu buvo užfiksuota, kad sukčiavimo aptikimo sistema neperžengus sistemos pralaidumo ribų apie 98 % transakcijos įverčių pateikė greičiau nei per 0,8 sekundės. Taip pat net 99 % įverčių pateikiami trumpiau nei per 1,2 sekundės. Tai pakankamai nedidelis atsako laiko nuokrypis nuo reikalavimuose apibrėžtos 1 sekundės tikslo. Įvertinus sistemos pralaidumo rodiklius buvo

nustatyta, kad naudojant visiškai primityvų sistemos diegimo variantą išnaudojant 2 vidutinio pajėgumo serverius sistema sėkmingai pasiekė 1000 transakcijų per sekundę pralaidumą ir užtikrino visiška sistemos stabilumą. Tuo tarpu padidinus serveriams išskiriamų resursų kiekį ir išskaidžius duomenų bazę per papildomus 2 serverius pralaidumas išaugo iki 2500 transakcijų per sekundę tuo pat metu išlaikant sistemos stabilumą.

Atlikus bandymus izoliuotoje aplinkoje nepavyko pasiekti nefunkciniuose reikalavimuose apibrėžto pralaidumo, tačiau negalima teigti, kad suprojektuota sukčiavimo aptikimo sistema yra netinkama realaus laiko sukčiavimo aptikimui. Sistema galinti įvertinti 2500 transakcijas per sekundę, per valandą galėtų įvertinti iki $2500 * 3600 = 9\,000\,000$ transakcijų. Svarbu nepamiršti, kad šį rezultatą pavyko pademonstruoti sistemoje su sąlyginai primityvia sistemos diegimo konfigūracija. Įgyvendinus 7 rekomenduojama sistemos diegimo konfigūraciją (žr. 2.4.6 poskyris), sistema turėtų užtikrinti kelis kartus didesnį pralaidumą ir peržengti 10 000 per sekundę apdorojamų transakcijų ribą.

3.4. Apibendrinimas

Apibendrinant galima teigti, kad pavyko sukurti sukčiavimo aptikimo sistemą gebančią aptikti akivaizdžius sukčiavimo atvejus bei teisingai diferencijuojančią transakcijas pagal jų rizikingumą. Taip pat sukurta sukčiavimo aptikimo sistema užtikrina lengvai interpretuojamų tarpinių rezultatų pateikimą. Vis dėlto, dėl pasirinktų naudojamo Bajeso tinklo optimizacijų sistema negali pateikti tinkamo jautrumo. Todėl siekiant pakankamo sistemos tikslumo realioje aplinkoje reikėtų peržiūrėti kriterijų grupavimą bei įvesti papildomų aptikimo kriterijų.

Ištyrus suprojektuotos sukčiavimo aptikimo sistemos efektyvumą sąlyginai primityviose diegimo konfigūracijose nustatyta, kad naudojant skirtingas sistemos diegimo konfigūracijas, sistema 99 % atvejų pateikia sukčiavimo įvertį greičiau nei per 1,2 sekundės šiek tiek nukrypstant nuo apsibrėžto 1 sekundės reikalavimo. Atliekant sistemos pralaidumo bandymus labai mažos apimties diegimo konfigūracijoje sistema sėkmingai pasiekė 1000 apdorojamų transakcijų per sekundę pralaidumą ir užtikrino visiška sistemos stabilumą. Padidinus sistemai skiriamų resursų kiekį ir pridėjus papildomus 2 duomenų bazės serverius pralaidumas išaugo iki 2500 patikimai apdorojamų transakcijų per sekundę. Efektyvumo tyrimo metu nepavyko pasiekti nefunkciniuose reikalavimuose apibrėžiamo 10 000 apdorojamų transakcijų per sekundę pralaidumo. Tačiau naudojant sąlyginai paprastas sistemos diegimo konfigūracijas, pavyko pasiekti aukštus pralaidumo rezultatus. Todel įgyvendinus sistemos diegimo konfigūracijos rekomendacijas, sistema turėtų atitikti keliamus pralaidumo reikalavimus.

Rezultatai ir išvados

Pirmiausia buvo atlikta mokėjimų saugumo analizė. Nustatyta, kad egzistuoja žinomi sukčiavimo metodai išnaudojantys saugumo spragas protokoluose bei techninėje įrangoje. Todėl, nepaisant taikomų saugumo priemonių, sukčiavimas bekontakčių mokėjimų atveju yra reali grėsmė. Prieita prie išvados, kad sukčiavimo aptikimo sistemos suteikia papildomą saugumą be protokolų ir rinkoje esančių įrenginių atnaujinimų.

Buvo išanalizuotas sukčiavimo aptikimo procesas. Nustatyta, kad duomenų gavybos metodai paremti klasifikavimu geba apdoroti daugelio dimensijų duomenis, prisitaikyti prie tikėtinų elgsenos šablonų. Todėl prieita prie išvados, kad duomenų gavyba paremta klasifikavimu yra vienas geriausiai sukčiavimo aptikimo įgyvendinimui tinkančių metodų. Išnagrinėjus duomenų gavybos procesus nustatyta, kad siekiant sukčiavimo aptikimo sistemos tikslumo būtinas išankstinis duomenų apdorojimas. Reikia identifikuoti aptikimui tinkamus atributus, pasirinkti agreguojamus sudėtinius atributus, atlikti duomenų normalizavimą, duomenų segmentavimą. Šios analizės metu taip pat prieita prie išvados, kad nors anksčiau sukčiavimo aptikimas buvo vykdomas kaip periodinis procesas po transakcijos patvirtinimo, aptikimas realiu laiku leistų sumažinti finansinius praradimus ir supaprastinti atsiskaitymo procedūras.

Atlikus sukčiavimo aptikimo proceso analizę buvo ištirti akademinėje literatūroje pateikiami sukčiavimo aptikimo metodai. Nustatyta, kad sukčiavimo aptikimas nėra tiriamas ir analizuojamas kaip programų sistemos kūrimo uždavinys. Tyrimai koncentruojasi tik į tam tikrą sistemos realizacijos dalį: duomenų apdorojimą, įverčių skaičiavimą ar naudojamą duomenų gavybos metodą. Vis dėlto, vienas svarbiausių sukčiavimo aptikimui keliamų reikalavimų yra efektyvumas. Tačiau norint užtikrinti sukčiavimo įverčio pateikimą realiu laiku būtina atsižvelgti į aptikimą įgyvendinančios sistemos kontekstą. Palyginus literatūroje apibrėžtas sukčiavimo aptikimo sistemų realizacijas, prieita prie išvados, kad naudojantis Bajeso tikimybiniais tinklais galima sukurti efektyvią ir tikslią sukčiavimo aptikimo sistemą.

Sukčiavimo aptikimo įgyvendinimui pasirinkus Bajeso tinklus buvo išanalizuoti jų veikimo principai. Ištirti du skirtingi tinklo pavidalai. Nustatyta, kad Bajeso tinklo duomenų saugyklos resursų poreikis auga eksponentiškai, tačiau kriterijų grupavimas į potinklius reikšmingai sumažina reikalingų duomenų saugyklų resursų kiekį. Tiriant tinklo pavidalus prieita prie išvados, kad dėl transakcijų vykdymo protokolų griežtumo, sukčiavimo aptikimui pilnas Bajeso tinklas nėra būtinas.

Naudojantis Bajeso tinklo sudarymo principais buvo apibrėžti du sukčiavimo aptikimo tinklai: pasiūlytas sukčiavimo įverčio apskaičiavimo būdas, pasiūlytas sistemos apmokymo būdas. Ištyrus pasiūlytų tinklų įgyvendinamumą nustatyta, kad tinklo sukonkretinimas leidžia sukurti efektyvesnį sukčiavimo aptikimo būdą. Sukonkretinus aptikimo tinklo kriterijų grupes

gaunamas efektyvus sukčiavimo aptikimo būdas, kuriam pakanka polinomiškai augančio operacijų kiekio.

Apibrėžus siūlomo naudoti tinklo struktūrą atlikta bazinių transakcijos atributų analizė. Apibrėžti išvestiniai transakcijų kriterijai leidžiantys įvertinti elgsenos šablonus. Pasiūlyti kriterijai galintys įvertinti transakcijos sumos, transakcijos laiko, transakcijų kiekio bei transakcijos vietos rizikingumą. Kiekvienam iš kriterijų buvo apibrėžtas būdas normalizuoti reikšmes bei suskirstyti jas į kategorijas tinkamas išreikšti tinklo būsenomis. Naudojant šiuos sukčiavimo kriterijus ir pasirinktą tinklo struktūrą apibrėžtas sukčiavimo aptikimo tinklas.

Apibrėžus sukčiavimo aptikimo tinklą pereita prie sistemos projektinių sprendimų. Apibrėžti reikalavimai sukčiavimo aptikimo sistemai. Atsižvelgiant į reikalavimus identifikuotos sistemos funkcijos bei loginiai sistemos komponentai įgyvendinantys šias funkcijas. Išanalizuotos techninių sprendimų alternatyvos, parinkti komponentų integracijos, realizacijos sprendimai. Išanalizavus komponentų integracijos būdus sukčiavimo aptikimo sistemos komponentų integracijai buvo pasiūlyta naudoti duomenų perdavimą eilėmis. Pasiūlyta išskirstytųjų skaičiavimų strategija leidžianti išskirstyti duomenų atnaujinimo procesus. Apibrėžtas podėlio naudojimas sistemoje. Taip pat buvo identifikuotos tinklo saugyklos esybės, pasiūlytas būdas apibrėžti lengvai pildomą kriterijų konfigūraciją sistemoje. Atsižvelgiant į apibrėžtą architektūrinį sprendimą buvo įgyvendinta sukčiavimo aptikimo sistema ir pagal nefunkcinius reikalavimus pasiūlyta sistemos diegimo konfigūracija.

Įgyvendinus sistemą buvo atliktas sukčiavimo aptikimo tikslumo bei efektyvumo tyrimas. Ištyrus sistemos tikslumą nustatyta, kad suprojektuota sukčiavimo aptikimo sistema sėkmingai identifikavo aiškius sukčiavimo atvejus bei teisingai diferencijavo transakcijas pagal jų rizikingumą. Vis dėlto, dėl netinkamo kriterijų suskirstymo į kriterijų grupes, sistema neužtikrina pakankamo jautrumo ir prastai identifikuoja mažiau aiškius sukčiavimo atvejus. Todėl siekiant pakankamo sistemos tikslumo reikėtų peržiūrėti kriterijų grupavimą, įvesti papildomų aptikimo kriterijų. Sąlyginai primityvioje diegimo konfigūracijose ištyrus suprojektuotos sukčiavimo aptikimo sistemos efektyvumą nustatyta, kad sistema gali 98 % atvejų pateikti sukčiavimo įvertį greičiau nei per 1 sekundę. Vertinant sistemos pralaidumą nesudėtingoje sistemos diegimo konfigūracijoje buvo pasiektas 2500 patikimai apdorojamų transakcijų per sekundę kiekis. Todėl buvo prieita prie išvados, kad įgyvendinus sistemą rekomenduojamoje sistemos diegimo konfigūracijoje sistema turėtų pateikti sukčiavimo įvertį realiu laiku.

Rezultatai

1. Identifikuotos galimos bekontakčių mokėjimų saugumo spragos ir sukčiavimo būdai.
2. Apibrėžtas duomenų paruošimo procesas bei Bajeso tinklas tinkamas sukčiavimo aptikimui.
3. Suprojektuota sukčiavimo aptikimo sistemos realizacija.

4. Įgyvendinta sukčiavimo aptikimo sistema realizuojanti apbrėžtą Bajeso tinklo struktūrą, sukčiavimo aptikimo kriterijus bei sistemos architektūrą.
5. Atliktas įgyvendintos sukčiavimo aptikimo sistemos tikslumo ir efektyvumo tyrimas.
6. Suprojektuota ir įgyvendinta sukčiavimo aptikimo sistema naudojanti apibrėžtą Bajeso tinklą, kuri geba identifikuoti aiškius sukčiavimo atvejus, tinkamai diferencijuoti transakcijų riziką bei pateikti įvertį realiu laiku.

Išvados

1. Sukčiavimas atliekant bekontakčius mokėjimus yra reali grėsmė.
2. Transakcijų duomenų rinkinio normalizavimas, skirstymas į kategorijas bei išvestinių atributų generavimas leidžia pagerinti sukčiavimo aptikimo tikslumą.
3. Naudojant Bajeso tinklą poreikis duomenų saugyklos resursams auga eksponentiškai priklausomai nuo kriterijų kiekio, tačiau kriterijų grupavimas į kriterijų grupes mažina duomenų saugyklos resursų poreikius.
4. Sukonkretinus sukčiavimo aptikimui naudojamą Bajeso tinklo realizaciją padidinamas efektyvumas, tai įmanoma nes visoms transakcijoms skaičiuojamas vieningas įvertis, o transakcijų protokolai griežti.
5. Bajeso tinklo kriterijus sugrupavus į kriterijų grupes, kurias galima identifikuoti prieš skaičiuojant įvertį, įverčio apskaičiavimui pakanka atlikti polinomiškai didėjantį operacijų kiekį.
6. Apmokant Bajeso tinklą reikia atlikti operacijų kiekį augantį eksponentiškai priklausomai nuo Bajeso tinklo kriterijų kiekio, todėl po kiekvienos transakcijos vykdyti apmokymą neefektyvu, reikalingas transakcijų grupavimas arba periodinis apmokymo vykdymas.
7. Bajeso tinklo kriterijų grupavimas mažina sukčiavimo aptikimo jautrumą.
8. Naudojant bazinius transakcijos atributus įmanoma įvertinti transakcijas inicijuojančio asmens elgsenos šablonus atsižvelgiant į sumos rizikingumą, kiekio rizikingumą, vietos rizikingumą, laiko rizikingumą.
9. Naudojant sukčiavimo aptikimui sukonkretintą Bajeso tinklą įmanoma sukurti sukčiavimo aptikimo sistemą pateikiančią sukčiavimo įverčius realiu laiku.

