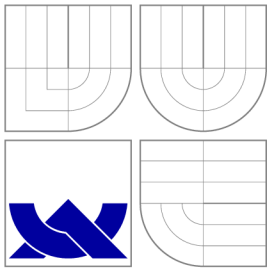
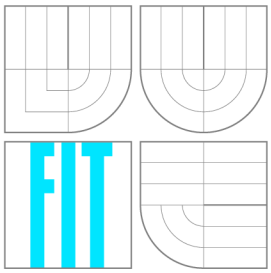


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

SYSTÉM PRO ROZPOZNÁVÁNÍ APT ÚTOKŮ

SYSTEM FOR DETECTION OF APT ATTACKS

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. ONDŘEJ HUJŇÁK

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. MAROŠ BARABAS

BRNO 2016

Zadání diplomové práce/18999/2015/xhujna00

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav inteligentních systémů Akademický rok 2015/2016

Zadání diplomové práce

Řešitel: **Hujňák Ondřej, Bc.**
Obor: Bezpečnost informačních technologií
Téma: **Systém pro rozpoznávání APT útoků**
System for Detection of APT Attacks
Kategorie: Bezpečnost

Pokyny:

1. Prostudujte současný stav znalostí o APT útocích se zaměřením na průniky do sítí, modelování chování uživatelů a útočníků.
2. Definujte klíčové charakteristiky pro rozpoznání APT útoků pomocí modelování chování uživatelů v síti a detekce anomálií.
3. Navrhněte strukturu a funkci systému pro detekci APT útoků v prostředí počítačové sítě a připojených uzlů.
4. Navržený systém implementujte.
5. Systém otestujte a zhodnoťte dosažené výsledky.

Literatura:

- Literatura podle pokynů vedoucího

Při obhajobě semestrální části projektu je požadováno:

- Body 1 až 3 zadání

Podrobné závazné pokyny pro vypracování diplomové práce naleznete na adrese http://www.fit.vutbr.cz/info/szz/

Technická zpráva diplomové práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap, které byly vyřešeny v rámci dřívějších projektů (30 až 40% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Barabas Maroš, Ing.**, UITS FIT VUT
Datum zadání: 1. listopadu 2015
Datum odevzdání: 25. května 2016

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav inteligentních systémů
612 66 Brno, Božetěchova 2

L.S.

doc. Dr. Ing. Petr Hanáček
vedoucí ústavu

Abstrakt

Práce se zabývá APT útoky, což jsou cílené a profesionálně vedené útoky vyznačující se dlouhou dobou trvání s využitím pokročilých technik. Práce shrnuje dosavadní znalosti o APT útocích a je v ní navrženo sedm symptomů využitelných pro zjištění, že daná organizace se nachází pod APT útokem. Na spolupůsobení symptomů je v práci navržen systém pro rozpoznávání APT útoků. Tento systém je rozpracován pro útoky v prostředí počítačové sítě a využívá modelování chování uživatelů v síti pro detekci anomálií. Detektor je založen na metodě k-nearest neighbors (k-NN). Schopnost rozpoznávání APT útoku v síťovém prostředí je ověřena implementací detektoru a jeho otestováním.

Abstract

The thesis investigates APT attacks, which are professional targeted attacks that are characterised by long-term duration and use of advanced techniques. The thesis summarises current knowledge about APT attacks and suggests seven symptoms that can be used to check, whether an organization is under an APT attack. Thesis suggests a system for detection of APT attacks based on interaction of those symptoms. This system is elaborated further for detection of attacks in computer networks, where it uses user behaviour modelling for anomaly detection. The detector uses k-nearest neighbors (k-NN) method. The APT attack recognition ability in network environment is verified by implementing and testing this detector.

Klíčová slova

APT útok, monitorování síťového provozu, behaviorální analýza, NBA, detekce anomálií, nearest neighbors, k-NN

Keywords

APT attack, network monitoring, behavioral analysis, NBA, outlier detection, nearest neighbors, k-NN

Citace

HUJŇÁK, Ondřej. *Systém pro rozpoznávání APT útoků.* Brno, 2016. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Barabas Maroš.

Systém pro rozpoznávání APT útoků

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana Ing. Maroše Barabase a uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Ondřej Hujňák
25. května 2016

Poděkování

Děkuji panu Ing. Barabasovi za vedení mé práce a profesnímu sdružení ISACA za poskytnuté materiály.

Obsah

Kapitola 1

Úvod a cíle práce

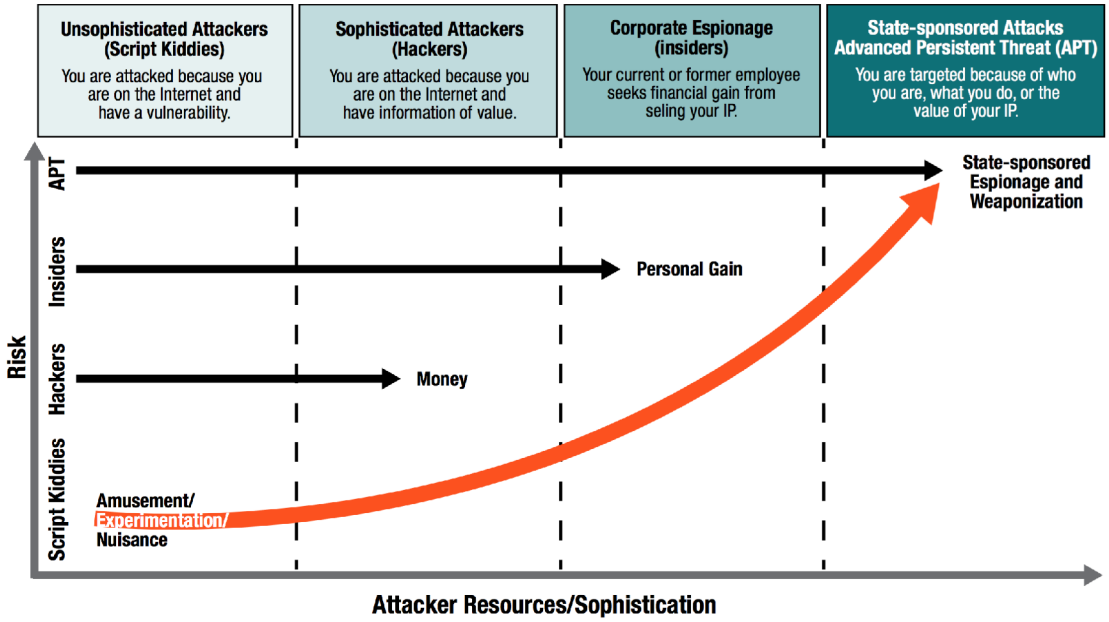
V dnešní informační éře se klade stále větší důraz na informační systémy a stále více aktiv je uchováváno v elektronické podobě. V elektronické formě jsou dnes k dispozici téměř veškeré informace od veřejných encyklopedických údajů po soukromá data jednotlivců i organizací. Tato data mohou představovat jak přímou hodnotu, jedná-li se například o elektronické peníze, či cenné papíry, tak i nepřímou hodnotu obsaženou například v citlivých informacích, jejichž únik znamená pro vlastníka v dnešním konkurenčním světě velké problémy.

S rozvojem internetu došlo k propojení uzlů po celém světě, což sice usnadnilo přístup k informacím, ale nese s sebou významně vyšší riziko neoprávněného přístupu, kterému je třeba čelit. Před dobou internetu bylo nutné pro získání dat použít přímý fyzický přístup k jejich úložišti, ať už se jednalo o data elektronická, či jiná. K datům, která jsou uložena na libovolném úložišti s přístupem k internetu se však dá přistoupit z jakéhokoli místa na světě. Z tohoto důvodu provází internet téměř od počátku problémy se zajištěním bezpečnosti, kterým se daří čelit s různým stupněm úspěšnosti.

V poslední době dochází k propojování různých zařízení přes internet stále více a tento trend se dále stupňuje s očekávaným nástupem tzv. Internet of Things. S rozvojem cloudových úložišť je k dispozici online stále více citlivých informací a zajištění jejich bezpečnosti je netriviálním úkolem, což dosvědčuje i velké množství útoků, ke kterým v poslední době došlo. Příkladem může být útok na službu Ashley Madison z července 2015, které byly odcizeny a následně zveřejněny citlivé informace o milionech uživatelů [11]. Jiným příkladem, který ukazuje, že ani velké státy nezvládají těmto hrozbám účinně čelit, je únik citlivých informací o státních zaměstnancích USA [21]. A posledním příkladem je únik informací z organizace Hacking Team, která se sama zabývá tvorbou software určeného k získávání informací, přesto však nedokázala útoku předejít [33]. Jak je z příkladů patrné, jsou útokům vystaveny velmi rozmanité organizace a zajištění bezpečnosti citlivých dat není samozřejmostí ani u státních organizací. Britská vláda pověřila společnost PwC vytvořením statistiky útoků na britské společnosti v roce 2015 a podle jejich výsledků došlo k nějakému úniku informací u 90% velkých britských organizací a ztráty těchto firem dosahovaly průměrně 1,46-3,14 milionů liber [29].

Každý počítač v síti dnes čelí různým útokům, které se velice liší jak frekvencí či intenzitou, tak i sofistikovaností. Velmi často se setkáváme s různými viry a to i velice jednoduchými až by se dalo říci „neškodnými“, v případě cílených útoků se však můžeme setkat s velice komplexními a technicky složitými útoky, které mohou vyžadovat i velmi dobrou znalost cíle [31]. Na obrázku 1.1 lze vidět diagram hrozeb jako závislost mezi rizikem a sofistikovaností útoku.

Nejčastější útoky jsou prováděny pomocí škodlivého software, který využívá některých



Obrázek 1.1: Diagram hrozeb v kyberprostoru, *převzato z [31]*

známých zranitelností. Tento software je rozšiřován plošně lidmi bez hlubších znalostí - tzv. Script Kiddies, kteří jsou většinou motivováni touhou experimentovat a poznávat. Ochrana proti tomuto typu útoků je zajišťována záplatováním známých zranitelností v software a rozpoznáváním škodlivého software podle jeho signatury pomocí antivirových nástrojů. Cílenější útoky jsou pak prováděny jedinci s lepšími znalostmi za účelem vlastního obohacení. Tito jedinci vytváří nástroje pro využití zranitelností a jsou schopni provézt i analýzu a přímý útok na svůj cíl. Ještě nebezpečnější jsou pak útočníci vynášející informace zevnitř. Jedná se o zaměstnance, kteří pro vlastní obohacení prodávají citlivé informace organizace. Největší dopad mají takzvané Advanced Persistent Threat (APT) útoky, což jsou sofistikované cílené útoky prováděné organizovanou skupinou útočníků s velmi dobrými znalostmi, které usilují všemi dostupnými prostředky o získání určitých informací nebo o napadnutí služeb cíle za účelem získání konkurenční výhody.

Tato práce se zabývá poslední skupinou útoků, tedy cílenými útoky vedenými sofistikovanými metodami. Tyto útoky znamenají pro organizace největší riziko, protože úspěšný útok s sebou nese velké ztráty a přitom jeho detekce je kvůli používaným metodám složitá. Jelikož naprostá většina těchto útoků probíhá po síti (ať už z vnější sítě, nebo vnitřní firemní sítě), je tato práce zaměřena na analýzu síťového provozu. Většina nyní dostupných nástrojů analyzuje provoz z pohledu dat, která jsou v něm zasílána, přičemž si tyto nástroje obvykle neumí poradit se šifrováním dat. Tato práce se zabývá analýzou chování účastníků v síti z pohledu metadat, která lze sledovat u veškeré síťové komunikace.

1.1 Cíle práce

Cíle diplomové práce jsou rozděleny na její hlavní cíl a pět postupových cílů, kterými bude tohoto hlavního cíle dosaženo.

1.1.1 Hlavní cíl práce

Hlavním cílem práce je navrhnout systém pro rozpoznávání APT (Advanced Persistent Threat) útoků a jeho rozpracování a ověření v prostředí počítačové sítě.

1.1.2 Postupové cíle práce

Pro dosažení hlavního cíle byly vytyčeny následující postupové cíle:

- Analýza současného stavu znalostí o APT útocích;
- Definice klíčových charakteristik pro rozpoznávání APT útoků;
- Navržení struktury a funkce systému pro rozpoznávání APT útoků v prostředí počítačové sítě;
- Implementace navrženého systému pro rozpoznávání APT útoků v prostředí počítačové sítě;
- Otestování navrženého systému pro rozpoznávání APT útoků a zhodnocení dosažených výsledků.

Pro splnění postupových cílů je nutné popsat charakteristiky APT útoků a nalézt symptomy, které indikují, že se organizace nachází pod APT útokem. Detekcí přítomnosti symptomů lze pak specifikovat systém, který vyhodnocuje jednotlivé symptomy a na jejich základě je schopen rozhodnout, zda je organizace pod APT útokem. Schopnost detekce APT útoku v prostředí počítačové sítě musí být ověřena vytvořeným detektorem.

1.2 Členění práce

Práce je rozdělena na pět hlavních kapitol odpovídajících logice řešení postupových cílů práce - *Úvod a cíle práce*, *APT útoky*, *Návrh systému pro rozpoznání APT útoku*, *Detekce abnormalit v síťové komunikaci* a *Závěr*.

Kapitola *Úvod a cíle práce* obsahuje motivaci a vymezení zaměření práce, podkapitolu *Cíle práce*, ve které je vytyčen hlavní cíl a postupové cíle vedoucí k jeho dosažení, a tuto podkapitolu *Členění práce*.

Kapitola *APT útoky* vymezuje APT útok, jak je v této práci chápán a obsahuje souhrn současných znalostí o těchto útocích. V podkapitole *Klíčové charakteristiky útoku* je popsáno, čím se APT útoky vyznačují a jak je odlišit od jiných útoků, dále je v podkapitole *Životní cyklus útoku* popsán průběh typického APT útoku a přiblíženy jednotlivé fáze, kteými APT útok prochází. Podkapitola *Současné způsoby obrany* se zabývá metodami a prvky, které lze využít pro obranu proti APT útokům.

V rámci kapitoly *Návrh systému pro rozpoznání APT útoku* jsou nejprve v podkapitole *Identifikace symptomů APT útoku* navrženy symptomy, podle kterých lze rozpoznat APT útoky. Způsoby, kterými lze sledovat přítomnost symptomů, jsou uvedeny v podkapitole *Možnosti detekce jednotlivých symptomů*. Na jejich základě je v podkapitole *Návrh architektury systému pro detekci útoků* navržen systém, který slouží k detekování, zda se organizace nachází pod APT útokem.

Systém pro rozpoznávání APT útoku je rozpracován pro detekci útoků v prostředí počítačové sítě v kapitole *Detekce anomalit v síťové komunikaci*. Nejdříve je v podkapitole

Behaviorální analýza popsán přehled způsobů provádění analýzy chování uživatelů a současně využití této analýzy pro detekování útoků. Na základě analýzy chování uživatelů v síťovém prostředí je v podkapitole *Návrh koncepce detektoru* specifikováno, jaké vlastnosti by měl detektor mít. Návrh architektury detektoru, popsaný v podkapitole *Návrh softwarové architektury*, zahrnuje vlastnosti vytyčené v předchozí podkapitole. Realizace detektoru je popsána v podkapitole *Implementace detektoru abnormálního chování* a ověření jeho funkčnosti a provedení zpětné vazby do návrhu detektoru obsahuje podkapitola *Praktické zkušenosti s detektorem.*

V kapitole *Závěr* je uvedeno dosažení postupových cílů a hlavního cíle práce. Doporučení z pohledu dalšího vývoje projektu je obsaženo v podkapitole *Další rozvoj systému pro rozpoznávání APT útoků.*

Kapitola 2

APT útoky

APT je zkratka Advanced Persistent Threat, tedy pokročilého profesionálně vedeného útoku, který se vyznačuje dlouhou dobou trvání, zpravidla v řádu měsíců až roků. APT útoky nejsou příležitostné či náhodné, ale cíl útoku je předem velice pečlivě nastudován.

Co přesně APT útok je a čím se odlišuje od jiných typů sofistikovaných útoků, ještě není zcela ustáleno. Někteří považují APT útok pouze za poslední článek ve vývoji útoků, jiní je považují za úplně nový přístup. Základní definice společnosti ISACA je definuje jednoduše jako *hrozbu, která je pokročilá a trvalá* [22]. Pokročilostí hrozby je myšleno použití netriviálních technik pro provedení útoku, trvalostí pak to, že APT útoky nejsou jednorázové akce, ale vyznačují se určitou delší dobou trvání. Útočníci se snaží dostat do cílového prostředí, kde setrvávají delší dobu a monitorují citlivé informace, nebo provádí jiné akce k dosažení svých cílů.

Cílem APT útoků se stávají nejčastěji velké organizace, kde mohou útočníci získat nejvíce ceněná data a nejčastější motivací je průmyslová či politická špionáž. Cílem však nemusí být jen data, ale také napadení nějaké služby a dokonce může také během probíhajícího útoku dojít ke změně cílů útočníka. Například může útočník usilovat o citlivé informace, po jejich získání je zneužije a dále sabotuje systémy napadeného [22]. Pro zjištění cílů útočníků je potřeba vzít v úvahu kým útočníci jsou, nebo kým jsou sponzorováni, a o jaká aktiva mohou mít zájem. Útočníků může být celá škála od organizovaných skupin, kterým jde o finanční obohacení, až po armády a rozvědky nepřátelských států, kterým jde o strategické informace.

2.1 Klíčové charakteristiky útoku

Jednou z věcí, kterou se APT útoky odlišují, je míra jejich zacílení. Tyto útoky nebývají náhodné a ani nenapadají plošně příliš mnoho zařízení. Jsou zamířeny na předem důsledně vytipované cíle s úmyslem získat přístup k určitým informacím nebo zdrojům (například v případě počítačového červa The Stuxnet Worm tento obsahoval omezení, kterými limitoval své rozšíření na cílové systémy [22]). Útočník nejdříve detailně zmapuje zamýšlený cíl, což mu umožňuje vytipovat si slabá místa pro průnik do cílového systému a provádět i útoky typu social engineering, při kterém se zaměřuje na osoby a spoléhá na selhání lidského faktoru. Velmi často jsou APT útoky vícevektorové, tedy využívají více způsobů kompromitace za účelem získání přístupu, přičemž se snaží využít více slabin v cílovém systému.

APT útoky jsou prováděny zkušenými odborníky s velmi dobrou znalostí dnešních tech-

nologií a představují většinou velmi pokročilé typy útoků využívající často zatím neznámých zranitelností (tzv. zero day), které útočníkům umožňují kompromitaci systémů a téměř se nedají odhalit pomocí tradičních postupů. Tradiční postupy založené na rozpoznávání signatur, tedy určitých sekvencí známých škodlivých programů, nemohou pro nově vyvinutý malware fungovat. Při úspěšné kompromitaci útočníci nasazují komplexní, často modulární, škodlivý software, který je schopen dále provádět velmi rozličné útočné akce v cílovém systému. Instalovaný software také může disponovat umělou inteligencí a často se snaží uniknout detekci, například přesouváním umístění škodlivého kódu nebo jeho šifrováním. Útočníci jsou schopni napadnout širokou škálu zařízení, o čemž svědčí například již zmíněný The Stuxnet Worm, který byl schopen infikovat i průmyslové počítače (PLC) [22].

