Univerzita Pavla Jozefa Šafárika v Košiciach

Prírodovedecká fakulta

SPRACOVANIE BEZPEČNOSTNÝCH INCIDENTOV CSIRT/CERT TÍMAMI

BAKALÁRSKA PRÁCA

Študijný odbor:	Aplikovaná Informatika
Školiace pracovisko:	Ústav Informatiky
Vedúci záverečnej práce:	RNDr. JUDr. Pavol Sokol PhD.
Konzultant:	Mgr. Ladislav Bačo

Košice 2018 Michal Pavúk

Vyhlásenie

Vyhlasujem, že som túto bakalársku prácu vypracoval samostatne na základe vedomostí získaných štúdiom a s pomocou uvedenej literatúry.

Michal Pavúk

Poďakovanie

Touto cestou by som sa rád poďakoval vedúcemu práce RNDr. JUDr. Pavlovi Sokolovi PhD. za poradenstvo pri vyhotovovaní práce, členom slovenského vládneho CSIRT-u, obvzlášť konzultantovi Mgr. Ladislavovi Bačovi za zdieľanie jeho praktických skúseností, členom mojej rodiny za ich podporu a trpezlivosť a v neposlednom rade každému, kto sa pričinil na spríjemnení môjho posledného roku štúdia na Univerzite Pavla Jozefa Šafárika v Košiciach.

Univerzita P. J. Šafárika v Košiciach
Prírodovedecká fakulta

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta:	Michal Pavúk
Študijný program:	Aplikovaná informatika (Jednoodborové štúdium, bakalársky I. st., denná forma)
Študijný odbor:	9.2.9. aplikovaná informatika
Typ záverečnej práce:	Bakalárska práca
Jazyk záverečnej práce:	slovenský
Sekundárny jazyk:	anglický

Názov: Spracovanie bezpečnostných incidentov CERT/CSIRT tímami

Názov EN: Security incidents handling by CERT / CSIRT teams

Cieľ: 1) Definovať požiadavky CERT/CSIRT tímov v oblasti spracovania bezpečnostných incidentov
2) Porovnať aktuálne prístupy k spracovaniu bezpečnostných incidentov CERT/CSIRT tímami.
3) Navrhnúť a implementovať systém na riešenie bezpečnostných incidentov pre CSIRT-UPJS. Systém vyhodnotiť.

Literatúra:
[1] MURDOCH, D. W. Blue Team Handbook: Incident Response Edition: a Condensed Field Guide for the Cyber Security Incident Responder. CreateSpace Independent Publishing, 2014.
[2] MAJ, Miroslav; REIJERS, Roeland; STIKVOORT, Don. Good practice guide for incident management. 2010.
[3] BOLLINGER, Jeff; ENRIGHT, Brandon; VALITES, Matthew. Crafting the InfoSec Playbook: Security Monitoring and Incident Response Master Plan. " O'Reilly Media, Inc.", 2015.
[4] CICHONSKI, Paul, et al. Computer security incident handling guide. NIST Special Publication, 2012, 800: 61.

Vedúci:	RNDr. JUDr. Pavol Sokol, PhD.
Konzultant:	Mgr. Ladislav Bačo
Ústav :	ÚINF - Ústav informatiky
Riaditeľ ústavu:	prof. RNDr. Viliam Geffert, DrSc.

Dátum schválenia: 09.05.2018

Abstrakt

S rastúcim počtom uchovávaných informácií, rastie aj potreba ich zabezpečenia a následne aj potreba formalizácie a štandardizácie bezpečnostných riešení. Táto úloha spadá pod kompetencie CSIRT/CERT tímov, spravujúcich bezpečnosť infraštruktúry vo svojej oblasti. Ich procesy sú komplexné a silne štandardizované, no aj napriek tomu každý tím, a jemu prislúchajúca organizácia, má svoje špecifiká. V práci analyzujeme požiadavky týchto tímov v oblasti riešenia bezpečnostných incidentov a postupy ich spracovania. Uvádzame spôsoby riešenia problematiky správy incidentov a im prislúchajúcich údajov. Výsledkom práce je návrh a implementácia systémov na riešenie bezpečnostných incidentov pre potreby tímu CSIRT-UPJS založených na platforme RequestTracker a IntelMQ.

Kľúčové slová: *CSIRT, CERT, kybernetická bezpečnosť, riešenie incidentov*

Abstract

With an increasing amount of stored data, the need for their protection becomes more prevalent. The same can be said for standardisation and formalisation of security-enhancing solutions. This is a task for CSIRT/CERT teams, managing cybersecurity threats to an infrastructure of their constituents. Despite heavy standardisation of complex incident handling processes, these teams face difficulties when coping with specifics of their organisation. In this thesis, we analyse the requirements of CSIRTs with focus on incident triaging and handling process. We present a comparison of solutions to problems, arising from the introduction of incident handling processes to an organisation. The result of this thesis are design and implementation of systems based on the RequestTracker and IntelMQ platforms. These are custommized to suit the needs of CSIRT-UPJS team.

Keywords: *CSIRT, CERT, cybersecurity, incident handling*

Obsah

Zoznam obrázkov

Zoznam tabuliek

Zoznam úkážok kódu

Úvod

Informatizácia spoločnosti a rozvoj informačných systémov so sebou priniesli aj mnohé nežiadúce faktory predstavujúce hrozbu pre počítačové siete, systémy a ich používateľov. Tieto faktory sú úzko späté s kyberkriminálnou činnosťou, ktorá ani v súčasnosti nie je dostatočne ošetrená príslušnou legislatívou. Kybernetické útoky na informačné systémy predstavujú čoraz väčšie riziko pre ich vlastníkov a používateľov. Systémy sa stávajú komplexnejšími a objem spracovaných a uchovávaných údajov sa zväčšuje. Prirodzene teda vzniká potreba ochrany týchto systémov a ich infraštruktúry. Touto problematikou sa zaoberajú bezpečnostní experti v CSIRT (*Computer Security Incident Response Team*), resp. CERT (*Computer Emergency Response Team*) tímoch.

Ich úlohy a kompetencie sú rôznorodé a závisia od požiadaviek spravovaných inštitúcií. Avšak jedným z hlavných cieľov takmer každého CSIRT tímu, je čo najrýchlejšie obnoviť činnosť systému po bezpečnostnom incidente a následne analyzovať a zdokumentovať incident. Procesy obnovy činnosti, posudzovania a riešenia incidentu sú definované v dokumentoch, ktoré tvoria základ informačnej bezpečnosti v organizácii (*tzv. bezpečnostná politika*). Aj napriek detailnému popisu postupov, si riešenie incidentov a ich dopadu vyžaduje hlboké interdisciplinárne znalosti. Tie sú zväčša zabezpečené špecializáciou jednotlivých členov tímu na rôzne oblasti systému. Efektívna komunikácia a správna koordinácia činnosti je teda nevyhnutnou prerekvizitou úspešnej správy systému a potencionálnych hrozieb. K tomu CSIRT tímy využívajú rôzne podporné nástroje.

Prácu sme konceptuálne rozdelili do štyroch kapitol. V prvej kapitole ponúkame úvod do problematiky riadenia bezpečnostných incidentov, definujeme základné pojmy, služby poskytované CSIRT tímami a taxonómiu počítačových bezpečnostných incidentov, ako aj etapy ich riešenia.

Druhá kapitola práce sa venuje prostriedkom pre riešenie počítačových bezpečnostných incidentov. Obsahuje prehľad súčasného stavu problematiky a kompiláciu jednotlivých riešení pre správu bezpečnostných incidentov.

V tretej kapitole predstavujeme súčasný proces riešenia incidentov v CSIRT-UPJS

a následne ponúkame návrh riešenia pre zefektívnenie tohto procesu. Analyzujeme požiadavky kladené na riešenie a spôsob, akým ich naša koncepcia napĺňa.

Vo štvrtej a zároveň poslednej kapitole popisujeme implementačné detaily navrhovaného riešenia. Prechádzame od inštalácie systémov, cez ich konfiguráciu až po vyhotovené integrácie s existujúcimi systémami, ktoré sú zavedené na UPJŠ.

1 Riadenie bezpečnostných incidentov

1.1 Incident

Z pohľadu kybernetickej bezpečnosti sa rozlišujú udalosti a počítačové bezpečnostné incidenty. Za **udalosť (event)** sa považuje akákoľvek pozorovateľná situácia, ktorá nastane v počítačovom systéme alebo sieti. Napr. pripojenie zariadenia, prihlásenie užívateľa, príchod požiadavky na server, odoslanie e-mailovej správy, či aktualizácia systému. Za **počítačový bezpečnostný incident (computer security incident)** sa považuje udalosť, ktorá buď narúša bezpečnostnú politiku, alebo je predpoklad, že by ju mohla narušiť, resp. znemožniť jej plnohodnotné vykonávanie. [1] Za počítačové bezpečnostné incidenty sa považujú napr. neobvykle množstvo podvodných e-mailov cielených na užívateľov siete, znemožnenie prevádzky kvôli útoku ransomware, alebo únik citlivých informácií bývalým zamestnancom.

V tejto práci sú pojmy incident, bezpečnostný incident a počítačový incident zameniteľné a ekvivalentné.

1.2 Taxonómia incidentov

Počet druhov počítačových bezpečnostných incidentov je vysoký. Z tohto dôvodu vznikla potreba systematickej taxonómie incidentov, obvykle podľa spoločných znakov. Na túto potrebu odpovedalo niekoľko organizácií. To viedlo k značnej fragmentácii a duplicite týchto delení (*viď. tabuľka 1.*). V rámci práce analyzujeme delenie podľa organizácií ENISA, Europol a eCSIRT. Porovnávacími metódami sme zistili, že taxonómia agentúry ENISA je podmnožinou delenia, ktoré predstavilo eCSIRT. To je spôsobené faktom, že na tvorbe delenia eCSIRT sa podieľalo niekoľko ľudí z ENISA:

1.2.1 Taxonómia incidentov podľa ENISA

Agentúra Európskej Únie pre Sieťovú a Informačnú bezpečnosť (European Union Agency for Network and Information Security – ENISA) je centrom pre znalcov v od-

bore sieťovej a kybernetickej bezpečnosti v rámci Európy. [2] Taxonómia publikovaná organizáciou ENISA vznikla v spolupráci viacerých CSIRT-ov počas ENISA/EC3 workshopu a bola publikovaná 1. 1. 2018. [3] Táto taxonómia rozlišuje nasledujúce hlavné kategórie incidentov:

- nevhodný obsah (spam, diskriminácia, vyhrážky, detská pornografia, propagácia násilia)
- škodlivý softvér (vírus, trójsky kôň, spyware, rootkit)
- zhromažďovanie informácií (skenovanie, sniffing, sociálne inžinierstvo)
- pokusy o vniknutie do systému (využitie známych zraniteľností, pokusy o prihlásenie, nové exploity)
- vniknutie do systému (kompromitácia účtu, kompromitácia aplikácie, bot)
- ohrozenie/obmedzenie dostupnosti (odmietnutie služby (DOS), distribuované odmietnutie služby (DDOS), sabotáž)
- ochrana údajov (neoprávnený prístup k údajom, neoprávnená zmena údajov)
- podvod (neoprávnené využívanie prostriedkov, porušenie autorských práv, phishing)
- iné (napr. testovanie).

1.2.2 Taxonómia incidentov podľa Europolu

Europol je európska agentúra pre vymožiteľnosť práva občanov štátov Európskej Únie[4]. Do jej kompetencií spadá aj oblasť kyberkriminality. Europol, v spolupráci s agentúrou ENISA, taktiež vydal taxonómiu bezpečnostných incidentov. Tá sa zameriava na bezpečnostné incidenty z pohľadu orgánov činných v trestnom konaní. Je napísaná v súlade s platnou legislatívnou úpravou Európskej Únie a na jej zostavovaní sa podieľal aj Policajný zbor Slovenskej republiky [5]. Táto taxonómia pre orgány činné v trestnom konaní a CSIRT-y, delí bezpečnostné incidenty do deviatich kategórií:

- malware: infekcia, distribúcia, kontrola (C&C), škodlivé pripojenie
- dostupnosť: odmietnutie služby, sabotáž

- zber údajov: skenovanie, sniffing, phishing
- pokus o prienik: pokus o zneužitie zraniteľnosti, pokus o prihlásenie
- prienik: úspešné zneužitie zraniteľnosti, kompromitácia účtu
- informačná bezpečnosť: neautorizovaný prístup, neautorizovaná zmena/odstránenie
- podvod: neoprávnené využívanie prostriedkov, krádež identity
- nevhodný obsah: spam, porušenie autorských práv, detská pornografia, rasizmus, propagácia násilia
- iné: neklasifikovaný incident, nešpecifikovaný incident.

