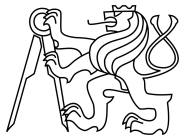
ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ



ZADÁNÍ DIPLOMOVÉ PRÁCE

Název:	Detekce phishingu
Student:	Bc. Filip Mudruněk
Vedoucí:	Ing. Jan Motl
Studijní program:	Informatika
Studijní obor:	Webové a softwarové inženýrství
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	Do konce letního semestru 2016/17

Pokyny pro vypracování

1. Definujte pojem Phishing a jeho specifika a prozkoumejte techniky, které se používají pro provedení tohoto typu útoku.
2. Proveďte srovnání moderních metod a nástrojů používaných pro boj s phishingem a navrhněte klasifikátor schopný rozeznat phishing od běžné elektronické komunikace.
3. Diskutujte použití relevantních technik strojového učení.
4. Navrhněte architekturu celého systému pro boj s phishingem s přihlédnutím k možnostem rozšiřitelnosti a škálovatelnosti.
5. Stanovte metriku pro měření úspěšnosti nasazení navrhovaného systému.
6. Vytvořte prototypovou implementaci celého systému. Implementaci otestujte na datech Jose Nazaria.

Seznam odborné literatury

Phishing Detection: A Literature Survey http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6497928
Threat Assessment by Patrick Hans Vikingsson
Phishing Attacks and Countermeasures by Zulfikar Ramzan
Investigating Information Structure of Phishing Emails Based on Persuasive Communication Perspective by Ki Jung Lee, Il-yeol Song
Gamification of Information Systems and Security Training : Issues and Case Studies by David Thornton, Guillermo Francia Iii
Is this a joke? The impact of message manipulations on risk perceptions by Dustin Ormond, Merrill Warkentin

L.S.

Ing. Michal Valenta, Ph.D.
vedoucí katedry

prof. Ing. Pavel Tvrdík, CSc.
děkan

V Praze dne 9. února 2016

České vysoké učení technické v Praze
Fakulta informačních technologií
Katedra softwarového inženýrství

Diplomová práce

Detekce phishingu

Bc. Filip Mudruněk

Vedoucí práce: Ing. Jan Motl

9. května 2016

Poděkování

Tímto bych chtěl poděkovat svému vedoucímu práce Ing. Janu Motlovi za jeho čas, věcné připomínky, konzultace a častý feedback. Dále bych rád poděkoval společnosti Excello za spolupráci. Umožnila mi nahlédnout blíže do světa problematiky emailové komunikace a předala mi cenné poznatky z praxe. V neposlední řadě bych chtěl poděkovat přítelkyni a rodině za podporu během psaní této práce.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 46 odst. 6 tohoto zákona tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou, a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla, a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu), licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 9. května 2016 .....

Odkaz na tuto práci

Mudruněk, Filip. *Detekce phishingu.* Diplomová práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2016.

Abstrakt

Cílem této práce je analýza technik a metod využívaných k phishingu a návrh automatického klasifikátoru schopného rozeznat phishing od běžné elektronické komunikace. Součástí této práce je rešerše existujících řešení, rozbor problematiky výběru vhodných dat a návrh řešení. Představuji zde klasifikaci na základě několika zdrojů příznaků – vlastnosti textu emailu, domény a cílového webu. Zásluhou navrhované vícestupňové klasifikace není ovšem vždy nutné extrahovat všechny příznaky. Současně je tímto vyřešen kompromis rychlosti zpracování a klasifikační přesnosti. Předložena je i prototypová implementace, na které je porovnávána úspěšnost čtyř klasifikátorů z pohledu několika klasifikačních metrik. S klasifikátorem Random Forest, zdegenerovaným do baggingu, se mi podařilo v několika metrikách překonat většinu ostatních diskutovaných prací.

Klíčová slova Phishing, sociální inženýrství, strojové učení, automatická klasifikace

Abstract

The aim of this work is to analyze the techniques and methods used for phishing and to design an automated classifier capable of distinguishing phishing

from ordinary electronic communication. Part of this thesis is to research existing solutions, analyse the issue of selecting the appropriate data and design a solution. I present classification based on several sources of features – email text, domain and target website. Due to the proposed multi-stage classification, however, it is not always necessary to extract all the features. At the same time, this solution deals with the balance of processing speed and classification accuracy. I present a prototype solution, which is used to compare the success rate of 4 classifiers based on several classification metrics. With the Random Forest classifier, degenerated into bagging, I managed to outperform most of other discussed works, according to several metrics.

Keywords Phishing, social engineering, machine-learning, automated classification

Obsah

Seznam obrázků

Seznam tabulek

Úvod

Problém nevyžádané pošty je jedním z hlavních témat posledních dvou dekád v oblasti počítačové bezpečnosti a elektronické komunikace. Čím více začínáme využívat a spoléhat se na elektronickou komunikaci, tím silnější je potřeba mít k dispozici prostředky stínící nás od nevyžádané pošty, podvodů a útoků šířených tímto kanálem.

Vývoj takových nástrojů zaznamenal v posledních letech znatelného pokroku a tyto nástroje dosahují dobrých výsledků ve filtrování spamu, newsletterů a jednoduchých podvodů. V této práci se avšak zaměřuji na mnohem užší a znatelně nebezpečnější podmnožinu nevyžádané pošty – phishing. Tam, kde neúspěch v odfiltrování spamu znamená pouze mírnou nepříjemnost pro koncového uživatele, phishing je reálnou hrozbou vedoucí k finančním ztrátám, odcizování internetové identity až po infiltrace a krádeže interních informací ze společností. Phishing cílí na nejslabší článek zabezpečení – uživatele. Zneužívá nepozornosti, neopatrnosti a technické neznalosti uživatelů.

Cílem této práce je identifikovat phishingové techniky a metody a využít těchto znalostí k sestavení klasifikátoru, který by byl schopen spolehlivě tyto útoky identifikovat v běžné komunikaci. Ač je možné tyto útoky provádět přes několik různých kanálů (SMS zprávy, sociální sítě, instant messaging atd.), zaměřuji se na médium, kde je tato hrozba nejznatelnější – emaily.

Kapitola 1

Základní pojmy

1.1 Definice phishingu

Phishing je podvodná technika získávání citlivých údajů jako jsou hesla či čísla kreditních karet. Phishing patří mezi techniky sociálního inženýrství – zneužívá důvěru uživatele vydáváním se za legitimní instituce či osoby. Phishing na sebe nejčastěji bere formu emailu vyzývající uživatele k zadání citlivých údajů do webu odkazovaného v emailu. Ten uživatele přenese na podvodnou stránku speciálně vytvořenou tak, aby byla vizuálně nerozpoznatelná od svého originálu, za který se vydává. Ukázky takových stránek na obrázku 1.1

Narozdíl od invazivních útoků cílících na technické zranitelnosti a nedostatky, phishing je velice těžce automaticky detekovatelný a patří v dnešní době mezi nejefektivnější typy útoků.

Název phishing vznikl jako homofonní přesmyčka z anglického *fishing* – rybaření. Stejně jako u rybaření, útočník nahodí návnadu a vyčkává, kdo „se chytí“. V češtině se také občas používá označení „rhybaření“.

V této práci se zaměřuji na phishing ve formě podvodných emailů. Tento typ emailů budu dále označovat jako *phishingové* či *phish* emaily a jejich protiklad, „čisté“ emaily, budu označovat jako *ham* (výraz *ham* je převzat z domény spammingu, kde se takto čisté emaily z historických důvodů označují.)

1.2 Variace phishingu

1.2.1 Spear phishing

Na rozdíl od plošného phishingu, který útočí na masy a spoléhá na to, že alespoň malé procento ze zasažených podlehne, spear phishing cílí na konkrétního jednotlivce či firmu. Útočník posbírá informace o svém cíli a využije je ke zvýšení důvěryhodnosti a šance na úspěch. Útočník např. zjistí informace o struktuře firmy, hierarchii a jménech zaměstnanců a využije je k vytvoření emailu, který budí dojem, že pochází od někoho zevnitř firmy, a je tak

(a) podvodný Facebook (b) podvodný eBay (c) podvodný Amazon

Obrázek 1.1: Ukázky podvodných stránek

mnohem snadněji uvěřitelný. Takovéto techniky bylo využito například pro zavedení slavného viru *Stuxnet*.

1.2.2 Whaling

Whaling neboli velrybaření je odnož spear-phishingu zaměřená na vyšší management společnosti (tito lidé jsou slangově označováni jako „velké ryby“). Zpráva je konstruována tak, že zdánlivě reaguje na nějakou událost ve firmě (nedávné uzavření obchodu, nákup techniky, legislativní změny). Typickým příkladem bývá zpráva zdánlivě pocházející od městského/daňového úřadu/-FBI vyzývající k urgentnímu vyplnění údajů. Důvodem cílení na vyšší management je, že tito lidé mají přístup do většiny podsystémů a teče přes ně velké množství firemních tajemství.

1.2.3 Pharming

Pharming neboli DNS-Based Phishing je technika obcházející nutnost uživatele otevřít link v emailu, který odkazuje na podvrženou stránku. Toho lze dosáhnout změnou způsobu, jakým se vyhodnotí DNS dotazy tak, aby po zadání legitimní adresy do prohlížeče došlo k zobrazení podvrhu. DNS dotazy se vyhodnocují dvojím způsobem.

1. **Soubor hosts**
   Hosts je v mnoha systémech prvotní zdroj informací o překladu doménového jména na konkrétní IP adresu. Útočník sem může podvrhnout adresu na podvodný web. Po zadání adresy do prohlížeče `www.anyweb.com` je kontaktován server na podvržené adrese a uživatel se nic nedozví (v adresním řádku zůstává původní `www.anyweb.com`). Obranou je používání HTTPS a certifikátů.

2. **DNS server**
   Další variantou je změnit DNS server, kterého se počítač oběti doptává na IP adresy. Pokud se útočníkovi podaří kompromitovat router oběti, může přenastavit adresu DNS serveru na svůj server a všechny stroje v síti pak budou přesměrovány na podvrženou adresu. Často není potřeba router nijak složitě dobývat – častým jevem je, že domácí routery mají nastaveno defaultní jméno a heslo. Přenastavit pak DNS server je jednoduché.

1.3 Historie phishingu

První popis konceptu phishingu je připisován Jerrimu Felixovi a Chrisu Hauckovi za jejich článek *System Security: A Hacker's Perspective* [2] z roku 1987, ve kterém zkoumali možnosti imitace služeb třetích stran. Samotný termín phishing je pak spojován s útoky na společnost America Online (AOL), při kterých byly phishingové techniky poprvé nasazeny.
Společnost AOL ve svých začátcích zprostředkovávala dial-up připojení k Internetu. Jejích služeb využívaly miliony Američanů a tak se brzy dostala do hledáčků hackerů. Jedním z nich byl i 17letý Koceilah Rekouche vystupující pod přezdívkou *Da Chronic*. AOL kromě připojení provozovala i chatovací platformu, ve které mohli sami uživatelé vytvářet tématické chatovací místnosti. Slabé moderování těchto místností vedlo k tomu, že se začaly mimo jiné objevovat i místnosti s pedofilní tématikou a místnosti pro výměnu dětské pornografie mezi uživateli. AOL administrátoři zasahovali proti místnostem s tématikou hackingu, ale pedofilní místnosti zůstávaly bez povšimnutí, jak tvrdí Rokouche v dokumentaci [1] programu *AOHell*, který napsal v roce 1995, za účelem napadení AOL.

> "One day I decided I had had enough. AOL constantly closed the "Hackers"Member room, but refuses to do anything about all the pedophilia rooms. I once IMed TOSAdvisor and asked him why he closes the Hacker room, but does not close the kiddie porn rooms. He did not reply, instead he cancelled my account. I guess we see where AOL's priorities lie. If AOL is going to do nothing about this type of sick behavior then I will do everything I can to screw AOL up. I think having 20,000+ idiots using AOHell to knock people offline, steal passwords and credit card information, and to basically annoy the hell out of everyone is a good start."

AOHell umožňoval uživateli se vydávat za zaměstnance AOL a klamat uživatele. Od nich se snažili získat čísla kreditních, sociálních či ID karet a přístupové údaje do jejich účtů. K velkému úspěchu AOHell vedly tři faktory

[1] `http://www.aolwatch.org/chronic2.htm`

1. Vágní politika ověřování nově založených účtů ze strany AOL.
2. AOHell byl navržen tak, aby s ním mohl operovat i úplný laik. UI programu bylo zjednodušeno tak, aby k provedení automatizovaného útoku stačilo pár kliknutí a následování jednoduchých instrukcí. To dokládá i obrázek jednoho z oken konfigurace programu 1.2. Obsahoval dokonce soubory s nápovědou s vysvětlením konceptu phishingu a vysvětlivkami, jak útoky provádět efektivně. Všechny tyto faktory spolu s faktem, že používání AOHellu bylo v té době téměř nepostihnutelné a tudíž bezpečné, vedlo k rychlému šíření programu.
3. V dnešní době, kdy se povědomí o phishingu rozšiřuje, začínají být lidé na síti více ostražití. V devadesátých letech minulého století však na tuto hrozbu nebyl nikdo připraven. AOHell navíc cílil na nově příchozí uživatele, kteří s prostředím internetu měli malé či vůbec žádné zkušenosti.

Obrázek 1.2: Konfigurační obrazovka AOHell [1]

V návaznosti na tyto útoky AOL provedl řadu bezpečnostních opatření. Mezi nimi například začal informovat své uživatele o tom, že jeho zaměstnanci nikdy po uživatelích nebudou vyžadovat čísla karet ani přihlašovací údaje. Toto oznámení se stalo standardem a vídáme ho v různých formách dodnes.

> "Employees of our company will never ask for your password or billing information"

V následujících letech doznal phishing masivního rozmachu. Častým cílem se staly online platební systémy. Začaly se vynořovat i útoky přižívujících se na exponovaných událostech ve světě.

- **9/11 ID check**
  Teroristické útoky 11. září 2001 byly zneužity pro sběr osobních údajů pod zástěrkou kontroly obětí útoků.

- **Ebola virus bait**
  Když v roce 2014 dosahoval virus Ebola svého vrcholu, sítí se začaly šířit emaily, které se vydávaly za varovnou zprávu od zdravotní instituce. Tyto emaily typicky nabádaly uživatele k otevření přílohy s instrukcemi k preventivním krokům proti viru Ebola. Opět se jednalo o sběr osobních údajů a modifikaci souboru *hosts*.

- **MH17**
  Nejasnostmi opředené sestřelení civilního letu MH17 Malajské aerolinky v roce 2014 stálo život téměř 300 lidí. Nejasné okolnosti incidentu byly skvělým prostředkem k nalákání informací-chtivých uživatelů na podvodné stránky. Z nich se distribuoval virus *Zeus* či uživatele přesměrovaly na jiné weby (především s pornografickou tématikou) za účelem umělého navýšení návštěvnosti.

- **Goodbye video**
  Po smrti herce Robina Williamse se sociálními sítěmi šířily odkazy na video údajného posledního rozloučení zachyceného na videu. Odkaz ovšem vedl na podvodnou verzi BBC, ze které se útočníci snažili uživatele dostat na podvodné dotazníky a formuláře.

Kapitola 2

Techniky phishingu

2.1 Zneužití dat

Studie Googlu [3] zkoumala, jak útočníci nakládají s odcizenými emailovými účty. Dle této studie útočníci vstoupí do účtu ve 20 % případů do 30 minut a v 50 % případů do 7 hodin. Po přístupu do účtu je nutné (převážně) manuální hledání dalších citlivých údajů – mezi nejčastěji hledanými jsou údaje bankovních účtů a převodů, hesla k dalším službám a kontakty na další potenciální oběti. Tyto kontakty jsou následně využity k odesílání zpráv typicky obsahujících žádost o zaslání malé částky na účet útočníka. Pokud útočník použije osobní údaje získané např. z emailové komunikace s touto osobou, dokáže vytvořit personalizovaný email, který je navíc odeslán z legitimní adresy oběti. Lidé v kontaktech oběti jsou tak statisticky 36krát náchylnější k podlehnutí phishingu.
Z tabulky 2.1 je zřejmé, že cílem phisherů jsou především instituce přímo spojené s penězi či služby, které jsou bohatým zdrojem osobních údajů.

Tabulka 2.1: Top 10 Identified Target - June 2016

1.	PayPal
2.	AOL
3.	Apple
4.	Facebook
5.	eBay, Inc.
6.	Google
7.	Yahoo
8.	Itau
9.	WalMart
10.	Bradesco

zdroj: PhishTank statistics - June 2015[4]

2.2 Typické znaky a používané techniky

- **Skrývání linku za obrázek**
  Místo textu se použije obrázek, po jehož rozkliknutí jsme přesměrováni na podvodný web.

- **Rozdílný text odkazu**

  Text odkazu (text mezi <A> tagy) se liší od samotné adresy odkazu. Link tak může vypadat jako `www.legitweb.com` a přitom vést na `www.fraudulentweb.com`.

- **Subdomény**
  `www.yourbank.example.com` budí dojem, že vede na sekci *example* domény *yourbank*. Domény se avšak vyhodnocují zprava doleva a tento odkaz ve skutečnosti vede do sekce *yourbank* na podvodné doméně *example*.

- **Homografní linky**
  Jedná se o linky takové, které vizuálně vypadají velice podobně jako jejich legitimní varianta. Toho lze docílit několika způsoby.