Zásadním rozdílem oproti běžným typům útoků je délka trvání útoku. Běžné útoky většinou po průniku do systému provedou požadovanou akci, jako je například získání informací, omezení funkčnosti služby, nebo nainstalování škodlivého programu a dále již útok neprobíhá. V případě instalace škodlivého programu může dojít k začlenění napadeného stroje do tzv. botnetu který může být útočníkem využit i později, ale zpravidla bývá napadený počítač konečným cílem a dále již k útoku nedochází. Naopak u APT útoků dochází v případě kompromitace k instalaci programů, které jsou pak vstupní branou pro automatizované i manuální útoky v cílovém systému. APT útoky jsou často velice dlouhodobě probíhající útoky, které kladou velký důraz na minimální riziko odhalení. Proto mohou APT útoky postupovat relativně pomalu a vyhnout se detekci pomocí skrývání komunikace v běžném provozu. Jako příklad si můžeme vzít napadení amerického úřadu pro personální management z června minulého roku, které probíhalo minimálně rok (od července 2014) [21]. U APT útoků se většinou jen velice obtížně zjišťuje doba, kdy byl systém infikován. Organizace jsou z kapacitních důvodů nuceny starší logy mazat a stává se tak, že detekovaný APT útok je vystopován až do počátku uložených logů a nelze stanovit dobu, kdy došlo k infikaci systému. Také rozsah útoku se vzhledem k jeho době trvání a velkým možnostem variability určuje jen ztěžka.

2.2 Životní cyklus útoku

Na obrázku 2.1 je znázorněn životní cyklus APT útoku s vyznačenými fázemi. Jak lze na první pohled vidět, je životní cyklus kruhový, protože po úspěšně provedeném primárním cíli útoku často nedochází k ukončení útoku, ale k vytipování dalších cílů s využitím znalostí získaných v předchozích krocích.

Fáze životního cyklu APT útoku

Výběr cíle (Target Selection) Výběr cílů bývá důkladný a jako cíl nejsou určeny pouze nějaká aktiva, ale mohou to být i podružné cíle, které útočníkům následně umožní další postup. Pokud se během probíhajícího útoku změní priority nebo se vyskytnou nové informace není neobvyklé, že se plán útoku, nebo i výběr cíle, změní.

Vnější zmapování cíle (Target Research) Před samotným APT útokem dochází k co nejúplnějšímu zmapování cíle s důrazem na infrastrukturu, identifikací vhodných zdrojů informací a hledání zranitelností, které by mohly sloužit k napadení cíle.

Kompromitace (Target Penetration) Kompromitace je první přímou fází útoku, kdy se útočníci pomocí informací získaných v předchozí fázi dostávají do systému. K tomu většinou nedochází na místech, které mají pro útočníky přímou hodnotu, jako jsou

Obrázek 2.1: Životní cyklus APT útoků, *převzato z [22]*

počítače vrcholových managerů organizace nebo přímo datová centra obsahující chtěné informace, protože ty bývají dobře zabezpečeny a přímý útok by byl velice nesnadný. Prvotní kompromitace se většinou vydává cestou nejmenšího odporu a zasahuje stroje zaměstnanců na nižších pozicích, nebo dokonce externích spolupracovníků. Přes tyto body se pak útočníci dostávají do systému, který pak mohou lépe prozkoumat zevnitř a postupně napadnout cílová zařízení [18].

Zavedení trvalého spojení (Command and Control) Když jsou útočníci v systému, instalují malware, který jim umožní přístup do vnitřní sítě a provádění útoků zevnitř. Malware se po instalaci typicky spojí s útočníky a zahájí monitorování, nebo čeká na případné další instrukce, přičemž se snaží zůstat nedetekován. Často bývá modulární[22] a po úspěšné infiltraci si stáhne dodatečné moduly, které rozšíří možnosti monitorování sítě a provádění útoků. Tento malware může fungovat do značné míry autonomně a odesílat citlivé informace útočníkům.

Zmapování vnitřní sítě (Target Discovery) Když má útočník k dispozici spojení do vnitřní sítě napadeného, dochází k automatickému či manuálnímu zmapování vnitřních struktur a instalaci dalších malwarů pro zajištění připojení i v případě odhalení a neutralizování původního spojení. S detailní znalostí vnitřních struktur a informací z vnitřní sítě (které mohou obsahovat samy o sobě citlivé informace, a dokonce i přístupové údaje) je možné detailně naplánovat další postup.

Filtrování informací (Data Exfiltration) Po zmapování vnitřní sítě lze účinně získat požadované informace. Tyto se většinou shromažďují na některém napadeném zařízení v síti oběti a jsou dále komprimovány a šifrovány.

Distribuce informací (Intelligence Dissemination) Jsou-li požadované informace k dispozici a připraveny na odeslání, nastává samotné odeslání těchto dat útočníkům. Pro

minimalizaci pravděpodobnosti detekce jsou přenášená data obvykle skryta mezi legitimní komunikaci a pro případ odhalení nejsou data zasílána přímo útočníkům, ale cestují přes několik proxy serverů, které slouží ke skrytí útočníků.

Zneužití informací (Information Exploitation) Po získání informací je mohou útočníci využít ihned, nebo je pouze archivují pro vlastní potřebu. Pokud zjištěné informace vedou ke změně priorit či cílů, může být hned zahájen další útok, který bude nyní již operovat s mnohem detailnějšími informacemi o cílovém systému. Případem informace, která je pouze archivována, může být průmyslová, či politická špionáž, která nemá okamžité uplatnění, ale až v budoucnosti, po provedení určité akce (uvedení produktu na trh, válka).

2.3 Současné způsoby obrany

Jak již bylo zmíněno, je obrana proti APT útokům značně komplikovaná a organizace nejsou na tento typ útoků připraveny, což je výsledkem velkého počtu úspěšných útoků v poslední době. K bezpečnosti v oblasti IT se příliš dlouhou dobu přistupovalo velmi benevoletně a finanční prostředky vynakládané na zajištění bezpečnosti byly směšné v porovnání s prostředky vynakládanými na rozvoj software. Až kolem roku 2005[31] dochází ke zlomu, kdy se začaly organizace, v důsledku nárustu kyberkriminality, více zajímat o bezpečnost a více do ní investovat. V roce 2010 mělo 86% obětí kyberůtoků k dispozici důkazy o napadení, přesto pouze 61% z nich odhalilo kompromitaci vlastním přičiněním, zatímco ostatní byly upozorněny třetí stranou [17].

Vzhledem k tomu, že dnes existuje obrovské množství malware, a v APT útocích jsou navíc často používány zcela nové a na míru vyrobené programy, je detekce APT útoku velmi obtížná. Navíc vzhledem k dlouhé době trvání odhalení APT útoků je pravděpodobné, že nyní detekované způsoby jsou již zastaralé a útočníci využijí zkušeností z úspěšných útoků pro tvorbu nových, sofistikovanějších a ještě hůře detekovatelných postupů.

V současné době nedochází k žádným speciálním akcím, kterými by organizace předcházely, nebo se bránily, APT útokům. Pro prevenci před těmito útoky jsou využity konvenční způsoby obrany, které zřídka bývají doplněny o určité heuristiky pro odhalování nových způsobů útoku nebo o nějakou formu umělé inteligence.

Jako první bývají nasazovány antivirové programy pro rozpoznávání nevyžádaných programů na jednotlivých počítačích. Tyto programy jsou schopny dobře rozpoznat známý malware a často nabízejí i doprovodné funkce pro předcházení kompromitace systému, jako je skenování souborů v sandboxovaném prostředí při stahování, varování uživatelů při přístupu na známé podvodné stránky a hlídání bezpečnostních aktualizací pro nainstalované programy. Mohou se snažit detekovat nové hrozby podle sledování chování procesů a sdílení veškerých poznatků v komunitě uživatelů.

Základní síťovou bezpečnost zajišťuje rozdělení sítě do logických celků a hlídání perimetru pomocí firewallu. Firewallů existuje více typů, jak hardwarové, tak i softwarové a jejich úkolem je podle nastavených pravidel povolit či zahodit síťovou komunikaci. Dříve jednoduché bezestavové systémy mohou dnes být dynamicky konfigurovány, udržují si svůj stav a dovolují relativně komplexní nastavení pravidel.

Pro monitorování síťového provozu se většinou používají systémy IDS (Intrusion Detection System) a IPS (Intrusion Prevention System). Tyto systémy analyzují události a hledají v nich hrozby porušující nastavené bezpečnostní politiky. Zatímco systémy IDS jsou pasivní a případné porušení pouze hlásí pověřeným osobám, IPS systémy umožňují

na nalezené hrozby automaticky reagovat např. nastavením nových pravidel firewallu. Tyto systémy se většinou nasazují na sledování síťového provozu (tzv. network-based), mohou však být nasazeny na stanicích (tzv. host-based) a sledovat tak události nastávající na dané stanici, jako je například vytížení procesoru RAM [20].

IDS a IPS systémy většinou fungují na principu hloubkové analýzy paketů a popisu pravidel - signatur, podle kterých odhalují hrozby podobně, jako antivirové programy [34]. Mohou být schopny detekovat přenášené soubory a odhalovat náhodné útoky na přihlašovací údaje (např. podle přihlašovacího jména guest, které bývá u méně sofistikovaných útoků často zkoušeno). Tyto systémy jsou relativně jednoduché na vytvoření a výkonné, ale trpí již dříve popsanými problémy s detekcí nových typů útoků, pro které zatím nebyly vytvořeny signatury [34].

Jiným typem IDS a IPS systémů jsou ty, které využívají principů behaviorální analýzy a snaží se tak rozpoznat útočníka od legitimního uživatele podle anomálií v chování. Většina systémů hledá tyto anomálie v síťovém toku, ale dají se sledovat i anomálie dat obsažených v hlavičkách paketů [30]. Do tohoto typu systémů spadají i systémy založené na stavové analýze protokolů, která používá modely chování jednotlivých protokolů specifikovaných tvůrci těchto protokolů a hlásí události v případě použití protokolu jiným způsobem. Velkým problémem IDS systémů fungujících na principu behaviorální analýzy je velké množství tzv. false positives, tedy upozornění na podezřelou aktivitu, která je však zcela nezávadná.

Vzhledem k velkému množství zdrojů bezpečnostních informací, jako jsou různé logy (systémové, aplikační) rozličných hlášení z firewallů a IDS/IPS, se ukázalo nezbytné zavést systém, který by je dokázal shromažďovat na jednom místě, agregovat a umožnit bezpečnostním analytikům zjednodušený pohled na celou síť. Takovým systémem je Security Information and Event Management (SIEM), který vznikl spojením Security Event Managementu (SEM), který se staral o shromažďování logů a jejich analýzu a Security Information Managementu (SIM), který analyzoval trendy a poskytoval analytikům vyšší abstrakci [14]. Shromažďováním těchto informací na jednom místě a jejich korelací se dají rozpoznat vzory, které se odlišují od běžného provozu a tak rychle identifikovat, analyzovat a reagovat na bezpečnostní incidenty. Přestože v teorii by měly být tyto systémy schopny detekovat hrozby téměř v reálném čase i v rozsáhlých sítích, v praxi to selhává kvůli velkému množství dat, se kterými musí pracovat. Navíc tyto systémy nedetekují útoky, které se maskují v běžném provozu a nejsou tedy zachyceny žádnou sondou a proto se neobjeví v logu [15].

Kapitola 3

Návrh systému pro rozpoznání APT útoku

V rámci této kapitoly je navržen systém, který je po implementaci v cílovém prostředí schopen detekovat pokročilé ICT útoky včetně APT útoků.

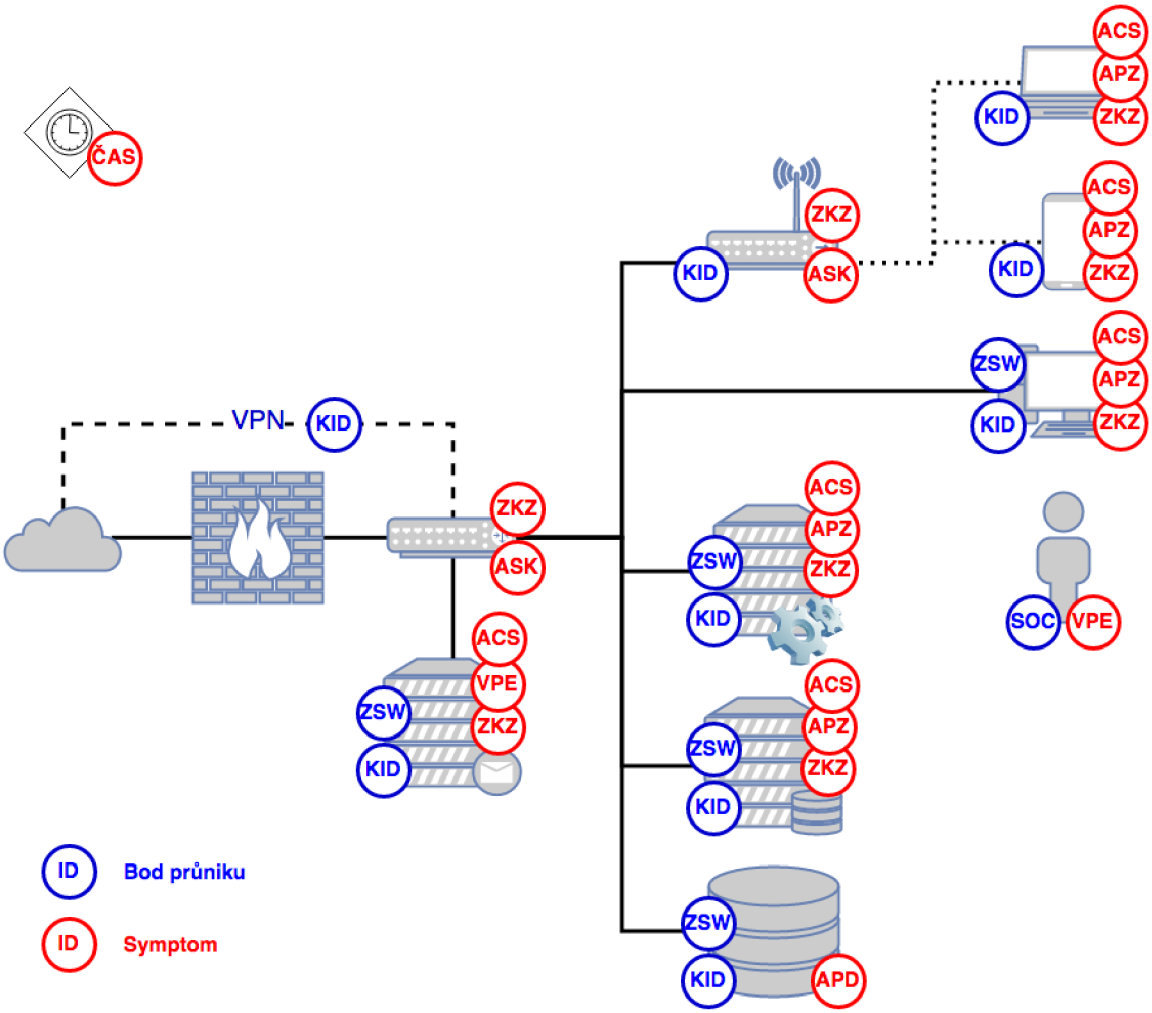
Nejprve je třeba identifikovat symptomy, které provází APT útoky, abychom byli schopni rozhodnout, které oblasti je třeba sledovat a pomocí čeho lze detekovat, jestli se organizace nachází pod APT útokem.

Symptomem se v práci rozumí příznak či průvodní jev obtížně pozorovatelného děje, stavu nebo procesu, který slouží k rozpoznání útoku. U každého nalezeného symptomu je dále rozebráno, jak jej lze detekovat a nakonec je navržen obecný systém, který na základě sledování symptomů určí, zda se organizace nachází pod APT útokem.

3.1 Identifikace symptomů APT útoku

Na obrázku 3.1 je ilustrativně vyobrazeno typické prostředí organizace, ve kterém hrozí APT útok, s bezpečnostními aktivy, které se v daném prostředí vyskytují. Modrá kolečka identifikují body průniku, které jsou dále přiblíženy v tabulce 3.1, červená pak znázorňují výskyt symptomů, podle kterých lze rozpoznat APT útok. Přehled symptomů je pak uveden v tabulce 3.2. V následujícím odstavci je popsán obrázek z pohledu bezpečnostních aktiv na něm uvedených.

Vlevo na obrázku 3.1 je uvedena ikona oblaku, která symbolizuje vnější síť, ke které je organizace připojena. Kromě fyzického připojení firewallu lze pozorovat virtuální VPN spojení, které směřuje přímo do vnitřní sítě. Toto VPN spojení je podstatné, protože obchází firewall, kterým proteče jako šifrovaný kanál a v poslední době bývají VPN připojení velmi častým bodem průniku APT útoků [13]. Pro zobrazení demilitarizované zóny (DMZ) byl zvolen třícestný firewall kvůli jednoduchosti jeho znázornění, přestože v praxi je DMZ obvykle řešena pomocí dvou firewallů. V DMZ je umístěn poštovní server, který reprezentuje různé servery nacházející se v DMZ. Velmi často využívají útočníci pro získání přístupu do sítě spear phishing[9], který přes tento server proudí, a pokud se útočníkům podaří kompromitovat poštovní server, mohou číst nešifrovanou firemní komunikaci. Směrem do vnitřní sítě je umístěn switch, který reprezentuje síťové prvky a na který směřuje VPN, která je tím zavedena přímo do vnitřní sítě. Do switche jsou pak zapojena další zařízení nacházející se ve vnitřní síti. Nahoře na obrázku je znázorněn bezdrátový přístupový bod (AP), pod ním běžná pracovní stanice, aplikační server, databázový server a datové úložiště. AP jsou dále



Obrázek 3.1: Typické prostředí APT útoků

žitá místa v síti, protože umožňují bezdrátový přístup do vnitřní sítě, který mohou útočníci zneužít. Významným problémem jsou v podnikových sítích zejména nepovolené AP, které si uživatelé nainstalují sami bez vědomí síťového administrátora. Aplikační servery obsahují většinou podpůrné nástroje pro fungování organizace. Jejich napadením může útočník získat přístup k datům se kterými aplikace pracuje a také může sabotovat organizaci pomocí odepření služeb daného serveru, nebo dezinformacemi které danou aplikací šíří. Databázový server se stará o správu veškerých dat a jeho napadením získá útočník moc nad všemi daty, která jsou obsažena v celé databázi. Přímým přístupem do úložiště dat získá útočník veškerá surová data organizace, některá data je ale nutné správně interpretovat, což nemusí být triviální (např. datové soubory databází, vnitřní soubory aplikací). V nejpravějším sloupci jsou zobrazena zařízení, se kterými přímo pracuje uživatel, který je reprezentován ikonou nejníže. Uživatel sám o sobě může být jak terčem útoku (např. již zmíněný phishing nebo sociální inženýrství), tak i zdrojem útoku. Nezanedbatelná část útoků je vedena z vnitřní sítě pomocí zaměstnanců, kteří vědomě z osobních důvodů, např. kvůli vydírání či finanční odměně, kompromitují systémy a data zaměstnavatele [27]. S uživatelem je spojena pevná pracovní stanice, mobilní zařízení a přenosný počítač, které jsou na obrázku znázorněny nad ikonou uživatele. S pevnou pracovní stanicí uživatel v kanceláři pracuje nejčastěji, tato

stanice je zpravidla pod správou IT oddělení organizace, přesto může být napadena a využita pro potřeby útočníka. Přenosná zařízení představují vyšší formu rizika, jelikož jsou často pod správou jednotlivých uživatelů a mohou být snáze odcizena. Při odcizení či zneužití mobilního telefonu získá útočník relativně málo hodnotných údajů, může však zneužít zařízení pro ukradení identity uživatele a přihlášení se do podnikové sítě. U přenosných počítačů je rizikem kromě krádeže identity i krádež dat na daném zařízení, opět však platí, že není tolik důležitá jako krádež identity. Pokud útočník získá přenosný počítač s důvěrnými daty, jedná se většinou o několik málo dokumentů, jejichž zcizení může pro organizaci představovat citelnou finanční ztrátu, krádež identity ale může útočníka oprávnit ke všem důvěrným datům v organizaci, což je likvidační.

