VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INFORMAČNÍCH SYSTÉMŮ
DEPARTMENT OF INFORMATION SYSTEMS

KYBERNETICKÉ HROZBY V KOMUNIKACI DNS

CYBER THREATS IN DNS COMMUNICATION

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE **PETR PŮČEK**
AUTHOR

VEDOUCÍ PRÁCE **doc. Ing. PETR MATOUŠEK, Ph.D., M.A.**
SUPERVISOR

BRNO 2023

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

Zadání bakalářské práce

144351

Ústav: Ústav informačních systémů (UIFS)
Student: **Půček Petr**
Program: Informační technologie
Specializace: Informační technologie
Název: **Kybernetické hrozby v komunikaci DNS**
Kategorie: Počítačové sítě
Akademický rok: 2022/23

Zadání:

1. Seznamte se s bezpečnostními hrozbami využívající komunikaci DNS, např. Project Sauron/Remsec, Alina POS, InvisiMole, DNS Messenger, SIGRed, Ripple20, DGA, NOD a další.
2. Proveďte rešerši dostupných nástrojů implementující výše uvedené útoky a volně dostupných datových sad s útoky na DNS.
3. Navrhněte způsob monitorování DNS tak, aby bylo možné výše uvedené hrozby detekovat. Prozkoumejte možnosti monitorování NetFlow/IPFIX, logování událostí či detekce signatur v zařízení IDS.
4. Implementujte způsob detekce bezpečnostních hrozeb na vybraných monitorovacích datech. Ověřte účinnost detekce na několika scénářích.
5. Zhodnoťte dosažené výsledky a navrhněte možné rozšíření detekce útoků v DNS.

Literatura:

- Daihes Y., Tzaban H., Nadler A., Shabtai A. (2021) MORTON: Detection of Malicious Routines in Large-Scale DNS Traffic. In: Bertino E., Shulman H., Waidner M. (eds) Computer Security - ESORICS 2021. LNCS, vol 12972. Springer, Cham. https://doi.org/10.1007/978-3-030-88418-5_35 .
- M. Grill, I. Nikolaev, V. Valeros and M. Rehak, "Detecting DGA malware using NetFlow," 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), 2015, pp. 1304-1309, doi: 10.1109/INM.2015.7140486.
- S. Torabi, A. Boukhtouta, C. Assi and M. Debbabi, "Detecting Internet Abuse by Analyzing Passive DNS Traffic: A Survey of Implemented Systems," in IEEE Communications Surveys & Tutorials, vol. 20, no. 4, pp. 3389-3415, Fourthquarter 2018, doi: 10.1109/COMST.2018.2849614.

Při obhajobě semestrální části projektu je požadováno:

- Body 1 - 3.

Podrobné závazné pokyny pro vypracování práce viz https://www.fit.vut.cz/study/theses/

Vedoucí práce: **Matoušek Petr, doc. Ing., Ph.D., M.A.**
Vedoucí ústavu: Kolář Dušan, doc. Dr. Ing.
Datum zadání: 1.11.2022
Termín pro odevzdání: 10.5.2023
Datum schválení: 18.10.2022

Abstrakt

Cílem této práce je seznámit čtenáře s hrozbami, které se vyskytují v systému DNS, a implementovat jejich detekci. V úvodu práce je provedena rešerše těchto hrozeb a získání datových sad, které je obsahují. Dále jsou představeny nalezené nebo vytvořené nástroje, které umožňují implementovat vybrané hrozby. Práce popisuje použitelnost různých monitorovacích systémů při detekci, jako je například logování na serveru BIND 9, záznamy IPFIX nebo systém IDS. Hlavním výstupem práce je vytvoření detekčního nástroje, který je otestován na detekci hrozeb jako jsou Alina POS, DNSMessenger a SIGRed. Vytvořený nástroj také poskytuje původní metodu pro detekci generovaných domén pomocí algoritmů DGA. Přínosem této práce je tedy implementace modulárního detekčního nástroje, který je snadno rozšiřitelný o další podporu monitorování a detekce nových hrozeb.

Abstract

The aim of this work is to introduce the reader to the threats that occur in the DNS system and implement their detection. The introduction of the work includes research on these threats and obtaining datasets that contain them. Additionally, discovered or created tools are presented that allow for the implementation of selected threats. The work also describes the usability of different monitoring systems for detection, such as logging on BIND 9 server, IPFIX records, or an IDS system. The main output of the work is the creation of a detection tool that is tested for the detection of threats such as Alina POS, DNSMessenger, and SIGRed. The created tool also provides original method for detecting generated domains using DGA algorithms. The benefit of this work is therefore the implementation of a modular detection tool that is easily expandable to support monitoring of additional types and detecting new threats.

Klíčová slova

útoky v komunikaci DNS, detekce hrozeb, malware AlinaPOS, útok DNSMessenger, hrozba SIGRed, algoritmy DGA

Keywords

DNS communication attacks, threat detection, AlinaPOS malware, DNSMessenger attack, SIGRed vulnerability, DGA algorithms

Citace

PŮČEK, Petr. *Kybernetické hrozby v komunikaci DNS*. Brno, 2023. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce doc. Ing. Petr Matoušek, Ph.D., M.A.

Kybernetické hrozby v komunikaci DNS

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana doc. Ing. Petra Matouška Ph.D., M.A. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....
Petr Půček
10. května 2023

Poděkování

Především bych rád poděkoval panu doc. Ing. Petru Matouškovi Ph.D. M.A. za jeho odborné vedení mé práce, velkou ochotu a mnoho přínosných rad. Také bych rád poděkoval panu Janu Rynešovi za prvotní inspiraci a poskytnutí informací či nástrojů k vybraným hrozbám.

Obsah

Kapitola 1

Úvod

V dnešním internetu hraje naprosto klíčovou roli překlad doménového jména na IP adresu, což je jedna z nabízených služeb systému DNS. Na něm závisí funkčnost většiny síťových aplikací poskytovaných na internetu. Z hlediska důležitosti pro navazování spojení různými protokoly není ve většině případů komunikace DNS v síťovém provozu blokována a ani dále nijak filtrována. To dává prostor potenciálním útočníkům tento komunikační kanál zneužít ve vlastní prospěch. Takový útok může proběhnout naprosto bez povšimnutí oběti či správce sítě, ve které k útoku došlo.

Cílem této práce je implementovat a vyhodnotit detekci bezpečnostních hrozeb v systému DNS. K dosažení tohoto cíle se nejdříve seznámíme s hrozbami využívající komunikaci DNS, provedeme rešerši dostupných nástrojů implementující výše uvedené útoky a rešerši volně dostupných datových sad s útoky na DNS. Dále pak prozkoumáme možnosti různých způsobů monitorování a zhodnotíme jejich vhodnost pro detekci vybraných hrozeb. Poté navrhneme způsoby detekce vybraných hrozeb a implementujeme příslušný detekční nástroj. Na závěr provedeme validaci detekčního nástroje a ověříme účinnost detekce na různých scénářích.

Práce je rozdělena na čtyři části. V první části jsou detailně popsány bezpečnostní hrozby využívající komunikaci DNS. Je provedeno jejich rozřazení podle povahy útoku a jsou popsány postupy, kterými útočníci zneužívají systém DNS.

V další části je provedena rešerše datových sad a nástrojů pro získání přehledu o hrozbách v systému DNS. Jsou zde blíže popsány průběhy získávání reálných dat komunikace jednotlivých hrozeb a postupy vytváření simulačních nástrojů, které budou dané hrozby přesvědčivě simulovat.

Ve třetí části je provedena analýza různých způsobů monitorovaní komunikace DNS a vyhodnocení jejich vhodnosti vzhledem k detekci hrozeb. Jsou zde uvedeny konkrétní podrobné návrhy detekce hrozeb zejména s ohledem na možnosti monitorování NetFlow/IPFIX, logování událostí a detekce pomocí signatur v zařízení IDS. Dále jsou zde popsány původní návrhy detekce hrozeb Alina POS, DNSMessenger, SIGRed a algoritmů DGA.

V poslední části práce je detailně popsán princip detekčního nástroje, jeho implementace a následné testování pro hrozby Alina POS, DNSMessenger, SIGRed a algoritmy DGA. Jsou zde dokumentovány využité postupy a výsledky a závěry testování úspěšnosti detekce na různých monitorovacích datech v rámci řady experimentů.

Kapitola 2

Bezpečnostní hrozby služby DNS

Jak již bylo zmíněno v úvodu této práce, útočníci využívají komunikace v rámci systému DNS ke škodlivým aktivitám právě vzhledem k nepostradatelnosti služeb tohoto systému. To vede k tomu, že síťovému provozu s komunikací DNS je mnohdy nechán nekontrolovaný průběh. Tento fakt útočníkům zvyšuje pravděpodobnost nepovšimnutí útoku a tudíž i jeho šance na úspěch.

Cílem této kapitoly je se důkladně seznámit s reálnými hrozbami a popsat, jakým způsobem zneužívají systém DNS a jeho protokol ke škodlivé komunikaci. Taktéž jsou zmíněné útoky v této kapitole rozřazeny do kategorií podle povahy bezpečnostní hrozby. U každé z kategorií je zprvu popsán obecný princip platný k celé skupině těchto útoků a dále již následuje popis konkrétních hrozeb. Na závěr této kapitoly je sestaveno porovnání všech hrozeb dle různých relevantních parametrů.

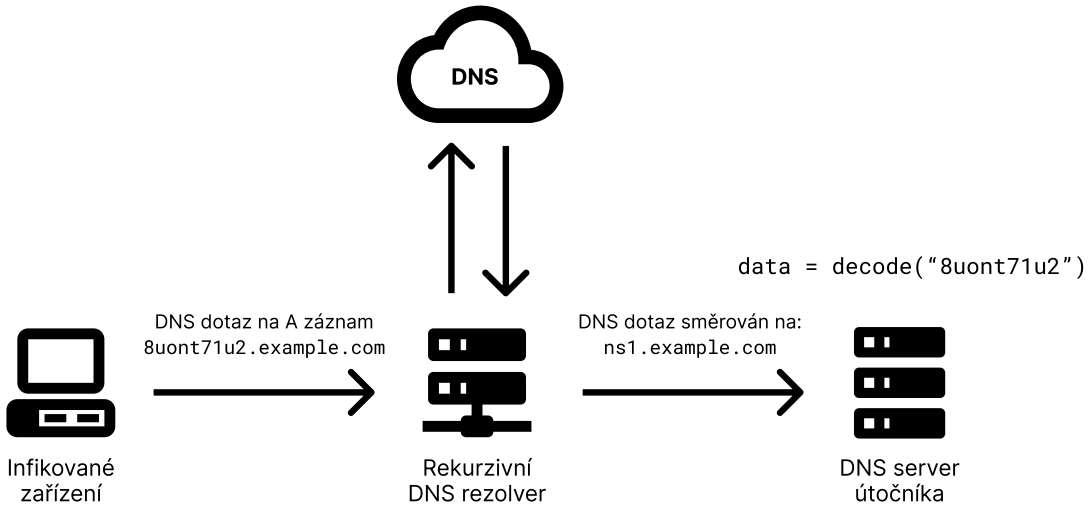
2.1 Exfiltrační útoky

Jedním z možných typů útoku s pomocí protokolu DNS je exfiltrace dat. Ta spočívá v extrakci potažmo citlivých údajů či souborů za využití DNS jakožto komunikačního kanálu mezi kompromitovaným zařízením a útočníkem, respektive jeho serverem DNS. Tomuto jevu se odborně říká tunelování DNS [28]. Využití této techniky útočníkem je znázorněno na obrázku 2.1.

Pro účely ilustrace je použita doména `example.com`, která je v tomto případě vlastněná útočníkem. K této doméně útočník taktéž zřídí primární server DNS `ns1.example.com`, na který budou posléze přicházet dotazy DNS. Mezitím na straně infikovaného klienta dojde k zakódování patřičného citlivého údaje pomocí zvoleného postupu útočníkem. Tento zakódovaný údaj pak napadené zařízení připojí jako subdoménu k doméně `example.com`. Výsledné doménové jméno, na který pošle klient dotaz DNS, je tedy například `8uont71u2.example.com`, kde „8uont71u2“ jsou zakódovaná data. Tento dotaz doputuje pomocí systému DNS až na útočníkův zmíněný server, kde jej útočník zpracuje dekódováním subdomény.

Značnou výhodou tohoto přístupu je jeho možnost působit nepozorovaně vzhledem k bezpečnostním síťovým prvkům. To mu přináší výhodu oproti jiným, k přenosu dat přímo určeným, protokolům. Exfiltraci pomocí DNS je však možné detekovat na různých monitorovacích prvcích v síti, avšak musí být k tomuto účelu nakonfigurovány [15]. Tyto principy jsou dále popsány v kapitolách 4 a 5.

Největší slabinou této metody je však její přenosová rychlost, která je limitována maximální délkou subdomény na 63 oktetů a taktéž horním limitem 255 oktetů na celé doménové



Obrázek 2.1: Příklad exfiltračního útoku pomocí protokolu DNS.

jméno [32]. Kvůli tomu je většinou nutné exfiltrovaná data rozdělit do vícero dotazů DNS a posílat je postupně.

2.1.1 Špionážní nástroj „ProjectSauron“

Jeden z útoků, který princip extrakce dat pomocí tunelování DNS využívá, je malware, jež dostal název ProjectSauron, někdy též nazývaný jako Remsec [21]. Pravděpodobně byl využíván již od června roku 2011. Působil v zemích jako například Rusko, Írán či Rwanda a cílil především na vládní organizace, armádu nebo výzkumná centra.

Samotný malware pracuje v podobě spustitelné programové knihovny, která je nahraná v infikovaném zařízení. Tato knihovna, maskující se jako falešná utilita, tak má přístup k různým citlivým údajům. Celý nástroj a všechny jeho dodatky jsou postaveny na skriptovacím jazyce Lua. Veškeré moduly a síťové protokoly, se kterými pracuje, používají silné šifrovací algoritmy, jako například RC5, AES a další [21].

Průběh útoku

V případě tohoto malwaru je protokol DNS zneužit především pro tunelování dat z napadeného zařízení. Postup tohoto procesu je následovný [21]:

1. Data jsou před jejich odesláním zakódována pomocí programu `basex` do formátu Base64.
2. Následně je využit připravený programový modul malwaru `dext`, který data zpracuje připojením dat k doménovému jménu a vygeneruje z nich sadu paketů DNS.
3. Tyto pakety se pak postupně nástrojem `nslu` odesílají jeden za druhým.

Důležitým znakem je odesílání těchto dat pouze po 30 bajtových blocích, což účinně snižuje možnost odhalení. Z hlediska takto nízkého limitu pro odesílání dat se tato metoda využívá především na menší systémová metadata. Schéma útoku z hlediska komunikujících zařízení je principiálně totožné s obecným schématem pro různé exfiltrační útoky, viz obrázek 2.1.

Druhým způsobem upotřebení protokolu DNS k útoku je užít ho jako kanál pro předávání informací o postupu útoku v reálném čase. V případě dosažení nějakého důležitějšího milníku v rámci útoku se provede dotaz DNS na unikátní doménu pro patřičnou oběť a útočník má tak přehled o současné fázi a úspěšnosti útoku [21].

2.1.2 Malware „Alina POS“

Dalším útokem založeném na exfiltraci dat je malware s názvem Alina POS. Tento škodlivý software je z rodiny malwaru „Point of Sale“ [11]. Ty se vyznačující mířením svých útoků na prodejní zařízení jako jsou například platební terminály a další obdobné systémy. Jejich cílem je odcizení platebních údajů a následné zneužití. Toho jsou útočníci většinou schopni docílit pomocí chyb v systému, které jim umožní se dostat do paměti RAM, kde jsou již zmíněné platební údaje v dešifrované podobě [38]. Tuto metodu právě využívá malware Alina POS.

U výše zmíněných zařízení jsou nastavené striktní bezpečnostní podmínky, které například nedovolují využívání nezabezpečeného protokolu HTTP pro přenos platebních údajů. Proto se zde nabízí využití protokolu DNS jakožto tunelovacího nástroje. Malware Alina POS zpočátku využíval k exfiltraci platebních údajů pouze protokol HTTPS nebo jeho kombinaci s protokolem DNS a až později kompletně přešel na tunelování pomocí DNS.

Průběh útoku

Zneužití protokolu DNS zde probíhá ve formě komunikačního kanálu mezi napadeným zařízením a útočníkovým serverem Command & Control (dále v této práci zkráceně jako C2). Schéma komunikujících zařízení tohoto útoku je zřejmé z obecného příkladu č. 2.1 topologie exfitračních útoků. Průběh útoku je následující [38]:

1. odcizené platební údaje z paměti RAM jsou zpočátku zakódovány algoritmem Base64.

2. S takto zakódovanými daty je dále provedena operace XOR s bajtem 0xAA.

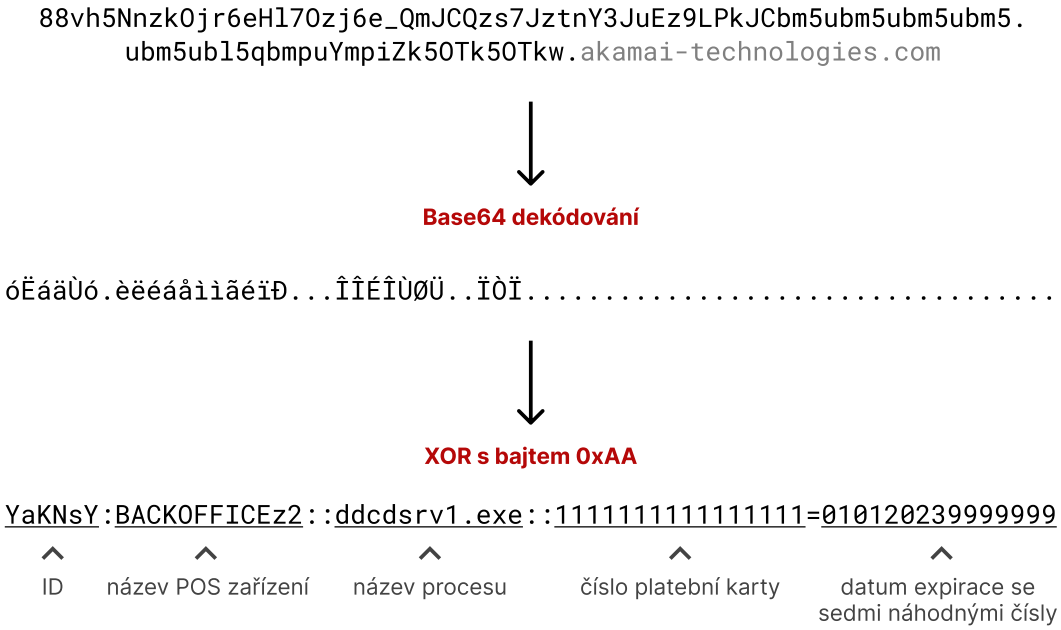
3. Tato data jsou pak dle již popsaného principu připojena jako subdoména k útočníkově doméně.

4. Posléze jsou vytvořeny dotazy DNS na doménu serveru C2, kde je žádán záznam typu A s daty ukrytými v doménovém jméně.

5. Na tyto dotazy pak server C2 v roli autoritativního doménového serveru většinou odpovídá záznamem s adresou IPv4 127.0.0.1.

Velké množství útoků se skrývalo za doménovým jménem `akamai-technologies.com` připomínající doménu americké společnosti Akamai Technologies. Malware Alina POS však využíval i jiných doménových jmen.

Na obrázku 2.2 je možné vidět strukturu komunikace s serverem C2. Prvních šest znaků je vyhrazeno pro identifikátor oběti přidělený útočníkem. Za oddělovačem „:“ je dále pravděpodobně systémový název napadeného zařízení. Následuje název procesu, který platební údaje získal z paměti a poslední část je již věnovaná samotným údajům o dané platební kartě [11].



Obrázek 2.2: Příklad dekódování a významu komunikačních dat se serverem C2.

2.2 Infiltrační útoky

Další možnou metodou zneužití protokolu DNS pro účely útoku může být infiltrace dat [23]. Zatímco přístup popsaný výše již počítal s tím, že malware se již do zařízení plně dostal jinou cestou, tak následující útoky využívají právě protokolu DNS pro zavedení různých modulů malwaru do napadeného systému. I v tomto případě je ale nutné, aby na zařízení oběti již malware s infiltračním modulem DNS byl, který poté zahájí tunelování dalších částí již v rámci komunikace DNS.

Zpravidla na začátku infiltrační komunikace DNS napadené zařízení pošle dotaz DNS na jeden z různých typů záznamů, kde pomocí podobného principu jako u exfiltrace zakóduje údaje o sobě. To pomůže útočníkovi lépe identifikovat oběť, a tedy i individuálně zacílit infiltraci škodlivých dat. Poté již útočník odpovídá na dotazy záznamem, kde jsou jistým způsobem zakódovaná data, která chce útočník do infikovaného zařízení dostat.

Může se jednat například o odpovědi DNS obsahující záznamy typu TXT, který nabízí díky své libovolné velikosti rychlejší způsob přenosu. Obvykle se taktéž používají odpovědi v podobě záznamu A. Adresa IPv4 z tohoto záznamu pak může být nositelem nějakého příkazu z serveru C2 útočníka, který se má na daném zařízení vykonat.

Existuje však nepřeberné množství možností, jak takovýto infiltrační nástroj využít. Různé záznamy nabízí jiný způsob, jak informaci přes ně přenést. Interpretace takových dat obsažených v paketech DNS je tak pouhým implementačním detailem útočníka.

2.2.1 Špionážní malware „InvisiMole“

Skupina InvisiMole působí již minimálně od roku 2013 v kyberšpionážních operacích především na Ukrajině a v Rusku [25]. Cílem jejich vyvinutého malwaru je především pořizování záznamu z webkamery a mikrofonu oběti, zjištění jejich polohy či získání přístupu k různým dokumentům. V roce 2019 přišli s inovovanou verzí tohoto malwaru, která podobně

jako předešlá verze cílila především na země z východní Evropy, konkrétně na organizace spojené s diplomacií či armádou. Tato nová verze si klade za cíl zlepšit nenápadnost útoku a snížit tak riziko odhalení.

Důležitou novinkou v nové verzi tohoto malwaru z hlediska této práce je využití protokolu DNS ke komunikaci se serverem C2 a taktéž i k infiltraci dat. Společně tak s druhým infiltračním nástrojem postaveném na protokolu TCP nabízí možnost stahovaní a spouštění různých aktualizací a dodatků malwaru ze serveru útočníka. Tento stahovač DNS je povětšinou využíván v pozdějších fázích útoku k provádění aktualizací [25].

Průběh útoku

Vzhledem ke komunikaci se serverem C2 tento modul DNS využívá vlastní implementaci tunelování DNS, jejíž postup je následující [25]:

1. z napadeného zařízení se posílají dotazy DNS na záznamy NULL a AAAA. Do takového dotazu jsou vloženy různé užitečné informace pro útočníka jako například typ požadavku, časová značka, název zařízení atd.

2. Tyto informace se převedou do binární zprávy, která se rozdělí na řetězce bitů s nejméně významným bitem na začátku.

3. Výsledný dlouhý bitový řetězec je doplněn tak, aby byl násobkem pěti. Dále je pak zakódován upraveným algoritmem Base32 s vlastním slovníkem.

4. Poté dojde k odeslání těchto dat na server útočníka.

5. Na tyto dotazy následně útočníkův server C2 odpovídá zmíněnými záznamy NULL a AAAA s patřičnými daty.

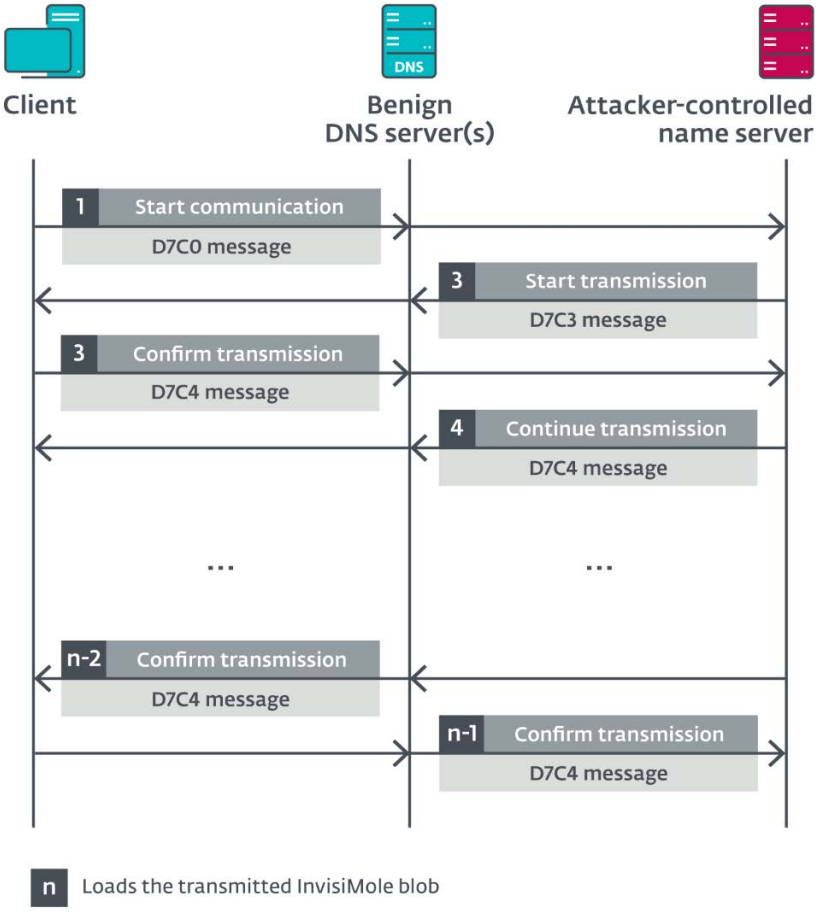
Jelikož je tato komunikace DNS vystavěna na protokolu UDP, tudíž je nespolehlivá, je nutné ještě přidat identifikátor přenosu [25].

