„Next-Gen“ antiviry

Pokročilejší zabezpečení nebo jen buzzword?

část I.

V dnešní době se můžeme čím dál častěji setkat s pojmem Next-Gen antivirus nebo antivirus příští generace. Co však tento výraz znamená? Přináší opravdu vyšší bezpečnost? Nebo se jedná pouze o marketingový tah, který má nalákat uživatele a zvýšit prodejnost produktu?

Dvoudílný seriál „Next-Gen“ antiviry přiblíží aktuální situaci na trhu s antivirovými produkty a popíše rozdíly mezi „běžnými“ antiviry a nově vznikajícími antiviry příští generace. Obsahem prvního dílu bude porovnání funkcí a popis principů detekce obou typů antivirů. V druhém dílu se můžete těšit na technické porovnání, účinnost detekce, nároky na systémové zdroje a další.

Trendy v oblasti malwaru se mění každým rokem. Ještě nedávno byl nejčastějším typem ransomware. Ten je však podle posledních zjištění společnosti Kaspersky Lab [1] na ústupu a na vrchol se postupně dostává malware těžící kryptoměny, který zneužívá zdroje počítače k dolování. Většina druhů tohoto malwaru využívá aktivní relaci v prohlížeči. Nedávno byl ale objeven nový typ toho malwaru – XMRig, který ke svému běhu nepotřebuje relaci v prohlížeči, ale běží na koncovém zařízení. [2]

Stále častěji se můžeme setkat také s malwarem, který nenapadne soubor uložený v počítači, ale sám sebe uloží do operační paměti počítače. Ta byla dříve skenována antivirovými programy pouze skenem na vyžádání, nikoli ochranou v reálném čase[1]. Vzhledem k rozšíření tohoto typu malwaru však musí antivirové společnosti do svých produktů zařadit i skenování operační paměti v reálném čase.

[1] Např. slovenský výrobce ESET uvedl funkci skenování operační paměti v reálném čase do firemních produktů v roce 2014, ačkoli skenování operační paměti na vyžádání umí již více než osm let. [3]

[2] Hash je výstup hashovací funkce, kdy vstup je řetězec o libovolné velikosti a výstup je řetězec o pevně dané velikosti. Malá změna ve vstupním řetězci má za následek velkou změnu výstupního řetězce.

Principy detekce dnešních antivirů

Běžné antiviry dnes detekují malware dvěma hlavními způsoby – detekcí na základě signatur a heuristickou detekcí. Signaturou můžeme nazvat hash[2] malwaru, jeho otisk. Tento typ detekce však není příliš účinný, neboť útočníci pozměňují malware právě proto, aby se vyhnuli těmto detekčním mechanismům (více o metodách využívaných útočníky pro skrytí malwaru viz Box 1 na následující straně). Detekce na základě signatur může detekovat pouze již známý malware, který byl někdy spatřen a analyzován a pro který byla signatura vytvořena. Tento typ detekce je neúčinný proti neznámým druhům malwaru, které ještě nemusely být analyzovány. Pokud tedy útočník vytvoří nový malware, určitou dobu trvá, než ho antivirové programy zachytí, analyzují a výrobci jej přidají do svých databází.

Heuristická detekce neboli také behaviorální (sledující chování aplikace) hledá určité instrukce nebo příkazy programu, které nejsou pro běžnou aplikaci typické. Příkladem těchto příkazů může být komunikace s Command&Control serverem, šifrování souborů, sebereplikace apod. Tento typ detekce umožňuje odhalit i neznámý malware, který doposud nebyl objeven a analyzován. Heuristická detekce je ze svého principu náročnější na systémové prostředky koncové stanice, může ovlivnit její výkon, ale na druhou stranu přináší lepší výsledky než detekce na základě signatur.

