Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií
FIIT-XXXX-XXXX

Bc. Zoltán Csengődy

IDENTIFIKÁCIA BEZPEČNOSTNÝCH RIZÍK A ANALÝZA DÁT Z PROSTREDIA POČÍTAČOVÝCH SIETÍ

Diplomová práca

Študijný program:	Inteligentné softvérové systémy
Študijný odbor:	9.2.5 Softvérové inžinierstvo, 9.2.8 Umelá inteligencia
Miesto vypracovania:	Ústav počítačového inžinierstva a aplikovanej informatiky
Vedúci práce:	Ing. Rudolf Grežo

máj 2019

Čestne vyhlasujem, že som túto prácu vypracoval samostatne, na základe konzultácií a s použitím uvedenej literatúry.

V Bratislave, 28.05.2019

Bc. Zoltán Csengődy

Anotácia

Slovenská technická univerzita v Bratislave
FAKULTA INFORMATIKY A INFORMAČNÝCH TECHNOLÓGIÍ
Študijný program: Inteligentné softvérové systémy

Autor: Bc. Zoltán Csengődy
Diplomová práca: Identifikácia bezpečnostných rizík a analýza dát z prostredia počítačových sietí
Vedúci diplomovej práce: Ing. Rudolf Grežo

máj 2019

Počítačové siete nás sprevádzajú každodenným životom, pričom jedným z aspektov pri práci s nimi je zvýšenie spoľahlivosti a bezpečnosti siete. S rozvojom tejto technologickej oblasti prichádzajú nové spôsoby a typy útokov, voči ktorým sa treba chrániť. Táto práca je venovaná výskumu v oblasti odhalenia počítačových útokov metódami strojového učenia. Cieľom tejto diplomovej práce je vytvorenie programového modulu, ktorý vhodným spôsobom dokumentuje vybrané algoritmy strojového učenia. Hlavnou motiváciou je vytvorenie jednotnej analýzy vplyvov rôznych nastavení klasifikačných algoritmov a rôznych spôsobov predspracovania vybraných dátových množín na výsledky odhalenia sieťových útokov. Súčasťou tejto práce je vlastný návrh riešenia, ktorý vyplýva z faktu, že hlavným nedostatkom použitia algoritmov strojového učenia je nedostatočná dokumentácia použitia, tvorba architektúry a nastavenia parametrov. Dnešný spôsob použitia týchto metód spočíva predovšetkým v skúšaní a optimalizácii najlepšieho riešenia pre daný model. Na základe rôznych nastavení a vstupov do metód dokážeme optimalizovať klasifikáciu a tým pádom pri vhodných nastaveniach dosahovať lepšie výsledky hodnotenia modelu. V tejto práci sa venujeme hľadaniu anomálií v sieťovej premávke a metódam, ktoré sú určené na ich odhaľovanie. Súčasťou práce je taktiež vhodné predspracovanie dát vybranej dátovej množiny a odhalenie závislostí medzi jeho atribútmi, ktoré majú značný vplyv na odhaľovanie útokov. Výstupom tejto práce je programový modul, ktorý porovnáva výhody a nevýhody použitých metód strojového učenia a výsledky interpretuje textovým aj grafickým spôsobom.

Annotation

Slovak University of Technology in Bratislava
FACULTY OF INFORMATICS AND INFORMATION TECHNOLOGIES
Degree course: Intelligent software systems

Author: Bc. Zoltán Csengődy
Master's thesis: Data analysis and security risk identification in computer networks
Supervisor: Ing. Rudolf Grežo

2019, May

Computer networks accompany us with everyday life, and one aspect of working with them is to increase network reliability and security. With the development of this technological area, new ways and types of attacks are coming, against which we must protect ourselves. This work is devoted to research in the field of detection of computer attacks by machine learning methods. The aim of this master thesis is to create a program module, which in an appropriate way documents the selected machine learning algorithms. The main motivation is to create a unified analysis of the effects of the different settings of the classification algorithms and the different methods of pre-processing the selected datasets to the results of network attack detection. Part of this work is a custom design solution, which stems from the fact that the main drawback of using machine learning algorithms is insufficient documentation of the use, creation of architecture and parameter settings. Today's use of these methods lies primarily in testing and optimizing the best solution for a given model. Based on the various settings and inputs to the methods we can optimize the classification and thus achieve better model evaluation results with appropriate settings. In this work, we look for anomalies in network traffic and methods intended to detect them. Part of the work is also a suitable pre-processing of the selected data set and revealing the dependencies between its attributes, which have a significant impact on the detection of attacks. The outcome of this work is a program module, that compares advantages and disadvantages of used methods of machine learning and interprets the results in both text and graphical way.

Obsah

1. Úvod **1**

2. Analýza problematiky **3**

2.1. Typy systémov na detekciu sieťových útokov **3**

2.1.1. Anomálne založené detekčné systémy 5

2.1.2. Charakteristiky IDS 7

2.2. Architektúra IDS **9**

2.3. Existujúce nástroje **10**

2.4. Spôsob vyhodnocovania IDS **12**

2.5. Sieťové útoky **16**

2.5.1. Detekcia anomálií 16

2.5.2. Detekcia zneužitia 17

2.5.3. Monitorovanie cieľa 17

2.5.4. Špionáž 17

2.5.5. Typy sieťových útokov 18

2.5.6. Odhalenie sieťových útokov 20

2.6. Strojové učenie **21**

2.6.1. Klasifikačné algoritmy 22

2.6.2. Neurónová sieť 25

2.7. Dátové množiny **28**

2.7.1. NSL-KDD 28

2.7.2. UNSW-NB15 30

2.7.3. ISCX 32

2.7.4. Predspracovanie dát 34

2.8. Zhodnotenie analýzy **35**

3. Špecifikácia požiadaviek **38**

3.1. Funkčné vlastnosti **38**

3.2. Nie-funkčné vlastnosti **38**

4. Návrh riešenia **39**

Bibliografia **40**

Príloha A: Plán práce **A-1**

Príloha B: Technická dokumentácia B-1

Príloha C: Obsah elektronického média C-1

Zoznam obrázkov

Obrázok 1 – Všeobecná architektúra IDS [23] 3
Obrázok 2 – Architektúra HIDS [23] 4
Obrázok 3 – Architektúra NIDS [23] 4
Obrázok 4 – Bodová, kontextová a kolektívna anomália [6] 6
Obrázok 5 – Funkcionality IDS [32] 9
Obrázok 6 – Confusion matrix [5] 13
Obrázok 7 – ROC krivka [27] 15
Obrázok 8 – Návrh platformy na odhalenie sieťových útokov [16] 20
Obrázok 9 – Prostredie odhalenia útoku [16] 21
Obrázok 10 – Štruktúra modulu na detekciu sieťových útokov [16] 21
Obrázok 11 – Neurón [24] 25
Obrázok 12 – Feed-forward neurónová sieť [10] 26
Obrázok 13 – Model neurónovej siete pre IDS [24] 27
Obrázok 14 – Proces klasifikácie sieťových útokov [24] 27

Zoznam tabuliek

Tabuľka 1 – Typy IDS [17] 5
Tabuľka 2 – Útoky v testovacom súbore dát NSL-KDD [22] 29
Tabuľka 3 – Tabuľka výsledkov experimentu v nástroji WEKA [8] 30
Tabuľka 4 – Tabuľka výsledkov pre sieťovú forenznú schému [19] 32
Tabuľka 5 – Tabuľka presností klasifikačných algoritmov pre ISCX IDS 2012 [29] 34
Tabuľka A.6 – Plán práce k DP I A-1
Tabuľka A.7 – Plán práce k DP II A-1
Tabuľka A.8 – Plán práce k DP III A-2

1. Úvod

S rozvojom internetových technológií, počítačové siete postupne menia životy ľudí a čoraz viac uľahčujú prácu a spôsob práce ľudí. Rozvoj tejto oblasti je veľmi rýchly a tým pádom je aj čoraz zraniteľnejší voči počítačovým útokom. S rozvojom tejto technologickej oblasti prichádzajú nové spôsoby a typy útokov.

Nebezpečenstvo počítačového útoku a jeho zabránenie je dôležitým aspektom, ktorý sa výskumníci v tejto oblasti snažia vyriešiť. Keďže počítačová sieť môže byť otvorená (voľne dostupná) a medzinárodne zdieľaná tak údaje, ktoré sú v nej prenášané nie sú v bezpečí. Preto je potrebné zaviesť technické opatrenia na zabezpečenie ochrany údajov v sieťovom prostredí.

Zhromažďovanie údajov v počítačovej sieti môže výrazne pomôcť pri detekcii sieťových útokov a pomáhať pri správe siete. Vďaka monitorovaniu, testovaniu, kontrole a vyhodnocovaniu v reálnom čase sú správcovia siete schopní získať informácie o výkonnosti sieťového systému, vyhodnotiť kvalitu služieb (QoS) a zistiť poruchu siete. Vďaka napredujúcej technológie 5G a podpore technológie Internet vecí (IoT), veľkoplošné a vysokorýchlostné siete sa stávajú súčasťou výskumu a vývoja s cieľom účinne zhromažďovať a analyzovať údaje o sieti.

Danú problematiku je potrebné riešiť z dôvodu predchádzania škodlivým útokom prostredníctvom predikcie na základe analýze dát. V rámci riešenia tejto problematiky je dôležité navrhnúť, vytvoriť a implementovať bezpečnostné metódy na zabránenie takýchto útokov. Definícia pojmu sieťová bezpečnosť je podľa autorov Xia a Wang [35] nasledovná: „Pomocou zachytávania a integrácie všetkých druhov informácií, ktoré odrážajú bezpečnostný stav, možno predpovedať trend bezpečnosti siete“ (Xia Wei-Wei a Wang Hai-Feng, 2010, str. 616). Predčasné odhalenie škodlivej činnosti zabezpečí lepšiu ochranu siete od budúcich trendov v tejto oblasti, ktoré prinášajú nové, komplexné a sofistikovanejšie útoky. Taktiež zabezpečuje vybudovanie nákladovo efektívnej stratégie v prípade nového útoku.

Táto práca je venovaná analýze dát z prostredia počítačových sietí. Vzhľadom na identifikáciu bezpečnostných rizík a útočnej premávky je potrebné analyzovať existujúce algoritmy a procesy zamerané na spracovanie dát tohoto typu. Naším cieľom je analýza súčasného stavu problematiky a metód použiteľné pri analýze dát z prostredia počítačových sietí. V tejto práci sa venujeme analýze systémom na detekciu sieťového narušenia - IDS, ich architektúre a spôsobu vyhodnocovania. Taktiež sa venujeme sieťovým útokom, opisujeme typy sieťových útokov a priblížime spôsob ich odhalenia. Bližšie opisujeme strojové učenie a jednotlivé algoritmy strojového učenia. Zvlášť v rámci kapitoly 2.6 Strojové učenie sa venujeme neurónovým sieťam. V našej práci sme pri výbere zohľadňovali kritériá, ktoré sú kladené na dnešné moderné metódy dolovania v dátach. To nás priviedlo k umelej inteligencii - neurónovým sieťam. Tie poskytujú v dnešnej dobe veľmi intuitívne

a moderné riešenia. Táto metóda klasifikácie je jedna z najpresnejších. Klasifikačná metóda ako náhodný les (Random Forest) má tiež veľký potenciál. Na záver analýzy sa venujeme vybraným dátovým množinám, kde uvádzame výsledky experimentov iných autorov pre budúce porovnanie s výsledkami našej práce a metódy predspracovania týchto dátových množín.

Táto práca má dostatočne priblížiť, vysvetliť jednotlivé pojmy a súvislosti medzi nimi a uviesť čitateľa do danej problematiky.

2. Analýza problematiky

V tejto kapitole sa budeme venovať analýze systémov na detekciu sieťových útokov (Intusion Detection System - IDS) a priblížime jednotlivé prístupy detekcií sieťových útokov.

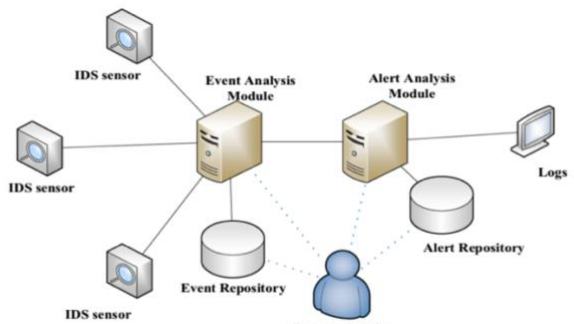
2.1. Typy systémov na detekciu sieťových útokov

Na základe predpokladu, že správanie sa útočníka na sieti je odlišné od bežného správania sa používateľa môžeme takúto sieťovú premávku identifikovať ako útočnú. Prostredníctvom skúmania anomálií v sieti je možné detegovať známe, ale aj neznámy typy útokov. Anomáliám sa venujeme v kapitole 2.1.1 Anomálne založené detekčné systémy.

Systémy na detekciu sieťových útokov sú implementované ako druhá obranná línia popri autentifikácií používateľa a ďalších bezpečnostných mechanizmov. IDS je softvér, hardvér alebo ich kombinácia, ktorý monitoruje počítačovú sieť pre odhalenie škodlivých aktivít alebo narušení siete. Narušenie siete je akt odhalenia nepriateľského používateľa (útočníka), ktorý sa pokúša získať neoprávnený prístup do siete alebo sa snaží narušiť služby a odmietnuť služby legitímnym používateľom. Pri odhalení narušenia siete, IDS systémy vytvárajú správy pre správcov bezpečnosti siete, ktorí sa rozhodnú o ďalších postupoch zaobchádzania sa s narušením. Tieto systémy môžu byť nasadené priamo u používateľa siete alebo priamo integrované v sieti na analýzu sieťovej premávky.

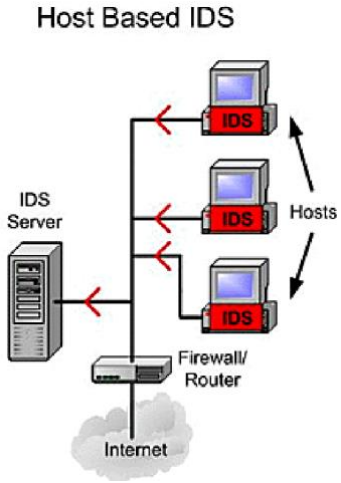
Bhattacharyya a Kalita [6] vo svojej knihe uvádzajú nasledujúce metódy detekcie anomálií v počítačovej sieti:

- **Intrusion Detection System (IDS)** – Systémy detekcie narušenia je nasadzovaný ako druhá obranná línia spolu s ďalšími preventívnymi bezpečnostnými mechanizmami, ako je autentifikácia používateľov a kontrola prístupu. Tento systém detekcie narušenia je založený na tom, že správanie útočníka je výrazne odlišné od správania bežného používateľa. Na nasledujúcom obrázku môžete vidieť všeobecnú architektúru IDS.