V dalších sekcích jsou blíže popsány jednotlivé symptomy, které indikují, že je organizace pod APT útokem.

identifikátor	**název**	**popis**
KID	Krádež identity	Ukradené přihlašovací údaje, sezení, či jiný způsob vydávání se za legitimního uživatele.
SOC	Sociální inženýrství	Zneužití lidského faktoru k získání citlivých informací, popřípadě přímo k vykonání záškodnické akce.
ZSW	Zneužití software	Využití zranitelností v software pro získání přístupu či informací z napadeného prostředí.

Tabulka 3.1: Přehled bodů průniku při APT útoku

identifikátor	**název**	**popis**
ACS	Abnormální chování software	Přístup na jiná místa v paměti, data, nebo vytěžování systému.
APD	Abnormální přístup k datům	Přístup na nezvyklá data, přístup mimo běžné cesty.
APZ	Abnormální použití zařízení	Podezřelá aktivita zařízení (přihlašování, vytížení, paměť).
ASK	Abnormální síťová komunikace	Změna v síťovém provozu uživatele nebo jeho trendech.
ČAS	Dlouhodobý průběh	Délka přítomnosti identifikovaného symptomu v prostředí.
VPE	Výskyt phishingových e-mailů	Kvalitní podvodné e-maily jsou často součástí prvních fází APT útoku.
ZKZ	Změna konfigurace zařízení	Jakákoli změna nastavení spravovaných zařízení.

Tabulka 3.2: Přehled symptomů APT útoku

3.1.1 Abnormální chování software

Pro získání přístupu do prostředí mohou být využity známé, ale i dosud neobjevené (takzvané zero-day) zranitelnosti v software, a to jak v aplikačním, tak i v serverovém software, či dokonce v nějaké součásti operačního systému. Nejčastěji dochází k napadení pomocí speciálního uživatelského vstupu, který není řádně ošetřen. Útočník pak může software využít

například pro eskalaci svých práv a k získání kontroly nad cílovým zařízením [9].

Kromě využití zranitelností pro přímý průnik lze zneužít software také ke sběru informací, kdy je útočník schopen přinutit software, aby mu zpřístupnil informace, na které nemá právo, nebo jej může dokonce útočník modifikovat tak, aby tyto informace sám sbíral a útočníkovi předal (například sledování bankovních údajů zadávaných ve webovém prohlížeči). Pokud se útočníkovi podaří upravit aplikaci tak, aby přijmala příkazy, které jí nějakým způsobem doručí, zajistí si útočník trvalý přístup k cílovému zařízení a může jej použít pro další průniky.

3.1.2 Abnormální přístup k datům

V dnešním informačním světě představují firemní data hodnotu, a proto se stávají terčem APT útoků téměř vždy. APT útoky jejichž cílem není krádež citlivých dat jsou spíše výjimkou. Získání citlivých údajů poskytuje útočníkovi kromě kompetitivní výhody také nové informace o cíli, které umožňují přesnější postup při dalším pokračování útoku.

Vzhledem k důležitosti dat je nutné věnovat vyšší úsilí kontrole přístupu k nim a detekovat nejen pokusy o přístup k tajným datům, ale kontrolovat veškeré operace s těmito daty. Je důležité od sebe oddělit tajná data od dat veřejných a to jak logicky, tak i fyzicky. Tajná data by neměla nikdy opustit vnitřní síť organizace.

3.1.3 Abnormální použití zařízení

Po získání přístupu k zařízení na něm útočník zpravidla provádí operace odlišné od běžného chování uživatele. Velmi často se útočník pokouší převzít kontrolu nad zařízením, aby získal přístup ke všem informacím dostupným na daném zařízení a dalších zařízení, které napadenému zařízení důvěřují. Proto se na zařízení objevují pokusy o eskalaci práv[31], kterého je zpravidla dosaženo zneužitím zranitelnosti v software, jak bylo zmíněno výše.

Abnormální použití zařízení lze rozpoznat změnami v přihlašování uživatele. Útočník nepřistupuje k zařízení stejným způsobem a lze tedy detekovat vzdálené přihlašování z neobvyklých adres, nebo dvojí přihlášení uživatele - lokálně a vzdáleně. Kromě způsobu přihlašování lze také pozorovat změny v době přihlášení, pokud je zařízení aktivní mimo standardní časový rámec, lze indikovat pravděpodobné napadení zařízení.

Pokročilé APT útoky se však snaží skrýt své aktivity v běžném provozu zařízení a nezpůsobují tak výrazné změny, jako je připojení k zařízení v nestandardní době. Přes to však lze detekovat více menších symptomů, včetně vyššího vytížení zařízení, zaplňování paměti nebo zaplňování pevného disku.

3.1.4 Abnormální síťová komunikace

Naprostá většina útoků využívá toho, že jsou dnes počítače propojeny internetovou sítí. Ta poskytuje útočníkům způsob, jak vynést informace z napadené organizace a také, jak vzdáleně útok řídit. Po prvotní kompromitaci systému je většinou instalován v cílovém prostředí software umožňující vzdálené ovládání a tento software se přihlásí do útočníkovy řídící sítě [1][19]. V poslední době také často dochází ke zneužití stávající VPN linky organizace[13], což umožní útočníkům nerušený a šifrovaný přístup do vnitřní sítě.

Útočník z počátku často nezná strukturu sítě v prostředí a tak je nucen ji nějakým způsobem odhalit, což se často skládá z různého skenování sítě. Skenování portů jako takové je pro APT útoky méně časté, jelikož je velice snadno odhalitelné. Pokud je však rozloženo v dostatečném časovém rozpětí, může klasické detekci uniknout. Protože je u APT útoků

kladen velký důraz na skrytí všech aktivit, je obvykle komunikace s řídící sítí útočníka šifrovaná a odesílání nasbíraných dat probíhá po malých částech během delšího období, čímž se skryje v normální komunikaci [19].

Je tedy zřejmé, že sledování síťového provozu a jeho analyzování je zásadní a nedílnou součástní jakéhokoli systému, který se zabývá detekcí moderních útoků v ICT prostředí.

3.1.5 Dlouhodobý průběh

Již v názvu APT je zmíněna perzistence a v kapitole 2.1 je popsáno, že se tyto útoky vyznačují dlouhou dobou trvání. Přes to, že se rozhodně jedná o určující prvek APT útoku, není samostatně měřitelný. Dlouhodobou povahu útoku lze detekovat pouze zpětně a je měřena pomocí ostatních symptomů. Pokud tedy výskyt symptomů indikuje podezření na APT útok, lze až zpětně pomocí analýzy zjistit délku trvání daného symptomu a podle ní indikovat, že se s pravděpodobností jedná o APT útok. Dlouhodobý průběh tedy můžeme chápat nikoli jako samostatný symptom APT útoku, ale jako charakteristiku u ostatních symptomů.

3.1.6 Výskyt phishingových e-mailů

Přesto, že podvodné e-maily nejsou specifikem pouze APT útoků a jsou jimi dnes zahlceny téměř všechny e-mailové schránky, velice se liší v důmyslnosti a zůstávají nejčastěji využívaným bodem pro průnik do cílového prostředí [1]. Pokročilé útoky stylu APT používají velice kvalitně vypadající podvodné e-maily, které jsou téměř nerozpoznatelné od regulérní pošty. Tento typ podvodných e-mailů se nazývá *spear phishing*[10] a útočník pro jejich vytvoření používá často detailní informace o napadeném prostředí a uživateli. Tato pošta pak snadno projde automatickými filtry a uživatel, který nepozná rozdíl od podnikové pošty, může nevědomky nainstalovat malware, nebo je přesměrován na podvodný web, který útočník nastražil.

3.1.7 Změna konfigurace zařízení

V podnikovém prostředí bývají zařízení pod správou IT oddělení, v poslední době se však rozmáhá princip BYOD (Bring Your Own Device). U zařízení, které spravuje IT oddělení, se předpokládá stabilní konfigurace, která se nemění. Jakákoli změna, jako je například instalace nového programu nebo otevření portu, indikuje narušení bezpečnosti. U pracovních stanic se může jednat o běžný virus, ale pokud se změní konfigurace síťových prvků, dá se předpokládat pokročilejší útok.

Po napadení zařízení často útočník instaluje backdoor[1], který mu umožní vzdálený přístup k systému, nebo modifikuje stávající aplikaci, aby plnila tento účel[9]. Tyto změny se mohou projevit naslouchánímna novém síťovém portu, nebo přesměrováním běžné komunikace. Kromě pasivního čekání na pokyny dochází u těchto typů malware ke kontaktování útočníka a předání informace o úspěšné kompromitaci systému [19].

Útočník však nemusí napadené zařízení kompromitovat pouze kvůli přístupu do sítě. Napadené zařízení může také pro útočníka po kompromitaci autonomně sbírat informace. Například má-li napadené zařízení vhodnou síťovou kartu, může sledovat veškerý provoz proudící přes ni i pokud není určen pro dané zařízení [8]. V případě bezdrátové sítě se pak jedná o veškerý provoz v okolí. Jiným příkladem je sbírání informací z okolí napadeného zařízení pomocí jeho senzorů, jako je mikrofon, kamera nebo i GPS.

Kromě softwarové změny konfigurace je možná i změna konfigurace HW. Útočník může oběti nainstalovat přídavný hardware, nebo nahradit stávající tak, aby nepozorovaně plnil i jinou funkci. Nejznámějším využitím změny HW konfigurace je instalace keyloggeru, který zaznamenává veškeré stisky klávesnice a útočník je schopen pomocí něj zjistit hesla a jiné citlivé údaje o oběti.

3.2 Možnosti detekce jednotlivých symptomů

Jak vyplývá z kapitoly 2 je detekce APT útoků značně komplexní problém vzhledem k tomu, že jsou APT útoky vedeny profesionálně odborníky a s důrazem na skrývání svých aktivit. Přes to, že APT útoky lze teoreticky detekovat stejnými způsoby jako jakékoli jiné ICT útoky, profesionálně vedené APT útoky často zůstávají pod rozlišovacími schopnostmi stávajících bezpečnostních řešení. APT útoky nelze detekovat jednoduchým systémem či zařízením, které by stačilo přidat do stávajícího systému v organizaci[19] (jako např. IDS), pro umožnění detekce tohoto typu útoků je nutné zahrnout bezpečnost již do návrhu struktury systémů organizace. Pro identifikaci kritických míst je vhodné použít analýzu rizik, jejímž výstupem je seřazení zkoumaných prvků (aktiv) podle kritičnosti a pravděpodobnosti, že se útočník zaměří právě na toto místo. Návrh struktury s oddělením kritických prvků od méně kritických umožňuje zaměřit se při obraně na důležitá místa a neplýtvat energií a financemi jinde.

Mnoho symptomů sleduje abnormality v použití jednotlivých elementů, ať už se jedná o software, data nebo komunikaci. Sledování těchto informací lze abstrahovat do sledování množiny *trojic použití*, kde trojice použití je definována jako [**Subjekt, Metoda, Objekt**]. *Subjekt* je ta entita, která provádí nějakou akci s *objektem* a *metoda* pak popisuje způsob a typ prováděné akce. V případě přístupu k datům pak může být subjektem uživatel, popřípadě proces vyžadující data, metodou je pak volání nějakého rozhraní, nebo přímý přístup a objektem jsou ovlivněná data.

V následujících sekcích jsou přiblíženy způsoby detekce jednotlivých symptomů.

3.2.1 Abnormální chování software

Pro sledování chování software jsou používány antivirové programy. Tyto programy prochází soubory přítomné na počítači a ověřují podle signatur, nejedná-li se o známý škodlivý software. V tomto základním pojetí jsou tyto programy schopny detekovat pouze ten malware, který je již znám a je pro něj vytvořena signatura a nejsou schopny odhalit nové hrozby a modifikaci či zneužití software.

Pokročilejší antivirové programy jsou schopny detekovat podezřelé chování i podle přístupů aplikace k některým funkcím operačního systému nebo zdrojům, které nejsou běžně používány. Díky tomu jsou schopny rozpoznat i některé hrozby, pro které zatím není vytvořena signatura, pokud je jejich chování při těchto přístupech dostatečně podobné známému vzorci.

Pro detekci změn v software lze využít techniku podepisování, kdy je pro aplikaci spočten její otisk pomocí nějaké hashovací funkce a tento otisk je pak zašifrován soukromým klíčem vydavatele. Pomocí tohoto otisku pak lze detekovat nejen změny v software, protože dojde ke změně otisku, ale pokud máme ověřený veřejný klíč vydavatele, lze ověřit, že nainstalovaný software nebyl dodatečně modifikován. Toto ověřování je rozšířeno zejména u mobilních aplikací a Linuxových repozitářů, u aplikací pro Windows dochází k podepisování důležitých součástí systému, jako jsou ovladače, také. Problémem stále zůstává bezpečná distribuce

veřejných klíčů vydavatelů, která bývá řešena pomocí centralizované správy software, jako jsou Linuxové repozitáře, Google Play store a úložiště ovladačů pro Windows. Pokud je software spravován centrálně, lze jej podepsat pomocí jednotného klíče a uživateli pak stačí mít k dispozici pouze jeden veřejný klíč.

Uzavírání aplikací do kontrolovaného prostředí, tzv. sandboxing, umožňuje kromě omezení přístupu aplikace také detekovat změny v chování. Aplikace je uzavřena ve svém prostředí, kde má přístup k datům a prostředkům, které potřebuje pro svůj běh, a všechny ostatní prostředky jsou jí skryty. Pokud detekujeme pokus aplikace přistoupit k datům mimo toto prostředí, je zde podezření, že byla napadena, protože při standardním použití by k podobným událostem nemělo docházet.

3.2.2 Abnormální přístup k datům

Pro detekci abnormalit přístupu k datům je nutné sledovat veškeré pokusy o načtení dat i jejich změny. Pro sledování těchto údajů však nestačí ukládat si informace v obslužném software, ale je potřeba součinnost operačního systému, který data spravuje. To z toho důvodu, že útočník může kromě přístupu pomocí standardních rozhraní využít i přímého přístupu k datům a zcela tak obejít obslužný software.

Protože sledování všech přístupů k datům je náročné, je potřeba data kategorizovat a oddělit od sebe citlivá data od těch, jejichž případný únik by neznamenal vážné bezpečnostní riziko. Kromě logického oddělení dat podle míry tajnosti je vhodné oddělit tato data i fyzicky. To umožňuje lepší správu přístupu k tajným datům a nasazení jiných politik pro přístupový systém. Pokud budou tajná data přístupná pouze přes dedikovaný přístupový server, je snažší nasadit na tento server bezpečný systém, který bude uchovávat větší množství informací o veškerých přístupech. Naproti tomu data, která jsou veřejná, budou umístěna na jiném zařízení, které nebude zbytečně zatíženo sbíráním informací o přístupu.

Neoprávněné pokusy o přístup na tajná data jsou jednoznačným varováním, že by se mohlo jednat o útok.[9] Kromě zamítnutých přístupů lze také sledovat způsob přístupu na data, jelikož útočníci se mohou často pokusit obejít standartní rozhraní ve snaze vyhnout se obranným mechanismům. Měla by tedy být kontrolována i integrita dat a ověřováno nejen kdo na data přistupuje, ale i jakým způsobem. Také lze detekovat, ke kterým datům uživatel přistupuje a sledovat, jestli se jeho chování nezmění a nezačne číst i data, která standartně nepotřebuje.

Dobrou praktikou je povolit uživateli přístup pouze na ta data, která reálně potřebuje ke své činnosti.

3.2.3 Abnormální použití zařízení

Pro sledování používání zařízení je nutné uchovávat provozní informace o využití zařízení. Pro získávání těchto informací je nutná plná podpora operačního systému, který musí generovat auditní události. V Unixovém prostředí probíhá toto nastavení přes auditního démona *auditd*, v prostředí MS Windows je pak toto nastavení součástí služeb operačního systému pod názvem *Security Auditing*.

Mezi vhodné události, které by měl audit sledovat, patří přihlašování uživatelů. To umožní identifikovat přihlášení nestandartního uživatele, jako je například nepoužívaný účet guest, přihlášení nestandartním způsobem, což může být například vzdálené přihlášení k pracovní stanici, ke které se uživatel vždy přihlašuje lokálně, nebo přihlášení v nestandardní době. Jiným vhodným ukazatelem jsou běžící procesy a jejich nároky na RAM a vytížení CPU, které odhalí, zdali nedošlo ke spouštění neznámých procesů nebo nedošlo ke

kompromitaci stávajících. Audit operačního systému umožňuje generovat i události související s prací se soubory a lze tedy detekovat vytváření, modifikace i čtení souborů.

Pomocí sběru a sledování těchto dat lze vytvořit model popisující běžnou činnost zařízení, který může být pak použit pro porovnávání s aktuálním stavem a rozhodnutí, zdali je aktuální použití zařízení normální či nikoli.

3.2.4 Abnormální síťová komunikace

K detekci abnormální síťové komunikace lze přistoupit ze dvou zcela odlišných směrů. Prvním je sledování jednotlivých komunikačních toků a detekce podezřelého obsahu komunikace (například pokus o navázání spojení na neexistující uzel). Druhým přístupem je sledování chování jednotlivých účastníků, jejich komunikačních partnerů a vzorů chování a detekování podezřelého chování.