Máme několik typů heuristické detekce, kdy každý funguje jiným způsobem. Emulace souborů (nebo také sandboxing či dynamická analýza) spustí soubor v řízeném virtuálním prostředí a zaznamenává všechny jeho aktivity. Pokud jsou aktivity škodlivé nebo potenciálně škodlivé, označí soubor jako malware. Sandboxová analýza je náročná na čas (často trvá i v řádu několika minut), protože je nutné malware spustit a nechat ho provést všechny instrukce. Některé typy malwaru dokážou detekovat svoji přítomnost ve virtuálním prostředí či v sandboxu a své instrukce vůbec neprovedou nebo provedou jen legitimní instrukce a sandbox je může označit jako neškodné. Stejně jako se zlepšují metody skrytí malwaru či detekce sandboxingu, zlepšují se současně techniky sandboxové analýzy, kterou výrobci používají ve svých produktech.

Dalším typem heuristické analýzy je analýza programového kódu souboru, tzv. statická analýza. Antivirový program se snaží pomocí reverzního inženýrství detekovat podezřelé instrukce v souboru, aniž by musel být soubor spuštěn. Na rozdíl od detekce signatur antivirový program nehledá konkrétní instrukce, ale pouze ty podezřelé či škodlivé, které mohou v počítači způsobit škody. I zde útočníci využívají obfuskačních metod pro znemožnění čitelnosti kódu.

Obfuskační metody a polymorfický malware — BOX 1

Obfuskační metody

Metodám, které útočníci využívají pro skrytí svého škodlivého kódu, se říká obfuskační. [4] Mají za cíl pozměnit programový kód malwaru tak, aby byl jen velmi těžko čitelný, aby bylo obtížné provést reverzní inženýrství, a mají zakrýt jeho škodlivý záměr. Malware využívající obfuskačních technik často provádí legitimní instrukce, aby nebyl podezřelý. Dále zaměňuje své instrukce za obdobné, které provádějí stejnou činnost, ale jiným způsobem, nebo mění a přesouvá části svého programového kódu, aby se změnila jeho signatura. Často se také používá metoda šifrování kódu za účelem ztížení čitelnosti.

Polymorfický malware

Jako polymorfický malware můžeme označit ten, který využívá obfuskačních technik ke skrytí škodlivého záměru. [4] Po každém infikování počítače pozmění svůj kód pomocí těchto technik, aby ztížil nebo úplně znemožnil detekci na základě signatur. Každé infikování tedy způsobí změnu malwaru.

Další Next-Gen produkty

Ani jedna z výše uvedených technik detekce nereaguje na aktuálně rapidně se měnící malware. Antivirové společnosti nestíhají držet rychlé tempo s útočníky, kteří modifikují malware velmi rychle nebo kombinují různé typy malwarových rodin mezi sebou. Výrobci antivirů reagovali na tento trend představením nových produktů, tzv. Next-Gen antivirů, tedy antivirů příští generace. Termín Next-Gen se v oblasti IT bezpečnosti objevuje často, např. v souvislosti s firewally.

Na rozdíl od současných antivirů nedisponují ty „nové“ detekcí na základě signatur ani statickou analýzou kódu. Protože se malware i jeho otisk mění velmi rychle, není detekce signatur na místě. Také statická analýza nereflektuje rychlé změny malwaru, proto ani ta není v antivirech příští generace využita.

Detekční metody Next-Gen antivirů

Podle Bena Johnsona [5] můžeme antivirové programy rozdělit na dva typy na základě toho, zda se při detekci malwaru omezuje antivirový program výhradně na danou stanici nebo bere v potaz také informace z dalších stanic v síti. Antiviry, které používají výhradně detekci na základě signatur nebo heuristickou detekci, obecně pohlížejí na koncovou stanici jako na jednotlivce. Při detekci škodlivého kódu se omezují výhradně na daný infikovaný soubor a nehledají souvislosti. V anglickém jazyce se používá označení „malware-centric view of endpoint security“. Naopak antiviry příští generace pohlížejí na koncovou stanici opačným způsobem – ne jako na jednotlivce, ale jako na součást celku. Zkoumají každý proces na každé stanici v síti a za pomoci speciálních algoritmů hledají souvislosti mezi detekovanými infikovanými soubory, blokovanými procesy, síťovými útoky na stanice apod. Používá se označení „system-centric view of endpoint security“. Události z různých stanic a chování různých procesů korelují, čímž získávají přehled o kompletním dění v síti i na koncových stanicích. Jejich účelem není pouze ochrana proti malwaru, ale komplexní zabezpečení koncové stanice.