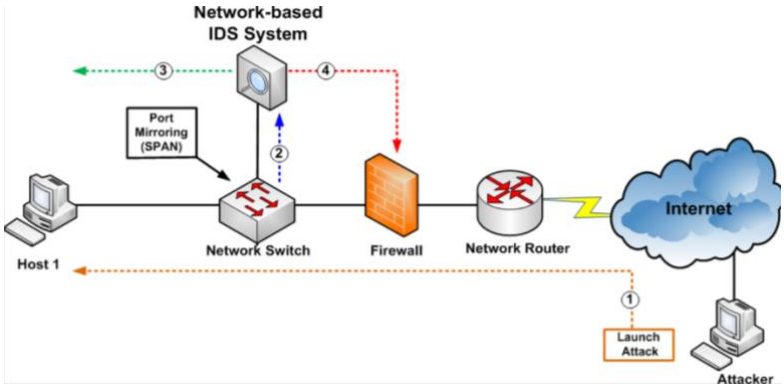
Obrázok 1 – Všeobecná architektúra IDS [23]

- **Host-based IDS (HIDS)** – Systém monitoruje vnútro výpočtového systému a sleduje, či niekto neobišiel bezpečnostnú politiku z vnútra (interne) alebo z vonka (externe). Podľa Jajish [13] HISD je softvérová aplikácia (agent) nainštalovaný na pracovných staniciach, ktoré sa majú monitorovať. Agenti monitorujú operačný systém a zapisujú údaje do protokolových súborov a/alebo spúšťajú poplach. Tento typ systému môže monitorovať iba jednotlivé pracovné stanice na ktorých sú agenti nainštalovaní, nemôže monitorovať celú sieť. HIDS môže zistiť internú aktivitu, ako napríklad program, ktorý pristupuje k zdrojom a pokúša sa k neoprávnenému prístupu.



Obrázok 2 – Architektúra HIDS [23]

- **Network-based IDS (NIDS)** – Keďže sieť je prepojená väčšinou s internetom pre komunikáciu so zvyškom sveta, tak NIDS číta všetky prichádzajúce pakety alebo toky a snaží sa nájsť podozrivé vzory. Podľa [13] Zvyčajne sa skladajú zo sieťového zariadenia s kartou sieťového rozhrania (NIC) pracujúci v promiskuitnom režime. V prípade ak je paket prepojený s podpisom útočníka, tak je generované upozornenie alebo je paket zaznamenaný do databázy.



Obrázok 3 – Architektúra NIDS [23]

Autori Liu, Yan a Pedrycz [17] vo svojej práci IDS ďalej rozdeľujú do päť typov (viď. tabuľku č. 1):

- Anomaly-Based Intrusion Detection System (ABIDS)
- Knowledge-Based Intrusion Detection System (KBIDS)
- Specification-Based Intrusion Detection System
- Hybrid Intrusion Detection System (HIDS)
- Other Intrusion Detection System (OIDS)

Tabuľka 1 – Typy IDS [17]

IDS	**Popis**
ABIDS	Odvodzuje model (profil) podľa prijateľných činností a správania a generuje poplach, ak sa monitorované činnosti alebo správanie sa výrazne odlišuje od tohto profilu.
KBIDS	Zachováva vzory konkrétnych útokov a spustí poplach ak sa pozorované udalosti zhodujú so vzormi.
SBIDS	Vyberá špecifikácie, ktoré definujú správne operácie siete s určitými obmedzeniami a identifikuje narušenie ak sa monitorované operácie odlišujú od špecifikácie.
HIDS	Je kombináciou ABIDS, KBIDS a SBIDS.
OIDS	Nepatrí do vyššie uvedených typov IDS.

2.1.1. Anomálne založené detekčné systémy

Anomálie v sieťach sa môžu vyskytnúť z viacerých dôvodov, napríklad z dôvodu prevádzky siete či z dôvodu škodlivej činnosti. Na základe takýchto výkyvov v sieti dokážeme relatívne ľahko identifikovať činnosť, ktorá je odlišná od bežnej činnosti siete a určiť, či je táto činnosť škodlivá alebo nie. Bhattacharyya a Kalita [6] vo svojej knihe tvrdia, že anomálie v sieti sú detekovateľné prostredníctvom strojového učenia, ktoré odhaľuje dve hlavné oblasti/kategórie vplyvu anomálií na počítačovú sieť. Anomálie podľa [6] môžu ovplyvňovať výkonnosť a bezpečnosť siete.

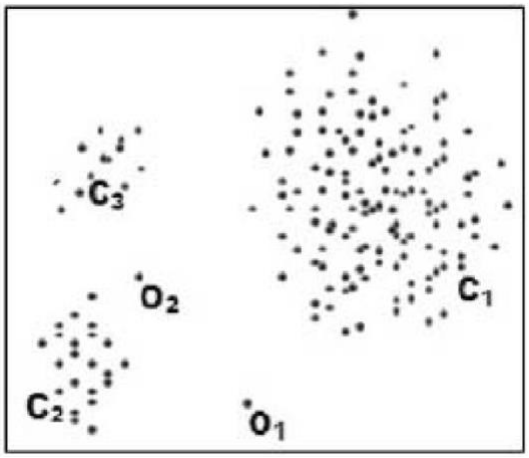
Našim hlavným cieľom je venovanie sa anomáliám spôsobujúce bezpečnostné riziká v sieti, a to konkrétne anomáliám spôsobené škodlivými aktivitami. Škodlivé činnosti v sieti môžu mať rôzne typy, ako sú bodové anomálie, kontextové anomálie či kolektívne anomálie.

Nasledujúci zoznam stručne opisuje predchádzajúce tri typy škodlivých činností:

- „**Bodová anomália:** Bodové anomálie sú prípady, ktoré sú mimoriadne alebo nezvyčajné vzhľadom na ostatné údaje. Napríklad mimoriadne výdavky na výpise kreditnej karty v

porovnaní s predchádzajúcimi transakciami. Na obrázku č. 4 môžete vidieť, že objekt *O1* je izolovaný od inej skupiny objektov *C1*, *C2* a *C3*. Objekt *O1* je bodová anomália.

- **Kontextová anomália:** V danom kontexte (napr. v rámci daného rozsahu), ak je inštancia anomálna alebo výnimočná. Na obrázku č. 4 môžete vidieť, že objekt *O2* je izolovaný v kontexte skupiny objektov *C2*. Objekt *O2* je kontextová anomália.
- **Kolektívna anomália:** Ak sa zistí s ohľadom na dané normálne správanie, že skupina prípadov sa odchýli anomálne, celá skupina anomálnych prípadov sa označí ako kolektívna anomália. Na obrázku č. 4 je skupina *C3* odlišná od skupín *C1* a *C2* z hľadiska počtu prípadov a kompaktnosti, a preto môže byť *C3* označená ako kolektívna anomália.“ (Dhruba Kumar Bhattacharyya a Jugal Kumar Kalita [6], 2014, str. 46)



Obrázok 4 – Bodová, kontextová a kolektívna anomália [6]

Podľa [6], účel anomálne založených detekčných systémov je analýza, porozumenie, charakteristika sieťovej premávky, a zároveň identifikácia a klasifikácia abnormálnej premávky. Anomálne klasifikačné metódy existujú v štyroch kategóriách:

- **Dohliadaná anomálna detekcia** – Technika trénovania potrebuje trénovaciu dátovú množinu, ktorá má označenú normálnu a anomálnu sieťovú premávku. V prípade dát, ktoré sa nepodarilo zaklasifikovať (označiť) sú porovnané voči modelu už klasifikovaných tried a na základe výsledku sú zaradené do príslušnej triedy. Nevýhodou tejto techniky je, že anomálne dáta sú málo porovnávané voči normálnym triedam. Ďalej, je náročné presne a reprezentatívne klasifikovať anomálnu sieťovú premávku.
- **Čiastočne dohliadaná anomálna detekcia** – Technika, pri ktorej trénovacia dátová množina má označenú iba normálnu sieťovú premávku. Anomálna triedy nemajú označenie a tak využitie tejto techniky je viac aplikovateľná.

- **Nedohliadaná anomálna detekcia** – Technika nepoužíva trénovaciu dátovú množinu a tým pádom je dosiahnutie presnej detekcie anomálnej sieťovej premávky náročné. Väčšinou pri tejto technike sa vychádza z predpokladu, že normálna sieťová premávka je vo väčšej miere zastúpená v testovacej dátovej množine ako anomálna.
- **Hybridná anomálna detekcia** – Technika kombinujúca dohliadanú a nedohliadanú metód detekcie. Táto technika má výhodu v tom, že vďaka dohliadanej technike má vysokú mieru odhalenia útokov a nízku mieru falošných poplachov. Na druhej strane, v prípade nedohliadanej technike má výhodu v odhalení neznámych útokov. Preto hybridná technika detekcie sieťových útokov je schopná identifikovať známe, ale aj neznáme sieťové útoky.

Podľa Samrina a Vasumathi [23] anomálne založené detekčné systémy majú nasledovné výhody:

- Na identifikáciu nových útokov sa nevyžaduje aktualizácia databázy.
- Po nainštalovaní softvéru je potrebná údržba.
- Súbežne sleduje správanie sa siete a vytvára profily sieťových aktivít.
- Najefektívnejšie identifikujte hrozby vo väčšom systéme.

a nevýhody:

- Abnormálne správanie sa siete v normálnej premávke neodošle upozornenie správcovi.
- Veľa falošných poplachov.

Ďalšími stratégiami odhalenia sieťových útokov sú detekčné systémy založené na podpisoch (*Signature-based IDS*) podľa Warzyński a Kołaczek [33] a systémy na rozpoznanie zneužitia (*Misuse IDS*) podľa Saxena a spol. [25]. Detekčné systémy založené na podpisoch sú založené na podpisoch známych útokov a pravidiel definovaných administrátorom. Takéto systémy môžu klasifikovať známe útoky porovnaním pozorovanej aktivity s uloženými vzormi, ale nemôžu identifikovať nové útoky. V prípade systémov na rozpoznanie zneužitia je najprv definované abnormálne správanie systému a potom všetky ostatné správania sú definované ako normálne. Je v rozpore s prístupom na detekciu anomálií, ktorý využíva opačný postup. Pri detekcii zneužitia je všetko neznáme, normálne správanie sa.

2.1.2. Charakteristiky IDS

Spôsob reakcie IDS systémov na sieťové útoky môže byť dvojaká podľa Ahmed a spol. [1]: *pasívna* a *aktívna* reakcia. V prípade pasívnych detekčných systémov, systém pri odhalení sieťového

útoku priamo nereaguje na útok, ale nechá rozhodnutie na iný systém. V prípade aktívneho detekčného systému, systém pri odhalení útoku priamo reaguje na útok napríklad zablokovaním sieťovej premávky pre detegovaného sieťového útočníka.

Jedným z najdôležitejších cieľov NIDS je odhalenie útočnej premávky v reálnom čase. V prípade aktívnych IDS je táto skutočnosť možná, pretože priamo reagujú v momente detekcie útoku. Žiaľ takéto systémy trpia vysokou mierou falošných poplachov, z čoho vyplývajú nasledovné nedostatky IDS [23]:

- **False positive**: Predpoveď falošných útokov. Ak je táto miera vysoká, potom normálny útok sa predpovedá ako útok.
- **False negative**: Vysoká miera falošne negatívnych hodnôt spôsobuje problém tak, že keď sa vyskytne narušenie siete, IDS nevytvorí žiadne upozornenie.
- **True positive**: Vyskytne sa vtedy, keď dôjde ku skutočnému útoku a IDS naň odpovie vyvolaním poplachu.
- **True negative**: Keď nenastane žiadny útok a IDS nevyvolá poplach.

Poplach [6] – Výstraha generovaná systémom NIDS, ktorá dostatočne a zmysluplne identifikuje dôvod, ktorý spôsobil udalosť poplachu a zdroj/cieľ útoku. Mal by pomáhať správcovi systému alebo analytikovi pri určovaní vhodnej reakcie na konkrétne upozornenie.

Bhattacharyya a Kalita [6] vo svojej knihe zadefinoval nasledujúce otvorené výzvy čakajúce na vyriešenie:

- **Obmedzenie výkonnosti pri práci v reálnom čase** – Vyvinutý NIDS by mal v ideálnom prípade v reálnom čase zachytiť a skontrolovať každý jeden paket podľa aktuálneho sieťového scenára pre vhodnú analýzu a presnú detekciu anomálie.
- **Zníženie falošného poplachu** – NIDS alebo iná metóda detekcie by sa mala vyhnúť vysokej miere hlásenia falošných útokov.
- **Redukcia dimenzie** – Vyvinúť vhodnú metódu na výber optimálneho súboru parametrov na detekciu anomálií bez znižovania výkonnosti detekcie.
- **Zvýšenie výpočtovej sily** – Na zvládnutie sofistikovanejších a komplexnejších vrstvených útokov sú mechanizmy na ochranu siete postavené na existujúcom systéme s pridruženými výpočtovými modulmi, ktoré v prípade rapídneho vývoja vysokorýchlostného internetu sú kontraproduktívne a môžu spôsobovať uviaznutia či zníženie výpočtovej sily.
- **Generický systém** – Platformovo nezávislý systém či metóda.

- **Spracovanie sofistikovaných anomálií** – Aktualizovanie NIDS alebo iných detekčných metód na aktuálne anomálie, ktoré sa vyskytli v lokálnej sieti alebo na internete.

Garg a Maheshwari [9] ďalej uvádzajú:

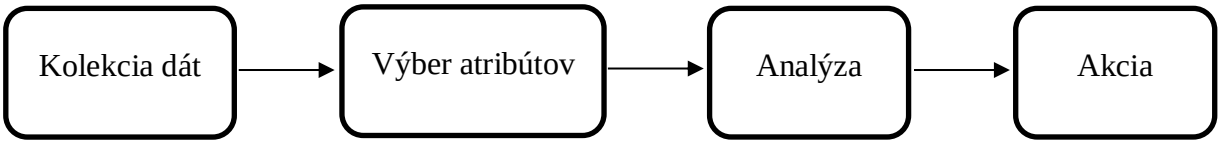
- **Špecifickosť**: Po identifikácii útoku musia byť k dispozícii dostatočné podrobné informácie, aby sa dosiahla lepšia reakcia.
- **Škálovateľnosť**: Možnosť aplikovať na veľké a malé siete.
- **A priori informácie**: Táto vlastnosť potrebuje vopred informácie týkajúce sa potenciálnych útočníkov a ich stratégií.

Bhattacharyya a Kalita ďalej spomínajú otvorené výzvy v súvislosti so spracovaním útokov ako: prispôsobovanie sa novým útokom, detekcia a spracovanie veľkých útokov, odhalenie útokov útočiace na sieťovú infraštruktúru, ovládanie, resp. odolávanie útokom s vysokou mierou frekvencie a identifikovanie nových, vynaliezavých útokov.

Autori Bhattacharyya, Kalita a Liu a spol. [17] sa však zhodnú na rovnakom názore/závere, že nedostatok spoľahlivých a kvalitatívnych údajov je veľkou prekážkou v presnosti klasifikácie útoku. Väčšina dátových množín je neúplná, keďže množstvo systémov založených na anomáliách bolo testované na množine vytvorených údajov a tým pádom hodnotenie systému je obmedzené kvalitou údajov. Taktiež sa znižuje efektívnosť zhromažďovania údajov súvisiacich s bezpečnosťou v dôsledku duplicity údajov a veľkosti ich množstva. Dátovým množinám sa venujeme v kapitole 2.7 Dátové množiny.

2.2. Architektúra IDS

Tiwari a spol. [32] vo svojom článku opisujú štyri základné moduly, z ktorých sa skladá vnútorná štruktúra každého IDS. Tieto moduly sú nasledovné: kolekcia dát, výber vhodných atribútov, analýza, akcia (viď. obrázok č. 5).



Obrázok 5 – Funkcionality IDS [32]

1. **Kolekcia dát** – Modul pre vstupné dáta pre IDS. Dáta sú zbierané a ukladané do súboru pre ďalšiu analýzu.
2. **Výber atribútov** – Modul pre výber vhodných ohodnotených atribútov z dátovej množiny.