Šaltiniai

[AS12] M. I. Alowais, L. K. Soon. Credit Card Fraud Detection: Personalized or Aggregated Model. *2012 Third FTRA International Conference on Mobile, Ubiquitous, and Intelligent Computing*. Vankuveris, 2012, p.114-119.

[BAS+16] A. C. Bahnsen, D. Aouada, A. Stojanovic, B. Ottersten. Feature engineering strategies for credi card fraud detection. *Expert Systems with Applications,* 2016, 51, p. 134-142.

[BVV15] B. Baesens, V. V. Vlasselaer, W. Verbeke. *Fraud analytic using descriptive, predictive, and social network techniques a guide to data science for fraud detection*. Wiley, 2015.

[Col16] P. Collinson. *Cashless Britain advances as contactless and debit cards thrive.* [žiūrėta 2016-12-11]. Prieiga per internetą: <https://www.theguardian.com/money/2016/may/23/cashless-britain-advances-contactless-debit-cards-thrive>

[DGS+97] J. R. Dorronsoro, F. Ginel, C. Sachez, C. S. Cruz. Neural Fraud Detection in Credit Card Operations. *IEEE Transactions on neural networks,* 1997, 8(4), p. 827-834.

[ESD16] ECB Statistical Data Warehouse. *Payment Statistics.* [žiūrėta 2017-06-11]. Prieiga per internetą: < http://sdw.ecb.europa.eu/reports.do?node=1000001390 >

[Emm16] M. J. Emms. *Contactless payments: usability at the cost of security?*. Daktaro disertacija. Newcastle university, 2016.

[EMV11] EMVCo. *Security and Key Management.* [žiūrėta 2017-04-08]. Prieiga per internetą: <https://www.emvco.com/specifications.aspx?id=223>

[FP97] T. Fawcett, F. Provost. Adaptive Fraud Detection. *Data Mining and Knowledge Discovery*. 1997, 1(3), p. 291-316.

[Gre12] A. Greenberg. *Hacker's Demo Shows How Easily Credit Cards Can Be Read Through Clothes And Wallets.* [žiūrėta 2017-06-10]. Prieiga per internetą:

<https://www.forbes.com/sites/andygreenberg/2012/01/30/hackers-demo-shows-how-easily-credit-cards-can-be-read-through-clothes-and-wallets>

[GR94] S. Ghosh, D. L. Reilly. Credit Card Fraud Detection with a Neural-Network. *Proceedings of the Twenty-Seventh Annual Hawaii International Conference on System Sciences*, Honolulu, 1994, p. 621-630.

[HA14] N. S. Halvaiee, M. K. Akbari. A novel model for credit card fraud detection using Artificial Immune Systems. *Applied Soft Computing, 2014, 24, p. 40–49.*

[HKP11] J. Han, M. Kamber, J. Pei. *Data mining: concepts and techniques*. Morgan Kaufmann, 2011.

[JKK+16] J. N. John, O. Kennedy, C. G. Kennedy, C. Anele, F. Olajide. Realtime Fraud Detection in the banking sector using data minin techniques/algorithm. *2016 International Conference on Computational Science and Computational Intelligence.* Las vegasas, 2016, p.1186-1191.

[KC16] Y. Kültür, M. U. Çağlayan. Hybrid approaches for detecting credit card fraud. *Expert systems, 2016, 34(2),* p. 1-13.

[KMN+15] C. Kier, G. Madlmayr, A. Nawratil, M. Schafferer, C. Schanes, T. Grechenig. Mobile Payment Fraud: A Practical View on the Technical Architecture and Starting Points for Forensic Analysis of New Attack Scenarios. *2015 Ninth International Conference on IT Security Incident Management IT Forensics*. Magdenburgas, 2015, p. 68-76.

[Kub96] J. Kubilius. *Tikimybių teorija ir matematinė statistika.* Vilniaus universiteto leidykla. 1996

[LRL10] I. Lacmanović, B. Radulović, D. Lacmanović. Contactless payment systems based on RFID technology, *The 33rd International Convention MIPRO.* Opatija, 2010, p.1114-1119.

[LYW13] Y. J. Lee, Y. R. Yeh, Y. C. F. Wang. Anomaly Detection via Online Oversampling Principal Component Analysis, *IEEE Transactions on Knowledge and Data Engineering,* 2013, 25(7), p.1460-1470.

[MDA+10] S. J. Murdoch, S. Drimer, R. Anderson, M. Bond. Chip and PIN is Broken. *2010 IEEE Symposium on Security and Privacy.* Oklandas, 2010, p. 433-446.

[Mon04] D. A. Montague. *Fraud Prevention Techniques for Credit Card Fraud*. Trafford Publishing, 2004.

[MTV+02] S. Maes, K. Tuyls, B. Vancheonwinkel, B. Manderick. Credit Card Fraud Detection Using Bayesian and Neural Networks. *Proceedings of the 1st international naiso congress on neuro fuzzy technologies*. Havana, 2002.

[Nea03] R. E. Neapolitan. *Learning Bayesian Networks.* Prentice-Hall. 2003

[RP11] S. B. E. Raj, A. A. Portia. Analysis on credit card fraud detection methods. *2011 International Conference on Computer, Communication and Electrical Technology (ICCCET).* Maruthakulam Tirunelveli, 2011, p. 152-156

[Tril11] M. Trillo. *Visa Transactions Hit Peak on Dec. 23.* [žiūrėta 2018-01-11]. Prieiga per internetą: <https://www.visa.com/blogarchives/us/2011/01/12/visa-transactions-hit-peak-on-dec-23/index.html>

[SAA15] S. M. Shariati, A. Abouzarjomehri, M. H. Ahmadzadegan. Investigating NFC technology from perspective of security, analysis of attacks and existing risk, *2015 2nd International Conference on Knowledge-Based Engineering and Innovation (KBEI),* Teheranas, 2015, p. 1083-1087.

[SKS+08] A. Srivastava, A. Kundu, S. Sural, A. Majumdar. Credit Card Fraud Detection Using Hidden Markov Model. *IEEE Transactions on Dependable and Secure Computing, 2008, 5(1), p. 37-48.*

[SYB+16] A. Srivastava, M. Yadav, S. Basu, S. Salunkhe, M. Shabad. Credit Card Fraud Detection at Merchant Side using Neural Networks. *2016 International Conference on Computing for Sustainable Global Development*. Naujasis Delis, 2016, p. 667-670.

[UCA16a] The UK Cards association. *Fifth of all card payments now contactless.* [žiūrėta 2016-12-11]. Prieiga per internetą: < http://www.theukcardsassociation.org.uk/news/fifthcontactlessNov2016.asp>

[UCA16b] The UK Cards association. *What is Contactless?.* [žiūrėta 2018-01-11]. Prieiga per internetą:

<http://www.theukcardsassociation.org.uk/contactless_consumer/what_is_contactless.asp>

[ZK17] Z. K. Zandian, M. Kayvanour. Systematic identification and analysis of different fraud detection approaches based on the strategy ahead. *International Journal of Knowledge-based and Intelligent Engineering Systems.* 21(2), 2017, p. 123-124.

[ZYL09] Y. Zhang, F. You, H. Liu. Behavior-Based Credit Card Fraud Detecting Model. *2009 Fifth International Joint Conference on INC, IMS and IDC*. Seulas, 2009, p.855-858.

[WB15] J. West, M. Bhattacharya. Some Experimental Issues in Financial Fraud Detection: An Investigation. *Proceedings of The 5th International Symposium on Cloud and Service Computing*. Honolulu, 2015, p.1155-1158.

[WHS16] Y. Wang, C. Hahn, K. Sutrave. Mobile payment security, threats, and challenges. *2016 Second International Conference on Mobile and Secure Services (MobiSecServ), Geinsvilis, 2016.*

Priedai

1 priedas. Kriterijų sąlyginės nepriklausomybės įrodymas

Keliama hipotezė, kad kai KG_i būsenos žinomos, galioja lygybė: $P(SUK|KG_1 \cap KG_2 \cap \dots \cap KG_k \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_k|}) = P(SUK|KG_1 \cap \dots KG_k)$

Pagal sąlyginės tikimybės apibrėžimą:

$$P(SUK|KG_1 \cap KG_2 \cap \dots \cap KG_k \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_k|})$$

$$= \frac{P(SUK \cap KG_1 \cap KG_2 \cap \dots \cap KG_k \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_k|})}{P(KG_1 \cap KG_2 \cap \dots \cap KG_k \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_k|})}$$

Basejo tinklo struktūra apibrėžia kurios mazgų tarpusavio priklausomybės laikomos reikšmingomis. Dviejų lygių Bajeso tinkle grupuojančiame kriterijus į kriterijų grupes galioja:

- kriterijai tarpusavyje nėra sujungti briaunomis, todėl kriterijai laikomi tarpusavyje nepriklausomais, galioja: $P(K_i) = P(K_i|K_j)$, $\forall i,j$ tokiais kad $i \neq j$, $i \leq n$, $j \leq n$, čia n – kriterijų tinkle kiekis;
- kriterijų grupės nėra sujungtos briaunomis, todėl kriterijų grupės laikomos tarpusavyje nepriklausomomis, galioja: $P(KG_i) = P(KG_i|KG_j)$, $\forall i,j$ tokiais kad $i \neq j$, $i \leq k$, $j \leq k$, čia k – kriterijų grupių tinkle kiekis;
- kriterijus briauna sujungtas tik su viena kriterijų grupe, kriterijus tarpusavyje nepriklausomas su kitomis grupėmis, todėl galioja: $P(K_i|KG_j) = P(K_{mi})$, $\forall i,j,m$ tokiais kad $m \neq j$, $i \leq n$, $j \leq k$, $m \leq k$, čia n – kriterijų tinkle kiekis, k – kriterijų grupių kiekis;
- sukčiavimo mazgas neturi tiesioginių briaunų su kriterijais, sukčiavimas ir kriterijai laikomi tarpusavyje nepriklausomais, su sąlyga, kad yra žinoma kriterijų grupė, todėl galioja: $P(K_{ij}|KG_i \cap SUK) = P(K_{ij}|KG_i)$, $\forall i,j$ tokiais kad $i \leq k$, $j \leq |KG_i|$, čia k – kriterijų grupių tinkle kiekis.

Jungtinės tikimybės išskleidžiamos pagal įvykių jungties tikimybės daugybos formulę:

$$P\left(SUK \cap KG_1 \cap KG_2 \cap \dots \cap KG_k \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_k|}\right)$$
$$= P(SUK) \cdot P(KG_1|SUK) \cdot \dots \cdot P(KG_k|KG_{k-1} \cap \dots KG_1 \cap SUK)$$
$$\cdot P(K_{11}|KG_k \cap \dots KG_1 \cap SUK) \cdot P(K_{12}|K_{11} \cap KG_k \cap \dots KG_1 \cap SUK) \cdot \dots$$
$$\cdot P\left(K_{1|KG_{1|}}|K_{1(|KG_1|-1)} \cap \dots \cap K_{11} \cap KG_k \cap \dots KG_1 \cap SUK\right) \cdot \dots$$
$$\cdot P(K_{k1}|K_{(k-1)|KG_{k-1|}} \cap \dots \cap K_{11} \cap KG_k \cap \dots KG_1 \cap SUK) \cdot \dots$$
$$\cdot P(K_{k|KG_k|}|K_{k(|KG_k|-1)} \cap \dots \cap K_{11} \cap KG_k \cap \dots KG_1 \cap SUK)$$

Bendru atveju laikoma, kad kiekvienas tinklo mazgas yra susijęs su visais kitais tinklo mazgais. Pritaikius dėl apibrėžtos Bajeso tinklo struktūros galiojančias lygybes, gaunama:

$$P\left(SUK \cap KG_1 \cap KG_2 \cap \dots \cap KG_k \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_k|}\right)$$
$$= P(SUK) \cdot P(KG_1|SUK) \cdot \dots \cdot P(KG_k|SUK) \cdot P(K_{11}|KG_k \cap \dots KG_1 \cap SUK) \cdot \dots$$
$$\cdot P\left(K_{1|KG_{1|}}|KG_k \cap \dots KG_1 \cap SUK\right) \cdot \dots \cdot P(K_{k1}|KG_k \cap \dots KG_1 \cap SUK) \cdot \dots$$
$$\cdot P\left(K_{k|KG_k|}|KG_k \cap \dots KG_1 \cap SUK\right)$$
$$= P(SUK) \cdot P(KG_1|SUK) \cdot \dots \cdot P(KG_k|SUK) \cdot P(K_{11}|KG_1 \cap SUK) \cdot \dots \cdot P\left(K_{1|KG_{1|}}|KG_1 \cap SUK\right)$$
$$\cdot \dots \cdot P(K_{k1}|KG_k \cap SUK) \cdot \dots \cdot P\left(K_{k|KG_k|}|KG_k \cap SUK\right)$$
$$= P(SUK) \cdot P(KG_1|SUK) \cdot \dots \cdot P(KG_k|SUK) \cdot P(K_{11}|KG_1) \cdot \dots \cdot P\left(K_{1|KG_{1|}}|KG_1\right) \cdot \dots$$
$$\cdot P(K_{k1}|KG_k) \cdot \dots \cdot P\left(K_{k|KG_k|}|KG_k\right)$$