Běžné firewally a IDS/IPS systémy používají první způsob. Firewally sledují, kdo se snaží komunikovat a jaký kanál (port) pro to chce využít a na základě sady pravidel rozhoduje zda je komunikace povolená, či nikoli. IDS/IPS systémy často fungují na principu hloubkové analýzy paketů, kdy se snaží zjistit obsah komunikace a reagují na takový obsah, který mají označen jako nežádoucí. Tyto systémy jsou schopny velmi efektivně rozpoznávat známé typy útoků podle komunikačních partnerů (skenování portů, přístupy na neaktivní adresy) nebo podle známého obsahu nebezpečných paketů. Nejsou však schopny rozpoznat novátorské přístupy a skrytí před těmito systémy nepředstavuje pro zkušeného útočníka příliš velký problém.

Druhým způsobem lze odhalit útočníka podle chování, které nějakým způsobem vybočuje z normálního vzoru komunikace. Sledujeme-li chování jednoho uživatele v síti a jsme-li schopni vysledovat vzory v jeho komunikaci, lze rozeznávat změny v těchto vzorech nebo detekovat podezřelé aktivity podle známých vzorů chování. Systémy založené na detekci chování jsou teoreticky schopny odhalit i zcela nové a zatím neznámé útoky a skrytí před nimi je složité. Proto je tento způsob detekce vhodnější pro APT útoky.

Chceme-li sledovat chování uživatele v síti pro detekci APT útoků, je nutné analyzovat veškerou jeho komunikaci. Pokud sledujeme chování uživatelů, stačí nám analyzovat odchozí komunikaci, podle ní totiž jsme schopni pozorovat akce uživatele, příchozí zprávy jsou většinou pouze reakcí na ty odchozí. Útočníci mohou sice zasílat do sítě příkazy nainstalovanému malware, což je příchozí komunikace, ale tyto příkazy se jen velmi těžko detekují a navíc pokročilý malware používaný při APT útocích může pracovat do značné míry autonomně, bez nutnosti příchozí komunikace. Nejsnadněji lze odhalit odchozí komunikaci po infikaci, kdy se malware přihlašuje do řídící sítě a nebo když malware odesílá útočníkům nasbíraná data.[19]

Útočník však také může obranné mechanismy postupem času naučit komunikaci mezi řídící sítí a malware považovat za normální, pokud dostatečně pomalu rozšiřuje normální chování uživatele o svou komunikaci tak, aby zůstal pod rozlišovací schopností detekčního mechanismu. Proto je vhodné kromě změn v chování na síti sledovat také dlouhodobější trendy například porovnáním aktuálního normálního chování uživatele s tím, jaké bylo před měsícem, či rokem.

3.2.5 Výskyt phishingových e-mailů

Odhalování spear phishingových e-mailů je velmi náročný až téměř nemožný úkol vzhledem k důmyslnosti, se kterou jsou vytvořeny. Standardní součástí každého e-mailového serveru

by měl být antiphishingový filtr. Tyto filtry jsou nejčastěji založeny na metodách strojového učení a rozpoznávají podvodné e-maily podle jejich struktury, zpracování přirozeného jazyka, kontrolou zpětného DNS záznamu zdrojového serveru a vznikají i protokoly pro ověřování autentičnosti e-mailů[4].

Pro napadení zařízení bývají v podvodných e-mailech nejčastěji využívány infikované přílohy, alternativně se útočníci snaží uživatele pomocí falešného odkazu přinutit navštívit podvrženou webovou stránku a malware si nevědomky nainstalovat [10]. Malware připojený k e-mailu má zřídka formu spustitelného souboru, který bývá téměř vždy odhalen, ale častěji dochází k infikování ZIP či RAR archivu nebo souborů běžně používaných v organizaci (jako jsou tabulky v programu Excel nebo PDF soubory)[10]. Mnoho antivirů umožňuje prozkoumat přílohy e-mailů a pokusit se detekovat přítomný malware.

Protože však spear phishingové e-maily často díky své důmyslnosti překonají antiphishingové filtry, nezbývá než upozornit na tato rizika uživatele a před otevřením příloh podezřelých e-mailů vždy ověřovat nezávisle jejich autentičnost a na případné nesrovnalosti upozornit. Pokud se v organizaci vyskytnou kvalitní spear phishingové e-maily, které obsahují firemní informace, je riziko, že se organizace nachází pod APT útokem, velmi vysoké. Pokud se však podaří odhalit APT útok díky detekci phishingových e-mailů, došlo tak v prvních fázích útoku a útok tedy zatím pravděpodobně nezpůsobil velké ztráty.

3.2.6 Změna konfigurace zařízení

Pro sledování změn konfigurace zařízení je nutné vytvořit systém, který bude sledovat vybrané konfigurace a upozorní na jejich změnu. V případě zařízení firmy Cisco s operačním systémem IOS stačí sledovat dva konfigurační soubory - *startup-config* a *running-config*. U počítačů je ale situace komplikovanější, vzhledem k jejich komplexitě totiž neexistuje jednotný konfigurační soubor. Je tedy nutné pomocí nějakého nástroje sledovat důležitá konfigurační nastavení a ověřovat, jestli nedošlo v běžícím systému k dynamické změně konfigurace.

Pro sledování integrity systému lze také použít TPM čip (Trusted Platform Module). Tento čip byl vyvinut jako bezpečnostní modul pro počítače a jednou z jeho funkcí může být i sledování, zda nedošlo k porušení integrity operačního systému [3]. Na základě podpisů jednotlivých konfiguračních souborů a hardwarové konfigurace by tento čip měl být schopen detekovat neoprávněné změny.

3.3 Návrh architektury systému pro detekci útoků

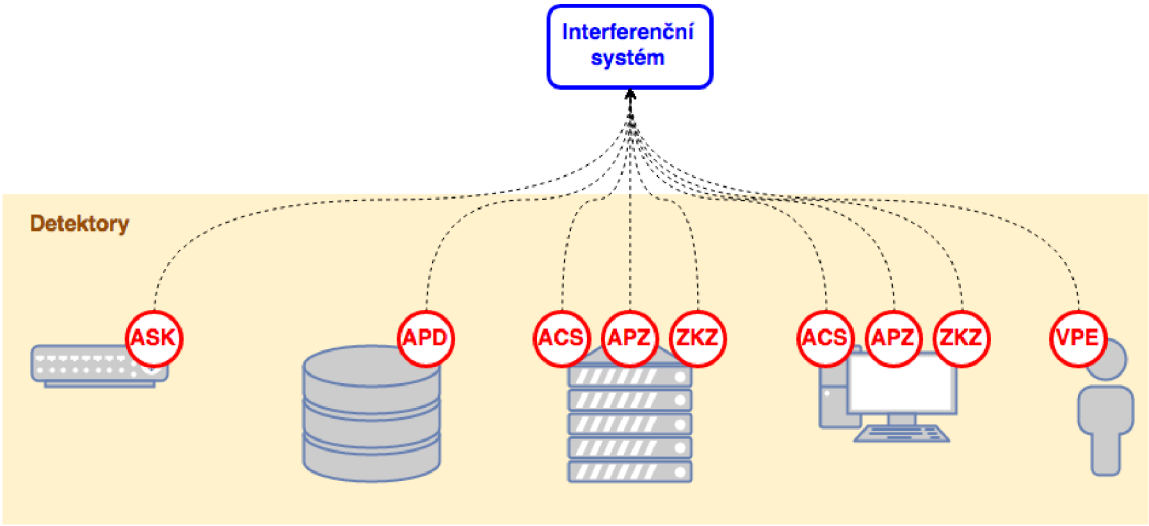
V předchozí kapitole byly navrženy způsoby detekování jednotlivých symptomů. Pokud chceme na základě těchto symptomů usuzovat, zdali je organizace pod APT útokem, je nutné zpracovat informace z detektorů jednotlivých symptomů a pomocí vhodné agregace detekovaných hodnot rozhodnout, s jakou mírou indikují symptomy, že se organizace nachází pod APT útokem.

Jednotlivé detektory ale nepracují s jednotnou metrikou, transformací jejich výstupů na společnou metriku definující míru pravděpodobnosti napadení bychom ztratili příliš mnoho informací, což by se negativně projevilo na citlivosti celého systému. Například detekce abnormálního přístupu na data by mohla využívat jako metriku vektor popisující citlivost dat, které byly tímto ovlivněny, na stupnici o třech úrovních - tajná, podniková a veřejná data a míru abnormality přístupu k nim. Pokud bychom chtěli tyto informace převést na

skalární hodnotu udávající míru napadení, došlo by ke ztrátě informace, která by v kombinaci s jiným symptomem (například detekování abnormální síťové komunikace a odesílání většího objemu dat) mohla být významná. Proto je nutné zpracovávat výstupy jednotlivých detektorů v rámci jejich vlastních metrik.

Na základě informací naměřených na jednotlivých detektorech lze pak pomocí nastavených pravidel vhodného interferenčního mechanismu rozhodnout, zda se organizace pravděpodobně nachází pod APT útokem, či nikoli. Vzhledem k povaze jednotlivých symptomů a mírám nejistoty při jejich kombinacích je nutné využít fuzzy logiku.

Pro vytváření pravidel interferenčního mechanismu je nutné vzít v úvahu fakt, že závažnost jednotlivých symptomů je závislá nejen na výstupu detektoru, ale i na jeho umístění. Pokud je hlášena změna konfigurace u kancelářského počítače, nejedná se o natolik významnou událost, jako když je stejná změna detekována v konfiguraci klíčového síťového prvku. Proto je nutné jednotlivá aktiva na kterých probíhá detekce hierarchicky kategorizovat a reflektovat toto v navrhovaných pravidlech.



Obrázek 3.2: Vysokoúrovňový návrh architektury systému sloužícího pro detekci APT útoků

Na obrázku 3.2 je zobrazena architektura navrhovaného systému, který sestává z množiny detektorů a interferenčního mechanismu. Architektura zahrnuje:

- detektor ASK (abnormální síťová komunikace),
- detektor APD (abnormální přístup k datům),
- detektor ACS (abnormální chování software),
- detektor APZ (abnormální použití zařízení),
- detektor ZKZ (změna konfigurace zařízení),
- detekce VPE (výskyt phishingových e-mailů),

které jsou vzájemně provázány interferenčním systémem. Interferenční systém na základě informací z detektorů pomocí fuzzy logiky rozhoduje, zda došlo k APT útoku.

Podle zadání diplomové práce je dále detailněji rozebrán a navržen detektor symptomu „abnormální síťová komunikace“, jako demonstrace tvorby jednotlivých detektorů.

Kapitola 4

Detekce abnormalit v síťové komunikaci

Jak je uvedeno v kapitole 3.1.4, je sledování síťové komunikaci nutné, chceme-li odhalit moderní útoky na ICT systémy. Z kapitoly 3.2.4 pak dále vyplývá, že pro detekci APT útoků je nevhodnější použít analýzu chování uživatelů v síti.

Tato kapitola se zabývá návrhem, implementací a otestováním detektoru abnormalit v síťové komunikaci. Nejprve je popsána analýza chování se zaměřením na síťové prostředí a její aktuální využívání v praxi. Následuje návrh koncepce detektoru, ve kterém je definováno, jakým způsobem lze sledovat síťovou komunikaci, které údaje z ní je třeba sledovat a jsou zde specifikovány vlastnosti, které má mít navrhovaný detektor. Další kapitola se zabývá návrhem samotného detektoru a jeho částí, je zde detailně popsána struktura detektoru včetně modelu, který detektor využívá pro rozpoznání normálního chování uživatele. Kapitola 4.4 obsahuje bližší informace o implementování základních součástí navrhovaného detektoru a poslední kapitola pak popisuje, co bylo při práci s detektorem zjištěno.

4.1 Behaviorální analýza

Behaviorální analýza popisuje obecně analýzu chování určitého subjektu. V ICT bývá pozorovaným subjektem zpravidla nějaký systém, či uživatel. Pokud je sledovaným subjektem uživatel, dá se behaviorální analýza považovat za dynamický biometrický systém, který se zabývá rozpoznáním vzorů chování v čase.

Diplomová práce je zaměřena na analýzu síťové komunikace, která bývá označována jako NBA (Network Behavior Analysis). NBA slouží k rozpoznávání vzorů v proudech paketů, jimiž je tvořen síťový provoz za účelem identifikace hrozeb. V zásadě existují dva přístupy, jak odhalit podezřelé chování, a to buď rozpoznáváním chování útočníka v síti podle známých způsobů chování útočníků - tzv. signature-based, nebo podle odchylek ve standardním chování sledovaného subjektu - tzv. anomaly-based [20].

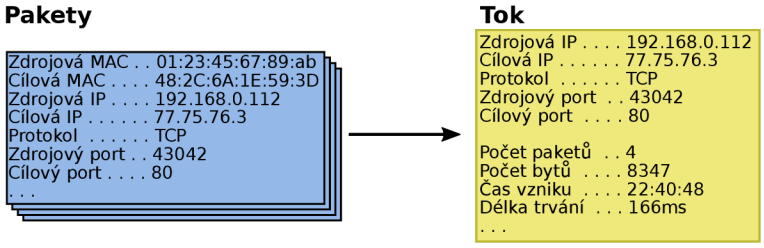
Signature-based analýzy využívají databázi pravidel tzv. signatur, podle kterých rozpoznají přítomnost hrozby. Při sledování chování v síti tedy dochází k porovnání aktuálního provozu s databází signatur a pokud je nalezena shoda, hlásí systém možnou hrozbu. Vzorový příklad: Typicky se útočník po získání přístupu k síti snaží zmapovat služby, které zde běží pomocí skenování portů. To se vyznačuje velkým množstvím krátkých spojení, v databázi signatur tedy bude pravidlo, že pokud se na síti objeví více než x krátkých spojení v krátkém čase, je zde podezření na potenciálně nebezpečnou aktivitu skenování portů a

systém zobrazí hlášení [24]. Tento typ analýzy je v principu relativně snadný, i když signatury mohou být i značně komplexní, a výkonný. Je však limitován známými signaturami, pokud tedy útočník použije novátorský přístup, tento typ analýzy jej nezachytí. Pro vyhnutí se detekci tedy stačí útočníkovi zvolit takový způsob, který není obecně známý, nebo imituje normální komunikaci.

Anomaly-based analýzy využívají profil chování uživatele. K problému vytvoření profilu normálního chování lze přistoupit dvěma způsoby. První možností je manuálně definovat, jak se daný uživatel chová a reálné chování porovnávat s tímto stavem. Manuálně vytvořit tento normální profil je však možné v zásadě jen pro silně omezený a zjednodušený provoz. Příkladem může být nasazení anomaly-based analýzy na sledování pouze jediného protokolu v síťovém provozu. Profil normálního chování pak může být odvozen z definice protokolu a hlášení anomálií potom upozorňuje na nestandartní užití protokolu. Druhou možností je rozdělení funkce systému na dvě fáze - v první fázi se nic nedetekuje a systém se pouze učí jak vypadá normální komunikace a tvoří si její profil, v druhé fázi pak porovnává aktuální provoz s profilem a pokud se výrazněji odlišuje, upozorní na možnou hrozbu.

Systémy většinou fungují tak, že sledují určité charakteristiky a pomocí statistických metod je porovnávají s nastaveným prahem. Mezi vhodné charakteristiky patří například množství přenesených dat v rámci jednoho spojení [15] nebo počty komunikačních partnerů [24]. Příkladem jiné charakteristiky může být podíl webových služeb na síťovém provozu. Pokud je standardně 13% a běžně se liší až o 10%, ale během sledování najednou dosahuje 40%, varuje systém před podezřelou aktivitou. Vzhledem k tomu, že anomaly-based analýzy neznají chování při specifických typech útoků, je varování vždy obecné a nemohou pojmenovat typ podezřelého chování, na rozdíl od signature-based, které mohou pojmenovat chování podle signatury, která mu odpovídá (například zmíněné skenování portů). Anomaly-based systémy jsou vhodné i pro dosud neznámé typy útoků a vyhnutí se detekci je velice obtížné a nutí útočníka sledovat běžnou komunikaci a skrývat v ní své akce velmi složitým způsobem. Problémem tohoto způsobu je ale velké množství hlášení v případech nízkého prahu a při vysokém prahu zase naopak příliš velká tolerance, což dává útočníkovi prostor ke skrytí své aktivity.

Analýza chování poskytuje vyšší míru abstrakce než klasické nástroje založené na hloubkovém analyzování paketů. Místo práce s velkým množstvím jednotlivých paketů pracují systémy pro analýzu chování s datovými toky [25] (označovanými anglicky flow), což je struktura charakterizující skupinu paketů, které mezi sebou mají určitou vazbu. Většinou jsou za jeden tok považovány pakety, které mají společnou zdrojovou a cílovou IP adresu, zdrojový a cílový port a číslo protokolu [24]. Kromě těchto informací obsahuje abstrakce datového toku také čas vzniku, délku trvání, počet přenesených paketů a bytů a může obsahovat i další údaje o daném toku.



Obrázek 4.1: Abstakce skupiny paketů datovým tokem

Podle analýzy chování se dá také do jisté míry rozlišit, jaké akce uživatel zrovna provádí

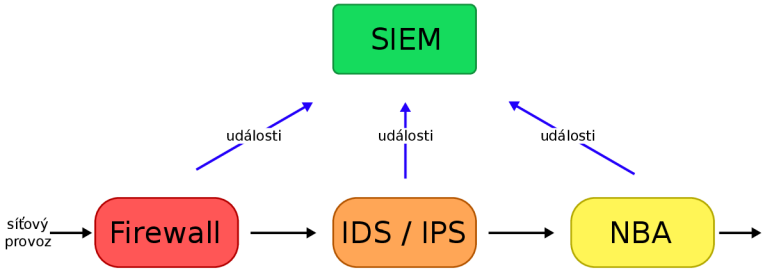
a to bez nutnosti analýzy dat, lze tedy detekovat akce uživatele i při použití šifrované komunikace. Podle metadat síťového toku pak můžeme například určit, jestli uživatel aktuálně prohlíží webové stránky (podle cílové IP by se teoreticky dalo zjistit i jaké), chatuje, nebo třeba odesílá soubory (pochopitelně bez analýzy dat se nedá zjistit jaké) [15].

Velkou výhodou behaviorální analýzy je nekonfliktnost s jinými systémy, ať už bezpečnostními či jinými, na síti. Vzhledem k tomu, že pasivně zpracovává síťové údaje a nijak do komunikace aktivně nezasahuje, lze ji použít v libovolném produkčním prostředí. Navíc pracuje NBA jen s metadaty síťového provozu a proto ji lze využít i pro šifrovanou komunikaci. Pochopitelně použití tunelovaného spojení jako VPN nebo IPsec efektivně skryje vnitřní komunikaci a ta se pro NBA tváří jako jeden tok, to však nebrání detekci nových koncových uzlů tunelů v síti a vyhodnocení vzniku nového tunelu jako potenciální hrozby.

V diplomové práci je pro detektor APT útoků v prostředí počítačové sítě zvolena anomaly-based analýza, která umožňuje zachytit i dosud neznámé typy útoků.