3. **Analýza** – Modul, ktorý pre pravidlami riadené IDS systémy analyzuje dáta a v prípade prichádzajúceho sieťového toku dát sú dáta porovnávané voči vzorom a podpisom. V prípade anomálnych IDS systémov je analyzované správanie sa siete a následne je aplikovaný matematický model.
4. **Akcia** – Modul definujúci reakciu na útok. Reakcia môže byť informatívna pre systémového/sieťového administrátora alebo aj samotný IDS systém (aktívny) môže rozhodnúť o akcii (napríklad zahodenie paketov).

[32] uvádzajú tri základné komponenty IDS: senzor, backend, frontend. Túto skutočnosť potvrdzuje aj práca autora Schaelicke a spol. [26], ktorí tvrdia, že senzor je častokrát implementovaný ako univerzálny počítačový systém so softvérom na detekciu narušenia siete. Samostatný systém môže byť hostiteľom databázy alebo podobného softvéru na zabezpečenie dlhodobého ukladania dát pre ďalšiu možnosť analýzy.

1. **Senzor** – Hlavnou funkciou tohto modulu je detekcia a report. Obsahuje rozhranie pre kolekciu dát a rozhranie pre sieťový manažment. Priebeh funkcionality senzora je nasledovný: senzor počúva na sieťovom rozhraní a ukladá zachytené dáta do buffer pamäte. Následne nástroj na detekciu analyzuje pozbierané dáta a spustí sa analýza sieťových protokolov. V tomto komponente prebieha aj detekcia podpisov a anomálií.
2. **Backend** – Predstavuje hlavnú funkcionalitu IDS. Každá senzorom zachytená udalosť je ukladaná do databázy a následne vyhodnocovaná, že aká akcia sa podnikne systémom.
3. **Frontend** – Modul predstavuje používateľské rozhranie a tzv. *Command & Controll*. Údaje, udalosti a log zápisy z backend-u sa zobrazia v tomto komponente, kde používateľ ich môže manažovať.

2.3. Existujúce nástroje

Bhattacharyya a Kalita [6] vo svojej knihe opísali veľké množstvo (20) nástrojov na detekciu útokov alebo narušenia, ktoré identifikujú známe aj neznáme útoky pomocou štatistických metód, dolovania v dátach alebo softvérových prístupov. V nasledujúcom zozname opíšeme niektoré vybrané nástroje.

1. **Bro** – Voľne dostupný nástroj pre platformu Linux, ktorý pasívne monitoruje sieťovú prevádzku a snaží sa identifikovať narušenia siete v reálnom čase. Je schopný identifikovať útoky založené na podpisoch, útoky orientované na udalosti a niektoré

nezvyčajné útoky. Umožňuje tiež sledovanie správania, viacvrstvovú analýzu, presadzovanie politík a činnosti pri registrácii paketov.

2. **Snort** – IDS je založený na ľahkých podpisoch, ktorý kontroluje návštevnosť protokolu TCP/IP s cieľom identifikovať narušenia siete na základe pravidiel funkcií a zhody obsahu.
3. **HIDE** – Hierarchický systém založený na anomáliách, vyvinutý pomocou štatistického modelovania a neurónových sietí. Skladá sa z niekoľkých úrovní, kde každá vrstva obsahuje niekoľko detektorov narušenia (*Intrusion Detection Agents* - IDA), ktoré sú IDS komponentmi, ktoré monitorujú činnosti hostiteľa alebo siete.
4. **CAD** – Používa dátovú štruktúru - stromy (*Change Aggregation Trees* - CAT) na detekciu distribuovaných záplavových útokov (DDoS) na úrovni toku. Hlavným cieľom je odhaliť náhle zmeny prevádzky vo viacerých sieťových doménach.
5. **MINDS** – Populárny nástroj založený na metódach dolovania v dátach. Využíva Netflow v. 5 na zbiera dáta zo sieťovej premávky pomocou nástrojov na odchytávanie toku v sieti. Pred vyhodnotením anomálnej premávky sa tieto dáta prečistia od nezaujímavých dát, resp. vzorov sieťovej premávky. Systém na detekciu anomálií využíva outlier algoritmus, ktorý priradí jednotlivým sieťovým spojeniam anomálnu hodnotu, na základe ktorej sa vyhodnocuje podozrivá sieťová premávka.
6. **N@G** – Hybridný IDS s podporou odhaľovania útokov na strane klienta (host-based) a siete (network-based). Analyzuje premávku v reálnom čase na základe štatistických techník na strane klienta. Zastrešuje kontrolu používateľského rozhrania a manažmentu dát. Podporuje ochranu Layerd Service Provider (LSP) Domain Name Server (DNS) a dokáže dynamicky aplikovať Access Control List (ACL) pre blokovanie sieťovej premávky, resp. útoku.

Po analýze existujúcich nástrojov na bezpečnosť sietí autori knihy [6] prišli k záveru, že:

- Väčšina existujúcich NIDS je závislý od viacerých vstupných parametrov používateľa a ich výkon je veľmi citlivý na tieto parametre.
- Takmer všetky NIDS na báze anomálií pracujú takmer v reálnom čase alebo offline. Navyše väčšina trpí veľkým počtom falošných poplachov.

Autori knihy [6] identifikovali nasledujúce nedostatky v existujúcich riešeniach vhodné na ďalší výskum:

- Vyvinúť detekčný systém útokov v reálnom čase pre útoky s nízkou a vysokou frekvenciou DDoS (Distributed Denial of Service) útokov bez ovplyvnenia používateľov alebo bežných služieb.
- Vyvinúť NIDS založený na detekcii útokov pomocou anomálií, ktorého schopnosť detegovať útoky závisí od minimálneho počtu užívateľských parametrov a je schopný zaobchádzať so známymi aj neznámymi útokmi v reálnom čase s minimálnym počtom falošných poplachov.

2.4. Spôsob vyhodnocovania IDS

Hodnotenie je dôležité pre pochopenie kvality použitého modelu alebo techniky. Na základe získaných hodnôt môžeme ladiť použitie parametrov v iteratívnom procese učenia sa, pre výber najprijateľnejšieho modelu alebo techniky z daného súboru modelov alebo techník. Preto existuje niekoľko kritérií na hodnotenie modelov a techník.

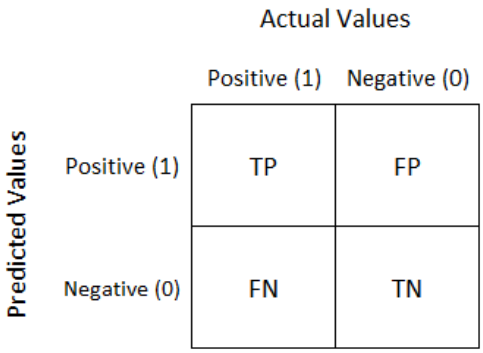
Podľa Schaelicke a spol. [26] výkon systému detekcie narušenia siete je charakterizovaný pravdepodobnosťou, že útok je detegovaný v kombinácii s počtom falošných upozornení. Rovnako dôležitá je však schopnosť systému spracovať prevádzku pri maximálnej rýchlosti, ktorú ponúka sieť s minimálnou stratou paketov. Významná strata paketov môže zanechať množstvo nezistených útokov a zhoršiť celkovú efektívnosť systému.

Autorka Bhardwaj [5] vo svojom diele uvádza, že pri klasifikačných problémoch je prirodzené merať výkon klasifikátora z hľadiska chybovosti. Klasifikátor predpovedá triedu každej inštancie ak je správne, počíta sa ako úspech, ak nie, tak ide o chybu. Miera chybovosti je len podiel chýb vykonaných v celom súbore inštancií a meria celkový výkon klasifikátora. Najznámejšie metódy vyhodnotenia výkonnosti klasifikátora sú nasledovné:

- **Cross-validation** – Rozdelenie dátovej množiny na menšie celky (*k* podmnožín) pre odhad rizika každého algoritmu. Časť údajov (trénovacia vzorka) sa používa na tréning každého algoritmu a zostávajúca časť (validačná vzorka) sa používa na analýzu. To znamená, že jedna podmnožina *k* sa použije na testovanie a ostatné *k-1* podmnožín na trénovanie. Tento postup sa opakuje pokiaľ sa nepoužije každá podmnožina *k* na test. Nakoniec sa výsledky testov skombinujú do výsledného odhadu a vyberie sa algoritmus s najmenším odhadovaným rizikom.
- **Holdout metóda** – Najjednoduchší druh krížovej validácie. Dátový súbor je rozdelený do dvoch skupín (trénovacia a validačná vzorka). Klasifikátor predpovedá výstupné hodnoty pre dáta v testovacej sade. Chybovosť klasifikátora sa spriemeruje v absolútnej hodnote a používa

sa na vyhodnotenie modelu. Hodnotenie môže do značnej miery závisieť od toho, ktoré dáta skončia v trénovacom súbore a ktoré skončia v testovacom súbore dát. Preto výsledok vyhodnotenia môže mať vysokú mieru variability.

- **Random sub-sampling** – Táto metóda vychádza z predchádzajúcej metódy, ktorá sa môže opakovať niekoľko krát kvôli zlepšeniu odhadu výkonnosti klasifikátora. Nevýhodou tejto metódy je, že nemá žiadnu kontrolu nad tým koľkokrát sa každý záznam použije na testovanie a tréning.
- **K-fold cross-validation** – Metóda na vylepšenie Holdout metódy. Dátový súbor je rozdelený do *k* podmnožín a metóda holdout sa opakuje *k*-krát. Zakaždým, keď sa jedna z *k* podmnožín použije ako sada testov, tak ostatné podmnožiny *k*-1 sa zostavia tak, aby vytvorili tréningovú množinu. Potom sa vypočíta priemerná chyba vo všetkých pokusoch *k*. Nevýhodou tejto metódy je, že tréningový algoritmus musí byť opakovaný *k*-krát.
- **Leave-one-out** – K-násobná krížová validácia, pričom *k* sa rovná *n*, kde *n* je počet inštancií v dátovej množine. Testuje sa vždy iba na jednom zázname a ostatné slúžia ako trénovacia vzorka. Týmto spôsobom sa zabezpečuje trénovanie na najväčšej možnej vzorke dát. Výpočet priemernej chybovosti sa použije na vyhodnotenie modelu.
- **Bootstrap** – Záznam, ktorý sa vybral na trénovanie sa vloží naspäť do pôvodného súboru záznamov, tým pádom môže byť s rovnakou pravdepodobnosťou znovu vybraný.
- **Confusion matrix** – Binárny klasifikačný model klasifikuje každú inštanciu do jednej z dvoch tried: správna a chybová trieda. Z toho vyplývajú štyri možné klasifikácie pre každý prípad: skutočný pozitívny (*true positive* - TP), skutočný negatívny (*true negative* - TN), falošný pozitívny (*false positive* - FP) alebo falošný negatívny (*false negative* - FN). Tieto štyri klasifikácie znázorňuje kontingenčná tabuľka, reps. Confusion matrix. (viď. obrázok č. 6).



Obrázok 6 – Confusion matrix [5]

Z matice môže odvodiť rad výkonnostných metrík modelu:

- **Presnosť** (precision) – je pomer správne predpovedaných pozitívnych pozorovaní k celkovým predpokladaným pozitívnym pozorovaniam a počíta sa nasledovne:

$$\text{Presnosť} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

- **Odvolanie** (precall) – je pomer správne predpovedaných pozitívnych pozorovaní ku všetkým pozorovaniam v skutočnej triede a počíta sa nasledovne:

$$\text{Odvolanie} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

- **F1-skóre** (F1-score) – je vážený priemer presnosti a spätného volania. Toto skóre berie do úvahy falošne pozitívy aj falošné negatívy a počíta sa nasledovne:

$$\text{F1} - \text{skóre} = \frac{2 * (\text{Odvolanie} * \text{Presnosť})}{\text{Odvolanie} + \text{Presnosť}}$$

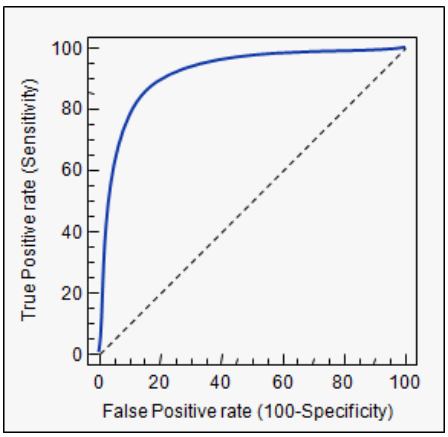
- **Správnosť** (accuracy) – predstavuje pomer správneho predpovedaného pozorovania k celkovým pozorovaniam a počíta sa nasledovne:

$$\text{Správnosť} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}}$$

Receiver Operating Curves (ROC) intuitívnejším a robustnejším spôsobom vizuálne sprostredkúva rovnaké informácie ako confusion matrix. ROC je dvojrozmerný graf, ktorý vizuálne zobrazuje výkonnostný a výkonový kompromis klasifikačného modelu. Pre skonštruovanie ROC krivky je potrebné zaviesť dve nové výkonnostné metriky: skutočná pozitívna miera (*True positive rate* - TPR), ktorej vzorec je rovnaký ako pre *odvolanie* a falošná pozitívna miera (*False positive rate* - FPR).

$$\text{Falošná pozitívna miera} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

Grafy ROC sú konštruované vykreslením skutočnej pozitívnej miery proti falošne pozitívnej miere (viď. obrázok č. 7).



Obrázok 7 – ROC krivka [27]

Autori Zaman a Lung [36] vo svojej práci vyhodnotili výkonnosť siedmich techník strojového učenia na dátovej množine Kyoto 2006+ a zistili, že ROC je vhodnejšia na vyhodnotenie techník ako metriky presnosť, odvolanie a správnosť.

Podľa Bhardwaj [5] na výber zaujímavých pravidiel zo súboru všetkých možných pravidiel je možné využiť obmedzenia rôznych meradiel významu a prínosu. Najznámejšie sú:

- **Podpora (support)** – Vyjadruje pravdepodobnosť, ako často sa v súbore údajov objavuje sada položiek. Je definovaný ako podiel transakcií *T* v súbore údajov, ktoré obsahujú množinu položiek *X*.
- **Spoľahlivosť (confidence)** – Vyjadruje podiel tých inštancií, ktoré vyhovujú pravidlu (X∪Y) oproti tým, na ktoré sa dá aplikovať pravidlo X.

$$\mathrm{conf}(X \Longrightarrow Y) = \frac{\mathrm{supp}(X \cup Y)}{\mathrm{supp}(X)}$$

Na meranie účinnosti IDS bolo navrhnutých niekoľko metrík podľa Kumar [14]. Tieto metriky môžu byť rozdelené do troch tried a to, prahová (*treshold*), hodnotiaca (*ranking*) a pravdepodobnostná (*probability*) trieda.

Prahové metriky zahŕňajú mieru klasifikácie (*Classification rate* - CR), F-meranie (*F-measure* - FM) a náklady na príklad (*Cost per example* - CPE). Nie je dôležité, ako je blízka predikcia k prahovej hodnote, dôležité je ak je nad alebo pod prahovou hodnotou. Hodnoty prahovej triedy ležia v rozsahu 0 až 1.

Trieda hodnotenia zahŕňa FPR, detekčnú frekvenciu (*Detection rate* - DR), presnosť (*Precision* - PR), oblasť pod krivkou ROC (AUC) a schopnosť detekcie narušenia (*Intrusion detection*

capability - CID). Tieto metriky merajú ako dobre sú inštancie útoku usporiadané pred normálnymi inštanciami a môžu byť interpretované ako súhrn výkonnosti modelu na všetkých možných prahoch.