Na grafické ukázce 2.3 lze vidět příklad infiltrační komunikace infikovaného zařízení s útočníkovým serverem. Napadený klient prvně přes benigní server DNS posílá žádost s identifikátorem 0xD7C0 pro zahájení komunikace. Takových identifikátorů existuje velké množství, viz [25]. Server po doručení požadavku zahájí přenos dat identifikátorem 0xD7C3. Po úspěšném přijetí těchto dat klient odpovídá dotazem DNS s identifikátorem 0xD7C4. Server dále pokračuje v zasílání do doby, kdy neodešle celá data. V případě že se ke klientovi nedostane daná část zasílaných dat, tak požadavek na ni může poslat až čtyřikrát. Na závěr klient načte a případně spustí infiltrovaná data na infikovaném zařízení.

Předcházení odhalení je v případě tohoto malwaru poměrně sofistikované. Například klient před kontaktováním serveru C2 zjišťuje přítomnost internetové konektivity pomocí dotazů DNS na legitimní domény. Taktéž se nepokouší kontaktovat server C2, pokud na daném stroji běží nějaký zachytávač paketů [25].

2.2.2 Útok „DNSMessenger“

Další infiltrační útok s příznačným názvem DNSMessenger je zahajován v otevření nebezpečného dokumentu Microsoft Word, který je oběti doručen emailem [12]. Tuto kybernetickou hrozbu lze nalézt ve dvou verzích, které se odlišují například právě i způsobem infiltrace a komunikace přes protokol DNS.



Obrázek 2.3: Znázornění komunikace klienta se serverem C2 pro infiltraci dat [25].

Průběh první verze útoku

První verze tohoto malwaru spočívá v doručení dokumentu MS Word maskujícím se jako oficiální zpráva od společnosti McAfee. Při otevření tohoto dokumentu dojde k infiltrování jedné z částí skriptu Powershell zakódovaného algoritmem Base64 a komprimovaného utilitou `gzip` [12].

Jedna z dalších částí tohoto skriptu je do cílového zařízení infiltrovaná již pomocí protokolu DNS. První iniciační dotaz DNS provede malware takovým způsobem, že náhodně vybere jednu z uložených domén, k ní připojí subdoménu `www` a čeká na odpověď. Pokud přijde očekávána odpověď DNS se záznamem typu TXT obsahující řetězec „www“, tak skript může pokračovat ve vykonávání infiltrace.

Po této prvotní komunikaci již infikovaný systém zažádá o záznam typu TXT s další částí skriptu Powershell, tentokrát však pomocí subdomény `mail`. Vzhledem k velikosti dat je použit po přenos transportní protokol TCP.

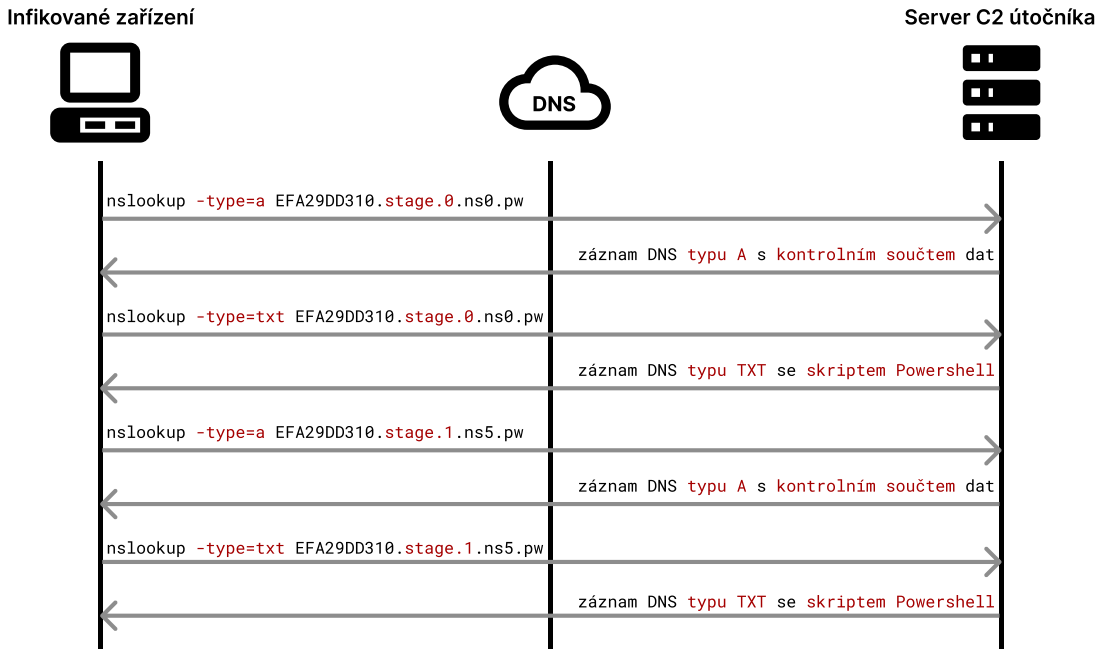
Tato právě infiltrovaná část skriptu dále vykonává komunikaci se serverem C2 pomocí protokolu DNS, ke které je opět vybrána náhodná doména. Pro navázání spojení se serverem C2 je využívána tzv. zpráva SYN, která se skládá z různých identifikátorů pro útočníka a taktéž určením typu zprávy. Ta je zakódována a přiložena jako subdoména, na kterou je poslán dotaz DNS na záznam TXT. Dále jsou již využívány tzv. zprávy MSG, které slouží

k oboustranné komunikaci ve formě zasílání příkazů k provedení a poté jejich výstupů. Pro tyto dotazy je příznačné zakódování výstupu příkazu do šestnáctkové soustavy a oddělování po 30 bajtových subdoménách [12].

Průběh druhé verze útoku

Druhá verze také využívá podobného principu s dokumentem MS Word, pouze se maskuje za jinou organizaci, konkrétně Securities and Exchange Commission (SEC) [13]. Tato nová verze přináší propracovanější způsob infiltrace za pomocí již zmiňovaných záznamů TXT ale nově i A záznamů.

Malware na začátku komunikace se serverem C2 získá sériové číslo systému z prostředí BIOS. To přetransformuje hašovací funkcí MD5 do nové podoby, z níž si vezme prvních deset bajtů. Dále vytvoří počítadlo pro počet infiltračních dotazů DNS a vybere náhodnou doménu ze seznamu. Z toho se složí unikátní doménové jméno ve tvaru: `(hašované sériové číslo).stage.(pořadové číslo dotazu).(náhodně vybraná doména)` [13].



Obrázek 2.4: Příklad infiltrace pomocí nástroje `nslookup.exe`.

Příklad průběhu infiltrační komunikace dle tohoto systému je zobrazen na obrázku 2.4 a je následující [13]:

1. Klient se zprvu zeptá na záznam DNS typu A na danou doménu.
2. Dostává se mu odpovědi s adresou IPv4, kterou reprezentuje jako jedno celé číslo, jež se dále převede do binární podoby.
3. Poté se zeptá na záznam typu TXT na stejnou doménu, kde mu server C2 odpoví zakódovaným obsahem.

4. Obsah klient vloží jako vstup do hašovací funkce MD5 z jejíž výsledku se prvních osm bajtů vloží do algoritmu pro kontrolní součet. Jeho výsledek se pak reprezentuje jako binární číslo, které se porovná s číslem získaným ze záznamu typu A.

5. Pokud se čísla shodují, data z obsahu záznamu TXT získaná pomocí této domény jsou připojena k ostatním a je opět náhodným výběrem zvolena nová doména společně s inkrementací pořadového počítadla. Pokud se neshodují, přenos se opakuje.

6. V případě, že klient obdrží odpověď DNS se záznamem A s adresou IPv4 0.0.0.0, tak je infiltrace úspěšně dokončena.

7. Takto získaná data jsou poté dekódovaná algoritmem Base64 a dekomprimována nástrojem `gzip`.

2.3 Útoky využívající chyb v systémech DNS

Další kategorií útoků, jimiž se tato práce zabývá, jsou hrozby cílené na chyby v knihovnách či aplikacích pracujících s protokolem DNS. Jedná se tedy o takové útoky, které lze realizovat odesláním specificky upraveného paketu DNS. Ten způsobí v lepším případě pouhý pád napadeného systému, v tom horším útočník získá administrátorská práva na napadeném stroji.

2.3.1 Hrozba „SIGRed“

První představovaný útok spočívá ve zneužití chybné implementace v aplikaci `dns.exe` jakožto serveru DNS na operačních systémech Windows Server. Této hrozbě bylo přiděleno unikátní číslo hrozby CVE-2020-1350 [4] a bylo mu uděleno nejvyšší skóre zranitelnosti [36].

Průběh útoku

Výše zmíněná aplikace implementuje funkce na zpracování všech různých odpovědí DNS. Zranitelnost jedné z nich je podrobněji popsána v [36]. Jedná se o funkci určenou ke zpracování odpovědí se záznamem typu SIG, který se dříve využíval v komunikaci DNSSEC, ale posléze byl nahrazen typem RRSIG (viz příloha A.4). Její zásadní vada spočívá v alokaci paměti na hromadě (heap) pro tuto odpověď, která se provádí funkcí `memcpy`. Tato alokační procedura je volána z funkce `RR_AllocateEx`, která bere jako parametry 16bitové registry. Při implementaci se očekávalo, že velikost odpovědi DNS nepřesáhne maximální hodnotu, kterou lze do takových registrů uložit, tedy 65535. Pokud by došlo k pokusu o vložení větší hodnoty než zmiňovaných 65535 do tohoto registru, tak dojde k jeho přetečení a tím i vynulování. To by dále způsobilo alokaci příliš malého prostoru pro odpověď a tudíž i havárii aplikace.

Pro dosažení takovéto velikosti nelze použít transportní protokol UDP, který je limitován na 512 bajtů, respektive 4096 u systémů, které to podporují. Aplikaci `dns.exe` se tak podle specifikace dle [10] pošle paket UDP s příznakem DNS Truncation (TC), který dá serveru najevo, že se má pokusit o opětovné spojení přes protokol TCP.

Jelikož v hlavičce protokolu DNS odesílaném po TCP jsou vyhrazeny pouhé dva bajty pro délku zprávy, tak maximální velikost je tím limitována na 65535 bajtů. Aby však funkce určená k výpočtu alokované paměti mohla přijít s hodnotou vyšší, než je výše zmíněná, tak je nutné využít komprese v paketech DNS. Konkrétně toho lze docílit vložením odkazu

do pole pro jméno signatáře (Signer's Name Field, viz struktura záznamu A.13), který neukazuje korektně na bajt s informací o délce již použitého jména v odpovědi DNS, ale přímo na nějaký znak z tohoto jména. Funkce `Name_PacketNameToCountNameEx`, jež má na starosti výpočet velikosti pole se jménem a dál jej předat zmíněné funkci `RR_AllocateEx`, tak použije ordinální hodnotu znaku, na který odkaz ukazuje, jako velikost nekomprimovaného jména. Společně tak s maximální využitím pole pro samotný podpis (Signature Field) a výsledkem funkce `Name_PacketNameToCountNameEx` dostáváme číslo větší než 65535.

Takto sestavený paket zaslaný na aplikaci `dns.exe` způsobí její pád. Výše zmíněnou praktiku lze aplikovat i na odpovědi se záznamem typu RRSIG, jelikož mají stejnou strukturu a využívají tak totožnou implementaci funkce pro zjištění jejich velikosti. [36]

Schéma útoku, které docílí pád serverové aplikace lze vidět na obrázku 2.5. Postup dle tohoto schématu je následovný [36]:

1. Cílem útočníka je zpočátku dostat záznam NS na jeho doménu do paměti serverové aplikace `dns.exe`. Toho docílí tak, že pošle dotaz DNS na tento záznam z počítače, který bude žádat o rezoluci právě zmíněnou aplikaci `dns.exe`.

2. Jelikož ta odpověď zatím nezná, dle konfigurace přeposílá dotaz na nakonfigurovaný předávací server, například rekurzivní server DNS společnosti Google.

3. Ten mu vrátí odpověď a aplikace `dns.exe` si ji tak uloží do paměti pro budoucí použití.

4. Díky tomuto může pak útočník zaslat ze zařízení žádost o záznam SIG pro jeho škodlivou doménu.

5. Aplikace `dns.exe` se na základě tohoto dotazu přímo zeptá autoritativního serveru DNS pro útočníkovu doménu a dostává škodlivý záznam typu SIG.

Chyba, které tento útok využíval, byla po nahlášení společnosti Microsoft v krátkém čase opravena aktualizací aplikace.

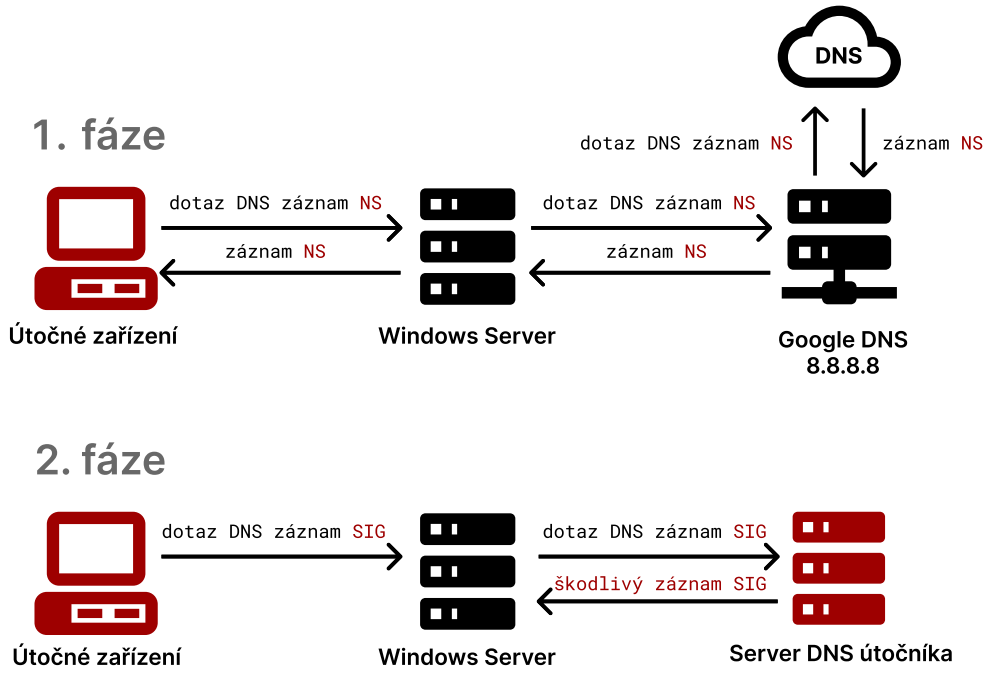
2.3.2 Soubor hrozeb „Ripple20“

Druhým zde zmiňovaným systémem, který je napadnutelný čtyřmi dosud odhalenými způsoby, je softwarová knihovna TCP/IP vyvíjená společností Treck, Inc [27]. V rámci této knihovny se využívá řada funkcí pracující s protokolem DNS. Tyto konkrétní chyby v systému při práci s dotazy DNS dostaly souhrnné označení hrozby CVE-2020-11901 [3].

Průběhy útoků

Soubor „Ripple20“ obsahuje následující hrozby, které jsou popsány níže [27]:

1. První hrozbou, které může útočník využít, je upravení pole `RDLENGTH`, které specifikuje velikost v bajtech pole `RDATA`. Toto pole může tedy útočník upravit na dostatečně malou hodnotu, která způsobí, že vnitřní funkce `tfDnsExpLabelLength` pro výpočet délky jména vrátí malé číslo. To vede k tomu, že se alokuje příliš malé množství pro buffer, do kterého následně funkce `tfDnsLabelToAscii` převede a zapíše doménové jméno z binární podoby do ASCII kódování. Tato funkce tak přistoupí mimo alokovanou paměť a nastane přetečení.



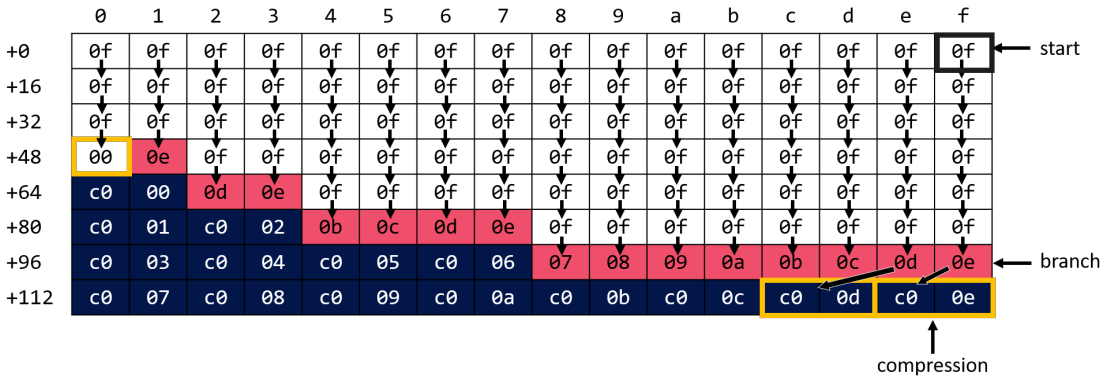
Obrázek 2.5: Schéma útoku na aplikaci `dns.exe` systému Windows Server.

2. V pořadí druhou hrozbou je možnost zneužití faktu, že zmíněná funkce `tfDnsExpLabelLength` pracuje s 16bitovou proměnnou pro ukládání celkové délky jména. Podstatou tohoto útoku je tedy docílit přetečení této proměnné a, podobně jako u předešlé hrozby, následně způsobit havárii systému pomocí funkce `tfDnsLabelToAscii`. Při využití bajtové matice, kterou lze vidět na obrázku 2.6, lze přetečení proměnné docílit principem komprese blíže popsaného v příloze A.3.2. Za pomoci ukazatelů v paketu DNS lze tak zdánlivě paket zvětšit.

   Mějme situaci, kdy útočník umístí do doménového jména odpovědi odkaz na blok 0xF, jak je znázorněno na obrázku 2.6. Funkce `tfDnsExpLabelLength` je navržena tak, aby procházela pouze bajty určující velikost jména. Tudíž pokud narazí na bajt 0xF a přečte z něj hodnotu 0x0F, tak přejde na bajt o řádek níž pod ním. Tam narazí opět na hodnotu 0x0F, atd. Tímto způsobem dojde až k červeně označenému poli s hodnotou 0x0E. Z ní přejde o řádek níž a o jedno doleva z pohledu předchozího bajtu. Tam se setkává s bajtem o hodnotě 1 v prvních dvou bitech, což značí ukazatel. Kam je ukazováno se definuje na zbylých 14 bitech v této dvojici bajtů označených modrou barvou. Funkce tak skočí na bajt 0xE, kde nastíněný proces pokračuje.

   Toto se děje až do doby, kdy se nenarazí na ukončující bajt s hodnotou 0x00. Díky tomuto principu a zvětšením takové matice na 64 sloupců, což je limit z hlediska maximální velikosti bajtu určující velikost jména, lze docílit přetečení 16bitového celého čísla a tudíž pád systému.

3. Dalším možným útokem je opět využít zranitelnost funkce `tfDnsExpLabelLength`. Ve starších verzích této knihovny se ještě nepoužíval ukazatel, který by značil konec



Obrázek 2.6: Struktura paketu DNS s využitím kompresních ukazatelů [27].

paketu DNS. Místo toho funkce iteruje přes všechny bajty dle uváděné délky, dokud nenarazí na bajt „NULL“. Toto může vést k situaci, kdy se funkce bude snažit číst data mimo určený buffer. S touto chybou se lze setkat už pouze v systémech, které pracují se starší verzí této knihovny.

4. Poslední zranitelností, se kterou se lze setkat již spíše na starších, již neudržovaných, systémech je způsob volby hodnoty transakčního ID. To bylo původně číslováno od nuly a bylo postupně inkrementováno. Útočník tak nepotřebuje provádět složité pokusy o odposlechnutí komunikace a vyčtení tohoto identifikátoru, ale může rovnou náhodně nějaký vyzkoušet.

2.4 Techniky ke zvýšení účinnosti útoku

V případě exfiltračních i infiltračních útoků je pro útočníka zásadní volba domény, přes kterou bude tunelování DNS provádět. Lze se setkat s doménovými jmény, které zdánlivě připomínají známé legitimní domény jako například u malwaru Alina POS, nebo naopak sadou náhodně pojmenovaných domén, kterou malware až v průběhu útoku vybere, viz útok DNSMessenger v kapitole 2.2.

Možnou ochranou proti těmto útokům tak může být přímé zablokování těchto konkrétních známých domén. Toto možné zmaření se útočníci snaží minimalizovat využíváním níže zmíněných postupů.

2.4.1 Algoritmus DGA

Jedním z možných přístupů, jak zvýšit pravděpodobnost úspěšnosti útoku ve formě získávání času, je Domain Generation Algorithm (DGA). Jedná se o metodu, která si klade za cíl generovat velké množství pseudo-náhodných doménových jmen, které poté útočník může použít k útoku [20]. Tato doménová jména pak mohou útočníci různě měnit a vyhnout se tak neúspěchu útoku kvůli zablokování určité domény.

Využití algoritmu

Takto vygenerovaná doménová jména pak přidělují svým serverům C2, které se u útoků s využitím protokolu DNS často používají, ale ne jen tam. Domény mohou být generovány na základě určitého hodnoty (v anglickém jazyce označované jako „seed“), která zajistí, že se pro tuto konkrétní hodnotu vygeneruje při každém spuštění stejná sada doménových jmen. Díky tomu generovaní může probíhat na straně oběti i útočníka, a oba mají k dispozici stejný seznam vygenerovaných domén. Útočník tak má přehled o doménových jménech, které malware použije při komunikaci, a může ji tak zaregistrovat pro účely útoku [20].

Příklad, jak může vypadat takové generované doménové jméno, je možno vidět níže.

```
1e7e95ac9287a3ec.com
```

Tato konkrétní ukázka je generována algoritmem DGA pro malware SharkBot [9].

Mnohdy se také využívají tzv. slovníkové DGA (Dictionary DGA). Místo generování domén složených z náhodných znaků se pomocí slovníku skládají názvy, které jsou pro člověka smysluplné.

2.4.2 Monitorování nových domén NOD

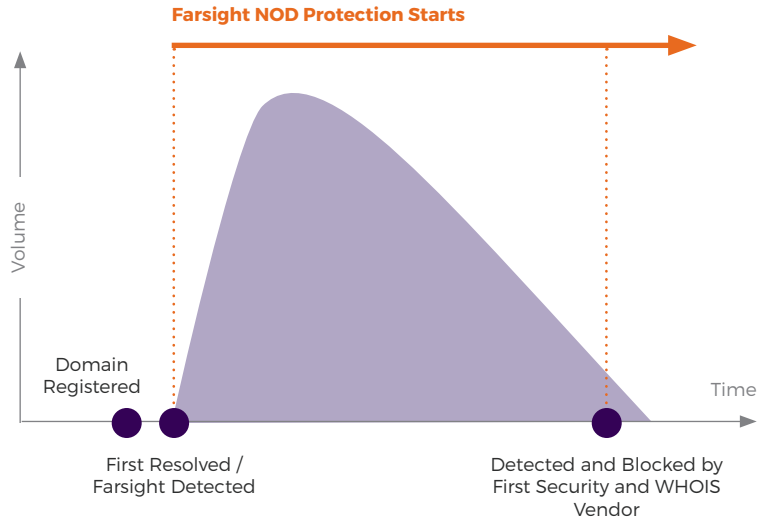
Dle [19] je v následujícím části práce popsána praktika sledování domén společností Farsight Security, Inc. Běžnou praxí v případě doménových jmen je, že jsou registrovány nejméně na jeden rok a obvykle poté obnovovány. Útočníci však využívají domén po velice krátkou dobu, aby mohli realizovat svůj útok, a poté tyto domény opustí.

Newly observed domains (NOD) lze doslova přeložit jako nově zpozorované domény. Pro společnost Farsight Security, Inc. je NOD datová služba, která provozuje globální síť tzv. „senzorů DNS“. Tyto senzory sbírají data z rekurzivních rezolverů a ukládají si komunikaci s autoritativními jmennými servery. Tato data pak ukládají do databáze DNSDB.

Pokud se v dané databázi objeví doména, která tam doposud nebyla, NOD vytvoří a rozešle zprávu, v které informuje o prvním použití daného doménového jména. Tato skutečnost však nemá žádnou souvislost s registrací takové domény. NOD systém detekuje pouze první dotaz na danou doménu, tudíž její registrace může být i měsíce před prvním použitím. Dle údajů platným k prvnímu čtvrtletí roku 2018 se v průměru každou sekundu detekují tímto způsobem více jak dvě domény druhého řádu. Průběh „života“ takové domény lze vidět na obrázku 2.7.

NOD dále tyto zpozorované domény sleduje u serverů domén nejvyššího řádu, které na tyto sledované domény delegují pomocí záznamů NS. Taktéž o nich sbírá informace u samotných autoritativních jmenných serverů těchto domén a kontroluje jejich přítomnost na známých černých listinách. Na všech těchto místech probíhá monitorování v průběhů sedmi dnů od prvního dotazu DNS s různými časovými intervaly.

Doména je v průběhu tohoto sledování považována za „mrtvou“, pokud se při pokusu o rezoluci této domény vrací informace o její neexistenci, či se objeví v zmíněných černých listinách. V tomto případě probíhá dál monitorování tohoto doménového jména, ale již tato doména nemůže přejít do „živého“ stavu [19].



Obrázek 2.7: Zjednodušený graf monitorování domény využívané k útokům [2].

2.5 Shrnutí

Výše v této kapitole byly představeny konkrétní příklady kybernetických hrozeb v komunikaci DNS.