Výše zmíněné speciální algoritmy hledající souvislosti mezi bezpečnostními incidenty v síti nejsou ničím jiným než algoritmy strojového učení. V dnešní době jsou v oblasti bezpečnosti velmi rozšířené a k detekci útoku nebo jiných škodlivých aktivit je využívají nejen výrobci antivirů. Jednoduše lze princip algoritmů strojového učení využitých v antivirových programech popsat tak, že vyhodnocují neznámé procesy a na základě více či méně kvalitních rozhodovacích mechanismů kategorizují procesy do několika skupin podle potenciálního rizika. Pokud riziko překročí stanovený práh, je soubor, který spustil proces, detekován jako malware. Podrobnější rozbor strojového učení bude uveden dále.

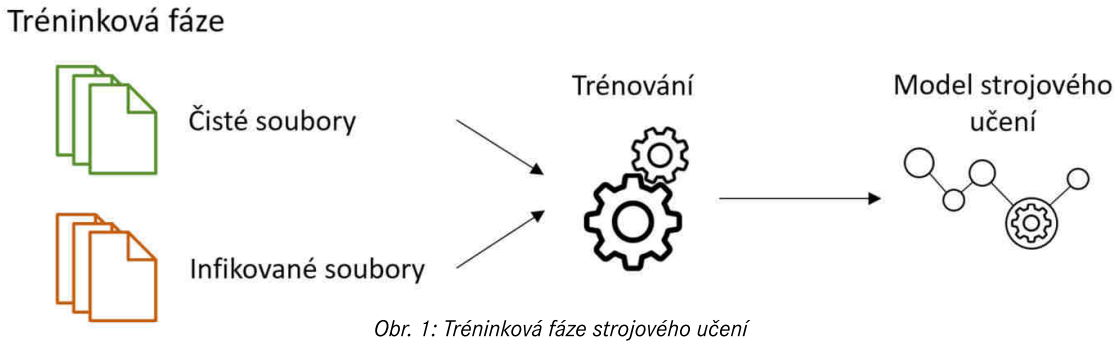
Další technologií, kterou antiviry příští generace využívají, jsou indikátory kompromitace systému, tzv. Indicators of Compromise (IOC). Typickým indikátorem kompromitace systému může být otisk malwaru (nebo také signatura), IP adresa, MD5 hash souboru, URL nebo doménové jméno serveru, se kterým proces komunikuje apod. [6,7] Jakmile je některý z těchto indikátorů zaznamenán, antivirus se zaměří na daný proces a sleduje jeho aktivitu. Pokud je podezřelá, nebo dokonce škodlivá, ohlásí detekci malwaru. Při detekci se nelze spoléhat pouze na tyto indikátory, protože hrozí velká pravděpodobnost chybného označení, tzv. false positive.

Strojové učení v akci

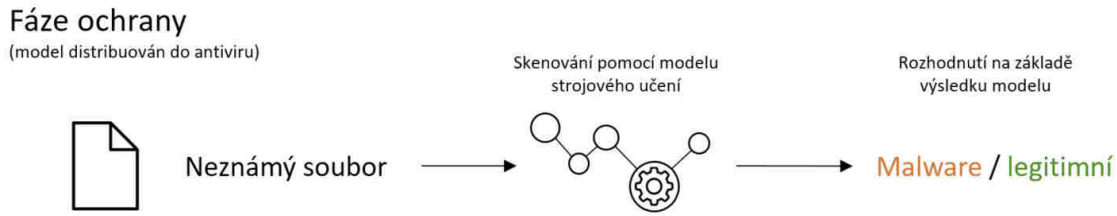
Pro vysvětlení principu strojového učení za účelem detekce malwaru je nutné vědět, co to strojové učení je, jak funguje a jaké rozeznáváme typy. Strojové učení představuje soubor metod, které dávají programu schopnost učit se bez explicitního naprogramování. V oblasti detekce malwaru to znamená, že program umí rozeznat malware od legitimního souboru bez nutnosti mu to vysloveně říci.