1.2.3 eCSIRT

Európska sieť CSIRT-ov vystupujúca pod skratkou eCSIRT je iniciatíva, ktorá vznikla v roku 2003. Skladá sa z viacerých CSIRT-ov operujúcich v Európe. V súčasnosti jej primárnou úlohou je správa systému na detekciu prieniku (*IDS - Intrusion Detection System*) a výmena zozbieraných údajov. Aby táto kooperácia bola možná, členské tímy eCSIRT sa dohodli na jednotnej taxonómii. Tá vychádza z delenia agentúry ENISA[6].

Tabuľka 1.1: Porovnanie taxonómií bezpečnostných incidentov

Kategória	ENISA	eCSIRT	Europol
Nevhodný obsah	Spam		
	Detská pornografia		
	Diskriminácia		
	Vyhrážky		
			Porušenie autorských práv
		Propagácia násilia	
Malware	Vírus		
	Trójsky kôň		Infekcia
	Spyware		
	Rootkit		
		Červ	
		Dialer	
			Distribúcia
			Kontrola
			Škodlivé pripojenie
Zber informácií	Skenovanie		
	Sniffing		
	Sociálne inžinierstvo		
			Phishing
Pokus o prienik	Známe zraniteľnosti		Zneužitie zraniteľnosti
	Nové exploity		
	Pokusy o prihlásenie		
Vniknutie	Kompromitácia účtu	Komp. privilegovaného účtu	Kompromitácia účtu
		Komp. neprivilegovaného účtu	
	Kompromitácia aplikácie		Zneužitie zraniteľnosti
	Bot		
Ohrozenie dostupnosti	Odmietnutie služby		
	Distribuované Odmietnutie Služby		
	Sabotáž		
		Neúmyselný výpadok	
Informačná bezpečnosť	Neoprávnený prístup		
	Neoprávnená zmena		
Podvod	Neoprávnené využívanie prostriedkov		
	Porušenie autorských práv		
	Phishing		
		Krádež identity	
Zraniteľnosť	Neúmyselná		
		Úmyselná	
Iné	Nezapadajúci do kategórií		
			Nešpecifikovaný
	Test		

1.3 CSIRT/CERT tímy

Správa a častokrát aj riešenie bezpečnostných incidentov spadá pod inštitúcie, alebo časti organizácie zvané CSIRT (*computer security incident response team*), alebo

CERT (*computer emergency response team*). Oba akronymy sú si z pohľadu aplikačnej praxe ekvivalentné. Skratka CERT je skôr používaná v Spojených Štátoch Amerických, zvyšok sveta používa skôr druhý pojem, avšak nie je to pravidlom. V tejto práci budeme používať označenie CSIRT.

CSIRT tímy sa líšia rozsahom kompetencií, veľkosťou (počtom členov), či zameraním, no všetky musia implementovať proces riadenia bezpečnostných incidentov v rámci ich rozsahu. Rozsah, činnosť, služby a postupy tímu sú popísané v dokumente **RFC2350** - expectations for computer security incident response[7]. Každý CSIRT by mal implementovať prevádzku popísanú týmto dokumentom. Ten tvorí štandardizovaný základ CSIRT-ov ako organizácií.

1.4 Služby poskytované CSIRT tímami

Služby poskytované CSIRT tímami sa zoskupujú do 3 kategórií [8][9]:

- reaktívne služby,
- proaktívne služby a
- služby zamerané na manažment kvality zabezpečenia.

Reaktívne služby. Sú poskytované na základe žiadosti, resp. udalosti ako odpoveď na bezpečnostný incident. Tvoria základ činnosti CSIRT-u. Medzi reaktívne činnosti patrí napríklad:

- správa incidentov,
- oznámenia a varovania,
- správa zraniteľností a
- správa artefaktov.

V rámci kategórie **proaktívnych služieb** sa nachádzajú služby poskytované za účelom asistencie pri príprave, zabezpečení či implementácii systémov. Ich výsledkom je redukcia pravdepodobnosti výskytu bezpečnostných incidentov na zabezpečených systémoch. ENISA stanovuje nasledujúce služby ako kľúčové:

- upozornenia na hrozby a zraniteľnosti,
- monitorovanie technológií a

- detekcia prienikov.

Medzi nekritické proaktívne služby radíme:

- bezpečnostné audity,
- konfigurácia a správa bezpečnostných nástrojov, aplikácií a infraštruktúry,
- vývoj bezpečnostných nástrojov a
- šírenie osvety o bezpečnosti.

Medzi **služby zamerané na manažment kvality zabezpečenia** spadajú služby obohacujúce existujúce procesy nezávislé od procesu riadenia incidentov. Sú zväčša vykonávané inými skupinami organizácie, predovšetkým IT oddelením. Príkladmi takýchto služieb sú:

- analýza rizík,
- plán kontinuity a plán obnovy činnosti,
- konzultačná činnosť,
- vzdelávanie a tréning a
- certifikácia produktov.

1.5 Združenia CSIRT-ov

1.5.1 FIRST

FIRST (*Forum of Incident Response and Security Teams*) je prvá medzinárodná konfederácia CSIRT-ov, ktorá vznikla v roku 1990, krátko po vzniku prvého CERT-u. Je zodpovedná za tvorbu technických prostriedkov, postupov, procesov a metodológii pre bezpečnostné tímy[10]. Intenzívne sa venuje aj vzdelávacej činnosti, a pre svojich členov usporadúva pravidelné školenia. V súčasnosti má FIRST viac ako 420 členov[11]. Členmi sa môžu stať ako tímy, tak aj jednotlivci. Pre prijatie do tejto komunity musí byť tím nominovaný aspoň dvomi existujúcimi členmi, v prípade jednotlivcov postačuje jeden sponzor[12]. Novoprijaté tímy musia zaplatiť jednorazový poplatok vo výške 800 amerických dolárov, potom 2000 amerických dolárov každý ďalší rok členstva.

1.5.2 TF-CSIRT

TF-CSIRT je pracovná skupina založená v roku 2000 organizáciou GÉANT (vtedy TERENA)[13]. TF-CSIRT poskytuje niekoľko služieb zameraných na skvalitnenie činnosti CSIRT tímov. Ich tréningový program TRANSITS obsahuje kvalitné vzdelávacie materiály ako pre nových tak aj skúsených členov CSIRT. Ďalšou zo služieb TF-CSIRT je Trusted Introducer - zoznam CSIRT/CERT tímov z celého sveta. Momentálne obsahuje kontaktné informácie takmer 350 tímov, delených do štyroch kategórií: kandidáti, listované, akreditované a certifikované tímy. Zápis do zoznamu je bezplatný, avšak akreditácia a certifikácia už sú spoplatnené. Platiace tímy získavajú prístup do neverejnej časti portálu a môžu sa zúčastniť uzavretých stretnutí v rámci užšej komunity.

1.6 Požiadavky

Požiadavky CSIRT/CERT vyplývajú z požiadaviek na nich kladených. Sú založené najmä na očakávaniach ich konštituentov. Konštituenti sú skupiny ľudí a/alebo organizácií, ktorým sú poskytované služby CSIRT-u. Inými slovami: sú to zákazníci CSIRT-u. Každý CSIRT by mal vedieť obhájiť svoju činnosť svojmu riadiacemu, resp. nadriadenému orgánu.

1.6.1 FIRST

FIRST publikoval dokument zvaný *FIRST CSIRT Framework* v ktorom poskytuje popis služieb a činností CSIRT-ov, spolu s očakávaniami, ktoré sú na nich kladené. Služby delí do siedmich kategórií:

1. správa incidentov,
2. analýza,
3. zabezpečenie informácií,
4. situačné povedomie,
5. komunikácia,
6. rozvoj kompetencií a
7. výskum-vývoj.

1.6.2 RFC2350

Dokument RFC2350 - Expectations for Computer Security Incident Response popisuje všeobecné očakávania Internetovej komunity od CSIRT tímov[7]. Slúži na poskytnutie stručného štrukturovaného prehľadu činností CSIRT tímov pre ich konštituentov. V dokumente sú popísané odporúčania pre publikáciu bezpečnostných pravidiel a postupov, ako aj manažment vzťahov s inými CSIRT-mi. Užitočnou súčasťou je aj návod pre zostavenie charty tímu, ktorá popisuje jeho ciele, pravidlá, metódy a služby.

1.7 Etapy riešenia incidentov

Aj napriek rozmanitej povahe a rozličným vlastnostiam CSIRT-ov pri riešení bezpečnostných incidentov, sa objavujú vzory procesov ich riešenia. Tie sú neskôr štandardizované na rôznych úrovniach. Podľa NIST (*National Institute for Standards and Technology*) sa proces riešenia incidentov delí do štyroch etáp[1]:

- Príprava riešenia incidentu,
- detekcia a analýza incidentu,
- zaistenie, odstránenie, obnova a
- analýza po incidente.

Tieto etapy tvoria cyklický proces, ktorý prebieha kontinuálne. Ďalšia etapa je zahájená v prípade, že nastane potreba tak učiniť. Napr. zo štádia prípravy proces postúpi do riešenia analýzy incidentu v prípade, že bol nejaký detegovaný. Naopak ak sa počas obnovy dôsledkov incidentu zistí, že škody sú väčšie než odhad určený prvotnou analýzou, proces sa vráti do predošlej etapy.

Existujú aj iné delenia. Ich etapy sú však zahrnuté vo vyššie uvedenom modeli. Mitropoulos a kol. [14] delí tretiu fázu na tri samostatné etapy; analýza po incidente chýba. Delenie podľa Maj a kol. [9] neobsahuje fázu prípravy a analýzu po incidente. Keďže, prieniky v deleniach procesu riešenia incidentov sú natoľko evidentné, podrobnejšie analyzujeme len delenie podľa NIST.

1.7.1 Príprava riešenia incidentu

Väčšina metodológii pre riadenie incidentov kladie dôraz na prípravu riešenia incidentu. Aj keď bezpečnostné tímy častokrát nie sú priamo zodpovedné za zabezpe-

čenie systémov, dôsledná príprava a prevencia incidentov je základom ich úspešného zvládnutia.[1] Organizácia by mala disponovať viacerými komunikačnými kanálmi, pre prípad zlyhania jedného z nich. Infraštruktúra pre riadenie incidentov, používaná CSIRT-om by mala byť vždy v pohotovostnom stave. Rovnako tak aj vybavenie používané ako terénnymi pracovníkmi CSIRT-u, tak aj zamestnancami komunikujúcimi s konštituentmi a tretími stranami. Na príprave sa významnou mierou podieľajú aj služby poskytované CSIRT-om:

- analýza rizík,
- sieťová bezpečnosť,
- prevencia malvéru a
- zvyšovanie bezpečnostného povedomia užívateľov.

1.7.2 Detekcia a analýza incidentu

Počet možných útočných vektorov (spôsobov vykonania útoku) rastie so zvyšujúcou sa komplexnosťou systémov konštituenta. Častokrát nie je možné počet hrozieb, resp. ich dopadov dostatočne minimalizovať, najmä z operačných dôvodov chránenej organizácie. Každý z útočných vektorov si vyžaduje špecifický spôsob riešenia. [1] Avšak prvotný krok pre detekciu možných incidentov je korektné monitorovanie systémov [15]. Tu sa vyžadujú odborné znalosti ako bezpečnostných expertov, tak aj odborníkov, ktorí spravujú dané systémy v organizácii. Správne konfigurovaný monitorovací subsystém by mal obsahovať všetky potrebné údaje pre plnohodnotnú analýzu stavu systému pred incidentom, počas neho (resp. v odhadovanom období) a po incidente. Tu ale môže nastať problém v prípade, že množstvo zaznamenaných údajov je natoľko privysoké, že práca s nimi sa stáva nepraktickou. Úlohou osoby zodpovednej za riešenie incidentu (incident handlera) je po tom, čo obdrží hlásenie o podozrivej aktivite, rozhodnúť na základe zozbieraných údajov, či sa jedná o bezpečnostný incident alebo nie. Tieto hlásenia môžu prichádzať z rôznych zdrojov, či už to bude osoba (nahlasovateľ), automatizovaný systém, či IDS zariadenie. Pracovníci CSIRT musia prešetriť každé z nich s rovnakou úrovňou dôslednosti. Existuje niekoľko detekčných systémov [1]:

- IDS - *Intrusion Detection System* - je obvykle hardvérové zariadenie monitorujúce aktivitu siete. V prípade, že udalosť, alebo sled udalostí, sa zhoduje s

niektorou zo signatúr v jeho databáze, systém vyprodukuje príslušné hlásenie. IDS častokrát produkujú falošné hlásenia, ktoré však musia byť analyzované.

- SIEM - *Security Information and Event Managment* - produkty podobné IDS, avšak analýza nie je vykonávaná na komunikácii siete v reálnom čase, ale na uchovaných záznamoch.