  - **úmyslné typografické chyby**
    Mezi takové patří vynechání písmen (např *twiter.com* vs. *twitter.com*) či přehození písmen (*senzam.cz* vs. *seznam.cz*).
  - **homografní písmena**
    Jednoduchým příkladem je záměna čísla 0 a písmene O nebo záměna „l“-malého L a „I“-velkého „i“.
  - **mezinárodní doménová jména**
    Adresy mohou obsahovat znaky z různých abeced. Např. arabské, čínské či azbuky. Existují tak znaky, které jsou vizuálně totožné, ale mají jiný Unicode kód. Například znak U+0430 – malé „a“ v azbuce je vizuálně totožný se znakem U+0061 – malé „a“ v latince. Jejich záměnou tak můžeme vytvořit link, který je vizuálně nerozpoznatelný od originálu.

- **Nechat uživatele adresu složit**
  Ve snaze oklamat automatické detektory, které kontrolují odkazy v textu, útočník vyzve příjemce zprávy, aby zadal do adresního řádku adresu sestávající z několik částí. *Zkopírujte následující adresu do adresního řádku 'fake fraudulent web.com' bez mezer.* Automat tak není schopen v textu rozpoznat link.

- **Zkracování domén**
  Služby pro zkracování URL se staly populární především kvůli omezením na délku textu v mobilních telefonech a na Twitteru. Je tím de facto

vytvořena proxi, díky které oběť netuší, kam link doopravdy vede dříve, než na něj klikne. Díky popularitě těchto služeb, oběť ve většině případů nepojme podezření tak, jak by tomu mohlo být, kdyby link vedl napřímo.

- **Přesun obsahu zprávy do přílohy**
  Po osobním rozhovoru s odborníky ze společnosti Excello[2], zaměřující se na problematiku nevyžádané pošty(především pak na spam), jsem byl seznámen s relativně novou taktikou, jakou zpozorovali ve svých filtrech. Aby útočníci obešli klasifikátory analyzující text emailu, jednoduše do emailu žádný text nedávají. Uživateli přijde prázdný email. Ten má ovšem k sobě připojenu přílohu. Do té útočník přesunul text své obvyklé kampaně. Tyto přílohy bývají v obvyklých dokumentových formátech – PDF, DOC, DOCX atd. Útočník tak může využít možností či nedokonalostí těchto formátů k ještě lepšímu maskování a imitaci. A v první řadě takto efektivně ztíží automatickou klasifikaci – jedinou obranou je algoritmus, který umí ze všech těchto formátů vyextrahovat důležité informace a dokáže obejít všechny obfuskační techniky, které tyto formáty dovolují.

[2] `https://www.excello.cz`

Kapitola 3

Phishing v praxi

3.1 Efektivita phishingu

Phishing a jeho odnože jsou vysoce efektivním nástrojem útočníků. Studie provedená v Google [3] naznačuje, že některé dobře zpracované phishingové kampaně dosahují až 45% úspěšnosti. Za úspěšnost/efektivnost zde považujeme procento uživatelů, kteří odešlou své citlivé údaje útočníkům. Průměrná kampaň dosáhne 14 % a i ty nejhorší, špatně provedené kampaně mají 3% efektivnost. Toto číslo se může na první pohled zdát jako malé až zanedbatelné, avšak s přihlédnutím k faktu, že útočník je schopen poměrně snadno rozeslat miliony emailů, pak efektivita 3 % znamená obrovské množství odcizených údajů a potencionálních škod.

3.2 Demografie útoků

V sekci 1.2 bylo popsáno několik kategorií phishingových útoků, z nichž některé se liší cílením na jiné typy uživatelů. To nám ovšem nedává žádnou informaci o tom, jak aktuálně phishingová scéna vypadá a kde je množství útoků nejvyšší. Z tabulky 3.1 sestavené z dat zveřejňovaných společností Kaspersky je patrné, že aktuálně nejvíce útoků je směřováno na Brazílii a Indii. Co se týče útoků na firmy, z dat společnosti Kaspersky za rok 2014 3.2 se překvapivě nezdá, že by útočníci cílili na menší společnosti (což by se mohlo zdát logické s přihlédnutím k tomu, že většina malých firem nemívá budget na pravidelná bezpečnostní opatření a technické prostředky), ale útoky jsou rozprostřeny takřka rovnoměrně.

3.3 Frameworky a nástroje

Phisheři podnikají útoky za účelem získání citlivých údajů. Takto sesbíraná data se pak snaží monetizovat. Data mohou dále přeprodávat na černém trhu

Tabulka 3.1: Top 10 zemí dle procenta napadených uživatelů - Q1 2015

Brazílie	18.28
Indie	17.73
Čína	14.92
Kazachstán	11.68
Rusko	11.62
Spojené arabské emiráty	11.61
Austrálie	11.18
Francie	10.93
Kanada	10.66
Malajsie	10.40

Data jsou vyjádřena jako procento uživatelů, na jejichž počítači byl zachycen phishingový útok k celkovému počtu uživatelů Kaspersky produktů v dané zemi. Zdroj: Kaspersky [5]

Tabulka 3.2: Podíl emailů identifikovaných jako phishing dle velikosti zasažené společnosti, 2014

Company Size	2014
1–250	1 in 1,401.5
251–500	1 in 1,253.5
501–1000	1 in 1,248.4
1001–1500	1 in 1,639.6
1501–2500	1 in 1,621.2
2501+	1 in 1,685.4

zdroj: Symantec Thread Report [6]

nebo častěji se pokusí sami data zúročit (převod peněz na osobní účty, rozšíření phishingové sítě, ...). Otázkou pak zůstává, kolik musí útočník zaplatit za spuštění takového útoku – jak je časově a finančně náročné takový útok uskutečnit, jakými technickými znalostmi musí útočník disponovat a vyplatí se vůbec?
Phishing není žádná novinka a tak není divu, že i nástroje k jeho provádění se vyvinuly z malých a neotesaných scriptů do podoby celistvých produktů. Na trhu se nachází hned několik open-source nástrojů dostupných zdarma.
Některé z nich disponují CLI (command line interface), jiné mají kompletní UI ve formě webové aplikace. Všechny mnou zkoumané nástroje nabízely robustní nastavení, dokázaly generovat přehledné statistiky, vytvářet celé phishingové kampaně v pár krocích a disponovaly hned několika předpřipravenými šablonami pro emaily. Dashboard jednoho z nástrojů je na obrázku 3.1.
Některé z dostupných frameworků:

- GoPhish [3]
- Cartero [4]
- Phishing Frenzy [5]
- SPF (Speed Phish Framework) [6]
- Social-Engineer Toolkit (SET) [7]
- Ninja phishing framework [8]

Proces vytvoření kompletní kampaně tak ve většině případů zahrnuje následující kroky:

1. Vytvoření šablony emailu
2. Vytvoření podvodné stránky (nebo naklonování originálu)
3. Nastavení emailového serveru
4. Odeslání emailů na specifikované adresy (většina nástrojů umožňuje i jejich sběr)
5. Vyhodnocení výsledků

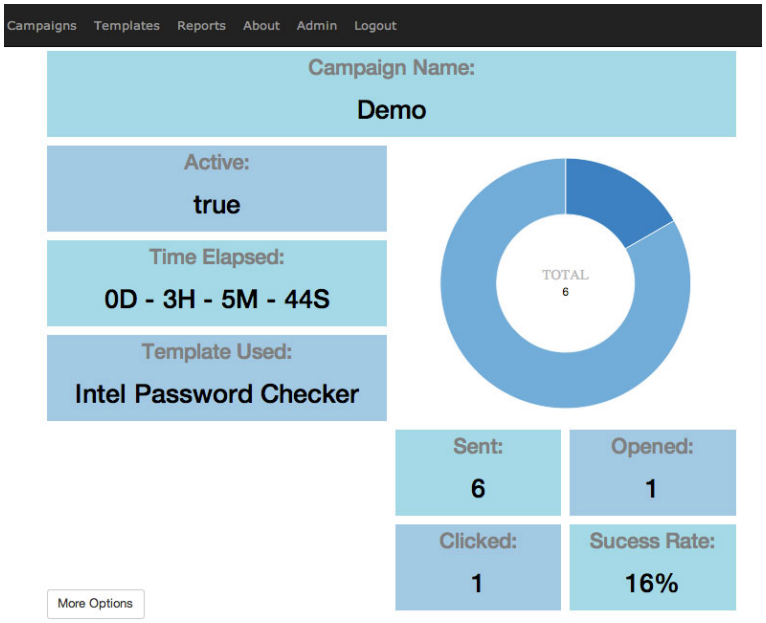
Výše zkoumané frameworky byly ty nejsnáze dostupné a zprovoznitelné (sloužící především jako dobrý nástroj pentesterů). Lze předpokládat, že na černém trhu deepwebu budou k dostání ještě profesionálnější a mocnější nástroje.

3.4 Prevence napadení

Phishing spadá do kategorie sociálního inženýrství – cílí na slabosti koncového uživatele. Mimo technická řešení je tedy nutné také uživatele/zaměstnance vzdělávat v možnostech prevence phishingového útoku.
Mezi základní opatření patří:

- **Nedůvěra k cizím zdrojům**
  Především u emailů došlých od neznámých lidí je třeba zvýšené ostražitosti – kontrolovat adresu odesílatele a nikdy nesdělovat své citlivé údaje po emailu. Pokud email obsahuje odkazy, kontrolovat na jakou adresu vedou. Zadávám-li na cílovém webu své údaje, vždy zkontroluji, že má platný certifikát.

[3] `https://getgophish.com/`
[4] `http://section9labs.github.io/Cartero`
[5] `http://www.phishingfrenzy.com`
[6] `https://github.com/tatanus/SPF`
[7] `https://www.trustedsec.com/social-engineer-toolkit`
[8] `http://sourceforge.net/p/ninjapf/home/Home`



Obrázek 3.1: Obrazovka vyhodnocení kampaně v nástroji Phishing Frenzy

- **Bezpečné nakládání s hesly**
  Hesla nikdy nikomu nesdělovat, neposílat. Dodržovat zásady silných hesel. Hesla si často měnit a zásadně nepoužívat stejná hesla do více služeb.

- **2-fázové ověřování**
  Two-factor authentication je metoda ověření identity pomocí kombinace dvou oddělených komponent. Příkladem budiž kreditní karta a pin. Jeden bez druhého je k ničemu. V případě online identity se nejčastěji využívá znalost hesla a vlastnictví fyzického zařízení. Mnoho online služeb dnes již umožňuje 2-fázové ověřování, které po zadání hesla vyžaduje ještě zadání jednorázového hesla zaslaného na mobilní telefon uživatele. I přestože útočník ukradne heslo, nebude se schopen do služby přihlásit.

- **V případě incidentu zasáhnout co nejrychleji**
  V případě podezření, že se k citlivým údajům dostala nepovolaná osoba, je třeba neprodleně znemožnit jejich zneužití – přístupová hesla změnit a změnit je i na dalších službách, kde používám stejné heslo nebo jeho variaci.

Kapitola 4

Aktivní boj proti phishingu

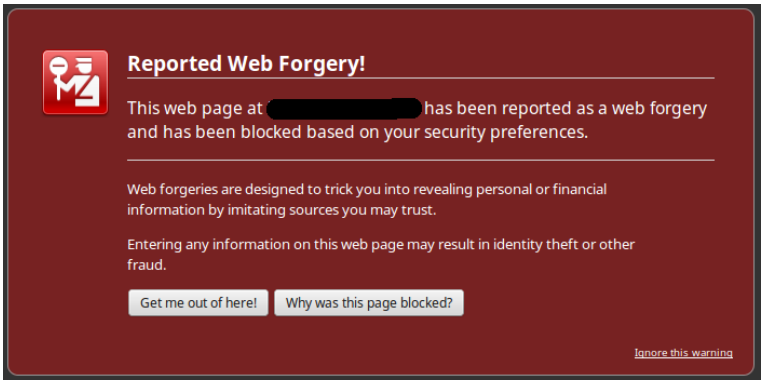
Tato sekce je zaměřena na odlišnou stranu boje s phishingem. Většina dostupných opatření zmiňovaných v této práci i na trhu je zaměřena na pasivní obranu proti phishingu. Za pasivní prvky považujeme všechny filtrační mechanizmy nasazené buď v elektronické schránce uživatele či přímo na mailovém serveru.
V této sekci se proto zaměřím na metody a iniciativy působící v oblasti aktivní obrany a s tím spojenými úskalími. V této oblasti se již často mažou hranice mezi spammingem, phishingem a scammingem.

4.1 Konvenční metody

4.1.1 Blokace podvodného webu

První nejzákladnější a nejrychlejší metodou (v případě phishingu s falšovanou webovou stránkou) je přidání tohoto webu na blacklist. K tomu dojde zpravidla na popud uživatele, který takový email dostal a nahlásil ho, nebo automaticky po vyhodnocení automatem. Těchto blacklistů je celá řada (vydavatelé prohlížečů si je často spravují sami). Zmiňme například blacklist *Safe Browsing* [9]. Tento blacklist spravovaný společností Google je využíván v prohlížeči Chrome a Firefox a zároveň obsahuje veřejné API [10], přes které je možné se ho dotázat na hodnocení jakéhokoli webu. *Safe Browsing* kromě nahlášených webů aktivně prohledává internet a snaží se odhalit podvodné weby dříve, než je někdo nahlásí (a potencionálně jim naletí).
Jakmile se uživatel pokusí otevřít web, který se nachází na blacklistu, je mu zobrazeno varovné oznámení a je mu doporučeno na web nevstupovat viz obrázek 4.1.

[9] `https://www.google.com/transparencyreport/safebrowsing`
[10] `https://developers.google.com/safe-browsing/`



Obrázek 4.1: Varovné oznámení v prohlížeči Chrome

4.1.2 Blokace IP adresy odesílatele

Dalším způsobem je pokusit se zablokovat přímo stroj, ze kterého je útok veden. Ať již stroj, ze kterého byl odeslán email, či na kterém běží podvodný web. V obou případech je snadné dohledat IP adresu (v případě mailu je tato informace v hlavičkách). S touto informací lze dohledat, pod kterého ISP [11] tato IP adresa spadá. Ten je kontaktován s informací, že z IP adresy, kterou spravuje, jsou vedeny útoky a ať zjedná nápravu. To zahrnuje v první fázi zjistit, zdali se jedná opravdu o útočníka nebo o infikovaný stroj. V ideálním případě následně proběhne napomenutí či odebrání IP adresy.

4.2 Nekonvenční metody

Při rešerši jsem ovšem narazil i na radikálnější metody. Ty se vesměs snaží útočníkům znesnadnit jejích práci a snížit jejich profitovost podsouváním falešných signálů. Ač některé zde zmíněné metody jsou aplikované pouze na některý z typů podvodů(spamming scamming, phishing), můžeme z nich vzít inspiraci a poučení a pokusit se je aplikovat i na ty ostatní (např. aplikace scambaitingu na phishing).

4.2.1 Scambaiting

Scambaiting je technika, při které se aktér vydává za nic netušícího uživatele a potenciální oběť scammingu. Koná tak za účelem mrhání času a zdrojů útočníka, sběru informací o útočníkovi a jeho metodách. Následně tyto informace veřejně odhaluje, aby na jeho podvody nenaletěl někdo jiný.

Nejprve se podívejme, jak takový scam vypadá. Útočník pošle potenciální oběti email, ve kterém žádá pomoc, nabízí výhodnou spolupráci atd. Cílem

[11] Internet Service Provider - zprostředkovatel internetu

je v oběti vyvolat pocit, že se snadnou cestou (obvykle za malý obnos peněz) může dostat k velké sumě peněz. Toto jsou některé z těchto modelů:

- **Nigerijský scam**
  Někdy též nazývaný „Nigerijský 419 scam“ či zkráceně „Nigerijský princ“ je jeden z nejznámějších. Útočník se vydává za manželku/syna/osobního asistenta nedávno zesnulého Nigerijského(či z jiného afrického státu) vládce. Žádá o pomoc s převedením jeho fondů mimo stát prostřednictvím účtu oběti. Výměnou je mu slíben podíl z těchto několikamilionových fondů. V následujících emailech pak podvodník žádá malé obnosy peněz pod záminkou zaplacení poplatků za převod peněz z fondů.

- **Závěť po zesnulém**
  Oběť je kontaktována emailem, ve kterém se útočník vydává za právníka zodpovědného za vykonání závěti nedávno zesnulého muže. Dodává, že se pokoušel dohledat potomky a dědice majetku zesnulého, ale žádné neměl. V závěti ovšem našli zmíněné právě jméno oběti. Po předání osobních informací a informací o bankovním účtu pak údajně zašlou peníze z dědictví na účet oběti.

- **Romance**
  Tento typ podvodu cílí na uživatele, jejichž emailové adresy se objevily na internetových seznamkách. Útočník předstírá romantické záměry a chce se více sblížit. Během tohoto procesu se snaží z oběti vymámit osobní údaje, bankovní účty, čísla pasu atd.

- **Nájemný zabiják**
  Útočník se vydává za nájemného zabijáka, který byl najat osobou blízkou oběti. Oběť je upozorněna, že si tato osoba u něj objednala její smrt. Zabiják je ale ochoten toto neudělat, pokud ho oběť přeplatí.