4.1.1 Aktuální využití v praxi

Behaviorální analýza se dnes používá jako doplňková služba k běžným způsobům založeným na ochraně perimetru pomocí firewallů a monitorování sítě pomocí IDS/IPS. Obvyklý způsob zapojení je zobrazen na obrázku 4.2, kde je síťový provoz nejprve filtrován firewallem, poté analyzován běžnými IDS/IPS a následně ještě analyzován pomocí NBA. Hlášení o potenciálních hrozbách je dále agregováno v SIEM systému.



Obrázek 4.2: Umístění NBA rámci struktury síťové bezpečnosti

Komerční nástroje pro zajištění bezpečnosti ve velkých korporacích již obsahují nějaké formy behaviorální analýzy a nebo nabízejí rozšiřující moduly, které tuto funkcionalitu přidávají. Nejčastěji fungují na principu rozeznávání chování útočníka v síťovém provozu, ale objevují se i systémy využívající anomaly-based analýzy. Detailní informace o fungování jednotlivých systémů jsou však kvůli zachování kompetitivní výhody nedostupné a nedá se tedy přesně říci, jakých principů jednotlivé společnosti využívají a co přesně pro detekci hrozeb využívají.

Příkladem může být společnost AdvaICT, která vyvíjela systém pro rozpoznávání útoků podle specifického chování útočníka. Toto chování bylo specifikováno množinou atributů toku spojených logickými podmínkami AND a OR, pokud se pak vyskytlo na síti podobné chování, rozhodoval se systém podle nastavitelných prahů zdali se jedná o hrozbu, či nikoli [24]. Tato společnost byla později skoupena společností INVEA-TECH a její NBA začleněna do produktu FlowMon, který již ale nabízí i detekci hrozeb podle anomálií v chování [7].

Jiným příkladem je McAfee Network Threat Behavior Analysis (NTBA), která je součástí McAfee Network Security Platform. NTBA zřejmě využívá anomaly-based behaviorální analýzu pro deketci hrozeb [12].

V současnosti jsou systémy využívající NBA dostupné v zásadě jen velkým společnostem a jsou považovány za nadstandardní vybavení. Vzhledem k tomu, že se jedná o technologii uvedenou do praxe relativně nedávno a zatím ne zcela spolehlivou, používá se vždy jen jako doplňek ke stávajícím technologiím založeným na signaturách a hloubkové analýze paketů. Do budoucna se však předpokládá velký vývoj technologií založených na analýze chování a zaujmutí až 80% trhu pro detekci útoků v síťovém provozu na úkor analýzy obsahu paketů [25]. Proto se v diplomové práci zabývám vytvořením detektoru anomálií provozu v počítačových sítích, který využívá anomaly-based analýzu obdobně jako podle dostupných informací činí NTBA.

4.2 Návrh koncepce detektoru

Z předchozích kapitol vychází, že nejvhodnějším způsobem detekce tohoto typu útoků je sledování síťové aktivity a detekce odchylek v běžném provozu. Je tedy třeba definovat, jakým způsobem bude síťová aktivita sledována, jak bude vytvářen model chování v síti a na základě čeho bude rozhodováno o tom, zdali je sledovaná aktivita abnormální.

K analýze chování lze přistoupit dvěma způsoby, a to buď rozpoznáváním chování útočníka v síťovém provozu, nebo ve sledování normálního provozu a hlášením anomálií. Druhý přístup je ze své podstaty mnohem obecnější a pokryje tak velkou škálu hrozeb včetně těch, které zatím neznáme a neumíme popsat. Proto je mnohem vhodnější pro odhalování APT útoků které velmi často využívají všech dostupných prostředků včetně na míru sestavených řešení k dosažení cíle. Protože jedním z význačných rysů APT útoků je, že probíhají v delším časovém období a snaží se o minimální možnost detekce, je třeba sledovat i relativně malé odchylky v síťovém provozu. To s sebou však nese problém jak zabránit, aby systém nevykazoval příliš velké množství potenciálních hrozeb a nezahlcoval tak bezpečnostní analytiky velkým množstvím převážně false-positive hlášeními.

V NBA nástrojích bývá síťový provoz reprezentován jako množina síťových toků. Profil chování může být tvořen pro síť jako celek, výsledný profil je pak však příliš obecný a tím poskytuje prostor pro skrytí útočných aktivit. Mnohem vhodnější je vytvářet profily chování jednotlivých zařízení, jelikož chování zařízení lze modelovat mnohem přesněji. Pro rozlišení zařízení se většinou používá zdrojová IP adresa, ta však není příliš spolehlivá. Vzhledem k rozšířenosti DHCP může docházet k tomu, že stejná IP adresa bude odpovídat různým zařízením s odlišnými profily chování [15]. Vhodnější identifikátor pro rozlišení jednotlivých zařízení je linková MAC adresa, při použití MAC adresy je však třeba mít na paměti, že při cestě v síti nezůstává konstantní a při přechodu paketů routery se mění. Pro použití MAC adresy je tedy nutné získávat síťová data ze všech segmentů sítě.

Abstrakce síťového provozu pomocí datových toků je sice vhodnější než množiny paketů, pro lepší analýzu chování by však bylo vhodné dále seskupit tyto toky podle uživatelů jimž náleží [15]. Vytvářením profilu pro dvojici [uživatel, zařízení] by bylo možné dále zpřesnit profil chování a tím detekovat i menší odchylky, navíc je z bezpečnostního pohledu vhodnější identifikovat přímo uživatele zodpovědného za vzniklou hrozbu. Přiřazení toku uživateli je však komplikovaný úkol, neboť se nedá vyřešit pouze na základě síťových dat. Je nutná kooperace se systémem pro zajištění správy identit a účtování a zjistit, který uživatel v daném okamžiku využívá které zařízení. Příkladem takového systému může být RADIUS server ve spojení s LDAP či NIS. Pokud se v síti využívá kontrola přístupu k síti například pomocí IEEE 802.1X, měla by být identita uživatele dohledatelná. Jiné proprietární řešení pro rozšíření bezpečnosti o určení uživatele poskytuje společnost Cyberoam s technologií označovanou jako Identity-based security, která rozšiřuje síťový model ISO/OSI o novou

vrstvu L8 s identifikací uživatele [6].

4.2.1 Sběr dat

Ke sbírání dat z internetového provozu lze použít sondy, které sledují datový provoz a zasílají o něm informace pro další zpracování. Tyto informace mohou sestávat z kompletního přepisu síťového provozu, nebo z nějaké abstrakce nad ním.

Surová data Pro kompletní záznamy síťového provozu se používá najčastěji formát *pcap*, což je zkrácenina z **p**acket **cap**ture. Tento formát obsahuje pakety, které se na síti vyskytly, včetně veškerých dat.

Datové toky Nejčastější abstrakce nad datovým provozem jsou tzv. datové toky popsané v kapitole 4.1. V rámci toku jsou k dispozici statistické informace o reprezentovaných paketech, ale nejsou k dispozici data v nich obsažená. Tyto informace mohou poskytovat sondy pomocí IPFIX protokolu, který je určen právě pro přenos informací o tocích a specifikovaný organizací IETF. IPFIX vychází z NetFlow verze 9, což je proprietární řešení firmy Cisco, a sjednocuje přenos těchto informací ze zařízení různých výrobců.

Vzhledem k tomu, že analýza chování nepotřebuje přístup k datům v paketech a pracuje se statistickými metodami, je mnohem vhodnější získávat informace o tocích. Pokud však chceme identifikovat zařízení podle MAC adres, nemůžeme použít standardní definici toků, jelikož ta zachycuje L3 vrstvu a vyšší. Až NetFlow verze 9 a z něj vycházející IPFIX umožňují pomocí šablon definovat obsah informací o toku a tím získat i MAC adresy účastníků.

4.2.2 Identifikace zařízení

V systémech pro sledování síťových dat se k identifikaci zařízení v naprosté většině případů používá IP adresa. Výhodou IP adresy je její dobrá dostupnost, protože je přítomna v každém IP paketu a v rámci sítě je unikátní. Nevýhodou je však snadná zaměnitelnost a nestálost. S rozvojem mobilních zařízení dochází k tomu, že se do sítě přihlašují uživatelská zařízení dynamicky a aby mohly komunikovat bývá jim přiřazena IP adresa pomocí DHCP protokolu, v případě IPv6 si pak mohou IP adresu dokonce sami vygenerovat. IP adresa nám tedy sice identifikuje unikátně jedno zařízení, ale pouze během jednoho sezení. Budeme-li identifikovat zařízení podle IP adresy a vytvoříme pro něj model chování, může se stát, že při příštím sezení bude tato IP adresa přidělena jinému zařízení a model chování nebude odpovídat. Chceme-li tedy použít IP adresu pro identifikaci, musíme zajistit, aby každé zařízení mělo svou pevnou a neměnnou IP adresu.

Jinou možností je využít MAC adresy, což je fyzická adresa síťového rozhraní, která je zcela unikátní pro zařízení a nezávislá na síťové vrstvě a použitých protokolech. Použití této adresy řeší problémy identifikace přenosných zařízení, přináší však problém dostupnosti. Zatímco IP adresa je při komunikaci v rámci sítě zpravidla konstantní a k přepisu dochází pouze na vnějším rozhraní sítě pokud je použit NAT, MAC adresa je přepsána nejbližším síťovým prvkem. Chceme-li tedy použít MAC adresu pro identifikaci, je nutné sbírat data ve všech segmentech sítě.

Kromě standardních síťových identifikátorů lze využít skutečnosti, že zařízení v podnikové síti bývají autentizována. Pokud síť vyžaduje autentizaci podle standardu IEEE 802.1X, je zařízení ověřováno vůči autentizačnímu serveru. Autentizační server tedy má

přístup k MAC adrese identifikující zařízení a může mít přístup i k identitě uživatele, pokud má každý uživatel unikátní přístupové údaje. Identifikaci zařízení by tedy bylo možné získat dotazem na autentizační server, problém však nastává v otázce mapování síťového toku na záznam uložený v autentizačním serveru. U IPv4 adres by bylo možné, pokud by autentizační server fungoval zároveň jako DHCP, vytvářet trojice: [uživatel, MAC zařízení, přidělená IP], díky kterým by se zařízení dalo pomocí IP adresy jednoznačně identifikovat. Autentizační server by ale musel udržovat historii sezení, aby bylo možné zpětně ověřit tyto hodnoty při pozdějším zpracování. Při použití IPv6 však má zařízení více IP adres a některé si volí samo a toto mapování pro něj není vhodné.

Tato práce používá pro identifikaci MAC adresu spolu s IP adresou. Díky této kombinaci lze využít jednoznačné identifikaci zařízení pomocí zcela unikátní MAC adresy a rozpoznat síťové prvky od koncových zařízení podle toho, že se na jednom fyzickém zařízení nachází více IPv4 adres.

4.2.3 Klíčové charakteristiky

Pro sledování chování síťového provozu je třeba definovat dvojí typ charakteristik. Prvním typem jsou definující charakteristiky, které definují daný síťový provoz a umožňují jej kategorizovat. Druhým typem jsou pak statistické charakteristiky, pomocí kterých lze popsat dění v síťovém provozu a na kterých lze detekovat anomálie.

Definující charakteristiky Pokud chceme vytvořit co nejpřesnější profil chování, je vhodné jej specifikovat pro dvojici [zařízení, uživatel], jak bylo zmíněno dříve v rámci této kapitoly. Zařízení budeme identifikovat pomocí dvojice *[MAC adresa, IP adresa]*, *identitu uživatele* je pak nutné získat externě. Dále je vhodné rozdělit profil podle užitého *protokolu* transportní vrstvy a u nejčastějších protokolů TCP a UDP lze dále sledovat chování jednotlivých aplikací podle *cílového portu*. Sledováním cílového portu lze rozpoznat ke které službě klient při odesílání požadavku přistupuje, u serveru pozbývá tato informace smysl, protože klientské aplikace často volí své porty náhodně. Problémem při sledování aktivit aplikací dle cílových portů jsou zejména P2P aplikace, které navazují četná spojení s mnoha protistranami na náhodných portech. V rámci jedné aplikace je pak třeba rozlišovat mezi protistranami se kterými probíhá komunikace, což se dá sledovat podle *cílové IP adresy* (MAC adresu nelze použít při spojeních mezi různými segmenty sítě).

Každá unikátní MAC adresa koncového zařízení, které se vyskytuje v síti, by měla mít svůj profil v modelu. Pokud bychom neodlišili koncové zařízení od síťových prvků, docházelo by k tomu, že by model chování pro routery oddělující segmenty sítě zahrnoval komunikaci celého segmentu sítě a nikoli pouze daného zařízení.

Statistické charakteristiky U definujících charakteristik lze sledovat jejich *rozptyl* a hlásit, pokud nastane nějaký výkyv. Například každý výskyt nového zařízení, uživatele či použití nového protokolu transportní vrstvy by měl být zcela jistě detekován, protože se jedná o značně konstantní prvky. U portů a IP adres však může docházet k používání unikátních či náhodných hodnot a tím podstatně širšímu rozptylu. U portů to bude nastávat při sledování odchozí komunikace serverů, protože klienti většinou využívají náhodné porty. U IP adres může být příkladem vyhledávání na webu, během kterého uživatel navštíví různé IP adresy včetně těch, na které ještě nikdy nepřistoupil. I zde se však není rozložení zcela uniformní a dájí se rozlišit shluky rozsahů s vyšší pravděpodobností přístupu. V rámci jednotlivých toků pak lze sledovat jejich *četnost* a rozptyl v čase a to podle *doby zahájení*

a *trvání*. Pro detekci přenosu většího množství dat je pak též vhodné sledovat *velikost přenesených dat* v rámci jednotlivých toků.

Šifrovaná spojení

Nejčastěji používané je šifrování na aplikační úrovni a jsou tedy šifrovaná pouze data, veškeré charakteristiky popsané výše jsou dostupné a pro detekci chování tedy nepředstavují žádný problém. Při použití šifrování na úrovni síťové, jako je například IPsec, jsou šifrované veškeré vyšší vrstvy včetně transportní a nedá se tedy rozlišit mezi aplikacemi. Je tedy nutné tato šifrovaná spojení sledovat zvlášť jako speciální typ aplikace.

4.2.4 Obecná specifikace detektoru

Navrhovaný systém pro rozpoznávání APT útoků sestává ze síťových sond, které sledují síťový provoz a odesílají o něm informace na server, kde dochází k porovnání aktuálního provozu s uloženým modelem.

Jako sondy lze využít dedikovaná zařízení, stejnou funkcionalitu však nabízí i některé síťové prvky jako routery a switche. Pokud použijeme sondu, která podporuje agregaci síťového provozu do toků, je nutné využít taková zařízení, která statistiky o tocích vytváří ze všech paketů, a ne zařízení, která příchozí pakety vzorkují a zakládají statistiky na těchto vzorcích, protože ty pak nejsou přesnou statistikou provozu (je to způsobeno tím, že zahrnují pouze vzorky s danou vzorkovací frekvencí) [23]. Při použití sond bez možností agregace do toků je nutné před dalším zpracováváním tuto agregaci provést na straně serveru.

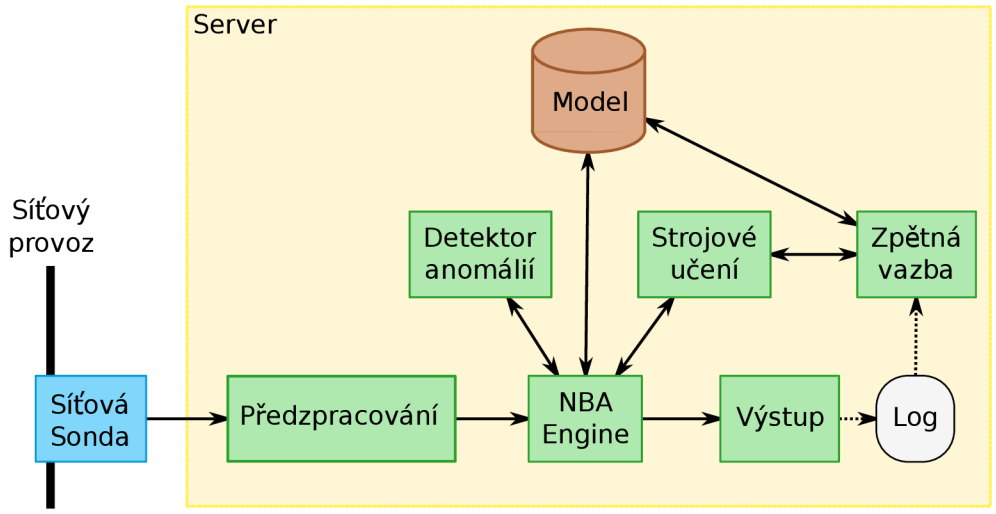
Serverová část se stará o zpracování dat ze sond v reálném čase - tedy tak, jak ze sond přijdou bez zbytečných prodlev. Přijatá data jsou klasifikována dle definujících charakteristik, které určují profil chování. Při nalezení odpovídajícího profilu lze další chování serveru rozlišit do dvou fází podle stavu profilu. Pokud je profil nekompletní, tedy při uvedení systému do provozu nebo přidání nového prvku (nová n-tice definujících charekteristik), použijí se data k vytvoření profilu chování daného prvku. Přidání nového prvku tedy neovlivní stávající profily, ale pouze vedou k vytvoření nového profilu pro daný prvek, který se nachází ve fázi vytváření profilu. Po uběhnutí určitého časového období je profil považován za kompletní a práce s ním přechází do fáze dvě, kde se detekují odchylky. Pomocí klasifikátoru je určena míra podobnosti dat ze sond s uloženým profilem. Pokud je tato míra podobnosti větší než nastavený práh, jsou data ze sondy použita pro aktualizaci profilu, je-li však tato hodnota nižší než práh, systém vyhodnotí chování jako podezřelé a zobrazí upozornění. Zobrazené upozornění musí obsahovat informace o zaznamenané odchylce a uživatel musí mít možnost toto chování označit za bezproblémové a rozšířit o ně profil.

Systém musí ukládat profily sítě permanentně, tedy tak, aby při restartu nedošlo k jejich ztrátě. Při startu systému pak musí být načteny všechny existující profily tak, aby se s nimi dalo pracovat stejně jako před ukončením běhu. Pokud by tento požadavek nebyl splněn, stačilo by útočníkovi způsobit restart systému a při vytváření nových profilů by již byl brán útočník jako normální provoz, což je nepřijatelné.

4.3 Návrh softwarové architektury

4.3.1 Popis komponent

Na obrázku 4.3 je uveden vysokoúrovňový návrh architektury rozlišující sondu sloužící pro získávání dat a serverovou část aplikace sloužící ke zpracování dat včetně výstupu.



Obrázek 4.3: Přehled architektury systému

Následující popis objasňuje účel jednotlivých modulů a komunikaci mezi nimi.