Analogiškai pagal aprašytas prielaidas gaunama:

$$P\left(KG_1 \cap KG_2 \cap \dots \cap KG_k \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_k|}\right)$$
$$= P(KG_1) \cdot \dots \cdot P(KG_k) \cdot P(K_{11}|KG_1) \cdot \dots \cdot P\left(K_{1|KG_{1|}}|KG_1\right) \cdot \dots \cdot P(K_{k1}|KG_k) \cdot \dots$$
$$\cdot P\left(K_{k|KG_k|}|KG_k\right)$$

Įstačius gautas išraiškas gaunama:

$$P(SUK|KG_1 \cap KG_2 \cap \dots \cap KG_k \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_k|})$$
$$= \frac{P\left(SUK \cap KG_1 \cap KG_2 \cap \dots \cap KG_k \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_k|}\right)}{P\left(KG_1 \cap KG_2 \cap \dots \cap KG_k \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_k|}\right)}$$
$$= \frac{P(SUK) \cdot P(KG_1|SUK) \cdot \dots \cdot P(KG_k|SUK)}{P(KG_1) \cdot \dots \cdot P(KG_k)}$$
$$\cdot \frac{P(K_{11}|KG_1) \cdot \dots \cdot P\left(K_{1|KG_{1|}}|KG_1\right) \cdot \dots \cdot P(K_{k1}|KG_k) \cdot \dots \cdot P\left(K_{k|KG_k|}|KG_k\right)}{P(K_{11}|KG_1) \cdot \dots \cdot P\left(K_{1|KG_{1|}}|KG_1\right) \cdot \dots \cdot P(K_{k1}|KG_k) \cdot \dots \cdot P\left(K_{k|KG_k|}|KG_k\right)}$$
$$= \frac{P(SUK) \cdot P(KG_1|SUK) \cdot \dots \cdot P(KG_k|SUK)}{P(KG_1) \cdot \dots \cdot P(KG_k)}$$

Gavus suprastintą išraišką pritaikomos tos pačios prielaidos atgaline tvarka, gaunamos jungties tikimybės:

$$P(SUK|KG_1 \cap KG_2 \cap \dots \cap KG_k \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_k|})$$
$$= \frac{P(SUK) \cdot P(KG_1|SUK) \cdot \dots \cdot P(KG_k|SUK)}{P(KG_1) \cdot \dots \cdot P(KG_k)}$$
$$= \frac{P(SUK) \cdot P(KG_1|SUK) \cdot P(KG_2|KG_1 \cap SUK) \cdot \dots \cdot P(KG_k|KG_{k-1} \cap \dots \cap KG_1 \cap SUK)}{P(KG_1) \cdot P(KG_2|KG_1) \cdot \dots \cdot P(KG_k|KG_{k-1} \cap \dots \cap KG_1)}$$
$$= \frac{P(SUK \cap KG_1 \cap \dots KG_k)}{P(KG_1 \cap \dots KG_k)} = P(SUK|KG_1 \cap \dots KG_k)$$

Todėl norint įvertinti sukčiavimo tikimybę pakanka žinoti kriterijų grupių būsenas:

$$P(SUK|KG_1 \cap KG_2 \cap \dots \cap KG_k \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_k|}) = P(SUK|KG_1 \cap \dots KG_k)$$

2 priedas. Kriterijų grupių įvykių jungties tikimybės apskaičiavimas

Keliama hipotezė, kad sukčiavimo ir kriterijų grupių įvykių jungties tikimybei galioja lygybė:

$$P(SUK \cap KG_1 \cap KG_2 \cap \dots \cap KG_k) = P(SUK) \cdot \prod_{i=1}^{k} P(KG_i|SUK)$$

Basejo tinklo struktūra apibrėžia, kurios mazgų tarpusavio priklausomybės laikomos reikšmingomis. Kriterijų grupės nėra sujungtos briaunomis, todėl kriterijų grupės laikomos tarpusavyje nepriklausomomis. Galioja: $P(KG_i) = P\left(KG_i \middle| KG_j\right)$, $\forall i, j$ tokiais kad $i \neq j$, $i \leq k$, $j \leq k$, čia k – kriterijų grupių tinkle kiekis. Taip pat kriterijų grupės tarpusavyje sujungtos tik per SUK mazgą ir laikoma, kad jos tarpusavyje nepriklausomos su sąlyga, kad žinomas SUK. Todėl galioja: $P(KG_i|SUK) = P\left(KG_i \middle| KG_j \cap SUK\right)$, $\forall i, j$ tokiais kad $i \neq j$, $i \leq k$, $j \leq k$, čia k – kriterijų grupių tinkle kiekis.

Įvykių jungties tikimybę galima išreikšti pagal daugybos formulę, tada pritaikius kriterijų grupių tarpusavio nepriklausomybę gaunama:

$$P(SUK \cap KG_1 \cap KG_2 \cap \dots \cap KG_k)$$
$$= P(SUK) \cdot P(KG_1|SUK) \cdot P(KG_2|KG_1 \cap SUK) \cdot \dots \cdot P(KG_k|KG_{k-1} \cap \dots \cap SUK)$$
$$= P(SUK) \cdot P(KG_1|SUK) \cdot P(KG_2|SUK) \cdot \dots \cdot P(KG_k|SUK) = P(SUK) \cdot \prod_{i=1}^{k} P(KG_i|SUK)$$

Iš to aišku, galioja lygybė sukčiavimo ir kriterijų grupių jungties tikimybei:

$$P(SUK \cap KG_1 \cap KG_2 \cap \dots \cap KG_k) = P(SUK) \cdot \prod_{i=1}^{k} P(KG_i|SUK)$$

3 priedas. Tikimybės apskaičiavimas kai nežinomos kriterijų grupės

Reikia rasti būdą apskaičiuoti: $P\left(SUK\middle|K_{11}\cap\ \dots\cap K_{1|KG_1|}\cap\dots\cap K_{k1}\cap\dots\cap K_{k|KG_k|}\right)$. Tikimybę galima būtų apskaičiuoti pagal klasikinį tikimybės apibrėžimą arba pagal Bajeso teoremą, tačiau dviejų lygių tinkle siekiama išnaudoti kriterijų grupavimą, todėl pateikiamas būdas apskaičiuoti atsižvelgiant į kriterijų grupes.

Iš sąlyginės tikimybės apibrėžimo žinoma, kad: $\mathrm{P}\left(SUK\middle|K_{11}\cap\ \dots\cap K_{1|KG_1|}\cap\dots\cap K_{k1}\cap\dots\cap K_{k|KG_k|}\right)=\frac{P\left(SUK\cap K_{11}\cap\dots\cap K_{1|KG_1|}\cap\dots\cap K_{k1}\cap\dots\cap K_{k|KG_k|}\right)}{P\left(K_{11}\cap\dots\cap K_{1|KG_1|}\cap\dots\cap K_{k1}\cap\dots\cap K_{k|KG_k|}\right)}$

Tikimybių teorijoje būtinasis įvykis žymimas – Ω. Tada, su bet kokiu įvykiu A galioja lygybė: $A\cap\Omega=A$. Tada: $P(A\cap\Omega)=P(A)$. Remiantis šia savybe galima pertvarkyti jungties tikimybes įvedant kriterijų grupes. Tegu KG_i^Ω žymi i – tosios kriterijų grupės būtinojo įvykio tikimybę. Tai reiškia, kad KG_i^Ω apibrėžiamas įvykiu aibe, kuri sudaryta iš visų galimų kriterijų grupės KG_i būsenų. $KG_i^\Omega=\{b_{i1},b_{i2},\dots,b_{i|KG_i|}\}$, čia $b_{i1},b_{i2},\dots,b_{i|KG_i|}$ – visos i – tosios kriterijų grupės būsenos. Iš to aišku, kad $KG_i^\Omega=KG_{i1}\cup KG_{i2}\cup\dots\cup KG_{i|KG_i|}$, čia KG_{ij} – įvykis žymintis, kad bus parinkta i – tosios kriterijų grupės būsena b_{ij}, $j\le|KG_i|$. Iš to gaunama: $A=A\cap KG_1^\Omega=A\cap\left(KG_{i1}\cup KG_{i2}\cup\dots\cup KG_{1|KG_1|}\right)$. Tada: $P(A)=P\Big(A\cap\big(KG_{i1}\cup KG_{i2}\cup\dots\cup KG_{1|KG_1|}\big)\Big)$. Pasinaudojus tikimybinių įvykių distributyvumu ir būtinojo įvykio adityvumu gaunama:

$$P\left(A\cap\left(KG_{i1}\cup KG_{i2}\cup\dots\cup KG_{1|KG_1|}\right)\right)=P\left((A\cap KG_{i1})\cup(A\cap KG_{i2})\cup\dots\cup\left(A\cap KG_{1|KG_1|}\right)\right)$$

$$=P(A\cap KG_{i1})+P(A\cap KG_{i2})+\dots+P\left(A\cap KG_{1|KG_1|}\right)=\sum_{j=1}^{|KG_1|}P(A\cap KG_{1j})$$

$$=\sum_{KG_1\in KG_1^\Omega}P(A\cap KG_1)$$

Apjungus su prieš tai gautomis išraiškomis: $P(A)=\sum_{KG_1\in KG_1^\Omega}P(A\cap KG_1)$. Analogišku principu, galima pridėti dar vieną būtinąjį įvykį, KG_2^Ω:

$$P\left(A\cap\left(KG_{11}\cup KG_{12}\cup\dots\cup KG_{1|KG_1|}\right)\cap\left(KG_{21}\cup KG_{22}\cup\dots\cup KG_{2|KG_2|}\right)\right)$$

$$=P\Big(\left(A\cap\left(KG_{11}\cup KG_{12}\cup\dots\cup KG_{1|KG_1|}\right)\cap KG_{21}\right)\cup\dots$$

$$\cup\left(A\cap\left(KG_{11}\cup KG_{12}\cup\dots\cup KG_{1|KG_1|}\right)\cap KG_{2|KG_2|}\right)\Big)$$

$$= P\Big((A \cap KG_{11} \cap KG_{21}) \cup \ldots \cup (A \cap KG_{1|KG_1|} \cap KG_{21}) \cup \ldots \cup (A \cap KG_{11} \cap KG_{2|KG_2|}) \cup \ldots$$

$$\cup (A \cap KG_{1|KG_1|} \cap KG_{2|KG_2|})\Big) = \sum_{\substack{KG_1 \in KG_1^{\Omega} \\ KG_2 \in KG_2^{\Omega}}} P(A \cap KG_1 \cap KG_2)$$

Apibendrinant: $P(A) = \sum_{\substack{KG_1 \in KG_1^{\Omega} \\ KG_2 \in KG_2^{\Omega}}} P(A \cap KG_1 \cap KG_2)$ Matoma, kad pridedant būtinuosius įvykius, pradinio įvykio tikimybė išlaikoma skaičiuojant įvykio A ir visų galimų pridėtų būtinųjų įvykių dedamųjų kombinacijų sumą. Kadangi turime k – kriterijų grupių tokiu pačiu principu galima išplėsti tikimybę pridedant visų kriterijų grupių būtinuosius įvykius: $P(A) = \sum_{\substack{KG_1 \in KG_1^{\Omega} \\ \ldots \\ KG_k \in KG_k^{\Omega}}} P(A \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)$. Kadangi, lygybė galioja su bet kokiu įvykiu A, vietoje A įstačius $SUK \cap K_{11} \cap \ \ldots \cap K_{1|KG_{1|}} \cap \ldots \cap K_{k1} \cap \ldots \cap K_{k|KG_k|}$, o rezultatą įstačius į sąlyginės sukčiavimo tikimybės išraišką gaunama:

$$P\left(SUK \middle| K_{11} \cap \ \ldots \cap K_{1|KG_{1|}} \cap \ldots \cap K_{k1} \cap \ldots \cap K_{k|KG_k|}\right)$$

$$= \sum_{\substack{KG_1 \in KG_1^{\Omega} \\ \ldots \\ KG_k \in KG_k^{\Omega}}} \frac{P(SUK \cap K_{11} \cap \ \ldots \cap K_{1|KG_{1|}} \cap \ldots \cap K_{k1} \cap \ldots \cap K_{k|KG_k|} \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k)}{P\left(K_{11} \cap \ \ldots \cap K_{1|KG_{1|}} \cap \ldots \cap K_{k1} \cap \ldots \cap K_{k|KG_k|}\right)}$$

Ši išraiška, leidžia įvertinti sukčiavimo tikimybę žinant tik kriterijų grupių būsenų duomenis ir atsižvelgiant į kiekvieną galima kriterijų grupių būseną. Vis dėlto išraiška sunkiai apskaičiuojama Bajeso tinkle, todėl bus suprastinta. Galima naudojantis daugybos formule, tačiau tokio pavidalo įvykių jungties tikimybės išraiška jau analizuota. Suvedame įvykių jungties tikimybę į prieš tai analizuotą sąlyginės tikimybės pavidalą (žr. 1 priedas):

$$P(SUK|KG_1 \cap KG_2 \cap \ldots \cap KG_k \cap K_{11} \cap \ \ldots \cap K_{1|KG_{1|}} \cap \ldots \cap K_{k1} \cap \ldots \cap K_{k|KG_k|})$$

$$= \frac{P(SUK \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k \cap K_{11} \cap \ \ldots \cap K_{1|KG_{1|}} \cap \ldots \cap K_{k1} \cap \ldots \cap K_{k|KG_k|})}{P(KG_1 \cap KG_2 \cap \ldots \cap KG_k \cap K_{11} \cap \ \ldots \cap K_{1|KG_{1|}} \cap \ldots \cap K_{k1} \cap \ldots \cap K_{k|KG_k|})}$$