Do triedy pravdepodobnosti patrí chyba strednej hodnoty (*Root mean square error* - RMSE). Hodnota RMSE leží v rozsahu 0 až 1. Trieda je minimalizovaná, keď sa predpovedaná hodnota pre každú triedu útoku zhoduje so skutočnou podmienenou pravdepodobnosťou, že táto trieda je normálna.

V prípade metód hodnotenia klastrovaním je vyhodnotenie oveľa náročnejšie. Jediný spôsob, ako realisticky vyhodnotiť klastrovanie je, že či sa výsledok zhlukovania v kontexte aplikácie ukáže ako užitočný. Klastrovanie možno vyhodnotiť z perspektívy dĺžky popisu pomocou princípu *Minimum Description Length* (MDL). Princíp MDL spočíva v tom, že najlepšia hypotéza pre daný súbor údajov je tá, ktorá vedie k najlepšej kompresii údajov. Technika klastrového učenia rozdeľuje tréningový súbor do *k* klastrov. Najlepšie klastrovanie podporí najefektívnejšie kódovanie.

2.5. Sieťové útoky

Podľa Liu a spol. [17] sieťový útok využíva diery v sieťovom systéme, v chybných protokoloch, v chybách hardvéru alebo softvéru a údaje sieťového systému pri neautorizovanom správaní. Sieťová komunikácia je v súčasnosti konfrontovaná so štyrmi druhmi bezpečnostných hrozieb: odpočúvanie informácií, prerušenie komunikácie, falšovanie správ a falšovania informácií. Ďalej sa tieto štyri typy útokov kategorizujú do dvoch kategórií a to, aktívne a pasívne útoky. Zachytávanie informácií je pasívny útok. Prerušenie komunikácie, manipulácia so správou, útok falšovania je aktívny útok. V súčasnosti predstavuje režim sieťového útoku viacero prostriedkov, voči čomu sa ľudia ťažko bránia. Tieto režimy by mohli byť rozdelené do štyroch kategórií: odmietnutie služby útoku, využitie typu útoku, zhromažďovanie informácií útok, falošné informácie útoku. Potom sa môžeme zamerať na niekoľko základných útočných prostriedkov, ako je napríklad: IP Spoofing Attack, ARP Attack, UDP Flooding Attack, TCP SYN Flooding Attack, ICMP Flooding Attack atď.

Sheela a spol. [28] vo svojej práci uvádzajú, že IDS využíva rôzne techniky na detekciu vniknutí do počítačovej siete. Používa na detekciu útočníkov jednoduchú techniku alebo kombináciu techník. Tieto techniky zahŕňajú detekciu anomálií, detekciu zneužitia, monitorovanie cieľa a špionáž.

2.5.1. Detekcia anomálií

Podľa [28] táto technika uchováva normálne správanie počítačovej siete, ako sú informácie o sieťových paketoch, informácie o softvéri, systémové log udalosti, informácie o operačnom

systéme, informácie o jadre operačného systému (kernel) atď. Ak sa deteguje rozdiel vo vyššie uvedených parametroch, tak sa zistí anomália a vygeneruje sa alarm. Detekcia anomálií je užitočná pri zisťovaní podvodov, vniknutí do siete a pri iných nezvyčajných činnostiach v systéme. Detekcia anomálie je označovaná aj ako detekcia založená na správaní sa. Identifikuje odchýlky systému od normálneho správania. Táto metóda má schopnosť odhaliť nové a neznáme útoky analýzou údajov o audite.

Žiaľ, táto metóda má vysokú mieru falošného poplachu. Niekedy môže byť legitímne správanie systému klasifikované ako anomálne a označené ako útok.

2.5.2. Detekcia zneužitia

Podľa [28] táto technika ukladá sekvenciu vzorov, útočné signály, vzory narušenia atď. do databázy. Uložené udalosti zo systému sa porovnávajú s uloženými informáciami v databáze zistených útokov. Ak sa zistí zhoda, systém vygeneruje alarm. Keďže táto metóda porovnáva podpisy, niekedy sa označuje ako detekcia založená na podpisoch. Tieto techniky automaticky aktualizujú svoju databázu na rôznych vstupných údajoch tak, aby zahŕňali nové typy útokov. Techniky detekcie zneužívania majú vysoký stupeň presnosti pri detekcii známych útokov a ich variantov. Tieto techniky však nedokážu zistiť neznáme/nové útoky, pretože sú závislé od existujúcich podpisov.

2.5.3. Monitorovanie cieľa

Podľa [28] táto technika hľadá modifikáciu na špecifických súboroch. Monitorovanie cieľa nehľadá anomálie. Funguje to ako korekčná kontrola, ktorá obnovuje súbor potom, čo bol súbor modifikovaný útočníkom. Využíva *cryptographic hash computing* na obnovenie upraveného obsahu. Táto technika je ľahko implementovateľná, pretože neustále sledovanie prevádzky správcom nie je potrebný. Posielanie alarmu do siete alebo do systému sa vykonáva vtedy, keď existuje nesúlad údajov (check sum). Výpočet check sum môže byť vypočítaný v rôznych intervaloch.

2.5.4. Špionáž

Podľa [28] táto technika deteguje útočníkov, ktorí zostávajú v sieti po dlhú dobu. Vo všeobecnosti útočníci dlhodobo kontrolujú zraniteľnosť systému a otvárajú porty a čakajú na ďalšiu dlhú dobu útoku. Táto technika kontroluje všetky takéto metodické útoky zhromažďovaním širokej škály údajov o celom systéme. Technika vyžaduje veľké množstvo vzoriek odobratých z rôznych počítačov a sietí na objavenie takýchto útokov. Na tento účel kombinuje detekciu anomálií a detekciu zneužitia.

2.5.5. Typy sieťových útokov

Autori Ananthi a Vengatesa [3] vo svojej práci vysvetľujú ako funguje jeden z najčastejších počítačových útokov, Denial of Service (DoS). V prípade DoS na zaplavenie servera paketmi (TCP/UDP) sa používa jeden počítač a jedno internetové pripojenie. Distributed DoS (DDoS) je botnet útok a je jedným z typov záplavového útoku. Počas tohto útoku sa útočné uzly pokúšajú naraz napadnúť jeden uzol (zvyčajne server). Namiesto jedného počítača a jedného internetového pripojenia využíva útok DDoS mnoho počítačov a mnoho pripojení. Počítače za takýmto útokom sú často distribuované po celom svete. Hlavným rozdielom medzi DoS a DDoS je, že cieľový server bude preťažený stovkami alebo dokonca tisíckami požiadaviek v prípade útoku DDoS. V dôsledku toho dochádza k pretečeniu pamäte a odmietnutiu poskytovanej služby (nemožno ďalej poskytovať danú službu). DoS ďalej vysvetľujú vo svojej práci autori Hussain a Mishra [11]. DoS je typ útoku, kedy útočník zahlcuje pamäťové prostriedky a tým pádom zabránia, aby slúžili legitímnym sieťovým požiadavkám. Takto napadnutý systém odopiera užívateľom prístup k počítaču alebo iným službám poskytované počítačom. DoS útok je iniciovaný tromi spôsobmi:

1. Zneužívanie legitímnych vlastností počítača.
2. Zacielenie implementačných chýb.
3. Využitie nesprávnej systémovej konfigurácie.

Útočník po úspešnom dokončení útoku ďalej poskytuje iné (svoje) služby, ktoré sú nedostupné pre autentické použitie a sú založené na rovnakých princípoch ako DoS útok, napríklad: *apache, smurf, neptune, ping of death, back, mail bomb, UDP storm* atď. DoS je jedným z útokov ktorý klasifikuje aj dátová množina NSL-KDD, ktorej sa venujeme v kapitole 2.7 Dátové množiny.

Ďalšie známe útoky podľa [11] sú:

- **Remote to Local** (R2L) – Týka sa neoprávneného prístupu od vzdialeného počítača. Útočník útočí na vzdialene umiestnený počítač odoslaním paketov cez internet. Útok využíva privilégiá, ktoré by mal mať lokálny používateľ na počítači. Príklady takýchto útokov sú: *xlock, xnsloop, phf, sendmail, dictionary* atď.
- **User to Root** (U2R) – Je spojený s neoprávneným prístupom lokálnych privilégií super používateľa (root). Pri týchto typoch útokov útočník začína v systéme s bežným používateľským kontom so snahou zneužiť zraniteľné miesta v systéme na získanie privilégií super užívateľov. Príklady: *perl, Xtream* atď.

- **Probe** – Útočník skenuje počítač alebo sieťové zariadenia na odhalenie zraniteľných alebo slabých miest, ktoré sa neskôr môžu zneužiť, aby sa narušil systém. Táto technika je primárne spojená s dolovaním v dátach ako: *satan*, *saint*, portsweep, *mscan*, *nmap* atď.
- **Scan** – Podľa Al-Jarrah a Arafat [2] skenovanie portov sa pokúša objaviť spustené služby na hostiteľskom počítači alebo sa pokúsi overiť dostupnosť určitej služby. Je dobre známe, že každá sieťová aplikácia bežiaca na počítači má jedinečné číslo portu, na ktoré počúva, ako napríklad port 80 pre prehliadanie webu. Zistením, ktoré služby sú spustené môže byť spustený určitý útok proti objavenej službe. Techniky skenovania portov sú rozdelené do troch typov na základe portov a hostiteľov:

    1. **Jeden hostiteľ – rôzne porty**: Útočník prehľadáva rôzne porty na určitom hostiteľovi, čo je typické správanie skenovania portov. Poradie portov nie je dôležité, skenovanie môže byť sekvenčné alebo náhodné.
    2. **Rôzny hostitelia – jeden port**: Útočník prehľadáva viacero hostiteľov naraz s rovnakým číslom portu. Tento útok je spustený proti sieti hostiteľov, ktorí hľadajú hostiteľov, ktorí prevádzkujú určitú službu ako DNS, SMTP alebo HTTP.
    3. **Rôzny hostitelia – rozdielne porty**: Útočník prehľadáva viacero hostiteľov naraz a každý hostiteľ je skenovaný iným portom. Toto je pokročilá technika skenovania portov, ktorá sa snaží skryť svoju aktivitu iniciovaním náhodných skenov portov medzi náhodnými hostiteľmi. Tento útok je najkomplikovanejším skenovaním portov.

Najznámejšie útoky na skenovanie portov sú: TCP scanning, ACK scanning, UDP scanning, FIN scanning, NULL scan, X-mas a UDP/ICMP Error a ďalšie.

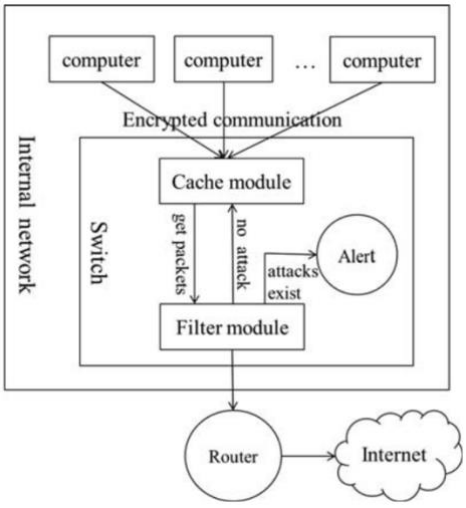
- **Eavesdropping** [28] – Útočník monitoruje komunikáciu iných ľudí neautorizovaným spôsobom. To môže byť napríklad odpočúvanie telefónnych hovorov, prezeranie e-mailov a správ a ďalších internetových služieb. Odposluch je ťažké zistiť, pretože nemá vplyv na normálnu prevádzku siete. Keďže je ťažké odhaliť tento typ útoku, tak je vo všeobecnosti najväčším problémom, ktorému väčšina správcov čelí v podniku. Použitím silných šifrovacích schém môžu byť dáta chránené pred týmto typom útoku.
- **Man-in-the-Middle** [28] – Pri tomto útoku útočník zachytáva konverzáciu medzi dvomi stranami a vydáva sa za nich, čím získa prístup k dôležitým informáciám. Obe strany sa domnievajú, že spolu priamo komunikujú, aj keď útočník sa nachádza v strede ich konverzácie. Tento typ útoku je najčastejšou hrozbou pre online bezpečnosť, pretože

útočníkovi je umožnené zachytiť a upraviť citlivé informácie v transakciách v reálnom čase. Útočník môže tiež zneužiť chyby zabezpečenia v konfiguráciách zabezpečenia siete.

2.5.6. Odhalenie sieťových útokov

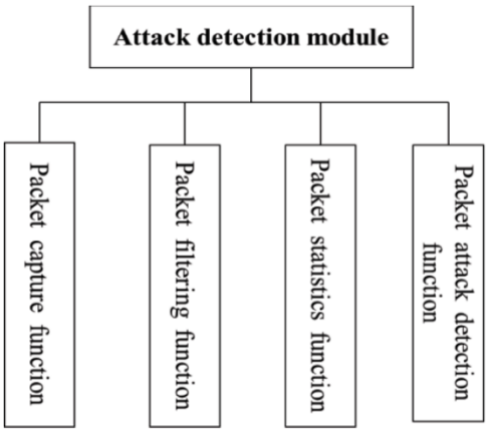
Sieťový útok odkazuje na zneužitie chýb a nedostatkov v bezpečnostných nastaveniach siete, chybných protokolov, chybu v hardvéri alebo softvéri, násilný útok na hardvér alebo softvér či na údaje pri neautorizovanom správaní sa v sieťovom prostredí. V súčasnosti predstavuje sieťový útok viacero prostriedkov, voči čomu sa ľudia len ťažko bránia. Preto je potrebné vyvinúť obranný mechanizmus na ochranu sieťovej premávky voči škodlivej činnosti, tak ako aj autori článku Li a spol. [16].

Li a spol. navrhli sieťový modul detekcie útokov. Modul je založený predovšetkým na filtrovaní paketov. Rozpoznáva typ sieťového útoku a má aj funkcionalitu monitorovania siete, či funkciu upozornenia, ktorá nemá vplyv na výkon základnej sieťovej komunikácie. Na nasledujúcom obrázku č. 8 môžete vidieť návrh platformy na odhalenie sieťových útokov.



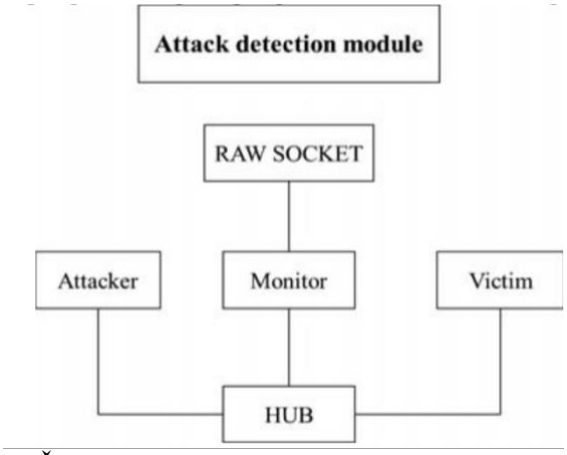
Obrázok 8 – Návrh platformy na odhalenie sieťových útokov [16]

Modul najskôr prijíma pakety na detekciu útoku, ktoré sú následne filtrované vo filtračnom module (Filter module). Potom podľa charakteristík útoku siete, sa analyzujú pakety. Vývojové prostredie sa skladá z troch počítačov (útočník, obeť a detektor), ktoré sú prepojené HUB-om, ako je znázornené na obrázku č. 9. Na detektore beží modul detekcie útoku. Tento spôsob spojenia je v súlade so skutočnou aplikáciou detekčného stroja, ktorý nebráni počítačom odosielať a prijímať dátové pakety. Pod inštaláciou operačného systému Linux je zabezpečená podpora raw socketov na zachytenie všetkých ethernet rámcov.