Algoritmy strojového učení se dělí do dvou skupin – s učitelem a bez učitele. Nejdříve se zaměříme na učení bez učitele. Využívá se k zařazení určitých dat do skupin podle podobnosti. Tento typ hledá společné parametry dat a kategorizuje je do skupin právě dle jejich podobnosti. Učení bez učitele není příliš vhodné k efektivní detekci malwaru, protože nedokáže posoudit, zda je daný soubor malware. Typickým představitelem této skupiny algoritmů strojového učení je clustering. Jedná se o proces seskupení dat na základě jejich podobnosti. [8] Může se využívat např. při kategorizování malwaru do jednotlivých malwarových rodin, ale už ne k posouzení, zda daný soubor je či není malware.

Druhou skupinou je učení s učitelem. Algoritmus dostane vstupní data (vzorky souborů), u kterých je stanoveno, zda se jedná o malware nebo o legitimní soubor. Fáze, kdy jsou algoritmu předkládána data, u nichž jsou známy výsledky, se nazývá tréninková. Data, nad kterými trénování probíhá, jsou tréninková data. Právě tréninková data jsou slabou stránkou všech algoritmů strojového učení využitých pro detekci malwaru. Tréninková množina musí být dostatečně obsáhlá, aby byl v budoucnosti algoritmus schopný detekovat i neznámý malware, pozměněný nebo zkombinovaný s dalšími malwarovými rodinami.



Obr. 1: Tréninková fáze strojového učení



Obr. 2: Fáze ochrany strojového učení

Jakmile je dokončena fáze trénování, výsledný systém strojového učení se může zapracovat do antivirového programu a distribuovat mezi uživatele. Ostré nasazení produktu již spadá do fáze, kdy jsou vstupní data neklasifikovanými vzorky souborů zachycenými v reálném světě, o kterých se musí rozhodnout, zda se jedná o malware.

Obvykle se jako algoritmus strojového učení pro detekci malwaru vybírá neuronová síť nebo rozhodovací stromy. [9] My se zaměříme na neuronovou síť. Během tréninkové fáze (viz obr. 1) jsou neuronové síti poskytnuty klasifikované vzorky souborů. U vzorků, které jsou označeny jako malware, jsou definovány určité znaky, které malware obsahuje. Může se jednat o specifické aspekty chování, informace o souboru, volání API funkcí apod. Tyto speciální znaky si neuronová síť uloží jako vzory, které využije při detekci.

Po naučení modelu rozpoznávat infikované i čisté soubory přichází druhá fáze – ochranná. Model je distribuován do antiviru uživatelů. V této fázi (viz obr. 2) se provádí skenování neznámých souborů a na základě naučených vzorů rozhoduje o tom, zda je neznámý soubor malware. Jakmile se zjistí, že některý ze zkoumaných souborů je podobný danému vzoru (má specifické chování, metadata, volá podezřelé API funkce atd.), označí soubor za malware. Výhodou těchto vzorů oproti klasické detekci na základě signatur je to, že vzory používané ve strojovém učení jsou mnohem obecnější a není nutné mít specifickou signaturu pro každý typ malwaru.

Počet vzorů je tedy minimální oproti počtu signatur, a tím můžeme radikálně snížit velikost virové databáze.

Měnící se malware není problémem

Antivirové společnosti mají rozsáhlé databáze vzorků škodlivého kódu i čistých souborů, avšak ani to nestačí pro zajištění vysoké pravděpodobnosti úspěchu detekce, která se v ideálním případě blíží hranici sta procent. Je to z toho důvodu, že každý den vzniká nový a nový malware, stejně tak jako vznikají nové neškodné soubory. Starý naučený model strojového učení nemůže se stejnými vzory fungovat stejně kvalitně po celou dobu využívání.