V rámci této práce byly hrozby rozčleněny do čtyř kategorií, jak je uvedeno v tabulce 2.1: exfiltrační útoky, infiltrační útoky, útoky využívající chyb v systémech DNS a techniky ke zvýšení účinnosti útoku. U každé z těchto čtyř skupin byl na začátku uveden obecný princip, dle kterého se hrozby do ní spadající řídí.

Dále byly ke každé kategorii uvedeny konkrétní příklady hrozeb, které se v internetu vyskytovaly či dodnes vyskytují. Ke každému z nich byl pak vždy uveden krátký popis útoku a jeho samotný průběh.

V této tabulce nejsou záměrně uvedeny příklady z poslední skupiny hrozeb, tedy technik ke zvýšení účinnosti útoku. Důvodem je to, že nejsou plnohodnotnou hrozbou, která zneužívá systém DNS. Jedná se, jak bylo zmíněno výše, o nástroje, které pouze pomáhají útočníkům zefektivnit jejich útok za pomocí různých technik spojených se systémem DNS.

Název	**Typ**	**Využívané záznamy**	**Stavovost**	**Účel**	**Zacílenost**
ProjectSauron	exfiltrace	–	ano	získání citlivých dat	vládní organizace, armády, výzkumná centra (Rusko, Írán, Rwanda)
Alina POS	exfiltrace	–	ano	odcizení údajů platebních karet	prodejní zařízení – např. platební terminály
InvisiMole	infiltrace	NULL, AAAA	ano	pořizování záznamu z webkamery a mikrofonu, zjištění polohy, získání citlivých dokumentů	organizace spojené s diplomacií či armádou (východní Evropa)
DNSMessenger	infiltrace	TXT, A	ano	infiltrace škodlivého skriptu Powershell	široká veřejnost (doručování škodlivého souboru emailem)
SIGRed	zneužití chyby	SIG	ne	pád napadeného systému nebo získání administrátorských práv	uživatelé serverů DNS v operačním systému Windows Server
Ripple20	zneužití chyby	libovolný	ne	pád napadeného systému nebo získání administrátorských práv	uživatelé knihovny TCP/IP od společnosti Treck, Inc

Tabulka 2.1: Přehled zmíněných hrozeb v systému DNS.

Kapitola 3

Přehled nástrojů a datových sad

V předešlé kapitole byly popsány jednotlivé hrozby zneužívající systém DNS. V této kapitole je provedena rešerše datových sad a nástrojů pro získání přehledu o hrozbách v systému DNS. Rešerše také slouží k lepšímu pochopení průběhu útoků a k návrhu jejich detekce.

V rámci této kapitoly jsou zprvu popsány výsledky rešerše datových sad, který souvisí s hrozbami v komunikaci DNS. Posléze jsou sepsány způsoby, jakými se povedlo implementovat či využít již existujícího nástroje pro simulaci jednotlivých útoků. Na závěr následuje dokumentace vytvořených datových sad pro účely testování implementovaného detekčního nástroje v rámci této práce.

3.1 Rešerše datových sad souvisejících se systémem DNS

V rámci získávání informací o jednotlivých hrozbách a hledaní nástrojů, které je implementují, byla provedena rešerše dostupných datových sad, jejíž dokumentací se zabývá tato kapitola níže.

Taktéž byly provedeny opakované pokusy získat datové sady přímo od týmů, které dané hrozby rozkryly, ale bohužel ve většině případu skončily tyto pokusy neúspěšně.

3.1.1 Datové sady s útoky na systém DNS

První nalezenou datovou sadou je soubor síťových dat v rámci simulace exfiltrace od Kanadského institutu pro kyberbezpečnost [30]. Sada vydaná v roce 2021 obsahuje velké množství zachyceného síťového provozu DNS využitého k exfiltraci různých typů souborů o rozlišných velikostech. Původní účel této datové sady je vytvoření velkého množství síťového provozu pro jeho následné experimentovaní s detekčním systémem na bázi umělé inteligence. Co se použitelnosti sady vzhledem k této práci týče, tak při jejím vytváření byla použita volně dostupná obecná implementace exfiltrace dat pomocí systému DNS. Z toho vyplývá, že v sadě nebyla nalezena žádná data pro konkrétní výše popisované hrozby.

Bohužel se nepovedlo v žádných dalších datových sadách nalézt komunikaci DNS výše uvedených hrozeb. Jedním z možných důvodu může být fakt, že většina zmiňovaných útoků cílila především na konkrétní malé množství velkých organizací a tudíž její záchyt není veřejně dostupný nebo neexistuje.

V případě datové sady z americké University of Southern California z roku 2013 [37] byl však nalezen jiný doposud nerozebíraný útok. Jedná se o tzv. amplifikační útok DoS pomocí systému DNS, který se umělým vytvářením paketů DNS se zdrojovou adresou oběti dotazuje na velké záznamy, které následně oběť zahltí. V rámci datové sady je zachycena komuni-

kace trvající přibližně 10 minut vytvořená mezi dvěma pracovníky univerzity v rozlišných lokalitách. Z hlediska této práce však nenabízí téměř žádný přínos, jelikož útoky typu DoS, byť na systém DNS, jsou záležitostí síťové vrstvy TCP/IP a tudíž nejsou předmětem této práce.

3.1.2 Enumerační datové sady s doménovými jmény

Další prostudovanou datovou sadou byla opět jedna ze sad Kanadského institutu pro kyberbezpečnost taktéž z roku 2021 [29]. Jejím účelem bylo vytvoření dostatečného množství doménových jmen, které jsou následně vstupem do detekčního nástroje postaveného na umělé inteligenci. Takový nástroj měl z doménového jména poznat případnou škodlivost domény a i typ útoku.

Doménová jména jsou v této datové sadě rozdělená do čtyř skupin: neškodné, malware, spamové a phishing domény. Ke každému doménovému jménu je vedeno velké množství dílčích informací jako například délka, zastoupení čísel ve jméně, počty jednotlivých znaků ve jméně atd.

Bohužel se však v této sadě nenachází žádná z domén, která byla využita u útoků specifikovaných výše v této kapitole.

3.1.3 Datové sady se síťovým provozem obsahující malware

Při hledání vhodných datových sad byly taktéž detailně prostudovány souhrnné sady [34], [33] a blogy[1] obsahující různé druhy malware, tedy nikoliv pouze útoky na systém DNS. Tyto sady obsahují zachycenou komunikaci pro daný druh malware a v některých případech i jeho implementaci.

V případě [33] se podařilo narazit na záchyt síťové komunikace jedné z verzí hrozby Alina POS v podobě souboru typu PCAP. Bohužel však zmíněný soubor neobsahoval vzhledem k této práci sledovanou komunikaci DNS.

U sady [34] a zmiňovaného blogu nebyla nalezena ani jedna ze zmiňovaných hrozeb v rámci systému DNS a ani žádná jiná relevantní.

3.1.4 Datové sady se zaměřením na domény DGA

K algoritmu DGA, který mnohé malware nástroje využívají k vytváření unikátního doménového jména pro servery C2, jak je blíže popsáno v kapitole 2.4, se podařilo najít poměrně obsáhlou datovou sadu HYDRA [14].

Tato datová sada obsahuje více jak 90 milionů domén druhého řádu z různých doménových rodin. Sada je v podobě velkého souboru typu CSV, kde pro každou doménu jsou uvedeny další specifické informace. Mezi takové údaje patří například název rodiny, pod kterou spadá, její délka, počet číslic v doménovém jméně a mnoho dalších.

3.1.5 Přehled datových sad

V následující tabulce 3.1 je možné vidět přehled prostudovaných datových sad. Ke každé ze sad je uveden jejich název, autor a stručně popsaný obsah.

[1] viz `https://www.malware-traffic-analysis.net/`

Název	Autor	Obsah	Délka	Počet paketů DNS	Typ souboru/ů
CIC-Bell-DNS-EXF-2021 Dataset	Canadian Institute for Cybersecurity	exfiltrační útoky pomocí obecného algoritmu	270,8 MB (80 h 20 m)	1 327 387	PCAP a CSV
DoS_DNS_ amplification-20130617	University Of Southern California	záchyt DoS útoku na systém DNS	5,4 GB (33 minut)	–	PCAP
CIC-Bell-DNS 2021 Dataset	Canadian Institute for Cybersecurity	enumerace různých doménových jmen	933,9 MB (637 h 28 m)	7 246 523	PCAP a CSV
HYDRA dataset	Fran Casino a další	doménová jména vytvořená pomocí DGA	23,7 GB	–	CSV

Tabulka 3.1: Přehled zmíněných datových sad.

3.2 Nástroje simulující jednotlivé útoky

V této části kapitoly je blíže popsán způsob, jakým jsou jednotlivé útoky implementovány pro jejich následnou simulaci. Daný popis obsahuje pouze ty hrozby, ke kterým se podařilo dohledat samotný nástroj či dostatečné množství informací dovolující je věrohodně odsimulovat.

3.2.1 Malware Alina POS

Za pomocí získané reálné ukázky jednoho z dotazů DNS tohoto útoku a detailního popisu průběhu, viz kapitola 2.1, byl vytvořen simulační nástroj.

Ten na začátku vygeneruje šestimístný identifikátor oběti, vybere náhodný název zařízení a název procesu z připravených seznamů, vytvoří náhodné číslo platební karty s její dobou expirace a na závěr vygeneruje sedm náhodných čísel. Všechny tyto atributy pak následně poskládá v zakódované podobě za sebe s danými oddělovači, připojí takto vytvořený řetězec k bázové simulační doméně a pošle na takový dotaz požadavek DNS na záznam typu A.

Připravený autoritativní server BIND 9 pro danou bázovou doménu tento požadavek zpracuje a vrátí na ní odpověď s adresou 127.0.0.1.

3.2.2 Útok DNSMessenger

Pro simulaci tohoto útoku byl využit interní nástroj společnosti Infoblox, který byl poskytnut pro účely této práce.

Ve webovém prostředí pro simulaci tohoto útoku se na začátku zadá příslušný skript Powershell, který chceme infiltrovat. Dále se uvede operačním systém, na který chceme skript dostat. Poté je vygenerován skript, který vložíme do konzole na testovacím systému a zahájí se tím tunelování pomocí záznamů DNS typu TXT.

3.2.3 Hrozba SIGRed

Tato hrozba je implementována s využitím nástroje z veřejného repositáře GitHub [2].

Na začátku se spustí zmíněný nástroj na zařízení v lokální testovací síti se vstupním parametrem bázové domény, například `skodliva-domena.cz`. Tento nástroj pak vyčkává souběžně na UDP a TCP spojení na portu 53 pro komunikaci DNS.

Z jiného zařízení je následně proveden požadavek na záznam typu SIG programem `nslookup` na danou bázovou doménu s připojenou subdoménou „9“, tedy v tomto případě

[2] viz `https://github.com/maxpl0it/CVE-2020-1350-DoS`

`9.skodliva-domena.cz`. Takový dotaz je přeposlán na nakonfigurovaný server DNS pro toto zařízení, kterým je v případě simulace server BIND 9.

Tento server simuluje úlohu aplikace `dns.exe` na systémech Windows Server, který je obětí tohoto útoku. Pro zjednodušení simulace testovací server DNS nevykonává rezoluci přeposláním na rekurzivní server DNS v internetu, ale rovnou dotaz přesměruje na škodlivé zařízení, na kterém běží již zmiňovaný nástroj.

Tento nástroj serveru prvně vrací odpověď DNS, v které žádá opětovné spojení pomocí transportního protokolu TCP. V další komunikaci již přeposílá paket, který by v případě starší verze aplikace `dns.exe` způsobil její pád nebo zmocnění se serveru.

3.2.4 Algoritmy DGA

K simulaci této techniky byl využit konkrétní algoritmus DGA pro malware s názvem SharkBot.

S použitím implementace programu, který generuje jednotlivá doménová jména dle reverzního postupu algoritmu DGA[3], byl naimplementován nástroj, který pro každé vytvořené doménové jméno vytvoří a zašle dotaz DNS na záznam A pro dané jméno.

Jelikož je pro tento konkrétní algoritmus vstupem nějaké datum, tak na začátku nástroj takové datum vygeneruje v intervalu od 1. 1. 2020 do 31. 12. 2022.

3.3 Vytvořené datové sady pro konkrétní útoky

Pro následné testování detekčního nástroje byly vytvořeny datové sady s jednotlivými útoky. Pro vytváření sad byly zvoleny takové útoky, které bylo možné simulovat. Vytvořené sady vychází z dále popsaného obecného principu.

Každá dále zmíněná datová sada byla vytvořena na základě záchytů paketů DNS pomocí programu Wireshark. Délka vytvořených sad je pro každou z nich okolo 10 minut záchytu. Výjimkou je sada s různými druhy algoritmů DGA, která je zhruba o polovinu kratší s výrazně vyšší frekvencí paketů DNS. V průběhu záchytu paketů byl konkrétní útok proveden vždy minimálně jednou. Taktéž byl každý útok doplněn o generovanou komunikaci běžných dotazů v náhodných intervalech.

Níže se nachází popis jednotlivých specifik datových sad pro konkrétní útoky. V tabulce 3.2 je pak možné vidět přehled datových sad pro jednotlivé útoky a bližší informace k nim, konkrétně délku záchytu, velikost souboru, počet paketů DNS a číslo paketu či paketů příslušících danému útoku.

3.3.1 Specifika jednotlivých datových sad

V rámci záchytu paketů pro datovou sadu k útoku Alina POS byla daná hrozba provedena třikrát. Každý vygenerovaný útok má jinou podobu dle všech možných forem, co byly v rámci rešerše nalezeny. K útokům DNSMessenger a SIGRed byly vytvořeny datové sady s jedním průběhem daného útoku. Pro generování datové sady pro algoritmy DGA byl použit specifický postup. V pravidelných intervalech byly zasílány pakety DNS, z nichž každý třetí je vygenerován pomocí různého algoritmu DGA. Ukázková doménová jména byla převzata z repozitáře s doménami DGA[3]. Tato datová sada tak nabízí širokou škálu domén generovaných různými algoritmy pro následné testování.

[3] viz `https://github.com/baderj/domain_generation_algorithms`

Obsah	Délka (velikost)	Počet paketů DNS	Rozmezí útoku
malware Alina POS	65 kB, 11 m 15 s	496	159-160, 220-221, 427-428
útok DNSMessenger	60 kB, 10 m 30 s	442	101-112
hrozba SIGRed	195 kB, 11 m 4 s	452	87-133
algoritmy DGA	287kB, 5 m 34 s	2454	každý 5. a 6. paket

Tabulka 3.2: Přehled vytvořených datových sad s vygenerovanými útoky.

3.4 Shrnutí

Na začátku této kapitoly byl zdokumentován proces rešerše datových sad s útoky DNS s uvedením konkrétních sad a zhodnocením jejich možného přínosu. Závěrem této rešerše je fakt, že se nepovedlo najít téměř žádné soubory obsahující komunikaci DNS alespoň jednoho z útoku z kapitoly 2.

Z tohoto důvodu byla velmi limitující implementace simulačních nástrojů těchto útoků. Především bylo operováno se závěry reverzních inženýrství jednotlivých hrozeb. Existující nástroje byly získány pouze k hrozbám DNSMessenger, SIGRed a algoritmům DGA. K malwaru Alina POS byl nástroj vytvořen v rámci této práce. K ostatním útokům se, bohužel, nepodařilo najít dostatečné množství dat k jejich simulaci.

Závěrem byly popsány vytvořené datové sady, která mají sloužit k otestování implementovaného detekčního nástroje. K vytvoření těchto sad byly využity zmiňované simulační nástroje a jejich účelem je především zastoupení nenalezených veřejných datových sad.

Kapitola 4

Návrh monitorování a detekce

V rámci předešlých kapitol byly blíže specifikovány konkrétní hrozby v komunikaci DNS s rozdělením do patřičných skupin a provedeny rešerše nástrojů a datových sad. Aby bylo možné přistoupit k hlavnímu cíli práce, kterým je detekce těchto hrozeb, je nutné prozkoumat systémy monitorování komunikace DNS a způsoby detekce pro vybrané útoky.

K monitorování sítí existuje mnoho nástrojů, kterými jsou například logování událostí konkrétních síťových aplikací, např. BIND 9 [5], analýza síťových toků NetFlow či její rozšíření IPFIX [8] nebo detekce signatur v zařízení IDS [6]. Každý z těchto monitorovacích systémů nabízí zachytávání jiných informací a rozdílný způsob jejich sběru. V této kapitole postupně porovnáme všechny čtyři zmíněné systémy vzhledem k jejich použitelnosti při budoucí detekci hrozeb v komunikaci DNS.

Jedním z cílů této kapitoly je zhodnotit jednotlivé systémy monitorování DNS, zda poskytují dostatečné informace k identifikaci všech simulovaných hrozeb. Dalším cílem je navrhnout specifický způsob detekce pro každý ze simulovaných útoků.

U každého monitorovacího systému je na začátku uveden jeho krátký popis a bližší specifikace. Dále je popsána konfigurace tohoto systému v testovacím prostředí a příklad s popisem jeho výstupu. Na konci je zhodnoceno, k jakým hrozbám je daný systém dostačující pro jejich detekci či jsou případně uvedeny důvody, proč vhodný není.

V závěru této kapitoly je dokumentován přesný postup detekce pro vybrané útoky. Tyto návrhy jsou pak dále implementovány a nasazovány do detekčního nástroje, viz kapitola 5.

4.1 Logování událostí na serveru DNS

Jednou z možností, jak monitorovat komunikaci DNS, je logovat přijaté dotazy DNS, viz příloha A.2.1. Pro konkrétní testování monitorování byla využita implementace serverové aplikace BIND 9.

Síťová aplikace BIND 9 nabízí široké možnosti nastavení logování. Logované informace jsou rozřazeny do různých kategorii jako například `security` informující o (ne)povolených dotazech DNS na tento server, `network` popisující síťové operace či `general` pro obecné informace o serveru BIND. Nejdůležitější kategorií z hlediska této práce je kategorie `queries`, která zaznamenává přijaté dotazy DNS a další specifikace k nim.

Každá z kategorií může být přiřazena do tzv. kanálů (channels), které udávají způsob logování, jako například ukládání do souboru, výpis na standardní výstup či použití protokolu Syslog. Taktéž se v těchto kanálech definuje závažnost zpráv, která se má logovat podle různých přednastavených úrovní. Rovněž se zde nastavuje, jaké se mají se samotnou

zprávou vytisknout dodatečné informace, např. časová značka, název kategorie atd. Více podrobností o způsobech a konfiguraci logování lze nalézt v manuálu aplikace [5].

4.1.1 Konfigurace a výstup monitorování

Po zprovoznění síťové aplikace BIND 9 na testovacím zařízení a jejím nastavení na „cache-only“ server bylo nakonfigurováno logování následovně:

```
logging {
    channel queries_log {
        file "/var/named/bind/queries.log";
        print-time yes;
        severity info;
    };
    category queries { queries_log; };
};
```

Jak je možné vidět z části konfiguračního souboru `named.conf`, pro logování byl zřízen kanál, který ukládá informace o přijatých dotazech DNS do souboru. K těmto informacím přikládá časovou značku.

Po zaslání dotazu na tento server za pomocí nástroje `nslookup` na zjištění záznam typu A pro doménové jméno `www.fit.vutbr.cz` dostáváme následující logovací záznam ve zmíněném souboru:

```
23-Dec-2022 17:21:44.681 client @0x7fcf28045548 127.0.0.1 #34025 (www.fit.v
utbr.cz): query: www.fit.vutbr.cz IN A + (127.0.0.1)
```

První část záznamu obsahuje čas přijetí dotazu DNS. Za ní následuje unikátní identifikátor klienta, v tomto případě `@0x7fcf28045548`. Dále můžeme vidět adresu IP a zdrojový port zaslaného dotazu. Na konci za označením `query` se nachází informace, že se jedná o dotaz na doménové jméno `www.fit.vutbr.cz` a že je žádán záznam třídy `IN` typu `A`. Znak `+` značí, že bylo žádáno použití rekurze u rezoluce. Tento systém však bohužel nenabízí záznam obsahující odpověď na daný dotaz.

4.1.2 Použitelnost monitorování při detekci

Malware „Alina POS“

Vzhledem k zakódování veškerých dat do doménového jména u útoku Alina POS, viz kapitola 2.1, je tento monitorovací systém zcela dostačující pro detekci tohoto útoku. Dle logovacího záznamu dokážeme dekódovat data z doménového jména a určit oběť útoku.

Na příkladu níže je uvedeno, jak vypadá simulace zmíněného útoku Alina POS zachycená tímto způsobem monitorování. Zvýrazněná část ukazuje na nejpodstatnější informaci z tohoto záznamu. Jedná se o doménové jméno, ve kterém jsou zakódované platební údaje.

```
22-Feb-2023 22:28:03.786 client @0x7fd8fc004de8 10.0.0.101#55028 (yevH-uHIk
Ojr6eHl7Ozj6e_QmJCQzMXE3s7c2MLF2d6Ez9LPkJCYn52bk52ZmZ.qSkpyfm5-Sl5uZmpmYmpi
fk5qcmJKbmw==.utocnikova-domena.cz): query: yevH-uHIkOjr6eHl7Ozj6e_QmJCQzMX
E3s7c2MLF2d6Ez9LPkJCYn52bk52ZmZ.qSkpyfm5-Sl5uZmpmYmpifk5qcmJKbmw.utocnikova
-domena.cz IN A +E(0)K (10.0.0.10)
```

Útok „DNSMessenger“

Naproti tomu u útoku DNSMessenger, viz kapitola 2.2, již tento systém nepostačí. Jak je možné vidět z ukázky logovacího záznamu zachyceného po spuštění simulace tohoto útoku, tento způsob monitorování neposkytuje dostatečné množství informací. Aby bylo možné útok plně identifikovat, je zapotřebí nahlédnout do větší hloubky paketu DNS a dostat se i k záznamu TXT. To logování nenabízí. Níže jsou uvedeny záznamy vytvořené při zaslání dotazu na infiltrační domény.

```
23-Feb-2023 21:01:56.962 client @0x7f459c0031f8 10.0.0.101#49407 (285a1.sta
ge.0.dnsmess-sim.com): query: 285a1.stage.0.dnsmess-sim.com IN TXT + (10.0.
0.10)
```

```
23-Feb-2023 21:01:57.890 client @0x7f45a01638a8 10.0.0.101#32951 (285a1.sta
ge.0.dnsmess-sim.com): query: 285a1.stage.0.dnsmess-sim.com IN TXT +T (10.0
.0.10)
```

Detekce pomocí kontroly doménového jména či shody s určitými subdoménami není dostatečná. Z ukázky výše je patrné, že daný simulační nástroj od společnosti Infoblox generuje jiné subdomény pro provedení útoku.

Také je vidět, že během útoku přijdou dva dotazy ze zařízení oběti. V druhém případě je již použito spojení pomocí transportního protokolu TCP, značeno zvýrazněným příznakem `T`.

Hrozba „SIGRed“

Také hrozba SIGRed, viz kapitola 2.3, není detekovatelná pomocí logování aplikací BIND 9. Informace o doménovém jméně a typu záznamu, na který je kladen dotaz DNS je sice relevantní informací, avšak není dostačující k určení, že se jedná o hrozbu. K tomu je nutné znát další parametry z paketu DNS. Příklad logovacího záznamu ze simulace tohoto útoku je uveden níže:

```
23-Feb-2023 20:32:29.491 client @0x7f4590002ce8 10.0.0.101#56657 (9.skodliv
a-domena.cz): query: 9.skodliva-domena.cz IN SIG + (10.0.0.10)
```

Tento logovací záznam byl vytvořen po provedení dotazu na záznam typu SIG ze zařízení uvnitř monitorované sítě.

Algoritmy DGA

Pro útoky využívající algoritmus DGA, viz kapitola 2.4, je logování plně vyhovujícím. Při žádost o rezoluci generované domény nám stačí vědět pouze zmíněné doménové jméno, což nám tento systém poskytuje, viz níže:

```
23-Feb-2023 21:25:25.268 client @0x7f45a0174718 10.0.0.101#54861 (ad954fca9
3c10fc0.xyz): query: ad954fca93c10fc0.xyz IN A +E(0) (10.0.0.10)
```

4.2 Monitorování NetFlow/IPFIX

Další možnou možností detekce útoků DNS je využití monitorování NetFlow, respektive IPFIX. Jeho základním principem je agregování zachycené síťové komunikace do tzv. toků.

Tok lze definovat jako sadu paketů IP s určitými společnými vlastnostmi, které procházejí přes sledovaný bod v síti v daném čase. Společnými vlastnostmi mohou být například zdrojová a cílová IP adresa, čísla portů či data z aplikačních vrstev, tedy z DNS [24].

V rámci této práce využíváme standardizovanou verzi IPFIX (z anglického IP Flow Information eXport), která vychází z proprietárního protokolu NetFlow v9 od společnosti Cisco. Důvodem pro zvolení právě této verze oproti NetFlow v9 je především otevřenost jejího standardu a možnosti využití v případě bezpečnostního monitorování.

Pro další informace ohledně monitorování NetFlow/IPFIX je možné nahlédnout do popisu [24] či případně přímo do standardu RFC 7011 [8].

4.2.1 Konfigurace a výstup monitorování

Pro záchyt dat a vytvoření záznamů toků IPFIX byla využita sonda Flowmon[1]. Tyto záznamy je možné exportovat do souboru typu CSV. Pro monitorování komunikace DNS se ke standardním elementům toku jako jsou adresy IP, čísla portů či velikost paketů přidávají i speciální elementy nadefinované společností Flowmon Networks, jejichž název začíná slovem `INVEA`.