Norint gauti šią išraišką, skaičiavimų rezultatas padauginamas ir iškart padalinamas iš sąlyginės tikimybės vardiklio, reikšmė nepakeičiama, tačiau gaunama:

$$\sum_{\substack{KG_1 \in KG_1^{\Omega} \\ \ldots \\ KG_k \in KG_k^{\Omega}}} \frac{P\left(SUK \cap K_{11} \cap \ \ldots \cap K_{1|KG_{1|}} \cap \ldots \cap K_{k1} \cap \ldots \cap K_{k|KG_k|} \cap KG_1 \cap KG_2 \cap \ldots \cap KG_k\right)}{P\left(K_{11} \cap \ \ldots \cap K_{1|KG_{1|}} \cap \ldots \cap K_{k1} \cap \ldots \cap K_{k|KG_k|}\right)}$$

$$= \sum_{\substack{KG_1 \in KG_1^{\Omega} \\ \dots \\ KG_k \in KG_k^{\Omega}}} \frac{P\left(SUK \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_{k|}} \cap KG_1 \cap KG_2 \cap \dots \cap KG_k\right)}{P\left(KG_1 \cap KG_2 \cap \dots \cap KG_k \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_{k|}}\right)}$$

$$\cdot \frac{P\left(KG_1 \cap KG_2 \cap \dots \cap KG_k \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_{k|}}\right)}{P\left(K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_{k|}}\right)}$$

$$= \sum_{\substack{KG_1 \in KG_1^{\Omega} \\ \dots \\ KG_k \in KG_k^{\Omega}}} \left(P\left(SUK \middle| KG_1 \cap KG_2 \cap \dots \cap KG_k \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_{k|}}\right)\right.$$

$$\left. \cdot P\left(KG_1 \cap KG_2 \cap \dots \cap KG_k | K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_{k|}}\right)\right)$$

Sudarytos Bajeso tinklo struktūros kriterijų grupės tarpusavyje sąlyginai nepriklausomos, kai žinomi visi kriterijai. Taip pat visi kriterijai tarpusavyje nepriklausomi. Tada A, B, C bet kokie įvykiai, tenkinantys sąlygą, kad A, B tarpusavyje sąlyginai nepriklausomi jeigu žinomas C. Iš to aišku, kad $P(B|C \cap A) = P(B|C)$. Tada $P(A \cap B|C) = \frac{P(A \cap B \cap C)}{P(C)} = P(A|C) \cdot P(B|A \cap C) = P(A|C) \cdot P(B|C)$. Šią savybę pritaikius kriterijų grupių priklausomybės nuo kriterijų tikimybei, gaunama: $P\left(KG_1 \cap KG_2 \cap \dots \cap KG_k | K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_{k|}}\right) = \prod_{i=1}^{k} P(KG_i | K_{i1} \cap K_{i2} \cap \dots \cap K_{ik})$. Taip pat žinoma, kad (žr. 1 priedas): $P(SUK|KG_1 \cap KG_2 \cap \dots \cap KG_k \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_{k|}}) = P(SUK|KG_1 \cap \dots \cap KG_k)$. Gautas išraiškas, įstatome į bendrą rezultatą:

$$\sum_{\substack{KG_1 \in KG_1^{\Omega} \\ \dots \\ KG_k \in KG_k^{\Omega}}} \left(P\left(SUK \middle| KG_1 \cap KG_2 \cap \dots \cap KG_k \cap K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_{k|}}\right)\right.$$

$$\left. \cdot P\left(KG_1 \cap KG_2 \cap \dots \cap KG_k | K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_{k|}}\right)\right)$$

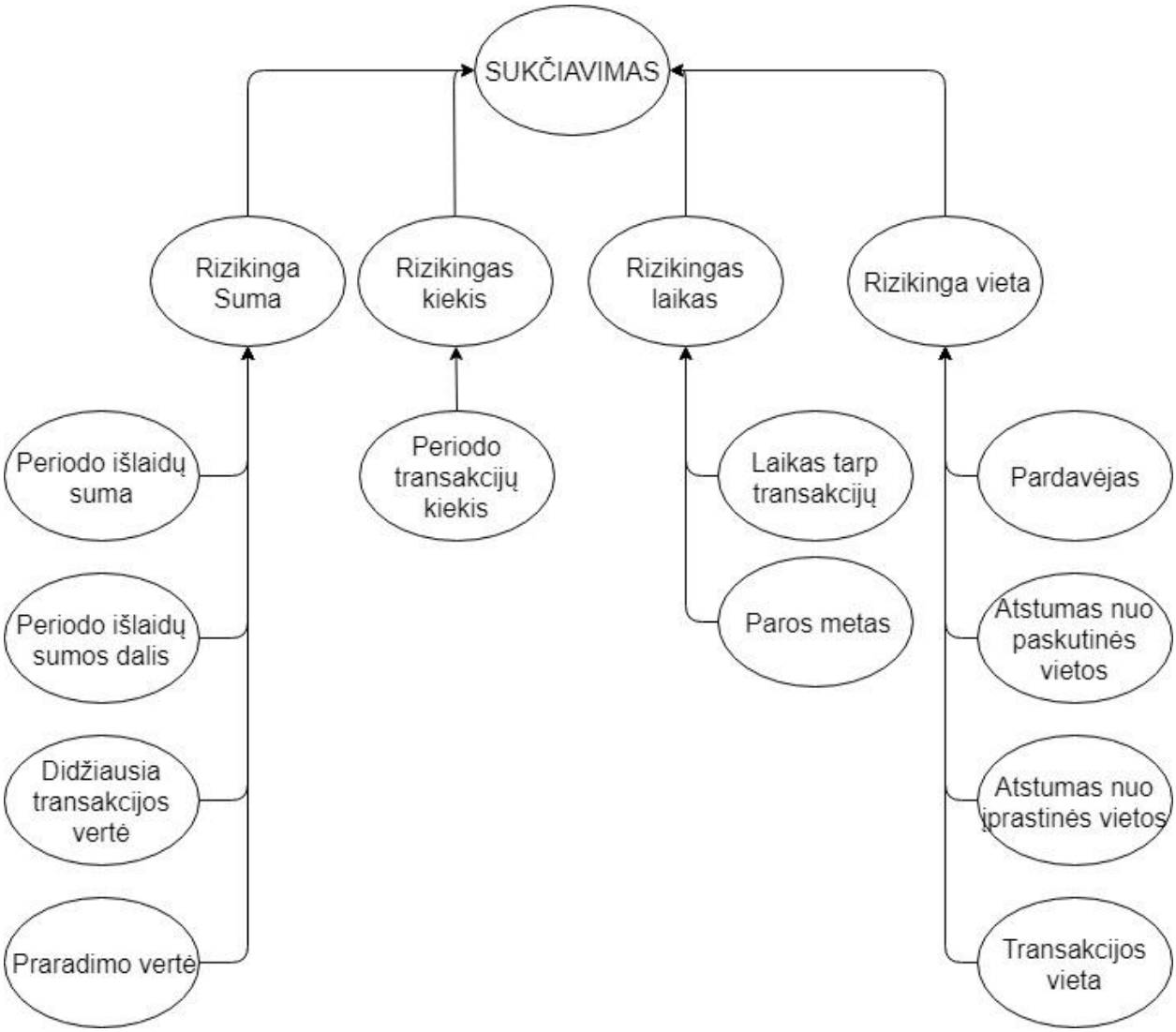
$$= \sum_{\substack{KG_1 \in KG_1^{\Omega} \\ \dots \\ KG_k \in KG_k^{\Omega}}} \left(P(SUK|KG_1 \cap \dots \cap KG_k) \cdot \prod_{i=1}^{k} P(KG_i | K_{i1} \cap K_{i2} \cap \dots \cap K_{ik})\right)$$

Apibendrinant, galima teigti, kad

$$P\left(SUK \middle| K_{11} \cap \dots \cap K_{1|KG_{1|}} \cap \dots \cap K_{k1} \cap \dots \cap K_{k|KG_{k|}}\right)$$

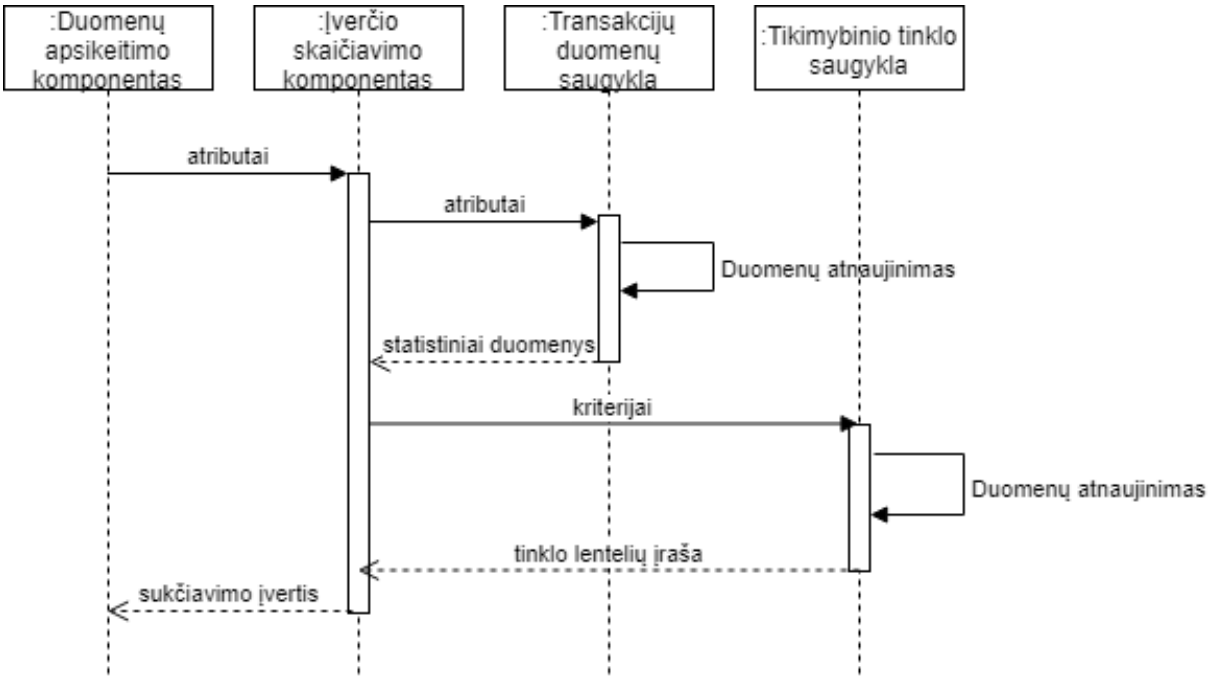
$$= \sum_{\substack{KG_1 \in KG_1^{\Omega} \\ \dots \\ KG_k \in KG_k^{\Omega}}} \left(P(SUK|KG_1 \cap \dots \cap KG_k) \cdot \prod_{i=1}^{k} P(KG_i | K_{i1} \cap K_{i2} \cap \dots \cap K_{ik})\right)$$

4 priedas. Tinklo struktūra



6 pav. Apibrėžto tinklo struktūra

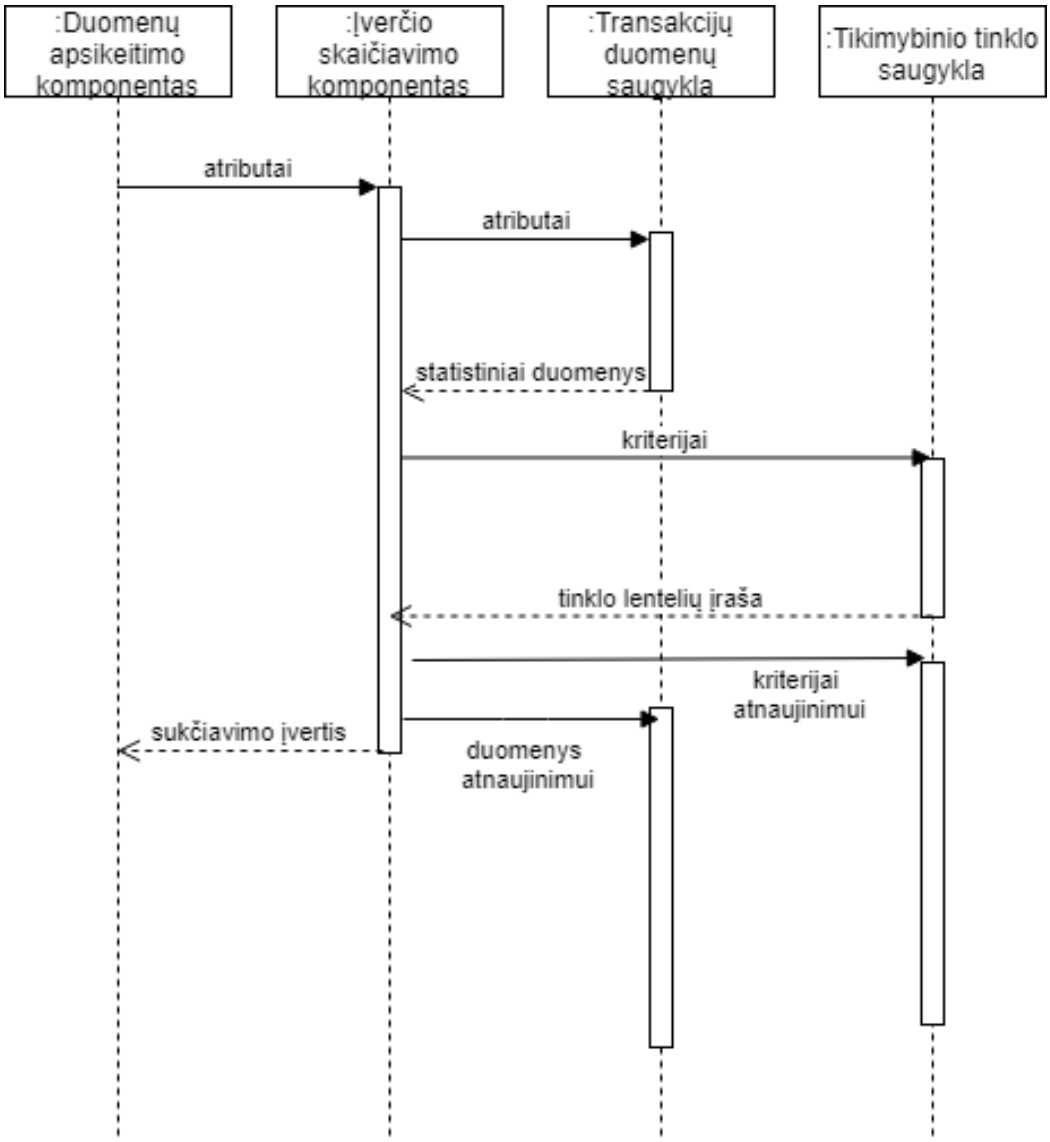
5 priedas. Sistemos komponentų integracijos sinchroninis įgyvendinimas



7 pav. Sistemos komponentų integracijos skaičiuojant sukčiavimo įvertį, sinchroninis įgyvendinimas

Duomenų apsikeitimo komponentas gavęs užklausą apdoroti transakciją, jos atributus perduoda įverčio skaičiavimo komponentui, kuris perduoda reikiamus atributus transakcijų duomenų saugyklos komponentui, kuris atnaujina reikalingus statistinius duomenis, bei gražina šiuos duomenis įverčio skaičiavimo komponentui. Skaičiavimo komponentas gavęs šiuos duomenis įvertina sukčiavimo kriterijų vertes ir perduoda jas tikimybinio tinklo saugyklai, kuri atnaujina reikalingas tikimybines vertes ir gražina kriterijų verčių tikimybines vertes kurias įverčio skaičiavimo komponentas panaudoja apskaičiuoti galutinį sukčiavimo įvertį, kurį gražina duomenų apsikeitimo komponentui.