Tento problém je nutné řešit tak, že naší neuronové síti poskytneme aktuální vzory. Jakmile se nový malware chová jinak, používá jiné metody a volá jiné funkce, je nutné vytvořit nový obecný vzor, který poté pošleme formou aktualizace virové databáze do antiviru. Opět se to může zdát podobné s detekcí na základě signatur, ale jak už bylo řečeno, vzory jsou oproti signaturám mnohem obecnější. Není tedy nutné provádět aktualizace virové databáze tak často. Malwarové rodiny jsou založeny na stejných principech, a tudíž jsou vzory stejné pro mnoho různých malwarů. [11]

Zastaralé funkce již nejsou potřeba

Prozatím jsme se bavili pouze o detekčních metodách, které využívají současné nebo Next-Gen antiviry. Ty ale disponují také funkcemi, které se v dnešní době označují za staromódní, či dokonce nefunkční.

Příkladem této funkce může být pravidelné skenování souborů v počítači. Tento typ skenování je opět postaven na detekci na základě signatur. V průběhu skenování je kontrolován soubor za souborem a ověřuje se, jestli některý z nich, resp. jeho otisk, není ve virové databázi. Jelikož Next-Gen antiviry nevyužívají detekci na základě signatur, neobsahují ani tuto funkci. Navíc velkým problémem současných antivirů je dopad na výkon, který je o to znatelnější právě v případě skenování souborového systému. Next-Gen antiviry nemají v tomto případě žádný dopad na výkon, jelikož výše popsanou funkcí nedisponují. [12]

Malware dokonce i bez souboru

Trendem poslední doby je malware, který neinfikuje soubor, ale běží v operační paměti počítače. Tento typ není detekovatelný běžnou ochranou v reálném čase nebo skenováním souborů. Útok pomocí tzv. fileless malware může využívat takové nástroje, jako např. PowerShell nebo Windows Management Instrumentation (WMI). Pomocí těchto nástrojů může útočník provádět škodlivou aktivitu na úrovni příkazového řádku, protože PowerShell i WMI jsou důvěryhodnými komponenty operačního systému Windows a většina bezpečnostních nástrojů je neskenuje a nesleduje jejich aktivity. Další možností je využít plánovač úloh v operačním systému a pomocí něj spouštět škodlivý skript.

Jak je zřejmé, tento typ útoku nevyužívá žádný infikovaný soubor a samotné škodlivé aktivity se provádějí v operační paměti a nevyužívají souborový systém. Tím se obejde běžná ochrana, která je naopak založena právě na detekci škodlivého kódu v souborovém systému.

Možnost bránit se tomuto typu malwaru ale přesto existuje. Nejjednodušeji toho lze docílit tím, že zakážeme PowerShell a WMI. Bohužel jsou tyto funkce využívány IT administrátory a ti bez nich často nedokážou efektivně spravovat své systémy. Mnohem častěji používanou metodou je behaviorální analýza. Tu nabízí většina Next-Gen, ale i dnešních antivirů. Antivirus sleduje chování procesů (je nutné, aby sledoval i legitimní procesy a byl v nich zahrnut právě i PowerShell), a pokud objeví podezřelé chování (spuštění PowerShellu neznámým procesem nebo jiný vzor chování), vyhodnotí situaci jako útok a informuje uživatele.