Síťová sonda Tato sonda je umístěna na vhodném místě v síti, monitoruje provoz sítě a informace zasílá na serverovou část. Výhodné je jako sondy využít síťové prvky jako jsou směrovače a přepínače, pokud umožňují odesílání informací o síťovém provozu. Neumožňují-li odesílání těchto informací, je nutné využít speciální zařízení. Na server jsou pak podle možností sondy zasílány informace o tocích, nebo informace o paketech proudících v síti, pokud sonda agregaci do toků neumožňuje.

Předzpracování Modul předzpracování zajišťuje přijmutí dat ze sond a jejich transformaci do vnitřních struktur. Protože pro behaviorální analýzu nejsou důležité obsahy jednotlivých paketů, ale trendy v používání sítě, obsahují vnitřní struktury informace o tocích. Pokud sonda neumí informace z paketů agregovat do toků, zajišťuje agregaci tento modul. Každý identifikovaný tok je pak předán NBA Engine pro zpracování.

NBA Engine Hlavní modul zajišťující behaviorální analýzu. Dostává vždy právě jeden tok z předzpracování, načte pro něj odpovídající profil z modelu a pokud je profil kompletní, zajistí porovnání daného toku s profilem a detekci anomálií. Pokud byl tok porovnán a bylo usouzeno, že se jedná o normální tok, zajistí aby byl profil aktualizován. Pokud byla detekována anomálie, k aktualizování profilu nedojde a jsou místo toho odeslány informace na modul výstupu. Pokud nebyl profil kompletní, NBA Engine pouze zajistí, aby byl daný tok zahrnut do profilu.

Model Obsahuje uložené profily jednotlivých aplikací a zajišťuje k nim přístup. Uložené modely musí být persistentní a musí být tedy průběžně ukládány na disk, aby v případě přerušení běhu systému nedošlo ke ztrátě profilů.

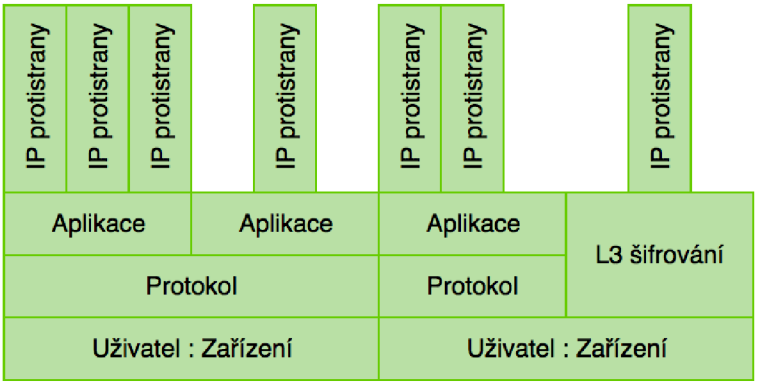
Detektor anomálií Tento modul slouží k určení míry podobnosti mezi tokem a jeho profilem. Pokud NBA Engine nalezne odpovídající profil k toku, zašle tok a jeho profil detektoru anomálií, který pomocí klasifikátoru určí míru podobnosti, kterou zašle zpět do NBA enginu.

Strojové učení Sada metod pro vytváření a aktualizaci profilů. Modul je využíván NBA Enginem, odkud přichází požadavky na začlenění toku do modelu, nebo aktualizaci modelu. Využívá jej také modul zpětné vazby pro explicitní začlenění toku do modelu.

Výstup Pokud byla detekována anomálie, modul výstupu transformuje údaje o nalezené anomálii - včetně míry podobnosti detekovaného toku s profilem - na zprávu pro uživatele. Zajistí pak také zobrazení zprávy vhodným způsobem, jako například výpis v textovém formátu do logu anomálií.

Zpětná vazba Modul zpětné vazby umožňuje uživatelům explicitně ovlivňovat profily v modelu. Pokud je v logu anomálií hlášen incident, ale uživatel po ověření usoudí, že se jedná o normální stav, který není ojedinělý a měl by být začleněn do profilu, může toho pomocí tohoto modulu dosáhnout.

4.3.2 Model chování uživatele



Obrázek 4.4: Složení modelu chování

Jak lze vidět na obrázku 4.4, je model chování rozdělen do vrstev ve stromové struktuře. Každá z nich obsahuje informace o užším pásmu komunikace až na detailní úroveň komunikace dvou uzlů. V každé z vrstev jsou uloženy údaje o identitě vrstvy, které popisují, která část komunikace v ní je zachycena, veškeré podvrstvy a statistické údaje popisující danou vrstvu. V případě protokolové vrstvy u TCP protokolu bude kromě podvrstev s používanými aplikacemi uložena i informace o počtu aplikací a jejich rozptylu, což umožňuje sledovat přístupy na nové aplikace a také doba poslední aktualizace modelu pro předejití stárnutí. Pokud uživatel přestal nějakou aplikaci používat, je nutné to pomocí těchto dat rozpoznat a model chování dané aplikace odebrat.

Uživatel : Zařízení je dvojice určující základní dělení modelu. Zařízení je jednoznačně určeno MAC adresou, uživatel je pak získán z identitního systému. Pomocí této dvojice lze filtrovat síťový provoz na komunikaci jediného účastníka na jediném zařízení.

Protokol omezuje sledovaný provoz na jediný IP protokol. Nejčastěji se jedná o TCP a UDP protokoly, ale je nutné sledovat i ostatní protokoly, jelikož např. přes ICMP protokol lze tunelovat libovolné spojení[36] a lze zde rozeznat VPN podle protokolů jako IPsec a GRE.

Aplikace je určena podle cílového portu u TCP a UDP protokolů. Protože je sledována odchozí komunikace, lze podle cílového portu rozlišit jednotlivé služby, ke kterým zařízení přistupuje.

Protistrana je určena cílovou IP adresou. V této vrstvě je zachycen model komunikace dvou uzlů v rámci aplikace.

4.3.3 Klasifikace datového provozu

Smyslem detektoru je klasifikovat provoz do dvou tříd - *normální provoz* a *abnormální provoz*. Tuto klasifikaci lze provést pomocí statistických metod, metod strojového učení, teorie her nebo jejich kombinací. Statistické metody shromažďují statistická data o provozu a model je tvořen průměrným vzorkem a prahem určujícím maximální povolenou odchylku. Pokud se potom testovaná data liší o více než zadaný práh, jsou klasifikována jako abnormální, jinak jsou považována za normální. U metod strojového učení je obvykle model tvořen jejich metodami, tento model je poté pomocí trénovacích dat zformován pro klasifikaci do potřebných tříd. Bohužel při hledání odchylek od normálního provozu máme k dispozici pouze zástupce jedné třídy a proto většina klasických metod strojového učení selhává. Pro tyto potřeby jsou použitelné jen některé algoritmy označované jako algoritmy pro hledání odchylek (ang. „outlier“), nejznámější jsou potom algoritmy K-nearest neighbor a Local Outlier Factor (LOF).

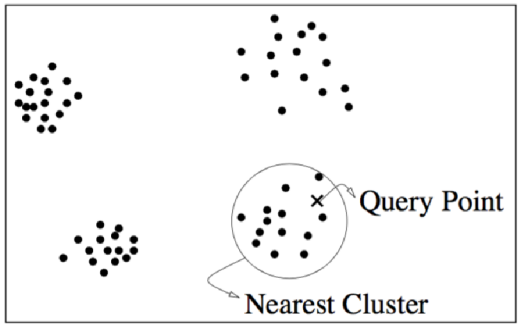
K-nearest neighbor

K-nearest neighbor algoritmus (označovaný většinou jen jako k-NN) je jeden ze základních algoritmů pro klasifikaci. Tento algoritmus chápe data jako množinu n-tic (matematicky vektorů), ve kterých hledá pro testovaný vzorek určitý počet nejbližších sousedů. Počet těchto sousedů je zadán parametrem k a vzdálenosti mezi n-ticemi jsou dány libovolnou distanční funkcí, nejčastěji je využívána Euclidovská vzdálenost definovaná jako:

$$d(x_i, y_i) = \sqrt{\sum_{i=1}^{k} (x_i - y_i)^2}$$

kde x a y jsou porovnávané vektory[32]. V těchto nalezených sousedech porovná algoritmus příslušnost do tříd a zařadí vzorek do té třídy, jejíž četnost je mezi sousedy nejvyšší. Při použití pro detekci anomálií zkoumá tento algorimus místo tříd sousedů pouze vzdálenost od jednotlivých sousedů, pokud je tato vzdálenost větší, než je obvyklé, je detekována anomálie. Na obrázku 4.5 je zobrazen model se shluky vzorků a testovaný vzorek, který je přiřazen do nejbližšího shluku.

Tento algoritmus, ač je v principu značně jednoduchý, je poměrně komplikované vhodně implementovat. Jeho spolehlivost je totiž velmi závislá na použité distanční funkci a testovacích datech[16]. Použití Euclidovské vzdálenosti jako distanční funkce předpokládá stejný význam a stejnou metriku pro všechny dimenze, v reálných datech se ale často stupnice pro jednotlivé dimenze diametrálně liší a některé dimenze mohou být dokonce určeny i textem[32]. Tento problém přetrvává i při použití jiných distančních funkcí jako například Hammingovy vzdálenosti nebo Minkowskiho metriky a je obecně platný pro různé klasifikační algoritmy. Proto je vhodné jednotlivé hodnoty transformovat a převést je na společnou metriku[26]. Pro transformaci hodnot lze použít různých metod jako je například Z-skóre



Obrázek 4.5: Klasifikace vzorku do shluku podle nejbližších sousedů, *převzato z [16]*

přemapovávající hodnoty podle jejich standartních odchylek na hodnoty se středem v 0 a maximální odchylce 1 podle vzorce:

$$x_{ij} = Z(x_{ij}) = \frac{x_{ij} - \overline{x_j}}{\sigma_j}$$

kde $\overline{x_j}$ označuje průměrnou hodnotu parametru j a σ_j jeho standartní odchylku[26]. Jinou možností je přemapovat hodnoty metodou *min-max* do intervalu *<0;1>* kde 0 označuje minimální hodnotu a 1 maximální pomocí vzorce:

$$MM(X_{ij}) = \frac{X_{ij} - X_{min}}{X_{max} - X_{min}}$$

kde X_{max} je maximální a X_{min} minimální hodnota[26].

Kromě problémů s různými metrikami u dimenzí se jako problematický jeví i samotný počet dimenzí. Zdánlivě by se mohlo zdát, že čím více dimenzí použijeme, tím přesnějších výsledků bude algoritmus dosahovat, ale při testování na reálných datech se prokázalo, že zvolení správného počtu dimenzí je netriviální úkol a po saturaci dochází k poklesu přesnosti[35]. Tento jev se nazývá *Curse of Dimensionality*. Pro redukci nadbytečných dimenzí lze využít speciálních metod, jako je například *Principal Component Analysis* (PCA), které procházejí trénovací data a odstraňují dimenze mající minimální vliv na klasifikaci[28]. Tyto metody jsou však často vysoce výpočetně náročné, protože musí porovnat celý trénovací soubor s výsledky klasifikátoru a určit míru s jakou se která dimenze projevila do cílové klasifikace.

Algoritmus k-NN používá jako svůj model celou trénovací množinu a testovací vzorky porovnává se vzorky z ní. Protože však trénovací množina může být dosti obsáhlá a pokud budeme chtít, aby se algoritmus s postupem času adaptoval na změny v datech, je nutné tuto množinu redukovat. Jedním z algoritmů redukce trénovacích dat je *Condensed Nearest Neighbor* (CNN), který se snaží vytvořit pro shluky jednotlivých vzorků prototypy, kterými lze v modelu celý shluk nahradit[37].

Specifickým problémem algoritmu k-NN je nalezení vhodného k, tedy počtu soustedů, kteří mají být porovnáváni. Parametr k je velmi závislý na použitých datech a zásadním způsobem ovlivňuje přesnost algoritmu. Speciálním případem je $k = 1$, kdy je porovnáván pouze nejbližší soused a na jeho základě je vzorek klasifikován. Použitím malých hodnot k dochází k rozdělení modelu na mnoho drobných oblastí podle jednotlivých trénovacích vzorků. Pokud však použijeme vyšší hodnoty k, je algoritmus schopen rozpoznat abnormální

vzorky v trénovacím souboru, které leží mimo oblasti hlavního shluku daných vzorků a tyto se pak ve výsledném modelu neprojeví. Zvyšováním hodnoty k tedy snižujeme vliv chyb v trénovacím souboru na cílový model, ale snižujeme tím také rozlišovací schopnosti modelu.

4.4 Implementace detektoru abnormálního chování

4.4.1 Předzpracování

Přestože se jako nejvýhodnější pro sběr dat jeví IPFIX, popřípadě NetFlow, jejich použití v testovacích podmínkách se ukázalo jako nesnadné pro nutnost použití síťové infrastruktury, složité nastavení sondy a nutnost implementace kolektoru do modulu předzpracování. Dalším problémem sběru toků v testovacím prostředí jsou implementace sond, které sbíraná data shromažďují a zasílají je na kolektor ve shlucích. Proto byl zvolen sběr surových síťových dat ve formátu *pcap* a z nich jsou poté v předzpracování tvořeny datové toky obsahující statistické informace.

Předzpracování je implementováno jako samostatný program *pcap2flow*, který umí načíst pcap soubor a zpracovat jej na CSV soubor s datovými toky. Tento program je vytvořen v jazyce *Python 2.7* a pro práci s pcap soubory používá knihovnu *Scapy*. Program je pak tvořen zejména třídou *FlowParser*, která umožňuje iterovat nad toky v pcap souboru a strukturou *Flow* reprezentující jeden datový tok. Pro vnitřní zpracování se využívají struktury *FlowHeader* obsahující hlavičku jednoho toku a *FlowStats*, kde se kumulují statistická data jednoho toku.

Hlavní program se provede, pokud je skript spuštěn jako program, tedy pokud je speciální proměnná *__name__* nastavena na *__main__*. Dojde ke zpracování argumentů pomocí standartní knihovny argparse a pokud byl zadán pcap soubor, dojde k jeho zpracování a vypsání nalezených toků ve formátu CSV. Každý řádek výstupu, kromě prvního, který obsahuje nadpisy sloupců, reprezentuje jeden nalezený tok a ve sloupcích jsou pak údaje o daném toku.

Namedtuple Flow je struktura obsahující popis jednoho datového toku. Jako datový typ byl zvolen *namedtuple*, protože se jedná o uspořádanou n-tici hodnot a *namedtuple* nám umožní si pojmenovat jednotlivé dimenze.

Namedtuple FlowHeader je struktura definující hlavičku datového toku, stejně jako u celého toku je i hlavička reprezentována datovým typem *nameduple*. Pro každý paket je vytvořena z jeho dat hlavička toku a porovnáno, jestli již tok se stejnou hlavičkou neexistuje. Pokud ano, je pouze aktualizován o statistická data z daného paketu, v opačném případě musí být vytvořen. Hlavička toku je tvořena MAC adresou odesílatele, IP adresami odesílatele i příjemce, použitým protokolem L4 vrstvy, IP Type of Service a zdrojovým a cílovým portem.

Třída FlowStats obsahuje statistické informace o jednom datovém toku. Na rozdíl od předchozích reprezentací informací o datovém toku bylo nutno vytvořit samostatnou třídu, protože datový typ *namedtuple* je neměnný a statistické informace je třeba aktualizovat s každým paketem spadajícím do daného toku.

Třída FlowParser zajišťuje zpracování toků ze souboru paketů. Chová se při tom jako iterátor a lze ji tedy využít ve smyčce, kde třída vrací postupně nalezené toky. V případě, že preferujeme zpracování celého souboru a vrácení pole toků, lze zavolat metodu *read_all*. Dále obsahuje třída metodu *next* pro načtení následujícího toku a privátní metody pro zpracování paketů - *_processTCPpacket*, *_processUDPpacket*, *_processICMPpacket*, pro uzavření toku - *_closeFlow* a pro převod *FlowHeader* a *FlowStats* na *Flow*.

FlowParser::next je metoda, které nalezne následující tok a vrátí o něm informace. Tato metoda načítá postupně pakety a ukládá si informace o nalezených tocích, jakmile je nějaký tok kompletní vrátí jej. Pokud již není na vstupu žádný paket, uzavře postupně všechny neukončené toky a vrací je. Bylo rozhodnuto, že pro potřeby této práce postačuje zpracování pouze TCP/IP paketů, proto je ostatní provoz touto metodou ignorován.

FlowParser::_processTCPpacket zpracovává jeden TCP paket. Nejprve je z dat obsažených v paketu zkonstruována hlavička toku a je ověřeno, zda obsahuje SYN příznak. Pokud ano, znamená to, že se jedná o nový tok a pokud je tok s danou hlavičkou nalezen jako otevřený, je uzavřen. Pokud se jedná o nový tok, je vytvořena jeho reprezentace v otevřených tocích a jsou do ní zaneseny statistické údaje o paketu, pokud je již nalezen otevřený tok se stejnou hlavičkou, jsou pouze statistické hodnoty aktualizovány.

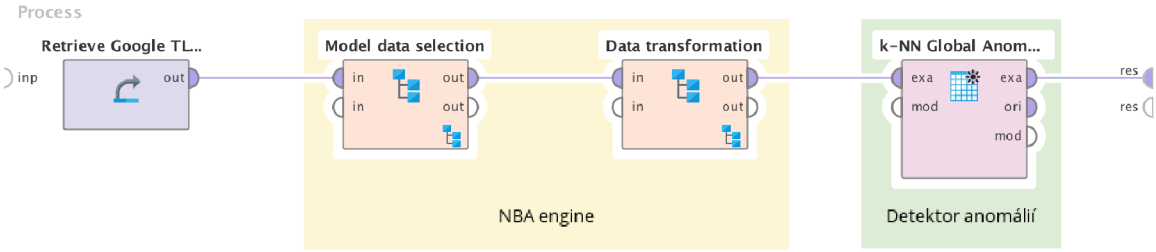
FlowParser::_closeFlow odstraní tok ze seznamu otevřených toků a vrátí jeho reprezentaci jako *Flow* objekt.

4.4.2 Klasifikace

Pro implementaci klasifikátoru pro rozpoznávání anomálií byla zvolena softwarová platforma RapidMiner, která je určena pro strojové učení, analýzu a práci s daty. Základní verze je navíc dostupná pod otevřenou licencí AGPL-3.0 a lze ji tedy použít a upravovat podle potřeb. Tato platforma umožňuje získat data z různých zdrojů, provádět nad nimi skupiny operací, které nazývá proces, a zobrazit a interpretovat jejich výsledky. Pro tvorbu a práci poskytuje platforma grafické rozhraní, k výpočetnímu jádru se ale dá připojit i pomocí API a začlenit jej tak do libovolného programu.

Základní verze umožňuje načítání z Open Source databází, Excelovských sešitů a souborů ve formátu CSV. Právě díky podpoře CSV a jednoduchosti tohoto formátu byl tento zvolen pro přenos dat mezi předzpracováním v Pythonu a klasifikátorem v RapidMineru. Proces je pak tvořen posloupností operátorů, což jsou bloky provádějící určitou činnost, které mají definovány vstupy, výstupy, prováděnou transformaci vstupů na výstupy a parametry této transformace[5].