- Antivírusový softvér deteguje rozličné formy malvéru, generuje hlásenia a aktívne zabraňuje infekcii monitorovaných zariadení.

- Monitorovacie služby tretích strán predstavujú ďalší zdroj informácií. Môže sa jednať o rôzne automatizované skenovania, sady blacklistov, či služby na ohodnotenie hrozieb.

Bollinger a kol. kladie dôraz aj na efektívne vyťažovanie metadát z prístupných datasetov. Tie majú rôzne formy od časových pečiatok, cez údaje o sieti útočníka až po lingvistickú analýzu doménových mien [15]. Zoskupovanie údajov na základe charakteristických čŕt je predpokladom efektívneho a včasného vyhodnotenia aj väčšieho objemu údajov.

Ďalším krokom analýzy incidentu je vymedzenie jeho rozsahu. NIST delí posudzovanie dopadu na tri kategórie:

- funkčný - úroveň obmedzenia štandardnej prevádzky činností organizácie,

- informačný - aký druh údajov bol exfiltrovaný, alebo zmenený a

- náklady na obnovu - množstvo časových a finančných prostriedkov pre obnovu činnosti.

Všetky tri kategórie majú 4 stupne dopadu, od žiadneho až po devastačný.

1.7.3 Zaistenie, odstránenie a obnova

Zaistenie (*containment*) je etapa, v ktorej sa minimalizuje rozsah rozšírenia dopadov incidentu. Existuje niekoľko stratégií, ich výber je závislý od typu incidentu, technických a operatívnych možností a znalostí a dostupnosti pracovníkov CSIRT.

Zaisťovanie informácií počas incidentu by malo podliehať rovnakým pravidlám ako zabezpečovanie dôkazového materiálu. V opačnom prípade by bolo možné napadnúť

prípadný súdny proces týkajúci sa najmä závažnejších incidentov. Častokrát je potrebné dbať na to, aby boli zaznamenané všetky úkony vedúce k zaisteniu údajov o incidente.

Po tom, čo zasiahnuté prvky infraštruktúry boli úspešne izolované od zvyšku organizácie, je potrebné odstrániť škodlivé komponenty. Príkladom je odstránenie malvéru, deaktivácia napadnutých používateľských účtov, či identifikácia a oprava využitých (exploitovaných) zraniteľností. Pri niektorých incidentoch sa proces odstránenia deje počas obnovy.

V etape obnovy, administrátori obnovia systém do plne funkčného stavu pred incidentom, pričom je potrebné zamedziť opakovanému útoku, a to najmä opravou zraniteľností, ktoré umožnili vznik incidentu. Túto opravu je potrebné vykonať aj na iných (nenapadnutých) komponentoch ak existuje podozrenie, že môžu byť napadnuté pomocou podobného útočného vektoru.

1.7.4 Analýza po incidente

Poslednou etapou je analýza po incidente tzv. "*Lessons Learned*". Aj napriek tomu, že sa jedná o jednu z najdôležitejších etáp, je častokrát zanedbávaná [1]. Jej úlohou je reflektovať nad uplynulými udalosťami. Slúži na analýzu a skvalitnenie procesu riadenia incidentov. Tím by počas nej mal identifikovať nedostatky v jednotlivých štádiách procesu, komunikácie a riešenia. Výsledkom by mala byť sada opatrení, ktorej aplikáciou sa predíde problémom, ktoré nastali počas riešenia incidentu. Ďalší benefit post-incident stretnutí je tvorba tréningových materiálov pre budúcich členov CSIRT. V prípadoch, že pri incidente boli využité neštandardné metódy, môžu tieto stretnutia predstavovať základ výskumu v oblasti kybernetickej bezpečnosti.

Údaje zozbierané počas incidentu môžu byť nápomocné pri riešení incidentov podobného charakteru v budúcnosti. Tieto údaje môžu byť opätovne zaradené do procesu analýzy rizík, či posudzovania efektivity CSIRT-u [1]. Takéto štatistiky sú mimoriadne užitočné pri preukazovaní potreby CSIRT-u svojím konštituentom [15].

2

Prostriedky

Riešenie bezpečnostných incidentov je komplexný proces vyžadujúci rýchlu a presnú kooperáciu väčšieho množstva organizácii a ľudí, s rôznymi stupňami znalostí a kompetencie. Pre zrýchlenie a zefektívnenie komunikácie a riadenia procesov pre riešenie bezpečnostných incidentov, mnohé CSIRT tímy využívajú podporné softvérové nástroje. Tie sú buď poskytované externým dodávateľom, alebo sú vyvíjané priamo zamestnancami CSIRT-u. Mnohé z riešení prvotne vyvinutých pre vlastné použitie sú vydané pod open-source licenciami, a teda sú použiteľné aj pre menšie tímy, ktoré nedisponujú prostriedkami a kapacitami pre vývoj vlastného riešenia.

2.1 Podobné práce

Pavel Kácha a kol. [16] predstavujú požiadavky a riešenie problému zvládania incidentu (incident handling) v organizácii CESNET. Je prispôsobené pre ich procesnú štruktúru riešenia incidentov, skladajúcu sa z troch úrovní:

- trénovaný personál,
- CESNET-CERT odborníci a
- centrum správy siete.

Tento článok sa venuje technickému riešeniu, založenému na Open-source Ticket Request System (OTRS). V článku autor vytýčil nasledujúce problémy:

- extrakcia a prehľadávanie metadát,
- kategorizácia incidentov,
- sanitácia vstupných údajov a
- kontrola životného cyklu incidentov.

Extrakcia prebieha pomocou regulárnych výrazov pre IP adresu v texte e-mailovej správy, resp. hlásenia. Následne sú extrahované IP adresy vyhľadané vo WHOIS databáze. Z databázy sú vytiahnuté len údaje o sieti a administrátorovi. Autor poukazuje na problémy spojené s nekonzistentnosťou údajov vo WHOIS databáze, ktorú riešia pomocou heuristického prehľadávania výpisu.

Automatizovanú kategorizáciu incidentov (resp. hlásení), rieši Kácha pomocou jednoduchej Bayesovej metódy. Samotná klasifikácia sa uskutočňuje pomocou programu *ifile*. Následne sa pomocou sady vlastných skriptov extrahujú výsledky. Autor a jeho tím kategorizujú incidenty podľa vlastnej taxonómie do 12 skupín: Spam, Bounce, Phishing, Pharming, Copyright, Trojan, Malware, Probe, DoS, Crack, Other a Unknown.

Pre redukciu množstva nevyžiadanej pošty, autor najprv zvažoval dve metódy: nefiltrovanie spamu, čo sa preukázalo ako neefektívne a náročné na ľudské kapacity, alebo parciálne filtrovanie za použitia DNSBL, Greylistov a Nolistingu. Avšak dospelo sa k záveru, že ani druhá metóda nie je postačujúca a množstvo spamu ostalo vysoké. Tretia predstavená (a napokon aj použitá) metóda je spojenie spamového filtra *SpamAssasin* a manuálne zostaveného whitelistu. Pre správy s falšovanými hlavičkami (*bounces*) článok neposkytuje riešenie, avšak autor uvádza, že takéto správy tvoria len zanedbateľné percento hlásení. Ďalšia metóda sanitácie vstupu spočíva v odstránení HTML a skriptov z mailových správ.

Ďalší problém, s ktorým sa stretáva CESNET-CERT, je vyššie množstvo nevyriešených tiketov. Pre vyriešenie tohto problému autor navrhol (a aplikoval) metódu pre zvýšenie priority (zviditeľnenie) ticketov, ktoré sú nečinné istý čas. Efektivita tejto metódy nie je komentovaná.

Penedo [17] popisuje jednotlivé prostriedky využívané CSIRT-mi (najmä CERT.PT). V prvej časti článku poskytuje prehľad prvkov infraštruktúry a spôsob, akým prispievajú k zefektívneniu činnosti CSIRT-u. Stručne poukazuje aj na nedostatky e-mailovej komunikácie, hlavne absenciu nenáročných spôsobov autentifikácie odosielateľa. V článku autor predstavuje spôsob segmentácie siete využívaný CERT.PT a stručne definuje jeho požiadavky. K procesom CSIRT-ov sa článok vyjadruje len implicitne.

Connell a Waits [18] sa zaoberajú zefektívnením určenia okolností, za ktorých udalosť nastala, a či udalosť je, alebo nie je incident. Hovorí o metodológii aplikovateľnej pre incidenty veľkého rozsahu t.j. je vyžadovaná spolupráca viacerých aktérov. Predstavený systém sa skladá z dvoch častí: model určujúci role a zodpovednosti a procesný model určujúci jednotlivé fázy identifikácie incidentov a vykonávanie týchto

fáz. Na tomto procesnom systéme predstavujú softvérové riešenie: *CERT Assesment Tool*. Predstavený päťkrokový popis jednotlivých etáp sa nezmieňuje o implementácii. Teda je obtiažne určiť, do akej miery sú jednotlivé procesy automatizované. Predpokladáme teda, že ide o manuálne riadený proces. Jednotlivé etapy majú veľa spoločného s odporúčaniami (best-practices) pre CSIRT tímy a len malá časť predstavuje niečo nové.

Hashemi a kol. [19] poskytujú prehľad problematiky automatizácie procesov pre riadenie incidentov. Stotožňujeme sa s ich názorom, že pri súčasnej komplexite systémov a objeme spracovávaných informácií je obtiažne spracovať všetky relevantné údaje manuálne. Takisto popisujú proces riešenia incidentov podľa NIST. Ten abstrahujú do sekcií, pre ktoré prezentujú konkrétnejšie kroky riešenia. Konkrétne softvérové systémy pre riešenie incidentov nie sú analyzované.

Tím výskumníkov z Malajzijskej Univerzity [20] predstavuje v tomto článku nový organizačný model CSIRT-ov: *Central Coordinate Contributed CSIRT*. Ten spočíva v užšom prepojení konštituentov s tímom. Okrem začlenenia už existujúcich zamestnancov priamo do jadra CSIRT-u, model pridáva nový subjekt do procesu - hodnotiacu komisiu (z angl. *steering committee*). Zdieľaním prostriedkov konštituentov sa tak rieši problém materiálneho zabezpečenia CSIRT-u, avšak novozavedená vrstva riadiacich mechanizmov, stojí za zváženie. Z článku vyplýva, že tento model používa Afgánsky národný CSIRT. Autorský tím sa v článku nevyjadril k jeho efektivite.

2.2 Porovnanie systémov pre zber údajov

Systém pre zber a harmonizáciu údajov (Message queuing system, skr. MQS) slúži na predspracovanie údajov z rôznych zdrojov ako napr. monitorovacie systémy, e-mailová komunikácia, bezpečnostné RSS kanály, a pod. Okrem predspracovania poskytujú MQS aj možnosti pre automatizovanie, anonymizáciu a obohatenie údajov (vyhľadanie DNS, GeoIP, WHOIS, ASN).

Tieto softvérové riešenia sú väčšinou neprenosné, nakoľko sú silne prispôsobené monitorovanej infraštruktúre a nastaveniam bezpečnostných procesov. Väčšie CSIRT tímy majú dostatočné kapacity pre vývoj vlastných riešení, ktoré sú častokrát publikované pod open-source licenciou pre začínajúce/menšie tímy. Spomedzi open-source MQS používané CSIRT tímami sme pre porovnanie vybrali:

- **AbuseHelper** - vyvinutý spoluprácou CERT-FI a CERT.EE [21]

- **CIF** (*Collective Intelligence Framework*) - od REN-ISAC (americký CSIRT pre .edu domény) [22]
- **ELK** (*ElasticSearch LogStash Kibana*) - kombinácia komerčných riešení pre spracovanie štrukturovaných údajov
- **IntelMQ** - nástroj pre správu incidentov, ktorý vznikol spoluprácou viacerých CSIRT týmov (*ENISA, CNCS, CERT.AT, CERT-EU, CERT.BE*) [23]
- **Megatron** - vlastné riešenie od CERT-SE [24]
- **Warden** - systém od CESNET-CSIRTs, ktoré je súčasťou ich platformy pre spracovanie incidentov
- **n6** - CERT Polska, postavený na komerčnom softvéri Splunk [25]

AbuseHelper je open-source projekt, ktorý začal v roku 2008 spoluprácou fínskeho CERT-FI, estónskeho CERT.EE a spoločnosti ClarifiedNetworks pre automatizované spracovanie notifikácií o incidentoch. Bol dostupný verejnosti v roku 2010 a odvtedy sa k vývoju pridružilo aj niekoľko iných (predovšetkým európskych) CSIRT-ov. Aj napriek tomu, že AbuseHelper je jeden z prvých systémov na predspracovanie údajov, potýka sa s mnohými problémami, ktoré neboli adresované autormi. Jednou z nich je absencia grafického rozhrania, ktorá spôsobuje neprehľadnosť toku údajov. Počas prvotných testov systému sme zistili, že v prípade výpadku jednej z častí systému sa stratili údaje aj z funkčných segmentov.