Níže je uveden příklad záznamu toku po provedení rezoluce doménového jména www.vutbr.cz vůči serveru DNS společnosti Google.

```
BYTES:                          132
PACKETS:                        2
START_SEC:                      2023-01-06 16:14:28.270433589
END_SEC:                        2023-01-06 16:14:28.270438418
L4_PROTO:                       17 (UDP)
BYTES_A:                        58
BYTES_B:                        74
L3_IPV4_SRC:                    150.150.150.10
L3_IPV4_DST:                    8.8.8.8
L4_PORT_SRC:                    58075
L4_PORT_DST:                    53
INVEA_DNS_ID:                   36112
INVEA_DNS_QUESTIONS_COUNT:      1
INVEA_DNS_QTYPE:                1  (A)
INVEA_DNS_QCLASS:               1  (IN)
INVEA_DNS_QNAME:                www.vutbr.cz
INVEA_DNS_CRR_TYPE:             1  (A)
INVEA_DNS_CRR_CLASS:            1  (IN)
INVEA_DNS_CRR_TTL:              299
INVEA_DNS_CRR_NAME:             www.vutbr.cz
INVEA_DNS_CRR_RDATA_LEN:        4
INVEA_DNS_CRR_RDATA:            147.229.2.90
INVEA_DNS_FLAGS_CODES_REQUEST:  256
INVEA_DNS_FLAGS_CODES_RESPONSE: 33152
INVEA_DNS_ANSWREC_COUNT_RESPONSE: 1
```

[1] viz `https://www.flowmon.com/cs/products/appliances/probe`

Výše uvedený záznam se skládá z mnoha políček, která mají následující význam:

`BYTES` – je celkový počet bajtů datagramů IP v rámci toku,

`PACKETS` – je počet paketů, ze kterých se tok skládá,

`START_SEC` – je čas začátku toku,

`END_SEC` – je čas konce toku,

`L4_PROTO` – určuje transportní protokol toku,

`BYTES_A` – je počet bajtů ve směru od zdrojové k cílové adrese – tedy dotazu DNS,

`BYTES_B` – je počet bajtů ve směru od cílové ke zdrojové adrese – tedy odpovědi DNS,

`L3_IPV4_SRC` – je zdrojová adresa IP dotazu DNS,

`L3_IPV4_DST` – je cílová adresa IP dotazu DNS,

`L4_PORT_SRC` – je zdrojový port dotazu DNS,

`L4_PORT_DST` – je cílový port dotazu DNS,

`INVEA_DNS_ID` – je ID transakce v rámci komunikace DNS,

`INVEA_DNS_QUESTIONS_COUNT` – je počet dotazů v rámci paketu DNS,

`INVEA_DNS_QTYPE` – určuje typ záznamu, který je požadován,

`INVEA_DNS_QCLASS` – určuje třídu záznamu, který je požadován,

`INVEA_DNS_QNAME` – je doménové jméno, na které je kladen dotaz,

`INVEA_DNS_CRR_TYPE` – je typ záznamu v rámci odpovědi,

`INVEA_DNS_CRR_CLASS` – je třída záznamu v rámci odpovědi,

`INVEA_DNS_CRR_TTL` – je hodnota TTL záznamu v rámci odpovědi,

`INVEA_DNS_CRR_NAME` – je doménové jméno, pro které je poskytován záznam,

`INVEA_DNS_CRR_RDATA_LEN` – je velikost záznamu,

`INVEA_DNS_CRR_RDATA` – je obsah záznamu,

`INVEA_DNS_FLAGS_CODES_REQUEST` – je nastavení příznaků při dotazu DNS,

`INVEA_DNS_FLAGS_CODES_RESPONSE` – je nastavení příznaků při odpovědi DNS,

`INVEA_DNS_ANSWREC_COUNT_RESPONSE` – je počet odpovědí v rámci paketu DNS.

Získaný záznam ze sondy obsahuje více parametrů o daném toku, avšak pro účely této práce byly vybrány pouze relevantní.

Z výše uvedeného záznamu je patrné, že při zpracování síťových dat došlo k agregaci paketů DNS s dotazem a odpovědí. Dále lze vyčíst, kdy daná komunikace proběhla, jaký byl transportní protokol a jaké jsou velikosti paketů s dotazem a odpovědí DNS.

Taktéž lze určit, že dotaz směřoval na rekurzivní server DNS na adrese 8.8.8.8, kde byl žádán záznam typu A doménového jména www.vutbr.cz. Na tento dotaz byla zaslána odpověď se záznamem o velikosti čtyři bajty s adresou 147.229.2.90. Ze záznamu toku je také možné vyčíst nastavení příznaků.

4.2.2 Použitelnost monitorování při detekci

Malware „Alina POS“ a algoritmy DGA

Jak je z příkladu patrné, tento typ monitorování obsahuje veškeré informace, které obsahoval i logovací systém aplikace BIND 9. Z toho vyplývá, že v případě útoku Alina POS, viz kapitola 2.1, a algoritmu DGA, viz kapitola 2.4, je monitorování IPFIX plně dostačující.

Následující příklad ukazuje schopnost tohoto systému detekovat útok Alina POS. Oproti monitorovaní za pomocí logování přináší další informace jako například záznam typu A v rámci odpovědi na dotaz.

```
START_SEC:                      2023-02-24 09:07:42.380631866
INVEA_DNS_QTYPE:                1 (A)
INVEA_DNS_QCLASS:               1 (IN)
INVEA_DNS_QNAME:                yevH-uHIkOjr6eHl7Ozj6e_QmJCQzMXE3s7c2ML
                                F2d6Ez9LPkJCYn52bk52ZmZ.qSkpyfm5-Sl5uZm
                                pmYmpifk5qcmJKbmw.utocnikova-domena.cz
INVEA_DNS_CRR_TYPE:             1 (A)
INVEA_DNS_CRR_CLASS:            1 (IN)
INVEA_DNS_CRR_RDATA_LEN:        4
INVEA_DNS_CRR_RDATA:            127.0.0.1
```

Zde je možné vidět zachycení dotazu na doménu generovanou algoritmem Sharkbot. U tohoto typu hrozeb lze tedy říci, že nám monitorovaní IPFIX podává ekvivalentní počet relevantních informací k detekci jako logování na serveru BIND 9.

```
START_SEC:                      2023-01-06 16:14:28.270433589
INVEA_DNS_QTYPE:                1 (A)
INVEA_DNS_QCLASS:               1 (IN)
INVEA_DNS_QNAME:                ad954fca93c10fc0.xyz
```

Útok „DNSMessenger“ a hrozba „SIGRed“

V předešlé části kapitoly při popisu logování serveru DNS bylo zmíněno, že pro detekci útoků DNSMessenger a SIGRed by bylo nutné nahlédnout přímo do obsahu paketu DNS. Monitorování IPFIX v dané konfiguraci to nabízí a je tedy možné přímo vidět obsah jednotlivých záznamů či zjistit velikost paketu DNS. Lze tedy říci, že za pomocí monitorování IPFIX je možné detekovat i tyto hrozby.

Další příklad ukazuje monitorování útoku DNSMessenger pomocí IPFIX. V rámci příkladu ukážeme pouze relevantní políčka ze zachyceného záznamu:

```
START_SEC:                      2023-02-24 09:09:01.339774788
L4_PROTO:                       17 (UDP)
INVEA_DNS_QTYPE:                16 (TXT)
INVEA_DNS_QCLASS:               1  (IN)
INVEA_DNS_QNAME:                285a1.stage.0.dnsmess-sim.com
INVEA_DNS_FLAGS_CODES_RESPONSE: 33696 (Truncated)

START_SEC:                      2023-02-24 09:09:02.260329077
L4_PROTO:                       6  (TCP)
```

```
INVEA_DNS_QTYPE:                    16 (TXT)
INVEA_DNS_QCLASS:                   1  (IN)
INVEA_DNS_QNAME:                    285a1.stage.0.dnsmess-sim.com
INVEA_DNS_CRR_TYPE:                 16 (TXT)
INVEA_DNS_CRR_CLASS:                1  (IN)
INVEA_DNS_CRR_RDATA_LEN:            64
INVEA_DNS_CRR_RDATA:                H4sIAAAAAAAACA61T/W/aMBD9V05RJpLRhI9u1dY
                                    o0qCUqdJEN2D9oqgK5gBviZ3ahkIR//suH7S...
```

Podobně jako u logovacího záznamu jsou zachyceny dva toky. Oba se dotazují na záznam typu TXT, avšak v prvním případě je navázáno spojení pomocí UDP, později pak TCP. Ve druhém toku dochází k přenosu skriptu PowerShell přes záznam TXT.

Na dalším příkladu jsou vidět dva toky útoku SIGRed:

```
START_SEC:                          2023-02-24 09:11:22.703920932
L4_PROTO:                           17 (UDP)
INVEA_DNS_QTYPE:                    24 (SIG)
INVEA_DNS_QCLASS:                   1  (IN)
INVEA_DNS_QNAME:                    9.skodliva-domena.cz
INVEA_DNS_FLAGS_CODES_RESPONSE:     33664 (Truncated)

BYTES:                              66636
START_SEC:                          2023-02-24 09:11:22.704965462
L4_PROTO:                           6  (TCP)
BYTES_A:                            591
BYTES_B:                            66045
INVEA_DNS_QTYPE:                    24 (SIG)
INVEA_DNS_QCLASS:                   1  (IN)
INVEA_DNS_QNAME:                    9.skodliva-domena.cz
INVEA_DNS_CRR_TYPE:                 NIL
INVEA_DNS_CRR_CLASS:                NIL
INVEA_DNS_CRR_TTL:                  NIL
INVEA_DNS_CRR_NAME:                 NIL
INVEA_DNS_CRR_RDATA_LEN:            NIL
INVEA_DNS_CRR_RDATA:
```

Podobně jako u útoku DNSMessenger je v prvním toku navazováno spojení pomocí TCP. Ve druhém je již zaslána škodlivá odpověď. O tom svědčí velikost přenesených dat ze strany serveru ke klientovi, v toku značená jako `BYTES_B`.

Monitorování pomocí IPFIX má však problém správně zpracovat odpověď DNS, což značí pole s hodnotou `NIL`. Zřejmě se tak děje kvůli tomu, že odpověď typu SIG není validní.

4.3 Detekce událostí systémem IDS pomocí signatur

Systém IDS (z anglického Intrusion Detection System) je softwarové či hardwarové řešení pro detekci škodlivého provozu v síti. Jedná se tedy o jeden z bezpečnostních prvků v síti, který pomáhá odhalit hrozby, které by například firewall nezachytil.

Běžně se rozlišují dva typy těchto systémů na základě přístupu, jak hrozby detekují: pomocí signatur nebo anomálií. V rámci této práce se bude dále pracovat se systémem detekujícím pomocí signatur.

Systém pracuje na principu porovnávání síťového provozu se vzory (signaturami), známých útoků uložených v databázi. Pokud se nějaká síťová aktivita shoduje se signaturou, je vytvořena výstraha [26].

Pro účely této práce byla zvolena implementace systému IDS Suricata[2] ve verzi 6 pro její poměrně velkou rozšířenost v praxi. Jedná se o open-source nástroj vytvořený organizací Open Information Security Foundation (OISF).

Hlavní roli v tomto systému hrají tzv. pravidla (rules), která plní úlohu zmiňovaných signatur v systému IDS. Základní formát takového pravidla je následující: *akce*, *hlavička* a *parametry*.

Akce určuje, co se má provést v případě, když se síťový provoz shoduje s tímto pravidlem. Další částí je hlavička, která určuje protokol, který by se měl shodovat, zdrojovou a cílovou adresu IP s portem a směr komunikace. Na závěr je nutné uvést parametry daného pravidla do kulatých závorek, které nám udávají textový řetězec, jež je součástí výstrahy v případě detekce. Dále také obsahuje samotné parametry, které se musí s patřičným síťovým provozem shodovat, jednoznačný identifikátor pravidla a další užitečné informace o signatuře. Podrobnější informace o nástroji Suricata je možné nalézt v oficiální dokumentaci, viz [6].

Příklad jednoho z volně dostupných pravidel z databáze Emerging Threats od společnosti Proofpoint[3] je uveden níže. Konkrétně se jedná o pravidlo detekující exfiltrační útok Alina POS, viz kapitola 2.1.

```
alert dns $HOME_NET any -> any any (msg:"ET MALWARE AlinaPOS Exfiltration v
ia DNS"; dns.query; content:".akamai-analytics.com"; nocase; endswith; pcre
:"/^[A-Z0-9_-]+\.akamai-analytics\.com$$/i"; reference:url, blog.centurylin
k.com/alina-point-of-sale-malware-still-lurking-in-dns/; classtype:command-
and-control; sid:2030441; rev:1; metadata:created_at 2020_07_02, deployment
Perimeter, former_category MALWARE, malware_family AlinaPOS, performance_im
pact Low, signature_severity Major, updated_at 2020_07_02;)
```

Jak z příkladu vyplývá, v případě shody provozu s pravidlem se vygeneruje výstraha na základě klíčového slova `alert`. Z hlavičky lze vyčíst, že se jedná o odchozí provoz DNS ze sítě oběti na libovolnou adresu. Na základě klíčových slov `dns.query` a `content` se detekují takové dotazy DNS, které obsahují doménové jméno `akamai-analytics.com` spojené s útokem Alina POS. Dále je klíčovým slovem `reference` uveden odkaz na další informace o tomto útoku. Součástí parametrové části pravidla je dále globálně unikátní identifikátor pravidla, typ signatury a další metadata.

4.3.1 Konfigurace a výstup systému

Systém Suricata je na testovacím zařízení nakonfigurován tak, že všechna vlastní pravidla jsou uložena v souboru `local.rules`.

V případě shody síťového provozu s jedním z pravidel z daného souboru dojde k vytvoření výstrah ve dvou separátních souborech. V prvním souboru `fast.log` dojde k vytvoření jednoduchého textového logovacího záznamu, který poskytne základní informace o možné probíhající hrozbě.

[2] viz `https://suricata.io/`

[3] viz `https://rules.emergingthreats.net/open/suricata-5.0/`

Druhý soubor `eve.json` poskytuje detailnější informace o podezřelém provozu detekovaného systémem Suricata. Obsah záznamu podezřelého provozu souboru typu JSON se mění na základě typu aplikačního protokolu, kterého případně využívá, a parametrů pravidla, se kterým se tento provoz shoduje.

Mějme následující pravidlo uložené ve zmíněném souboru `local.rules`.

```
alert dns any any -> any any (msg:"DNS dotaz na fit.vutbr.cz"; dns.query; c
ontent:"fit.vutbr.cz"; nocase; sid:1;)
```

Pokud dojde k vytvoření dotazu DNS na záznam typu A pro doménové jméno `www.fit.vutbr.cz` například nástrojem `nslookup`, vygeneruje se výstraha.

Logovací záznam s touto výstrahou se nachází v souboru `fast.log`. Jeho podoba je uvedena níže.

```
02/13/2023-17:44:53.750701  [**] [1:1:0] DNS dotaz na fit.vutbr.cz [**] [Cl
assification: (null)] [Priority: 3] {UDP} 127.0.0.1:48372 -> 127.0.0.1:53
```

Pro podrobnější informace je nutné se podívat do již zmiňovaného souboru `eve.json`, jehož obsah je uveden zde:


```
{
    "timestamp": "2023-02-13T17:44:53.750701+0100",
    "flow_id": 184650995954797,
    "in_iface": "ens33",
    "event_type": "alert",
    "src_ip": "192.168.235.129",
    "src_port": 48372,
    "dest_ip": "192.168.235.2",
    "dest_port": 53,
    "proto": "UDP",
    "tx_id": 0,
    "alert": {
        "action": "allowed",
    "gid": 1,
        "signature_id": 1,
        "rev": 0,
        "signature": "DNS dotaz na fit.vutbr.cz",
        "category": "",
        "severity": 3
    },
    "dns": {
        "query": [
        {
            "type": "query",
            "id": 57502,
            "rrname": "www.fit.vutbr.cz",
            "rrtype": "A",
            "tx_id": 0
        }
```

```
        ]
    },
    "app_proto": "dns",
    "flow": {
        "pkts_toserver": 1,
        "pkts_toclient": 0,
        "bytes_toserver": 87,
        "bytes_toclient": 0,
        "start": "2023-02-13T17:44:53.750701+0100"
    }
}
```


Kromě parametrů a hlavičky samotného pravidla se zde dozvídáme i bližší informace o samotném dotazu DNS. Taktéž je poskytnuta specifikace daného toku, tedy počtu přenesených paketů a bajtů v obou směrech a čas začátku.

4.3.2 Přehled dostupných signatur pro detekci útoků DNS

Za účelem získání již nabízených způsobů detekce jednotlivých hrozeb pro vlastní návrh detekčního nástroje byla provedena rešerše dostupných signatur pro detekci útoků DNS. V rámci této rešerše byly zkoumány konkrétně dvě komerční sady. Všechna dále zmiňovaná pravidla v této kapitole jsou uvedena v příloze B.

Signatury pro systémy Suricata

První sada signatur nese název Emerging Threats a je udržována společností Proofpoint. Tato sada je nabízena ve dvou verzích: Open, tedy bezplatná a Pro, placená. V rámci této práce byla prostudována verze Open[4].

V této sadě se nachází přesně 42 734 signatur k lednu roku 2023. Z těchto pravidel se bezmála čtvrtina z nich zabývá systémem DNS, konkrétně 9 873.

V rámci této sady lze tedy nalézt mnoho pravidel pro detekci různých útoku na systém DNS. Mezi těmito útoky jsou například hrozby typu DoS nebo útoky zaměřené na podvržení záznamu v paměti cache. Taktéž lze najít pravidla pro detekci různých hrozeb využívající DNS jako tunelovací komunikační nástroj skrz server C2. Co se týče detekčních pravidel pro jednotlivé hrozby zmiňované v kapitole 2, tato sada pokrývá útoky Alina POS, DNSMessenger a SIGRed.

Signatury pro systémy Snort

Další prostudovaná sada signatur nese název Snort Subscriber Rule Set[5]. Tato sada je vytvořena pro uživatele systému IDS Snort a to ve dvou verzích: pro registrované uživatele a pro předplatitele. Během rešerše byla využita verze pro registrované uživatele, která je bez poplatku a nabízí stejnou sadu pravidel jako verze pro předplatitele. Jejím jediným omezením je zpoždění aktualizací signatur o 30 dnů oproti aktuální placené verzi.

Podobně jako sada Emerging Threats obsahuje více jak 40 tisíc pravidel, konkrétně 44 362 k únoru roku 2023. Oproti této sadě však obsahuje podstatně méně pravidel týkajících se systému DNS – pouze 527, tedy něco málo nad jedno procento. Tato pravidla

[4] viz `https://rules.emergingthreats.net/open/suricata-5.0/`

[5] viz `https://www.snort.org/downloads/#rule-downloads`

se opět zaměřují na širokou škálu různých útoků jako tunelování dat nebo škodlivé pakety DNS. Taktéž obsahuje velké množství konkrétních doménových jmen na černé listině.

V této sadě byla nalezena pravidla pro detekci útoků DNSMessenger a SIGRed.

Pravidla pro detekci malwaru „Alina POS“

K útoku Alina POS se v sadě Emerging Threats nachází čtveřice pravidel, které si kladou za cíl upozornit na možný průběh této hrozby. Tato pravidla detailně nestudují obsah paketu a ani se nezaměřují na pravidelné znaky tohoto útoku. Příklad části jednoho z pravidel je uveden níže.

```
dns.query; content:".analytics-akadns.com"; nocase; endswith; pcre:"/^[A-Z0
-9_-]+\.analytics-akadns\.com$/i";
```

Jak je z ukázky patrné, tak jediným způsobem detekce je shoda s jedním z doménových jmen, které je známo, že je s touto hrozbou spojováno.

V sadě Snort Subscriber Rule Set se k tomuto útoku nenachází ani jedno pravidlo.

Pravidla pro detekci útoku „DNSMessenger“

Pravidlo pro detekci útoku DNSMessenger v rámci sady Emerging Threats je již sofistikovanější oproti detekci hrozby Alina POS. Zaměřuje se především na shodu hlavičky paketu DNS a přítomnosti řetězce „H4sIA“, který má být obsažen v záznamu DNS. Klíčová slova uvedená níže jsou určující pro detekci tohoto řetězce.

```
content:"|00 10 00 01|"; content:"H4sIA"; distance:7; within:5;
```

Sada pro systémy Snort taktéž obsahuje signatury pro tento útok. Oproti pravidlu ze sady Emerging Threats nestuduje obsah paketu DNS do hloubky, ale pouze sleduje doménové jméno, na které je kladen dotaz. Pravidlo sady systému Snort zkoumá, jak lze vidět níže, zda je délka první subdomény přesně deset znaků a zda je druhou subdoménou řetězec „stage“.

```
content:"|01 00 00 01 00 00 00 00 00 00 0A|",depth 11,offset 2; content:"|0
5|stage",within 6,distance 10,nocase; content:"|00 10 00 01|",within 45;
```

Taktéž je ověřováno, že se jedná o dotaz na záznam typu A.

Druhým větším rozdílem oproti pravidlu Emerging Threats je směr komunikace. Toto pravidlo se snaží útok zachytit při dotazování se serveru C2, zatímco pravidlo Emerging Threats útok detekuje až při samotné infiltraci dat ze serveru C2.

Pravidla pro detekci hrozby „SIGRed“

Sada Emerging Threats nabízí pravidla i pro třetí ze zmiňovaných útoků a tou je hrozba SIGRed. Detekce tohoto útoku je postavena na dvojici pravidel, u nichž je klíčovým rozdílem, zda je škodlivá odpověď DNS se záznamem SIG odesílaná ze serveru DNS ke klientovi či naopak. Společná část těchto pravidel je uvedena níže:

```
byte_test:2,>=,0xfeea,0; content:"|00 00 18|"; within:76; content:"|00 00 1
8|"; distance:12; within:64; fast_pattern; content:"|c0|"; distance:2; with
in:1; content:"|00 18|"; distance:1; within:2;
```

U obou těchto pravidel je prováděna kontrola na velikost odpovědi DNS, konkrétně zda je větší než 65 258 bajtů. Taktéž se kontroluje, zda je dotazováno a odpovídáno na záznamu typu SIG, jehož hrozba SIGRed využívá, viz kapitola 2.3.

Podobně tomu je i u signatury v sadě Snort Subscriber Rule Set, která nabízí čtveřici pravidel, jež se liší ve směru komunikace a hodnotě v poli `OPCODE` u hlavičky DNS. Toto pole, viz příloha A.3, obsahuje informaci o typu paketu DNS. V tomto případě může jít o běžný dotaz nebo o paket v rámci přenosu zón mezi autoritativními servery. Případ, kdy se jedná o běžný dotaz, je uveden níže.

```
content:"|FF|",depth 1; byte_test:1,=,0,4,bitmask 0x78; content:"|00 18|",
depth 40,offset 22;
```

Podobně jako pravidla ze sady Emerging Threats se ověřuje velikost paketu DNS, zda je větší než 65 280 bajtů, a že v něm figuruje záznam typu SIG.

Pravidla pro detekci algoritmů DGA

Pro detekci algoritmů DGA je ve většině případů u signatur využito tzv. černé listiny. Detekce tak probíhá pouze na základě výčtu známých vygenerovaných domén pomocí nějakého z algoritmů. Pokud je položen dotaz na jedno z jmen, pro které je vytvořeno pravidlo, tak dojde k vytvoření výstrahy. Na příkladu níže je uvedena část pravidla detekující komunikaci malwaru Zeus se serverem C2 ze sady Emerging Threats:

```
content:"|01 00 00 01 00 00 00 00 00 00|"; depth:10; offset:2; content:"|10
|xvcewyydwsmdgaju|02|ru|00|"; nocase; distance:0;
```

4.3.3 Použitelnost systému při detekci

Z popisu výše a rešerše uvedených sad pravidel je patrné, že tento nástroj je vhodný na detekci všech doposud zpracovávaných útoků v této kapitole.

Taktéž, jak vyplynulo z rešerše volně dostupných pravidel, pro všechny tyto útoky, tedy Alina POS, DNSMessenger, SIGRed a různé typy DGA, existují již vytvořená pravidla. Tento fakt potvrzuje možnost detekce těchto hrozeb s pomocí systému Suricata.

4.4 Způsob detekce jednotlivých hrozeb

Tato část kapitoly si klade za cíl představit vytvořené návrhy v rámci této práce pro detekci konkrétních hrozeb blíže popsaných v kapitole 2. U každého z návrhů je popsán princip detekce krok za krokem s dovysvětlením účelu. Dle těchto návrhů bude dále provedena implementace detekčního nástroje popsaná v následující kapitole.

4.4.1 Malware „Alina POS“

Tento malware zacílený na odcizení citlivých údajů z platebních zařízení je detailně popsán v kapitole 2.1. Jak je v této kapitole zmíněno, tento malware využívá k provedení útoku pouze pole doménového jména, do které vloží zakódované údaje určené k exfiltraci.

Jednou z cest, jak takový útok detekovat, je zaměřit se tedy přímo na doménové jméno. Zbytek paketu dotazu DNS není pravděpodobně nijak útočníkem manuálně upravován.

Mějme dva dotazy DNS, které se dotazují na záznam typu A na následující doménové jména, respektive jejich textové podoby:

```
88vh5NnzkOjr6eHl7Ozj6e_QmJCQzs7JztnY3JuEz9LPkJCbm5ubm5ubm5ubm5.ubm5ubl5qbmp
uYmpiZk5OTk5OTkw.akamai-technologies.com

yeTLxcbvkOjr6eH_-pCYkPrDxM0.akamai-technologies.com
```

1. Na začátku z takovýchto řetězců odstraníme oddělovače subdomén, tedy „.“. Po odstranění teček dostáváme takovéto řetězce:

   ```
   88vh5NnzkOjr6eHl7Ozj6e_QmJCQzs7JztnY3JuEz9LPkJCbm5ubm5ubm5ubm5ubm5ubl5qbmpu
   YmpiZk5OTk5OTkwakamai-technologiescom

   yeTLxcbvkOjr6eH_-pCYkPrDxM0akamai-technologiescom
   ```

2. Dále pak provedeme dekódovací operace pro následné porovnání s regulárním výrazem. Proces dekódování je detailně zobrazen na obrázku 2.2 v kapitole 2.1. V případě, že by útočník začal k zakódovaní používat operaci XOR s jiným bajtem než 0xAA, tak lze pomocí hrubé síly získat všechny možné řetězce a ty dále testovat.