Obrázok 9 – Prostredie odhalenia útoku [16]

Na obrázku č. 10 je znázornená konštrukcia modulu na detekciu sieťových útokov, ktorá slúži predovšetkým na zachytenie dátových paketov, ich filtrovanie, robenie dátovej štatistiky a na odhalenie útoku.



Obrázok 10 – Štruktúra modulu na detekciu sieťových útokov [16]

2.6. Strojové učenie

Strojové učenie môže byť klasifikované mnohými rôznymi spôsobmi, ale najčastejšie sa rozlišuje strojové učenie pod dohľadom a bez dohľadu. Strojové učenie pod dohľadom je potrebné naučiť na sade príkladov pre danú problematiku, pričom strojové učenie bez dohľadu dokáže sa sám naučiť vzory z poskytnutých údajov bez akéhokoľvek ľudského zásahu alebo poradenstva.

Bhattacharyya a Kalita [6] vysvetľujú, že strojové učenie s dohľadom sú metódy prediktívneho modelu pre normálové a anomálne triedy, kedy model porovnáva dátové inštancie s modelom pre určenie do ktorej triedy patrí. Na rozdiel strojové učenie bez dohľadu predpokladá, že normálové prípady sú oveľa častejšie ako anomálne prípady a anomálne prípady sú štatisticky odlišné od bežných prípadov. Avšak ak tieto predpoklady nie sú pravdivé, tak takéto metódy vyhodnocovania trpia vysokými falošnými poplachmi.

Strojové učenie s dohľadom:

- Rozhodovacie a regresné stromy
- Klasifikačné a regresné stromy (CART)
- Podporné vektorové stroje (SVM)

Strojové učenie bez dohľadu:

- Algoritmy hierarchického zhlukovania
- Algoritmy zhlukovania založené na hustote
- Modulárne algoritmy
- Softvérové algoritmy založené na soft computingu

V prípade strojového učenia bez dohľadu existuje mnoho ďalších typov, ktoré sme vyššie nezahrnuli napríklad: Skryté Markov modely, Algoritmus maximalizácie očakávaní, Soft computing, Rough sets, Evolučný algoritmus a ďalšie.

[6] NIDS systém vyžaduje ľudský zásah vo vytvorení, overení a nasadení pravidiel. Na riešenie tohto problému boli zavedené NIDS s anomáliami, ktoré používajú algoritmy strojového učenia. Problém falošných poplachov však stále postihuje väčšinu takýchto systémov. Preto vhodná kombinácia metód strojového učenia pre túto problémovú oblasť môže viesť k lepšiemu rozvoju NIDS.

Na využití umelej inteligencie (UI) – neurónových sietí v prostredí počítačových sietí, konkrétne pre IDS sa zhodnú aj autori článku [15], Li a Dong. Tvrdia, že UI má mnoho užitočných vlastností ako: paralelizmus, vysoký stupeň tolerancie voči chybám, seba-reguláciu, seba-poznávanie a silné nelineárne mapovanie a zovšeobecnenie pre komplexný systém.

2.6.1. Klasifikačné algoritmy

Na detekciu anomálií sa vo všeobecnosti používajú algoritmy strojového učenia. Tieto algoritmy vytvárajú model detekcie alebo predikčný model vo fáze učenia sa pomocou tréningového algoritmu na tréningových dátach. Tento predikčný model sa potom testuje na nových údajoch v testovacej fáze. Vstupné dáta potrebujú predspracovanie, aby boli zrozumiteľné pre algoritmy strojového učenia. Vstupné údaje predstavujú pozorovania alebo záznamy, a každý záznam je reprezentovaný atribútom.

Algoritmy, ktoré vyžadujú plne označené údaje, sa nazývajú dohliadané algoritmy učenia. Algoritmy, ktoré nepotrebujú označené údaje sa nazývajú algoritmy učenia bez dozoru. Tieto algoritmy nájdu skrytý vzor v údajoch, keďže dokážu nájsť vzťah medzi údajmi a ich triedou.

Poslednou klasifikáciou algoritmov strojového učenia sú čiastočne dohliadané algoritmy, ktoré nepotrebujú všetky záznamy, aby boli označené.

Nasledujúci zoznam podľa Jain a Bhupendra [12] uvádza niektoré klasifikačné algoritmy na odhalenie sieťových útokov:

- **K-means** – Na vyriešenie hlavného problému klastrovania je algoritmus K-means najbežnejší a najjednoduchší algoritmus učenia sa bez dozoru. Rozdeľuje *n* pozorovaní do *k* klastrov, každé pozorovanie patrí do klastra s najbližším priemerom a slúži ako prototyp klastra. Výsledkom je rozdelenie dátového priestoru do Voronoiho buniek.
- **Bayesovské siete** – Používajú sa na vyjadrenie poznatkov o nejednoznačnej doméne súboru údajov. Pravdepodobnostné závislosti medzi zodpovedajúcimi náhodnými premennými sú reprezentované hranami tohto modelu. Uzly reprezentujú premenné a hrany kódujú podmienené závislosti medzi premennými. Bayesovské siete sú riadené acyklické grafy (*Directed Acyclic Graphs* - DAG). Stavy náhodnej premennej a tabuľka podmienenej pravdepodobnosti (*Conditional Probability Table* - CPT) sa nachádza v každom uzle.
- **J48** – Voľne dostupný klasifikátor algoritmu C4.5. C4.5 je program, ktorý vytvára rozhodovací strom založený na súbore označených vstupných údajov. Rozhodovacie stromy sa môžu použiť na klasifikáciu, a preto sa C4.5 často označuje ako štatistický klasifikátor. Podľa Mehmood a Rais [18], rozhodovací strom závisí od atribútov v tréningových dátach, pretože z nich pracuje na základe získaných informácií. Koreňový uzol (root) obsahuje funkciu, ktorá má najviac informácií. Algoritmus J48 je navrhnutý s tými funkciami, ktoré ľahko riešia medzery, ktoré sú prítomné v ID3.
- **ID3** – Dohliadaný algoritmus, ktorý využíva rozhodovací strom založený na matematických výpočtoch. Vykonáva zhora nadol greedy vyhľadávanie prostredníctvom danej tréningovej množiny na testovanie každého atribútu v každom uzle, na vytvorenie rozhodovacieho stromu.
- **NB Strom** – Vysoko škálovateľný hybridný prístup pre veľké databázy. Je vhodný pre prípady, keď je pre klasifikáciu mnoho relevantných atribútov. V takýchto prípadoch je databáza veľká a je žiaduca interpretovateľnosť klasifikátora , atribúty nie sú nevyhnutne nezávislé (t.j. atribúty nie sú podmienene nezávislé). NB strom výrazne zlepšuje výkonnosť svojich zložiek indukciou vysoko presných klasifikátorov.
- **Náhodný les** – Po anglicky Random Forest je jedným z algoritmov klasifikácie stromov. Hlavným cieľom tohto algoritmu je zvýšiť klasifikátory stromov na základe koncepcie lesa. Náhodné klasifikátory lesov majú akceptovanú mieru presnosti a môžu byť implementované na spracovanie hodnôt šumu súboru údajov. Počas klasifikácie nie je proces opätovnej

modifikácie. Pri implementácií tohto algoritmu by sa mal počet stromov v lese odhadnúť, pretože každý jednotlivý strom v rámci lesa predpovedá očakávaný výstup. Potom sa použije technika hlasovania, ktorá sa používa na výber očakávaného výstupu. Náhodný les je pomalý v trénovaní, je náchylný pretrénovaniu a taktiež je príliš jednoduchý pre komplexné problémy.

- **Rozhodovacie stromy** – Jednotkové stromové štruktúry, ktoré predstavujú rozhodovacie súbory. Tieto súbory generujú pravidlá, ktoré sa používajú na klasifikáciu údajov.
- **Support Vector Machine** – Nová generácia učiacich sa algoritmov. Používa sa na klasifikáciu a regresiu. SVM sú v popredí v oblasti strojového učenia vďaka dôsledným matematickým základom z optimalizácie a teórie štatistického učenia. Podľa [18] SVM oddeľuje triedy pomocou hyper plánu a používa údaje označené triedou v tréningovej fáze rovnako ako ostatné klasifikačné algoritmy učenia pod dohľadom. Aj keď SVM je binárny klasifikátor, môže byť použitý aj na klasifikáciu viacerých tried. Na klasifikáciu viacnásobných tried sa používajú dve rôzne metódy, *one-vs-all* a *one-vs-one*.
- **Multi Layer Perceptone** – Mapuje množinu vstupných dát na vhodnú množinu výstupov. Je graf pozostávajúci z viacerých vrstiev, kde každá vrstva je plne pripojená k ďalšej. MLP využíva spätné šírenie (generalizácia), kontrolovanú výučbovú techniku pre tréning siete. Môže klasifikovať údaje, ktoré nie sú lineárne oddeliteľné.

Pri klasifikačných algoritmoch je dôležité brať do úvahy vhodný výber parametrov. Na nájdenie optimálnych hyperparametrov modelu sa používa *Grid-search*. Optimálny výber hyperparametrov zaručuje presnejšie predpovede modelu. Hyperparameter modelu je vlastnosť, ktorá je mimo modelu a ktorého hodnota sa nedá odhadnúť z údajov. Hodnota hyperparametru sa musí nastaviť pred začiatkom procesu učenia.

Podľa Williamsa a spol. [34] na výber vhodných parametrov sa používajú algoritmy na báze konzistencie (*Consistency-based Feature selection* - CON) a na báze korelácie (*Correlation-based Feature Selection* - CFS). Tieto algoritmy hodnotia rôzne kombinácie funkcií na identifikáciu optimálnej podmnožiny.

- **Na báze konzistencie** – Vyhodnocuje podmnožiny parametrov a vyberie optimálnu podmnožinu. Optimálna podmnožina je najmenšia podmnožina parametrov, ktoré môžu identifikovať inštancie triedy rovnako konzistentne ako kompletná množina.
- **Na báze korelácie** – Používa hodnotiacu heuristiku, ktorá skúma užitočnosť jednotlivých parametrov spolu s úrovňou vzájomnej korelácie medzi vlastnosťami. Vysoké skóre je priradené k podmnožinám obsahujúcim atribúty, ktoré sú vysoko korelované s triedou a majú nízku vzájomnú koreláciu.

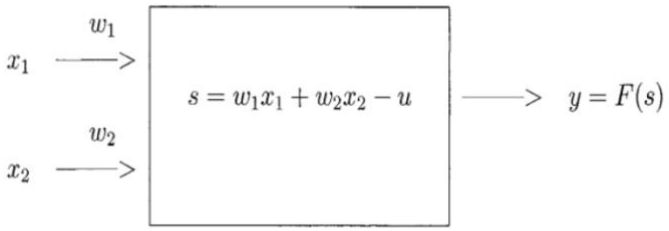
Podmnožiny parametrov podľa [34], ktoré sa majú vyhodnotiť sa generujú použitím nasledovných techník vyhľadávania podmnožín:

- **Greedy search** – Pre danú rodičovskú množinu skúška všetky možné potomkové podmnožiny prostredníctvom pridania alebo odstránenia parametrov. Potomková podmnožina, ktorá ukazuje najvyššiu mieru úspechu potom nahrádza nadradenú rodičovskú podmnožinu a proces sa opakuje. Proces sa ukončí, keď už nie je možné vykonať ďalšie zlepšenia.
- **Best First search** – Je podobná predchádzajúcemu algoritmu, ale na rozdiel od Greedy, Best First search vytvára nové podmnožiny založené na pridaní alebo odstránení parametrov aktuálnej podmnožiny. Má však schopnosť spätne sa pohybovať pozdĺž cesty výberu podmnožiny, aby sa preskúmali rôzne možnosti, keď aktuálna cesta už nevykazuje zlepšenie. Aby sa predišlo spätnému šíreniu, obmedzuje sa počet nevylepšených podmnožín.

2.6.2. Neurónová sieť

Haddadi a spol. [10] tvrdia, že neurónová sieť je inšpirovaná ľudským nervovým systémom a používa sa v rôznych oblastiach, ako je rozpoznávanie vzorov, optimalizácia, riadenie a pod. Dokáže riešiť komplexné nelineárne problémy a výsledky hodnotenia sú presné na základe veľkého počtu parametrov, ktoré zvyšujú predvídateľnosť.

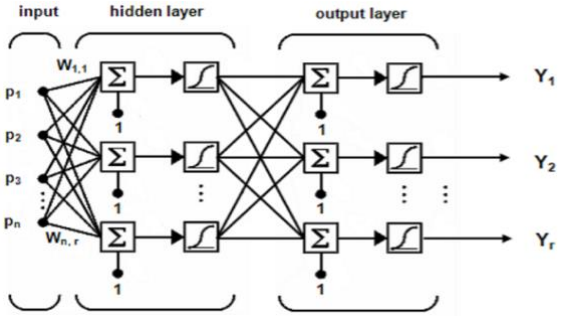
Neurónová sieť sa skladá z niekoľkých procesných jednotiek (uzlov) a riadených väzieb medzi nimi. Tieto spojenia reprezentujú vzťah medzi vstupnými a výstupnými neurónmi. Podľa Sani a spol. [24] výsledok transformácie je určený charakteristikami uzlov a váhami určenými pre prepojeniami medzi nimi. Teda nastavením váh môže byť výstup regulovaný. Proces aktualizácie váh a prahov sa nazýva učenie. Na obrázku č. 11 môžete vidieť jednoduchý neurón, kde parametre *x1* a *x2* predstavujú vstup, *w1* a w2 váhy, *u* - prah, *s* - súhrnný blok, *F* - aktivačnú funkciu a *y* - výstup. V prípade ANN je potrebné si uvedomiť, že predtým ako začneme proces trénovania je potrebné zadefinovať nastavenie neurónovej siete. Je potrebné určiť počet neurónov, počet vrstiev neurónovej siete, algoritmus a prenosovú funkciu pre trénovanie.



Obrázok 11 – Neurón [24]

Neurónová sieť je klasifikovaná do dvoch tried/architektúr na základe spojení: Feed-forward a Recurrent siete.

- **Feed-forward** – Podľa [10] a [24] neuróny sú usporiadané tak, že vstupy do prvej vrstvy neurónov sú vstupmi do neurónovej siete. Výstup každého neurónu v prvej vrstve je vstupom do každého neurónu v druhej vrstve a tak sa to opakuje vo všetkých nasledujúcich vrstvách, až kým nedôjdeme k výstupnej vrstve, ktorej výstupy sú jediným výstupom neurónovej siete. Úlohou tréningového procesu je nájsť správne váhy pre neurónové spojenia, ktoré v kombinácii so vstupmi dosiahnu požadovaný výstup. Tento proces sa uskutočňuje algoritmom spätného šírenia. Na obrázku č. 12 môžete vidieť takúto neurónovú sieť.



Obrázok 12 – Feed-forward neurónová sieť [10]

- **Recurrent** – Sieť svoje výstupy vracia späť do vlastných vstupov.