Collective Intelligence Framework (**CIF**) je open-source verzia systému SES (Security Event System) pre zber a správu údajov o hrozbách vytvoreného v americkom REN-ISAC a CSIRT. Ponúka rýchle spracovanie kanálov o bezpečnostných hrozbách, viacero typov rozhraní (grafické, webové a rozhranie príkazového riadku) ako aj knižnicu pre programovací jazyk Python. Žiaľ, CIF je už vo svojej tretej spätne nekompatibilnej verzii a dokumentácia projektu je veľmi strohá. [26]

Mnoho CSIRT-ov (predovšetkým zo súkromného sektoru) využíva k riešeniu incidentov už existujúcu monitorovaciu infraštruktúru. Jedným z "industry-standard" riešení je sada programov od spoločnosti Elastic: vyhľadávací engine **ElasticSearch**, zberač logov **LogStash** a vizualizačné rozhranie **Kibana**. Ich prepojenie je častokrát postačujúce pre potreby niektorých CSIRT-ov. Nízke náklady na zavedenie a nenáročná rozšíriteľnosť sú veľmi presvedčivé dôvody pre aplikáciu tzv. ELK stack-u.

Navyše veľa spoločností aktívnych v oblasti Big-Data už pravdepodobne disponuje odborníkmi, ktorí sú znalí týchto systémov.

IntelMQ je nástroj pre automatizovaný zber a agregáciu údajov z viacerých zdrojov. Vznikol v roku 2015 v rakúskom národnom CSIRT-e CERT-AT v rámci projektu IHAP (*incident handling automation project*). Jeho modulárna architektúra, inšpirovaná vyššie spomínaným AbuseHelper-om, poskytuje jednoduché a robustné nástroje s vysokou úrovňou prispôsobiteľnosti.

Megatron je riešenie od švédskeho národného CSIRT-u CERT-SE. Jeho vývoj sa začal v roku 2009. V súčasnosti je používaný odhadom 3 až 4 CSIRT tímami [25]. Megatron poskytuje veľmi jednoduchý, no flexibilný spôsob získavania a úpravy údajov pomocou regulárnych výrazov. Obohacovanie údajov je takisto podporované už v predvolenej konfigurácii. Bohužiaľ Megatron má len užívateľské rozhranie prístupné len z príkazového riadku.

V rámci infraštruktúry organizácie CESNET sa používa ich vlastný systém zvaný **Warden**. Ten poskytuje jednoduché, no robustné rozhranie pre spracovanie a vizualizáciu údajov. Nevýhodou je chýbajúca verejne prístupná dokumentácia, avšak CESNET-CERTS poskytuje technickú podporu pre inštaláciu a konfiguráciu. **Warden** je silne previazaný s databázou údajov Warden projektu, uloženými vo formáte IDEA (*Intrusion Detection Extensible Alert*), ktorý chce CESNET-CERTS postupne štandardizovať. [27] Vybrané časti infraštruktúry CSIRT-UPJS, taktiež prechádzajú na IDEA formát. Pre nedostatok dostupných údajov, sme Warden z tabuľky porovnania vyradili.

CERT.PL má vo vlastnej réžii systém zvaný **n6**. Komunikácia so systémom prebieha výhradne pomocou HTTP a SMTP protokolov. Okrem štandardnej funkcionality týchto systémov n6 poskytuje aj rozšírené možnosti obohacovania údajov, pri riešení incidentov, pri ktorých došlo k prevzatiu kontroly nad počítačmi (začlenenie do botnetu). Nevýhodou je chýbajúca harmonizácia údajov, nakoľko n6 zachováva pôvodný formát údajov v akom ich zdroj odoslal. [28] Navyše pre prístup k zdrojovému kódu treba kontaktovať CERT.PL

Tabuľka 2.1: Tabuľka porovnania softvérových riešení pre zber údajov

	AbuseHelper	CIF	ELK Stack	IntelMQ	Megatron	n6
Autor	CERT-FI/EE	REN-ISAC	Elastic	IHAP/CERT-AT	CERT-SE	CERT-PL
Licencia	MIT	BSD3	Apache 2.0	MIT	Apache 2.0	
Jazyk	Python	Python	Java	Python	Java	
Databáza	*SQL	Postgres	FS/NoSQL	Postgres	MySQL	FS/NoSQL
Komunikácia	XMPP/XML	Protocol Buffer	JSON	JSON	Central DB	XMPP/XML
Zdroje						
Súbory	Áno	Áno	Áno	Áno	Áno	Áno
E-mail (csv)	Áno	Áno	Áno	Áno		Áno
E-mail (ARF)	Áno	Nie	Áno (ext)	Áno		Áno
IRC	Áno	Áno	Nie	Nie		
JSON	Áno	Áno	Áno	Áno	Áno	Áno
JSON (IDEA)	Nie	Nie	Áno	Áno	Nie	
URL Stream	Áno	Áno	Áno (ext)	Áno	Áno	Áno
Generic RSS	Áno	Áno	Áno (ext)	Nie	Áno	
RequestTracker	Nie	Nie	Nie	Áno	Nie	Nie
AlienVault	Nie	Nie	Nie	Áno		
Splunk	Nie	Áno	Áno (ext)	Nie	Nie	Áno
ElasticSearch	Nie	Áno	N/A	Áno		
Výstupy						
Súbor	Áno	Áno	Áno	Áno	Áno	Áno
RestAPI	iba Wiki	Nie	Áno	Áno		
E-mail	Áno	Nie	Nie	Áno	Áno	Áno
ElasticSearch	Nie	Nie	N/A	Áno	Nie	
Obohatenie						
ASN číslo	Áno	Áno	Nie	Áno	Áno	Áno
Chýbajúca IP	Áno	Áno	Nie	Áno	Áno	Áno
DNS	Áno	Áno	Nie	Áno	Nie	Nie
GeoIP	Áno	Nie	Áno (ext)	Áno	Nie	Nie
Cymru WHOIS	Áno	Áno	Nie	Áno	Nie	Nie
Iné						
Anonymizácia	Hromadná	Hromadná	Selektívna	Hromadná	Hromadná	Hromadná
Druh spracovania	Async stream	Sync polling	Sync stream	Async stream	Batch	Async/Batch

2.3 Porovnanie tiketovacích systémov

Tiketovací systém pre správu incidentov (*z angl. incident ticket system*) je druh počítačového softvéru pre manažment incidentov v rámci organizácie (CSIRT tímu). Umožňuje tvorbu, úpravu, posudzovanie a riešenie rozličných problémov, spravidla bezpečnostných incidentov. Jednotka obsahujúca informácie o incidente a priebehu jeho riešenia sa nazýva **tiket** (*lístok*). Tiket obsahuje záznam celej, predovšetkým elektronickej komunikácie všetkých zúčastnených strán, údaje o súčasnom stave a priebehu riešenia incidentu, prípadne podporné informácie o dotknutých systémoch. Tikety sú usporiadané do **radov** (*queues*). Rad predstavuje usporiadanie tiketov podľa

určitého kritéria, napr. času kedy bol vytvorený, alebo aktív, ktorých sa týka. Používatelia (predovšetkým incident handler-i) majú prístup k radom tiketov, ktoré sú v rámci ich kompetencie. Incident handler-i si, buď vyberajú z týchto tiketov alebo ich nadriadený im niektoré pridelia. V tiketovacích systémoch je teda možné rýchlo dopátrať relevantnú osobu, zodpovednú za riešenie daného incidentu.

Tento podporný tiketovací systém sa principiálne podobá helpdeskovým systémom a/alebo systémom pre zaznamenávanie chýb softvéru (*bugtracker*). Ako bolo spomenuté, okrem jednoduchej komunikačnej a kategorizačnej funkcionality, majú tiketovacie systémy pre správu incidentov aj veľké množstvo funkcií pre efektívne zhromažďovanie a analýzu údajov. Medzi ne patrí napríklad dohľadanie údajov o zariadení s príslušnou IP adresou, kontrola prítomnosti externej IP adresy na zoznamoch adries so škodlivou činnosťou (*blacklist*), či dohľadanie kontaktných údajov relevantných osôb v databázových systémoch, alebo systémoch pre zber a harmonizáciu údajov.

V rámci našej práce sme preskúmali 3 riešenia, ktorým sa bližšie venujeme v nasledujúcich podkapitolách.

2.3.1 OTRS

Open-source Ticket Request System, (OTRS) [29] je bezplatný tiketovací systém. OTRS sa využíva na širokú škálu situácií, kde je potreba kontroly údajov o určitej entite alebo udalosti. Môže sa jednať o žiadosti o technickú podporu, poruchové hlásenia, sťažnosti, a čo je relevantné pre účely našej práce, hlásenia o počítačových bezpečnostných incidentoch. Viacero CSIRT-ov používa OTRS pre správu incidentov. Niektoré ako napr. Federálny úrad pre Informačnú bezpečnosť v Nemecku, vytvorili vlastné systémy založené na OTRS.

OTRS je vytvorený v jazyku perl s javascript-om vo webovom rozhraní. To používa vlastný jazyk pre šablóny (template-y) zvaný DTL (*Dynamic Template Language*). Systém podporuje všetky štandardné databázové systémy: MySQL, PostreSQL, Oracle, DB2 a MSSQL Server. V ekosystéme OTRS sa nachádza vyše 140 rozšírení, avšak väčšina z nich nerieši problém správy počítačových bezpečnostných incidentov.

2.3.2 RequestTracker

Request Tracker [30] je open-source program od spoločnosti Best Practical. RT je funkcionalitou veľmi podobné OTRS, avšak po preskúmaní sme identifikovali niekoľko rozdielov. Prvým je možnosť komunikácie so systémom nielen pomocou HTTP

a e-mailov, ale aj rozhrania príkazového riadku (*CLI*). To umožňuje jednoduchší spôsob automatizácie a poskytuje kontrolu nad systémom aj v prípade, že webový server bude nedostupný. Druhým rozdielom je, že RT ponúka od vezie 4.4.0 možnosť kategorizovaného uchovávania poznatkov, priamo z webového rozhrania (*articles*). Táto funkcionalita je v súlade s požiadavkami CSIRT-UPJS, nakoľko eliminuje potrebu externého "wiki"systému. Nevýhodou RT, na rozdiel od OTRS, je kompatibilita len s UNIX-ovými operačnými systémami.

2.3.3 RequestTracker pre Incident Response

Request Tracker for Incident Response (RTIR) [31] je špeciálna distribúcia RT, prispôsobená pre plnenie potrieb CSIRT/CERT tímov. Prvotná verzia bola tvorená v spolupráci s francúzskym JANET-CERT [32]. Neskôr bola rozšírená deviatimi európskymi CSIRT-mi. Oproti klasickej distribúcii RT, RTIR ponúka niekoľko vyššie spomenutých možností, ako vyhľadávanie v RIPE registri, či monitorovanie blokovaných častí spravovanej siete.

2.3.4 Zhrnutie

Porovnaním sme zistili, že rozdiely medzi nimi sú minimálne a po prispôsobení pre naše potreby, takmer zanedbateľné. Naším potrebám sa však najviac približoval RTIR, nakoľko v prípade, že by sme sa rozhodli pre iné alternatívy, museli by sme nanovo implementovať väčšiu časť jeho funkcionality. Toto rozhodnutie by bolo neefektívne a teda v ideologickom rozpore s cieľmi našej práce.

3

Koncepcia

3.1 Proces

V súčasnosti (t.j. pred zavedením nášho riešenia) je proces riadenia bezpečnostných incidentov na CSIRT-UPJS silne založený na e-mailovej a telefonickej komunikácii. Po nahlásení neobvyklej udalosti zamestnancami, študentmi alebo monitorovacími prostriedkami, niektorý z členov CSIRT-u posúdi či sa jedná o incident, a na základe subjektívneho hodnotenia mu priradí závažnosť. Ak si to situácia vyžaduje, skontaktuje aj ostatných členov CSIRT-UPJS, prípadne osobu (resp. administrátora) zodpovednú za daný úsek siete, či zariadenie. V prípade, že je nevyhnuté odpojiť napadnuté zariadenie od zvyšku infraštruktúry, "ad-hoc" incident handler sa spojí so správcami siete v Centre Informačných a Komunikačných Technológií (CIAKT). Tí zariadenie izolujú od zvyšku siete, aby sa predišlo ďalším škodám.