Zdánlivě zastaralé modely
Tyto modely se již objevují s drobnými obměnami řadu let. Může být s podivem, proč útočníci nepoužívají důmyslnější a důvěryhodnější formy. Na takto zjevné podvody nemůže naletět příliš velké procento lidí. Tento fenomén rozebírá Harley Cormac ve své práci *Why do Nigerian Scammers Say They are from Nigeria?*[7]. V ní prezentuje jednoduché zdůvodnění – útočníci také trpí velkým poměrem *fales positives*. Útočníkovým nejcennějším zdrojem je čas. Rozeberme modelový případ, kdy útočník rozešle svou podvodnou zprávu 10 000 lidem. Tato zpráva vypadá velice autenticky a věrohodně – odpoví tak na ní 5 000 lidí. Útočník musí s každým z těchto lidí začít dlouhý a náročný proces, kdy se ho dalšími zprávami snaží přimět, aby mu zaslal peníze. V průběhu této fáze ale mnoho lidí zbystří, uvědomí si, že se jedná o podvod a konverzaci ukončí (ty označujeme jako *false positives*). Útočníkovi se tak podaří dovést až k zaslání peněz např. pouhých 5 lidí.

Nyní uvažujme stejný případ – zpráva zaslaná stejným 10 000 lidem. Tentokrát ovšem použije nějaký zdánlivě zřejmý podvodný email – například model Nigerijského prince. Úmyslně do něj umístí i několik gramatických chyb (což slouží i pro zmatení klasifikátorů). Na takhle zřejmý podvod odpoví pouhých 20 lidí. Tímto se ovšem útočníkovi podařilo velice rychle vyselektovat skupinu silně nezkušených a důvěřivých uživatelů. Do finální fáze se mu opět podaří dostat 5 lidí. Oproti minulému případu ho to ovšem stálo mnohem méně času a energie a zvýšil tak silně svou profitovost.

Průběh Scambaitingu

Při samotném scambaitingu pak aktér předstírá, že podvodné zprávě věří. Snaží se s útočníkem udržet co nejdelší „řetěz“ zpráv a nenápadně z něj dostat co nejvíce informací, zatímco mu sám podsouvá informace falešné. Tím plýtvá časem a energií útočníka, který tak tento čas nemůže věnovat opravdovým obětem, ze kterých by měl zisk.
Komunita zabývající se těmito aktivitami využívá řady triků, jak toho dosáhnout a jak zmařit co nejvíce času útočníka – mezi oblíbené patří požadování vyplnění zdlouhavých formulářů s detaily či požadování zaslání dalších důkazů autentičnosti/fotek. Na webu serveru 419Eater[12], která slouží jako platforma pro scambaiting, nalezneme celou sekci věnovanou tipům, jak tuto aktivitu provádět efektivně.
Zajímavým zdrojem je pak video-přednáška Jamese Veitricha *This is what happens when you reply to spam email*[8], ve které popisuje své zkušenosti a bizarní situace vzniklé scambaitingem.

Automatizace

Našel jsem i několik zmínek o pokusu tento proces automatizovat za pomoci umělé inteligence. Jedním z nich je projekt *Autobait*[13]. Umělá inteligence v něm obsahovala několik modelových „personalit“, každá sestávající z 40 atributů. Autor uvádí, že projekt dočasně pozastavil i z toho důvodu, že veškerý generovaný provoz šel přes jeho ISP a ten ho vzhledem k obsahu zpráv blokoval pro podezření ze spammingu.

Dalším zajímavým projektem v tomto odvětví je projekt Rogera Andersona *The Jolly Roger Telephone Company* [14], který se zaměřuje na nevyžádané telemarketingové telefonní hovory. Jeho umělá inteligence v sobě nese přednahrané zvukové záznamy a dokáže automaticky reagovat na telemarketera na druhé straně linky. Stejně jako v předchozích případech platí, že pokud ztrácí čas rozhovorem s automatem, nemůže tento čas věnovat na opravdové lidi.

[12] `http://www.419eater.com/html/baiting.htm`
[13] `http://www.autobait.com`
[14] `http://www.jollyrogertelco.com`

Využití pro phishing
V případě phishingu zpravidla k žádné vícefázové komunikaci nedochází. Snažit se tak mrhat útočníkovým časem nedává smysl. Můžeme však převzít myšlenku podsouvání falešných informací. Mohli bychom phisherům záměrně podsouvat přihlašovací údaje ke speciálním účtům k tomu vyhrazeným. U těch bychom mohli monitorovat, z jakých IP adres se na ně přistupuje. Mohli bychom také sledovat, co phisheři na daných účtech dělají a hledají. V případě emailových účtů phisheři po vniknutí často hledají další citlivé informace a především další přístupové informace. Pokud bychom podvrhli i obsah těchto emailových schránek, mohli bychom phishery uvrhnout do nekonečné smyčky procházení falešných účtů.

4.2.2 Make Love Not Spam

V roce 2004 představila společnost *Lycos Europe* (zabývající se převážně e-commerce, web hostingem a komunikačními nástroji) program nazvaný *Make Love Not Spam*. Jednalo se o nástroj pro uživatele pro účely kolaborovaného boje proti spamu.

Program samotný fungoval jako screensaver[15] – ve chvíli, kdy nebyl počítač delší dobu využíván a zahálel, spustil se screensaver. Ten záhy začal provádět DDoS útok na známé webservery spammerů, čímž způsoboval jejich nedostupnost. Program tak fungoval na stejném principu jako klasické botnety. S tím rozdílem, že uživatelé si byli tohoto softwaru vědomi a mohli se rozhodnout, zda se v těchto aktivitách chtějí podílet.

Lycos Europe přestal tento program distribuovat po pouhých dvou měsících. Důvody byly dva: společnost se dostala pod palbu kritiky ze strany internetových a bezpečnostních expertů. Druhých důvodem bylo, že touto aktivitou naštvali velké množství spammerů, kteří začali podnikat na služby společnosti Lycos rozsáhlé protiútoky.

Využití pro phishing
Tato taktika by byla proti phishingu stejně účinná – systematický DDoS nápor na weby phisherů by způsobil jejich nedostupnost. Pokud by web nebyl dostupný, žádný uživatel by se na něj nedostal, ergo nikdo by podvodu nepodlehl.
DDoS útoky s sebou nesou několik zásadních problémů, kvůli kterým nelze tuto techniku doporučit.

- Morální správnost takového počínání je zpochybnitelná. De facto se tímto snižujeme k používání podobných technik, který používají sami útočníci.

[15] screensaver=spořič obrazovky

- Tento útok generuje obrovské množství neužitečného síťového provozu. Pokud je v síti nasazen nějaký systém monitoringu, může toto počínání vyhodnotit (korektně) jako nebezpečné. Zpravidla pak ISP takového uživatele odpojí od přístupu k Internetu.
  Stejně tak dojde k zahlcení sítě na straně cíle protiútoku. Ač se může podařit vyřadit z provozu webserver phishera, následky přetížení sítě můžou pocítit i ostatní, zcela nevinní, uživatelé Internetu spadající pod stejného ISP.

- I pokud známe IP adresu webserveru, ze kterého přichází spam/běží na něm phishingový web, v žádném případě to neznamená, že se jedná o fyzický stroj patřící útočníkovi. Běžnou taktikou je pro tyto účely využívat napadené počítače nic netušících uživatelů (botnet).

4.2.3 Spamování spammerů

Zajímavým nápadem je pak obrátit techniky spammerů proti nim samotným.

Projekt *myTrashMail.com*[16] má na svém webu seznam náhodně vygenerovaných emailových adres. Cílem je využít spamerské crawlery (malé programy určené k systematickému prohledávání Internetu a sběru emailových adres z webů), aby do svých databází ukládaly velké množství těchto podvržených adres. Pokud chcete tento projekt podpořit, umístíte na svou webovou stránku odkáže na tento web. Pokud na vaši stránku takovýto crawler zavítá, dostane se tak přes odkaz i na stránku tohoto projektu, ze kterého sesbírá falešné adresy. Tímto způsobem vyčerpávají spameři svoje zdroje na prázdné emailové schránky. Pokud bychom ale místo náhodně generovaných adres zveřejňovali spamerské emailové adresy, dosáhli bychom efektu přesměrování spamu do schránek samotných spamerů.

Podobný nápad je přihlašování spamerských mailboxů k odběru náhodných newsletterů. Přihlášením emailu k odběru novinek u desítky např. internetových obchodů lze dosáhnout velice rychlého zaplňování těchto mailboxů.

Využití pro phishing

Tyto techniky již považuji za hranicí morálnosti. Cílené přihlašování emailu k odběru newsletterů je v některých státech dokonce považováno za akt obtěžování a je považováno za nezákonné.

[16] `http://www.mytrashmail.com/anti_spam.aspx`

Kapitola 5

Metodologie

5.1 Data

Pro detekci phishingu, stejně tak jako pro jakoukoliv jinou klasifikační úlohu, jsou naprosto klíčová trénovací data a jejich kvalita. Rozmanitost a rozsah vstupních dat hraje silnou roli v kvalitě výsledného klasifikátoru a především pak při jeho následném užití v praxi na datech nových.

V této sekci naleznete srovnání veřejně dostupných datasetů vhodných pro tento typ úlohy. Při výběru byly brány v potaz jak kvalitativní hodnoty datasetů, tak jejich využití v předešlých projektech. Použití stejných datasetů umožní relevantnější zhodnocení a porovnání výsledků s ostatními pracemi.

5.1.1 Klasické benchmarkové datasety

Dataset Joseho Nazaria
Jose Nazario je americký bezpečnostní analytik a výzkumník. Od roku 2004 sbíral a ručně vybíral phishingové vzorky ze svých emailových schránek a schránek svých kolegů. Vznikl tak unikátní dataset, který vyniká svou kvalitou a velikostí. Dataset byl několikrát upravován a rozšiřován. V době psaní této práce jsem z datasetu byl schopen získat 8229 phishingových emailů. Jedná se o největší veřejně dostupný [17] dataset phishingu. Pro svou velikost a kvalitu byl tento dataset v minulosti použit v řadě výzkumných projektů, zahrnující práce Fette et al. [10], Abu-Nimeh et al. [11], Gonzalez et al. [12] nebo Toolan et al [13].

Enron dataset
Tento dataset byl původně vytvořen při vyšetřování bankrotu energetické společnosti Enron Corporation v roce 2001 nařízeným federálním úřadem pro regulaci energie. Jedná se o kompletní emailovou komunikaci 158 zaměstnanců

[17] `http://monkey.org/~jose/phishing`

a obsahuje přes půl milionu zpráv. Tento dataset byl následně uveřejněn pro výzkum a akademické účely. Dostupný je na adrese z webu Carnegie Mellon University [18]. Korpus od té doby prošel několika filtracemi (odstraněna byla čísla kreditních karet, čísla sociálních a ID karet, telefonní čísla atd.). Zachycena byla kompletní komunikace. V datech se tak nachází jak *čisté*(ham) emaily, tak i nevyžádané emaily. Tento dataset byl následně roztříděn Ionem Androutsopouluem na ham a spam emaily [19]. Ačkoli spam emaily již dále nejsou roztříděny na phishingové, a tudíž pro klasifikační účely nevhodné, čisté emaily jsou velice cenné a jedná se o jeden z největších veřejně dostupných datasetů legitimních emailů.

SpamAssassin corpus
SpamAssassin je open-source anti-spamový projekt. V rámci tohoto projektu se autoři rozhodli zveřejnit část korpusu (jak spam, tak ham) z roku 2002 a 2003. Ham emaily jsou zde rozděleny na *easy* a *hard*. Do první kategorie spadají emaily, které je snadné odlišit od spamu (neobsahují žádné spam signatury, HTML atd.). Do té druhé pak spadají ty, které sdílí nějaké znaky se spamem – obarvený text, HTML, fráze frekventované ve spamu atd.
Po sloučení všech ham emailů pak dostáváme 6950 zpráv.

5.1.2 Nevýhody klasických datasetů

Výše zmíněné datasety sdílejí jednu nevýhodu – jsou zastaralé. Stejně jako se mění používaná mluva, mění se v čase i obsah emailů. Se zvyšujícím trendem zapojování informačních technologií do každodenního života, vkládáme do našich emailů více odkazů(jak ve smyslu hyperlinků, tak ve smyslu verbálním) na online služby a technologie. Přibývá zkracování výrazu do zkratek, zkracuje se délka, ale zvyšuje množství atd.

Na straně phishingu je tento problém ještě výraznější. Se zlepšováním detekcí na straně emailových providerů a filtrů jsou útočníci nuceni své taktiky měnit a vylepšovat. Starší dataset tak hůře reflektuje reálné hrozby.

Dalším, možná ne zcela zjevným požadavkem je, aby dataset byl co nejpestřejší – chtěli bychom vzorek, který pokrývá velké spektrum sociálních, věkových i geografických skupin. Tuto vlastnost bohužel tyto datasety do jisté míry postrádají.

Proč je tedy takový problém sehnat nové vzorky ham a phish emailů? V případě ham emailů je v první řadě email považován za velice citlivé médium, často nesoucí velice citlivé údaje určené pouze pro dva páry očí. Provozovatelé mailingových služeb jsou tak zavázáni tato data dále veřejně nesdílet. Stejně tak samotní uživatelé nechtějí svá data zveřejňovat (už jen z toho důvodu, že by to bylo velice necitlivé ke všem, se kterými dotyčný vedl ko-

[18] `www-2.cs.cmu.edu/~enron`
[19] `www.aueb.gr/users/ion/data/enron-spam`

munikaci). V případě phishingových emailů je pak problém, že se jedná o relativně specifickou výseč z početné skupiny nevyžádané pošty. Je tak nutné tyto vzorky ručně vybírat.

5.1.3 Alternativní datasety

Emaily Hillary Clintonové
Tento dataset byl vytvořen z veřejně dostupných emailů americké političky Hillary Clintonové z let 2008-2014. Emaily byly uveřejněny ministerstvem zahraničních věcí v následnosti na aféru kolem používání osobního mailového serveru pro vládní záležitosti. Emaily byly uveřejněny v PDF formátu. Ty je možné prohlížet přímo z webu ministerstva [20].
Komunita kolem portálu Kaggle [21] se pokusila tato data zpracovat pro další analýzu. Došlo tak k vyčistění a normalizaci dokumentů a převedení do CSV formátu a do SQL databázového souboru.

Nicméně ani po těchto snahách se dataset neobešel bez několika neduhů, které jsou zapříčiněny faktem, že data jsou extrahována z PDF, a ztratilo se tedy velké množství metadat o emailu. Vyparsovaná data odeslání či přijetí tak nemají jednotný formát či zcela chybí. Další čistění/filtrace před použitím je tudíž nezbytná. Výsledkem je pak 5636 emailů. Důležitým problémem, převážně pro tuto práci, jest, že zcela chybí informace o HTML formátování. Je k dispozici pouze textová reprezentace emailu. Přicházíme tak o informaci o použitých scriptech, cílech odkazů a všech dalších elementech, které nemají nutně vizuální reprezentaci.

Osobní emaily
Další z možností je využít vlastní zdroje – osobní emailové komunikace. Toto je dobrý zdroj v tom ohledu, že téměř každý z nás již emailovou schránku (zpravidla několik let) vlastní. Pro tento projekt jsem vzal jednu ze svých schránek. Ta čítala 599 emailů z období konce roku 2013 až do začátku roku 2016. Tato schránka byla vedena ve službě Gmail, která umožňuje jednoduchý export emailů v MBOX formátu přímo z webu. Po zažádání se do několika minut emaily zabalí a zpřístupní se odkaz na stažení. Ne všechny mailové služby přímý export umožňují – v takovém případě je třeba sáhnout po emailových klientech, kteří dokáží emaily zabalit přímo na počítači uživatele, ať již explicitně či za pomoci dodatečných pluginů.

Ač se tato možnost může jevit jako ideální zdroj *ham* emailů, nese si s sebou několik nevýhod. Je třeba si uvědomit, že se zpravidla jedná o vzorek jedné osoby z jedné sociální skupiny a jedné geografické lokace. Pro příklad můj vzorek emailů je z českého prostředí, z technických kruhů osoby do 30 let. Tento vzorek se bude diametrálně lišit např. od vzorku pracovních emailů 50letého novináře z Kanady či 10letého teenagera z Velké Británie.

[20] `https://foia.state.gov/Search/Results.aspx?collection=Clinton_Email`
[21] `wwww.kaggle.com`

Dataset	Link count		IP link count		Max subdomains	
	median	mean	median	mean	median	mean
Hillary	0	0.0017	0	0	0	0.0002
Nazario	1	3.0364	0	0.4483	1	1.0013
Personal	1	4.8495	0	0	0	0.5050
SpamAssassin	0	2.4439	0	0.0043	0	0.0530

Tabulka 5.1: Statistické porovnání datasetů

Dataset	Word count		Word length medians		Punctuation		Digits count	
	median	mean	median	mean	median	mean	median	mean
Hillary	12	74.2	4	3.9	2	14.5	0	7.5
Nazario	185	240.8	4	17.1	43	174.3	14	163.2
Personal	250	793.7	5	6.4	139.5	1424.1	64	835.1
SpamAssassin	141.5	277.8	4	4.4	91	163.9	20	42.1

Tabulka 5.2: Statistické porovnání datasetů

Porovnání
V tabulkách 5.1 a 5.2 pak nalezneme statistické srovnání datasetů, ze kterého můžeme vypozorovat větší či menší nuance. U datasetu Hillary Clintonové je pak zřejmý dopad absence HTML tagů a špatné konverze linků zmíněných výše. Z viditelné rozdílnosti datasetů pak vyplývá, že pro vytvoření modelu je vhodné datasety kombinovat, čímž zajistíme dostatečnou variabilitu a různorodost jak geografickou tak časovou.