   Po provedení těchto operací na zmíněném doménovém jméně dostáváme následující řetězce. Pro lepší přehlednost byly odstraněny znaky z konců těchto řetězců, které nebyly v rozsahu kódování ASCII.

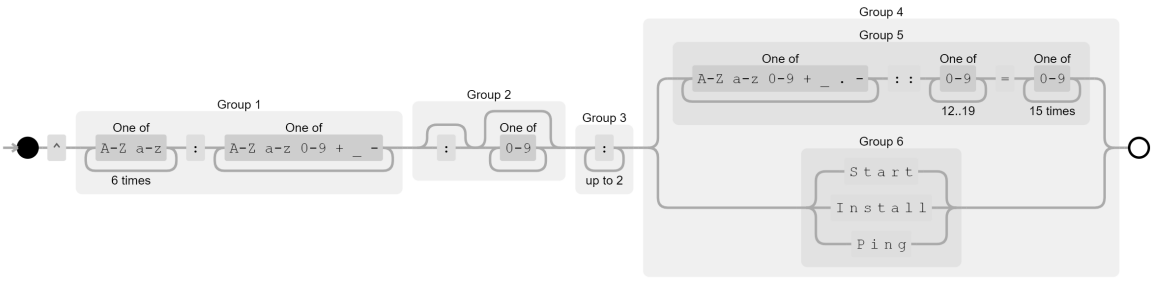
   ```
   YaKNsY:BACKOFFICEz2::ddcdsrv1.exe::1111111111111111=010120239999999

   cNaolE:BACKUP:2:Ping
   ```

3. Na závěr jsou takovéto řetězce otestovány na shodu s vytvořeným a dále blíže popsaným regulárním výrazem. Jeho zápis ve standardu PCRE (Perl Compatible Regular Expressions) je uveden níže.

   ```
   /^([A-Za-z]{6}:[A-Za-z0-9+_-]+)(:?[0-9]*)(:{1,2})(([A-Za-z0-9+_.-]+::[
   0-9]{8,19}=[0-9]{15})|(Start|Install|Ping))/
   ```

   Na obrázku 4.1 je znázorněn význam jednotlivých částí regulárního výrazu. Na začátku je detekováno, zdali se jedná o začátek řetězce s doménovým jménem. Je to z toho důvodu, že zakódovaná data se tímto útokem přidávají do nejlevější části doménového jména.



Obrázek 4.1: Zobrazení regulárního výrazu.

V první skupině regulárního výrazu následuje kontrola na přítomnost unikátního identifikátoru o šesti znacích složených z velkých a malých písmen. Za oddělovačem „:“ by se měl nacházet název infikovaného zařízení. Níže lze vidět, že tuto část mají oba dříve dekódované řetězce společnou.

```
YaKNsY:BACKOFFICEz2::ddcdsrv1.exe::1111111111111111=010120239999999

cNaolE:BACKUP:2:Ping
```

V druhé skupině se detekuje případná přítomnost oddělovače „:“ a libovolného čísla. Tato část se nenachází v řetězcích při tomto útoku vždy. Jak je vidět níže, tak ji v tomto případě obsahuje pouze druhý řetězec.

```
YaKNsY:BACKOFFICEz2::ddcdsrv1.exe::1111111111111111=010120239999999

cNaolE:BACKUP:2:Ping
```

V pořadí třetí skupina řeší další možnou rozdílnost v řetězcích tohoto útoku. Oddělovačem v tomto bodě může být jeden znak „:“ a nebo dva. Jejich rozdílnost je opět znázorněna níže:

```
YaKNsY:BACKOFFICEz2::ddcdsrv1.exe::1111111111111111=010120239999999

cNaolE:BACKUP:2:Ping
```

V poslední částí detekce shody s regulárním výrazem se toto porovnávání větví na dvě varianty:

(a) V první se ověřuje přítomnost názvu procesu, který platební údaje získal. Tento řetězec může mít neomezenou délku a musí být neprázdný. Za posledním oddělovačem „::“ by se již mělo nacházet číslo platební karty v rozsahu 8 až 19 číslic dle běžných rozsahů[6]. Za oddělovačem „=“ se pak testuje přítomnost 15 číslic, z nichž první osm určuje datum expirace platební karty.

S prvním uvedeným řetězcem dojde, jak je ukázáno níže, ke shodě.

```
YaKNsY:BACKOFFICEz2::ddcdsrv1.exe::1111111111111111=010120239999999
```

(b) Druhá varianta v rámci větvení slouží k detekci jiné podoby útoku a to zasíláním zpráv serveru C2 z napadeného zařízení.

Po názvu zařízení se oproti první variantě testování shody liší v tom, že je detekován jeden řetězec z následující množiny: „Start“, „Install“, „Ping“. Tyto řetězce značí název prováděného příkazu.

Na ukázce níže vidíme, že se druhý zmiňovaný řetězec v této části shoduje.

```
cNaolE:BACKUP:2:Ping
```

[6] viz `https://www.ansi.org/news/standards-news/all-news/2016/07/announcing-major-changes-to-the-issuer-identification-number-iin-standard-28`

V případě shody s daným regulárním výrazem je možné takový paket DNS identifikovat jako útok Alina POS.

4. Ověření detekce lze provést kontrolou odpovědi DNS na exfiltrující dotaz u typů monitorování, které to umožní. Jelikož se oficiálně jedná o dotaz na záznam typu A, tak v odpovědi se pravděpodobně vrátí adresa 127.0.0.1.

Dalším možným způsobem detekce je zavést tzv. černou listinu (blacklist) doménových jmen, které útočník k exfiltraci využívá. Jedná se o tato doménová jména: `analytics-akadns.com`, `akamai-analytics.com`, `akamai-information.com`, `akamai-technologies.com`, `sync-akamai.com`.

4.4.2 Útok „DNSMessenger“

Jak již bylo zmíněno v kapitole 4.2, k úspěšné detekci infiltračního útoku je zapotřebí nahlédnout hlouběji do paketů DNS.

Na obrázku 4.2 je možné vidět záchyt komunikace simulace takového útoku. Zpočátku je proveden dotaz z napadeného zařízení na doménu serveru C2 na záznam typu TXT. Tento dotaz není nijak neobvyklý a je tradičně zasílán pomocí protokolu UDP. Na něj je vrácena odpověď od serveru C2 s příznakem Truncated.

Protocol	Length	Info
DNS	104	Standard query 0x1246 TXT 285a1.f950033561.dnsm.in.geoffjamessmith.com
DNS	104	Standard query response 0x1246 TXT 285a1.f950033561.dnsm.in.geoffjamessmith.com
TCP	74	32951 → 53 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=2293080802 TSecr=0
TCP	74	53 → 32951 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM TSval=42825208
TCP	66	32951 → 53 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=2293080806 TSecr=4282520853
DNS	130	Standard query 0x866a TXT 285a1.f950033561.dnsm.in.geoffjamessmith.com
TCP	66	53 → 32951 [ACK] Seq=1 Ack=65 Win=65152 Len=0 TSval=4282520854 TSecr=2293080807
DNS	11…	Standard query response 0x866a TXT 285a1.f950033561.dnsm.in.geoffjamessmith.com TXT
TCP	66	32951 → 53 [ACK] Seq=65 Ack=1095 Win=64128 Len=0 TSval=2293080812 TSecr=4282520859
TCP	66	32951 → 53 [FIN, ACK] Seq=65 Ack=1095 Win=64128 Len=0 TSval=2293080813 TSecr=428252
TCP	66	53 → 32951 [FIN, ACK] Seq=1095 Ack=66 Win=65152 Len=0 TSval=4282520860 TSecr=229308
TCP	66	32951 → 53 [ACK] Seq=66 Ack=1096 Win=64128 Len=0 TSval=2293080813 TSecr=4282520860

Obrázek 4.2: Záchyt paketů z komunikace při simulaci útoku DNSMessenger.

Na tento příznak napadený klient reaguje opětovným odesláním stejného dotazu, tentokrát však pomocí protokolu TCP. Jmenný server C2 pak již přes tento transportní protokol zašle odpověď se záznamem TXT obsahujícím zakódovaný skript Powershell. Podrobnější informace o průběhu tohoto útoku lze nalézt v kapitole 2.2.

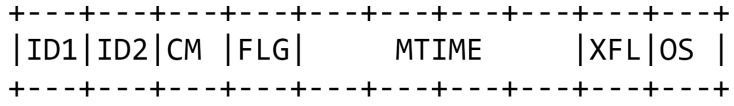
Jak již bylo zmíněno, na dotazování ze strany napadeného klienta není nic nestandardního. Detekovat útok je možné až při inspekci odpovědi DNS, která infiltruje škodlivý skript. V odpovědi je nejdříve vhodné ověřit, zda je dotaz i záznam v odpovědi typu TXT.

Druhým znakem pro detekci je počáteční podobnost obsahu záznamů TXT napříč dotazy. V rámci testování za pomocí simulačního nástroje bylo provedeno vícero dotazů na infiltraci různých skriptů Powershell. Na příkladu níže jsou uvedeny tři záznamy TXT obsahující odlišné zakódované skripty.

```
$e='H4sIAAAAAAAACA82UwW7TQBCGz/FTjJRITiSCkisSB+oEtSINFTZClXxZ7HGzymbXzI4...
$e='H4sIAAAAAAAACA32SwWrDMAyGz/VTCDJwAOvfYLeNrTB2SDZ6dhO58UjtYistpfTdZ8d...
$e='H4sIAAAAAAAACA2VRy27CMBC85ytWVg5FwlEf50qlVHOcoBFG4li5zgIusZ3aDhRR/r1...
```

Jak je možné si povšimnout, začátek vždy obsahuje přiřazení do proměnné `$e` a následuje zakódovaný obsah vždy začínající na „H4sIAAAAAAAACA“. Jak je popsáno v kapitole 2.2, skript Powershell se před vložením do záznamu TXT komprimuje nástrojem `gzip` a poté se zakóduje algoritmem Base64.

Na obrázku 4.3 je uvedena struktura hlavičky souboru komprimovaného pomocí nástroje `gzip`.



Obrázek 4.3: Struktura hlavičky souboru komprimovaného nástrojem `gzip` [17].

Pole `ID1` a `ID2`, která zabírají dohromady velikost dva bajty, mají pevně nastavenou hodnotu na 0x1F8B. Následuje bajt `CM` určující metodu komprimace. Nástroj `gzip` používá metodu „deflate“ o hodnotě 0x8. Za touto hodnotou je opět jednobajtové pole pro určení typu komprimovaného souboru. Pro případ skriptu Powershell se pravděpodobně bude jednat o textový soubor, tedy hodnotu 0x0. Poté hlavička dále obsahuje pole `MTIME` pro poslední modifikační čas originálního souboru nebo hodnotu 0x0, pokud takový čas není určen. Zbývající pole `XFL` slouží pro určení typu komprese a pole `OS` nám říká typ souborového systému, na kterém došlo ke kompresi [17].

Pokud vezmeme opakovanou hodnotu z ukázek výše, tedy „H4sIAAAAAAAAC“ a dekódujeme ji algoritmem Base64, dostáváme hexadecimální hodnotu 0x1F8B08000000000002. Ta odpovídá hlavičce nástroje `gzip`, kde proběhla komprese textového souboru metodou „deflate“ bez modifikačního času.

Je tedy pravděpodobné, že minimálně začátek hlavičky, tedy čtyři bajty označující nástroj `gzip`, metodu „deflate“ a kompresi textového souboru, bude obsažen v každém paketu tohoto útoku. Lze tedy provádět testování, zda se začátek obsahu záznamu TXT neshoduje s řetězcem „$e=’H4sIA“.

Podobným přístup k detekci je použit i u pravidla systému IDS sady Emerging Threats, viz kapitola 4.3.

Opět jako v případě malwaru Alina POS lze detekovat i doménová jména spojená s tímto útokem, konkrétně: `ns0.pw`, `ns0.site`, `ns0.space`, `ns0.website`, `ns1.press`, `ns1.website`, `ns2.press`, `ns3.site`, `ns3.space`, `ns4.site`, `ns4.space`, `ns5.biz`, `ns5.online`, `ns5.pw`.

4.4.3 Hrozba „SIGRed“

Další hrozbou, jež byla detailněji rozebrána v kapitole 2.3 a je pro ni navržen způsob detekce, je hrozba SIGRed. Tato hrozba si klade za cíl způsobit pád aplikace `dns.exe` na operačních systémech Windows Server.

Obrázek 4.4 ukazuje posloupnost komunikace v rámci tohoto útoku. Nejdříve dojde k položení dotazu DNS ze zařízení kontrolovaného útočníkem na aplikaci `dns.exe`, v tomto případě naslouchající na adrese 10.0.0.10. Ta přepošle dotaz na server DNS útočníka, který zasílá paket DNS nad transportním protokolem UDP s příznakem Truncated. Aplikace `dns.exe` se pokouší navázat spojení přes protokol TCP a očekává odpověď na záznam typu SIG. V rámci záznamu se nachází škodlivý obsah, který způsobí pád aplikace `dns.exe`.

Podobně jako u simulace útoku DNSMessenger nelze do doby zaslaní patřičné odpovědi DNS určit, že se jedná o hrozbu. Předešlá komunikace do posledního paketu DNS totiž

Source	Destination	Protocol	Length	Info
10.0.0.101	10.0.0.10	DNS	80	Standard query 0x5dd8 SIG 9.skodliva-domena.cz
10.0.0.10	10.0.0.234	DNS	103	Standard query 0xd45c SIG 9.skodliva-domena.cz OPT
10.0.0.234	10.0.0.10	DNS	124	Standard query response 0xd45c SIG 9.skodliva-domena.cz SOA ns1.9.skodliva-domena.cz
10.0.0.10	10.0.0.234	DNS	129	Standard query 0x1176 SIG 9.skodliva-domena.cz OPT
10.0.0.234	10.0.0.10	DNS	19…	Standard query response 0x1176 SIG 9.skodliva-domena.cz SIG

Obrázek 4.4: Záchyt paketů z komunikace při simulaci hrozby SIGRed.

nevykazuje žádné známky neobvyklého nebo škodlivého provozu. Možné dřívější zachycení útoku lze realizovat označením dotazu na záznam SIG za podezřelý. Jelikož již byl nahrazen záznamem RRSIG, tak není příliš standardní, a tudíž může být potenciálně škodlivý. Případně lze tento předpoklad spojit s faktem, že doménové jméno začíná číslicí.

Pravděpodobně přesvědčivější formou detekce je zkoumání škodlivé odpovědi obsahující záznam typu SIG. Nejdříve je tedy nutné ověřit, že je dotaz i odpověď na záznam SIG.

Druhou testovací položkou by mohla být samotná velikost paketu DNS. V případě, že by ukazatel mířil na nejnižší kardinální hodnotu povolených znaků, tedy 45, tak by stačilo, aby byl paket větší než 65 491 bajtů. Tato velikost by s kombinací škodlivého ukazatele v paketu DNS způsobila přetečení bufferu aplikace `dns.exe`.

Dalším znakem k pozorování je samotná hodnota ukazatele v paketu DNS, konkrétně v jednom z polí záznamu SIG. Příklad paketu se škodlivým záznamem typu SIG je uveden na obrázku 4.5.

```
0000  ff eb 11 76 81 a0 00 01  00 01 00 00 00 00 01 39   ···v····  ·····•9
0010  0f 73 6b 6f 64 6c 69 76  61 2d 64 6f 6d 65 6e 61   ·skodliv a-domena
0020  02 63 7a 00 00 18 00 01  c0 0c 00 18 00 01 00 00   ·cz·····  ········
0030  00 20 ff b9 00 01 05 00  00 00 00 20 68 76 a2 1f   · ·····  ··· hv··
0040  5d 2c ca 1f 9e 04 c0 0d  00 0f ff ff ff ff ff ff   ],·····  ········
0050  ff ff ff ff ff ff ff ff  ff 0f ff ff ff ff ff ff   ·····  ·····
0060  ff ff ff ff ff ff ff ff  ff 0f ff ff ff ff ff ff   ·····  ·····
0070  ff ff ff ff ff ff ff ff  ff 0f ff ff ff ff ff ff   ·····  ·····
0080  ff ff ff ff ff ff ff ff  ff 0f ff ff ff ff ff ff   ·····  ·····
0090  ff ff ff ff ff ff ff ff  ff 00 00 00 00 00 00 00   ·····  ·····
00a0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ·····  ·····
```

Obrázek 4.5: Část paketu DNS se škodlivým záznamem SIG.

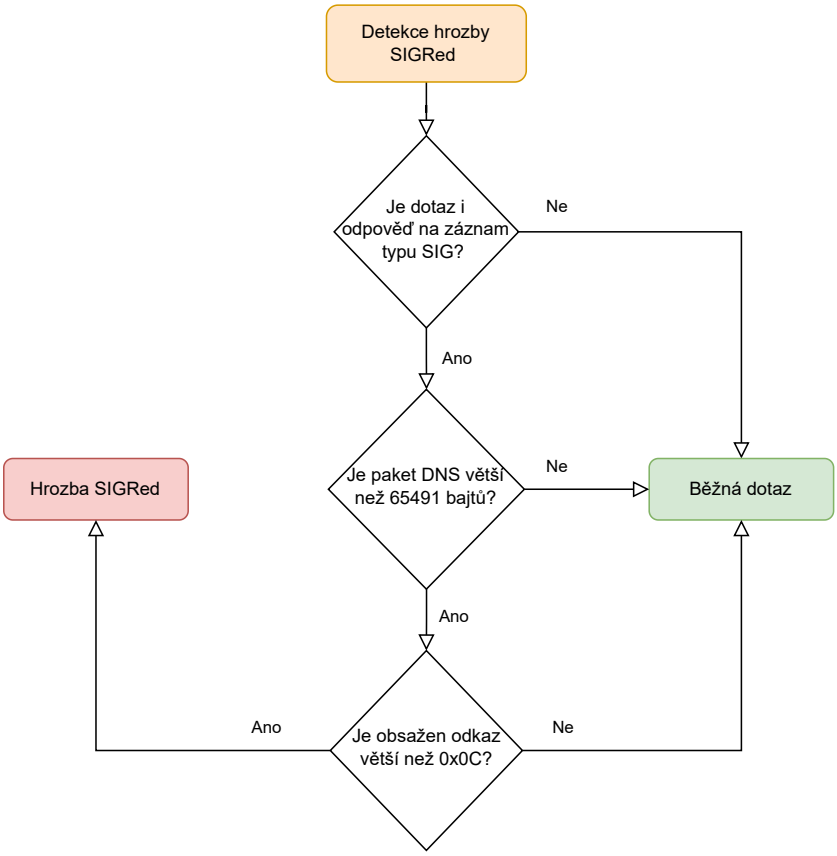
Pokud je hodnota ukazatele vyšší než 0x0C, tak z toho vyplývá, že neukazuje na první bajt doménového jména. V tomto případě tedy ukazuje na první znak doménového jména, tedy číslo devět. Ten má dle tabulky ASCII kardinální hodnotu 57 a aplikace `dns.exe` z toho pochopí, že je potřeba takový prostor alokovat pro první subdoménu.

Na obrázku 4.6 je uveden postup detekce graficky zpracovaný v podobě vývojového diagramu.

4.4.4 Algoritmy DGA

V rámci této kapitoly není kladen důraz na detekci konkrétního malwaru, který využívá vlastní implementaci DGA, ale na možnost detekce širokého spektra algoritmů.

V první části je provedena rešerše dostupných nástrojů a technik využívaných k detekci algoritmů DGA. Rozsahem je tato rešerše stručnější, jelikož se jedná o poměrně rozsáhlé téma, kterým se podrobněji zabývají jiné práce, například [16, 22, 35]. V druhé části se nachází návrh způsobu detekce v rámci implementovaného detekčního nástroje.



Obrázek 4.6: Vývojový diagram průběhu detekce hrozby SIGRed.

Dostupné systémy a metody detekce

V rámci studie [16] je představen návrh systému detekující malware na základě dotazů na škodlivá doménová jména, který nese název MORTON. Vstupem nástroje je komunikace DNS malwarů získaná z logovacích záznamů. Jedná se o malwery, které komunikují v pravidelných intervalech se serverem C2. Systém MORTON je schopen detekovat domény vytvořené algoritmem DGA.

Proces detekce je založen na transformaci vstupních dat na vektor výkonové spektrální hustoty (PSD – z angl. Power Spectral Density), který charakterizuje intenzitu periodické komunikace při různých frekvencích. Nejdříve jsou z dat vyčleněny domény považované za bezpečné. Zbylá vyfiltrovaná data jsou agregována podle času a vložena do diskrétní Fourierovy transformace, která vytvoří zmíněný vektor PSD. Tento vektor je následně vstupem pro neuronovou síť, která data klasifikuje a určí, zdali jsou dané dotazy DNS vytvořeny malwarem či nikoliv.

V této studii je taktéž tento nástroj porovnán s podobnými systémy Baywatch a WARP oproti nimž je přibližně 13krát rychlejší.

Druhý prostudovaný odborný článek se zaměřoval na detekci malwaru využívajících algoritmy DGA pomocí monitorování NetFlow [22]. Monitorování popsané v tomto článku probíhalo pro komunikaci DNS každé stanice v lokální síti pomocí sondy. V této práci je vyzdvižena výhoda monitorování IPFIX z důvodu, že je jím dnes vybaveno velké množství směrovačů.

Navržený detekční algoritmus spočívá ve výpočtu koeficientu $\rho(a)$ pro zařízení a. Ten značí poměr mezi počtem provedených dotazů DNS $\delta(a)$ a počtem unikátních adres IP kontaktovaným zařízením a označený jako $\pi(a)$ [22]:

$$\rho(a) = \frac{\delta(a)}{\pi(a) + 1}. \tag{4.1}$$

Přičtením hodnoty jedna ve jmenovateli se předchází situaci, že by byl jmenovatel nulový. Obvyklá hodnota u běžných zařízení tohoto koeficientu je nízká. V případě vysoké hodnoty lze předpokládat, že zařízení bylo napadeno malwarem využívající DGA. Důvodem je, že adresa serveru C2 zůstává stejná, ale často se mění jeho doménové jméno. Z toho plyne, že je v čitateli vysoká hodnota a ve jmenovateli nízká.

Druhým způsobem detekce je přímo zkoumání doménového jména. V odborném článku [35] je dostupný výčet jednotlivých nástrojů, které se zabývají detekcí různých hrozeb v rámci komunikace DNS.

Jedním z nich je systém Exposure, který je z přehledu nástrojů dle [35] jediný ze zmiňovaných schopen efektivní detekce doménových jmen generovaných pomocí DGA. Činí tak na základě analýzy pasivního, již proběhlého, provozu DNS. Konkrétně doménová jména DGA detekuje výpočtem zastoupení numerických znaků v doménovém jméně a poměrem nejdelšího smysluplného podřetězce vůči celému doménovému jménu. Tento postup je jedním ze vstupních parametrů pro detekci malwaru zneužívající systém DNS. Jako většina nástrojů představených v této práci využívá i systém Exposure neuronovou síť k posouzení škodlivosti doménového jména.

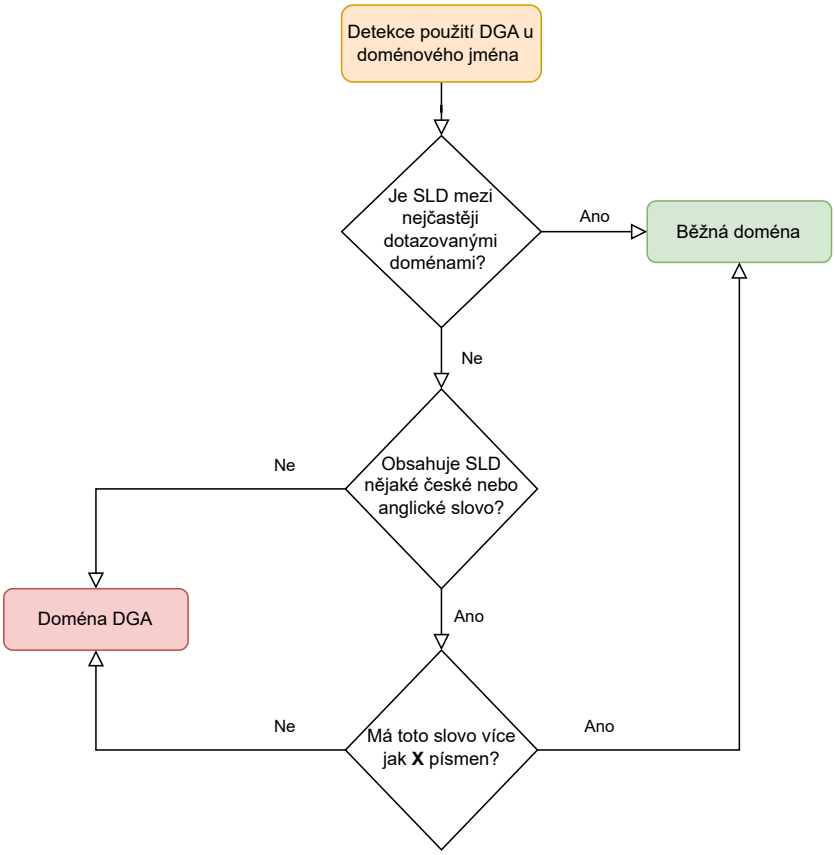
Výstupem studie [35] je návrh nového systému, které bude zpracovávat vstupní data podstatně rychleji než zmiňované nástroje a přiblíží se tak detekci v reálném čase. Takový systém má taktéž detekovat škodlivé domény včetně těch vytvořených algoritmy DGA.