V rámci klasifikace byly implementovány dva moduly z návrhu v kapitole 4.3, a to *NBA Engine* a samotný *Detektor anomálií*. Protože metoda k-NN používá jako model množinu prvků, načítá NBA Engine seznam toků z CSV souborů, ze kterých vybere pouze ty, které jsou pro daný model platné a ty pak dále upraví na formu vhodnou pro detektor anomálií. Detektor anomálií pak pro jednotlivé toky spočte míru anomality a zpřístupní výsledky výstupnímu zpracování RapidMineru, který umožňuje jejich zobrazení a interpretaci. Na obrázku 4.6 je zobrazen celý proces, který se skládá z obou těchto modulů.



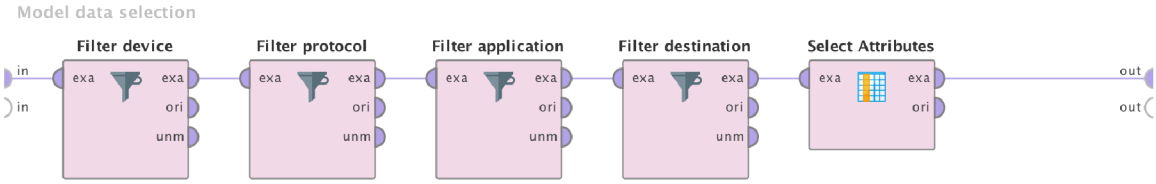
Obrázek 4.6: Proces vypočítávající hodnoty míry anomality pro jednotlivé toky

NBA Engine

NBA Engine je tvořen dvěma subprocesy - *Model data selection* a *Data transformation*, kde se první subproces stará o výběr pouze těch toků a informací v nich, které jsou pro daný model relevantní a druhý proces se stará o transformaci těchto dat do podoby vhodné pro výpočet Euclidovské vzdálenosti v metodě k-NN.

Model data selection je tvořen operátory *Filter Examples*, umožňující definovat filtr, který bude aplikován na vstupní data a výstupem operátoru budou pouze ta data, která splňují zadaná pravidla. Pomocí posloupnosti těchto operátorů, která je uvedena na obrázku 4.7, omezí tento podproces zpracovávaná data na jediný model komunikace dvou zařízení podle vrstev uvedených v kapitole 4.3.2.

Po vyfiltrování pouze těchto toků je, kvůli vyhnutí se Curse of Dimensionality popsané v kapitole 4.3.3, nutné vybrat z toků pouze ty dimenze, které mají dopad na určení třídy. Pro výběr pouze některých dimenzí ze vzorku je použit operátor *Select Attributes*, který umožní definovat, které dimenze se mají objevit na výstupu a které nikoli. Do modelových dat nebyly zahrnuty hodnoty použité k jejich filtrování, jelikož jsou u všech vzorků totožné, ani zdrojový port, který bývá u odchozí komunikace z klientských stanic většinou volen náhodně. Zahrnuta také nebyla informace o IP Type of Service, která byla u všech testovaných položek shodná a nepřispívá tedy ke klasifikaci.



Obrázek 4.7: Operátory použité pro výběr dat pro klasifikátor v RapidMineru

Data transformation slouží k vyřešení problému s různými metrikami dimenzí popsaném v kapitole 4.3.3 tím, že je transformuje na jednotnou metriku. To je provedeno pomocí operátoru *Normalize*. Tento operátor umožňuje tranformovat původní metriky dimenzí na nové pomocí jedné ze 4 metod - Z-transformací, transformací rozsahu, proporční transformací a interkvartilním rozsahem. Z-tranformace odpovídá *Z-skóre* popsanému v kapitole 4.3.3 a transormace rozsahu je *min-max* metoda popsaná tamtéž. Protože metoda *min-max*

transformuje hodnoty na rozsah $< 0; 1 >$, který lze přirozeněji interpretovat například jako hodnotu ve fuzzy logice, byla zvolena tato transformace, po jejímž provedení jsou data připravena pro metodu k-NN.

Detektor anomálií

V základní verzi RapidMineru je implementován klasifikátor k-NN, který slouží k zařazení dat do tříd.

Německé výzkumné středisko umělé inteligence (DFKI[2]) zveřejnilo rozšiřující modul pro RapidMiner, který obsahuje mimo jiné použití k-NN pro detekci anomálií. Základní verze RapidMineru sice neumožňuje instalaci rozšíření pomocí GUI rozhraní a repozitáře *RapidMiner Marketplace*, ale modul lze doinstalovat manuální cestou. Po instalaci je možné použít operátor *k-NN Global Anomaly Score*, který příjmá data, u kterých spočte *míru anomality*, což je průměrná vzdálenost od k nejbližších sousedů. Kromě průměrování vzdálenosti lze tento operátor také nastavit pro nalezení n-tého nejbližšího souseda a skóre pak nastavit podle něj. Pro výpočty vzdáleností je implementováno mnoho metod včetně Euclidovské a Manhatanské vzdálenosti. Protože při k-NN je nejvíce času spotřebováno na výpočet vzdáleností od ostatních vzorků, aby mohly být vypočteni nejbližší sousedi, umožňuje tento operátor vypočtené hodnoty uchovat jako model a znovu je pak načíst.

Operátor použitý při implementaci detektoru anomálií byl nastaven na výpočet vzdáleností pomocí Euclidovy metody, pro kterou byla data optimalizována. Počet porovnávaných sousedů byl nastaven na hodnotu 2, která byla zvolena jako vhodná vzhledem k testovacím datům. Nastavení nízké hodnoty je problematické, protože při výskytu více anomálních toků může dojít pouze k jejich porovnání navzájem a výsledná vzdálenost není vypovídající. U vyšších počtů kontrolovaných sousedů je zapotřebí více dat v modelu, aby byla detekce možná.

4.5 Praktické zkušenosti s detektorem

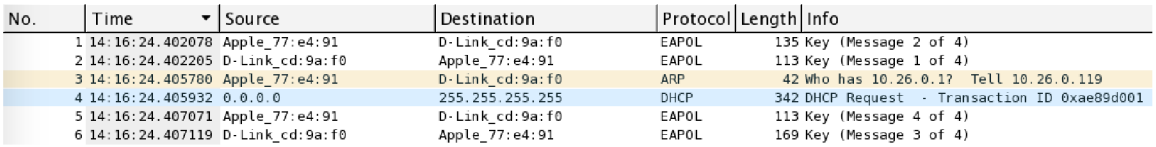
Součásti detektoru implementované podle popisu v předešlé kapitole byly testovány na síťových datech zachycených na vývojovém stroji. Nejprve byla data odchycena ze síťového rozhraní zařízení pomocí programu *Wireshark* a uložena ve formátu *pcap*. Následně byla data předzpracována programem *pcap2flow* na CSV soubor s toky. Tento soubor byl importován do repozitáře v RapidMineru a použit jako datový zdroj pro procesy nad ním.

4.5.1 Získávání dat

Testovací data byla odchycena programem *Wireshark*. Tento program umožňuje zaznamenávat veškeré pakety procházející síťovým rozhraním, v případě bezdrátové karty přepnuté do tzv. promiskuitního módu je pak schopen zaznamenávat veškeré pakety, které prochází bezdrátovou sítí.

Tento program umožňuje kromě živého zachytávání paketů také ukládat a načítat pakety ve formátu *pcap* a interpretovat data zachycených paketů, tedy zobrazit rámce s jednotlivými parametry v čitelné podobě a rozpoznat o jaký typ paketu se jedná. Náhled zachycených paketů v tomto programu je zobrazen na obrázku 4.8.

Bylo zachyceno několik testovacích vzorků zahrnujících připojení k síti, nešifrovanou i šifrovanou komunikaci s webovými stránkami a připojení síťového souborového systému. Kromě takto zachycených dat byla použita i data nasbíraná v prostředí školní sítě, tato data



No.	Time	Source	Destination	Protocol	Length	Info
1	14:16:24.402078	Apple_77:e4:91	D-Link_cd:9a:f0	EAPOL	135	Key (Message 2 of 4)
2	14:16:24.402205	D-Link_cd:9a:f0	Apple_77:e4:91	EAPOL	113	Key (Message 1 of 4)
3	14:16:24.405780	Apple_77:e4:91	D-Link_cd:9a:f0	ARP	42	Who has 10.26.0.1? Tell 10.26.0.119
4	14:16:24.405932	0.0.0.0	255.255.255.255	DHCP	342	DHCP Request - Transaction ID 0xae89d001
5	14:16:24.407071	Apple_77:e4:91	D-Link_cd:9a:f0	EAPOL	113	Key (Message 4 of 4)
6	14:16:24.407119	D-Link_cd:9a:f0	Apple_77:e4:91	EAPOL	169	Key (Message 3 of 4)

Obrázek 4.8: Přihlašování do bezdrátové sítě se zabezpečením WPA v programu Wireshark

však byla využita pouze pro testování předzpracování, vzhledem k jejich velkému objemu a příliš komplexní struktuře zahrnující komunikaci velkého počtu uzlů.

4.5.2 Předzpracování

Předzpracování nasbíraných paketů a vytvoření souborů toků s jejich reprezentacemi bylo testováno na všech datech. Odladění programu probíhalo na vývojovém stroji za pomoci testovacích dat a manuální kontrolou bylo ověřeno, že statistické informace o tocích odpovídají testovaným paketům. Program *Wireshark* umožňuje seskupit TCP a UDP pakety do *proudů*, což jsou pakety se stejnou čtveřicí - zdrojový a cílový port a zdrojová a cílová IP adresa. Také možnost filtrovat pakety podle jednotlivých políček v rámci jako například podle TCP SYN příznaku byla velice vhodná.

device	source IP	destination IP	IP protocol	IP ToS	source port	destination port	opened	finished	packet count	start time	end time	bytes
ac:bc:32:77:e4:91	10.26.0.119	172.217.16.99	6	0	55820	80	True	True	13	13:17:57	13:18:01	997
ac:bc:32:77:e4:91	10.26.0.119	172.217.16.99	6	0	55817	80	True	True	14	13:17:57	13:18:01	1041
ac:bc:32:77:e4:91	10.26.0.119	172.217.16.99	6	0	56035	443	True	True	26	13:18:04	13:19:06	2076

Obrázek 4.9: Příklad výstupu předzpracování se třemi nalezenými toky

Po úspěšném otestování na datech z vývojového stroje byl program otestován na školním serveru na větším rozsahu dat odchycených ze školní sítě. Původní implementace programu *pcap2flow* předpokládala, že toků je v datovém provozu podstatně menší množství, nežli paketů. V reálném prostředí se ukázalo, že se v síti přenáší i větší množství toků o velmi malé délce a neplatí tedy, že by se síťový provoz sestával zpravidla z menšího množství dlouhých toků obsahujících mnoho paketů. Z tohoto důvodu byl program upraven, neboť původní implementace uchovávala veškeré toky v operační paměti pro další práci s nimi, což bylo u testovaného pcap souboru neúnosné. Po úpravě je možné nad toky iterovat, program tedy v operační paměti udržuje pouze ty toky, které zatím nejsou ukončeny a ukončené toky umožňuje vracet ihned při uzavření a tímto způsobem lze iterativně zpracovat i větší soubory.

4.5.3 Testování v RapidMineru

Soubory se síťovými toky vytvořené pomocí *předzpracování* v předešlé kapitole byly následně použity pro otestování detektoru anomálií implementovaném na platformě RapidMiner. Tyto soubory byly importovány do datového repozitáře a následně nastavovány jako datový vstup pro implementovaný proces. Takto získaná data byla následně pomocí podprocesu *Model data selection* filtrována pouze na relevantní data, která byla transformována pomocí *Data transformation* na jednotnou metriku. Poté byla data zpracována pomocí metody k-NN a spočteny hodnoty vzdáleností od sousedů pro jednotlivé toky. Výsledná

data byla interpretována a bylo ověřeno, že implementovaný systém je schopen detekovat anomality v síťovém provozu.

Příklad č. 1 - Příprava dat pro detekci anomálií

Pro tento příklad byly využita data z 24.3.2016 nasbíraná na vývojovém stroji při úvodním připojení k síti a spuštění webového prohlížeče a několika služeb. Nasbíraná data byla nejprve filtrována na komunikaci mezi dvěma zařízeními pomocí podprocesu *Model data selection*, jehož výstup je zobrazen na obrázku 4.10. Z celého nasbíraného vzorku jsou ve výsledné komunikaci pouze data zařízení s MAC adresou *ac:bc:32:77:e4:91* a IP adresou *10.26.0.119*, a to jeho komunikace protokolem *TCP* s aplikací běžící na portu *80*, tedy s webovým serverem, na zařízení s IP adresou *104.24.107.76*.

Row No.	opened	finished	packet count	start time	end time	bytes
1	True	True	349	1:17:55 PM...	1:18:02 PM...	16744
2	True	True	391	1:17:55 PM...	1:18:02 PM...	17074
3	True	True	486	1:17:55 PM...	1:18:02 PM...	22502
4	True	True	464	1:17:55 PM...	1:18:02 PM...	23161
5	True	True	530	1:17:55 PM...	1:18:02 PM...	23233
6	True	True	409	1:17:55 PM...	1:18:02 PM...	18802

Obrázek 4.10: Příklad informací vybraných ze síťového provozu blokem *Model data selection*

Tato data obsahují pouze statistické informace ze síťových toků, které této komunikaci patří a jsou vhodná pro následnou detekci anomálií. Před samotnou detekcí je však potřeba transformovat metriky jednotlivých dimenzí, zde sloupců, jinak by byla detekce více ovlivněna vyššími hodnotami jako je počet bytů a paketů a nesprávně by bylo přistupováno i k časovým známkám. Proto jsou v bloku *Data transformation* tyto hodnoty převedeny do rozsahu $< 0; 1 >$, kde 0 značí minimální hodnotu ve vzorku a 1 maximální. Výstup transformace uvedených dat je zobrazen v obrázku 4.11.

Row No.	packet count	bytes	opened	finished	start time	end time
1	0	0	1	1	1	1
2	0.232	0.051	1	1	1	1
3	0.757	0.887	1	1	1	1
4	0.635	0.989	1	1	1	1
5	1	1	1	1	1	1
6	0.331	0.317	1	1	1	1

Obrázek 4.11: Příklad síťových toků připravených ke klasifikaci metodou k-NN

Vzhledem ke skutečnosti, že nasbírané toky byly všechny korektně otevřeny i uzavřeny, je hodnota těchto sloupců stejná. Stejně tak doba zahájení a ukončení spojení je shodná ve všech vzorových tocích, proto lze pozorovat stejnou vlastnost i u transformovaných hodnot.

Kde však lze pozorovat změny, je počet přenesených paketů a bytů od minimálního počtu v toku č. 1 až po maximální v toku č. 5.

Příklad č. 2 - Detekce anomálie v načítání webové stránky

Tento příklad demonstruje detekci anomálie při načtení šifrované webové stránky. Pomocí protokolu TLS byla několikrát načtena úvodní stránka vyhledávače Google, poté v něm byly vyhledány odkazy na stránky s pojmem „NBA“ a nakonec zobrazeny obrázky vztahující se k tomuto pojmu. Ve výsledných datech pak lze rozlišit jednotlivé přístupy pomocí uložených statistických informací v síťových tocích, které je popisují.

Selekcí a přípravou dat se zabýval příklad č. 1, proto je v tomto příkladu uveden pouze pro ilustraci výběr ze seznamu statistických informací z relevantních toků v obrázku 4.12, ale blíže se tento příklad přípravou dat nezabývá.

Row No.	opened	finished	packet count	start time	end time	bytes
11	True	False	9	9:35:40 AM...	9:35:40 AM...	877
12	True	False	9	9:35:40 AM...	9:35:40 AM...	877
13	True	True	7	9:34:50 AM...	9:35:04 AM...	317
14	True	False	277	9:35:40 AM...	9:35:42 AM...	18497
15	True	False	10	9:35:40 AM...	9:35:40 AM...	930
16	True	False	10	9:35:40 AM...	9:35:40 AM...	921
17	True	True	13	9:34:50 AM...	9:35:04 AM...	1054

Obrázek 4.12: Výběr z informací o tocích použitých pro detekci anomálií

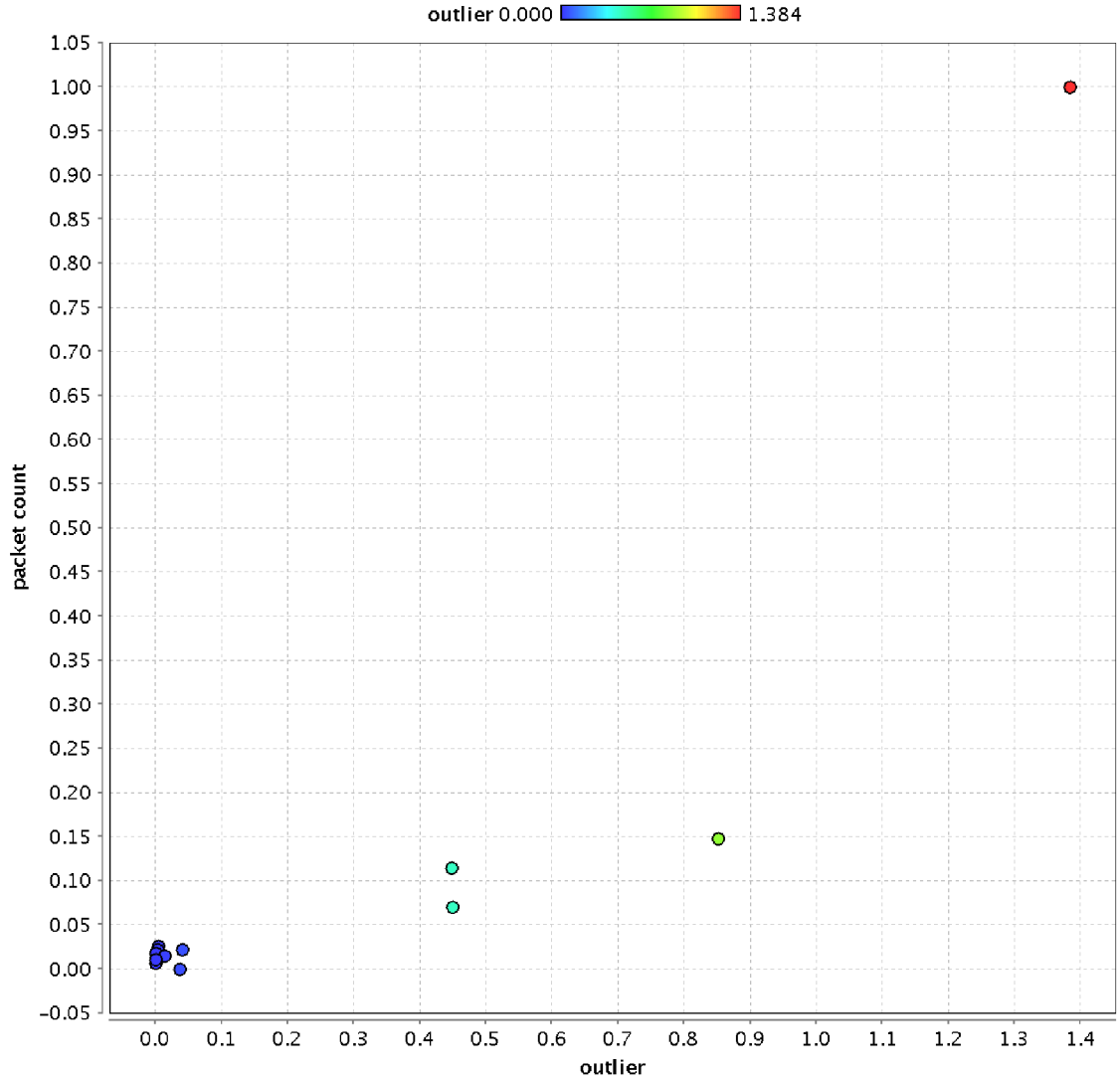
Po průchodu blokem pro detekci anomálií pomocí metody k-NN jsou jednotlivé toky označeny mírou abnormality, což je průměrná vzdálenost od k nejbližších sousedů. V tomto příkladě bylo zvoleno $k = 2$, vzdálenost tedy odpovída průměru mezi vzdálenostmi ode dvou sousedů. Výstup z tohoto bloku pro toky z obrázku 4.12 je uveden v obrázku 4.13, míra abnormality je spočtena ve zvýrazněném sloupci označeném *outlier*.