5.2 Algoritmus

5.2.1 Extrakce příznaků

Tato sekce je věnována volbě a následné strojové extrakci příznaků z dat. Příznaky určují, jaké vlastnosti dat bereme při klasifikaci v úvahu. V dostupné literatuře nalezneme nespočet výčtů příznaků použitých pro klasifikaci phishingových emailů. Ač v každém projektu je tento výčet trochu jiný, je viditelný jejich průnik. Mezi tyto „bázové“ příznaky můžeme zařadit např. počet odkazů v textu, celkový počet slov, přítomnost klíčových slov jako *click* a *here* atd.

Mezi jednotlivými autory pak nalezneme další speciální příznaky – Brandon Azad [14] používá např. průměrný počet znaků na slovo, počet unikátních slov po stemmingu nebo počet explicitních výskytů čísel portů v URL.

Dalším zajímavým příznakem je využití tzv. TF-IDF (term frequency–inverse document frequency). Jedná se o metodiku popisující relevanci/důležitost slova

pro dokument. Hodnota TF-IDF je složena ze dvou složek – první je počet výskytů slova v textu (zpravidla vydělený celkovým počtem slov, aby se předešlo zvýhodňování dlouhých textů). Druhou složkou je důležitost slova. Čím častěji se slovo vyskytuje v sadě dokumentů, tím nižší má důležitost. Jinými slovy IDF značí, jak silnou informační hodnotu slovo přináší – zdali se jedná o slovo časté či vzácné (měřeno přes celou sadu dokumentů). Tím se vyřeší problém, že některá slova se obecně vyskytují v textu častěji (např. anglické *the*). Využití nalezneme v pracích Fette et al. [10] či Zhang et al. [15].

V pracích Fette et al.[10] a Bergholz et al. [16] pak najdeme zajímavý nápad využít jako příznak výstup ze spam filtru. Konkrétně využívají open-source projekt SpamAssassin [22] s výchozím nastavením. Využívají tak toho, že phishing má některé podobné znaky jako spam.

Pro klasifikaci byly použity následující příznaky:

- **Počet linků v textu**
  Numerický příznak udávající počet hyperlinků v textu.

- **Počet linků v IP podobě**
  Další z phishingových metod, jak zastřít informaci o tom, kam odkaz vede, je místo doménového jména použít přímo IP adresu stroje, na kterém je spuštěn phishingový web (např. `http://52.215.81.234/paypal-login`). Uživatel tak do poslední chvíle neví, kam odkaz doopravdy vede. Dalším důvodem použití odkazů v IP formě může být, že phisheři pro své útoky využívají sítě zombie počítačů [23], na kterých automaticky spouští své weby. Zmiňme také, že registrace domén je úkon vyžadující čas a finance. Na druhou stranu, většina legitimních webů zároveň své doménové jméno má. Nalezení IP linků není triviální. Zvolil jsem použití regulárního výrazu z webu Regular-Expressions.info [24], který se ukázal jako nejlépe fungující pro IPv4.

  ```
  (?:(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.){3}
  (?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)
  ```

- **Počet slov**
  Celkový počet slov. Přesněji počet slov v textu mimo HTML tagy.

- **Délka slova**
  Medián délky slov v textu.

[22] `http://spamassassin.apache.org`

[23] Zombie je označení pro počítač napadený virem a je zneužíván útočníkem pro provedení dalších útoků bez vědomí vlastníka tohoto počítače

[24] `http://www.regular-expressions.info/examples.html`

- **Maximální počet teček v URL (počet subdomén)**

  V sekci 2.2 je popsáno zneužití subdomén pro zmatení. Větší množství subdomén znamená více teček v URL. Například `www.yourbank.example.someweb.com` má 2 subdomény. Fette et al. [10] také poukazují na možnost vložení přesměrování přímo do URL jako `www.google.com/url?q=http://www.badsite.com/update.cgi`. Takto budí dojem, že je hostována na `google.com`, ale ve skutečnosti dojde k přesměrování na `badsite.com`. Tento příznak značí maximální počet teček nalezených ve všech URL.

- **Klamavé linky**
  Za klamavé odkazy považuji takové, u kterých neodpovídá hodnota atributu *href* a použitý text odkazu. Není tím ovšem myšleno, že místo např. `www.fake-paypal.com` se zobrazí např. *click here*. Tento příznak se snaží odhalit případy, kdy se jako text odkazu použije jiná adresa.

  ```
  <a href="www.fake-paypal.com">www.paypal.com</a>
  ```

  Toto je velice účinná technika, neb málokdo předpokládá, že když už je uvedena adresa, tak že by byla takto zfalšována.

- **Počet obrázků**
  Pro vyvolání silnějšího pocitu legitimnosti lze využít vložení obrázků s logy a banery webu, který se snaží útočník napodobit (ať už přímým vložením nebo jen odkazem na obrázek).

- **Výskyt JavaScriptu**
  JavaScriptem lze dosáhnout sofistikovanějších ukrývajících technik. JavaScript detekuji dvojím způsobem - hledám výskyt tagu <script> a hledám string „javascript“ uvnitř tagu <a>. Toto je binární příznak.

- **Výskyt formuláře**
  HTML formuláře jsou jednou z nejjednodušších forem sběru informací. Útočník může vložit formulář přímo do těla emailu a nepotřebuje tak již ani dedikovaný podvodný web. Vložením *action* atributu do formuláře pak určí, kam se mají data odeslat.

  ```
  <FORM action="http://www.paypal-site.com/profile.php"
  method="post">
  ```

  Toto je binární příznak.

- **Množství interpunkce**
  V tomto příznaku se snažím zachytit veškerou interpunkci objevující se v textu. K tomu využívám výčet z Python modulu *string.punctuation*. Do výsledného součtu pro tento příznak se pak započítává výskyt těchto znaků:

```
!"#$%&'()*+,-./:;<=>?@[\]^_‘{|}~
```

- **Číslovky**
  Zajímavým příznakem je pak množství číslic obsažených v textu. Jinými slovy počet výskytů některého ze znaků:

  ```
  0123456789
  ```

- **Výskyt specifických slov**
  Ručním procházením dat lze vypozorovat některé opakující se vzorce, co se týče samotného textu. Jedná se nám zpravidla o sběr informací o účtech formou kliknutí na odkaz a přesměrování na podvodnou stránku. Můžeme tedy očekávat výskyt slov jako *account*, *sign*, *click* či *member*. Útočníci se také nezřídka snaží v uživateli vyvolat pocit neodkladnosti a urgence (např. „Váš účet byl dočasně z bezpečnostních důvodů zablokován. Přihlaste se pro ověření totožnosti“). Detekuji tedy i výskyty slov *verify* a *suspension*. Ke každému slovu (termu) přísluší jeden příznak značící počet výskytů tohoto slova. Výslednou množinu slov jsem tedy sestavil z vlastních poznatků a analýzou dat a k těm jsem připojil seznam, který používal Chandrasekaran et al. v práci *Phishing email detection based on structural properties* [17]. Výsledný výčet zvolených slov je v tabulce 5.3.

FREE	INCONVENIENCE
SIGN	INFORMATION
MEMBER	LIMIT
VERIFY	LOG
ACCOUNT	HOUR
CLICK	PASSWORD
SUSPENSION	RECENTLY
ACCESS	RISK
BANK	SOCIAL
CREDIT	SECURITY
IDENTITY	SERVICE

Tabulka 5.3: Klíčová slova jako příznaky

Postupem času může tento výčet zastarávat vinou nových technik phisherů stejně tak, jako přirozeným vývojem používaného jazyka mezi lidmi. Zároveň pozorování má ani ostatních výzkumníků nemusí být zcela přesná. V ideálním případě bychom měli tedy tento postup automatizovat. Vhodným rozšířením této práce by tak v budoucnu bylo přidání mezikroku text-miningu. Z množiny emailových dokumentů (textů) bychom od-

stranili stopová slova a aplikovali *stemming*[25]. Na výslednou matici bychom následně aplikovali některou z feature-selection technik (např. Chi-squared [26]).

- **WHOIS**
  Jeden příznak udávající počet dní od založení vybrané domény. Druhý pak počet dní od poslední změny doménových údajů. K vypočtení se nejdříve provede dotaz na WHOIS [27] na danou doménu. Z odpovědi vyparsuji datum vytvoření a změny. Výsledné příznaky pak vzniknou odečtením těchto dat od data přijetí emailu. Problematika dotazování na WHOIS je pak rozebrána v podsekci 5.2.1.1.

5.2.1.1 Problematika dotazování přes WHOIS

Při implementaci a následném testování jsem narazil na několik problémů spjatých se získáváním informací o doménách. WHOIS je query-response protokol, ze kterého je možné získat informace o vlastníkovi, kontaktních údajích, datu založení či expirace atd. Tato data jsou využívána síťovými administrátory k diagnóze problémů. Také najdou využití v boji proti spamu a phishingu jako nástroj pro vystopování původu. První problém, který je zřejmý hned z prvního použití, je absence jakékoliv standardizace odpovědi ze serveru. Ač existují specifikace pro WHOIS (např. poslední RFC 3912), forma odpovědi není nijak vynucována. Implementátor WHOIS služby tak sám volí, která pole budou v odpovědi obsažena a v jakém formátu. Tím se velice stěžuje strojové zpracování. Tento problém je typicky adresován podpůrnými knihovnami pro jednotlivé jazyky. Ty zpravidla obsahují řadu mutací jmen vracených polí a snaží se namapovat odpověď na nějaký zpracovatelný objekt. Úspěšnost tak již z principu není 100% a v krajním případě vede k zanesení chyb – často jsem se setkával s případem, kdy služba vrátila odpověď, ve které stálo, že dotazovanou doménu se nepodařilo najít a připojila i meta informace o serveru. Tyto informace pak knihovna mylně naparsovala jako validní data odpovědi. Druhým závažným problémem jest, že provozovatelé WHOIS serverů si často nepřejí, aby byla tato data zpracovávána automaticky a ve větším objemu viz podmínky užití WHOIS serveru organizace VeriSign:

> "You are not authorised to access or query our Whois database through the use of electronic processes that are high-volume and automated except as reasonably necessary to register domain names or modify existing registrations" [28]

[25] Stemming - proces nalezení kmene slova

[26] `https://en.wikipedia.org/wiki/Pearson's_chi-squared_test`

[27] WHOIS – protokol pro dotazování na informace o internetových zdrojích (doménách, IP adresách, autonomních systémech atd.)

[28] zdroj: `https://www.verisign.com/en_GB/whois-lookup`

České sdružení CZ.NIC, která spravuje .cz domény, je ve svých podmínkách užití ještě explicitnější:

> "UPOZORNĚNÍ: Požadavky na poskytnutí dat nebo informací jsou zaznamenávány. V případě, že je požadavek nebo série požadavků vyhodnocen jako útok směřující k poškozování činnosti síťových služeb nebo jako snaha nepovoleně shromažďovat data v rozporu se stanoveným účelem, může tato skutečnost vést k zablokování přístupu k informačním službám sdružení CZ.NIC, případně k dalším nezbytným krokům." [29]

Nikde jsem nenarazil na přesnější specifikaci „velkého objemu" a zdali to, k čemu v tomto projektu WHOIS používám (boj proti phishingu), již spadá mimo záměry užití těchto služeb. Věřím, že např. s CZ.NIC by se dala domluvit spolupráce o lepším zpřístupnění WHOIS dat. Takto by bylo však nutné se domluvit se všemi provozovateli WHOIS serverů, což je prakticky nemožné. V praxi to znamená, že někteří provozovatelé umožní i několik tisíc dotazů za minutu bez větších limitací. Jiní pak, jako třeba CZ.NIC, umožní dotazování maximálně cca 5 krát za minutu a dlouhodobě může dojít k úplnému zablokování služby. Častější frekvence dotazování vede k dočasnému omezení ze strany provozovatele a místo odpovědí se začne vracet varovná zpráva.

> "Your connection limit exceeded. Please slow down and try again later."

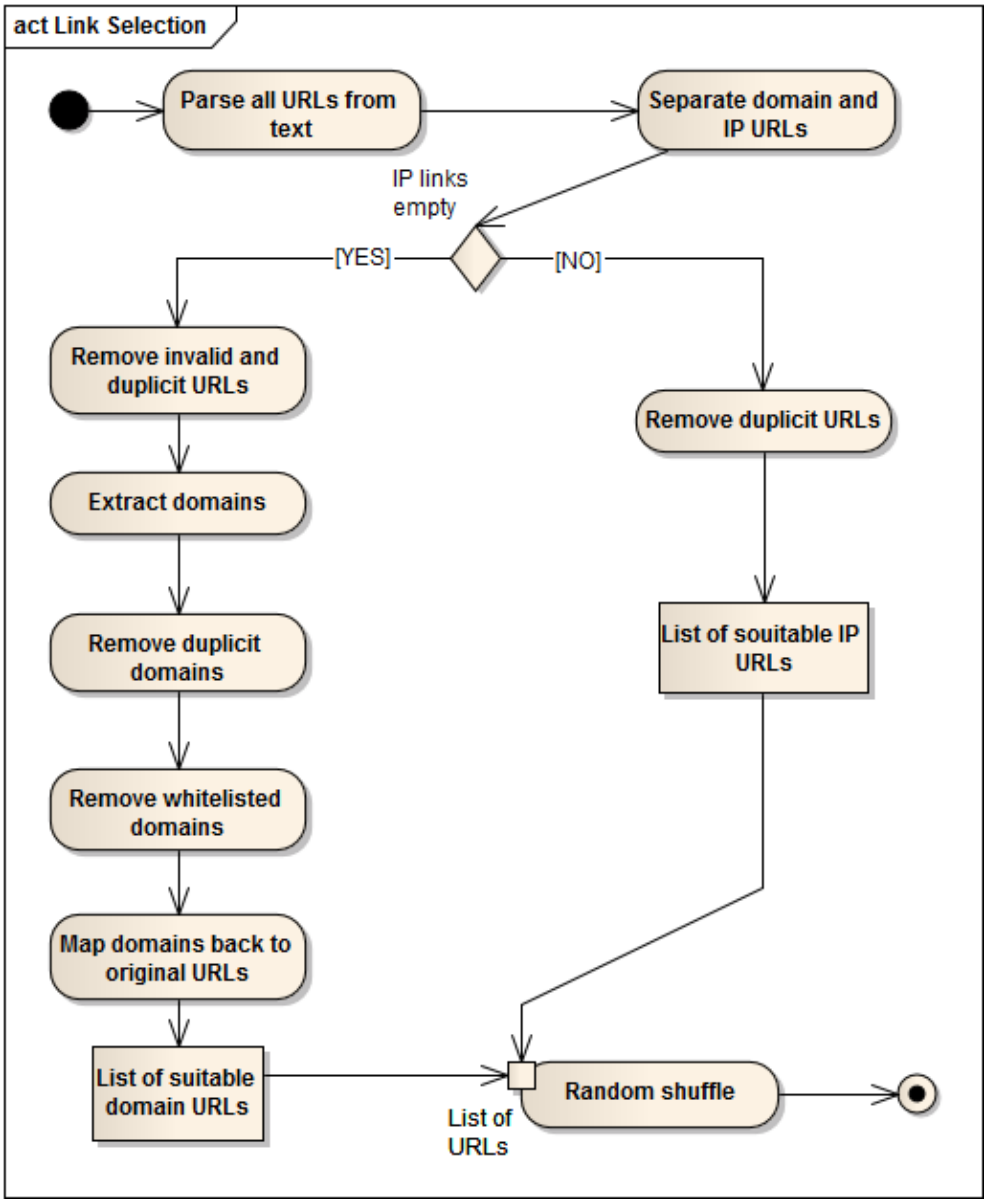
Z toho plyne, že každý běh scriptu na dotazování může skončit s jinými výsledky(v návaznosti na rychlost vracení odpovědí od serverů se zvětšuje časová proluka mezi dotazy a tak klesá/stoupá nebezpečí vyčerpání limitu).

5.2.1.2 Problém nalezení phishingového odkazu

Součástí klasifikační logiky tohoto projektu je otestování, zda se email snaží uživatele odkázat na stránku, která vykazuje charakteristiky phishingové stránky. Jsme tak postaveni před výzvu takový odkaz v textu emailu najít. Text může odkazů obsahovat vícero a ne všechny musí nutně vést na phishingovou stránku. V praxi se tento jev vskutku objevuje často a slouží k další obfuskaci a zmatení automatických filtrů. Řešením by se mohlo zdát otestování všech odkazů v textu a následně vzít v potaz pouze ten, který vykazuje nejvyšší podobnost s phishingovými weby. Tento postup je však prakticky neefektivní a náročný na zdroje – provedení WHOIS dotazu i stažení webové stránky jsou časově náročné úkony závislé na vnějších faktorech. Stažená stránka také může být relativně objemná. Pokud bychom chtěli vystavět systém, který bude zpracovávat velký objem emailů, zde by vzniklo velice úzké hrdlo.

[29] zdroj:`https://www.nic.cz/page/306/ucel-pouziti-poskytovanych-dat-a-informaci`

Navrhl jsem tedy algoritmus, který se snaží z nalezených odkazů vybrat malou podmnožinu, která je vhodná k dalšímu testování. Průběh algoritmu znázorňuje obrázek 5.1



Obrázek 5.1: Algoritmus výběru vhodného odkazu

Algoritmus vrací množinu odkazů. Tím umožníme celému systému tento proces škálovat – pokud máme dostatek volných zdrojů, můžeme následně otestovat všechny vrácené odkazy. V opačném případě z množiny vybereme např. pouze jeden prvek. Množina je náhodně zpřeházená. Tím se usnadní

následný náhodný výběr prvků – stačí jednoduše vzít ten první.