Detekce domén DGA v rámci představeného systému v [35] je založena na použití algoritmu PPM (Prediction per Partial Matching). Díky němu je možné predikovat následující textový symbol analyzováním předešlých symbolů v doménovém jméně. Při zpozorování nového doménového jména je použit klasifikátor PPM na výpočet průměrné pravděpodobnosti vzhledem k jednotlivým symbolům. Z tohoto výpočtu je pak spočítán poměr mezi ním a referenční hodnotou. Pokud je tato hodnota menší než prahová, v tomto případě 0,8, tak je doménové jméno považováno za škodlivé.

Návrh detekčního mechanismu

Pro detekci domén generovaných pomocí algoritmů DGA byl zvolen přístup lexikálního zkoumání doménového jména. Tento postup byl vybrán z důvodu jeho poměrně jednoduché implementace. Dalším důvodem zvoleného postupu je možnost detekce hned z prvního dotazu na doménu DGA, jelikož zvolený postup nevyžaduje delší statistiku počtu provedených dotazů používanou u některých nástrojů výše. Postup detekce je zobrazen na vývojovém diagramu 4.7.

Při lexikální analýze je zkoumána doména druhého řádu popřípadě třetího, pokud je v doménovém jméně obsažena doména TLD s rozšířením v doméně druhého řádu (SLD), např. britská „co.uk“. Na začátku tedy dojde k vyčlenění této relevantní části doménového jména, které je podrobena další analýze. Doména druhého či třetího řádu je společně s vyšším řády hledána v seznamu nejčastěji dotazovaných doménových jmen. Pro toto vy-



Obrázek 4.7: Vývojový diagram detekce doménového jména generovaného pomocí DGA.

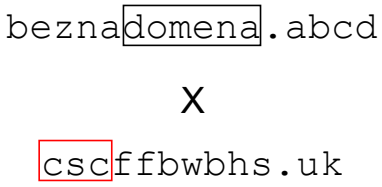
hledávání byl využit seznam od společnosti Cloudflare[7] obsahující jeden milion nejčastěji dotazovaných doménových jmen.

V případě, že se zkoumané doménové jméno v tomto seznamu nenachází, pokračuje se další kontrolou. V této fázi se zkoumá pouze doména druhého či třetího řádu bez ostatních subdomén. Tato část dotazovaného doménového jména je blíže zkoumána na přítomnost anglických či českých slov. K tomuto prohledávání je využit algoritmus Aho-Corasick [7], kde je trie (prefixový strom) konstruován z anglických a českých slov. Co se týče anglického slovníku, byla využita sada 12dicts[8]. K získání seznamu českých slov byl použit abecední slovník českého národního korpusu [1]. Tento slovník byl upraven pro použití k vyhledávání slov v doménových jménech odstraněním diakritiky. Následně bylo provedeno sloučení těchto dvou slovníků a odstranění vzniklých duplicit.

Pokud algoritmus Aho-Corasick nenajde žádnou shodu, je doména považovaná za generovanou pomocí algoritmu DGA. Pokud dojde ke shodě, tak je zaznamenána její délka. Pokud došlo k vícero shodám, tak je počítáno s tou nejdelší z nich. Tato délka se následně porovná s prahovou hodnotou, v diagramu označenou jako X. Pokud je menší nebo rovna, tak je doména opět považována za vygenerovanou. Výchozí prahová hodnota byla zvolena tři. Její hodnota byla optimalizována v experimentu v kapitole 5.3.

Na obrázku 4.8 je možné vidět znázornění hledání shody slovníkového řetězce s podřetězci v doménových jménech. Mějme fiktivní doménové jméno `beznadomena.abcd`, které

[7] viz `https://radar.cloudflare.com/domains`
[8] viz `http://wordlist.aspell.net/12dicts/`



Obrázek 4.8: Příklad nálezu slovníkového slova v doménových jménech.

nebylo vytvořeno pomocí žádného algoritmu DGA. Jelikož toto doménové jméno nebylo nalezeno v seznamu nejčastěji dotazových jmen, tak se blíže zkoumá jeho část `beznadomena`. V ní je nalezeno mnoho shod, z nichž je nejdelší shoda se slovem „doména“ z českého slovníku o délce šest znaků. Oproti tomu druhé doménové jméno `cscffbwbhs.uk` je vygenerováno jedním z algoritmů DGA. Opět toto jméno nebylo nalezeno v seznamu nejčastěji dotazovaných doménových jmen a tudíž je provedeno hledání slovníkových řetězců ve jméně `cscffbwbhs`. Jak je vidět, v náhodném rozmístěni znaků byla nalezena nejdelší shoda o délce tři znaky, konkrétně akademický titul „CSc.“ opět z českého slovníku, což nepřesahuje stanovenou prahovou hodnotu.

V rámci využití této navržené detekční metody je nutné počítat s následujícími limitacemi. Tato metoda není účinnou pro detekci tzv. slovníkových algoritmů DGA blíže popsaných v kapitole 2.4, jelikož záměrně generují doménová jména obsahující slova určitého jazyka. Taktéž tato metoda nebude účinná na specifických webech různých národů v jiném jazyce než angličtině či češtině. Další možnou slabinou může být využití neobvyklé zkratky v doménovém jméně, která není obsažena ve slovníku.

4.5 Shrnutí

V rámci této kapitoly byly představeny tři různé monitorovací přístupy pro zachycení komunikace DNS. U každého z nich byla blíže uvedena konfigurace konkrétního systému, ukázka výstupů a úvaha nad možností detekce jednotlivých hrozeb.

Prvním z nich byl systém logovaní na „cache-only“ serveru DNS. Jelikož představený systém logovaní na serveru BIND 9 přímo nepodporuje zaznamenávání odpovědí na dané dotazy DNS, je poměrně limitující z hlediska detekce. Konkrétně u útoku DNSMessenger by byla možná detekce omezena pouze na formu blacklistu domén spojených s tímto útokem. Podobný problém nastává i u hrozby SIGRed, u níž je nutné znát formu odpovědi na daný dotaz na záznam typu SIG. U malwaru Alina POS nebo různých algoritmů DGA však nevzniká problém s detekcí, jelikož k jejich odhalení postačí znát doménové jméno z dotazu DNS.

Druhým zpracovaným systémem bylo monitorování IPFIX. Konkrétně byla k monitorování využita sonda Flowmon. Tento způsob monitorovaní již nabízí ucelený tok s dotazem DNS i odpovědí na něj. Z tohoto záznamu toku pak lze jednoduše vyčíst všechny parametry z paketů DNS. Tento fakt umožňuje detekci všech zmiňovaných útoků.

Posledním testovaným nástrojem byl detekční systém IDS Suricata. U tohoto typu nástroje byla provedena podrobná rešerše dostupných signatur pro různé systémy IDS. Z výsledků této rešerše vyplývá, že na útoky Alina POS, DNSMessenger a SIGRed existují volně dostupná pravidla, které tyto útoky jsou schopny detekovat. Taktéž byly nalezeny pravidla pro detekce doménových jmen různých algoritmů DGA či řady jiných útoku na systém DNS. To prokázalo, že tento systém je schopen detekovat všechny zmiňované hrozby. Současně

byla provedena rešerše dostupných pravidel ze sady pro systémy Snort. Bylo zjištěno, že tato sada poskytuje pravidla pro detekci útoků DNSMessenger, SIGRed a některé algoritmy DGA.

V tabulce 4.1 je uveden jednoduchý přehled schopnosti různých systémů detekovat jednotlivé hrozby v rámci systému DNS.

V závěrů této kapitoly byly taktéž představeny původní návrhy detekčních metod pro hrozby Alina POS, DNSMessenger, SIGRed a algoritmy DGA. K algoritmům DGA byla provedena rešerše existujících detekčních nástrojů a jejich metod.

Monitorování / **Hrozba**	Logování	IPFIX	IDS
Alina POS	✓	✓	✓
DNSMessenger	✗	✓	✓
SIGRed	✗	✓	✓
DGA	✓	✓	✓

Tabulka 4.1: Přehled použitelnosti monitorovacích systému pro dané hrozby.

Kapitola 5

Implementace a testování nástroje

V minulé kapitole byly představeny jednotlivé způsoby monitorování komunikace DNS společně s návrhem způsobu detekce vybraných hrozeb. Na tuto část je navázáno popisem implementace detekčního nástroje a jeho následné testování. Cílem této kapitoly je tedy představit navržený nástroj a otestovat jej na různých datových sadách.

Nejdříve je představen obecný návrh detekčního nástroje, kde je podrobněji popsána jeho modularita a celková funkčnost. Na tuto část navazuje samotný popis implementace nástroje. Dále je uveden bližší popis implementovaného detekčního algoritmu. Tuto kapitolu uzavírá dokumentace testování v podobě popsaných experimentů a jejich zhodnocení.

5.1 Obecný návrh detekčního nástroje

V následující kapitole je představena koncepce detekčního nástroje. Takový nástroj si především klade za cíl odhalení hrozeb v komunikaci DNS. Taktéž by měl fungovat na datech z různých monitorovacích systémů. Po průchodu všech monitorovacích dat by měl uživateli podat údaje o případných hrozbách a části komunikace, v které byly detekovány.

Při návrhu byl kladen důraz na modularitu nástroje, která umožňuje jeho snadné rozšíření v budoucnosti. Navrhovaný detekční nástroj je tvořen čtyřmi moduly:

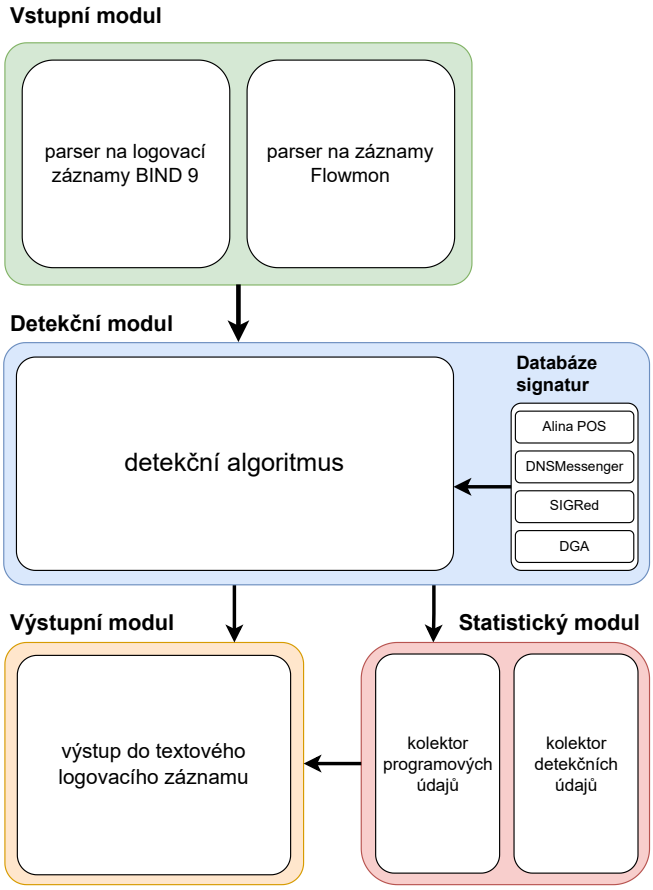
1. vstupní modul,
2. detekční modul,
3. statistický modul,
4. výstupní modul.

Bližší popis těchto modulů lze nalézt dále v této kapitole. Moduly jsou navrženy tak, aby spolu komunikovaly pouze pomocí přesně stanovených abstraktních datových struktur a rozhraní. Provázanost těchto modulů a celkové schéma navrženého detekčního nástroje je zobrazeno na obrázku 5.1.

5.1.1 Vstupní modul

Vstupní modul si klade za cíl zpracovat data z podporovaných monitorovacích systémů a přetransformovat je do unifikované podoby, což je jeho výstupem.

Podporovanými monitorovacími způsoby mohou dle obrázku 5.1 například být logovací záznamy ze serveru BIND, viz kapitola 4.1, nebo záznamy toků ze sondy IPFIX od



Obrázek 5.1: Návrh detekčního nástroje rozděleného na jednotlivé moduly.

společnosti Flowmon, viz kapitola 4.2. V případě, že zvolený způsob monitorovaní nedokáže poskytnout všechny atributy unifikovaného toku DNS, tak se nechá chybějící atribut prázdným a dle režimu detekčního nástroje je na něj brán ohled či nikoliv.

5.1.2 Detekční modul

V rámci detekčního modulu nejdříve dojde k načtení všech dostupných signatur hrozeb systému DNS. Tyto signatury definují pravidla, které musí unifikovaný tok ze vstupního modulu splňovat, aby byl označen za podezřelý.

V případě že byl tok označen za podezřelý splněním všech povinných podmínek jedné ze signatur hrozeb, tak je vytvořen záznam uchovávající odkaz na signaturu hrozby a podezřelý tok. Výstupem tedy je soubor těchto záznamů o potenciálně škodlivých tocích.

5.1.3 Výstupní a statistický modul

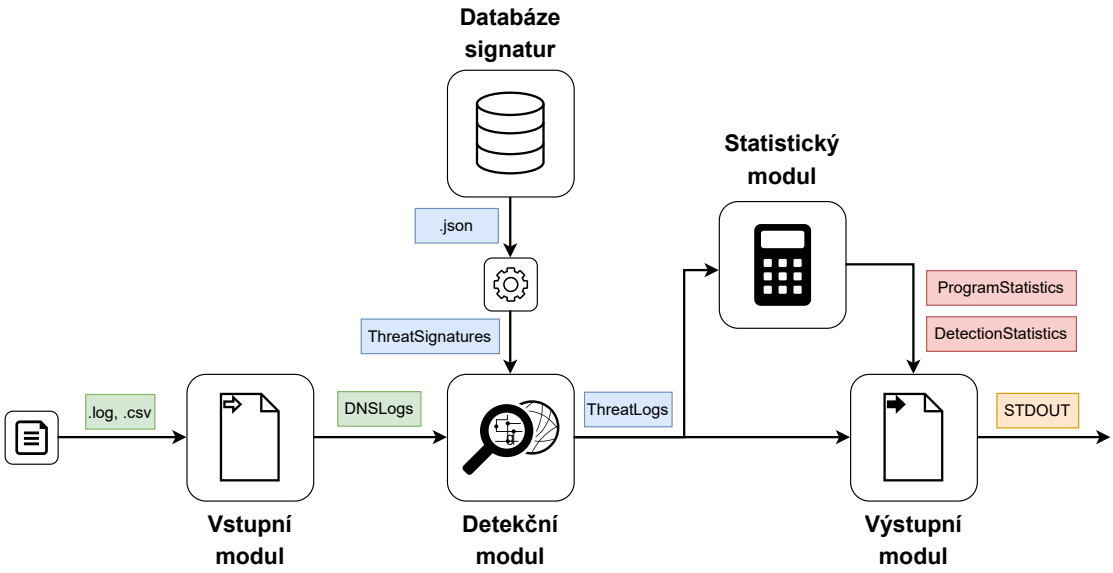
Výstup detekčního modulu putuje do statistického a výstupního modulu. Ve statistickém modulu je podroben analýze a výpočtu různých statistických hodnot, např. počet zpracovaných toků, počet vstupních souborů, počet načtených signatur, poměr zastoupení hrozeb atd. Detailní popis je uveden v příkladu výstupu v příloze D. Taktéž se k výstupu statistického modulu přidávají informace o vstupních argumentech nástroje.

Tato zpracovaná data statistický modul předává výstupnímu modulu k jejich zobrazení uživateli. Výstupní modul uživateli zobrazí nejen výčet jednotlivých podezřelých toků a informace k nim, ale taktéž statistické údaje zmíněné výše.

5.2 Implementace detekčního nástroje

V rámci této části kapitoly jsou popsány vybrané části implementace detekčního nástroje. Samotný nástroj je složen z mnoha částí, které byly implementovány samostatně. V této kapitole jsou však zmíněny pouze podstatné částí implementace. Podrobnější programovou dokumentaci lze nalézt přímo ve zdrojových kódech na přiloženém médiu, viz příloha C.

Pro implementaci byl zvolen skriptovací jazyk Python, konkrétně ve verzi 3.11. Jazyk Python byl vybrán hlavně z důvodu jeho velice dobré použitelnosti pro skriptování včetně zpracovávání různých typů souborů. Samotné schéma funkcionality nástroje je zobrazeno na obrázku 5.2.



Obrázek 5.2: Ukázka funkcionality detekčního nástroje.

Z obrázku 5.2 je patrné, že vstupem pro tento nástroj jsou různé typy souborů dle monitorovacího systému. Ty se pak zpracují pomocí vstupního modulu do jednotné podoby logovacího záznamu DNS, více k této problematice je uvedeno v následující kapitole. Takto strukturované logovací záznamy DNS jsou vstupem pro detekční modul. Do detekčního modulu rovněž vstupují nadefinované signatury s hrozbami systému DNS po jejich zpracovaní do abstraktního datového typu.

Po průchodu všemi logovacími záznamy DNS detekční modul předává pole s potenciálně škodlivými logovacími záznamy DNS. Záznamy dále putují do statistického a rovněž do výstupního modulu. Ve statickém modulu dojde k vytvoření programových a detekčních statistik na základě vstupu z detekčního modulu a tyto statistiky jsou dále předány do výstupního modulu. Výstupní model pak zobrazuje logovací záznamy o hrozbách a statistiky na standardní výstup.

5.2.1 Unifikace různých monitorovací dat na vstupu

K tomu, aby bylo možné detekovat hrozby nehledě na monitorovací systému je zapotřebí, aby probíhala detekce na unifikovaných vstupních datech. Pro každý typ vstupního monitorovacího systému je implementován zpracovávající modul, který naplní instanci třídy `DNSLog`, jejíž definice atributů je uvedena níže:

```
class DNSLog:
    bytes_sum: Final[Optional[int]]
    start_sec: Final[Optional[datetime]]
    l4_proto: Final[Optional[int]]
    bytes_question: Final[Optional[int]]
    bytes_answer: Final[Optional[int]]
    ip_client: Final[Optional[IPv4Address | IPv6Address]]
    ip_server: Final[Optional[IPv4Address | IPv6Address]]
    port_client: Final[Optional[int]]
    port_server: Final[Optional[int]]
    dns_id: Final[Optional[int]]
    dns_question_type: Final[Optional[int]]
    dns_question_class: Final[Optional[int]]
    dns_question_name: Final[Optional[str]]
    dns_answer_type: Final[Optional[int]]
    dns_answer_class: Final[Optional[int]]
    dns_answer_ttl: Final[Optional[int]]
    dns_answer_name: Final[Optional[str]]
    dns_answer_rdata_len: Final[Optional[int]]
    dns_answer_rdata: Final[Optional[bytes]]
    dns_flags_question: Final[Optional[bytes]]
    dns_flags_answer: Final[Optional[bytes]]
```

Jednotný logovací záznam komunikace DNS je navržen tak, aby obsahoval veškeré podstatné informace o paketu s dotazem i odpovědí DNS. Taktéž se zde nachází dodatečné atributy pro adresaci komunikace za pomocí adres IP a čísel portů či jiné dodatečné informace jako jsou velikosti komunikačních toků. Během zpracování vstupních souborů se tedy převádí jednotlivá monitorovací data na hodnoty atributů instance třídy `DNSLog`. V případě, že monitorovací systém není schopen z důvodu špatného zpracovaní paketů či přímo jeho limitací naplnit všechny atributy, tak se ponechá atribut prázdný.

5.2.2 Struktura signatur hrozeb a jejich zpracovaní

Pro snadnou rozšiřitelnost detekčního nástroje byla navržena strukturovaná forma signatury hrozby systému DNS. Každá taková signatura hrozby je uložena jako objekt v souboru typu JSON. Příklad takové signatury je uveden níže, konkrétně se jedná o hrozbu Alina POS.

```
{
    "id_num": 1,
    "name": "Alina POS",
    "description": "exfiltration of credit card information",
    "info_url": "https://blog.lumen.com/alina-point-of-sale-malware-still-l
```

```
                urking-in-dns/",
    "conditions": [
    {
        "cond_type": "mandatory",
        "description": "domain name contains an encoded string with credit
                        card details",
        "attribute_field": "dns_question_name",
        "operations": [
        {
            "parameter": ".",
            "operation": "REMOVE_CHAR_STR"
        },
        {
            "operation": "BASE64_DECODE"
        },
        {
            "operation": "XOR_BRUTEFORCE"
        },
        {
            "parameter": "^([A-Za-z]{6}:[A-Za-z0-9+_-]+)(:?[0-9]*)(:{1,2})(
                         ([A-Za-z0-9+_.-]+::[0-9]{12,19}=[0-9]{15})|(Start
                         |Install|Ping))",
            "operation": "REGEX_CHECK"
        }
]
    },
    {
        "cond_type": "additional",
        "description": "DNS answer on record type A~contains IPv4 address 1
                        27.0.0.1",
        "attribute_field": "dns_answer_rdata",
        "operations": [
        {
            "parameter": "127.0.0.1",
            "operation": "EQUAL"
        }
        ]
    },
    {
        "cond_type": "blacklist",
        "description": "second-level domain is on the blacklist",
        "attribute_field": "dns_question_name",
        "operations": [
        {
            "parameter": "analytics-akadns\\.com|akamai-analytics\\.com|aka
                         mai-information\\.com|akamai-technologies\\.com|s
                         ync-akamai\\.com",
```

```
            "operation": "REGEX_CHECK"
        }
        ]
    }
    ]
}
```

Z příkladu plyne, že signatura má unikátní identifikátor, název, stručný popis a odkaz na další informace o této hrozbě. Dále je složena z pole podmínek, které je nutné splnit k označení logovacího záznamu DNS za hrozbu. Tyto podmínky jsou rozděleny do třech kategorií: povinné, blacklist a dodatečné. Povinné podmínky je nutné splnit k označení logovacího záznamu za škodlivý. Blacklist podmínky jsou takové, které se zaměřují na detekci útoků pomocí indikátorů kompromitace. Dodatečné podmínky pak nemusí být nutně splněny k detekci, ale slouží jako další informace o útoku. Jedná se o takové podmínky, které nejsou zásadní pro označení logovacího záznamu za hrozbu, ale v útocích se mohou vyskytnout.

Každá podmínka pak má svůj popis a atribut logovacího záznamu DNS, ke kterému se vztahuje. Nad kopií tohoto atributu je následně prováděna posloupnost definovaných operací, z niž poslední musí vrátit pravdivostní hodnotu k vyhodnocení podmínky. V ukázce výše je možné vidět například operaci `EQUAL` k posouzení rovnosti mezi parametrem operace a kopií atributu. Operace `REGEX_CHECK` pak například hledá shodu kopie atributu s regulárním výrazem definovaným v parametru operace.

5.2.3 Detekce pomocí kontroly platnosti podmínek

Detekce je implementována pomocí postupného procházení logovacích záznamů DNS seřazených dle času. Na každém logovacím záznamu je testován soubor podmínek každé načtené signatury. Jak již bylo blíže popsáno v předchozí kapitole, detekce rozlišuje tři typy podmínek (povinná, blacklist a dodatečná). Aby došlo k označení logovacího záznamu DNS za škodlivý, tak je nutné buď splnění jedné z blacklist podmínek či splnění všech povinných, případně kombinace. Pomocí matematické zápisu lze splnění podmínek vyjádřit následovně:

$$\exists B \vee \forall M, \tag{5.1}$$

kde B jsou blacklist podmínky a M povinné podmínky.

V rámci optimalizace detekce se vykonává testování podmínek v pořadí: blacklist, povinné a dodatečné. Pokud není splněna jedna z povinných podmínek a zároveň nebyla splněna žádná z blacklist podmínek, tak se pokračuje inspekcí záznamu DNS na jiné signatuře. Na příkladu uvedeném níže je úryvek kódu implementující tento detekční mechanismus.

```
for signature in signatures:
|   conditions_matches: dict[DetectionCondition, Optional[bool]] = {}
|   matches_at_least_one_cond: bool = False
|   matches_blacklist_cond: bool = False
|   continue_inspecting: bool = False
|
|   for condition in signature.conditions:
|   |   cond_match = check_match_condition(dns_log, condition)
|   |   if not matches_at_least_one_cond and cond_match and
```

```
|   |      condition.cond_type != 'additional':
|   |   |   matches_at_least_one_cond = True
|   |
|   |   if not matches_blacklist_cond:
|   |   |   if condition.cond_type == 'blacklist' and cond_match:
|   |   |   |   matches_blacklist_cond = True
|   |   |   elif condition.cond_type == 'mandatory':
|   |   |   |   if (strict_mode and cond_match is None) or (not cond_match
|   |   |   |   |   and cond_match is not None):
|   |   |   |   |   continue_inspecting = True
|   |   |   |   |   break
|   |
|   |   conditions_matches[condition] = cond_match
|
|   if not continue_inspecting and matches_at_least_one_cond:
|   |   return PotentialThreatLog(dns_log, signature, conditions_matches,
|   |                             matches_blacklist_cond)
```

Striktní režim

Vzhledem k limitacím monitorovacích systému či jejich případným nedokonalostem může dojít k tomu, že nejsou všechny atributy instance třídy `DNSLog` naplněny hodnotami. Pokud nějaká podmínka vyžaduje práci s atributem, který není dostupný, tak se ve výchozím nastavení tato podmínka zahodí a kontrolují se další podmínky.

Jelikož tento efekt však může vést k velkému množství falešných pozitiv, tak byl implementován tzv. striktní režim. Ten vyžaduje, aby všechny požadované atributy v povinných podmínkách byly dostupné k úspěšné detekci. Nevýhodou tohoto přístupu však je, že v případě částečného nedokonalého zpracování paketu monitorovacím systémem není možné útok nikdy detekovat.

5.3 Testování nástroje na datových sadách

V následující části kapitoly je provedena dokumentace jednotlivých experimentů pro otestovaní implementovaného detekčního nástroje. U každého z experimentů jsou na začátku definovány konkrétní vstupy. Dále jsou popsány přesné výstupy, nad kterými je provedena diskuze z níž vyplývá zhodnocení daného experimentu.

Všechny experimenty byly provedeny v testovacím prostředí na virtuálním stroji. Konkrétně testování probíhalo s technickými parametry uvedenými v tabulce 5.1.