Náš návrh spočíva vo formalizácii a vylepšení existujúcich praktík. Po konzultáciách s členmi CSIRT-UPJS a správcami z CIAKT, sme identifikovali niekoľko častí procesu, ktoré sú buď repetitívne, a teda je možné ich do určitej miery automatizovať, alebo je možné ich zefektívniť.

CSIRT-UPJS plánuje prejsť na lepšie štrukturovaný trojstupňový organizačný model. Jeho úlohou je odbremeniť zamestnancov CIAKT od správy incidentov a priradiť tieto úlohy vyhradeným incident handlerom. Model sa skladá z troch úrovní:

1. Prvá línia - členovia prvej línie vykonávajú prvotnú analýzu okolností popísaných v hlásení. Rozhodujú o tom, či daná udalosť je incidentom, alebo nie. V prípade, že sa skutočné jedná o incident, je na ich zvážení, či sú schopní riešiť incident sami (napr. s nim už majú skúsenosti, alebo postup pre jeho riešenie je popísaný v interných dokumentoch), alebo je potrebná hlbšia analýza, resp. znalosti špecialistu. V druhom prípade incident eskalujú do druhej línie. Prvá línia je zväčša tvorená študentmi UPJŠ.

2. Druhá línia - po eskalácii incidentu do druhej línie, preberá riadenie incidentu

zamestnanec, ktorý má najvhodnejšie predpoklady pre jeho včasné vyriešenie. To nastáva najmä v prípade, že sa jedná o nový druh incidentu, resp. incidentu ktorý sa udial za neštandardných podmienok a daný zamestnanec sa napr. venuje relevantnej problematike v rámci svojej výskumnej činnosti. Na základe jeho zistení a kooperácie s prvou líniou, incident handler druhej línie vykoná príslušné opatrenia.

3. Centrum správy siete - v prípade, že sa jedná o závažný incident, pri ktorom by mohlo dôjsť k nárastu vzniknutých škôd, môže buď incident handler prvej, alebo druhej línie podať podnet na centrum správy siete (*z angl. network operations center - NOC*). Tí zablokujú patričný komponent siete (stanicu, podsieť, uzol). Centrum správy siete by malo byť kontaktované ohľadom zablokovania len v krajných prípadoch a mala by tomu predchádzať stručná diskusia medzi viacerými členmi CSIRT-u. Túto úlohu budú zastávať zamestnanci CIAKT, ktorí sú tiež členmi CSIRT-UPJS.

Medzi najčastejšie vykonávané akcie patrí vyhľadávanie zariadenia podľa IP adresy a následné vyhľadanie kontaktných informácií osoby zodpovednej za zariadenie. Tieto akcie sú v našom systéme automatizované a vykonávajú sa hneď po obdržaní hlásenia.

Ďalej sme registrovali, že častým problémom je vedenie záznamov o tom, ktorí lokálni administrátori už boli upovedomení o incidente. Nakoľko trvá aj niekoľko týždňov, kým odošlú akúkoľvek formu odpovede, je obtiažne sledovať komunikáciu s nimi. To navrhujeme riešiť časovými limitmi pri jednotlivých lístkoch. V prípade, že sa stav incidentu nezmení vyše týždňa, zodpovednej osobe je odoslaná ďalšia upomienka.

Nakoľko v poslednej dobe sme zaznamenali nárast incidentov s podvodnými mailovými správami (*phishing*), usúdili sme, že by bolo vhodné do systému zaviesť spôsob, akým získať kontaktné informácie na správcov webhostingových spoločností. Rozhodli sme sa pre jednoduché, no efektívne riešenie: WHOIS databázu. Tá obsahuje informácie o registrátoroch domén, a pretože takmer všetky phishing-ové kampane obsahujú odkazy na bezplatné zdieľané hostingy, toto riešenie je pre potreby CSIRT-UPJS postačujúce.

3.2 Riešenie

3.2.1 Predspracovanie údajov

Riešenie bezpečnostných incidentov si vyžaduje dôkladný prieskum vzniknutej situácie. Pri veľkej infraštruktúre (*relatívne k veľkosti CSIRT tímu*) môže byť obtiažne ustrážiť všetky relevantné údaje. Predspracovanie údajov bude riešené už spomínaným **systémom na predspracovanie a harmonizáciu údajov**. Najprv sme zvažovali použiť systém AbuseHelper. Hlavným dôvodom, prečo sme nakoniec od AbuseHelpera upustili, boli vysoké nároky na rozšírenie funkcionality, neprehľadná konfigurácia a chýbajúca podpora exportu údajov do tiketovacích systémov. Napokon sme zvolili systém IntelMQ, vďaka veľkému množstvu skriptov riadiacich tok a transformácie údajov tzv. *botov*. Tieto skripty sú delené do štyroch skupín [33]:

- **Zberače** (*Collectors*) - slúžia na získavanie údajov z externých služieb a monitorovacích prvkov infraštruktúry
- **Parsery** (*Parsers*) - vstupné údaje prevedú do internej reprezentácie. Zároveň slúžia ako prvý krok harmonizácie údajov
- **Expertné systémy** (*Experts*) - ich úlohou je kategorizovať, analyzovať a prípadne obohatiť vstupné údaje
- **Výstupy** (*Outputs*) - exportujú údaje mimo IntelMQ systém.

Výhodou transformácie údajov pomocou sietí botov je možnosť vyvažovania zátaže tzv. **load balancing**. Pri niektorých typoch transformácií, najmä pri analýzach a obohatení údajov, dochádza k zníženiu priepustnosti systému (počtu spracovaných správ). Vyvažovanie záťaže spočíva vo vytvorení viacerých inštancií rovnakého typu bota a deľby správ medzi nich. Navyše, siete botov sú odolné voči poruchám ako sú napr. nekorektný formát vstupných údajov, výpadok monitorovacieho prvku, či nedostupnosť externého kanálu. Porucha sa síce prejaví na absencii údajov z týchto zdrojov, ale systém funguje ďalej s prvkami, ktoré sú v prevádzke.

Systém IntelMQ je vďaka svojej architektúre jednoducho rozšíriteľný a dobre integrovateľný. V sieti UPJŠ sa nachádza niekoľko podsystémov, ktorých údaje sú vitálne pre rýchlu odpoveď na bezpečnostný incident. Jedným z takýchto systémov je W3SA, ktorý slúži na evidenciu sieťových komponentov. Priradenie IP adresy z logov

týkajúcich sa incidentu ku konkrétnemu zariadeniu a jeho správcovi, je ukážkový príklad prínosov vzájomnej integrácie. Ďalším príkladom je prepojenie IntelMQ s interným systémom **info.upjs.sk** obsahujúcim kontaktné údaje o všetkých zamestnancoch univerzity. Prepojenie výrazne urýchli riešenie incidentov týkajúcich sa e-mailov, ako napr. spam a phishing.

Napokon IntelMQ je za pomoci výstupu do REST API schopný vytvárať tikety o incidentoch v systémoch pre správu incidentov. Do novovzniknutého tiketu budú zahrnuté už extrahované, spracované a obohatené údaje, pripravené pre zamestnanca CSIRT tímu zodpovedného za riešenie incidentov.

3.2.2 Rozhranie pre Incident Handlera

Ako bolo spomenuté v úvode, údaje o incidente a priebehu jeho vyšetrovania sú centralizované v systéme pre riešenie incidentov. Zatiaľ čo agregátor údajov popísaný v predošlej sekcii bol rozhraním systému a služieb, tiketovací systém je rozhranie pre osoby podieľajúce sa na riešení incidentu.

My sme sa po dôslednom zvážení rozhodli pre open-source softvér zvaný **Request Tracker** (*RT*) od spoločnosti Best Practical Solutions, LLC. RT poskytuje služby tiketovacieho systému v dvoch verziách: klasický RT a RTIR (*Request Tracker for Incident Response*) prispôsobený pre CSIRT tímy. Verzia pre správu incidentov bola vytvorená v spolupráci s JANET CSIRT a GÉANT [32]. Rozhranie sme prispôsobili organizačnej štruktúre CSIRT-UPJS. Každej roli v procese riadenia incidentov je priradená fronta s príslušnými oprávneniami. Jedna z front je vyhradená pre automatizované hlásenia incidentov. Webové rozhranie je taktiež prispôsobené činnostiam vykonávaným počas posudzovania incidentov. Navyše sme doň integrovali funkcionalitu poskytovanú systémom pre zber údajov.

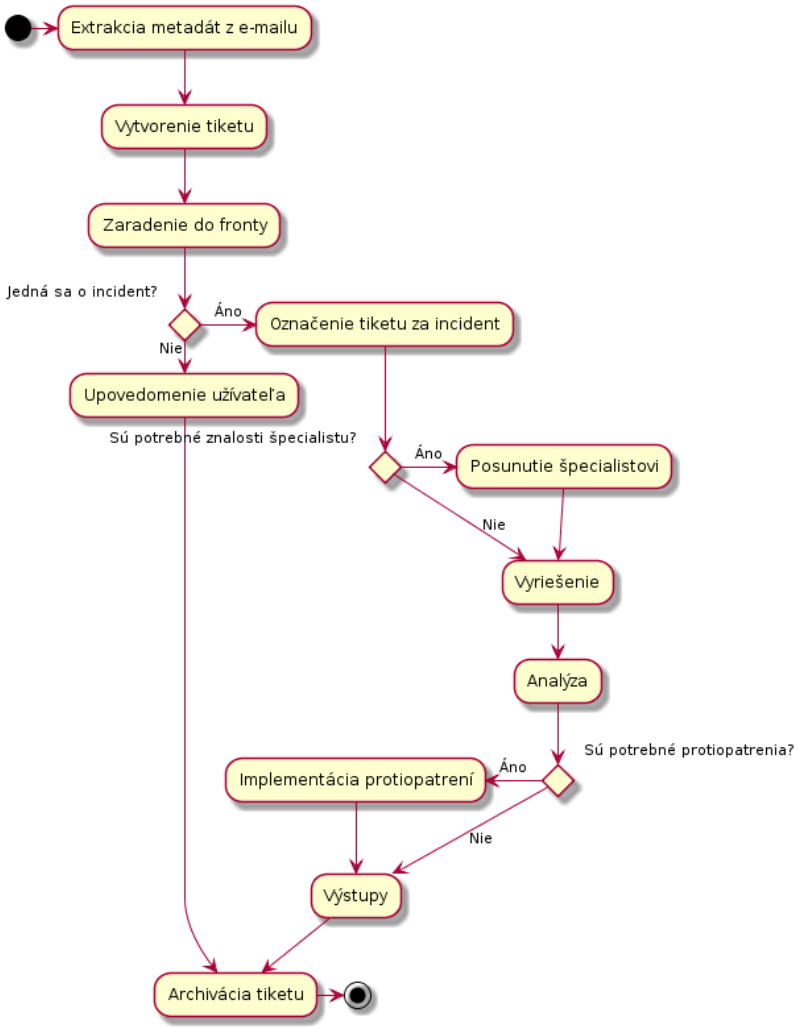
3.2.3 Webový Formulár

Webový formulár predstavuje jeden z primárnych komunikačných kanálov pre nahlasovanie bezpečnostných incidentov. Na rozdiel od elektronickej pošty, nahlasovateľ je oboznámený s údajmi, ktoré musí uviesť pre včasné vyriešenie problému. Zároveň je možné do istej miery garantovať korektnosť uvedených údajov ešte pred samotným odoslaním. Formulár, ako komunikačný kanál, je odporúčaný aj v dokumente RFC2350. Počas konzultácií s Mgr. Ladislavom Bačom dňa 3. 5. 2018, sme prišli k zisteniu, že úroveň využívania formulára ako prostriedku pre nahlasovanie incidentov

je nízka. V tomto smere, aj vzhľadom na výhody štrukturovaného hlásenia incidentu, by bolo potrebné zvýšiť povedomie konštituentov o dostupných kanáloch pre hlásenie incidentov.

3.3 Ilustračný príklad

Užívateľ organizácie obdržal nadmerné množstvo e-mailov s nežiadúcim obsahom. Odosielateľ mal e-mailovú adresu rovnakej organizácie. Postihnutý užívateľ bol zaškolený, že v takomto prípade má upovedomiť firemný CSIRT. Na vopred určenú adresu *abuse@csirt.example.org*, preposlal e-mail s nežiadúcim obsahom, spolu s krátkym popisom jeho situácie. Akonáhle mailový server obdrží jeho správu, odošle ju do tiketovacieho systému pre správu incidentov, kde sa automaticky vytvorí nový tiket a zaradí sa do fronty nevybavených lístkov. Následne osoba zodpovedná za riešenie incidentov (*incident handler*) môže postupovať napr. podľa Obr. 1.