Tento algoritmus je pak vhodným místem pro další budoucí rozšiřování a zlepšování. Zařadit bychom mohli informaci o pořadí odkazů (např. odkazy, které se objevují na konci emailu často vedou na stránku s kontakty či na stránky podpory. A to i v případě phishingových emailu, které tuto techniku používají k vyvolání pocitu autentičnosti). Dále bychom moli brát v potaz umístění odkazu vůči ostatnímu obsahu – zdali je odkaz obklopen textem či je od něj vizuálně oddělen. V neposlední řadě by se dala využít informace o zobrazovaném textu odkazu (a případně jeho formátování). Např. odkaz s textem *Click here!*, napsaný červeným písmem dvojnásobnou velikostí než okolní text, by jistě byl vhodným kandidátem. Nabízí se i využití text-miningových technik k automatické extrakci takovýchto pravidel.

5.2.2 Klasifikátory

5.2.2.1 Naive Bayes

Naive Bayes klasifikátor je založen na Bayesově teorému podmíněné pravděpodobnosti s předpokladem nezávislosti mezi jednotlivými příznaky. Jinými slovy i pokud na sobě příznaky závisí, z pohledu Naive Bayes klasifikátoru všechny příznaky přispívají nezávisle k pravděpodobnosti náležitosti do jedné z výsledných tříd. Jedná se o klasifikátor výpočetně nenáročný, škálovatelný a s malým nárokem na objem trénovacích dat. Již od šedesátých let minulého století je úspěšně využíván pro klasifikaci, především pro klasifikaci textu, kde vykazuje dobré výsledky.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{5.1}$$

Bayes teorém

5.2.2.2 Rozhodovací strom

Rozhodovací stromy jsou intuitivní technika vyčnívající především pochopitelností a snadnou interpretací výsledků. Jedná se o grafovou strukturu, kde každý uzel koresponduje s jedním ze vstupních atributů/příznaků. Pro atributy nabývající nominálních hodnot platí, že uzel má tolik odchozích hran, kolik je možných hodnot atributu. U numerických atributů se používá rozdělení intervalu hodnot na disjunktní množiny. Každá odchozí hrana pak představuje jeden takovýto interval. Uzly takto vytvořeného stromu pak nabývají hodnot predikovaného atributu – v případě klasifikace phishingu je to binární označení *phishing* vs. *ham*. Cesta z vrcholu (kořene) grafu k listu pak tedy představuje samotné rozhodovací pravidlo.

Budování rozhodovacího stromu je parametrizovatelný proces. Z těch nejvýznamnějších stojí za zmínku kritérium pro volbu atributu ke štěpení. Kritérií pro štěpení je velké množství [18]. Dále popíši dvě základní:

- **Information gain**
  Tento postup je založen na snižování entropie datasetu. Entropie pro množinu prvků $i \in \{1, 2, ..., m\}$ je dána vzorcem:

$$I_E(f) = -\sum_{i=1}^{m} f_i \log_2 f_i \tag{5.2}$$

  1. Vypočte se entropie datasetu.
  2. Dataset je rozdělen podle všech atributů.
  3. Pro každý subset se vypočte entropie. Jejich součet dá celkovou entropii.
  4. Odečtením od původní entropie datasetu, získáme *Information gain* neboli pokles míry entropie.
  5. Vybereme atribut, který vykazuje největší *Information gain*.

$$IG(T, a) = H(T) - H(T|a) \tag{5.3}$$

- **Gini index**
  Jedná se pouze o jiný pohled na homogenitu (čistotu) datasetu.

$$I_G(f) = 1 - \sum_{i=1}^{m} f_i^2 \tag{5.4}$$

5.2.2.3 Random Forest

Random Forest (Náhodný les) je klasifikátor založený na rozhodnutí skupiny náhodných stromů – to jest stromů, které při každém štěpení pracují pouze s náhodnou podmnožinou atributů. Po vytvoření této skupiny stromů (lesa) probíhá klasifikace tak, že se vzorek oklasifikuje na všech těchto stromech a jako výsledek se vezme výsledek/třída s nejvyšším zastoupením. Touto metodou lze dosáhnout snížení náchylnosti k overfittingu (zaujetí). Další výhodou náhodných lesů je, že dokáží efektivně pracovat i s velkou množinou atributů a malou množinou trénovacích dat. Na druhou stranu jsou tyto výhody vykoupeny vyšší výpočetní náročností – všechny stromy musí provést svůj výpočet. Random Forest je parametrizován podobně jako rozhodovací strom. Přidává pak řadu parametrů, z nichž je patrně nejdůležitější právě počet generovaných stromů.

5.2.2.4 SVM

Support Vector Machine vytváří z trénovacích dat vnitřní model, kde každá trénovací instance je reprezentována jako bod v prostoru tak, že mezi instancemi patřícími do jedné třídy je co možná největší mezera od instancí patřících do třídy druhé. Uprostřed této mezery vzniká klasifikační/dělící linie. Čím širší mezera, tím lépe model generalizuje na neznámá data. Nové vzorky jsou následně klasifikovány dle toho, na kterou stranu od této linie spadají.

Z významných parametrů stojí za zmínku především regularizační parametr C. Klasické SVM se snaží najít takovou linii, která rozdělí *všechny* pozitivní a negativní instance. Neumí tedy pracovat se šumem. Užitím regularizačního parametru C jsme schopni zvýšit toleranci k šumu a tím dosáhnout lepší generalizace. Přesněji, C představuje míru postihu (penalty) pro chybnou klasifikaci instance. Vyšší hodnota C vede k nižší chybovosti na trénovacích datech, ale horší generalizaci a vice versa.

5.3 Klasifikační metriky

Tato sekce se zabývá metrikami pro evaluaci výkonnosti a vzájemné porovnávání klasifikátorů. Proč jsou metriky potřeba? Při pohledu na výsledky klasifikace (po běhu na testovacích datech) je třeba si uvědomit, že pouhý údaj o počtu správně oklasifikovaných vzorků není dostatečně vypovídající údaj. To vyplývá z faktu, že pro většinu klasifikačních úloh platí, že cena za špatnou klasifikaci není stejná pro všechny třídy. Dalším problémem je pak nevyvážené zastoupení daných tříd v datech (např. jeden typ převažuje nad ostatními).

Pro ilustraci vezměme následující příklad klasifikace opětovného výskytu rakoviny:
Vezměme klasický dataset rakoviny prsu [30] obsahující záznamy 286 žen, které trpěly rakovinou. U 85 žen se rakovina do 5 let objevila znovu, u 201 nikoliv. Představme si model, který pro všechny případy předpovídá, že se rakovina do 5 let znovu neobjeví. Tento model správně předpoví pro 70,28 % (201/286) případů. To se může zdát jako dobrý výsledek. Avšak toto je velice špatný model, neboť 85ti rizikovým ženám nepředpoví žádný problém (a následně např. nedoporučí další vyšetření). Tento případ vizualizuje tabulka 5.4.

Na druhou stranu si představme model, který vždy předpovídá znovuobjevení rakoviny. Tento model bude fungovat pouze v 29,72 % případech. 201 ženám tak nesprávně předpoví, že jim hrozí zdravotní obtíže viz tabulka 5.5.

V tomto případě budeme nejspíše ochotnější připustit druhý model, ač dosahuje menší úspěšnosti. Potřebujeme tedy metriky, které jsou schopné cenu jednotlivých chyb reflektovat. Nejdříve nadefinujme označení vzorků:

[30] `http://archive.ics.uci.edu/ml/datasets/Breast+Cancer`

n=286	Předpověď: **Neobjeví**	Předpověď: **Objeví**	
Skutečnost: **Neobjevila**	TN = 201	FP = 0	201
Skutečnost: **Objevila**	FN = 85	TP = 0	85
	286	0	

Tabulka 5.4: Tabulka záměn pro negativně-zaujatý klasifikátor

n=286	Předpověď: **Neobjeví**	Předpověď: **Objeví**	
Skutečnost: **Neobjevila**	TN = 0	FP = 201	201
Skutečnost: **Objevila**	FN = 0	TP = 85	85
	0	286	

Tabulka 5.5: Tabulka záměn pro pozitivně-zaujatý klasifikátor

(a) vzorek je negativní - ve výše uvedeném příkladu je negativní takový vzorek, u kterého se do 5 let rakovina *neobjeví*. V případě problematiky phisihingu bude nadále v textu označovat negativní vzorky takové, které phishingem *nejsou* (i.e. jedná se o legitimní email).

(b) vzorek je pozitivní - pozitivní vzorek (email) je takový, který považujeme za phishing.

Dále nadefinujme 4 možné výstupy klasifikace pro každý vzorek:

- True positive **(TP)** - vzorek je phishing a byl korektně označen jako phishing.
- False positive **(FP)** - vzorek byl označen jako phishing, ale jedná se o legitimní email.
- True negative **(TN)** - vzorek byl korektně označen jako legitimní.
- False negative **(FN)** - vzorek byl označen jako legitimní, ale jedná se o phishing.

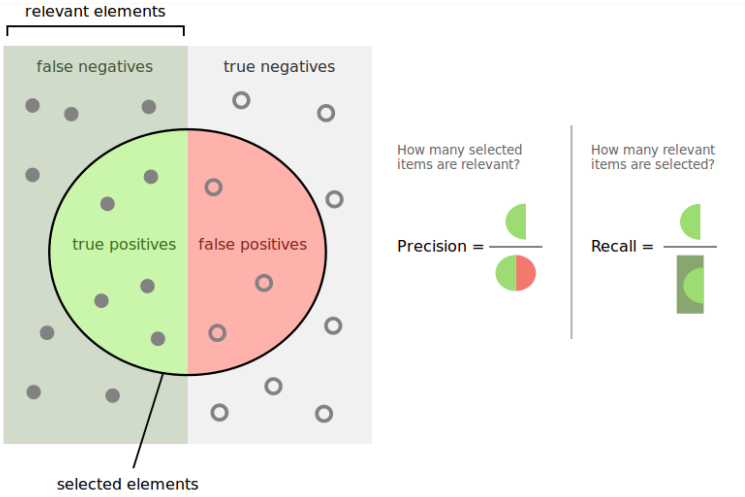
5.3.1 Přesnost (accuracy)

Základní metrikou je přesnost. Je to jednoduchý poměr korektně klasifikovaných vzorků ku celkovému počtu klasifikovaných. Přesnost je jednoduchá na pochopení a interpretaci. Na druhou stranu se jedná o informativně *slabší* metriku, neboť nebere v úvahu rozdílnou cenu za misklasifikaci jednotlivých tříd.

$$Accuracy = \frac{TP + TN}{P + N} \tag{5.5}$$

5.3.2 Precision & Recall

Precision a recall jsou jedny z nejzákladnějších metrik. Jejich vzájemný vztah je nejlépe vidět na obrázku 5.2.



Obrázek 5.2: Vztah Precision a Recall[31]

Precision (predicition positive value, přesnost)
Udává, kolik z označených je označeno správně. Jinými slovy poměr *korektně* označených jako pozitivní ku všem označeným jako pozitivní.

$$Precision = \frac{TP}{TP + FP} \tag{5.6}$$

Recall (true positive rate, citlivost)
Poměr korektně označených jako pozitivní ku všem pozitivním. Neboli kolik ze všech pozitivních se podařilo úspěšně označit.

[31] zdroj: `https://commons.wikimedia.org/wiki/File:Precisionrecall.svg`

$$Recall = \frac{TP}{TP + FN} \tag{5.7}$$

5.3.3 F-measure

F-measure (nebo také F1 score) kombinuje recall a precision. Dá se interpretovat jako harmonický průměr přesnosti a citlivosti, kde F1 dosahuje nejlepší hodnoty v 1 a nejhorší v 0.

$$\begin{aligned} F1 &= 2 * \frac{precision * recall}{precision + recall} \\ F1 &= \frac{2TP}{2TP + FP + FN} \end{aligned} \tag{5.8}$$

5.3.4 AUC

AUC představuje plochu pod ROC křivkou. Výhoda této metriky je, že si dokáže poradit i s nerovnoměrným rozložením instancí do tříd (např. když 97 % je pozitivních). Hodnota AUC se pohybuje v intervalu od nuly do jedné. Ideální klasifikátor dosahuje hodnoty 1. Klasifikátor, který označí vše špatně, bude mít hodnotu 0. Zároveň platí, že klasifikátor, který každý vzorek označí zcela náhodně, bude i na takovýchto datech dosahovat hodnoty +- 0.5. Tato metrika nám tedy dokáže říci více než přesnost, která by na takovýchto datech selhala (i náhodný klasifikátor by na takto nevyvážených datech dosahoval vysoké přesnosti).

5.4 Architektura

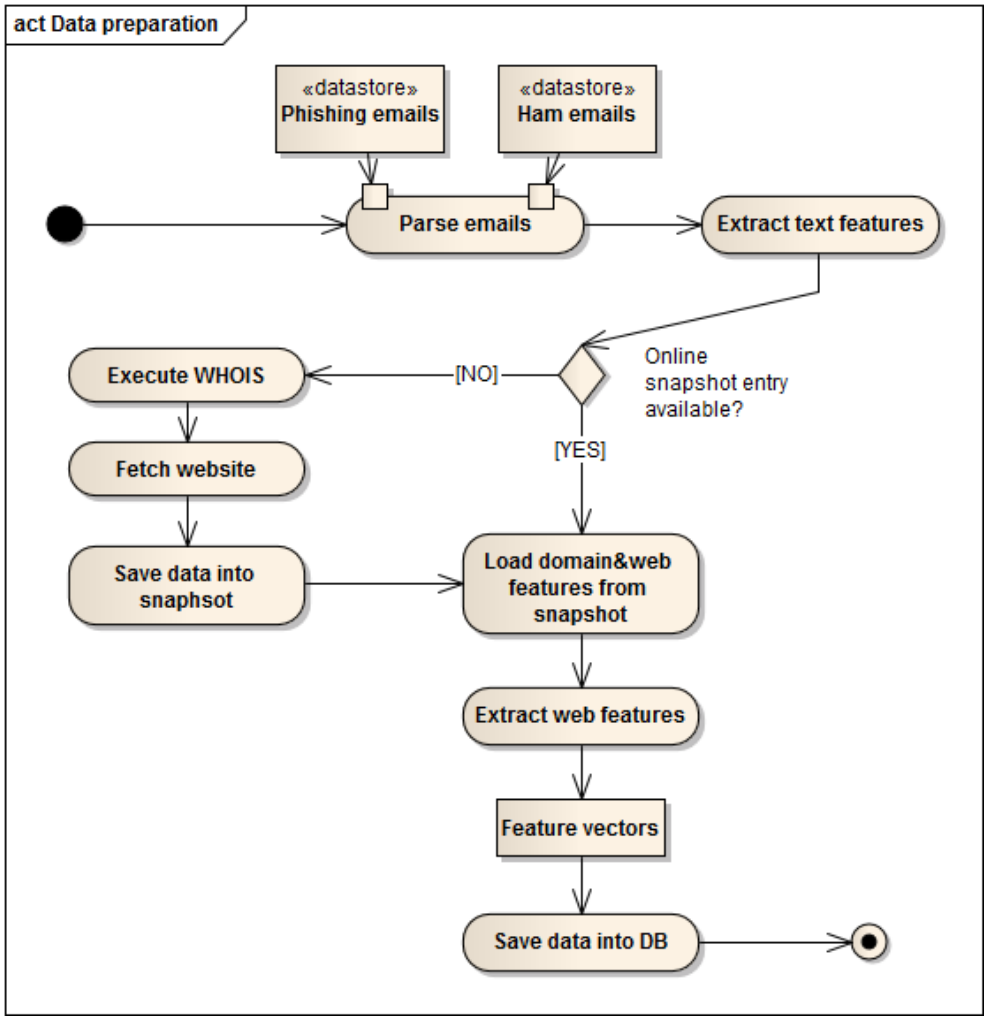
V této sekci se zaměřím na celkovou architekturu sytému a jeho nároků. Klasifikační ekosystém sestává ze dvou hlavních částí – trénování a evaluace modelu a užití modelu pro klasifikaci příchozích vzorků. V této sekci tyto dva procesy detailněji rozeberu a v návaznosti na ně navrhnu vhodné architektonické řešení.

5.4.1 Trénovací fáze

Účelem této fáze je vytvoření klasifikačního modelu (v případě více modelů vybrat ten nejvhodnější) z dostupných trénovacích dat. Jinými slovy, vstupem tohoto procesu jsou trénovací data ve formě emailů a výstupem je klasifikační model. Tento proces je dále rozdělen do dvou podčástí – příprava dat a samotné trénování.

5.4.1.1 Příprava dat

Na schématu 5.3 je vidět detailní průběh přípravy dat používaný v tomto projektu.



Obrázek 5.3: Proces přípravy dat pro trénování modelu

Online otisk

Důležitou komponentou je zde „otisk“ (snapshot) informací o online zdrojích. Tento otisk zajistí, že data se nebudou v čase měnit a degradovat. V této práci extrahuji z emailů informace o stáří domén a vlastnostech cílového webu. V okamžiku prvního použití bychom byli schopni tyto extrahovat, ale pokud bychom chtěli o několik týdnů později model přetrénovat (například po přidání nového příznaku či úpravě extrakčních algoritmů), tyto informace by již mohly být pozměněné či úplně nedostupné (web byl zablokován, WHOIS zá-

znam pozměněn). Data už by nebyla validní a zpětně obnovitelná. Neplatí tu tedy intuitivní pravidlo, že čím novější informace, tím lépe. Je nutné pracovat s daty v té podobě, v jaké byla v době poslání emailu. Můžeme předpokládat, že model bude, vzhledem k novým trendům a technikám phishingu, třeba přeučovat často.