Operační systém	Ubuntu 22.04.2 LTS
Jádro	Linux 5.19.0-38-generic
Architektura	x86-64
Procesor	Intel Core i7-1165G7 @ 2.80GHz
Počet jader	4
Paměť RAM	DDR4
Velikost RAM	8 GB

Tabulka 5.1: Přehled technického vybavení k testování.

5.3.1 Experiment č. 1 – Validace detekčního nástroje

Vstupy a cíle

Cílem tohoto experimentu je provést detekci nad datovou sadou neobsahující žádný detekovatelný útok. Výsledkem takového experimentu by měla být detekce bez jediného pozitiva. Vstupem je datová sada s běžným provozem DNS, která si klade za cíl validovat tento detekční nástroj, to je vyloučit přítomnost detekovatelných hrozeb.

V rámci toho experimentu do detekčního nástroje vstupují dvě datové sady. Jedna z nich je odsimulovaná sondou Flowmon v podobě záznamů IPFIX v souboru typu CSV. Konkrétně se jedná o datovou sadu CIC-Bell-DNS-EXF-2021 obsahující exfiltrační komunikací DNS dle obecného algoritmu. Tato datová sada je blíže specifikována v kapitole 3.1. Druhou datovou sadu je dvoudenní záchyt komunikace DNS z páteřní sítě VUT.

Výstupy

Získané výstupy detekčního nástroje pro první datovou sadu jsou uvedeny v tabulce 5.2.

Soubor	**Délka vstupu**	**Falešná pozitiva**	**Trvání**
heavy_compressed-flowmon.csv	14,3 MB (5 h, 47 m)	0	1,372 s
heavy_exe-flowmon.csv	15,5 MB (5 h, 41 m)	0	1,462 s
heavy_text-flowmon.csv	13,8 MB (11 h, 36 m)	0	1,384 s
heavy_video-flowmon.csv	15,8 MB (6 h, 16 m)	0	1,435 s
light_audio-flowmon.csv	3,9 MB (2 h, 50 m)	0	1,028 s
light_exe-flowmon.csv	5,6 MB (57 m)	1 (DNSMessenger)	1,077 s
light_image-flowmon.csv	0,5 MB (5 m)	0	0,844 s
light_text-flowmon.csv	2,4 MB (32 m)	0	0,929 s
light_video-flowmon.csv	4,2 MB (42 m)	0	1,034 s

Tabulka 5.2: Výstup pro datovou sadu CIC-Bell-DNS-EXF-2021 při normálním režimu.

Po spuštění detekční nástroje nad výše zmiňovanými vstupními daty byla nalezena jedna hrozba, konkrétně DNSMessenger. Záznam o podezřelé komunikaci je uveden níže.

```
27-Mar-2023 16:36:06.130 | DNSMessenger: infiltration of malicious Powershe
ll script | client: 192.168.20.38#60136 <--> server: 192.168.20.72#53 (UDP)
| query: 643.ZML0E5PhhKMP-4g.cicresearch.ca
    Checked conditions:
        [MANDATORY] DNS question is for record type TXT
    Uncheckable conditions:
        [MANDATORY] DNS answer is for record type TXT
        [MANDATORY] TXT record contains "H4sIA" which is gzip header
                    encoded in Base64
```

Po opětovném spuštění nástroje, tentokrát však ve striktním režimu, již nebyla nalezena žádná hrozba. Výsledky tohoto spuštění detekčního nástroje nad již zmiňovanou datovou sadou jsou uvedeny v tabulce 5.3.

Experimenty jsme zopakovali nad daty z páteřní sítě VUT. Při normálním režimu došlo k záchytu hrozby DNSMessenger přibližně u 1,5 % logovacích záznamů. U všech takto zachycený hrozeb však při spuštění nástroje ve striktním režimu již poté k opakovanému záchytu nedošlo, tedy podobně jako u datové sady CIC-Bell-DNS-EXF-2021.

Soubor	Délka vstupu	Falešná pozitiva	Trvání
heavy_compressed-flowmon.csv	14,3 MB (5 h, 47 m)	0	1,36 s
heavy_exe-flowmon.csv	15,5 MB (5 h, 41 m)	0	1,416 s
heavy_text-flowmon.csv	13,8 MB (11 h, 36 m)	0	1,334 s
heavy_video-flowmon.csv	15,8 MB (6 h, 16 m)	0	1,4 s
light_audio-flowmon.csv	3,9 MB (2 h, 50 m)	0	1,071 s
light_exe-flowmon.csv	5,6 MB (57 m)	0	1,133 s
light_image-flowmon.csv	0,5 MB (5 m)	0	0,844 s
light_text-flowmon.csv	2,4 MB (32 m)	0	0,939 s
light_video-flowmon.csv	4,2 MB (42 m)	0	1,041 s

Tabulka 5.3: Výstup pro datovou sadu CIC-Bell-DNS-EXF-2021 při striktním režimu.

Při detekci nad daty ze sítě VUT byl na rozdíl od první sady dvanáctkrát detekován útok DNSMessenger na základě položeného dotazu na doménové jméno z černé listiny. Konkrétně šlo o doménové jméno `ns5.biz`. Jeden z takových záznamů je uveden zde:

```
12-Apr-2023 19:16:45.084 | DNSMessenger: infiltration of malicious Powershe
ll script | client: 143.116.85.184#56699 <--> server: 158.185.149.113#53 (U
DP)| query: ns5.biz
    Checked conditions:
        [BLACKLIST] second-level domain is on the blacklist
    Uncheckable conditions:
        [MANDATORY] DNS answer is for record type TXT
        [MANDATORY] TXT record contains "H4sIA" which is gzip header encode
                    d in Base64
```

Diskuze

Falešný nález v prvním běhu nástroje se vstupní datovou sadou CIC-Bell-DNS-EXF-2021 je především způsoben nedostatkem informací k jednoznačné detekci. Konkrétně nastala situace, kdy byl chybně označen logovací záznam DNS jako útok DNSMessenger z důvodu přítomnosti záznamu typu TXT.

V druhém běhu se zapnutým striktním režimem však již k tomuto jevu nedošlo. Chybějící políčka v monitorovacích záznamech, která jsou pro detekci DNSMessenger důležitá, znemožnila označení logovacího záznamu za hrozbu.

Obdobně tomu bylo i u valné většiny označených záznamů k hrozbě DNSMessenger u dat ze sítě VUT. Výjimku tvořily právě dotazy na zmiňované doménové jméno `ns5.biz`. Při bližší analýze komunikace přímo ze souborů PCAP však nebyly nalezeny žádné známky útoku DNSMessenger. Taktéž při hledání informací o dané doméně na webovém portále VirusTotal bylo zjištěno, že některé antivirové řešení ji sice stále označují za škodlivou ale naprostá většina již ne. Je tedy možné, že doménu využívá jiný, dnes již legitimní, subjekt.

5.3.2 Experiment č. 2 – Detekce tří vybraných hrozeb

Vstupy a cíle

V rámci druhého experimentu je cílem otestovat úspěšnost detekce vybraných hrozeb popisovaných v kapitole 2. Konkrétně je cílem detekovat hrozby Alina POS, DNSMessenger a SIGRed ve vytvořených datových sadách. Bližší popis sad lze nalézt v kapitole 3.3.

Datové sady byly následně simulovány pro vytvoření logovacích záznamů na serveru BIND 9 a záznamů IPFIX ze sondy Flowmon. Každému útoku tak náleží dva soubory dle typu monitorování.

Výstupy

Testování probíhalo postupným spuštěním detekčního nástroje s konkrétními soubory vytvořenými monitorovacím systémem. V tabulce 5.4 je uveden přehled výstupu detekčního nástroje.

Soubor	Délka vstupu	Hrozba	TP	TN	FP	FN	Přesnost	Režim	Trvání
alina_pos-flowmon.csv	259,9 kB (11 m, 16 s)	Alina POS	3	254	0	0	100 %	normální	1,546 s
alina_pos-flowmon.csv	259,9 kB (11 m, 16 s)	Alina POS	3	254	0	0	100 %	striktní	1,321 s
alina_pos-bind.log	31,6 kB (11 m, 16 s)	Alina POS	3	254	0	0	100 %	normální	1,351 s
alina_pos-bind.log	31,6 kB (11 m, 16 s)	Alina POS	3	254	0	0	100 %	striktní	1,324 s
dnsmess-flowmon.csv	232,5 kB (10 m, 30 s)	DNSMessenger	1	220	0	0	100 %	normální	1,322 s
dnsmess-flowmon.csv	232,5 kB (10 m, 30 s)	DNSMessenger	1	220	0	0	100 %	striktní	1,321 s
dnsmess-bind.log	28,0 kB (10 m, 30 s)	DNSMessenger	1	220	0	0	100 %	normální	1,310 s
dnsmess-bind.log	28,0 kB (10 m, 30 s)	DNSMessenger	0	220	0	1	99,5 %	striktní	1,278 s
sigred-flowmon.csv	236,4 kB (11 m, 4 s)	SIGRed	1	225	0	0	100 %	normální	1,350 s
sigred-flowmon.csv	236,4 kB (11 m, 4 s)	SIGRed	0	225	0	1	99,6 %	striktní	1,282 s
sigred-bind.log	28,1 kB (11 m, 4 s)	SIGRed	0	225	0	1	99,6 %	normální	1,292 s
sigred-bind.log	28,1 kB (11 m, 4 s)	SIGRed	0	225	0	1	99,6 %	striktní	1,243 s

Tabulka 5.4: Přehled výstupu detekčního nástroje.

V prvním sloupci tabulky 5.4 je uveden název souboru z něhož dle koncovky `flowmon.csv` či `bind.log` je možné odvodit monitorovací systém, který soubor vytvořil. Nad každým souborem byl spuštěn detekční nástroj dvakrát, a to v normálním a striktním režimu. Taktéž je zachycena doba trvání detekce.

Diskuze

U hrozby Alina POS je z výsledků experimentu zřejmé, že typ monitorování ani režim detekčního nástroje neměl žádný vliv na úspěšnost detekce. Ve všech případech byl nástroj schopen detekovat všechny tři podoby útoku zmiňované v kapitole 3.3.

Útok DNSMessenger byl úspěšně detekován bez ohledu na režim detekčního nástroje u monitorování IPFIX ze sondy Flowmon. Při logování na serveru BIND 9 je však detekce úspěšná pouze při použití normálního režimu. Níže je uveden příklad detekčního záznamu vytvořeného na základě průchodu souboru `dnsmess-bind.log` v normálním režimu při logování BIND 9:

```
20-Mar-2023 12:29:08.289 | DNSMessenger: infiltration of malicious Powershe
ll script | client: 10.0.0.101#41407 <--> server: 10.0.0.10 (TCP)| query: 2
85a1.stage.0.dnsmess-sim.com
    Checked conditions:
        [MANDATORY] DNS question is for record type TXT
```

```
        [ADDITIONAL] TCP is used for transmission
    Uncheckable conditions:
        [MANDATORY] DNS answer is for record type TXT
        [MANDATORY] TXT record contains "H4sIA" which is gzip header
                    encoded in Base64
```

Jak je z příkladu zřejmé, tento záznam toku DNS byl detekován pouze na základě jediného povinného pravidla. Toto pravidlo kontroluje, zda se jedná o dotaz na záznam TXT. To může vést k možným falešným pozitivům v rámci detekce, jelikož by mohl být každý dotaz na záznam typu TXT označen za škodlivý. Protože monitorovací systém neposkytl všechny potřebné údaje k detekci, tak není možné zkontrolovat zbylé podmínky, které jsou uvedeny v příkladu výše. Naproti tomu při spuštění nástroje ve striktním režimu hrozba DNSMessenger není detekována. Je tomu tak z důvodu, že se k těmto nekontrolovatelným podmínkám přistupuje jako k nesplněným.

Poslední útok, který byl předmětem tohoto experimentu, je hrozba SIGRed. Co se týče úspěšnosti detekce v rámci logování pomocí serveru BIND 9, tak jak již bylo popsáno v kapitole 4.1, je tento útok tímto způsobem nedetekovatelný.

Záznam pozitivní detekce v souboru `sigred-flowmon.csv` vytvořeném monitorováním IPFIX ze sondy Flowmon s normálním režimem nástroje je uveden níže:

```
27-Mar-2023 16:16:12.836 | SIGRed: vulnerability in the Windows DNS server
| client: 10.0.0.10#33479 <--> server: 10.0.0.234#53 (TCP)| query: 9.skodli
va-domena.cz
    Checked conditions:
        [MANDATORY] DNS question is for record type SIG
        [MANDATORY] TCP is used for transmission
        [MANDATORY] Byte stream is from server to client is bigger than 65,
                    500
    Uncheckable conditions:
        [MANDATORY] DNS answer is for record type SIG
        [ADDITIONAL] DNS Signer's name field in SIG record contains bigger
                     pointer value than 0x0C
```

Ze záznamu plyne, že nedokonalostí monitorovacího systému Flowmon, viz kapitola 4.2, není možné zkontrolovat, zda je odpověď na záznam typu SIG. Také není možné nahlédnout do obsahu odpovědi. To vede k tomu, že ve striktním režimu nástroje není takový tok označen za hrozbu SIGRed.

Na závěr tohoto experimentuje lze podotknout, že během detekce nad všemi vstupními soubory detekční nástroj nevytvořil žádný záznam o falešných pozitivech. Riziko možných falešných pozitiv při detekci hrozby DNSMessenger nad jiným soubory bylo zmíněno výše.

5.3.3 Experiment č. 3 – Detekce domén generovaných pomocí DGA

Vstupy a cíle

Cílem tohoto experimentu je otestovat úspěšnost záchytu potenciálně vygenerovaných domén algoritmy DGA. Taktéž je cílem zjistit, zda generuje navržená metoda detekce falešná pozitiva a zvolit ideální prahovou hodnotu, o níž byla řeč v rámci kapitoly 4.4.

Do detekčního nástroje v rámci tohoto experimentu vstupuje datová sada s dotazy na doménová jména generovaná pomocí algoritmů DGA. Její bližší popis lze nalézt v kapi-

tole 3.3. Taktéž jsou využita data ze zachycené komunikace v rámci páteřní sítě VUT použitá v prvního experimentu.

Výstupy

V tabulce 5.5 je uveden přehled úspěšnosti detekce generovaných domén pomocí algoritmů DGA na vytvořené datové sadě. V rámci experimentu byly voleny prahové hodnoty, tedy nejmenší možné velikosti slovníkového řetězce v doménovém jméně, v intervalu od dvou do pěti. V tabulce 5.5 je taktéž uvedena procentuální úspěšnost při vyjmutí doménových jmen, které byly generovány pomocí tzv. „slovníkových DGA“. Ty, jak bylo zmíněno v kapitole 4.4, není zvolený způsob detekce schopný zachytit.

Prahová hod.	TP	TN	FP	FN	Přesnost
2	23	1195	32	345	76,36 %
3	163	1110	117	205	79,81 %
4	318	1024	203	50	84,14 %
5	351	897	330	17	78,24 %

Tabulka 5.5: Přehled úspěšnosti detekce domén DGA při různých prahových hodnotách.

V datové sadě z páteřní sítě VUT bylo označeno 0,33 % záznamů jako využívající potenciálně generovaná doménová jména. V následují tabulce 5.6 jsou uvedena konkrétní doménová jména a jejich četnost v rámci detekce. K detekci byla nastavena prahová hodnota čtyři vzhledem k její nejvyšší přesnosti.

Doménové jméno	Prověřovaná část	Počet detekcí	Zastoupení
match.bnmla.com	bnmla	2014	2,16 %
watch-online.49n7wqynho5u.top	49n7wqynho5u	1835	1,97 %
extension.7tv.gg	7tv	1374	1,47 %
nextcloud.stefka.eu	stefka	1315	1,41 %
nsa.ten.cz	ten	1131	1,21 %
spark-prod-sk.gnp.cloud.upctv.sk	upctv	876	0,94 %
wpad.kompan.net	kompan	852	0,91 %
ns3.sysct.cz	sysct	820	0,88 %
ns2.sysct.cz	sysct	820	0,77 %
wpad.reno.local	reno	720	0,75 %

Tabulka 5.6: Přehled deseti nejčastěji zachycených potenciální domén DGA v síti VUT.

Diskuze

Jak plyne z údajů v tabulce 5.5, se zvyšující se prahovou hodnotou roste počet detekovaných True positives. Současně se zvyšuje počet zachycených False positives. V případě prahové hodnoty o hodnotě dva je sice množství falešných pozitiv nejnižší, ale zároveň je nejvyšší počet False negatives, což snižuje přesnost detekce. U prahové hodnoty pět je naopak množství zachycených domén DGA (True positives) největší, ale současně je nejnižší počet False negatives. Negativně při této prahové hodnotě vychází počet False positives, z čehož plyne, že každé čtvrté běžné doménové jméno je označeno za generované. Proto jako optimální

se jeví prahová hodnota čtyři, jelikož vykazuje nejvyšší přesnost detekce oproti ostatním prahovým hodnotám.

K výsledkům experimentu nad vytvořenou datovou sadou s doménami DGA je však vhodné podotknout, že běžná doménová jména, na která byly kladeny dotazy během vytváření datové sady, pochází ze seznamu běžných domén v rámci sady CIC-Bell-DNS-2021, viz kapitola 3.1. Tento seznam obsahoval mnoho doménových jmen neobsažených v seznamu nejčastěji dotazovaných doménových jmen od společnosti Cloudflare. Taktéž tato doménová jména mnohdy pocházela z národních domén zemí, kde se používá primárně jiný než anglický či český jazyk. To může být důvodem pro relativně nízkou přesnost detekce.

Vzhledem k záchytům doménových jmen při spuštění detekčního nástroje nad daty ze sítě VUT lze konstatovat, že mnoho záchytů je způsobeno nedosažením minimální délky smysluplného slova v doménovém jméně. Například u zkoumaných částí doménových jmen `7tv` či `ten` zjevně došlo k nesplnění minimální délky slova ve jméně. U některých doménových jmen, například `49n7wqynho5u`, se skutečně na první pohled zdá, že se jedná o generované domény, a tak je jejich označení za potenciálně škodlivé legitimní.

5.3.4 Experiment č. 4 – Rychlost detekce

Vstupy a cíle

Cílem tohoto experimentu je zdokumentovat rychlost detekce v závislosti na velikosti vstupního souboru a najít možnosti jejího zvýšení. Pro tento účel jsou využity datové sady CIC-Bell-DNS-2021, CIC-Bell-EXF-DNS-2021 a data ze sítě VUT, jelikož obsahují soubory typu CSV o různých velikostech. Tyto velikosti se nachází v rozsahu od jednotek megabajtů až po necelé dva gigabajty.

Experimentování je koncipováno ve formě postupného spouštění nástroje nad zmíněnými daty s jiným nastavením. Konkrétně jde o situaci s normálním nastavením, dále pak normálním nastavením bez detekčního mechanismu pro algoritmy DGA a na závěr ve striktním režimu.

Výstupy

V tabulce 5.7 jsou uvedeny vypočtené průměrné rychlosti při spouštění nástroje v daných režimech. Tyto průměrné hodnoty byly získány z rychlostí zpracovaní jednotlivých souborů uvedených v tabulce 5.8.

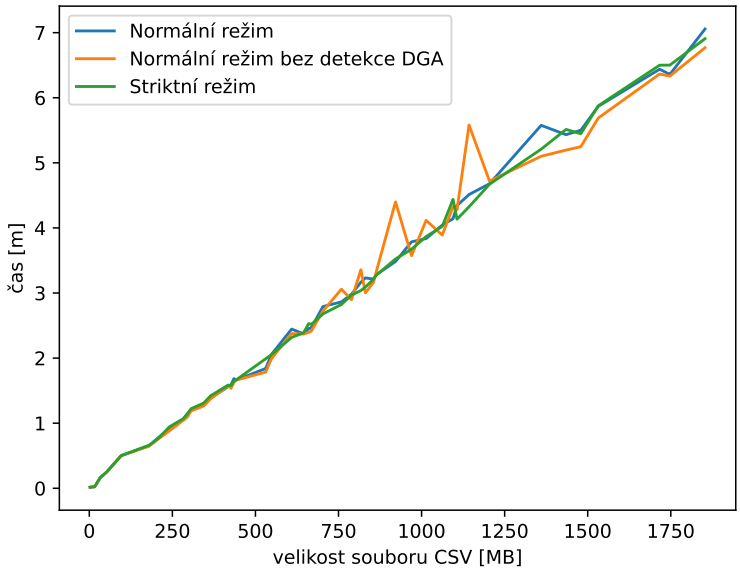
Režim detekce	**Průměrná rychlost zpracování**
Normální	4,67 MB/s
Normální bez detekce DGA	4,68 MB/s
Striktní	4,67 MB/s

Tabulka 5.7: Přehled rychlostí zpracování záznamů DNS při různých režimech.

V grafu 5.3 je zobrazena závislost délky běhu detekčního nástroje na velikosti vstupního souboru CSV. V tomto grafu jsou znázorněny tři různé křivky dle zmiňovaných režimů.

Diskuze

Z výstupů experimentu je zřejmé, že průměrná rychlost zpracování je nezávislá na režimu detekce. Překvapivé je to zejména při detekci algoritmů DGA, kdy dochází k náročným



Obrázek 5.3: Graf závislosti rychlosti detekce na velikosti vstupního souboru.

prohledávacím operacím a bylo by tedy možné předpokládat, že dojde ke zpomalení běhu nástroje. Z výsledků měření vyplývá, že slovníkové prohledávání či hledání v seznamu o milionu prvků je dostatečně optimalizováno tak, že nemá na rychlost detekce signifikantní vliv.

Zrychlení rychlosti detekce je možné dosáhnout přepsáním implementace do jiného než interpretovaného jazyka nebo optimalizací detekčního algoritmu. Také lze k možné akceleraci detekce zvolit lepší hardwarové vybavení. Práce se konkrétním postupem zrychlení detekce nezabývala.

5.4 Shrnutí

V rámci této kapitoly byl představen návrh detekčního nástroje a jeho modularita v rámci rozdělení na vstupní, detekční, statistický a výstupní modul. Dále byla popsána implementace nástroje dle tohoto návrhu s důrazem na podstatné částí, konkrétně zpracování vstupních dat, strukturu signatur s hrozbami a detekční algoritmus.

Během druhé části kapitoly byly blíže dokumentovány experimenty testující implementovaný nástroj. Byla provedena validace nástroje testující přítomnost falešných pozitiv. Z dalších provedených experimentů vyplynulo, že nástroj je schopen detekovat všechny hrozby, u nichž byl navržen způsob detekce s výjimkou detekce algoritmů DGA, kde byly nalezeny limitace zvolené detekční metody. U všech experimentů proběhlo zdůvodnění případných falešných pozitiv během detekce. Experimenty byla zdokumentována závislost rychlosti detekčního nástroje na velikosti vstupních souborů.