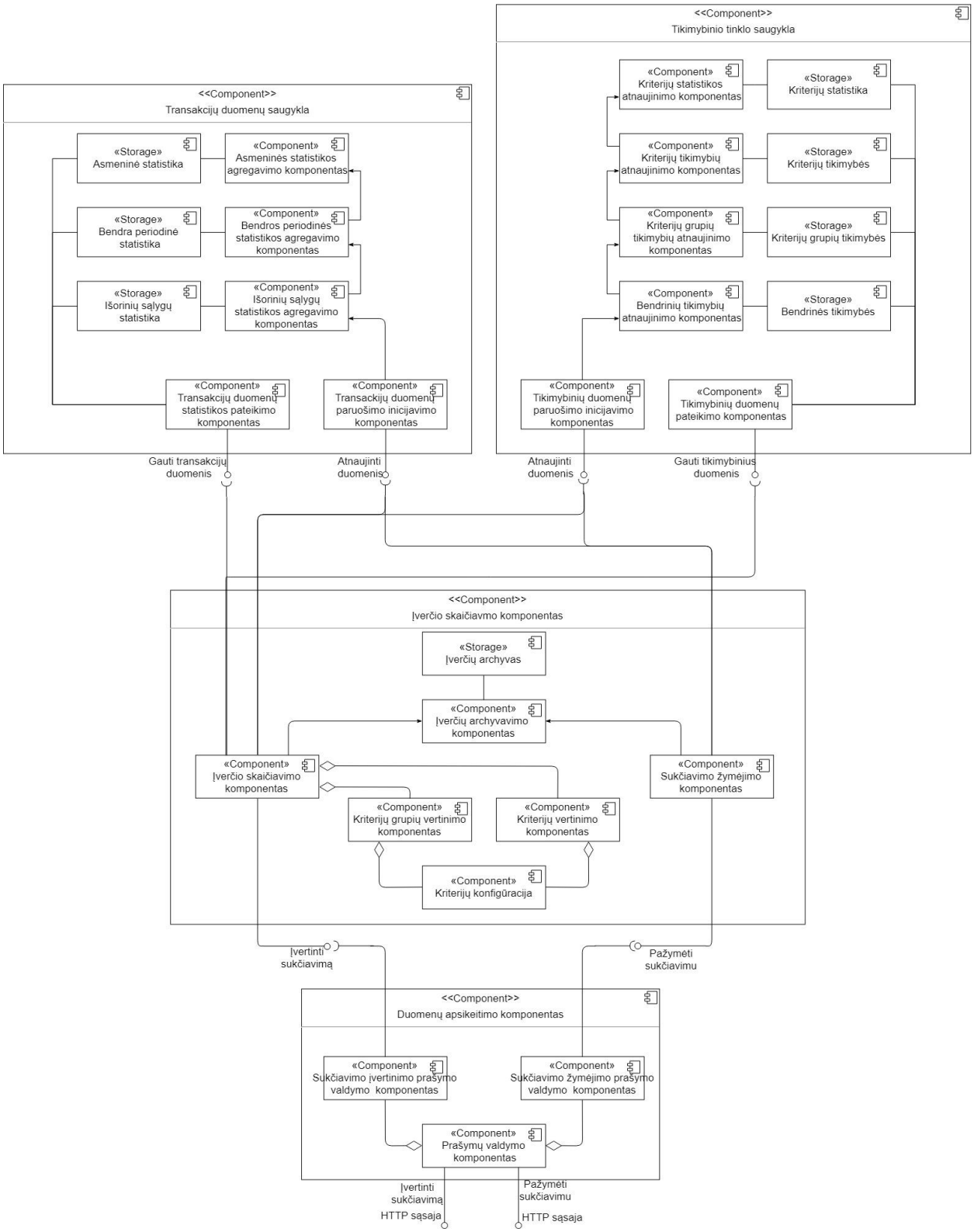
6 priedas. Sistemos komponentų integracija skaičiuojant sukčiavimo įvertį



8 pav. Sistemos komponentų integracijos skaičiuojant sukčiavimo įvertį, siūlomas įgyvendinimas

Duomenų apsikeitimo komponentas gavęs užklausą apdoroti transakciją, jos atributus perduoda įverčio skaičiavimo komponentui, kuris perduoda reikiamus atributus transakcijų duomenų saugyklos komponentui, kuris gražina reikiamus statistinius duomenis kriterijų rizikų įvertinimui. Skaičiavimo komponentas gavęs šiuos duomenis įvertina sukčiavimo kriterijų vertes ir perduoda jas tikimybinio tinklo saugyklai, kuri nurodytiems kriterijams gražina kriterijų verčių tikimybines vertes kurias įverčio skaičiavimo komponentas panaudoja apskaičiuoti galutinį sukčiavimo įvertį. Šis galutinis įvertis gražinamas duomenų apsikeitimo komponentui, inicijuojamas transakcijų bei tinklo duomenų saugyklų atnaujinimas nelaukiant atsako.

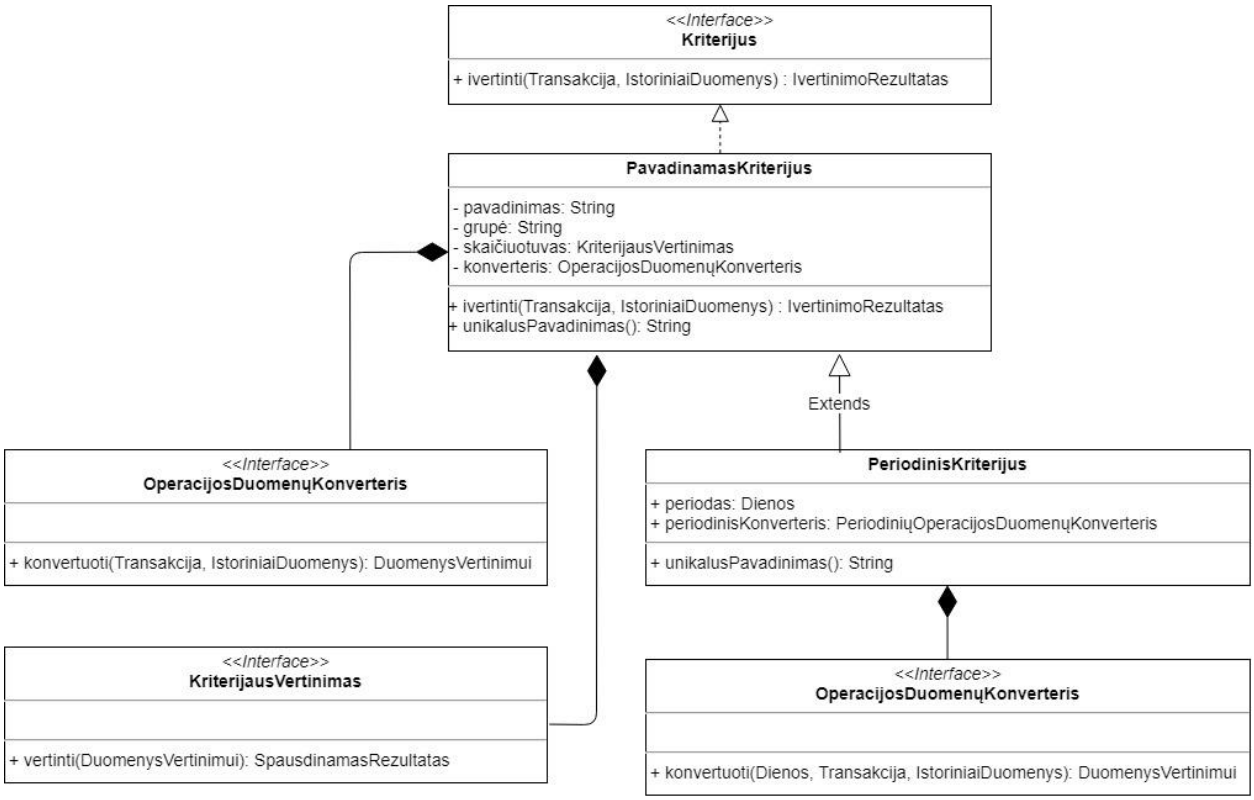
7 priedas. Sistemos komponentai



9 pav. Sistemos komponentų diagrama

Sistemos komponentų diagrama detalizuoja 4 loginius sistemos komponentus, bei apibrėžia komponentų tarpusavio sąsajas.

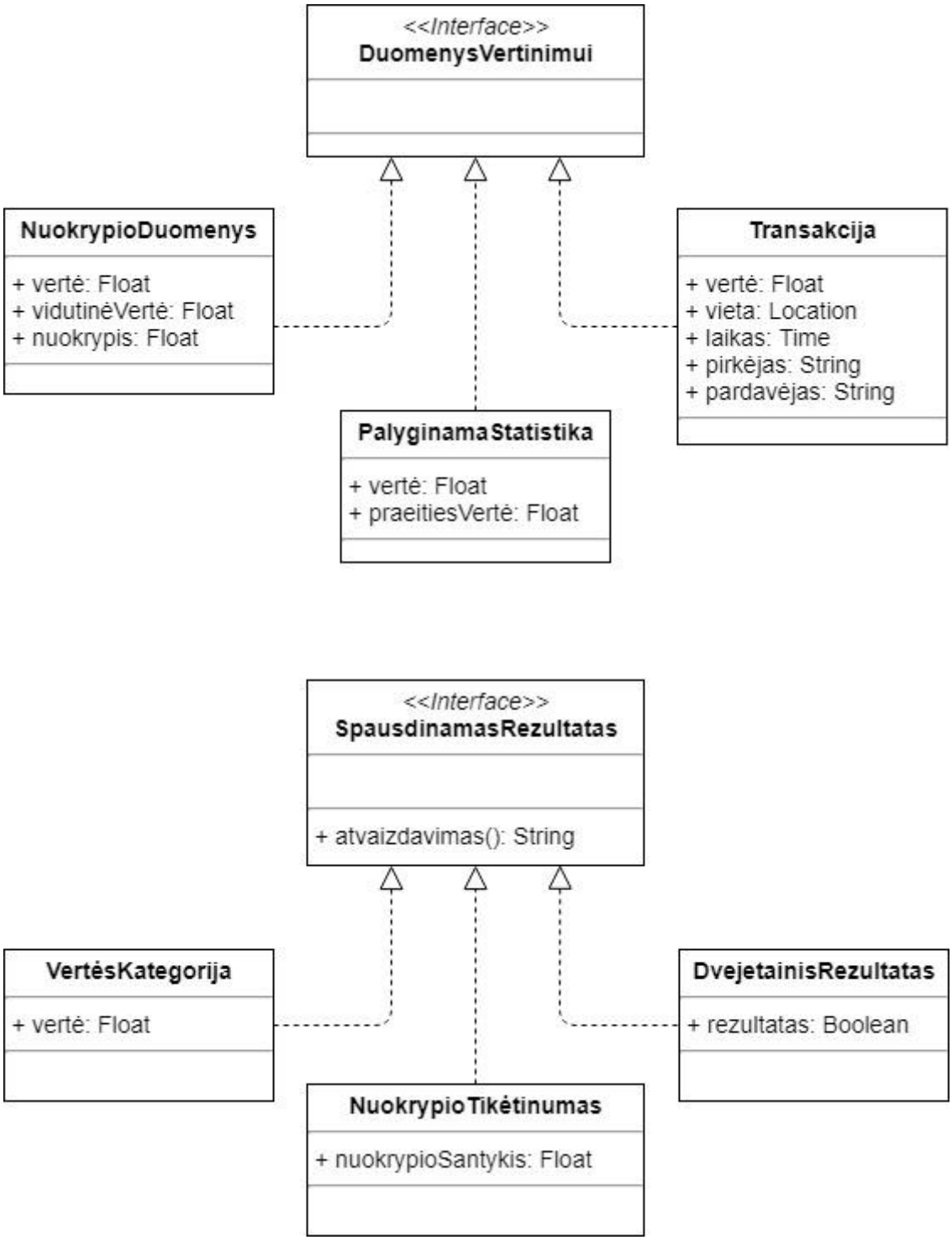
8 priedas. Kriterijų vertinimo komponento realizacija



10 pav. Kriterijų vertinimo komponento realizacija

Kriterijų vertinimo komponentų realizacija apibrėžia esybių kompoziciją įgalinančią sukčiavimo kriterijų apsirašymą sistemoje nedubliuojant skaičiavimų logikos. Pagrindinė kriterijaus realizacija – „PavadinamasKriterijus“ komponuojama iš operacijos duomenų konverterio ir kriterijaus vertinimo esybės. Kiekvienam kriterijui aprašomas unikalus operacijos duomenų konverteris iš transakcijos duomenų ištraukiantis duomenis su kuriais reikia atlikti skaičiavimus. Šie duomenys perduodami pasirinktai kriterijaus vertinimo realizacijai. Kadangi dauguma kriterijų panašūs, pakanka vos kelių kriterijų vertinimo realizacijų.