Tento otisk proto pro každý vybraný odkaz obsahuje informace o doméně a uložené HTML cílového webu. Při prvním běhu programu se tak tento otisk naplní a při každém dalším běhu se již používají pouze uložená data. Výjimkou jsou nově přidané vzorky či vzorky, pro které se napoprvé tyto informace nepodařilo získat.

Životnost tohoto otisku a jeho údržbu bychom měli spojit s životností testovacích dat. K udržení kroku s novými trendy je třeba odstraňovat staré záznamy.

Výstupní formát přípravy dat
Extrakcí ze všech vstupních emailů vzniká množina vektorů příznaků vhodná pro další použití. Ten můžeme buď rovnou zapsat do databáze nebo ho nejdříve vyexportovat do nějakého snadno parsovatelného formátu (např. CSV, JSON, XML).

5.4.1.2 Trénování modelu

Prototypový průběh trénování modulu nalezneme na schématu 5.4 z programu RapidMiner. Samotný validační modul pak na schématu 5.5. Tento proces se skládá ze 4 hlavních součástí:

1. Načtení dat
2. Natrénování modelu
3. Vyhodnocení efektivity
4. Export modelu

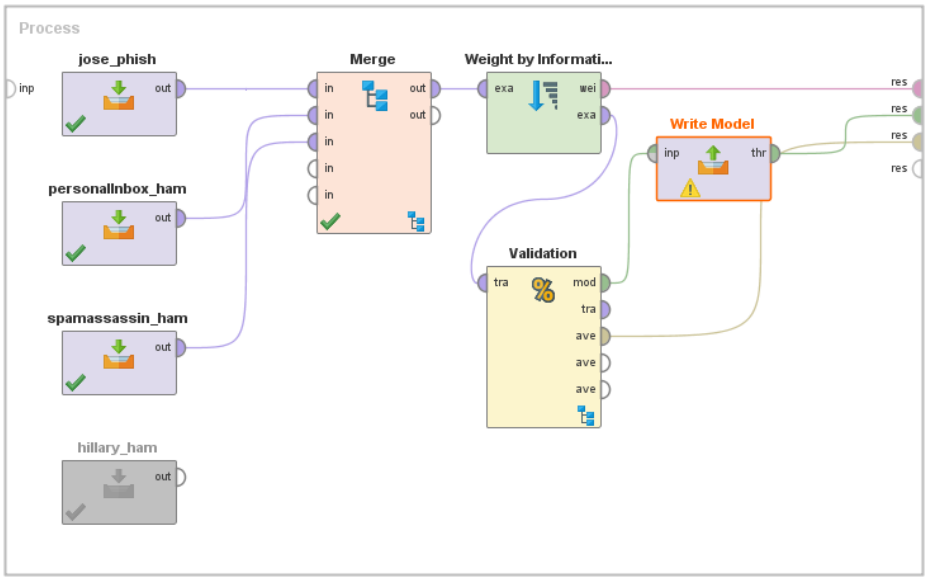
Nespornou výhodou nástroje RapidMiner je, že jsme schopni rychle měnit vstupní data a testovat jednotlivé klasifikátory.

5.4.2 Klasifikační fáze

V této fázi již máme k dispozici natrénovaný model a můžeme klasifikovat příchozí poštu. Jako vstup tak bereme jeden email a jako výstup získáváme označení *phishing* či *ham*. Proces je znázorněn na diagramu 5.6.

Klasifikace je zde rozdělena do tří fází:

1. **Offline klasifikace**
   Nejprve jsou vyextrahovány příznaky, které je možno získat bez závislosti na vnějších zdrojích a Internetu. Email je oklasifikován na základě



Obrázek 5.4: Trénování modelu v RapidMiner

těchto příznaků. Pokud je míra jistoty (confidence) klasifikátoru vyšší, než nastavený práh, klasifikace je u konce.

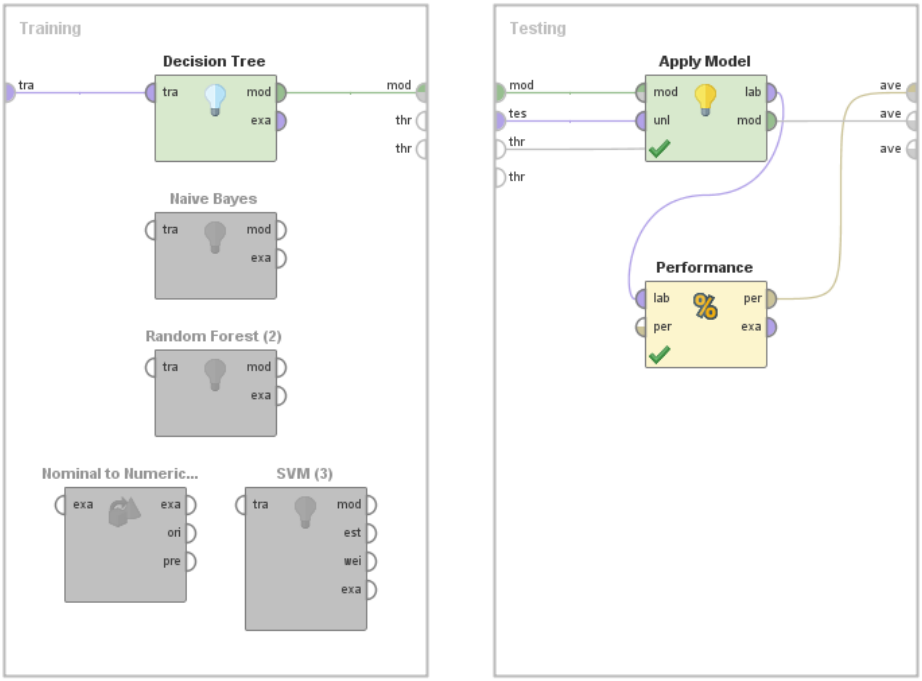
Správné nastavení tohoto prahu je otázkou dlouhodobějšího testování a pozorování v reálném provozu.

2. **Doménová klasifikace**
   V opačném případě je proveden dotaz na informace o doméně. Provede se nová klasifikace s těmito informacemi. Při dostatečné jistotě opět klasifikace končí.

3. **Online klasifikace**
   Je staženo HTML cílového webu. Finální klasifikace je provedena s příznaky získanými z tohoto webu.

Registr aktuálních hrozeb

V diagramu je představen koncept registru „aktuálních hrozeb“. Phishing, stejně tak jako jiné formy nevyžádané masové pošty, přichází ve vlnách. Útočník vytvoří svou kampaň a začne ji distribuovat potencionálním obětem. V krátkém okamžiku tak lidé obdrží ten stejný (či velice podobný) email. Abychom ušetřili výpočetní zdroje, je vhodné při prvním zadržení takovýto email (či jeho hash) přesunout do seznamu aktuálních hrozeb. Ve chvíli, kdy zachytíme další email z této kampaně, již ho nemusíme klasifikovat a rovnou ho označíme jako hrozbu/zablokujeme. Výběr správné a efektivní podobnostní/hashovací funkce je mimo rámec této práce. Nejedná se totiž o triviální problém. Klasické



Obrázek 5.5: Validační proces v RapidMiner

hashovací funkce jako SHA-1 či MD5 dělají přesný opak toho, co vyžadujeme – tyto funkce při malé změně dokumentu generují velkou změnu ve vygenerovaném hashi. Mezi vhodnými funkcemi zmiňme například hashovací algoritmus TLSH[19] či metriku *Levenshteinova vzdálenost*.

Krom samotných emailů bychom si mohli udržovat i informace o IP adrese odesílatele a efektivně tak blokovat i další jeho kampaně. Tento blacklist [32] by pravděpodobně velice rychle nabíral na velikosti a je nutné tak nasadit i proces údržby a promazávání tohoto seznamu.

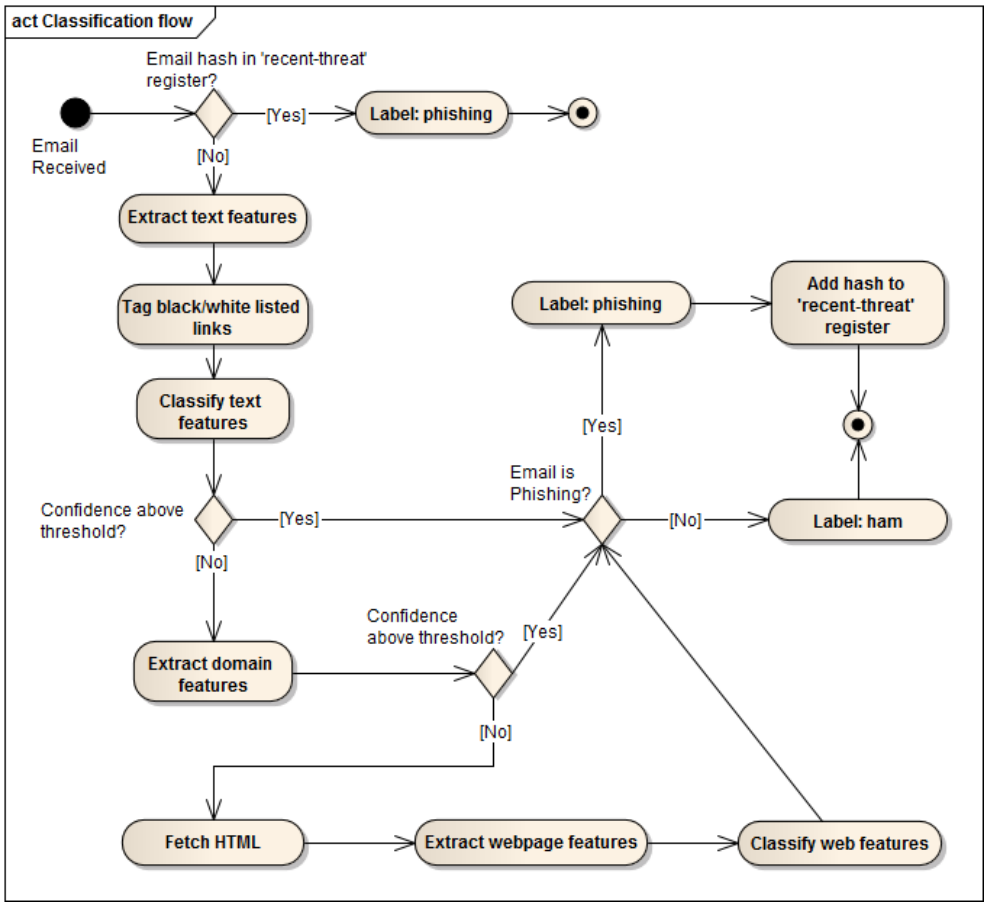
Doménový whitelist

V algoritmu využívám whitelist důvěryhodných domén k odfiltrování odkazů nezajímavých pro další testování. Pro svou prototypovou implementaci jsem použil whitelist sestavený ze dvou zdrojů – nejnavštěvovanější stránky dle serveru Alexa[33] a ze stránek, které jsou častým cílem phishingu dle serveru Phishtank[34]. Výsledný whitelist čítá 31 domén. I takto malý whitelist ovšem dosahuje vysoké efektivity. Jeho užití demonstruje binární příznak *all_domains_whitelisted*, který udává, zdali jsou všechny nalezené odkazy na whitelistu

[32] blacklist=seznam známých podvodných zdrojů, whitelist=seznam známých důvěryhodných zdrojů

[33] `http://www.alexa.com/topsites`

[34] `http://www.phishtank.com/stats.php`



Obrázek 5.6: Proces klasifikace

(případ prázdné množiny je zde brán stejně, jako kdyby všechny odkazy na whitelistu byly). Statistiku tohoto příznaku pro jednotlivé datasety naleznete v tabulce 5.6.

Horší výsledek na datasetu osobních emailů přikládám faktu, že se jedná o silně zaměřený dataset jediné osoby obsahující mnoho technicky zaměřené komunikace. Na ostatních ham datasetech pak vidíme jednoznačně pozitivní výsledek a na phishingových datech pak pozorujeme opak. Toto indikuje silný příznak, jak dokládá i tabulka 6.3 *Information Gain Ratio*, kde se tento příznak umístil na druhém místě.

Předpokládám tedy, že užití komplexnějších whitelistů by vedlo ještě k vyšší přesnosti klasifikátoru.

5.4.3 Návrh systému

Na diagramu 5.7 prezentuji návrh architektury systému a jeho komponent. Tento diagram představuje souhrnný pohled na komponenty identifikované při

Hillary - ham	99.9 %
Nazario - phish	21.8 %
Personal - ham	57.2 %
SpamAssassin - ham	95 %

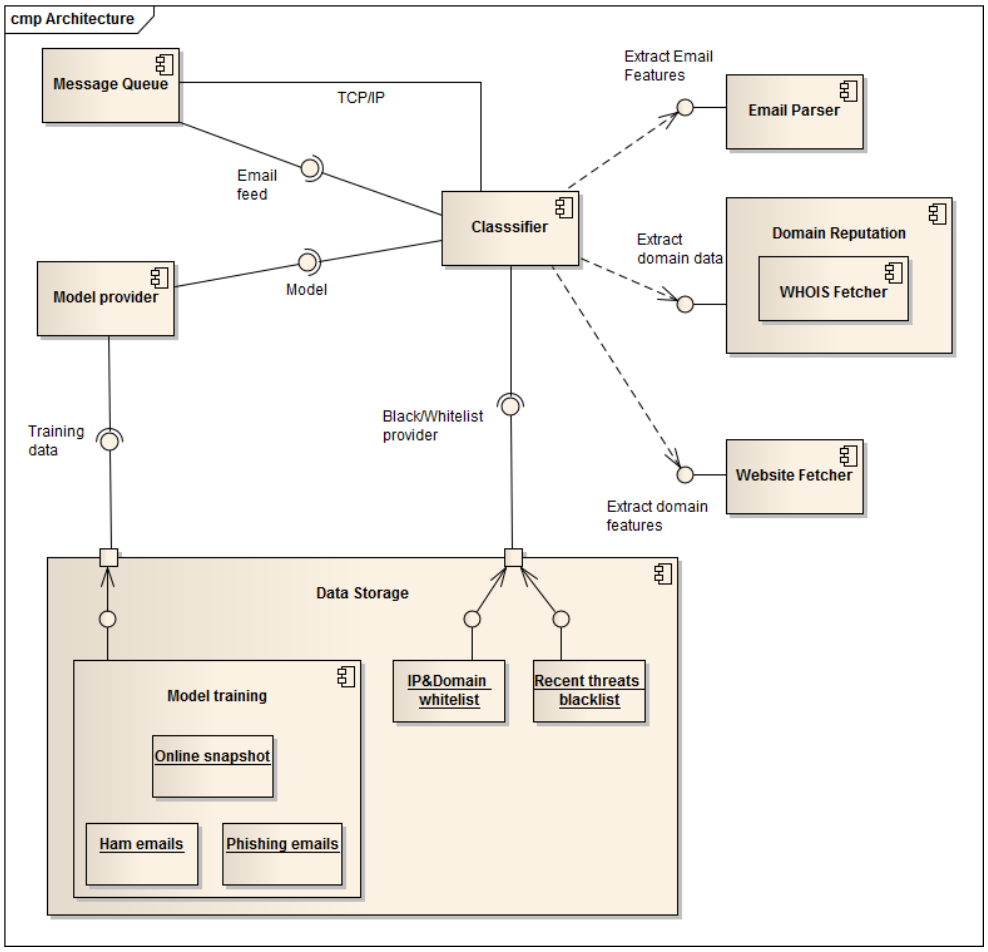
Tabulka 5.6: Poměr emailů, pro které platí, že všechny jeho odkazy jsou na whitelistu

analýze výše zmíněné trénovací a klasifikační fáze. Důležitou komponentou je fronta vstupních zpráv (message queue). Ta slouží jako jakýsi buffer pro ukládání příchozích emailů. Z této fronty jsou tyto emaily postupně odebírány klasifikátorem. Zavedením této fronty zpráv do systému dosáhneme lepšího škálování – v závislosti na množství příchozích emailů můžeme přidávat/ubírat počet instancí klasifikátoru. Ušetříme tak výpočetní zdroje a dokážeme dynamicky reagovat na výkyvy ve frekvenci příchozích emailů. Pro implementaci této fronty máme k dispozici hned několik řešení (např. open-source projekt RabbitMQ[35].

Zpětná vazba a měření úspěšnosti systému
Měření klasifikační úspěšnosti nasazeného systému nelze provádět stejně, jako se měří úspěšnost na testovacích datech. Nemáme totiž k dispozici informaci o opravdové třídě emailu (ham/phishing). Musíme se tak spolehnout na zpětnou vazbu uživatelů systému. Forma této zpětné vazby závisí na konkrétním nasazení systému. Nejjednodušší formou je pak vytvoření speciální phishingové složky. Do té by se přesouvala veškerá pošta vyhodnocená klasifikátorem jako phishing. Pokud by uživatel přemístil email z této složky do složky s běžnou komunikací, dostáváme signál o *false positive*. Naopak pokud uživatel označí email z běžné komunikace jako phishing, dostáváme signál o *false negative*. Jak false positives, tak false negatives je vhodné následně zařadit do trénovacího datasetu.