Obr. 3.1: Ilustračný príklad riešenia bezpečnostného incidentu v organizačnej štruktúre CSIRT-UPJS

3.4 Požiadavky na riešenie

Ako sme už v predchádzajúcich sekciách spomínali, podporné softvérové prostriedky CSIRT-ov sa od seba líšia v závislosti od rozsahu, služieb a činnosti tímov. Na tieto systémy sú kladené rozličné požiadavky, vyplývajúce z procesov CSIRT-u, legislatívy krajiny, v ktorej tím operuje, zmluvnej dohody s konštituentmi, plánu kontinuity a obnovy spravovanej organizácie, či znalostí členov tímu samotného. My tieto požiadavky delíme na technické a právne.

3.4.1 Technické

Medzi technické požiadavky kladené na infraštruktúru riešenia bezpečnostných incidentov môžeme zaradiť:

- Nároky na hardvér a systémové prostriedky,
- dostupnosť,
- integrita údajov,
- dôvernosť.

Minimálne testované hardvérové požiadavky pre naše riešenie sú:

- Operačný systém Linux (testované na Ubuntu Server 17.04),
- procesor s architektúrou x86, resp. x64,
- aspoň 1GB pamäte,
- pripojenie do siete internet.

Sme toho názoru, že infraštruktúra CSIRT-u obsahujúca podporné nástroje by mala byť sama o sebe bezpečná t.j. aby počet incidentov na tejto infraštruktúre bol minimálny. Nakoľko softvérové nástroje sú priamo v réžii CSIRT-u, nič nebráni jeho členom ich dôsledne zabezpečiť. Existuje veľké množstvo postupov a odporúčaní pre bezpečnostný hardening (*spevnenie*), ktorého popis a metódy však spadajú mimo rozsah našej práce.

Ďalšou požiadavkou je stabilita. Výpadok interných služieb tímu počas bezpečnostného incidentu, by mohol ohroziť nielen priebeh vyšetrovania incidentu, ale aj dôveru konštituentov v kompetentnosť členov CSIRT. Vyvolanie akýchkoľvek pochybností považujeme za zlyhanie celého tímu. Každopádne tímy musia počítať aj s prácou v obmedzených podmienkach. Ako to už z ich názvu vyplýva, sú to **podporné** nástroje, slúžiace na rýchlejšie vyriešenie incidentu. Preto tímy by sa mali zamerať na procesy, nie na nástroje.

Bezpečnosť a stabilitu zabezpečíme aj tým, že naše riešenie je stavané modulárnym spôsobom, ktorý umožňuje menej náročnú aktualizáciu na novšiu verziu. Navyše predstavujeme pár skriptov pre rýchlu aktualizáciu/nasadenie oboch komponentov riešenia (viď. Implementácia). Použitie aktuálnych verzií softvéru bolo jednou z požiadaviek od CSIRT-UPJS.

Integritu údajov vrámci riešenia poskytujeme vo forme konfigurácie. Tá je vyhotovená s ohľadom na odporúčania vydané Slovenským vládnym CSIRT-om [34]. Podstatná časť zabezpečenia integrity, závisí od koncových používateľov riešenia. Ako autori riešenia, máme dôveru v personál CSIRT-UPJS a v ich prístup k zabezpečeniu internej siete.

Dôvernosť údajov je zabezpečená vo forme regulácie prístupu osôb k údajom na niekoľkých úrovniach. Prístup do internej siete CSIRT-u majú len členovia CSIRT, prípadne iný zamestnanci univerzity. Priamy prístup z internetu, nie je možný. Tiketovací systém má prístup k údajom riešený pomocou skupín používateľov s rozličnými oprávneniami. Do konfiguračnej časti systému pre zber údajov má prístup len jeho správca.

3.4.2 Právne požiadavky

Právne požiadavky vyplývajú z niekoľkých všeobecne záväzných právnych predpisov. Jedná sa najmä o tie predpisy, ktoré stanovujú pre organizáciu nutnosť riešenie bezpečnostných incidentov. Príkladmi týchto predpisov sú:

- nariadenie Európskeho parlamentu a Rady EÚ 2016/679 o ochrane fyzických osôb pri spracúvaní osobných údajov a o voľnom pohybe takýchto údajov, tiež známe pod skratkou GDPR (*General Data Protection Regulation*) [35],
- smernica Európskeho parlamentu a Rady EÚ 2016/1148 o opatreniach na zabezpečenie vysokej spoločnej úrovne bezpečnosti sieti a informačných systémov v Únii [36],

- Zákon č. 122/2013 Z. z. o ochrane osobných údajov a o zmene a doplnení niektorých zákonov [37], ktorý bude od 25. mája 2018 nahradený zákonom č. 18/2018 Z. z. [38],
- Zákon č. 69/2018 Z. z. o kybernetickej bezpečnosti a o zmene a doplnení niektorých zákonov [39], ktorý bude od 25. mája 2018 pozmenený [38].

4

Implementácia

4.1 Inštalácia a konfigurácia

Korektná inštalácia systémov IntelMQ a Reuqest Tracker je komplexný proces, nakoľko je nutné kompilovať obe aplikácie z ich zdrojového kódu. Aj keď existujú už existujúce balíčky pre Reuqest Tracker, tie sú zastaralé a aj počas tvorby nášho riešenia nastala situácia, kedy PPA repozitáre smerovali na nefunkčný odkaz. Preto sme sa rozhodli implementovať skripty pre jednoduchú a plne automatizovanú inštaláciu ako systému RT, tak aj systému IntelMQ. Tieto skripty sme využívali spolu s virtualizačnou technológiou, ktorú spravujeme pomocou programu Vagrant [40], ktorý slúži na správu virtuálnych strojov. Výhodou vývoja vo virtualizovanom prostredí je možnosť jeho prispôsobenia tak, aby sa čo najviac podobal skutočnému prostrediu, v ktorom budú systémy pracovať. Tým sme predišli veľkému množstvu prípadných chýb pri nasadzovaní systémov. Naše skripty zjednodušujú proces inštalácie na použitie jediného príkazu `vagrant up`, ktorý vykoná nasledujúce kroky:

1. Stiahnutie aktuálneho obrazu disku, obsahujúceho operačný systém
2. Naštartovanie virtuálneho stroja a nastavenie sieťových rozhraní
3. Spustenie nášho inštalačného skriptu
4. Inštalácia povinných a/alebo voliteľných závislostí
5. Konfigurácia databázy a webového serveru
6. Prevzatie zdrojového kódu aplikácie
7. Kompilácia programu
8. Konfigurácia programu na nastavené parametre
9. Nastavenie zabezpečenia aplikácie

10. Voliteľná inštalácia modulov a nástrojov pre vývoj

Priebeh všetkých krokov je podrobne zaznamenaný a v prípade zlyhania niektorého z nich je možné inštaláciu spustiť z bodu, kde bola prerušená. Najprv sme zvažovali použitie štandardného nástroja `make`, ale ten nie je zahrnutý v predvolenej inštalácii väčšiny Linuxových distribúcií.

```
...
while true; do
  if [[ $# -ne 0 ]]; then
    case "$1" in
      install-dependencies|ideps) install-dependencies; shift ;;
      install-mysql|imysql) install-mysql; shift ;;
      install-apache|iapache) install-apache; shift ;;
      install-rt|irt) install-rt; shift ;;
      install-rtir|irtir) install-rtir; shift ;;
      post-install|post) post-install; shift ;;
      ssl-setup|ssl) ssl-setup; shift ;;
      apache-configure|capache|apachec) apache-configure; shift ;;
      rt-configure|crt|rtc) rt-configure; shift ;;
      development|dev) development; shift ;;
      all) all; break ;;
      *) all; break ;;
    esac
  else
    message "All done!"
    break
  fi
done
```

Ukážka kódu 1: Časť kódu zabezpečujúca výber inštalačných krokov

Zvoleným rozdelením inštalačného procesu môže administrátor interaktívne spustiť len určité segmenty v špecifikovanom poradí napr.:

`./deploy.sh ideps install-rt rt-configure post`. To je výhodné v prípade, že na serveri už je nainštalovaná databáza alebo webový server. Aby sme docielili plne automatizovaný inštalačný proces, museli sme pri niektorých príkazoch emulovať interakciu užívateľa.

```
...
function install-mysql {
    message "Installing MySQL..."

    echo "mysql-server mysql-server/root_password password $password"\
     | debconf-set-selections
    echo "mysql-server mysql-server/root_password_again password $password"\
     | debconf-set-selections
    apt-get -y install mysql-server mysql-client libmysqlclient-dev

    message "Configuring MySQL..."
    echo -e "$username\n$password\nn\nY\nY\nY\nY\n"\
     | mysql_secure_installation
    systemctl restart mysql

    message "MySQL installed."
}
...
```

Ukážka kódu 2: Časť kódu s emuláciou užívateľského vstupu pre interaktívne programy

Kompilácia (najmä systému RT) je najdlhšou časťou procesu. Počas našich testov trvala na minimálnej požadovanej hardvérovej konfigurácii od 15 do 25 minút. Tu sa vyskytol problém, že pri deaktivovanom detailnom výpise (*verbose*), administrátorovi chýbala spätná odozva o stave kompilácie. Na druhej strane, detailný výpis bol veľmi neprehľadný. Preto sme pridali výstup prijateľnejší pre užívateľa, ktorý daný problém rieši. Do logov sa vždy zaznamenáva detailný výstup.

```
...
message "Installing RT dependencies (this may take a while)..."
cpanm --notest DBD::SQLite
__counter=0
set +e
cpanm --notest $(make testdeps | grep MISSING | grep -v SOME\
                  | cut -d' ' -f1 | awk '$1=$1' | tr '\n' ' ')
  | while read l; do
    if [[ $l = *"Successfully installed"* ]]; then
      __counter=$((__counter + 1))
      message "[$__counter/$__total] $l"
    fi
done
set -e
...
```

Ukážka kódu 3: Časť kódu zodpovedná za korektný výpis stavu kompilácie

4.2 Webový Formulár

Formulár je vyhotovený v dvoch funkčných verziách: pre užívateľov a pre správcov systémov, a každá z nich v dvoch jazykových verziách: slovenčine a angličtine. Verzia pre správcov je obohatená o možnosť uviesť podrobnejšie technické detaily spojené s incidentom. Jednou z požiadaviek kladených na tento formulár je korektná funkcionalita aj v prehliadačoch nepodporujúcich technológiu javascript. Tú sme prešetrili aj pomocou textových prehliadačov. Formulár je zabudovaný priamo do webovej stránky CSIRT-UPJS, ktorej sme takisto autormi. Formulár sa odošle na mailový server, kde je preň vytvorený tiket a je zaradený do fronty tiketov, tak ako akékoľvek iné hlásenie. Najprv sme zvažovali priame vkladanie do RequestTracker-u cez jeho rozhranie, no zavedenie ďalšieho vstupného kanálu, by si vyžadovalo dodatočnú konfiguráciu, ktorá by bola totožná s tou pre extrakciu údajov z e-mailových správ.

Obr. 4.2: Snímka webového formulára, užívateľská verzia, slovenský jazyk

4.3 Request Tracker

Request Tracker sme prispôsobili trojstupňovému procesu riadenia incidentov, popísaného v kapitole 3.1. Vytvorili sme niekoľko radov pre tikety:

- Všeobecné správy a oznámenia - sem idú tikety týkajúce sa vnútorných záležitostí CSIRT-u, je to forma interného mailing-listu
- Hlásenia incidentov - obsahuje hlásenia z e-mailových schránok a webového formulára, jedná sa o udalosti, pri ktorých ešte nebolo rozhodnuté či ide o incident
- Incidenty - tikety o udalostiach, ktoré spĺňajú definíciu bezpečnostného incidentu
- Vyšetrovania - tikety závažnejších incidentov, ktoré si vyžadujú hlbšiu analýzu
- Protiopatrenia - incidenty, pri ktorých bolo potrebné dočasne zablokovať časť infraštruktúry

K týmto radom sme priradili skupiny užívateľov s rôznymi oprávneniami. Všetky skupiny majú prístup k radu s oznámeniami s výnimkou lokálnych administrátorov.