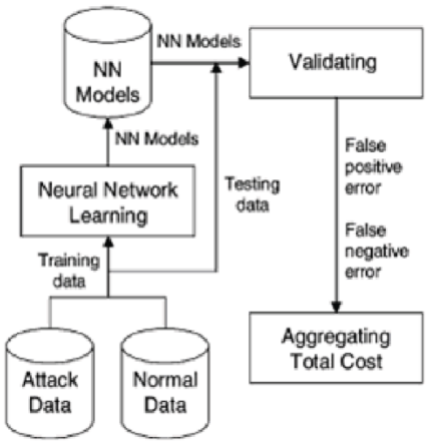
Podľa Haddadi a spol. [10] IDS založené na neurónových sieťach trpia nasledujúcimi problémami:

  - **Pretrénovanie** – Ak je neurónová sieť pretrénovaná časťou údajov. V takom prípade je len veľmi málo chýb v tréningu, ale vysoký podiel chýb v teste, pretože neurónová sieť stráca schopnosť generalizácie.
  - **Nedostatok pamäte** – Obrovské množstvo dát má za následok vysokú pamäťovú náročnosť. Neurónové siete trpia nízkou pamäťou vo fáze tréningu. Výber správnej tréningovej funkcie môže tento problém riešiť.
  - **Réžia** – V zložitých neurónových sieťach je veľa výpočtov a to spôsobuje režijné náklady. Réžia rastie zložitosťou systému.

Typický model neurónovej siete pre IDS je znázornený na obrázku č. 13. V tomto modely je predpoklad, že prichádzajúce pakety sú extrahované pomocou ľubovoľne dostupnej metódy. Pre tréningovú fázu, máme dátovú množinu pre útočnú a normálnu premávku. Je potrebné poznamenať,

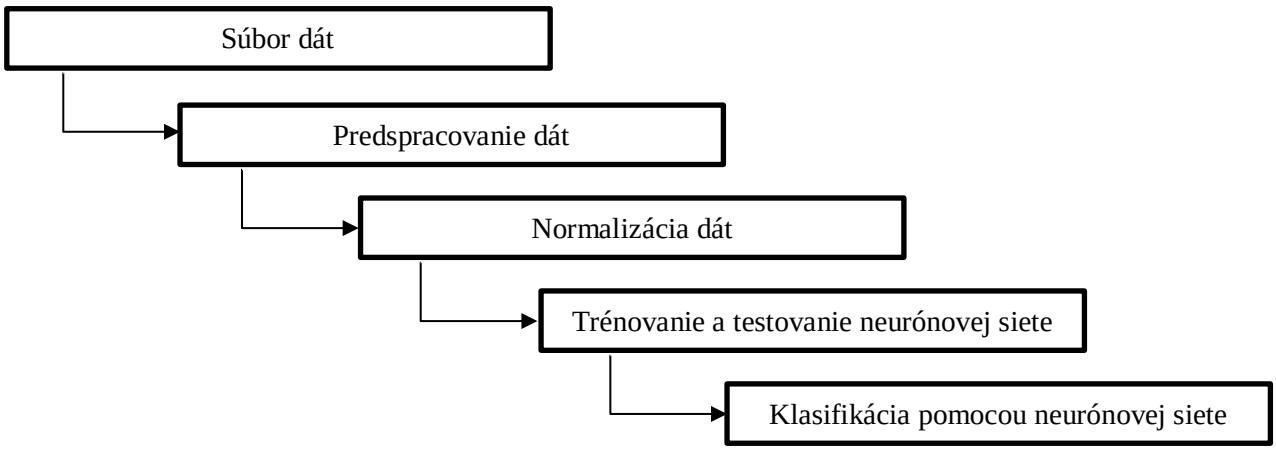
že dátové súbory a vzdelávací modul neurónovej siete sú pripojené k modelu neurónovej siete z dôvodu výberu vhodnej metódy na trénovanie. Pre každý tréningový cyklus je výstup porovnaný s očakávaným výstupom na validačnom module. Trénovanie prebieha nepretržite, kým model neurónovej siete nie je plne natrénovaný. V niektorých IDS trénovanie sa vykonáva pravidelne alebo nepretržite.

Pre vhodný návrh neurónovej siete je potrebné brať do úvahy predchádzajúce tri problémy (pretrénovanie, nedostatok pamäte, réžia), ktorými trpia neurónové siete.



Obrázok 13 – Model neurónovej siete pre IDS [24]

Na nasledujúcom obrázku č. 14 môžete vidieť proces spracovania dátovej množiny a klasifikácie sieťových útokov.



Obrázok 14 – Proces klasifikácie sieťových útokov [24]

2.7. Dátové množiny

Dátová množina je súbor údajov. Najčastejšie dátový súbor zodpovedá obsahu jednej databázovej tabuľky, kde každý stĺpec tabuľky predstavuje konkrétnu premennú a každý riadok zodpovedá prvku z príslušného súboru údajov.

2.7.1. NSL-KDD

Dátové množiny Kanadského inštitútu pre kybernetickú bezpečnosť [7] sú používané na celom svete univerzitami, súkromným priemyslom a nezávislými výskumníkmi.

Dátová množina NSL-KDD je vylepšenou verziou starej dátovej množiny KDD'99. Revathi a Malathi [22] tvrdia, že až 75-78% záznamov tvorilo duplicitu v starej verzii KDD. Vykonala sa štatistická analýzu tohto súboru údajov a zistili sa problémy, ktoré výrazne ovplyvňujú výkonnosť IDS a vedú k zlému hodnoteniu prístupov detekcie anomálií. Na vyriešenie týchto problémov sa navrhol nový súbor údajov - NSL-KDD, ktorý sa skladá len z vybraných záznamov z kompletného súboru údajov KDD.

Podľa inštitútu pre kybernetickú bezpečnosť v Kanade [7], NSL-KDD nezahŕňa nadbytočné duplicitné záznamy a tak výkon klasifikátorov nie je ovplyvnená metódami, ktoré majú lepšiu mieru detekcie na častých záznamoch. Inštitút ďalej uvádza, že počet vybraných záznamov z každej skupiny obtiažnosti je nepriamo úmerný percentu záznamov v pôvodnom súbore údajov KDD. V dôsledku toho, miery klasifikácie metód strojového učenia sa líšia v širšom rozsahu, čo znamená dosahovanie presnejších hodnotení pre rôzne metódy strojového učenia. Ďalej, počet záznamov v trénovacej a testovacej sade údajov sú primerané, čo umožňuje vykonávanie experimentov na kompletnom súbore dát, bez potreby náhodného výberu menšej časti. Výsledky hodnotenia experimentov tak budú konzistentné a porovnateľné.

Podľa [22] NSL-KDD sa skladá z 21 rôznych útokov z 37 prítomných v súbore testovacích dát. Známe typy útokov sú prítomné v súbore údajov pre trénovanie, zatiaľ čo nové útoky sú v súbore dát pre testovanie, t.j. nie sú dostupné v trénovacom súbore údajov. Typy útokov sú zoskupené do štyroch kategórií: *DoS, Probe, U2R* a *R2L*. Tabuľka č. 2 uvádza známe typy útokov.

Tabuľka 2 – Útoky v testovacom súbore dát NSL-KDD [22]

Trieda útoku	**Typ útoku**
DoS	Back, Land, Neptune, Pod, Smurf, Teardrop, Mailbomb, Processtable, Udpstorm, Apache2, Worm
Probe	Satan, IPsweep, Nmap, Portsweep, Mscan, Saint
U2R	Guess_password, Ftp_write, Imap, Phf, Multihop, Warezmaster, Xlock, Xsnoop, Snmpguess, Snmpgetattack, Httptunnel, Sendmail, Named
R2L	Buffer_overflow, Loadmodule, Rootkit, Perl, Sqlattack, Xterm, Ps

Dhanabal a Shantharajah [8] vo svojej práci detailne opisujú jednotlivé atribúty dátovej množiny NSL-KDD a taktiež opisuje experiment v automatizovanom nástroji WEKA na dolovanie v dátach. WEKA implementuje mnoho štandardných procesov dolovania v dát, ako je čistenie, predspracovanie, klastrovanie, klasifikácia, regresia, vizualizácia a výber funkcií dát. Je použitý na vykonávanie klasifikačných experimentov na 20 percentnom NSL-KDD dátovom súbore. Dátový súbor sa predspracuje a normalizuje na rozsahu 0 až 1. Normalizácia dát je dôležitá, pretože niektoré klasifikátory poskytujú lepšiu mieru presnosti na normalizovanom súbore dát. Dhanabal a Shantharajah vo svojej práci použili metódu výberu atribútov na základe korelácie. Dimenzia atribútov sa znížila z počtu 41 na 6. Následne sa vykonala klasifikácia pomocou algoritmov J48, SVM a Naïve Bayes. Výsledky sú zaznamenané v tabuľke č 3.

Tabuľka 3 – Tabuľka výsledkov experimentu v nástroji WEKA [8]

Klasifikačný algoritmus	**Názov triedy útoku**	**Presnosť (%)**
J48	Normal	99.8
	DoS	99.1
	Probe	98.9
	U2R	98.7
	R2L	97.7
SVM	Normal	98.8
	DoS	98.7
	Probe	91.4
	U2R	94.6
	R2L	92.5
Naïve Bayes	Normal	74.9
	DoS	75.2
	Probe	74.1
	U2R	72.3
	R2L	70.1

Podľa Thomasa a Pavithrana [31] aj napriek tomu, že súbor údajov NSL-KDD trpí niektorými problémami, je veľmi efektívny súbor údajov, ktorý možno použiť na výskumné účely. Dnes je ťažké získať reálne súbory údajov vzhľadom na povahu bezpečnostnej domény, a preto súbor údajov NSL-KDD sa považuje za jeden z najlepších pre výskum detekcie anomálií.

2.7.2. UNSW-NB15

Poľa oficiálnej stránky [30] UNSW-NB15 je neupravený dátový súbor sieťových paketov vytvorený nástrojom IXIA PerfectStorm v laboratóriu Cyber Range v Austrálskom centre pre kybernetickú bezpečnosť (*Cyber Range Lab of the Australian Centre for Cyber Security* - ACCS). Bol vytvorený na generovanie bežného hybridu skutočných moderných činností a syntetického súčasného správania pri útokoch.

Na zachytenie sieťovej prevádzky vo forme paketov sa používa nástroj *tcpdump*. Tento nástroj sa použil na zachytenie 100 GB surovej dátovej prevádzky. UNSW-NB15 súbor údajov má 9 typov útokov: *Fuzzers*, *Analysis*, *Backdoors*, *DoS*, *Exploits*, *Generic*, *Reconnaissance*, *Shellcode* a *Worms*. Dátový súbor pozostáva z 2 540 044 záznamov a 49 atribútov/vlastností s označením triedy.

Podľa Moustafa a Slay [21] nástroj IXIA obsahuje všetky informácie o nových útokoch, ktoré sa priebežne aktualizujú zo stánky CVE (*Common Vulnerabilities and Exposures*). CVE je stránka/slovník verejne známych zraniteľností a ohrození bezpečnosti informácií.

Moustafa a Slay [20] opisujú päť kategórií, do ktorých sú jednotlivé vlastnosti dátového súboru začlenené:

- **Vlastnosti toku** – Zahŕňa atribúty medzi hostiteľmi.
- **Základné** – Zahŕňa atribúty, ktoré predstavujú protokoly.
- **Obsahové** – Zahŕňa atribúty TCP/IP, obsahuje aj niektoré atribúty HTTP služieb.
- **Časové** – Obsahuje časové atribúty, napr. čas príchodu medzi paketmi, čas návratu protokolu TCP.
- **Ďalšie** – Túto kategóriu možno ďalej rozdeliť do dvoch skupín:
    - **Funkcie všeobecného účelu** – Každá vlastnosť má svoj vlastný účel, podľa ochrany služby protokolov.
    - **Funkcie pripojenia** – Sú vytvorené z toku 100 záznamov pripojenia na základe poradia poslednej časovej vlastnosti.

Moustafa a Slay [19] navrhli sieťovú forenznú schému, kde prvý krok zachytáva sieťové pakety cez sniffing a ukladá zachytené pakety do databázy, aby sa uľahčilo vyšetrovanie útokov. Druhý krok vyberá dôležité funkcie a odstraňuje cudzie a redundantné informácie, ktoré by mohli negatívne ovplyvniť odhalenie útokov. Dôležité funkcie/atribúty sa vyberajú využívajúc štatistiku - chí-kvadrát. Tretí krok skúma útoky a ich pôvod pomocou novej techniky correntropy-variation. Výsledky sú zaznamenané v nasledujúcej tabuľke č. 4 (výsledky sú uvedené v percentách).

Tabuľka 4 – Tabuľka výsledkov pre sieťovú forenznú schému [19]

Typ útoku	**Veľkosť vzorky**		
	100 000	**200 000**	**300 000**
Normal	92.12	93.16	93.29
Analysis	88.26	89.45	90.22
DoS	95.71	95.13	97.55
Exploits	76.47	77.82	77.19
Fuzzers	64.33	65.23	66.28
Generic	83.56	87.52	88.87
Reconnaissance	58.38	59.24	60.32
Backdoors	54.42	71.23	72.42
Shellcode	65.76	66.48	65.98
Worms	45.82	45.92	48.87

Navrhnutá technika sa porovnala s tromi najmodernejšími prístupmi, a to technikami: podporného vektorového stroja (*Filter-based Support Vector Machine* - FSVM), multivariačnou korelačnou analýzou (*Multivariate Correlation Analysis* - MCA) a umelou metódou techniky imunitného systému (*Artificial Immune System* - AIS). Na základe porovnaní, sieťová forenzná schéma je lepšia, pokiaľ ide o presnosť a frekvenciu falošného poplachu. Forenzná technika poskytuje najlepšie výsledky, pretože odhaduje hodnoty correntropie pre normálne a testované vzorky a následne identifikuje vzorky, ktoré sú viac ako dve štandardné odchýlky od priemeru normálnych vzoriek ako útoky.

2.7.3. ISCX

Podľa Kanadského inštitútu pre kybernetickú bezpečnosť [7] sa v ISCX zavádza systematický prístup na generovanie požadovaných súborov údajov. Základný princíp je založený na koncepcii profilov, ktoré obsahujú podrobné popisy útokov a abstraktné distribučné modely pre aplikácie, protokoly alebo sieťové entity nižšej úrovne. Vytvoria sa profily pre agentov, ktorí generujú reálnu prevádzku pre protokoly HTTP, SMTP, SSH, IMAP, POP3 a FTP. Agenti boli potom naprogramovaní tak, aby ich činnosť čo najúčinnejšie napodobnila aktivitu užívateľa. Útokové scenáre boli navrhnuté a vykonané tak, aby vyjadrili skutočné prípady škodlivého správania. Aplikovali sa v reálnom čase na fyzických zariadeniach prostredníctvom ľudskej pomoci.

Dátová množina UNB ISCX IDS 2012 sa skladá z označených sieťových záznamov, vrátane úplného paketového zaťaženia vo formáte *pcap*. Záznamy sú označené dvadsiatimi atribútmi a podľa

Soheily-Khah a spol. [29] má dátová množina viac ako dva miliónov záznamov z čoho 2% dát predstavujú útok. Skladá sa zo sieťovej aktivity zaznamenávanú počas siedmich dní. Tieto aktivity predstavovali normálnu sieťovú premávku a útočnú (infiltrácie siete z vnútra, HTTP DoS, DDoS pomocou IRC Botnet a Brute Force SSH) a má nasledujúce charakteristiky:

- **Realistická sieťová prevádzka** – V ideálnom prípade súbor údajov by nemal vykazovať žiadne nežiadúce sieťové vlastnosti. Toto zabezpečuje jasnejší obraz o skutočných účinkoch útokov na sieť.
- **Označenie súboru údajov** – Táto charakteristika má obrovský význam pri hodnotení rôznych detekčných mechanizmov. Vytváranie súboru údajov v kontrolovanom a deterministickom prostredí umožňuje rozlišovať anomálnu aktivitu od normálnej sieťovej prevádzky.
- **Zachytenie celkovej sieťovej interakcie** – Tieto informácie sú nevyhnutné pre hodnotenie a správnu interpretáciu výsledkov.
- **Úplné zachytenie** – Sieťové záznamy sú vytvorené v kontrolovanom prostredí *testbed*, čím sa úplne odstránia obavy týkajúce sa súkromia súvisiace so zdieľaním sieťových stôp, t.j. sa zachová prirodzenosť výsledného súboru údajov.
- **Rôzne scenáre narušenia** – Prostredníctvom vykonávania útočných scenárov a uplatňovania abnormálneho správania sa vytvoril rôznorodý súbor útokov na základe nedávnych trendov v bezpečnostných hrozbách.