Z technického hlediska je velice důležité také vyhodnocovat dobu potřebnou ke zpracování jednoho emailu a s tím související celkovou propustnost systému (počet emailů za sekundu). Musí platit, že propustnost je vyšší než frekvence, s jakou emaily přichází.

[35] `https://www.rabbitmq.com/`



Obrázek 5.7: Architektura systému

Kapitola 6

Výsledky

V této kapitole naleznete výsledky měření, které jsem provedl na implementovaném prototypu. Ve svém návrhu klasifikačního procesu jsem představil myšlenku několika-úrovňového zpracování. V první fázi se provádí klasifikace na základě vlastností textu emailu, ve druhé na základě doménových informací a ve třetí na základě cílového webu. Do druhé a třetí fáze se přechází pouze v případě, že fáze předchozí nedosáhla dostatečné klasifikační jistoty.

Jak již bylo zmíněno, životnost phishingových webů je velice krátká (dle odborníků ze společnosti Excello se jedná o horizont pouhých několika hodin). Následkem toho je nemožné otestovat účinnost fáze dva a tři na starých datasetech. V sekci Architektura jsem představil metodu, jak tento problém vyřešit – ukládáním online otisku v momentě přidání nového vzorku do datasetu. Předpokládáme tedy postupný proces naplňování datasetu do velikosti dostatečné pro následné trénování klasifikátoru. Tento proces může trvat i několik měsíců a vyžaduje, abychom měli přístup ke zdroji proudu emailů. Časový horizont této diplomové práce byl však příliš krátký k vytvoření takového datasetu o dostatečné velikosti.

V této kapitole uvádím klasifikační výsledky pro první fázi – tedy klasifikaci dle příznaků vyextrahovaných z textu emailu. Klasifikační výsledky z této fáze tudíž představují spodní hranici klasifikačního potenciálu tohoto sytému, neboť předpokládám, že následující dvě fáze by výsledky ještě znatelně vylepšily.

6.1 Porovnání s ostatními pracemi

V této sekci se budu věnovat porovnání výsledků mých měření s pracemi se stejným zaměřením (výsledky naleznete v tabulce 6.2). Pro tato porovnání jsem se snažil vybrat co nejrelevantnější práce, ve kterých jsou zároveň dosažené výsledky prezentovány s co nejvyšší mírou detailu. Ač se to může zdát jako jednoduché kritérium, ve většině podobných prací najdeme pouze zhodnocení přesnosti (*accuracy*) klasifikátoru či množství false positives/negatives.

Srovnání s takovými pracemi by pak nemělo dostatečnou vypovídací hodnotu. Seznam prací vybraných ke srovnání naleznete níže.

Dataset použitý pro tato měření vnikl sloučením několika datasetů. Jako zdroj phishingových emailů jsem zvolil dataset Jose Nazaria. Jako zdroj ham emailů jsem pak zvolil osobní mailbox pro jeho aktuálnost. K němu jsem pak připojil ham emaily ze SpamAssasinu. Touto kombinací již dosáhneme velice dobrého poměru tříd. V tabulce 6.1 je k nalezení detailnější pohled na tento sloučený dataset.

Všechny hodnoty prezentované v této kapitole jsou pak výsledkem 10-fold Cross Validace.

Celkový počet vzorků	15776
Phishingových vzorků	8228
Ham vzorků	7548
Poměr tříd (phish:ham)	52 % : 48 %
Počet extrahovaných příznaků	32

Tabulka 6.1: Detaily sloučeného datasetu použitého pro měření

Abu-Nimeh
Abu-Nimeh et al. [11] použili 1171 phishingových emailů z korpusu Jose Nazaria a 1718 ham emailů z vlastních mailových schránek. Pro klasifikaci se rozhodli zaměřit na analýzu samotného textu. Pro klasifikaci tak vybírají 43 příznaků vyjadřující počet výskytů jednotlivých slov. Výběr těchto slov provádějí technikou odstranění stopových slov (používají seznam 424 nejpoužívanějších anglických stopových slov), následným stemmingem a ohodnocením TF-IDF. Porovnávají užití logistické regrese, Random Forest, Support Vector Machine, Bayesian Additive Regression Tree i neuronové sítě. Z hlediska AUC a F-meassure si nejlépe vedly neuronové sítě a Random Forest, proto v porovnávací tabulce používám právě tyto dva výsledky.

Fette
Fette et al. [10] ve své práci popisují svůj algoritmus PILFER založený na klasifikátoru Random Forest. Pro svůj projekt využili 6950 ham emailů ze SpamAssassin korpusu a 860 phishingových z korpusu Jose Nazaria. Algoritmus PILFER extrahoval pouhých 10 příznaků – přítomnost IP linků, stáří domény, klamavé odkazy, přítomnost odkazu s textem *here* nebo *click*, HTML email, celkový počet odkazů, počet různých domén v textu, maximální počet teček v odkazu, přítomnost JavaScriptu a hodnocení SpamAssassinu.

Bergholz
Bergholz et al. [16] ve svém projektu používají 16364 ham emailů a 3636

phishingových emailu získaných od partnerů projektu. Z nich extrahují 27 příznaků. Všechny tyto příznaky jsou extrahovány přímo ze zdrojového textu – autoři nepoužívají žádné příznaky vyžadující informace o cílových doménách/webech. Mezi nimi jsou 4 strukturální příznaky, 8 příznaků popisující odkazy, 4 příznaky pro použité technologie (HTML, scriptování, JavaScript, formuláře), 2 příznaky ze SpamAssassinu a 9 příznaků pro klíčová slova (account, update, confirm, verify, secur, log, notif, click, inconvenien). Dále pak používají dynamické markovské řetězce a vlastní CLTOM model, který indikuje třídu emailu podle co-existence specifických slov (např. pokud se v jedné zprávě vyskytují obě slova *account* a *click*, pak tato zpráva je pravděpodobně phishingová). Výstup těchto dvou modelů používají jako další dva příznaky. Pro samotnou klasifikaci pak využívají SVM klasifikátor.

Author	**Accuracy**	**F-measure**	**Precision**	**Recall**	**AUC**
Abu-Nimeh-NNet		85.45 %	94.15 %	78.28 %	0.9448
Abu-Nimeh-RF		90.24 %	91.71 %	88.88 %	0.9442
Fette	99.49 %	97.64 %	98.92 %	96.38 %	
Bergholz	99.88 %	99.46 %	100 %	98.93 %	
Decision tree	97.16 %	97.26 %	97.92 %	96.61 %	0.972
Naive Bayes	82.13 %	79.44 %	98.11 %	66.75 %	0.959
SVM	89.56 %	89.02 %	97.54 %	81.88 %	0.961

Tabulka 6.2: Porovnání výsledků klasifikátoru

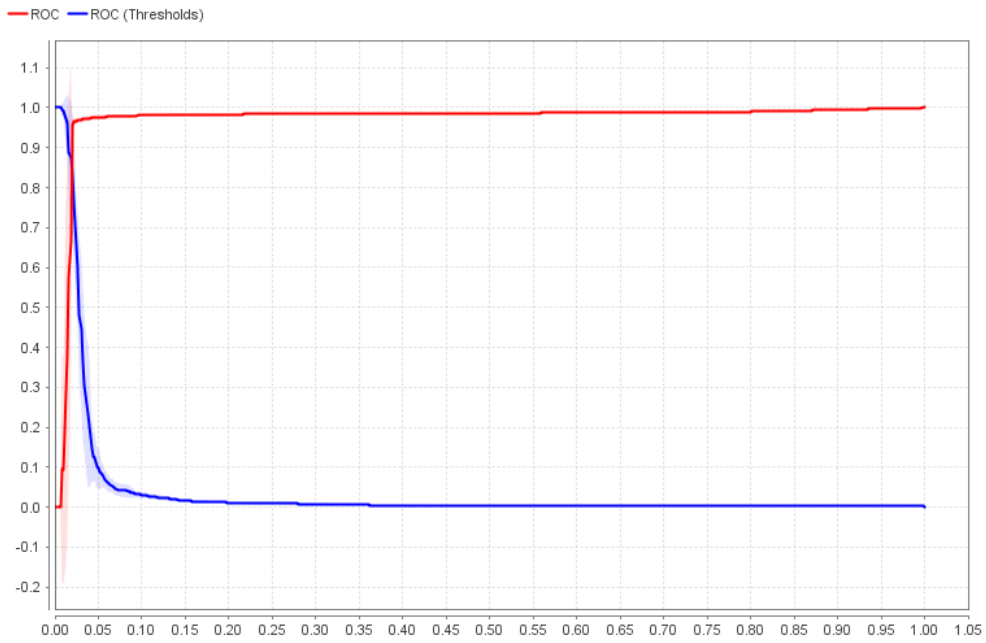
Zhodnocení

Z výsledků v tabulce 6.2 je evidentní, že rozhodovací strom si vede, na této úloze pro tato data, zdaleka nejlépe. Strmý průběh ROC křivky nalezneme na obrázku 6.1. Výsledný strom je značně komplexní. Dosahuje hloubky 18ti úrovní a celkový počet uzlů vystoupal na 650.

Překvapující je relativně špatný výsledek SVM. Dalo by se namítat, že to může být způsobeno špatným nastavením parametru C, na něž je SVM velice citlivé. Výsledky uvedené ve srovnávací tabulce jsou však nejlepším výsledkem optimalizačního procesu parametrů, jehož průběh je promítnut do grafu 6.2. Průběh optimalizace pro jednotlivé datasety pak naleznete v příloze B.

Nedílnou součástí výsledků je pak zhodnocení užitečnosti jednotlivých příznaků. K tomu využívám metriky *Information Gain Ratio*. Tabulku příznaků seřazenou dle této metriky naleznete v 6.3. Na nejvyšších místech vidíme především příznaky zaměřené na vlastnosti odkazů v emailu. Pokud z této tabulky pro přehlednost vyndáme pouze keyword-based příznaky, dostaneme srovnání důležitosti pro jednotlivá slova viz tabulka 6.4, které s velkým náskokem vévodí výraz *account*, což není až tak překvapující. Na druhou stranu výraz *sign*, který bychom čekali, že půjde ruku v ruce s *account*, se umístil velice nízko.

Není taky možné se spolehnout při výběru na intuici a ještě zřetelněji nám to poukazuje na nutnost nasazení text-miningu.



Obrázek 6.1: ROC křivka pro rozhodovací strom

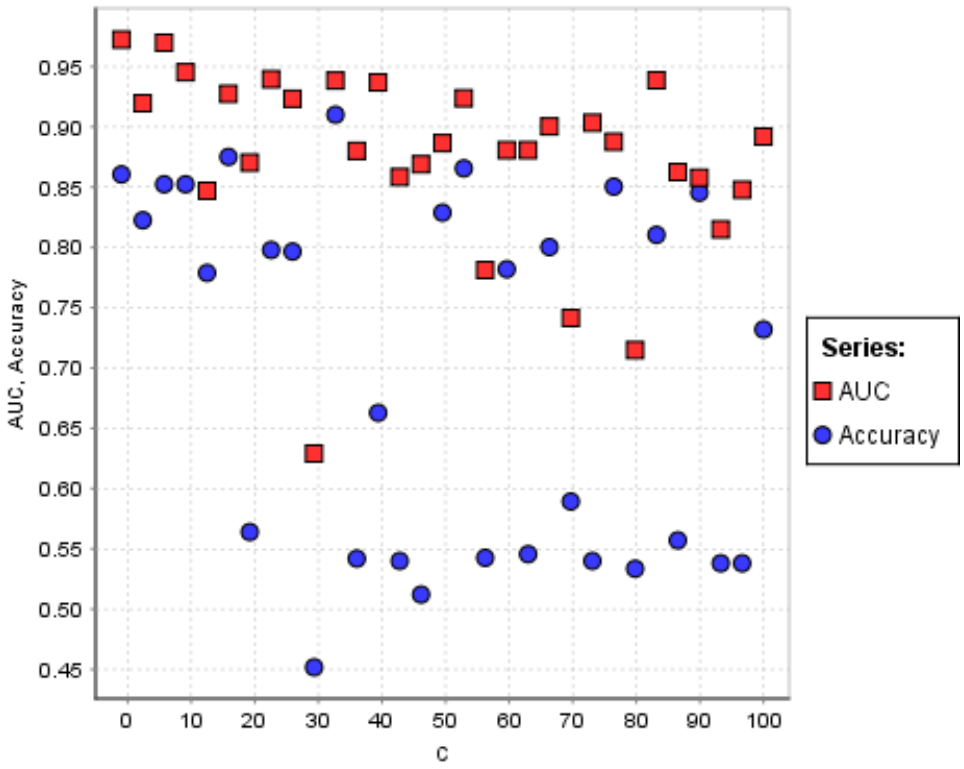
Random Forest

Výborné výsledky na rozhodovacím stromu podnítily myšlenku vyzkoušet klasifikátor Random Forest, kterým bychom měli teoreticky být schopni dosáhnout ještě lepších výsledků. Nejprve jsem provedl optimalizaci základních parametrů pro les s jedním stromem. Nejlepší nalezené nastavení jsem následně použil pro měření pro les se 100 stromy. Tímto postupem se mi ale nepodařilo dosáhnout uspokojivých výsledků (přesnost 90.5 %, F-measure 90.78 %).

Rozhodl jsem se provést optimalizaci pro širší výčet parametrů přímo pro les velikosti 100. Tak vzniklo 3456 kombinací nastavení pro optimalizaci. Na počítači se 4-jádrovým procesorem Intel Core i7 2.8GHz a 8GB RAM tento proces optimalizace (za použití 10-fold cross validace) trval přes 66 hodin. Podařilo se dosáhnout výborných výsledků uvedených v tabulce 6.5.

Zajímavostí je, že Random Forest na této úloze dosahoval nejlepších výsledků s nastavením *subset ratio* na 1.0 (tento parametr určuje poměrnou velikost množiny náhodně vybíraných atributů k testování). Nastavení 1.0 pak tedy znamená, že strom má při rozhodování k dispozici kompletní množinu atributů. De facto se dá říci, že tímto Random Forest zdegradoval do *baggingu* (bootstrap aggregating[36]).

[36] `https://en.wikipedia.org/wiki/Bootstrap_aggregating`



Obrázek 6.2: Optimalizace parametru C pro SVM

6.2 Výsledky na jednotlivých datasetech

V této sekci se zaměřuji na výkonnost klasifikátorů vůči jednotlivým datasetům. Účelem těchto měření je identifikovat, která data jsou pro jaké klasifikátory problematická či zda má výběr datasetů vůbec nějaký dopad na výsledky klasifikátorů.

Nazario+Personal Kombinace Nazariova phishingu a osobních emailů z větší míry kopíruje výsledky na sloučeném datasetu v ohledu všech 3 klasifikátorů. Viz tabulka 6.6.

Nazrio+SpamAssassin Na tomto datasetu pozorujeme mírný kvalitativní nárůst pro Naive Bayes a především pak zlepšení rozhodovacího stromu, který dosáhl 98.3 % f-measure a překonal tak výsledy Fette et al. na stejném datasetu. Viz tabulka 6.7

Nazario+Hillary Při použití ham datasetu Hillary Clintonové je zcela evidentní výkonnostní nárůst všech klasifikátorů viz tabulka 6.8. Tento rapidní

Rank	Feature	Information Gain Ratio weight
1	link_count	0.4680
2	all_domains_whitelisted	0.4088
3	term_account	0.3261
4	subdomains_max	0.2463
5	term_suspension	0.2290
6	ipLink_count	0.2247
7	term_bank	0.1903
8	term_information	0.1860
9	term_inconvenience	0.1782
10	term_verify	0.1726
11	term_security	0.1533
12	term_limit	0.1438
13	term_access	0.1430
14	term_log	0.1374
15	term_identity	0.1257
16	term_password	0.1244
17	term_member	0.1234
18	term_click	0.1127
19	term_service	0.1118
20	words_length_median	0.1117
21	term_credit	0.1092
22	form_count	0.1034
23	term_free	0.0964
24	word_count	0.0892
25	term_recently	0.0814
26	term_sign	0.0794
27	punctuation_count	0.0770
28	term_risk	0.07392
29	digits_count	0.0714
30	term_hour	0.0691
31	term_social	0.0652
32	has_javascript	0.0319

Tabulka 6.3: Information Gain Ratio

nárůst je, předpokládám, zapříčiněn řídkostí extrahovaných příznaků v návaznosti na nepřítomnost informace o HTML formátování. Bohatě formátované phishingové emaily je tak snazší rozpoznat od ham textových zpráv.