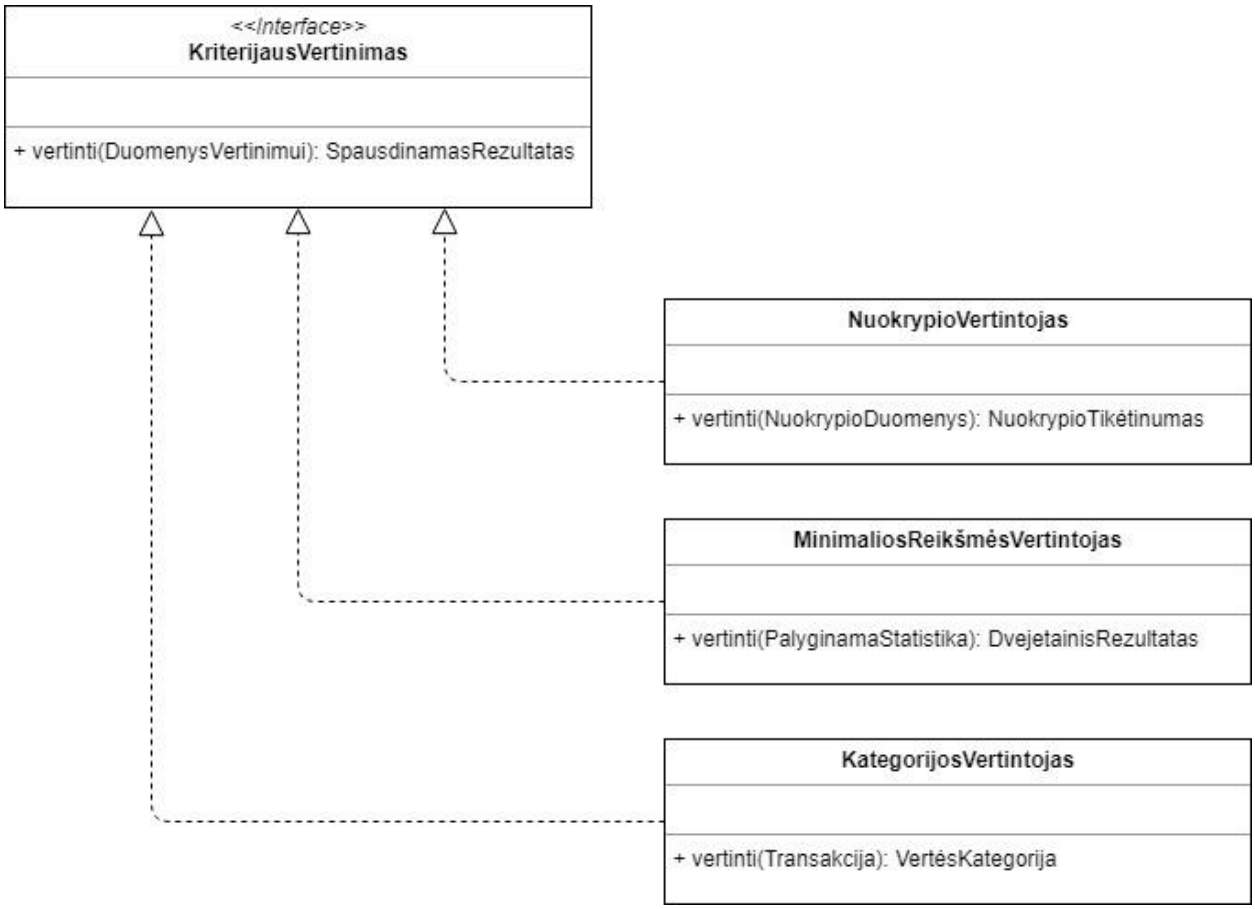
9 priedas. Kriterijų vertinimo komponento įeigos, išeigos



11 pav. Kriterijaus vertinimo įeigų ir išeigų realizacijos

Skirtingos kriterijų realizacijos priima skirtingas įeigas ir rezultato pavidalu gražina skirtingas išeigas. Todėl siekiant apibendrinti įeigos ir išeigos apdorojamą naudojamos duomenų vertinimui ir spausdinamo rezultato abstrakcijos. Skirtingos kriterijų vertinimo realizacijos reikalauja apsibrėžti duomenų vertinimui realizaciją bei spausdinamo rezultato realizaciją.

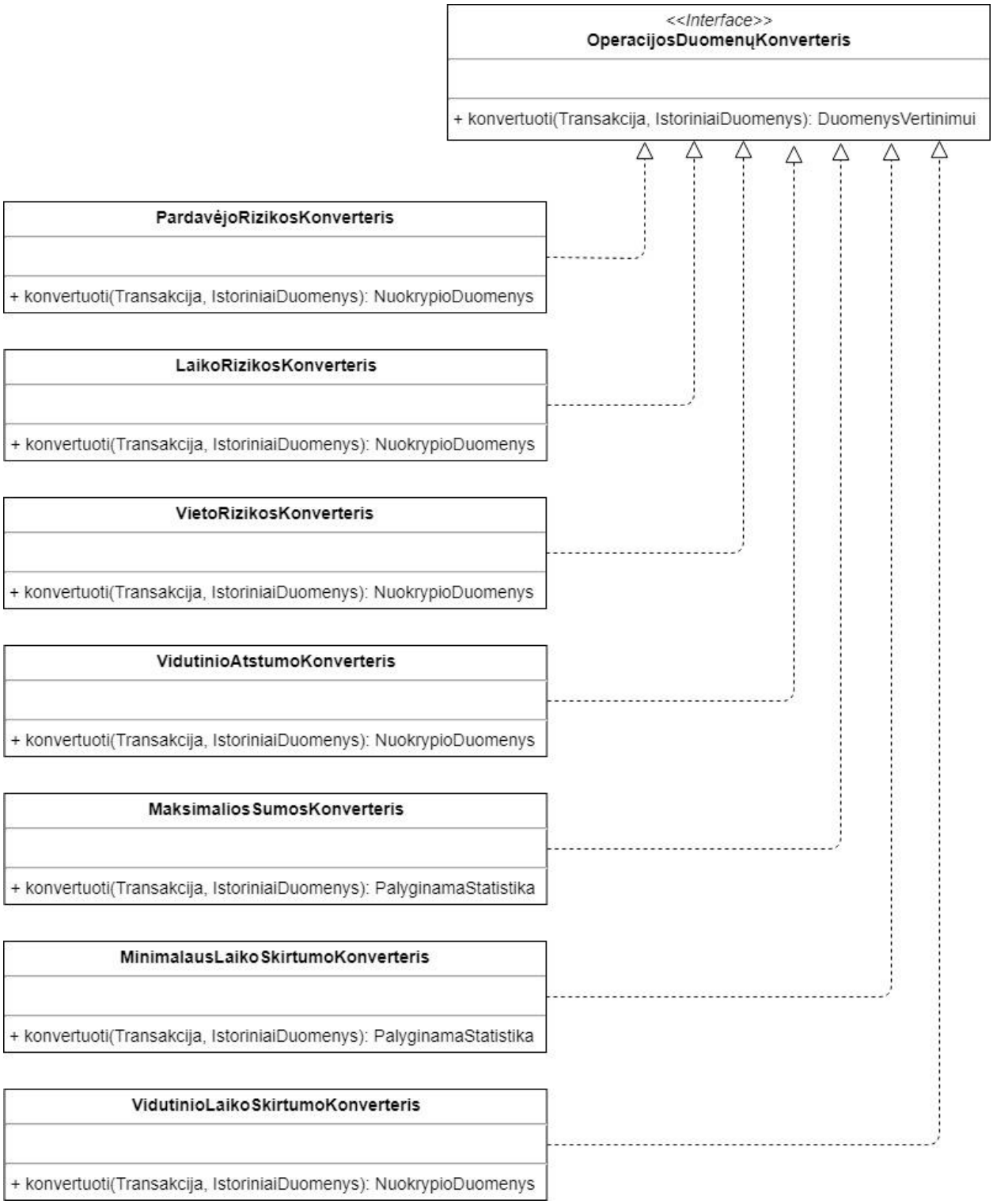
10 priedas. Kriterijų vertinimo realizacijos



12 pav. Kriterijų vertinimo realizacijos

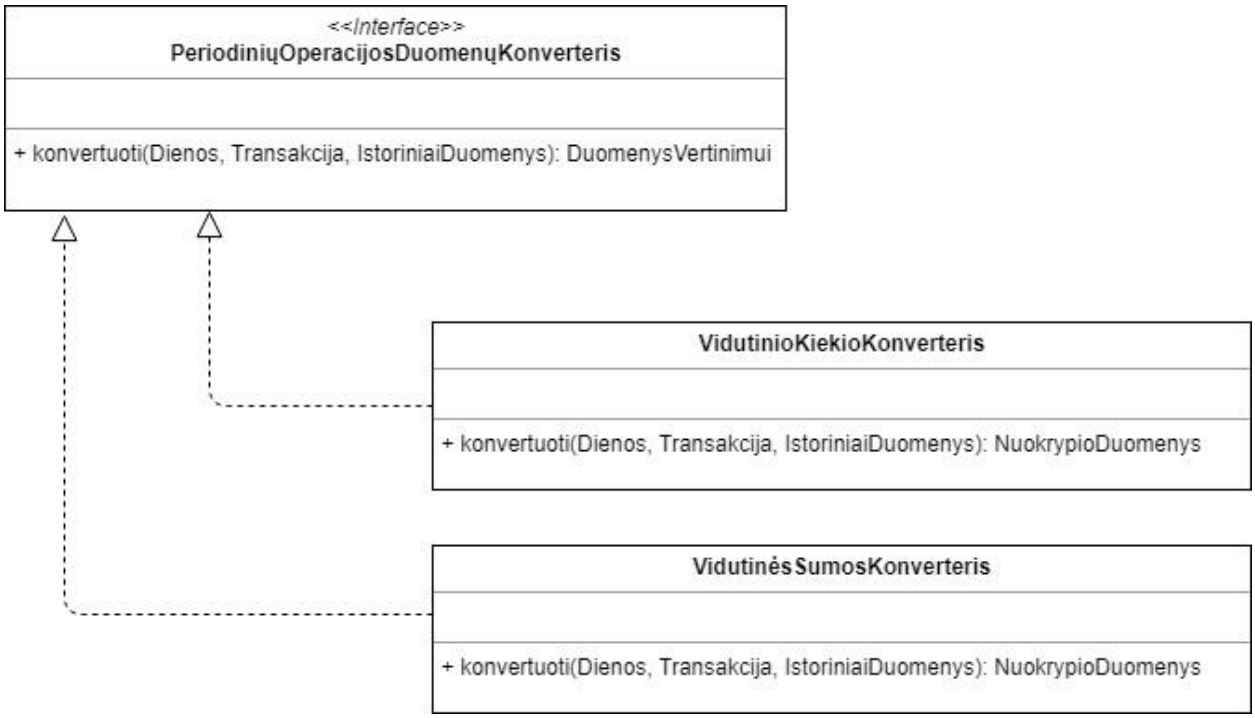
Pateikiamos sistemoje naudojamos kriterijų vertinimo realizacijos. Nuokrypio vertintojas geba palyginti konkrečią reikšmę su istoriniais duomenimis lygindamas pagal tai kiek standartinių nuokrypių nuo vidutinės reikšmės yra nutolusi reikšmės. Minimalios reikšmės vertintojas gali įvertinti ar reikšmė yra mažesnė nei pasitaikančios istoriniuose duomenyse. Kategorijos vertintojas gali priskirti reikšmę iš anksto apsibrėžtai kategorijai.