Definice Threat Intelligence **BOX 2**

Threat Intelligence je služba poskytující informace o aktuálních bezpečnostních hrozbách. Poskytuje informace o mechanismu hrozby, kontextu, indikátorech a důsledcích hrozby společně s radami, jak se proti těmto hrozbám chránit. [10]

Endpoint Detection and Response

Novinkou posledních zhruba dvou let se stala funkce Endpoint Detection and Response[3]. Nástroje, které obsahují tuto funkci, aktivně monitorují koncové stanice a servery a sbírají logy, informace o přenášených paketech nebo o chování procesů. Tyto informace porovnávají s již zmíněnými indikátory kompromitace systému získanými např. ze služeb Threat Intelligence (viz Box 2). Dále tyto informace využívají k forenzní analýze útoku, tedy odkud byl útok veden, jakým způsobem infikoval počítač, jak se dál šířil atd.

Dříve, než může být malware detekován běžným antivirem, např. když komunikuje s Command&Control serverem, je pomocí indikátorů kompromitace odhalen (antivirus, který nevyužívá těchto indikátorů kompromitace, by malware zachytil až ve chvíli, kdy začne škodit v systému). Jakmile je detekován, proběhne automatická akce antivirového programu, která (dle konfigurace) může např. zablokovat spojení s řídícím Command&Control serverem nebo izolovat stanici a pokusit se malware odstranit.

Opravdu Next-Gen?

Výrobci Next-Gen antivirů, kteří uvádějí, že nevyužívají detekci na základě signatur, používají tzv. signature-less detekci. Ani to ale

[3] Podrobné rozebrání funkce EDR najdete v článku „Když prevence nestačí aneb co se skrývá pod zkratkou EDR?“ Pavla Krátkého v DSM 1/2018.

není zcela přesné. Výše uvedené indikátory kompromitace jsou svým způsobem také signatury. Někdo musel sepsat pravidla, jak může vypadat kompromitace systému, jaké IP adresy jsou k útoku využívány, na jaké domény komunikuje malware při připojení na Command&Control server apod. Stejně tak vzory u strojového učení můžeme svým způsobem považovat za signatury.

Nemůžeme tedy obecně říci, že metody detekce úplně postrádají princip signatur. Ovšem zásadní rozdíl je v tom, že při použití strojového učení a indikátorů kompromitace je počet vzorů mnohem nižší než při běžné detekci stavěné na otiscích souborů. Avšak detekce na základě signatur při ochraně v reálném čase je velmi rychlá a pouze nepatrné množství souborů je chybně označených jako malware.

Na první pohled by se mohlo zdát, že Next-Gen antiviry jsou v detekci mnohem dále než dnešní běžně užívané antivirové programy. Opak je ale pravdou. Antivirové programy a techniky detekce se neustále vyvíjejí a výrobci přidávají další a další funkce pro zlepšení zabezpečení. Nedošlo k žádnému razantnímu skoku v technikách detekce malwaru. V oblasti antivirových produktů, ale nejen v ní, je vývoj postupný. Reflektuje trendy v oblasti malwaru a nové techniky útočníků. Výrobci těch „běžných“ antivirů už také zabudovali do detekčních mechanismů korelaci událostí z celé sítě, algoritmy strojového učení nebo další výše popsané techniky. [13]

Abychom se ale mohli dobře orientovat na trhu antivirových produktů, můžeme říci, že stěžejní body Next-Gen antivirů jsou:

- nahrazení detekce na základě signatur behaviorální analýzou,
- zabudování strojového učení,
- využití indikátorů kompromitace,
- detekce útoků bez infikování souborů,
- funkce Endpoint Detection and Response.

Tyto funkce však nejsou součástí pouze Next-Gen antivirů, ale postupně je do svých řešení zabudovávají menší i větší hráči na poli antivirového softwaru.

Ať už se označení Next-Gen antivirus uchytí a budeme ho v budoucnu běžně používat, nebo ne, je nutné sledovat trendy v oblasti malwaru a kontrolovat, zda výrobce vašeho bezpečnostního řešení včas reaguje na měnící se techniky útočníků. Výše uvedené funkcionality je rozhodně důležité vyžadovat od kvalitního antivirového programu, který je základním prvkem bezpečnosti každé sítě.

V příštím dílu tohoto článku se podíváme na úspěšnost detekce běžných i Next-Gen antivirů a také porovnáme dopad na výkon koncové stanice vybraných produktů z obou kategorií.