Autori [29] píšu, že mnoho článkov ukázalo, že pokiaľ ide o heterogénne mnohorozmerné údaje, klasifikátor Náhodný les patrí medzi najúčinnejšie metódy. Preto navrhli klasifikačnú metódy postavenú na spomínanom klasifikátore. Výsledky predspracovania K-menas metódy sú použité ako vstupné údaje pre klasifikátor Náhodný les.

Autori navrhli, aby toky boli klasifikované podľa ich aplikačnej vrstvy (HTTPWeb, SSH, FTP, ICMP atď.). Vzhľadom na to, že bežné prevádzkové modely sú odlišné v závislosti na aplikáciu alebo službu, je oveľa efektívnejšie postaviť detektor narušenia pre každú z týchto vrstiev. Vo svoje práci porovnávajú navrhovaný algoritmus detekcie narušenia algoritmami ako sú: SVM, algoritmus vyhľadávania najbližšieho suseda (Nearest Neighbor Search – NNS), Naïve Bayes, Rozhodovacie stromy, Neurónová sieť a Náhodný les. Porovnávala sa presnosť, miera detekcie, a frekvencia falošných poplachov. V tabuľke č. 5 môžete vidieť výsledky porovnania na základe presnosti (accuracy). Hodnoty sú uvedené v percentách.

Tabuľka 5 – Tabuľka presností klasifikačných algoritmov pre ISCX IDS 2012 [29]

	SVM	**NNS**	**Naïve Bayes**	**Rozhodovací strom**	**Neurónová sieť**	**Náhodný les**	**K-means a Náhodný les**
HTTPWeb	98.99	99.70	98.04	99.89	99.02	99.88	99.91
SSH	99.47	99.90	99.22	99.87	99.89	99.89	99.98
ICMP	99.83	99.95	99.90	99.99	99.93	99.99	100.00
FTP	99.62	99.95	99.54	99.97	99.94	99.97	99.97
DNS	99.98	99.99	96.18	99.98	99.98	99.99	99.99

Na základe predchádzajúcej tabuľky je zrejmé, že metóda navrhnutá autormi [29] má najlepšie výsledky presnosti, potom nasleduje Rozhodovací strom a algoritmus vyhľadávania najbližšieho suseda.

2.7.4. Predspracovanie dát

Podľa Bahdwaj [4] údaje v surovom stave sú vysoko citlivé na šum, chýbajúce hodnoty a nekonzistentné rozdelenie. Kvalita údajov vo vysokej miere ovplyvňuje výsledky klasifikačných metód. Spracovanie údajov je jedným z najkritickejších krokov v procese dolovania v dátach, ktoré sa zaoberá prípravou a transformáciou pôvodného súboru údajov. Metódy predspracovania údajov sú nasledovné:

- **Čistenie dát** – Metóda založená na doplnení chýbajúcich hodnôt, vyhladení šumu v údajoch, identifikácii a odstránení vybočujúcich hodnôt (outliers) a riešení nezrovnalostí. V prípade doplnenia chýbajúcich hodnôt sa používajú nasledovné metódy:
  - Vynechanie n-tice, ak chýba označenie triedy.
  - Manuálne doplnenie hodnôt.
  - Doplnenie globálnej konštanty (napr. „neznáme“).
  - Doplnenie priemeru všetkých hodnôt.
  - Doplnenie priemeru určitej triedy.
  - Doplnenie najpravdepodobnejšej hodnoty.

  V prípade odstránenia šumu v údajoch sa používajú nasledovné metódy:
  - Lineárna, viacnásobná lineárna regresia.
  - Vybočujúce hodnoty možno identifikovať pomocou kombinácie počítačovej a ľudskej kontroly.
  - Vybočujúce hodnoty môžu byť detegované klastrovaním, kedy podobné hodnoty sú usporiadané do skupín.

- **Transformácia dát** – Táto činnosť transformuje údaje do vhodnej formy pre metódy dolovania v dátach. Transformácia dát zahŕňať nasledujúce metódy:
    - Normalizácia, t.j. transformácia údajov do rozsahu -1.0 – 1.0 alebo 0.0 – 1.0.
    - Odstránenie šumu.
    - Agregácia údajov.
    - Zovšeobecňovanie údajov.
- **Redukcia dát** – Proces redukcie objemu alebo rozmerov (počet atribútov) súboru dát. Práca s redukovaným súborom údajov je efektívnejšia. Nasledujúci zoznam vymenúva metódy redukcie dát:
    - Aplikovanie agregačných operácií pre údaje v konštrukcii dátovej kocky.
    - Odstránenie nepodstatných, nerelevantných alebo nadbytočných atribútov.
    - Zmenšenie veľkosti súboru údajov kompresiou, napríklad wavelet transformácia.
    - Nahradenie alebo odhad údajov menšími alternatívnymi údajmi, ako sú parametrické metódy alebo neparametrické metódy (klastrovanie, vzorkovanie alebo použitie histogramov).
    - Generovanie diskretizácie a konceptovej hierarchie, kde surové hodnoty údajov pre atribúty sú nahradené rozsahmi alebo vyššími koncepčnými úrovňami.

2.8. Zhodnotenie analýzy

V tejto kapitole sme sa venovali analýze danej problematiky. Analýzu problémovej oblasti pokladáme za kľúčovú, pretože sme sa dopracovali k podstatným výsledkom, ktoré pokladáme za potrebné k ďalšej práci na danej úlohe.

Podrobnejšie sme sa venovali typom systémom na detekciu narušenia siete. Popísali sme jednotlivé typy IDS z hľadiska nasadenia. Podrobnejšie sa venovať anomálne založeným detekčným systémom sme pokladali za dôležité, pretože takéto systémy majú schopnosť detegovať doteraz neznáme typy útokov. Potreba venovať sa týmto typom IDS je dôležité, pretože v počítačových sieťach sa objavujú nové zraniteľné miesta a útoky sú čoraz sofistikovanejšie. Opísali sme jednotlivé charakteristiky IDS, pretože si myslíme, že je potrebné vedieť limity týchto detekčných systémov, ak chceme prispieť do tejto problémovej oblasti výsledkom našej práce. V krátkosti sme sa venovali architektúre detekčných systémov. Opísali vybrané existujúce nástroje na detekciu útokov alebo narušenia a na záver tejto sekcie sme opísali jednotlivé spôsoby vyhodnocovania IDS. Táto kapitola je veľmi dôležitou súčasťou analýzy, pretože vďaka nej dokážeme pochopiť kvalitu použitého modelu alebo techniky, ktorá sa použila pri klasifikácii.

Ďalej sme sa venovali analýze sieťových útokov. Keďže IDS využíva rôzne techniky na detekciu vniknutí do počítačovej siete, sme pokladali analýzu tejto oblasti taktiež za kľúčovú. Popísali sme jednotlivé typy sieťových útokov, proti ktorým budeme konfrontovaný aj v našej ďalšej práci. Vďaka naštudovanej literatúry sme zahrnuli do kapitoly sieťové útoky aj všeobecný proces odhaľovania sieťových útokov vo forme obranného mechanizmu.

Ďalšia kľúčová oblasť v rámci tematiky, ktorú táto práca rieši je strojové učenie. Klasifikačné algoritmy budú tvoriť jadro tejto práce, a preto analýza existujúcich algoritmov a podrobnejší popis najpoužívanejšej metódy – neurónová sieť je veľmi dôležitá.

Na záver sme zanalyzovali tri najpoužívanejšie dátové množiny. Opísali sme základné charakteristiky vybraných dátových množín. V tejto kapitole sa nám podarilo na základe preštudovanej literatúry uviesť výsledky experimentov citovaných autorov pre budúce porovnanie s našou prácou. Myslíme si, že takáto forma analýzy je lepšia z hľadiska možnosti porovnania sa voči existujúcim riešeniam nad vybranými dátovými množinami ako samotná analýza dátových množín bez údajov z experimentov. Ďalej sa nám podarilo spísať kroky potrebné pre prípravu dát, pred tým než ich použijeme ako vstup pre metódy dolovania v dátach. Príprava dát je dôležitou súčasťou pri získavaní údajov, pretože údaje v reálnom svete sú zvyčajne neúplné, obsahujú šum a sú nekonzistentné.

3. Špecifikácia požiadaviek

V tejto kapitole sú špecifikované funkčné aj nie-funkčné vlastnosti navrhovaného riešenia.

3.1. Funkčné vlastnosti

Vytváraný programový modul bude vhodným spôsobom dokumentovať vybrané metódy strojového učenia. Výstupom modulu budú grafické vizualizačné prvky na vhodnú reprezentáciu výsledkov. Funkcie programového modulu budú pokrývať nasledujúce oblasti:

- Výber rôznych klasifikačných metód strojového učenia.
- Výber dátovej množiny, na ktorú sa má aplikovať zvolená metóda/y.
- Zobrazenie parametrov vybranej metódy.
- Vytvorenie výstupu klasifikácie v textovej a grafickej podobe, napríklad tabuľky výsledkov a grafy.
- Porovnanie rôznych metód.
- Možnosť zobraziť a uložiť proces spracovania a predprípravy dátovej množiny.
- Možnosť zobrazenia a uloženia výstupov vyhodnotenia.

3.2. Nie-funkčné vlastnosti

Navrhované riešenie bude implementovaný ako programový modul, a preto nebude mať používateľské rozhranie. Modul musí byť voľne prístupný pre verejnosť. Taktiež musí byť zabezpečená vhodná dokumentácia. Spôsob ovládania modulu musí byť prehľadné a intuitívne pre používateľa a bez redundantných informácií. Ovládanie, nastavovanie parametrov a prístup k výstupom modulu nesmú byť náročné, aby sa používateľ vedel rýchlo zorientovať a používať vytvorený programový modul.

4. Návrh riešenia

Na základe podrobnej analýzy anomálií, anomálne založených detekčných systémov a klasifikačných algoritmov, ktoré sa bežne používajú na detekciu anomálnej sieťovej premávky sme sa rozhodli vybrať práve tento smer, ktorým sa naša práca bude uberať.

V tejto práci navrhneme a implementujeme programový modul, ktorý bude implementovať rôzne metódy strojového učenia pre odhalenie útočnej/anomálnej sieťovej premávky. Táto práca bude mať výskumný charakter z hľadiska hlbšej analýzy jednotlivých metód strojového učenia, ktorým v jednotlivých kapitolách nižšie v tejto práci sa budeme podrobne venovať. Implementujeme ich nad vybranou dátovou množinou ale množinami s cieľom porovnať výsledky hodnotení a vyhodnotiť, ktorá metóda je pre aký typ problematiky sieťového narušenia najvhodnejšia. Zameriame sa predovšetkým na hľadanie anomálií v sieťovej premávke a metódam ktoré sú určené na ich odhaľovanie. Veľkým problémom a nedostatkom v tejto oblasti je vhodné nastavenie metód strojového učenia. Preto naša práca bude venovaná aj výskumu vhodných nastavení parametrov pre vybrané metódy. Ako bolo spomenuté aj v kapitole 2.6.2 Neurónová sieť, veľa závisí od nastavenia neurónovej siete, kde je potrebné určiť počet neurónov, počet vrstiev neurónovej siete, algoritmus a prenosovú funkciu pre trénovanie. Podobným spôsobom je potrebné určiť parametre aj pre ostatné metódy strojového učenia. Na základe rôznych nastavení a vstupov do metód dokážeme optimalizovať klasifikáciu a tým pádom pri vhodných nastaveniach dosahovať dobré výsledky hodnotenia modelu. Spomenuli sme vstupné dáta pre klasifikačnú metódu, ktoré sú taktiež kľúčové pre dosahovanie dobrých výsledkov. V tejto práci bude kľúčové vhodné predspracovanie vybranej dátovej množiny, a odhalenie závislostí medzi jeho atribútmi, ktoré majú značný vplyv na odhaľovanie útokov. Na základe konzultácii s vedúcim tejto práce vybrané metódy budú implementované na vybraných dátových množinách, ktorým sme sa venovali v kapitole 2.7 Dátové množiny. Vybraným dátovým množinám je potrebné dokonale porozumieť. Zanalyzujeme jednotlivé atribúty vybraných dátových množín a po absolvovaní dostatočnej analýzy dát a pochopení súvislostí medzi atribútmi predspracujeme a transformujeme dáta do formy vstupu pre metódy dolovania v dátach. Následne na základe analýzy spôsobu vyhodnocovania metód strojového učenia (viď. kapitolu 2.4 Spôsob vyhodnocovania IDS) vhodným spôsobom vyhodnotíme vybrané metódy a na základe výsledkov spíšeme výhody a nevýhody použitých metód. Výsledky použitých metód porovnáme a interpretujeme textovým a grafickým spôsobom.

Našim cieľom je, aby táto práca prispela svojim výsledkom v tejto problémovej oblasti stým, že spíšeme jednotnú analýzu vplyvov rôznych nastavení metód strojového učenia a rôznych spôsobov predspracovania vybraných dátových množín.

Bibliografia

[1] AHMED, Martuza, Rima PAL, Md. Mojammel HOSSAIN, Md. Abu Naser BIKAS a Md. Khalad HASAN. NIDS: A Network Based Approach to Intrusion Detection and Prevention. In: 2009 International Association of Computer Science and Information Technology - Spring Conference [online]. IEEE, 2009, 2009, s. 141-144 [cit. 2019-05-12]. DOI: 10.1109/IACSIT-SC.2009.96. ISBN 978-0-7695-3653-8. Dostupné z: http://ieeexplore.ieee.org/document/5169326/

[2] AL-JARRAH, Omar a Ahmad ARAFAT. Network Intrusion Detection System using attack behavior classification. In: 2014 5th International Conference on Information and Communication Systems (ICICS) [online]. IEEE, 2014, 2014, s. 1-6 [cit. 2019-05-12]. DOI: 10.1109/IACS.2014.6841978. ISBN 978-1-4799-3023-4. Dostupné z: http://ieeexplore.ieee.org/document/6841978/

[3] ANANTHI, J. Vijitha a S. VENGATESAN. Detection of various attacks in wireless adhoc networks and its performance analysis. In: 2017 International Conference on Inventive Computing and Informatics (ICICI) [online]. IEEE, 2017, 2017, s. 754-757 [cit. 2019-05-12]. DOI: 10.1109/ICICI.2017.8365237. ISBN 978-1-5386-4031-9. Dostupné z: https://ieeexplore.ieee.org/document/8365237/

[4] BHARDWAJ, Anshu. Data Preprocessing Techniques for Data Mining. Data Mining Techniques and Tools for Knowledge Discovery in Agricultural Datasets [online]. New Delhi: Division of Computer Applications Indian Agricultural Statistics Research Institute (ICAR), s. 139-144 [cit. 2019-05-22]. Dostupné z: http://iasri.res.in/ebook/win_school_aa/notes/Data_Preprocessing.pdf

[5] BHARDWAJ, Anshu. Evaluation Measures for Data Mining Tasks. Data Mining Techniques and Tools for Knowledge Discovery in Agricultural Datasets [online]. New Delhi: Division of Computer Applications Indian Agricultural Statistics Research Institute (ICAR), s. 145-152 [cit. 2019-05-19]. Dostupné z: http://iasri.res.in/ebook/win_school_aa/notes/Evaluation_Measures.pdf

[6] BHATTACHARYYA, Dhruba Kumar a Jugal Kumar KALITA. Network Anomaly Detection [online]. Chapman and Hall/CRC, 2013 [cit. 2019-05-12]. DOI: 10.1201/b15088. ISBN 9780429166877.