Row No.	outlier	packet count	bytes	opened	finished	start time	end time
11	0	0.007	0.031	1	0	1	0.947
12	0	0.007	0.031	1	0	1	0.947
13	0.036	0	0	1	1	0	0
14	1.384	1	1	1	0	1	1
15	0.000	0.011	0.034	1	0	1	0.947
16	0	0.011	0.033	1	0	1	0.947
17	0.002	0.022	0.041	1	1	0	0

Obrázek 4.13: Ukázkové hodnoty výstupu k-NN detekce anomálií

Lze pozorovat výraznou změnu hodnoty míry abnormality u toku označeném číslem 14, který byl identifikován jako abnormální. Jedná se o datový tok, který odpovídá načtení

obrázků, přenesené množství paketů a bytů tedy prudce vybočuje z normálního provozu určeného načítáním úvodní stránky Google. Kromě tohoto extrému lze ve výstupu detektoru abnormalit pozorovat i načtení hledaných odkazů, které se též odlišuje, ne však tak zřetelně.



Obrázek 4.14: Graf znázorňující shluky síťových toků s identifikací silně abnormální hodnoty

V grafu na obrázku 4.14 lze snadno identifikovat rozdíly v testovaných tocích. Anomální tok číslo 14 je zobrazen v pravém horním rohu, zatímco shluk reprezentující normální provoz udaný načítáním úvodní stránky je v levém spodním rohu. Také lze identifikovat načtení odkazů, které je zobrazeno zeleným bodem vpravo od normálního provozu.

Kapitola 5

Závěr

Cíle diplomové práce, jak jsou uvedeny v kapitole 1 *Úvod a cíle práce*, byly doaženy. Naplněním postupových cílů došlo i k naplnění hlavního cíle, kterým bylo navržení systému pro rozpoznávání APT útoků a jeho rozpracování a ověření v prostředí počítačové sítě.

5.1 Postupové cíle práce

5.1.1 Analýza současného stavu znalostí o APT útocích

Co přesně APT útok je a čím se odlišuje od jiných typů sofistikovaných útoků ještě není v odborné literatuře zcela ustáleno. V rámci diplomové práce jsem prostudoval dostupnou literaturu a seznam relevantní literatury uvádím v závěru práce. Za klíčovou považuji příručku profesního sdružení ISACA - *Advanced Persistent Threats: How to Manage the Risk to Your Business[22]*, z jejíhož pojetí APT útoku v práci vycházím. Za APT útoky se považují cílené, profesionálně vedené útoky vyznačující se zejména dlouhou dobou trvání, obtížností jejich rozpoznání a použitím netriviálních technik pro provedení útoku.

Vymezení klíčových charakteristik APT útoku, jeho životního cyklu a současných způsobů ochrany proti útoku uvádím v kapitole 2 *APT útoky*.

Současné způsoby rozpoznávání APT útoku v prostředí počítačových sítí pomocí modelování chování uživatelů jsou blíže rozebrány v kapitole 4.1 *Behaviorální analýza*.

5.1.2 Definice klíčových charakteristik pro rozpoznávání APT útoků

Charakteristiky APT útoků, podle kterých lze usuzovat na vlastní napadení organizace, jsou v práci zavedeny jako takzvané *symptomy* z důvodů jejich obtížné pozorovatelnosti a proměnlivosti. V práci jsem navrhl celkem 7 následujících symptomů:

- abnormální chování software,
- abnormální přístup k datům,
- abnormální použití zařízení,
- abnormální síťová komunikace,
- dlouhodobý průběh,
- výskyt phishingových e-mailů,

- změna konfigurace zařízení.

Navržené symptomy jsou podrobně popsány v kapitole 3.1 *Identifikace symptomů APT útoku.*

5.1.3 Navržení struktury a funkce systému pro rozpoznávání APT útoků v prostředí počítačové sítě

Nejdříve jsem v kapitole 3.2 *Možnosti detekce jednotlivých symptomů* navrhl jakým způsobem lze detekovat jednotlivé symptomy. Na jejich spolupůsobení je postaven návrh obecného systému pro rozpoznávání APT útoků, který je blíže popsán v kapitole 3.3 *Návrh architektury systému pro detekci útoků.*

Podle zadání diplomové práce jsem se blíže zabýval rozpoznáváním APT utoků v prostředí počítačové sítě, zaměřil jsem se tedy na symptom *abnormální síťová komunikace.* Navrhl jsem detekční systém využívající modelování chování uživatelů v síti a zjištěných anomálií v jejich chování. V kapitole 4.3 *Návrh softwarové architektury* uvádím vhodnou architekturu detekčního systému a popis funkcionaity jednotlivých komponent tvořících tento systém.

5.1.4 Implementace navrženého systému pro rozpoznávání APT útoků v prostředí počítačové sítě

Detekční systém pro rozpoznávání APT útoku v prostředí počítačové sítě jsem programově zpracoval pro komponentu *Předzpracování* v jazyce Python tak, aby bylo možné převést surová síťová data na jejich abstrakci ve formě síťových toků. Modelování chování uživatelů je postaveno nad zavedenými síťovými toky a tím je umožňeno detekovat anomální chování uživatelů. Pro detekci anomálního chování uživatelů jsem využil platformu RapidMiner a algoritmus K-nearest neighbors (k-NN) z důvodu jeho dobré aplikovatelnosti pro detekci anomálií a univerzálnosti použití. V RapidMineru jsem implementoval komponenty *Detekce anomálií* a *NBA Engine.* Z výstupních údajů poskytovaných RapidMinerem je možné usuzovat na abnormální chování uživatele.

Podrobnější informace jsou uvedeny v kapitole 4.4 *Implementace detektoru abnormálního chování.*

5.1.5 Otestování navrženého systému pro rozpoznávání APT útoků a zhodnocení dosažených výsledků

Testovací data jsem získal odchycením síťového provozu na vývojovém zařízení pomocí programu *Wireshark.* Nad těmito daty jsem ověřil funkčnost předzpracování na síťové toky. Po ověření funkčnosti jsem nechal předzpracovat větší rozsah síťových dat získaných z prostředí školní sítě odpovídajících reálnému provozu. Data z reálného provozu ukázala, že původně navržený režim dávkového předzpracování síťových toků může být omezen velikostí operační paměti, a proto jsem režim změnil na průběžné ukládání výsledků určených pro další zpracování komponentou *NBA Engine.*

Na výstupu z detektoru jsem zjistil, že navržený systém pro rozpoznávání APT útoků je schopen rozpoznat anomálii v síťovém provozu a detektor je podle navržené architektury implementovatelný. Výsledky dosažené tímto testováním jsou uvedeny v kapitole 4.5 *Praktické zkušenosti s detektorem.*

5.2 Hlavní cíl práce

V diplomové práci jsem navrhl systém pro rozpoznávání APT utoků a v prostředí počítačové sítě jsem vytvořil detekční systém pro rozpoznávání APT útoku podle modelování chování uživatelů a otestoval jeho funkčnost.

5.3 Další rozvoj systému pro rozpoznávání APT útoků

V rámci diplomové práce jsem navrhl obecný systém pro detekci APT útoků postavený na sedmi symptomech, realizačně jsem prověřil detekci abnormálního chování uživatelů v prostředí počítačové sítě. Další navazující rozvoj systému pro rozpoznání APT útoků v celé šíři této problematiky by měl být založen na navržení, implementaci a otestování detektorů ostatních symptomů.

Literatura

[1] APT1. [online], [cit. 2016-04-19].
URL http://intelreport.mandiant.com/Mandiant_APT1_Report.pdfm

[2] RapidMiner Extension: Anomaly Detection. [online], [cit. 2016-05-18].
URL http://madm.dfki.de/rapidminer/anomalydetection

[3] The Trusted Platform Module Explained. [online], [cit. 2016-05-22].
URL http://www.cryptomathic.com/news-events/blog/the-trusted-platform-module-explained

[4] Landing another blow against email phishing. [online], 2012, [cit. 2016-05-20].
URL https://security.googleblog.com/2012/01/landing-another-blow-against-email.html

[5] RapidMiner Studio - Manual. [online], 2014, [cit. 2016-05-17].
URL http://docs.rapidminer.com/downloads/RapidMiner-v6-user-manual.pdf

[6] Cyberoam's Layer 8 Technology. [online], ©1999-2014, [cit. 2016-01-10].
URL https://www.cyberoam.com/downloads/Whitepaper/CyberoamLayer8Technology.pdf

[7] FlowMon ADS - Popis produktu. [online], ©2007 – 2015, [cit. 2016-01-09].
URL https://www.invea.com/data/flowmon/flowmon_ads_pb_cz.pdf

[8] Network Monitoring basics. Promiscuous Mode, Hubs and Switches. [online], ©2008-2015, [cit. 2016-05-20].
URL https://landetective.com/products/internet-monitor/manual/traffic-analysis.html

[9] Combating Advanced Persistent Threats. [online], ©2011, [cit. 2016-04-16].
URL http://www.mcafee.com/us/resources/white-papers/wp-combat-advanced-persist-threats.pdf

[10] Spear-Phishing Email: Most Favored APT Attack Bait. [online], ©2012, [cit. 2016-05-10].
URL http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/white-papers/wp-spear-phishing-email-most-favored-apt-attack-bait.pdf

[11] Ashley Madison Data Breach. [online], ©2014, [cit. 2016-01-01].
URL http://ashleymadisondatabreach.com

[12] McAfee Network Threat Behavior Analysis. [online], ©2014, [cit. 2016-01-09].
URL http://www.mcafee.com/us/resources/data-sheets/ds-network-threat-behavior-analysis.pdf

[13] M-Trends® 2015: A View From The Front Lines. [online], ©2015, [cit. 2016-04-22].
URL https://www2.fireeye.com/rs/fireye/images/rpt-m-trends-2015.pdf

[14] Agrawal, K.; Makwana, H.: A Study on Critical Capabilities for Security Information and Event Management. *International Journal of Science and Research*, ročník 4, č. 7, 2015: s. 1893–1896, ISSN 2319-7064.

[15] Alotibi, G.; Li, F.; Clarke, N.; aj.: Behavioral-Based Feature Abstraction From Network Traffic. Proceedings of the International Conference on Information Warfare and Security, 2015, s. 1–9.

[16] Beyer, K.; Goldstein, J.; Ramakrishnan, R.; aj.: When Is "Nearest Neighbor"Meaningful? 1999.

[17] Cutler, T.: The Anatomy of an Advanced Persistent Threat. [online], 2010-12-06, [cit. 2016-01-05].
URL http://www.securityweek.com/anatomy-advanced-persistent-threat

[18] Donohue, B.: What is APT? [online], 2013-06-11, [cit. 2016-01-03].
URL https://blog.kaspersky.com/apt/2050/

[19] Hudson, B.: Advanced Persistent Threats: Detection, Protection and Prevention. [online], [cit. 2016-04-24].
URL http://i.crn.com/custom/Sophos_Advanced_Persistent_Threats.pdf

[20] Karen Scarfone, P. M.: Guide to Intrusion Detection and Prevention Systems (IDPS). [online], 2007-02, [cit. 2016-01-06].
URL http://csrc.nist.gov/publications/nistpubs/800-94/SP800-94.pdf

[21] Krebs, B.: Catching Up on the OPM Breach. [online], 2015-06-15, [cit. 2016-01-01].
URL https://krebsonsecurity.com/2015/06/catching-up-on-the-opm-breach/

[22] Lacey, D.: *Advanced Persistent Threats: How to Manage the Risk to Your Business*. ISACA, 2013, ISBN 978-1-60420-348-6.

[23] Merchant, S.: Which to use for security analytics, IPFIX or sFlow? [online], ©2016, [cit. 2016-01-12].
URL https://www.gigamon.com/blog/2015/05/08/which-to-use-for-security-analytics-ipfix-or-sflow/

[24] Minařík, P.: Behaviorální analýza útoků v síti. *Data Security Management*, ročník 3, 2011: s. 18–21.

[25] Minařík, P.; Labský, M.: Behaviorální analýza datového provozu v praxi. *Data Security Management*, ročník 2, 2014: s. 28–31.

[26] Mohamad, I. B.; Usman, D.: Standardization and Its Effects on K-Means Clustering Algorithm. [online], 2013, [cit. 2016-05-15].
URL http://maxwellsci.com/print/rjaset/v6-3299-3303.pdf

[27] Musa, S.: Advanced Persistent Threat - APT. [online], 2014-03, [cit. 2016-01-06].
URL https://www.academia.edu/6309905/Advanced_Persistent_Threat_-_APT

[28] Pechenizkiy, M.: The Impact of Feature Extraction on the Performance of a Classifier: kNN, Naïve Bayes and C4.5. [online], [cit. 2016-05-16].
URL
http://www.win.tue.nl/~mpechen/publications/pubs/PechenizkiyCanAI05.pdf

[29] PwC: 2015 Information security breaches survey. [online], 2015, [cit. 2016-01-01].
URL https:
//www.pwc.co.uk/assets/pdf/2015-isbs-technical-report-blue-digital.pdf

[30] Qin, M.; Hwang, K.: Anomaly-based Intrusion Detection from Traffic Datamining on Internet Connections. [online], 2004-12-07, [cit. 2016-01-07].
URL http://gridsec.usc.edu/hwang/papers/AnomalyBasedIDSDec2004.pdf

[31] von Roessing, R. M.; Benetis, V.; Dimitriadis, C. K.; aj.: *Transforming Cybersecurity*. ISACA, 2013, ISBN 978-1-60420-360-8.

[32] Sayad, S.: K Nearest Neighbors - Classification. [online], ©2010-2016, [cit. 2016-05-15].
URL http://www.saedsayad.com/k_nearest_neighbors.htm

[33] Schneier, B.: Hacking Team Is Hacked. [online], 2015-06-15, [cit. 2016-01-01].
URL https://www.schneier.com/crypto-gram/archives/2015/0715.html#13

[34] Singh, D. K.; Gupta, J. K.: An approach for Anomaly based Intrusion detection System using SNORT. *International Journal of Scientific & Engineering Research*, ročník 4, č. 9, 2013: s. 648–652, ISSN 2229-5518.

[35] Spruyt, V.: About the Curse of Dimensionality. [online], 2014, [cit. 2016-05-16].
URL http://www.datasciencecentral.com/profiles/blogs/
about-the-curse-of-dimensionality

[36] Stødle, D.: Ping Tunnel. [online], 2011, [cit. 2016-05-16].
URL http://www.cs.uit.no/~Edaniels/PingTunnel/

[37] Sutton, O.: Introduction to k Nearest Neighbour Classification and Condensed Nearest Neighbour Data Reduction. [online], 2012, [cit. 2016-05-16].
URL http://me.seekingqed.com/files/intro_KNN.pdf

Přílohy

Seznam příloh

Příloha A

Obsah CD

Přiložené CD obsahuje text práce, vzorová data a programy použité v této práci.

V kořenovém adresáři je uloženo PDF s tímto textem práce a tři adresáře - data, implementation a text.

data Tento adresář obsahuje nasbíraná síťová data ve formátu pcap a jejich předzpracované údaje ve formátu CSV.

implementation V tomto adresáři lze nalézt zdrojový soubor *pcap2flow.py* obsahující skript v jazyce Python pro předzpracování síťových dat a zdrojové soubory procesů pro platformu RapidMiner. Jsou zde přítomnu celkem 3 procesy - *Anomaly Detector* s implementací samotného detektoru a dva příklady, které odpovídají příkladům z kapitoly 4.5. Dále je zde obsažena složka *RapidMiner Extensions* s použitým rozšířením pro RapidMiner a soubor *README* popisující instalaci jednotlivých komponent.

text Tento adresář obsahuje zdrojové soubory programu LaTeX sloužící k vytvoření této zprávy.

Příloha B

Použité pojmy a zkratky

APT Advanced Persistent Threat - dlouhodobé, pokročilé hrozby.

Backdoor - program umožňující neautorizovaný přístup do systému.

DDoS Distributed Denial of Service - distribuovaný útok typu odepření služby.

Firewall - program, či zařízení, filtrující síťový provoz podle zadaných pravidel.

IDS Intrusion Detection System - systém detekce útočníků.

IOT Internet of Things - označení vestavných zařízení propojených na internet.

IPFIX IETF protokol pro přenos síťových toků.

IPS Intrusion Prevention System - systém prevence útočníků.

k-NN k-nearest neighbor - algoritmus použitelný pro klasifikaci dat.

L2, L3, L4 - označení vrstev v ISO/OSI modelu.

Log - soubor se sledovanými informacemi, většinou o běhu nějakého systému.

Malware Malitious Software - obecně škodlivý software.

NAT Network Address Translation - program provádějící překlad síťových adres mezi lokálním a globálním prostorem.

NBA Network Behavior Analysis - analýza chování sítě.

NetFlow Cisco proprietární protokol pro přenos síťových toků.

P2P Peer-to-peer - označení architektury, kde si jsou jednotliví účastníci komunikace rovni.

PLC Programmable Logic Controller - průmyslový počítač.

Proxy server - slouží pro přeposílání dat a často anonymizaci jejich zdroje nebo cíle.

Social Engineering - získávání informací pomocí manipulace lidí.

Spear phishing - metoda útoků založených na cíleném podvrhávání oficiálních dokumentů jako jsou e-maily nebo webové stránky.

SIEM Security Information and Event Management - systém pro správu bezpečnostních informací a událostí.

TPM Trusted Platform Module - kryptografický čip sloužící jako bezpečné úložiště klíčů v zařízení.

VPN Virtual Private Network - virtuální síť sloužící k, často šifrovanému, propojení různých podsítí.