David Pecl
David.Pecl@aec.cz

David Pecl

Bezpečnostní specialista se zaměřením na ochranu koncových stanic a serverů ve společnosti AEC a.s. V současné době se věnuje také detekci zranitelností a oblasti mobilní bezpečnosti.

POUŽITÉ ZDROJE

[1] PIKE, Sarah. Cryptominers gain ground. Kaspersky Lab Daily [online]. 2018-06-27 [cit. 2018-07-24]. Dostupné z: https://www.kaspersky.com/blog/cryptominers-almost-double/22898/
[2] *March's Most Wanted Malware: Cryptomining Malware That Works Even Outside the Web Browser on the Rise.* Check Point Blog [online]. [cit. 2018-07-24]. Dostupné z: https://blog.checkpoint.com/2018/04/13/marchs-wanted-malware-cryptomining-malware-works-even-outside-web-browser-rise/
[3] Advanced Memory Scanner. ESET [online]. [cit. 2018-07-31]. Dostupné z: https://support.eset.com/kb3603/
[4] SELAMAT, Nur Syuhada, Fakariah Hani Mohd ALI a Noor Ashitah Abu OTHMAN. Polymorphic Malware Detection [online]. Praha: International Conference on IT Convergence and Security (ICITCS), 2016 [cit. 2018-07-31]. DOI: 10.1109/ICITCS.2016.7740362 Dostupné z: https://ieeexplore.ieee.org/document/7740362/
[5] JOHNSON, Ben. What is Next-Generation Antivirus? Carbon Black [online]. 2016-10-11 [cit. 2018-07-25]. Dostupné z: https://www.carbonblack.com/2016/11/10/next-generation-antivirus-ngav
[6] LOCK, Hun-Ya. Using IOC (Indicators of Compromise) in Malware Forensics. SANS Institute [online]. 2013-21-02 [cit. 2018-07-25]. Dostupné z: https://www.sans.org/reading-room/whitepapers/forensics/ioc-indicators-compromise-malware-forensics-34200
[7] Indicators of Compromise. Trend Micro [online]. 2013-21-02 [cit. 2018-07-25]. Dostupné z: https://www.trendmicro.com/vinfo/us/security/definition/indicators-of-compromise
[8] NAGPAL, Anuja. Clustering – Unsupervised Learning. Towards Data Science [online]. [cit. 2018-07-31]. Dostupné z: https://towardsdatascience.com/clustering-unsupervised-learning-788b215b074b
[9] Machine Learning for Malware Detection. Kaspersky Lab [online]. [cit. 2018-07-24]. Dostupné z: https://media.kaspersky.com/en/enterprise-security/Kaspersky-Lab-Whitepaper-Machine-Learning.pdf
[10] MCMILLAN, Rob. Definition: Threat Intelligence. Gartner [online]. 2013 [cit. 2018-08-20]. Dostupné z: https://www.gartner.com/doc/2487216/definition-threat-intelligence
[11] GAVRILUȚ, Dragoș, Mihai CIMPOEȘU, Dan ANTON a Liviu CIORTU. Malware detection using machine learning [online]. Mragowo: International Multiconference on Computer Science and Information Technology, 2009, 735-741 [cit. 2018-07-24]. DOI: 10.1109/IMCSIT.2009.5352759. ISSN 2157-5525. Dostupné z: https://ieeexplore.ieee.org/document/5352759/
[12] MANSOUR, Adam. "Next-Gen" Anti-Virus Vs. Anti-Virus is there a difference? IntelliGO Blog [online]. 2017-09-25 [cit. 2018-07-24]. Dostupné z: http://www.intelligonetworks.com/blog/next-gen-av-vs-av
[13] HARLEY, David. Next-gen security software: Myths and marketing. WeLiveSecurity [online]. 2017-02-13 [cit. 2018-07-24]. Dostupné z: https://www.welivesecurity.com/2017/02/13/next-gen-security-software-myths-marketing/