Zhodnocení

Na výsledcích prezentovaných v této sekci je zcela patrné, že výběr ham da-

Rank	Keyword	Information Gain Ratio weight
3	ACCOUNT	0.3261
5	SUSPENSION	0.2290
7	BANK	0.1903
8	INFORMATION	0.1860
9	INCONVENIENCE	0.1782
10	VERIFY	0.1726
11	SECURITY	0.1533
12	LIMIT	0.1438
13	ACCESS	0.1430
14	LOG	0.1374
15	IDENTITY	0.1257
16	PASSWORD	0.1244
17	MEMBER	0.1234
18	CLICK	0.1127
19	SERVICE	0.1118
21	CREDIT	0.1092
23	FREE	0.0964
25	RECENTLY	0.0814
26	SIGN	0.0794
28	RISK	0.07392
30	HOUR	0.0691
31	SOCIAL	0.0652

Tabulka 6.4: Information Gain Ratio pro klíčová slova

Classifier	Accuracy	F-meassure	Precision	Recall	AUC
Random Forest	98.23 %	98.29 %	98.69 %	97.90 %	0.997

Tabulka 6.5: Výsledky Random Forest se 100 stromy

tasetu hraje podstatnou roli pro výkonnost klasifikátorů. Dává nám poučení, že při pohledu na jakékoliv výsledky klasifikace bychom měli dbát zvýšené pozornosti a zjišťovat, na jakých datech bylo měření prováděno. To dokládá měření na datasetu Hillary Clintonové, kde lze dosáhnout velice dobrých testovacích výsledků. V reálném provozu bychom s modelem natrénovaným na těchto datech podobných výsledků již pravděpodobně nedosáhli.

Classifier	Accuracy	F-meassure	Precision	Recall	AUC
Decision Tree	96.93 %	98.36 %	98.20 %	98.36 %	0.917
Naive Bayes	70.29 %	81.13 %	99.4 %	68.55 %	0.914
SVM	86.23 %	92.07 %	98.92 %	86.19 %	0.937

Tabulka 6.6: Výsledky pro dataset Nazario+Personal

Classifier	Accuracy	F-meassure	Precision	Recall	AUC
Decision Tree	98.17 %	98.3 %	98.8 %	97.81 %	0.987
Naive Bayes	88.31 %	88.25 %	96.88 %	81.04 %	0.961
SVM	87.68 %	87.25 %	99.4 %	77.76 %	0.979

Tabulka 6.7: Výsledky pro dataset Nazario+SpamAssassin

Classifier	Accuracy	F-meassure	Precision	Recall	AUC
Decision Tree	98.64 %	98.85 %	99.19 %	98.52 %	0.991
Naive Bayes	97.07 %	97.51 %	98.34 %	96.69 %	0.977
SVM	85.90 %	86.52 %	99.98 %	76.25 %	0.991

Tabulka 6.8: Výsledky pro dataset Nazario+Hillary

Závěr

Provedl jsem analýzu phishingových technik a nástrojů. Současně byl představen výčet příznaků, podle kterých je možné phishing detekovat. Těchto informací jsem využil k vytvoření prototypové implementace, na které jsem provedl měření úspěšnosti na několika datasetech. Prezentoval jsem srovnání s výsledky ostatních prací se stejnou tématikou.

Pro tento typ úlohy dosahovaly nejlepších výsledků klasifikační algoritmy z rodiny rozhodovacích stromů. S klasifikátorem Random Forest se mi tak podařilo dosáhnout klasifikační přesnosti 98.23 % a F-measure 98.29 %. Význam těchto kvalitativních metrik byl v práci důkladně rozebrán a v návaznosti na to byla diskutována i problematika srovnávání s výsledky ostatních prací, pro které je nutný pohled na více výsledných metrik.

Techniky strojového učení se tak ukazují jako velice účinné. Předpokládám, že přidáním dalších extrahovaných příznaků a začlenění pokročilejších technik, jako je text-mining, by bylo možné efektivitu systému ještě zvýšit. V práci jsem využíval krom strojového učení i konvenčních technik, jako je black a whitelisting, které vedou jak k rapidnímu zvýšení klasifikační úspěšnosti, tak efektivnímu chodu a šetření výpočetního výkonu.

Literatura

[1] Rekouche, K.: Early Phishing. *CoRR*, ročník abs/1106.4692, 2011. Dostupné z: http://arxiv.org/abs/1106.4692

[2] Felix, J.; Hauck, C.: System security: a hacker's perspective. *Interex Proceedings*, ročník 1, 1987: s. 6–6.

[3] Bursztein, E.; Benko, B.; Margolis, D.; aj.: Handcrafted fraud and extortion: Manual account hijacking in the wild. In *Proceedings of the 2014 Conference on Internet Measurement Conference*, ACM, 2014, s. 347–358. Dostupné z: http://services.google.com/fh/files/blogs/google_hijacking_study_2014.pdf

[4] PhishTank: Statistics for June 2015. https://www.phishtank.com/stats/2015/06/.

[5] Shcherbakova, T.; Vergelis, M.; Demidova, N.: Spam and Phishing in the First Quarter of 2015. https://securelist.com/analysis/quarterly-spam-reports/69932/spam-and-phishing-in-the-first-quarter-of-2015/, 2015.

[6] Symantec: Internet Security Threat Report 2015. https://www4.symantec.com/mktginfo/whitepaper/ISTR/21347931_GA-internet-security-threat-report-volume-20-2015-appendices.pdf, 2015.

[7] Herley, C.: Why do nigerian scammers say they are from nigeria? In *WEIS*, 2012.

[8] Veitch, J.: This is what happens when you reply to spam email, December 2015, [Video file].

[9] Liszewski, A.: Today's Hero Made an AI That Annoys Telemarketers For As Long As Possible. http://gizmodo.com/todays-hero-made-an-ai-that-annoys-telemarketers-for-as-1756344562, 2016.

[10] Fette, I.; Sadeh, N.; Tomasic, A.: Learning to detect phishing emails. In *Proceedings of the 16th international conference on World Wide Web*, ACM, 2007, s. 649–656. Dostupné z: http://www.cs.cmu.edu/~tomasic/doc/2007/FetteSadehTomasicWWW2007.pdf

[11] Abu-Nimeh, S.; Nappa, D.; Wang, X.; aj.: A comparison of machine learning techniques for phishing detection. In *Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit*, ACM, 2007, s. 60–69. Dostupné z: http://lyle.smu.edu/~sabunime/pub/ecrime07_mlphish.pdf

[12] Gonzalez, H.; Nance, K.; Nazario, J.: Phishing by form: The abuse of form sites. In *Malicious and Unwanted Software (MALWARE), 2011 6th International Conference on*, IEEE, 2011, s. 95–101. Dostupné z: https://monkey.org/~jose/tmp/PHISHING-FINAL-03-KN.pdf

[13] Toolan, F.; Carthy, J.: Phishing detection using classifier ensembles. In *eCrime Researchers Summit, 2009. eCRIME'09.*, IEEE, 2009, s. 1–9. Dostupné z: ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5342607

[14] Azad, B.: Identifying Phishing Attacks. http://cs229.stanford.edu/proj2013/Azad-IdentifyingPhishingAttacks.pdf.

[15] Zhang, Y.; Hong, J. I.; Cranor, L. F.: Cantina: a content-based approach to detecting phishing web sites. In *Proceedings of the 16th international conference on World Wide Web*, ACM, 2007, s. 639–648. Dostupné z: https://www.cs.cmu.edu/~jasonh/publications/www2007-cantina-final.pdf

[16] Bergholz, A.; De Beer, J.; Glahn, S.; aj.: New filtering approaches for phishing email. *Journal of computer security*, ročník 18, č. 1, 2010: s. 7–35. Dostupné z: http://cordis.europa.eu/pub/fp7/ict/docs/security/antiphish-paper_en.pdf

[17] Chandrasekaran, M.; Narayanan, K.; Upadhyaya, S.: Phishing email detection based on structural properties. In *NYS Cyber Security Conference*, 2006, s. 1–7. Dostupné z: http://www.albany.edu/iasymposium/proceedings/2006/chandrasekaran.pdf

[18] Rokach, L.; Maimon, O.: Decision trees. In *Data mining and knowledge discovery handbook*, Springer, 2005, s. 165–192. Dostupné z: http://www.ise.bgu.ac.il/faculty/liorr/hbchap9.pdf

[19] Oliver, J.; Cheng, C.; Chen, Y.: TLSH – A Locality Sensitive Hash. In *Cybercrime and Trustworthy Computing Workshop (CTC), 2013 Fourth*, Nov 2013, s. 7–13, doi:10.1109/CTC.2013.9.

Příloha A

Seznam použitých zkratek

GUI Graphical User Interface

UI User Interface

XML Extensible Markup Language

AOL America Online

TP True Positive

TN True Negative

FP False Positive

FN False Negative

DNS Domain Name System

URL Uniform Resource Locator

CLI Command Line Interface

API Application Program Interface

ISP Internet Service Provider

TF-IDF Term Frequency–Inverse Document Frequency

SVM Support Vector Machine

ROC Receiver Operation Characteristics

AUC Area Under Curve

PDF Portable Document Format

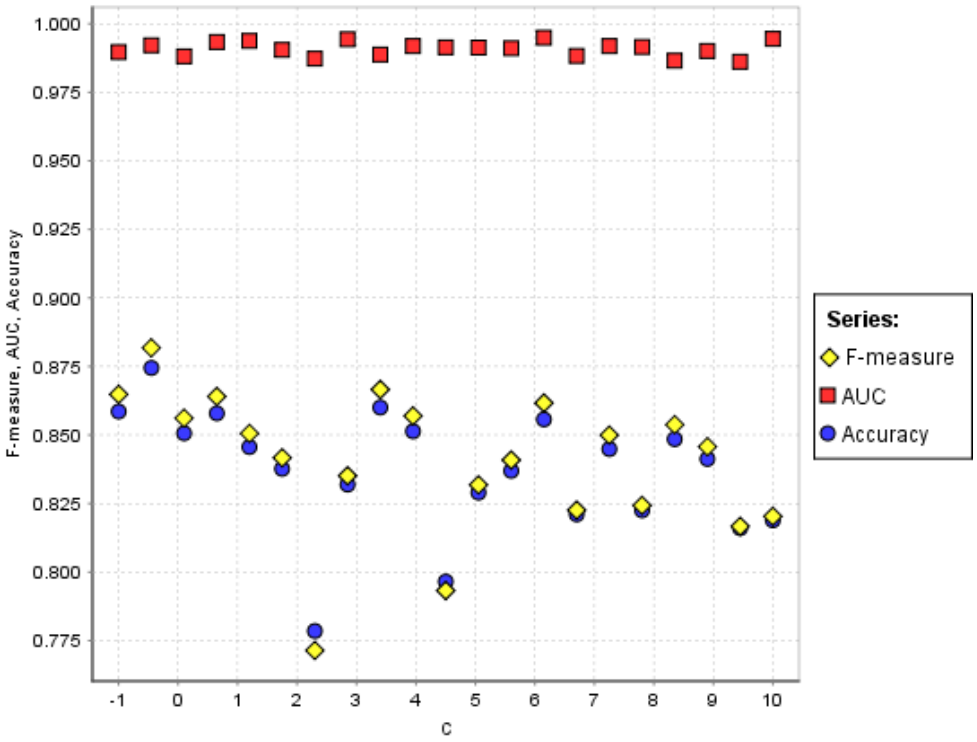
HTML HyperText Markup Language

CSV Comma Separated Values

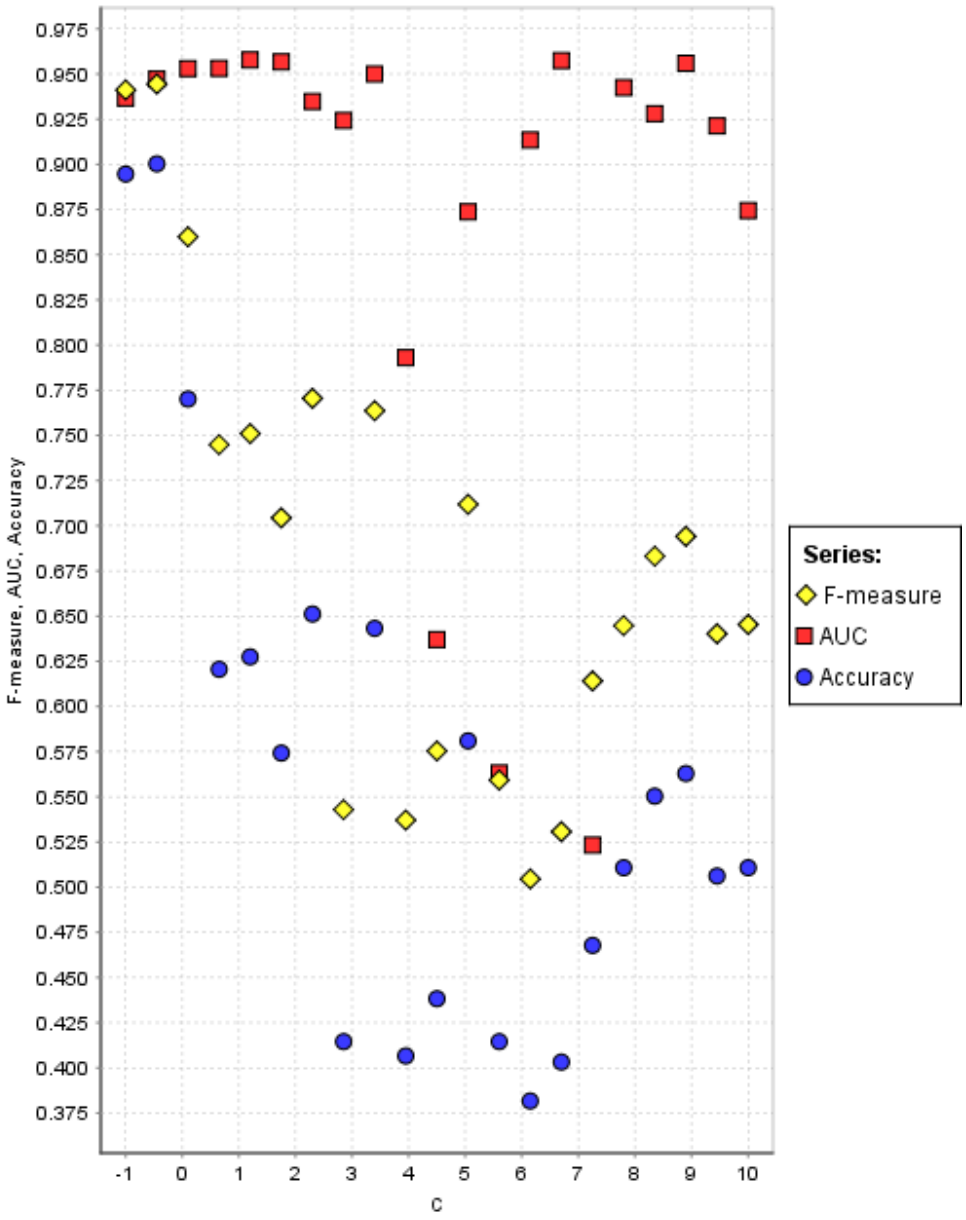
DDoS Distributed Denial-of-Service

Příloha B

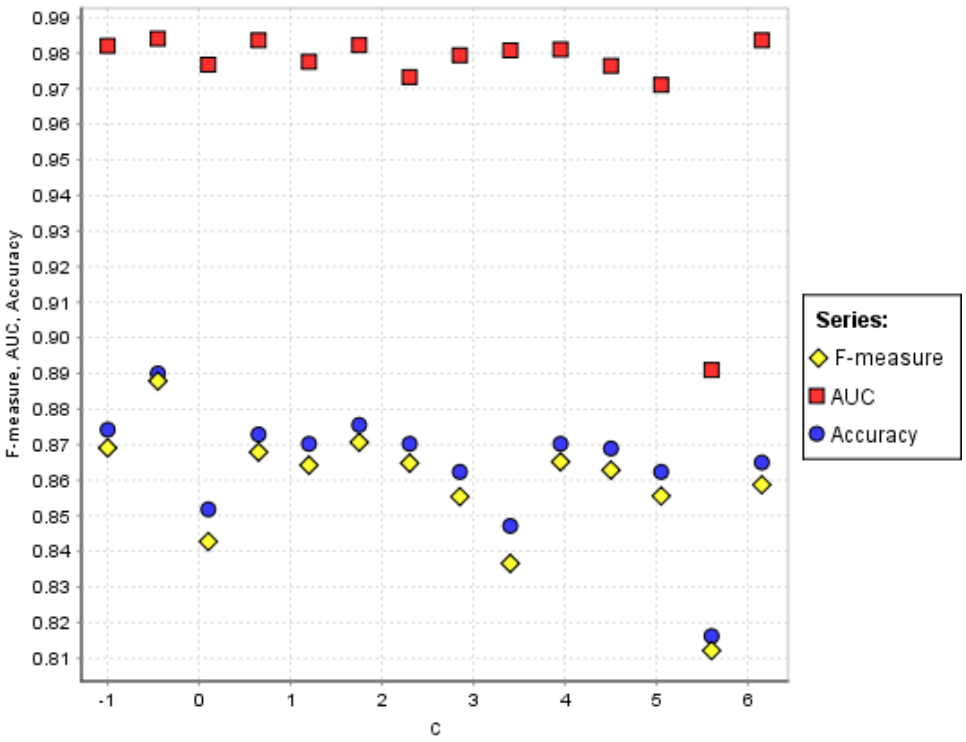
Průběh měření



Obrázek B.1: Optimalizace parametru C pro dataset Hillary



Obrázek B.2: Optimalizace parametru C pro dataset Personal



Obrázek B.3: Optimalizace parametru C pro dataset SpamAssassin

Příloha C

Obsah přiloženého CD

- `readme.txt` ..... stručný popis obsahu CD
- `src`
  - `impl` ..... zdrojové kódy implementace
  - `thesis` ..... zdrojová forma práce ve formátu LaTeX
- `text` ..... text práce
  - `thesis.pdf` ..... text práce ve formátu PDF
  - `thesis.ps` ..... text práce ve formátu PS
  - `assets` ..... obrázky použité v práci