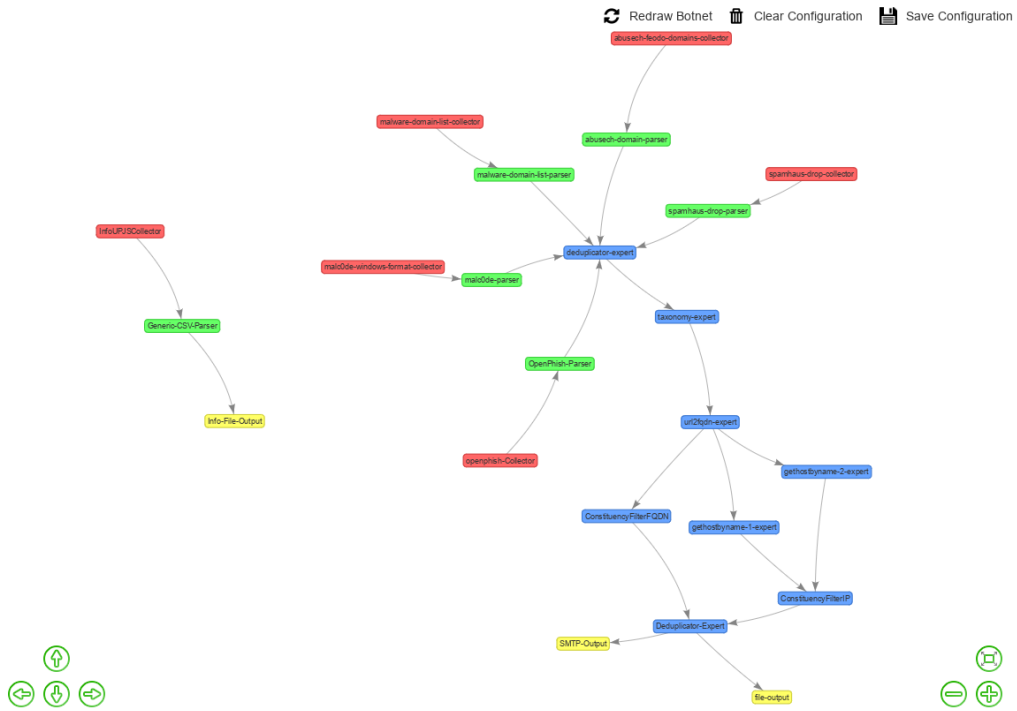
- Incident handleri I. línie - majú prístup k hláseniam a fronte incidentov, môžu v nich vytvárať, spájať a deliť tikety
- Incident handleri II. línie - majú rovnaký prístup, ako členovia prvého radu. Navyše môžu pristupovať aj k radu incidentov vyžadujúcich si odbornejšie preskúmanie, môžu prideľovať tikety incident handlerom I. línie a prezerať rad s protiopatreniami
- Administrátori CIAKT - nemajú prístup k radom o incidentoch a analýzach, avšak môže im byť pridelený na úrovni tiketu. Vo fronte s protiopatreniami vedú záznamy o blokovaných častiach siete.
- Lokálni administrátori - nemajú prístup ani k jednému z radov, ale môžu podávať hlásenia. Nimi nahlásené incidenty sú im prístupné.

Z prichádzajúcich hlásení sú extrahované relevantné údaje. Pre získané IP adresy sa vykoná ich vyhľadanie buď pomocou WHOIS, alebo W3SA (viď. integrácie).

4.4 IntelMQ

Boty pre zber a parsovanie údajov, sú nakonfigurované tak, aby zbierali údaje z rôznych blacklistov, predovšetkým pre spam. Následne expertné boty dodatočne spracujú prichádzajúce správy a tie, ktoré sa týkajú konštituencie CSIRT-UPJS sú odoslané mailom do tiketovacieho systému. Boty filtrujú na základe regulárnych výrazov pre doménové mená, alebo IP adresy v spravovanom rozsahu. Preklad z url na IP adresu bol počas našich testov najpomalší článok spracovania a teda sme pridali ešte jedného bota pre DNS vyhľadávanie.

Ako dodatočný zdroj údajov sme pridali aj integráciu s portálom info.upjs.sk. IntelMQ pravidelne stiahne novú verziu kontaktov, spracuje ju a uloží na disk. Odtiaľ je využívaná inými programami.



Obr. 4.3: Mapa celého botnetu v IntelMQ

4.5 Integrácie

Podstatná časť prispôsobenia systémov pre potreby CSIRT-UPJS je integrácia RT a IntelMQ s už existujúcimi systémami, ktoré sú na univerzite už zavedené. Ako bolo spomínané, prepojenie informačných systémov a automatické obohatenie údajov je prvým krokom k zefektívneniu poskytovaných služieb a redukcii času potrebného pre vyriešenie incidentu [9]. Niektoré z nich sú riešené ako modifikácie IntelMQ, resp. Request Tracker-u, iné ako malý subsystém.

4.5.1 WHOIS

Request Tracker je nakonfigurovaný, aby po obdržaní hlásenia e-mailom, resp. webovým formulárom extrahoval všetky IP adresy, rozsahy IP adries a doménové mená z hlásenia. Následne sú buď vyhľadané v systéme W3SA (pozri nižšie), alebo vo WHOIS databáze. Nemá význam vykonávať WHOIS výpis na IP adresách z rozsahu konštituentov CSIRT-UPJS, nakoľko v RIPE databáze sa nachádza kontakt buď na CSIRT-UPJS, alebo na CIAKT. V prípade, že sa jedná o IP adresu mimo chráneného rozsahu, vygenerujú sa dva výpisy:

- plný výpis - obsahuje všetky údaje zapísané v databáze a

- skrátený výpis - obsahuje len relevantné kontaktné údaje.

Krátky výpis už poskytuje incident handler-ovi pripravené informácie, ktoré bude potrebovať, avšak z dôvodu nekonzistentnosti databázy [16], poskytujeme aj plný výpis. Navyše Reuqest Tracker umožňuje zobraziť všetky zaznamenané incidenty, v ktorých figurovala daná IP adresa. Prípadne je možné spraviť výpis manuálne.

```
NetName:        RIPE-ERX-158-190-0-0
Organization:   RIPE Network Coordination Centre (RIPE)
OrgName:        RIPE Network Coordination Centre
OrgTechName:   RIPE NCC Operations
OrgTechPhone:  +31 20 535 4444
OrgTechEmail:  hostmaster@ripe.net
OrgAbuseName:   Abuse Contact
OrgAbusePhone:  +31205354444
OrgAbuseEmail:  abuse@ripe.net
netname:        UPJSNET
phone:          +421 55 234 1510
phone:          +421 55 2341515
```

Ukážka kódu 4: Ukážkový príklad skráteného WHOIS výpisu pre IP adresu 158.197.32.105

4.5.2 W3SA

W3SA je názov označujúci interný systém UPJŠ, slúžiaci pre správu sieťových zariadení. Sú v ňom zaznamenané [41]:

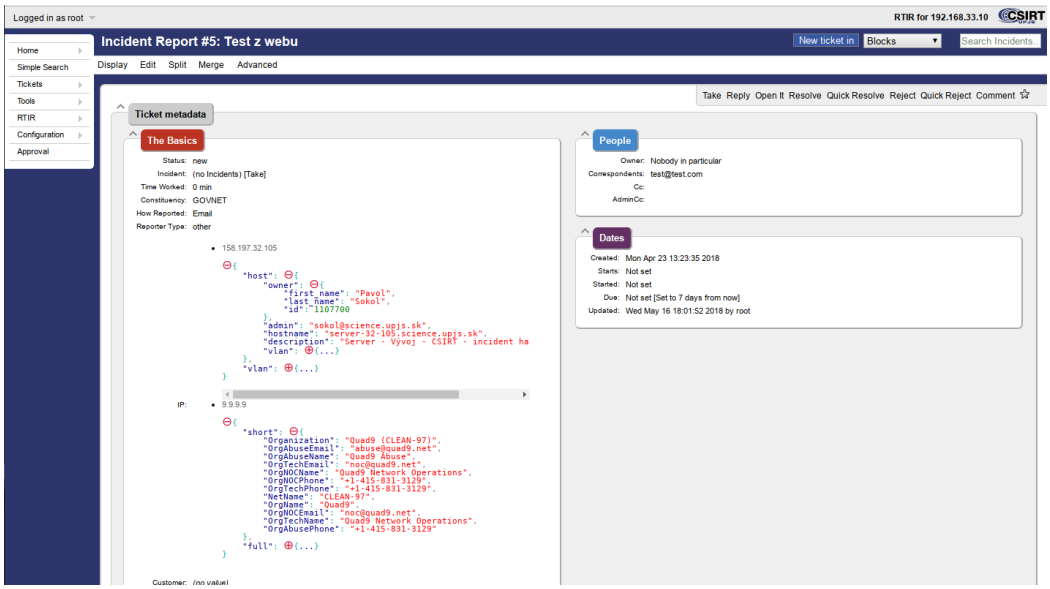
- užívateľské kontá a emailové adresy
- zariadenia (PC, VoIP, smerovače, prepínače,...)
- virtuálne siete
- kabeláž
- SSL certifikáty

Incident handleri ho využívajú pre zistenie osoby zodpovednej za zariadenie s danou IP adresou. Takmer všetky koncové zariadenia v sieti UPJŠ majú pridelenú samostatnú verejnú IP adresu.

```
{
  "host": {
    "owner": {
      "first_name": "Pavol",
      "last_name": "Sokol",
      "id": 1107700
    },
    "admin": "sokol@science.upjs.sk",
    "hostname": "server-32-105.science.upjs.sk",
    "description": "Server - Vývoj - CSIRT - incident handling (Pavúk)",
    "vlan": {
      "group": "pf",
      "id": 32,
      "name": "PF_Hnet",
      "description": "Výskumná sieť bezpečnosti PF..."
    }
  },
  "vlan": {
    "netmask": 24,
    "group": "pf",
    "name": "PF_Hnet",
    "description": "Výskumná sieť bezpečnosti PF...",
    "gateway": "158.197.032.001",
    "id": 32
  }
}
```


Ukážka kódu 5: Ukážkový výstup zo systému W3SA

Pôvodne sme zamýšľali integrovať vyhľadávanie IP adries vo W3SA priamo do RequestTracker-u, avšak po jeho aktualizácii na novšiu verziu (z 4.0.22 na 4.4.2), sme zistili, že niektoré zmeny týkajúce sa webového rozhrania, neboli spätne kompatibilné. Aby sme v budúcnosti predišli podobným problémom, rozhodli sme sa extrahovať túto funkcionalitu do samostatnej webovej aplikácie, ktorej výstup sa integruje do rozhrania RT. Tá je napísaná v Python-e a využíva mikro-framework Flask. Aplikácia beží na webovom serveri Apache2, ktorý je využívaný aj RequestTrackerom, no zatiaľ čo ten používa mod_{perl}, naša brána do W3SA je spúšťaná ako WSGI (Web Server Gateway Interface) skript. Údaje prevzaté z W3SA sú dočasne uložené v SQLite databáze (cache), pre ďalšie požiadavky na danú adresu. Tak znižujeme zátaž na hlavný server.



Obr. 4.4: Snímka obrazovky s interaktívnym výpisom z W3SA v RT

Naše prepojenie do W3SA pred spustením skontroluje či štruktúra databázy je v poriadku. Touto kontrolou predchádzame situácii, kedy by mohlo dôjsť k narušeniu integrity údajov. Pred kontaktovaním W3SA sa skontroluje či sa daná IP adresa už nenachádza v pomocnej databáze, ak áno vrátime výsledok z lokálneho úložiska, v opačnom prípade sa program autorizuje na vzdialenom serveri, stiahne požadované údaje a uloží ich do svojej databázy.

```
def w3sa_lookup(ip_addr):
    lookup_data = ""
    with sqlite3.connect(app.config['DATABASE']) as conn:
        cur = conn.cursor()
        cur.execute('SELECT data FROM cache WHERE ip = ?', [ip_addr])
        rows = cur.fetchall()
        if len(rows) == 0:
            result = fetch(ip_addr)
            if result[0] == 200:
                lookup_data = result[1]
                cur.execute("INSERT INTO cache (ip, date, data)" +
                            " values (?, datetime('now'), ?)"
                            ,[ip_addr, lookup_data])
        else:
            lookup_data = rows[0][0]
    return lookup_data
```

Ukážka kódu 6: Funkcia vykonávajúca vyhľadanie vo W3SA

Avšak vyhľadávanie v univerzitnej sieti sa deje len v prípade, že daná IP adresa sa nachádza na zozname IP adries spadajúcich pod ochranu CSIRT-UPJS. Ak adresa je mimo pôsobnosti CSIRT-UPJS vykoná sa dopyt pomocou WHOIS (viď. predošlá sekcia). Kontrola prebieha nasledovne:

```
def in_constituency(ip_addr):
    addr = int(ipaddress.ip_address(ip_addr))

    for iprange in app.config['CONSTITUENCY']:
        n = ipaddress.ip_network(iprange)
        netw = int(n.network_address)
        mask = int(n.netmask)
        if ((addr & mask) == netw):
            return True
    return False
```

Ukážka kódu 7: Funkcia overujúca, či IP adresa je z určených rozsahov

Záver

Problematika kybernetickej bezpečnosti je úzko spojená s vývojom spoločnosti, technologickým pokrokom, ako aj s neustálou snahou napádať počítačové systémy organizácii, či už zámerne, alebo vplyvom iných okolností. Požiadavky na CSIRT tímy jednotlivých organizácií, ako aj nové druhy bezpečnostných incidentov, s ktorými sa CSIRT tímy budú stretávať, so sebou určite prinesú nové výzvy.