11 priedas. Duomenų vertinimui paruošimo realizacijos



13 pav. Duomenų paruošimo realizacijos

12 priedas. Periodinių duomenų vertinimui paruošimo realizacijos

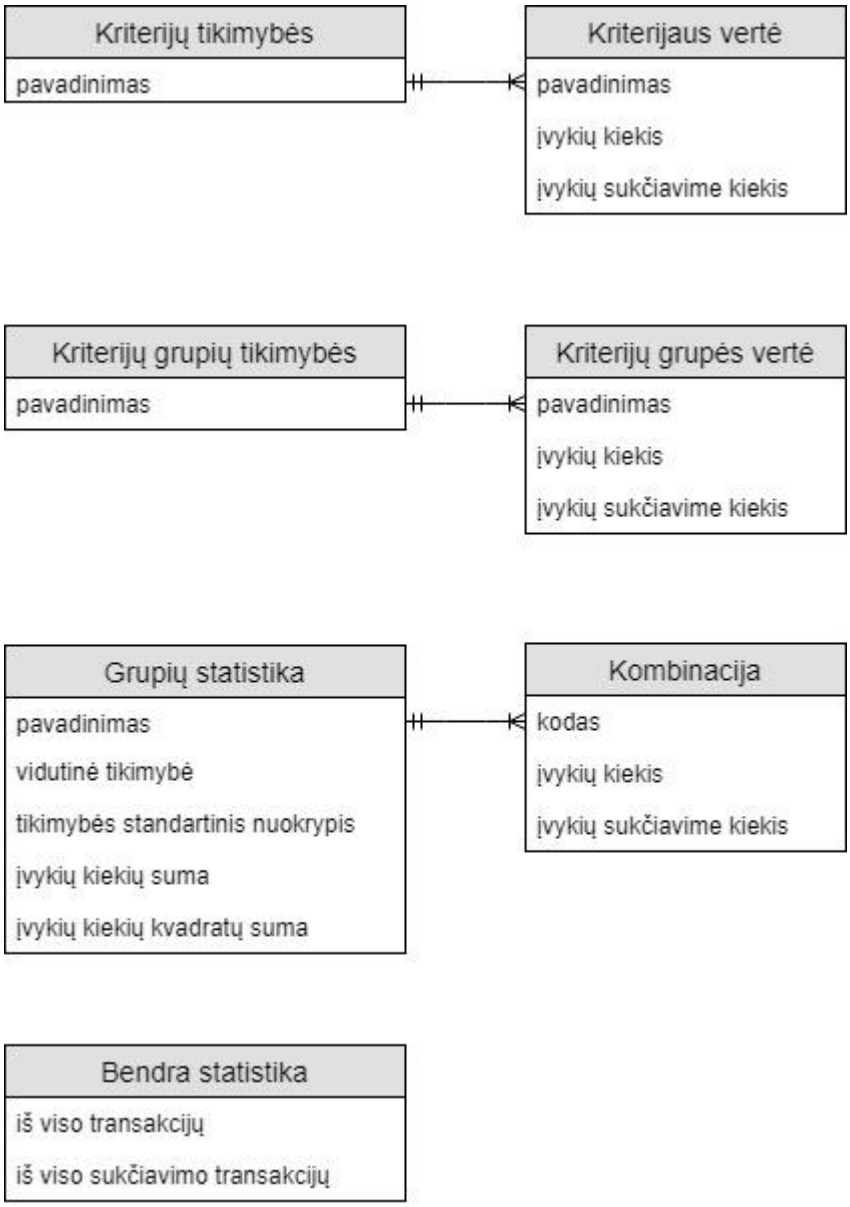


14 pav. Periodinių duomenų vertinimui paruošimo realizacijos

13 priedas. Tarpinių sukčiavimo aptikimo sistemos rezultatų pavyzdys

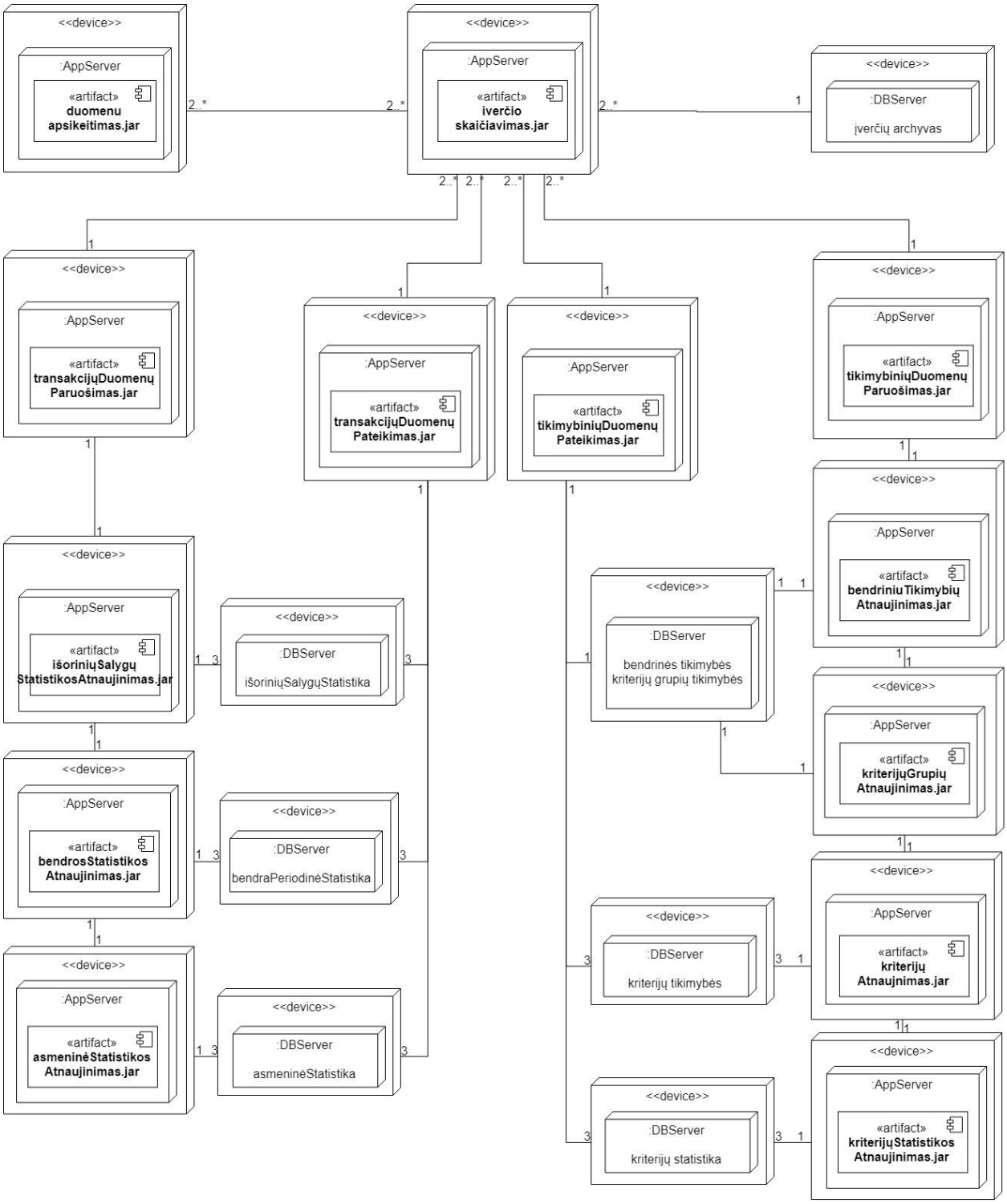

```
{
    "id" : "50070",
    "data" : {
      "amount" : 500,
      "debtor" : "11",
      "creditor" : "100",
      "time" : "2018-04-12T03:00:00.000",
      "longtitude" : 22.540000915527344,
      "latitude" : 50.25
    },
    "criteria" : [
      {
          "name" : "LOCATION",
          "value" : "VERY_HIGH_RISK",
          "criteria" : [
            {"CREDITOR_RISK" : "EXPECTED"}
            {"LOCATION_RISK" : "MUCH_MORE_THAN_EXPECTED"},
            {"AVERAGE_DISTANCE_FROM_LAST_LOCATION":"MORE_THAN_EXPECTED"},
            {"AVERAGE_DISTANCE_FROM_COMMON_LOCATION" : "LESS_THAN_EXPECTED"},
          ]
      },
      {
          "name" : "AMOUNT",
          "value" : "VERY_HIGH_RISK",
          "criteria" : [
            {"AVERAGE_SUM/P1D" : "MUCH_MORE_THAN_EXPECTED"},
            {"AVERAGE_SUM/P7D" : "MUCH_MORE_THAN_EXPECTED"},
            {"AVERAGE_SUM/P30D" : "MUCH_MORE_THAN_EXPECTED"},
            {"AVERAGE_PERIOD_AMOUNT_RATIO/P1D" : "EXPECTED"},
            {"AVERAGE_PERIOD_AMOUNT_RATIO/P7D" : "EXPECTED"},
            {"AVERAGE_PERIOD_AMOUNT_RATIO/P30D" : "MUCH_MORE_THAN_EXPECTED"},
            {"MAX_SPENT_AMOUNT" : "TRUE"},
            {"AMOUNT_CATEGORY" : "BIG_AMOUNT"}
          ]
      },
      {
          "name" : "COUNT",
          "value" : "EXPECTED_RISK",
          "criteria" : [
            {"AVERAGE_COUNT/P1D" : "MUCH_MORE_THAN_EXPECTED"},
            {"AVERAGE_COUNT/P7D" : "MUCH_MORE_THAN_EXPECTED"}
            {"AVERAGE_COUNT/P30D" : "MUCH_MORE_THAN_EXPECTED"}
          ]
      },
      {
          "name" : "TIME",
          "value" : "VERY_HIGH_RISK",
          "criteria" : [
            {"AVERAGE_TIME_BETWEEN_TRANSACTIONS":"MUCH_LESS_THAN_EXPECTED"},
            {"TIME_RISK":"MORE_THAN_EXPECTED"},
            {"MIN_TIME_BETWEEN_TRANSACTIONS":"TRUE"}
      }
    ]
}
```

14 priedas. Tinklo saugyklos esybių sąryšiai



15 pav. Tinklo saugyklos esybių sąryšiai

15 Priedas. Rekomenduojama tinklo diegimo konfigūracija

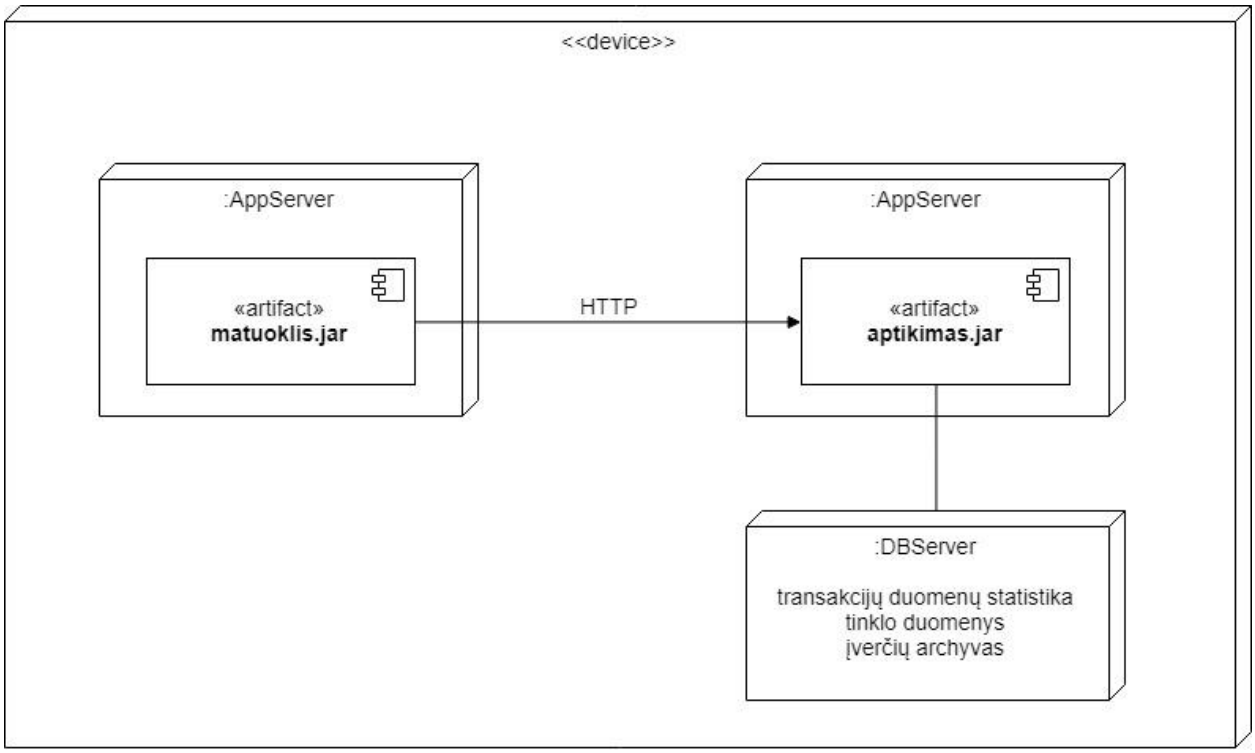


16 pav. Rekomenduojama diegimo konfigūracija

16 priedas. Pirmojo scenarijaus transakcijos tarpinio rezultato pavyzdys


```
{
    "id" : "255078",
    "data" : {
        "amount" : 350,
        "debtor" : "33",
        "creditor" : "203",
        "time" : "2018-04-13T22:00:00.000",
        "longtitude" : 22.540000915527344,
        "latitude" : 50.25
    },
    "criteria" : [
      {
        "name" : "LOCATION",
        "value" : "HIGH_RISK",
        "criteria" : [
          { "LOCATION_RISK" : "MUCH_MORE_THAN_EXPECTED" },
          { "AVERAGE_DISTANCE_FROM_LAST_LOCATION" : "MORE_THAN_EXPECTED" },
          { "AVERAGE_DISTANCE_FROM_COMMON_LOCATION" : "LESS_THAN_EXPECTED" },
          { "CREDITOR_RISK" : "EXPECTED" }
        ]
      },
      {
        "name" : "AMOUNT",
        "value" : "VERY_HIGH_RISK",
        "criteria" : [
          { "AVERAGE_SUM/P7D" : "MUCH_MORE_THAN_EXPECTED" },
          { "AVERAGE_PERIOD_AMOUNT_RATIO/P1D" : "EXPECTED" },
          { "AVERAGE_SUM/P30D" : "MORE_THAN_EXPECTED" },
          { "AVERAGE_PERIOD_AMOUNT_RATIO/P30D" : "MUCH_MORE_THAN_EXPECTED" },
          { "AVERAGE_PERIOD_AMOUNT_RATIO/P7D" : "EXPECTED" },
          { "AVERAGE_SUM/P1D" : "MUCH_MORE_THAN_EXPECTED" },
          { "MAX_SPENT_AMOUNT" : "FALSE" },
          { "AMOUNT_CATEGORY" : "BIG_AMOUNT" }
        ]
      },
      {
        "name" : "COUNT",
        "value" : "EXPECTED_RISK",
        "criteria" : [
          { "AVERAGE_COUNT/P1D" : "MUCH_MORE_THAN_EXPECTED" },
          { "AVERAGE_COUNT/P30D" : "MUCH_MORE_THAN_EXPECTED" },
          { "AVERAGE_COUNT/P7D" : "MUCH_MORE_THAN_EXPECTED" }
        ]
      },
      {
        "name" : "TIME",
        "value" : "VERY_HIGH_RISK",
        "criteria" : [
          { "AVERAGE_TIME_BETWEEN_TRANSACTIONS" : "LESS_THAN_EXPECTED" },
          { "TIME_RISK" : "MUCH_MORE_THAN_EXPECTED" },
          { "MIN_TIME_BETWEEN_TRANSACTIONS" : "FALSE" }
        ]
      }
    ]
}
```