[7] Canadian Institute for Cybersecurity [online]. Fredericton [cit. 2019-05-13]. Dostupné z: https://www.unb.ca/cic/datasets/index.html

[8] DHANABAL, L. a S. P. SHANTHARAJAH. A Study On NSL-KDD Dataset For Intrusion Detection System Based On Classification Algorithms. International Journal of Advanced Research in Computer and Communication Engineering [online]. 2015, 6(4), 446-452 [cit. 2019-05-13]. DOI: 10.17148/IJARCCE.2015.4696. ISSN 2278-1021. Dostupné z: https://pdfs.semanticscholar.org/1b34/80021c4ab0f632efa99e01a9b073903c5554.pdf

[9] GARG, Akash a Prachi MAHESHWARI. A hybrid intrusion detection system: A review. In: 2016 10th International Conference on Intelligent Systems and Control (ISCO) [online]. IEEE, 2016, 2016, s. 1-5 [cit. 2019-05-12]. DOI: 10.1109/ISCO.2016.7726909. ISBN 978-1-4673-7807-9. Dostupné z: http://ieeexplore.ieee.org/document/7726909/

[10] HADDADI, Fariba, Sara KHANCHI, Mehran SHETABI a Vali DERHAMI. Intrusion Detection and Attack Classification Using Feed-Forward Neural Network. In: 2010 Second International Conference on Computer and Network Technology [online]. IEEE, 2010, 2010, s. 262-266 [cit. 2019-05-12]. DOI: 10.1109/ICCNT.2010.28. ISBN 978-1-4244-6961-1. Dostupné z: http://ieeexplore.ieee.org/document/5474495/

[11] HUSSAIN, Jamal a Aishwarya MISHRA. Performance Analysis of Some Neural Network Algorithms using NSL-KDD Dataset. International Journal of Computer Trends and Technology [online]. 2017, 50(1), 43-49 [cit. 2019-05-12]. DOI: 10.14445/22312803/IJCTT-V50P107. ISSN 22312803. Dostupné z: http://www.ijcttjournal.org/archives/ijctt-v50p107

[12] JAIN, Anurag; BHUPENDRA, L. Classifier selection models for intrusion detection system (IDS). Informatics Engineering, an International Journal (IEIJ), 2016, 4.1: 1-11.

[13] JAJISH, Thomas. Intrusion Detection Systems (IDS), Network Intrusion Detection System (NIDS), Host Intrusion Detection System (HIDS), Signatures, Alerts, Logs, False Alarms. Sensor. Free Networking tutorials, System Administration Tutorials and Security Tutorials - omnisecu.com [online]. [cit. 2019-05-12]. Dostupné z: http://www.omnisecu.com/security/infrastructure-and-email-security/intrusion-detection-systems-ids.php

[14] KUMAR, Gulshan. Evaluation Metrics for Intrusion Detection Systems - A Study. International Journal of Computer Science and Mobile Applications [online]. 2014, (11), 11-17 [cit. 2019-05-12]. ISSN 2321-8363. Dostupné z: https://www.researchgate.net/publication/311108073_Evaluation_Metrics_for_Intrusion_Detection_Systems-A_Study

[15] LI, Jing a Chunbo DONG. Research on Network Security Situation Prediction-Oriented Adaptive Learning Neuron. In: 2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing [online]. IEEE, 2010, 2010, s. 483-485 [cit. 2019-05-12]. DOI: 10.1109/NSWCTC.2010.247. ISBN 978-1-4244-6597-2. Dostupné z: http://ieeexplore.ieee.org/document/5480921/

[16] LI PENG, YUEMING LU, JIEFU GAN a HANG CHENG. Design and implementation of network attacks detection module. In: Third International Conference on Cyberspace Technology (CCT 2015) [online]. Institution of Engineering and Technology, 2015, 2015, 5 .-5 . [cit. 2019-05-12]. DOI: 10.1049/cp.2015.0861. ISBN 978-1-78561-089-9. Dostupné z: https://digital-library.theiet.org/content/conferences/10.1049/cp.2015.0861

[17] LIU, Gao, Zheng YAN a Witold PEDRYCZ. Data collection for attack detection and security measurement in Mobile Ad Hoc Networks: A survey. Journal of Network and Computer Applications [online]. 2018, 105, 105-122 [cit. 2019-05-12]. DOI: 10.1016/j.jnca.2018.01.004. ISSN 10848045. Dostupné z: https://linkinghub.elsevier.com/retrieve/pii/S1084804518300043

[18] MEHMOOD, Tahir a Helmi B Md RAIS. Machine learning algorithms in context of intrusion detection. In: 2016 3rd International Conference on Computer and Information Sciences (ICCOINS) [online]. IEEE, 2016, 2016, s. 369-373 [cit. 2019-05-13]. DOI: 10.1109/ICCOINS.2016.7783243. ISBN 978-1-5090-2549-7. Dostupné z: http://ieeexplore.ieee.org/document/7783243/

[19] MOUSTAFA N., SLAY J. (2018) A Network Forensic Scheme Using Correntropy-Variation for Attack Detection. In: Peterson G., Shenoi S. (eds) Advances in Digital Forensics XIV. DigitalForensics 2018. IFIP Advances in Information and Communication Technology, vol 532. Springer, Cham

[20] MOUSTAFA, Nour a Jill SLAY. The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set. Information Security Journal: A Global Perspective [online]. 2016, 25(1-3), 18-31 [cit. 2019-05-19]. DOI: 10.1080/19393555.2015.1125974. ISSN 1939-3555. Dostupné z: http://www.tandfonline.com/doi/full/10.1080/19393555.2015.1125974

[21] MOUSTAFA, Nour a Jill SLAY. UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In: 2015 Military Communications and Information Systems Conference (MilCIS) [online]. IEEE, 2015, 2015, s. 1-6 [cit. 2019-05-13]. DOI: 10.1109/MilCIS.2015.7348942. ISBN 978-1-4673-7007-3. Dostupné z: http://ieeexplore.ieee.org/document/7348942/

[22] REVATHI, S. a A. MALATHI. A Detailed Analysis on NSL-KDD Dataset Using Various Machine Learning Techniques for Intrusion Detection. International Journal Of Engineering Research & Technology [online]. IJERT, 2013, 12(2), 1848-1853 [cit. 2019-05-13]. ISSN 2278-0181. Dostupné z: https://www.ijert.org/research/a-detailed-analysis-on-nsl-kdd-dataset-using-various-machine-learning-techniques-for-intrusion-detection-IJERTV2IS120804.pdf

[23] SAMRIN, Rafath a D VASUMATHI. Review on anomaly based network intrusion detection system. In: 2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT) [online]. IEEE, 2017, 2017, s. 141-147 [cit. 2019-05-12]. DOI: 10.1109/ICEECCOT.2017.8284655. ISBN 978-1-5386-1205-7. Dostupné z: http://ieeexplore.ieee.org/document/8284655/

[24] SANI, Yusuf, Ahmed MOHAMEDOU, Khalid ALI, Anahita FARJAMFAR, Mohamed AZMAN a Solahuddin SHAMSUDDIN. An overview of neural networks use in anomaly Intrusion Detection Systems. In: 2009 IEEE Student Conference on Research and Development (SCOReD) [online]. IEEE, 2009, 2009, s. 89-92 [cit. 2019-05-12]. DOI: 10.1109/SCORED.2009.5443289. ISBN 978-1-4244-5186-9. Dostupné z: http://ieeexplore.ieee.org/document/5443289/

[25] SAXENA, Aumreesh Ku., Sitesh SINHA a Piyush SHUKLA. General study of intrusion detection system and survey of agent based intrusion detection system. In: 2017 International Conference on Computing, Communication and Automation (ICCCA) [online]. IEEE, 2017, 2017, s. 471-421 [cit. 2019-05-12]. DOI: 10.1109/CCAA.2017.8229866. ISBN 978-1-5090-6471-7. Dostupné z: http://ieeexplore.ieee.org/document/8229866/

[26] SCHAELICKE, Lambert, Thomas SLABACH, Branden MOORE a Curt FREELAND. Characterizing the Performance of Network Intrusion Detection Sensors. VIGNA, Giovanni, Christopher KRUEGEL a Erland JONSSON, ed. Recent Advances in Intrusion Detection [online]. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, 2003, s. 155-172 [cit. 2019-05-12]. Lecture Notes in Computer Science. DOI: 10.1007/978-3-540-45248-5_9. ISBN 978-3-540-40878-9. Dostupné z: http://link.springer.com/10.1007/978-3-540-45248-5_9

[27] SCHOONJANS, Frank. ROC curve analysis. MedCalc [online]. Belgium [cit. 2019-05-06]. Dostupné z: https://www.medcalc.org/manual/roc-curves.php

[28] SHEELA EVANGELIN PRASAD, S. N., M. V. SRINATH a Murtaza SAADIQUE BASHA. Intrusion Detection Systems, Tools and Techniques – An Overview. Indian Journal of Science and Technology [online]. 2015, 8(35) [cit. 2019-05-12]. DOI: 10.17485/ijst/2015/v8i35/80108. ISSN 0974-5645. Dostupné z: http://www.indjst.org/index.php/indjst/article/view/80108

[29] SOHEILY-KHAH, Saeid, Pierre-Francois MARTEAU a Nicolas BECHET. Intrusion Detection in Network Systems Through Hybrid Supervised and Unsupervised Machine Learning Process: A Case Study on the ISCX Dataset. In: 2018 1st International Conference on Data Intelligence and Security (ICDIS) [online]. IEEE, 2018, 2018, s. 219-226 [cit. 2019-05-19]. DOI: 10.1109/ICDIS.2018.00043. ISBN 978-1-5386-5762-1. Dostupné z: https://ieeexplore.ieee.org/document/8367767/

[30] The UNSW-NB15 Dataset Description [online]. 2018 [cit. 2019-05-14]. Dostupné z: https://www.unsw.adfa.edu.au/unsw-canberra-cyber/cybersecurity/ADFA-NB15-Datasets/

[31] THOMAS, Rajesh a Deepa PAVITHRAN. A Survey of Intrusion Detection Models based on NSL-KDD Data Set. In: 2018 Fifth HCT Information Technology Trends (ITT) [online]. IEEE, 2018, 2018, s. 286-291 [cit. 2019-05-13]. DOI: 10.1109/CTIT.2018.8649498. ISBN 978-1-5386-7147-4. Dostupné z: https://ieeexplore.ieee.org/document/8649498/

[32] TIWARI, Mohit, Raj KUMAR, Akash BHARTI a Jai KISHAN. Intrusion Detection Systems. International Journal of Computer Science and Mobile Applications [online]. 2017, (5), 38-44 [cit. 2019-05-12]. ISSN 2320-8163. Dostupné z: https://www.researchgate.net/publication/316599266_INTRUSION_DETECTION_SYSTEM

[33] WARZYŃSKI, Arkadiusz; KOŁACZEK, Grzegorz. Intrusion detection systems vulnerability on adversarial examples. In: 2018 Innovations in Intelligent Systems and Applications (INISTA). IEEE, 2018. p. 1-4.

[34] WILLIAMS, Nigel, Sebastian ZANDER a Grenville ARMITAGE. A preliminary performance comparison of five machine learning algorithms for practical IP traffic flow classification. ACM SIGCOMM Computer Communication Review [online]. 2006, 36(5) [cit. 2019-05-22]. DOI: 10.1145/1163593.1163596. ISSN 01464833. Dostupné z: http://portal.acm.org/citation.cfm?doid=1163593.1163596

[35] XIA WEI-WEI a WANG HAI-FENG. Prediction model of network security situation based on regression analysis. In: 2010 IEEE International Conference on Wireless Communications, Networking and Information Security [online]. IEEE, 2010, 2010, s. 616-619 [cit. 2018-11-10]. DOI: 10.1109/WCINS.2010.5541853. ISBN 978-1-4244-5850-9. Dostupné z: http://ieeexplore.ieee.org/document/5541853/

[36] ZAMAN, Marzia a Chung-Horng LUNG. Evaluation of machine learning techniques for network intrusion detection. In: NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium [online]. IEEE, 2018, 2018, s. 1-5 [cit. 2019-05-21]. DOI: 10.1109/NOMS.2018.8406212. ISBN 978-1-5386-3416-5. Dostupné z: https://ieeexplore.ieee.org/document/8406212/

Príloha A: Plán práce

V tabuľke č. A.1 je uvedený plán práce v rámci prvej etapy riešenia diplomovej práce.

Tabuľka A.6 – Plán práce k DP I

DP I	
Týždeň	**Cieľ**
1-2	Analýza existujúcich riešení IDS (štruktúra, metódy, algoritmy).
3-5	Analýza metód spracovania veľkej množiny dát z prostredia počítačových sietí.
6-9	Analýza a zaúčanie sa do strojového učenia.
9-10	Odskúšanie získaných znalostí na vzorových príkladoch. Experimentovanie.
11-12	Úprava a dokončenie dokumentácie.

Jednotlivé body vyššie uvedeného plánu boli mierne náročné na vypracovanie. Podarilo sa nám zanalyzovať rôzne aspekty vybranej problémovej oblasti.

Zanalyzovali sme IDS systémy, ich architektúru, existujúce riešenia zaoberajúce sa odhalením sieťových útokov a spôsob vyhodnocovania úspešnosti IDS. Taktiež sa nám podarilo dostatočne zanalyzovať sieťové útoky, dátové množiny a strojové učenie. V tejto etape diplomovej práce sme nestihli aplikovať získané znalosti (experimentovať) na vzorových príkladoch.

Navrhli sme vlastné riešenie na základe preštudovania danej problematiky. Stanovenie podmienok na prvú etapu sa nám podarilo splniť a dokument sa tiež úspešne upravil do finálnej podoby na základe podmienok vedúceho práce.

V tabuľke č. A.1 je uvedený plán práce v rámci druhej etapy riešenia diplomovej práce.

Tabuľka A.7 – Plán práce k DP II

DP II	
Týždeň	**Cieľ**
1-2	Zbieranie dát z prostredia počítačových sietí a ich predpríprava.
3-5	Návrh vlastného riešenia.
6-9	Implementácia zadania.
9-10	Analýza výsledkov vlastného riešenia. Odhalenie nedostatkov. Zapracovanie zmien do implementácie.
11-12	Úprava a dokončenie dokumentácie. Príprava na obhajobu DP II.

V tabuľke č. 1 je uvedený plán práce v rámci tretej etapy riešenia diplomovej práce.

Tabuľka A.8 – Plán práce k DP III

DP III	
Týždeň	**Cieľ**
1-2	Korekcia chýb návrhu na základe pripomienok od vedúceho projektu.
3-5	Testovanie a zhodnotenie výsledkov implementovaného nástroja.
6-9	Finálne úpravy v projekte. Kontrola a odhaľovanie chýb.
9-10	Dokončenie dokumentácie.
11-12	Príprava na obhajobu DP III.

Príloha B: Technická dokumentácia

Príloha C: Obsah elektronického média