Prvým cieľom našej práce bolo definovať požiadavky CSIRT/CERT tímov v oblasti spracovania bezpečnostných incidentov. V prvej kapitole práce sme uviedli porovnanie taxonómií bezpečnostných incidentov. Zadefinovali sme činnosti CSIRT/CERT tímov a požiadavky na nich kladené. Taktiež porovnávame modely riešenia incidentov, ktoré určujú očakávania od CSIRT/CERT tímov.

Druhému cieľu práce, porovnaniu aktuálnych prístupov k spracovaniu bezpečnostných incidentov, sa venujeme v druhej kapitole. V nej analyzujeme aktuálne metódy riadenia incidentov a softvérové riešenia pre ich spracovanie. Porovnali sme desať softvérových riešení a posúdili sme ich aplikovateľnosť v rámci potrieb CSIRT-UPJS.

Napokon hlavným cieľom práce bolo navrhnúť a implementovať systém na riešenie bezpečnostných incidentov pre CSIRT-UPJS. Splnenie tohto cieľu popisujeme v tretej a štvrtej kapitole práce. V kapitole "Koncepcia"poskytujeme prehľad procesu riešenia incidentov v prostredí CSIRT-UPJS. Následne sme vyhotovili koncepciu softvérového riešenia pre efektívnu správu incidentov a preukázali sme, akým spôsobom spĺňa technické požiadavky naň kladené. V poslednej kapitole uvádzame implementačné detaily riešenia. Celý proces inštalácie a prvotnej konfigurácie navrhovaných systémov sme automatizovali. Nakonfigurovali sme zvolené softvérové riešenia pre potreby CSIRT-UPJS, a prepojili sme ich so systémami, ktoré sú už zavedené na UPJŠ.

Nami navrhnutý systém a implementácia poskytujú, podľa nášho názoru, solídny základ pre efektívny proces riadenia bezpečnostných incidentov a činnosť tímu CSIRT-UPJS. Veríme, že tento systém bude prínosom a bude využívaný pre ďalší rozvoj a aktualizácie, podľa vývoja bezpečnostnej situácie, na zabezpečenie kybernetickej bezpečnosti siete UPJŠ.

Prílohy

A

DVD médium

Obsah DVD média:

```
+-- IntelMQ/
|   |-- configurations/ - konfiguračné súbory pre IntelMQ
|   |   |-- defaults.conf - defaultné nastavenia pre nové boty
|   |   |-- harmonization.conf - zoznam harmonizovaných údajov
|   |   |-- pipeline.conf - prepojenia medzi botmi
|   |   `-- runtime.conf - boti a ich nastavenia
|   |-- guest_deploy.sh - inštalačný skript pre IntelMQ
|   |-- users.txt - zoznam kontaktov z info.upjs.sk
|   `-- Vagrantfile - konfigurácia pre testovaciu VM
|-- RequestTracker/
|   |-- deploy.sh - inštalačný skript pre RequestTracker
|   |-- extensions/
|   |   `-- RT-Extension-W3SA/ - modul pre integráciu W3SAGateway
|   |-- reload_extensions.sh - pomocný skript pre zavedenie modulov
|   |-- setup_env.sh - pomocný skript pre prácu s RT CLI
|   |-- Vagrantfile - konfigurácia pre testovaciu VM
|   `-- whois-bot.sh - skript pre automatizovaný WHOIS výpis
`-- W3SAGateway/
    |-- W3SAGateway/ - súbory pre integráciu W3SA do RT
    |   |-- __init__.py - hlavný súbor modulu
    |   |-- static/ - pomocné javascript súbory pre interaktivitu
    |   |-- templates/ - HTML šablóna pre výpis
    |   |-- venv3/ - virtuálne prostredie pre python
    |   `-- whois-bot.sh - skript pre automatizovaný WHOIS výpis
    `-- w3sa-gateway.wsgi - integrácia W3SAGateway do Apache2
```

Zoznam použitej literatúry

[1] Cichonski, Millar, Grance, and Scarfone, *Computer security incident handling guide*, 2012, vol. 800, no. 61.

[2] ENISA. (2018) About ENISA — ENISA. [prevzaté 16. marca 2018]. [Online] Dostupné z: https://www.enisa.europa.eu/about-enisa

[3] ENISA. (2018) Reference incident classification taxonomy. [prevzaté 16. marca 2018]. [Online] Dostupné z: https://www.enisa.europa.eu/publications/reference-incident-classification-taxonomy/at_download/fullReport

[4] Europol. (2018) About europol | europol. [prevzaté 17. marca 2018]. [Online] Dostupné z: https://www.europol.europa.eu/about-europol

[5] Europol. (2018) Reference incident classification taxonomy. [prevzaté 17. marca 2018]. [Online] Dostupné z: https://www.europol.europa.eu/sites/default/files/documents/common_taxonomy_for_law_enforcement_and_csirts_v1.3.pdf

[6] (2004) eCSIRT.net. [prevzaté 11. mája 2018]. [Online] Dostupné z: http://www.ecsirt.net/cec/index.html

[7] Brownlee and Guttman, "RFC2350: Expectations for computer security incident response," *Internet RFCs*, 1998.

[8] West-Brown, Stikvoort, Kossakowski, Killcrece, Ruefle, and Zajicek, "Handbook for computer security incident response teams (CSIRTs)," Carnegie Mellon University, Software Engineering Institute, Tech. Rep., 2003, [prevzaté 7. apríla 2018]. [Online] Dostupné z: https://resources.sei.cmu.edu/asset_files/Handbook/2003_002_001_14102.pdf

[9] Maj, Reijers, and Stikvoort, *Good practice guide for incident management.* European Network and Information Security Agency (ENISA), 2010.

[10] (2003) Mission statement. [prevzaté 21. apríla 2018]. [Online] Dostupné z: https://www.first.org/about/mission

[11] (2018) FIRST teams. [prevzaté 16. mája 2018]. [Online] Dostupné z: https://www.first.org/members/teams/

[12] (2017) Becoming a member. [prevzaté 16. mája 2018]. [Online] Dostupné z: https://www.first.org/membership/

[13] (2015) TF-CSIRT - home for the computer security incident response teams. [prevzaté 16. mája 2018]. [Online] Dostupné z: https://tf-csirt.org/

[14] Mitropoulos, Patsos, and Douligeris, "On incident handling and response: A state-of-the-art approach," *Computers & Security*, vol. 25, no. 5, pp. 351–370, 2006.

[15] Bollinger, Enright, and Valites, *Crafting the InfoSec Playbook: Security Monitoring and Incident Response Master Plan.* ÖŔeilly Media, Inc.", 2015.

[16] Kácha, "Adapting the ticket request system to the needs of CSIRT teams," *WSEAS Transactions on Computers*, vol. 8, no. 9, pp. 1440–1450, 2009, [prevzaté 18. apríla 2018]. [Online] Dostupné z: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.551.7896&rep=rep1&type=pdf

[17] Penedo, "Technical infrastructure of a CSIRT," in *Internet Surveillance and Protection, 2006. ICISP'06. International Conference on.* IEEE, 2006, pp. 27–27.

[18] Connell and Waits, "The CERT assessment tool: Increasing a security incident responder's ability to assess risk," in *Technologies for Homeland Security (HST), 2013 IEEE International Conference on.* IEEE, 2013, pp. 236–240.

[19] Hashemi, Babaeizadeh, Nowruzi, Jazi, Shahmoradi, and Samani, "A comprehensive semi-automated incident handling workflow," in *Telecommunications (IST), 2012 Sixth International Symposium on.* IEEE, 2012, pp. 1065–1070.

[20] Jalal, Shukur, and Mokhtar, "3C-CSIRT model a sustainable national CSIRT for afghanistan," in *Electrical Engineering and Informatics (ICEEI), 2017 6th International Conference on.* IEEE, 2017, pp. 1–4.

[21] Van Heurck and Durveaux, "Abusehelper, call for an open discussion." Prezentované na 31. stretnutí TF-CSIRT v Instanbule, 16. - 17. September 2010, 2010.

[22] Pearson, "Research and education networking information sharing and analysis center," 2015, [prevzaté 20. januára 2018]. [Online] Dostupné z: https://www.bu.edu/tech/files/2015/08/pearson.pdf

[23] Network and ENISA. (2015) Incident handling automation — ENISA. [prevzaté 20. januára 2018]. [Online] Dostupné z: https://www.enisa.europa.eu/topics/csirt-cert-services/community-projects/incident-handling-automation

[24] Johnson and Pestana. (2011) Megatron. [Online] Dostupné z: https://www.cert.se/megatron/megatron-telia2011.pdf

[25] Kijewski and Pawliński, "Proactive detection and automated exchange of network security incidents," CERT Polska, Tech. Rep., 2012. [Online] Dostupné z: https://www.cert.pl/wp-content/uploads/2015/12/MP-IST-111-18.pdf

[26] (2018) csirtgadgets/bearded-avenger: CIF V3 – the fastest way to consume threat intelligence. [prevzaté 18. apríla 2018]. [Online] Dostupné z: https://github.com/csirtgadgets/bearded-avenger

[27] (2018) The project [warden]. [prevzaté 15. apríla 2018]. [Online] Dostupné z: https://warden.cesnet.cz/en/about_project

[28] (2018) n6 - network security incident exchange. [prevzaté 15. apríla 2018]. [Online] Dostupné z: https://n6.cert.pl/

[29] Official site. [prevzaté 16. apríla 2018]. [Online] Dostupné z: https://otrs.com/

[30] (2018) RT - best practical solutions. [prevzaté 16. apríla 2018]. [Online] Dostupné z: https://bestpractical.com/request-tracker/

[31] (2018) RT for incident response - best practical solutions. [prevzaté 16. apríla 2018]. [Online] Dostupné z: https://bestpractical.com/rtir/

[32] JANET CSIRT, "RTIR incident handling work-flow," Tech. Rep., [prevzaté 21. januára 2018]. [Online] Dostupné z: https://bestpractical.com/s/janet-workflow.pdf

[33] (2018) Bots - intelmq. [prevzaté 12. apríla 2018]. [Online] Dostupné z: https://intelmq.readthedocs.io/en/latest/Bots/

[34] (2013) Linux hardening v1. [prevzaté 17. mája 2018]. [Online] Dostupné z: https://www.csirt.gov.sk/doc/Hardened_v1.pdf

[35] Európsky parlament a Rada Európskej únie, "Nariadenie európskeho parlamentu a rady EÚ 2016/679 o ochrane fyzických osôb pri spracúvaní osobných údajov a o voľnom pohybe takýchto údajov, ktorým sa zrušuje smernica 95/46/es (všeobecné nariadenie o ochrane údajov)," 2016, [prevzaté 17. mája 2018]. [Online] Dostupné z: http://eur-lex.europa.eu/legal-content/SK/TXT/HTML/?uri=CELEX:32016R0679&from=EN

[36] Európsky parlament a Rada Európskej únie, "Smernica európskeho parlamentu a rady EÚ 2016/679 o opatreniach na zabezpečenie vysokej spoločnej úrovne bezpečnosti sietí a informačných systémov v Únii," 2016, [prevzaté 17. mája 2018]. [Online] Dostupné z: http://eur-lex.europa.eu/legal-content/SK/TXT/HTML/?uri=CELEX:32016L1148&from=EN

[37] Národná rada Slovenskej republiky, "Zákon č. 122/2013 z. z. o ochrane osobných údajov a o zmene a doplnení niektorých zákonov," 2013, [prevzaté 17. mája 2018]. [Online] Dostupné z: https://www.slov-lex.sk/static/pdf/2013/122/ZZ_2013_122_20140415.pdf

[38] Národná rada Slovenskej republiky, "Zákon č. 18/2018 z. z. o ochrane osobných údajov a o zmene a doplnení niektorých zákonov," 2018, [prevzaté 17. mája 2018]. [Online] Dostupné z: https://www.slov-lex.sk/static/pdf/2018/18/ZZ_2018_18_20180525.pdf

[39] Národná rada Slovenskej republiky, "Zákon č. 69/2018 o kybernetickej bezpečnosti a o zmene a doplnení niektorých zákonov," 2018, [prevzaté 17. mája 2018]. [Online] Dostupné z: https://www.slov-lex.sk/static/pdf/2018/69/ZZ_2018_69_20180401.pdf

[40] (2018) Vagrant by hashicorp. [prevzaté 14. mája 2018]. [Online] Dostupné z: https://www.vagrantup.com/

[41] Ondrej. (2014) W3SA. [prevzaté 12. mája 2018]. [Online] Dostupné z: http://www.salstar.sk/doc/w3sa/