17 Priedas. Tikslumo vertinimo sistemos diegimo konfigūracija



17 pav. Tikslumo vertinimo sistemos diegimo konfigūracija

18 Priedas. Tikslumo vertinimo rezultatų pasiskirstymas

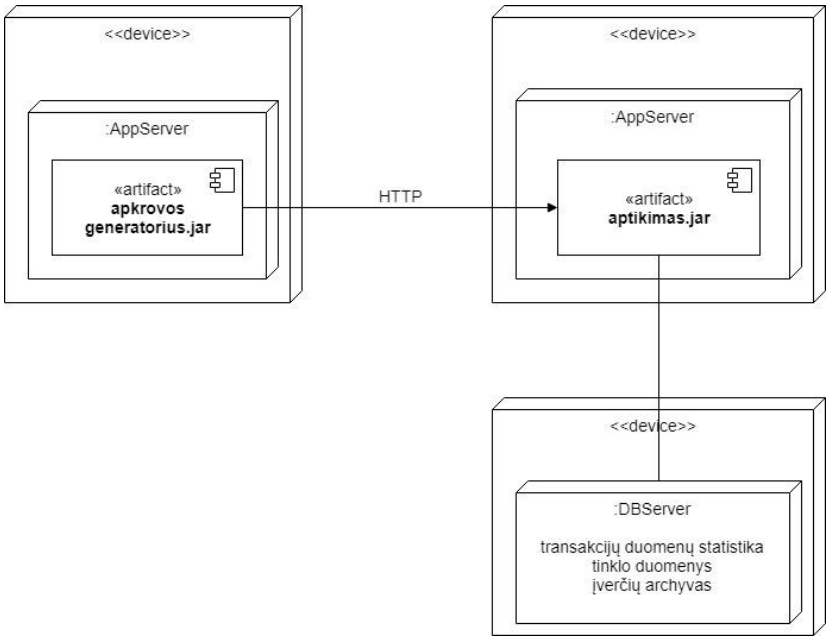
4 lentelė. Tikslumo vertinimo rezultatų pasiskirstymas

	0,0 - 0,1	0,1 - 0,2	0,2 - 0,3	0,3 - 0,4	0,4 - 0,5	0,5 - 0,6	0,6 - 0,7	0,7 - 0,8	0,8 - 0,9	0,9 - 1,0
Scenarijus 1	0%	0%	0%	30%	0%	0%	0%	0%	0%	70%
Scenarijus 2	0%	0%	0%	52%	0%	0%	0%	0%	0%	48%
Scenarijus 3	90%	0%	0%	10%	0%	0%	0%	0%	0%	0%
Scenarijus 4	98%	0%	0%	0%	0%	2%	0%	0%	0%	0%
Scenarijus 5	100%	0%	0%	0%	0%	0%	0%	0%	0%	0%

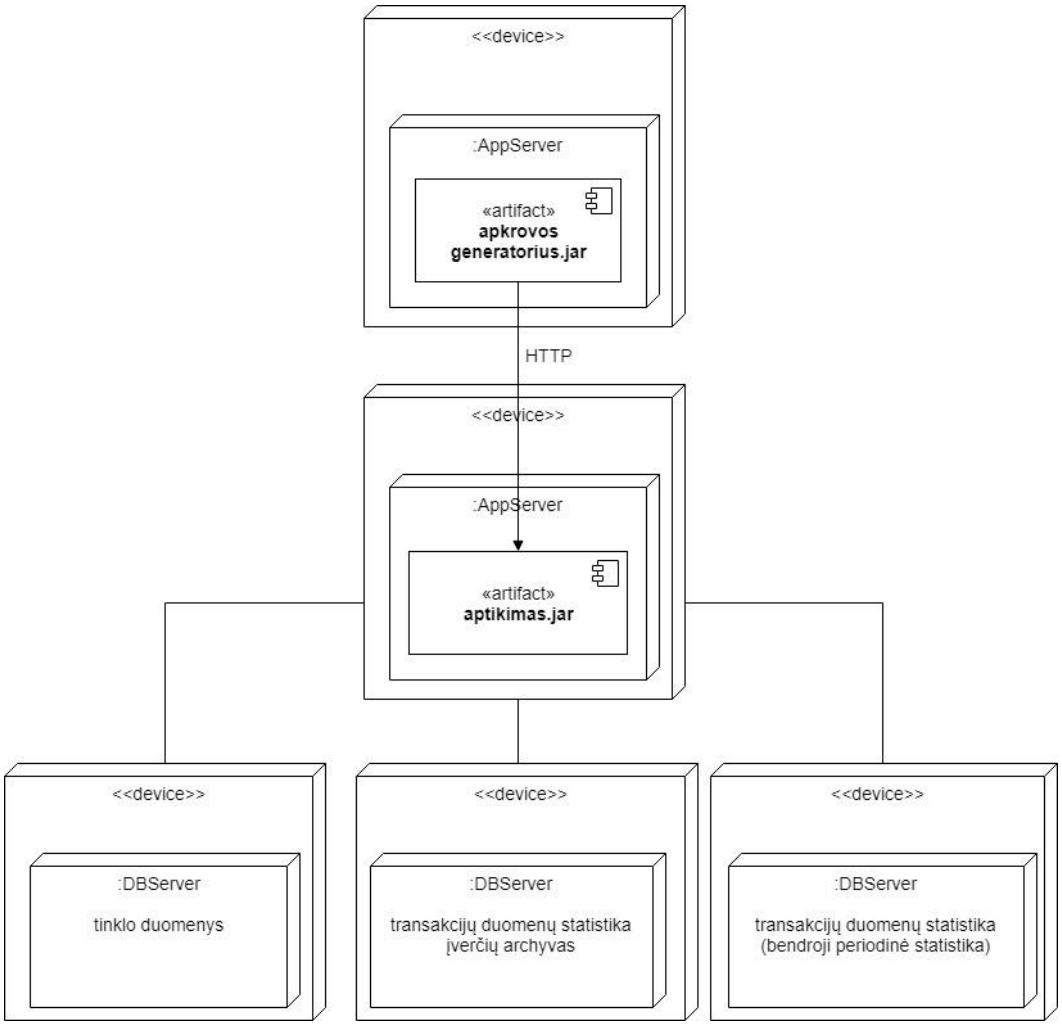
5 lentelė. Tikslumo vertinimo scenarijų statistika

	Scenarijus 1	Scenarijus 2	Scenarijus 3	Scenarijus 4	Scenarijus 5
Vidurkis	0,7939	0,6497	0,0537	0,0123	0,0017
Standartinis nuokrypis	0,3009	0,3283	0,0945	0,0730	0,0024
Minimali reikšmė	0,3342	0,3342	0,0053	0,0000	0,0000
Maksimali reikšmė	0,9989	0,9989	0,3342	0,5232	0,0053

19 Priedas. Efektyvumo vertinimo diegimo konfigūracijos

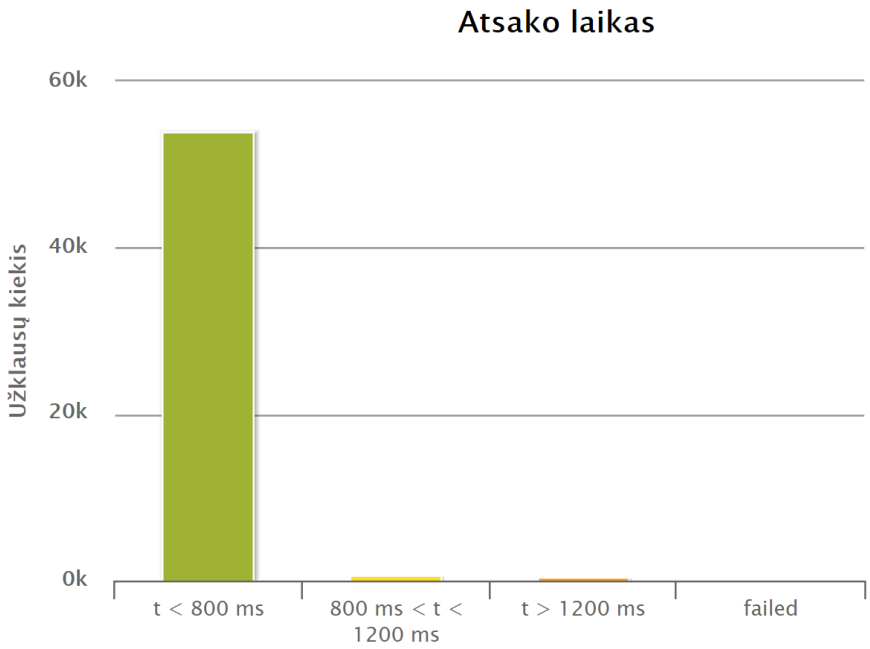


18 pav. Efektyvumo bandymo mažo diegimo konfigūracija

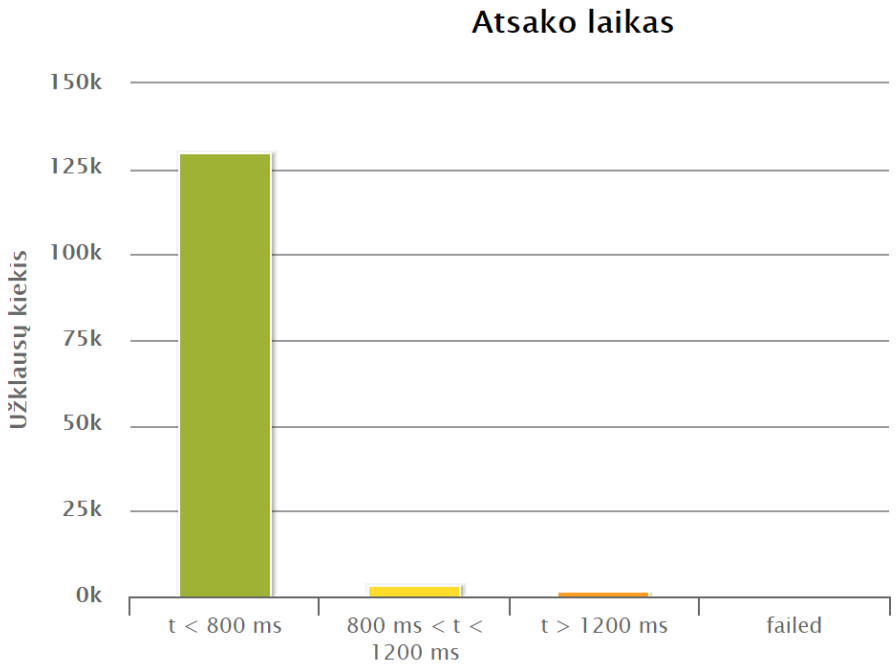


19 pav. Efektyvumo bandymo vidutinio diegimo konfigūracija

20 Priedas. Atsako laiko pasiskirstymas



20 pav. Mažos diegimo konfigūracijos bandymo atsako laiko pasiskirstymas



21 pav. Vidutinės diegimo konfigūracijos bandymo atsako laiko pasiskirstymas

21 Priedas. Papildomi priedai

Kartu su darbu fizinėje laikmenoje pateikiami papildomi priedai. Laikmenoje pateikiama:

- Java virtualiojoje mašinoje vykdomas sukčiavimo aptikimo sistemos artefaktas.
  Laikmenoje – priedai/aptikimas.jar;
- sukurtos sukčiavimo aptikimo sistemos išeities kodas.
  Laikmenoje – priedai/išeities_kodai/fraud_detection.
  Versijų kontrolės sistemoje: https://github.com/hudas/fraud-detection;
- duomenų generatoriaus bandymams išeities kodas.
  Laikmenoje – priedai/išeities_kodai/data_generator;
- duomenų pateikimo tikslumo bandymams išeities kodas.
  Laikmenoje – priedai/išeities_kodai/data_publisher;
- apkrovos generatoriaus įrankio konfigūracija
  Laikmenoje – priedai/išeities_kodai/fraud-meter;
- efektyvumo vertinimo tarpiniai rezultatai ir duomenų rinkiniai
  Laikmenoje – priedai/efektyvumo_vertinimas;
- tikslumo vertinimo tarpiniai rezultatai ir duomenų rinkiniai.
  Laikmenoje – priedai/tikslumo_vertinimas.