Soubor	Velikost	Čas - norm.	Čas - bez DGA	Čas - strikt.
benign-flowmon.csv	440,6 MB	1 m 39 s	1 m 39 s	1 m 39 s
dns-2023041710.csv	963,9 MB	3 m 24 s	4 m 49 s	3 m 40 s
dns-2023041711.csv	1094,7 MB	4 m 8 s	4 m 19 s	4 m 26 s
dns-2023041712.csv	1359,8 MB	5 m 34 s	5 m 6 s	5 m 12 s
dns-2023041713.csv	1531,7 MB	5 m 52 s	5 m 41 s	5 m 52 s
dns-2023041714.csv	1479,2 MB	5 m 30 s	5 m 14 s	5 m 26 s
dns-2023041715.csv	1716,3 MB	6 m 26 s	6 m 21 s	6 m 30 s
dns-2023041716.csv	1853,0 MB	7 m 3 s	6 m 46 s	6 m 54 s
dns-2023041717.csv	1747,2 MB	6 m 21 s	6 m 19 s	6 m 30 s
dns-2023041718.csv	1434,9 MB	5 m 25 s	5 m 11 s	5 m 30 s
dns-2023041719.csv	1139,0 MB	4 m 19 s	4 m 11 s	4 m 16 s
dns-2023041720.csv	547,9 MB	2 m 3 s	1 m 59 s	2 m 3 s
dns-2023041721.csv	365,3 MB	1 m 23 s	1 m 22 s	1 m 25 s
dns-2023041722.csv	294,0 MB	1 m 8 s	1 m 5 s	1 m 8 s
dns-2023041723.csv	239,1 MB	53,687 s	52,199 s	54,877 s
dns-2023041800.csv	210,3 MB	47,374 s	45,955 s	46,749 s
dns-2023041801.csv	306,3 MB	1 m 11 s	1 m 11 s	1 m 13 s
dns-2023041802.csv	418,0 MB	1 m 33 s	1 m 34 s	1 m 35 s
dns-2023041803.csv	528,9 MB	2 m 2 s	1 m 55 s	1 m 59 s
dns-2023041804.csv	703,0 MB	2 m 47 s	2 m 43 s	2 m 40 s
dns-2023041805.csv	758,6 MB	2 m 51 s	3 m 3 s	2 m 49 s
dns-2023041806.csv	817,4 MB	3 m 9 s	3 m 21 s	3 m 2 s
dns-2023041807.csv	789,0 MB	2 m 58 s	2 m 53 s	2 m 58 s
dns-2023041808.csv	855,3 MB	3 m 12 s	3 m 9 s	3 m 12 s
dns-2023041809.csv	816,7 MB	2 m 59 s	2 m 58 s	3 m 4 s
dns-2023041810.csv	831,0 MB	3 m 13 s	3 m 0 s	3 m 5 s
dns-2023041811.csv	863,4 MB	3 m 16 s	3 m 19 s	3 m 16 s
dns-2023041812.csv	969,8 MB	3 m 47 s	3 m 34 s	3 m 40 s
dns-2023041813.csv	921,5 MB	3 m 29 s	4 m 24 s	3 m 31 s
dns-2023041814.csv	1106,6 MB	4 m 20 s	4 m 16 s	4 m 8 s
dns-2023041815.csv	1142,8 MB	4 m 30 s	5 m 34 s	4 m 19 s
dns-2023041816.csv	1204,5 MB	4 m 40 s	4 m 43 s	4 m 40 s
dns-2023041817.csv	1062,2 MB	4 m 2 s	3 m 53 s	4 m 1 s
dns-2023041818.csv	1013,7 MB	3 m 50 s	4 m 7 s	3 m 52 s
dns-2023041819.csv	812,9 MB	3 m 5 s	2 m 58 s	3 m 4 s
dns-2023041820.csv	644,6 MB	2 m 22 s	2 m 22 s	2 m 23 s
dns-2023041821.csv	426,6 MB	1 m 35 s	1 m 32 s	1 m 34 s
dns-2023041822.csv	282,5 MB	1 m 3 s	1 m 2 s	1 m 4 s
dns-2023041823.csv	240,8 MB	55,003 s	52,896 s	56,576 s
dns-2023041900.csv	180,1 MB	39,166 s	38,814 s	39,736 s
dns-2023041901.csv	176,8 MB	40,374 s	40,671 s	41,369 s
dns-2023041902.csv	200,0 MB	46,034 s	45,402 s	46,527 s
dns-2023041903.csv	285,6 MB	1 m 3 s	1 m 1 s	1 m 2 s
dns-2023041904.csv	344,1 MB	1 m 16 s	1 m 15 s	1 m 18 s
dns-2023041905.csv	435,3 MB	1 m 41 s	1 m 37 s	1 m 38 s
dns-2023041906.csv	531,0 MB	1 m 50 s	1 m 47 s	1 m 59 s
dns-2023041907.csv	638,6 MB	2 m 39 s	2 m 23 s	2 m 26 s
dns-2023041908.csv	668,0 MB	2 m 27 s	2 m 24 s	2 m 31 s
dns-2023041909.csv	660,2 MB	2 m 27 s	2 m 23 s	2 m 31 s
dns-2023041910.csv	609,0 MB	2 m 26 s	2 m 22 s	2 m 19 s
heavy_compressed-flowmon.csv	14,7 MB	1,372 s	1,372 s	1,372 s
heavy_exe-flowmon.csv	15,9 MB	1,462 s	1,462 s	1,462 s
heavy_text-flowmon.csv	14,1 MB	1,384 s	1,384 s	1,384 s
heavy_video-flowmon.csv	16,2 MB	1,435 s	1,435 s	1,435 s
light_audio-flowmon.csv	4,0 MB	1,028 s	1,028 s	1,028 s
light_exe-flowmon.csv	5,7 MB	1,077 s	1,077 s	1,077 s
light_image-flowmon.csv	0,5 MB	0,844 s	0,844 s	0,844 s
light_text-flowmon.csv	2,4 MB	0,929 s	0,929 s	0,929 s
light_video-flowmon.csv	4,3 MB	1,034 s	1,034 s	1,034 s
malware-flowmon.csv	96,1 MB	30,099 s	30,099 s	30,099 s
phishing-flowmon.csv	51,9 MB	14,855 s	14,855 s	14,855 s
spam-flowmon.csv	33,1 MB	9,858 s	9,858 s	9,858 s

Tabulka 5.8: Přehled rychlostí zpracování jednotlivých souborů při různých režimech.

Kapitola 6

Závěr

Práce naplnila vytýčené cíle popsané v úvodu. Nejdříve jsme provedli rešerši konkrétních hrozeb v systému DNS. V rámci práce byly tyto hrozby rozčleněny do čtyř kategorií: exfiltrační útoky, infiltrační útoky, útoky využívající chyb v systémech DNS a techniky ke zvýšení účinnosti útoku.

V rámci kapitoly 3 jsme nejdříve popsali výsledky rešerše datových sad souvisejících se systémem DNS a popsali způsoby, jakými se povedlo implementovat či využít již existujícího nástroje pro simulaci jednotlivých útoků. Při rešerši dostupných datových sad se však nepodařilo najít reálný záchyt komunikace většiny útoků popsaných v rešerši. Důvodem může být, že tyto útoky cílily především na úzkou skupinu organizací. Existující nástroje jsme získali pouze k hrozbám DNSMessenger, SIGRed a algoritmům DGA. K malwaru Alina POS jsme vytvořili původní nástroj v rámci této práce. Uvedené hrozby jsme následně vybrali k návrhu jejich detekce. Na závěr kapitoly popisujeme vytvořené datové sady pro účely testování implementovaného detekčního nástroje.

V rámci kapitoly 4 jsme provedli zhodnocení logování na serveru BIND 9, monitorování IPFIX se sondou Flowmon a systému IDS Suricata vzhledem k jejich vhodnosti k detekci hrozeb Alina POS, DNSMessenger, SIGRed a algoritmy DGA. Z provedených experimentů s logováním na serveru BIND 9 jsme zjistili, že s pomocí tohoto způsobu monitorování lze úspěšně detekovat útoky Alina POS a algoritmy DGA. Protože logování neposkytuje informace k odpovědi na dotaz DNS, tak není s jeho pomocí možné detekovat útoky DNSMessenger a SIGRed. Po provedení podobného experimentu s monitorováním IPFIX se sondou Flowmon jsme zjistili, že systém poskytuje dostatečné množství informací k detekci všech vybraných hrozeb. Narazili jsme však na problém se zpracováním záznamu SIG při útoku SIGRed, při němž bylo pole s odpovědí prázdné. Pro systémy IDS Suricata a Snort jsme navíc provedli rešerši dostupných pravidel pro detekci výše zmíněných útoků. Z výsledků této rešerše jsem dospěli k tomu, že systém IDS Suricita je schopen detekce výše zmiňovaných hrozeb, jelikož k nim byla nalezena dostupná pravidla. Dále jsme navrhli a popsali původní způsoby detekce hrozeb Alina POS, DNSMessenger, SIGRed, které jsme použili k implementaci detekčního nástroje. K algoritmům DGA jsme provedli rešerši dostupných detekčních nástrojů a jejich metod a současně navrhli původní detekční metodu, která byla použita k implementaci detekčního nástroje. S návrhem jsme také představili limitace detekčních možností navržené metody.

V kapitole 5 této práce jsme představili návrh detekčního nástroje, jež byl koncipován jako maximálně modulární pro jeho snadnou rozšiřitelnost. Konkrétně jsme jej rozdělili na vstupní modul, detekční modul, statistický modul a výstupní modul. Dále jsme blíže popsali implementaci detekčního nástroje. Zdokumentovali jsme unifikaci vstupní dat, s pomocí

které vstupní modul sjednotí data komunikace z různých monitorovacích systémů. Také jsme nadefinovali strukturu pro signatury útoků, s kterými následně navržený detekční algoritmus testuje vstupní data. Dále jsme provedli sadu experimentů za různých podmínek. Z provedených experimentů jsme zjistili, že nástroj byl validován vždy až na případ, kdy nebyla vyplněna všechna potřebná pole v záznamu o toku DNS. To nastalo pouze v případě jednoho ze souborů datové sady CIC-Bell-DNS-EXF-2021, kdy byl označen tok se záznamem TXT za hrozbu DNSMessenger. V dalších experimentech jsme ověřovali schopnost detekce hrozeb Alina POS, DNSMessenger a SIGRed nad vytvořenými datovými sadami. Z výsledků tohoto experimentu jsme zjistili, že nástroj je schopen detekovat při monitorovaní IPFIX na sondě Flowmon všechny tři hrozby v normálním režimu a Alina POS s DNSMessenger ve striktním režimu. V případě logování na serveru BIND 9 v normálním režimu detekční nástroj nalezl hrozby Alina POS a DNSMessenger, nikoliv však SIGRed a v případě striktního režimu pouze hrozbu Alina POS. Poté jsme přistoupili k experimentům testujícím původní detekční metodu algoritmů DGA. Dospěli jsme k závěru, že nejvyšší přesnost vykazuje nejmenší možná velikost slovníkového řetězce v doménovém jméně o hodnotě čtyři. Dalšími experimenty jsme zdokumentovali závislost rychlosti detekčního nástroje na velikosti vstupních souborů. Z výstupů experimentu jsme zjistili, že průměrná rychlost zpracování je nezávislá na režimu detekce.

Na práci by bylo možné navázat v oblasti rozšíření podporovaných hrozeb. Takové rozšíření je právě díky modularitě nástroje poměrně snadné, jelikož stačí vytvořit novou signaturu dle definované struktury. Taktéž je možné nástroj jednoduše rozšířit o podporu jiných monitorovacích systémů. K zvýšení přesnosti detekce algoritmů DGA je možné budoucí zapojení strojového učení či expandovaní slovníku o další cizojazyčná slova a zkratky. Pro akceleraci detekce je možné optimalizovat algoritmus tak, aby nebylo nutné sekvenčně procházet všechny signatury, a tím tak snížit jeho časovou složitost.

Literatura

[1] *Český národní korpus: Abecední a retrográdní slovníky.* Ústav Českého národního korpusu FF UK, 2016.

[2] *Newly Observed Domains (NOD)* [online]. 2018 [cit. 2022-10-09]. Dostupné z: `https://www.farsightsecurity.com/assets/media/download/Farsight_NOD_Overview.pdf`.

[3] *CVE-2020-11901.* National Vulnerability Database, červen 2020.

[4] *CVE-2020-1350.* National Vulnerability Database, červenec 2020.

[5] *BIND 9 Administrator Reference Manual: Release 9.19.9-dev* [online]. Internet Systems Consortium, prosinec 2022 [cit. 2022-12-23]. Dostupné z: `https://bind9.readthedocs.io/en/v9.19.9/`.

[6] *Suricata User Guide: Release 6.0.10* [online]. Open Information Security Foundation, leden 2023 [cit. 2022-12-23]. Dostupné z: `https://suricata.readthedocs.io/en/suricata-6.0.10/`.

[7] Aho, A. a Corasick, M. Efficient string matching: An aid to bibliographic search. *Commun. ACM.* Červen 1975, sv. 18, s. 333–340. DOI: 10.1145/360825.360855.

[8] Aitken, P., Claise, B. a Trammell, B. *Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information.* IETF RFC 7011, září 2013.

[9] Bader, J. *The Domain Generation Algorithms of SharkBot* [online]. Binary Reverse Engineering Blog, červen 2022 [cit. 2022-12-23]. Dostupné z: `https://bin.re/blog/the-dgas-of-sharkbot/`.

[10] Bellis, R. *DNS Transport over TCP – Implementation Requirements.* IETF RFC 5966, srpen 2010.

[11] Black Lotus Labs. *Alina Point Of Sale Malware Still Lurking In DNS* [online]. Lumen, červenec 2020 [cit. 2022-10-07]. Dostupné z: `https://blog.lumen.com/alina-point-of-sale-malware-still-lurking-in-dns/`.

[12] Brumaghin, E. a Grady, C. *Covert Channels and Poor Decisions: The Tale of DNSMessenger* [online]. Talos, březen 2017 [cit. 2022-10-09]. Dostupné z: `https://blog.talosintelligence.com/2017/03/dnsmessenger.html`.

[13] Brumaghin, E. a Grady, C. *Spoofed SEC Emails Distribute Evolved DNSMessenger* [online]. Talos, říjen 2017 [cit. 2022-10-09]. Dostupné z: `https://blog.talosintelligence.com/2017/10/dnsmessenger-sec-campaign.html`.

[14] Casino, F., Lykousas, N., Homoliak, I., Patsakis, C. a Hernandez Castro, J. *HYDRA dataset (1.0)* [dataset]. Zenodo, červenec 2020. Dostupné z: https://doi.org/10.5281/zenodo.3965397.

[15] Cornea, C. *Data Exfiltration over DNS Queries via Morse Code* [online]. Duben 2020 [cit. 2022-10-05]. Dostupné z: https://corneacristian.medium.com/data-exfiltration-over-dns-queries-via-morse-code-efc9e09f56fe.

[16] Daihes, Y., Tzaban, H., Nadler, A. a Shabtai, A. MORTON: Detection of Malicious Routines in Large-Scale DNS Traffic. In: Bertino, E., Shulman, H. a Waidner, M., ed. *Computer Security – ESORICS 2021*. Cham: Springer International Publishing, 2021, s. 736–756. ISBN 978-3-030-88418-5.

[17] Deutsch, P. *GZIP file format specification version 4.3*. IETF RFC 1952, květen 1996.

[18] Eastlake, D. *Domain Name System Security Extensions*. IETF RFC 2535, březen 1999.

[19] Foremski, P. a Vixie, D. P. The Modality of Mortality In Domain Names. In: Farsight Security. *Virus Bulletin Conference*. říjen 2018.

[20] Gillis, A. S. *Domain Generation Algorithm (DGA)* [online]. TechTarget, červenec 2021 [cit. 2022-10-09]. Dostupné z: https://www.techtarget.com/searchsecurity/definition/domain-generation-algorithm-DGA.

[21] Global Research and Analysis Team. *The ProjectSauron APT* [online]. Kaspersky Lab, srpen 2016 [cit. 2022-10-05]. Dostupné z: https://securelist.com/faq-the-projectsauron-apt/75533/.

[22] Grill, M., Nikolaev, I., Valeros, V. a Rehak, M. Detecting DGA malware using NetFlow. *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*. 2015, s. 1304–1309. DOI: 10.1109/INM.2015.7140486.

[23] Hinchliffe, A. *DNS Tunneling: how DNS can be (ab)used by malicious actors* [online]. Unit 42, březen 2019 [cit. 2022-10-07]. Dostupné z: https://unit42.paloaltonetworks.com/dns-tunneling-how-dns-can-be-abused-by-malicious-actors/.

[24] Hofstede, R., Čeleda, P., Trammell, B., Drago, I., Sadre, R. et al. Flow Monitoring Explained: From Packet Capture to Data Analysis With NetFlow and IPFIX. *IEEE Communications Surveys & Tutorials*. 2014, sv. 16, č. 4, s. 2037–2064. DOI: 10.1109/COMST.2014.2321898.

[25] Hromcová, Z. a Cherepanov, A. *InvisiMole: The Hidden Part of the Story: Unearthing InvisiIMole's Espionage Toolset and Strategic Cooperations* [online]. Eset, červen 2020 [cit. 2022-10-07]. Dostupné z: https://www.welivesecurity.com/wp-content/uploads/2020/06/ESET_InvisiMole.pdf.

[26] Khraisat, A., Gondal, I., Vamplew, P. a Kamruzzaman, J. *Survey of intrusion detection systems: techniques, datasets and challenges*. červenec 2019. DOI: 10.1186/s42400-019-0038-7. ISSN 2523-3246.

[27] Kol, M., Schön, A. a Oberman, S. *Ripple 20 CVE-2020-11901.* JSOF Research Lab, srpen 2020.

[28] Krčmář, P. *Tunelujeme provoz pomocí DNS: cesta ven ze sítě.* Root.cz, květen 2022.

[29] Mahdavifar, S., Maleki, N., Lashkari, A. H., Broda, M. a Razavi, A. H. *Classifying Malicious Domains using DNS Traffic Analysis.* Calgary, Kanada: The 19th IEEE International Conference on Dependable, Autonomic, and Secure Computing (DASC), říjen 2018.

[30] Mahdavifar, S., Salem, A. H., Victor, P., Garzon, M., Razavi, A. H. et al. *Lightweight Hybrid Detection of Data Exfiltration using DNS based on Machine Learning.* Beijing Jiaotong University, Weihai, Čína: The 11th IEEE International Conference on Communication and Network Security (ICCNS), prosinec 2018.

[31] Matoušek, P. *Síťové aplikace a jejich architektura.* Brno: VUTIUM, 2014. ISBN 978-80-214-3766-1.

[32] Mockapetris, P. *Domain Names – Implementation and Specification.* IETF RFC 1035, listopad 1987.

[33] Parker, M. *Collection of Pcap files from malware analysis* [online]. únor 2015 [cit. 2022-12-20]. Dostupné z: `https://contagiodump.blogspot.com/2013/04/collection-of-pcap-files-from-malware.html`.

[34] Stratosphere. *Stratosphere Laboratory Datasets* [dataset]. 2015. Dostupné z: `https://www.stratosphereips.org/datasets-overview`.

[35] Torabi, S., Boukhtouta, A., Assi, C. a Debbabi, M. Detecting Internet Abuse by Analyzing Passive DNS Traffic: A Survey of Implemented Systems. *IEEE Communications Surveys & Tutorials.* 2018, sv. 20, č. 4, s. 3389–3415. DOI: 10.1109/COMST.2018.2849614.

[36] Tzadik, S. *SIGRed – Resolving Your Way into Domain Admin: Exploiting a 17 Year-old Bug in Windows DNS Servers.* Chech Point Research, červenec 2020.

[37] University Of Southern California – Information Sciences Institute. *DoS_DNS_amplification-20130617* [dataset]. IMPACT, 2014. DOI: 10.23721/109/1353940.

[38] Zhang, E. *What is Point-of-Sale (POS) Malware? How It Works and How to Protect Your POS System* [online]. Digital Guardian, listopad 2017 [cit. 2022-10-07]. Dostupné z: `https://digitalguardian.com/blog/what-point-sale-pos-malware-how-it-works-and-how-protect-your-pos-system`.

Příloha A

Princip systému DNS

Jak již bylo řečeno v úvodu této práce, tak systém DNS (Domain Name System) hraje naprosto stěžejní roli v dnešním internetu. Bez jeho funkcionality bychom nemohli využívat naprostou většinu služeb, které jsou na internetu nabízeny.

Zřejmě nejrozšířenější službou tohoto systému je bezesporu překlad doménových jmen na IP adresy. Číselné adresy, jakožto jednoznačný identifikátor zařízení v síti, jsou mnohem více vhodné než například textové řetězce vzhledem ke strojovému zpracovávání. Pro běžného uživatele by bylo však problémové si zapamatovat 32bitové číslo v případě adresy IPv4 či dokonce 128bitové číslo u adresy IP verze 6. Proto existuje databáze doménových jmen, které jsou mnohem uživatelsky přívětivější pro zapamatování a služba DNS, která tyto jména mapuje na zmíněné číselné adresy.

Mimo tuto službu však systém DNS nabízí mnoho dalších, například opačný překlad z adresy IP na doménové jméno, překlad aliasů zařízení či určení poštovních serverů pro danou doménu.

V této příloze je v první části vysvětlen pojem doména, její kategorizování a registrování společně s ukázkou formátu doménového jména. Poté je detailněji popsán princip komunikace v rámci systému DNS a zařízení, které jsou k této komunikaci potřeba. Na závěr této přílohy je objasněn síťový protokol, které služby DNS využívají k jejich funkčnosti, a na závěr jsou uvedeny typy záznamů, které patřičné informace z již zmíněné databáze nesou.

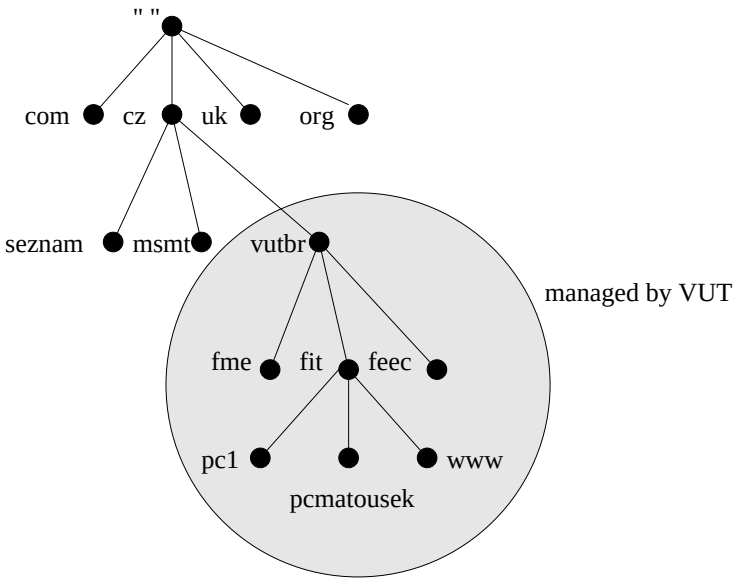
V rámci této kapitoly se vyskytují především převzaté informace z [31] a RFC standardů 1035 [32] a 2535 [18].

A.1 Doména a doménové jméno

Hlavními pojmy spjatými se systémem DNS jsou *doména* a *doménové jméno*. V následující části bude tato problematika přiblížena.

Jak již bylo řečeno, systém DNS je rozsáhlou databází různých dat spojených s doménovými jmény. Pokud si tuto databázi představíme jako kořenový strom, kde každý uzel má jednoho předchůdce a může mít několik následovníků, tak doména je pak jeho podstromem. Zjednodušeně řečeno tedy jde o soubor uzlů, které mají jeden společný kořen v rámci tohoto podstromu. Vizualizace tohoto stromu je uvedena na obrázku A.1, kde šedou barvou je označena doména `vutbr`.

Doménovým jménem pak rozumíme cestu v tomto grafu od uzlu ke kořeni, respektive názvy jednotlivých uzlu zapsaných za sebou a odděleném znakem tečky. Ukázka takového doménového jména může být následující:



Obrázek A.1: Příklad uspořádání doménových jmen v kořenovém stromě [31].

`www.fit.vutbr.cz.`

V tom případě se jedná o plné doménové jméno (Fully Qualified Domain Name), jelikož obsahuje i kořenovou doménu, jejíž název je prázdný řetězec — z toho plyne tečka na konci. Ta se však ve většině případů nezadává, jelikož při překladu doménového jména je kořenová doména doplněna vždy, viz specifikace protokolu DNS v kapitole A.3.

Každému jménu oddělenému tečkou v tomto jméně, jako například `www`, `fit` či dalším, se říká subdoména (angl. label), jež může být složena dle standardu [32] z maximálně 63 znaků a nesmí obsahovat jiné znaky než písmena anglické abecedy, číslice a pomlčku. Pro doménové jméno dále platí, že může dohromady obsahovat maximálně 255 znaků včetně oddělovačů, a to i oddělovače názvu kořenové domény. Taktéž každá subdoména nemůže začínat ničím jiným než písmenem a musí končit písmenem nebo číslicí. Co se týče relevantnosti malých či velkých písmen, tak v tom případě je ignorována — angl. case-insensitive.

Těchto výše zmíněných limitů využívají útočníci, jak je blíže popsáno v kapitole 2, k připojení různých informací či konkrétní dat přímo ke zbytku doménového jména.

A.1.1 Kategorie domén

Jednotlivé domény pojmenováváme a kategorizujeme dle jejich vzdálenosti od kořenové domény. Pokud je doména ve vzdálenosti jedna od kořenové domény, nazýváme ji jako *doménu prvního řádu* (Top Level Domain, zkráceně TLD). Mezi takové domény patří tzv. národní domény (country code TLD), např. `cz`, `sk` či `de`, nebo generické domény (generic TLD), do nich patří např. `com`, `edu`, `net` a další.

Doménám pod doposud zmíněnými říkáme *domény druhého řádu*, a tak postupně pokračujeme.

A.1.2 Správa a registrace domén

K tomu, aby mohli útočníci realizovat škodlivou aktivitu zneužívající systém DNS, musí k tomuto účelu ve většině případů vlastnit nějakou konkrétní doménu.

Hlavní roli ve správě stávajících a registraci nových domén hraje organizace ICANN (Internet Corporation for Assigned Names and Numbers), která má za úkol spravovat, přidělovat a ukládat doménové jména do této distribuované databáze. Musí tedy zajistit, aby v doménách stejného řádu nevznikaly duplicitní názvy, a tudíž bylo každé doménové jméno unikátní.

Jelikož je však tento databázový prostor značně rozsáhlý, je organizace ICANN nucena delegovat zmíněné úkoly na tzv. akreditované registrátory doménových jmen. Sama tedy pak spravuje domény nejvyšší úrovně a ty nižší ponechává dalším organizacím.

Pro správu domén v České republice je zvolena organizace CZ-NIC. I ta však dále deleguje přidávání nových záznamů na jí oprávněné registrátory.

A.2 Způsob komunikace v systému DNS

Systém DNS je komplexní soubor řady komponent, které dohromady zajišťují služby, jež tento systém nabízí. Konkrétně je složen ze tří hlavních prvků: serverů DNS, resolverů DNS a samotného prostoru doménových jmen. O prvních dvou je v této části kapitoly podán krátký popis jejich významu v systému DNS. Taktéž je zde zmíněna jejich role v samotné komunikaci, která je základem pro fungovaní tohoto systému, ale taktéž terčem zneužití pro útočníky.

A.2.1 Servery DNS

Servery DNS plní v systému DNS důležitou roli uchovávání části dat této velké distribuované databáze. Ve své podstatě jsou to aplikace, které mají za úkol odpovídat na dotazy, které k nim doputují. Na tyto dotazy pak server odpovídá záznamy DNS, které má lokálně uložené v souboru.

Server v systému DNS může poskytnout dva typy odpovědí:

- autoritativní – odpovídá informacemi, které jsou pod jeho správou, ať už z role tzv. primárního serveru DNS, který jako jediný drží přesné informace o dané doméně nebo sekundárního serveru DNS, který tyto informace přejímá od primárního serveru,
- neautoritativní – odpovídá informacemi, které si předem uložil do paměti při již v minulost kladeném dotazu. Těmto serverům se říká záložní servery DNS a slouží tedy jako proxy server pro zrychlení procesu vyhledávání informací v databázi DNS.

A.2.2 Resolvery DNS

Resolvery v systému DNS pak plní funkci samotného dotazování a předávání odpovědí zpět tomu, kdo informaci požadoval. Jedná se tedy o klientskou aplikaci, která ostatním aplikacím, které vyžadují určité informace z databáze DNS, takové informace dodá posláním dotazu, interpretací doručené odpovědi a poskytnutím patřičné informace.

Aby tuto funkcionalitu mohl vykonávat, je zapotřebí, aby měl k dispozici adresu alespoň jednoho serveru DNS. Ten mu buď odpověď poskytne nebo mu dá informaci, kde se má jinde doptat.

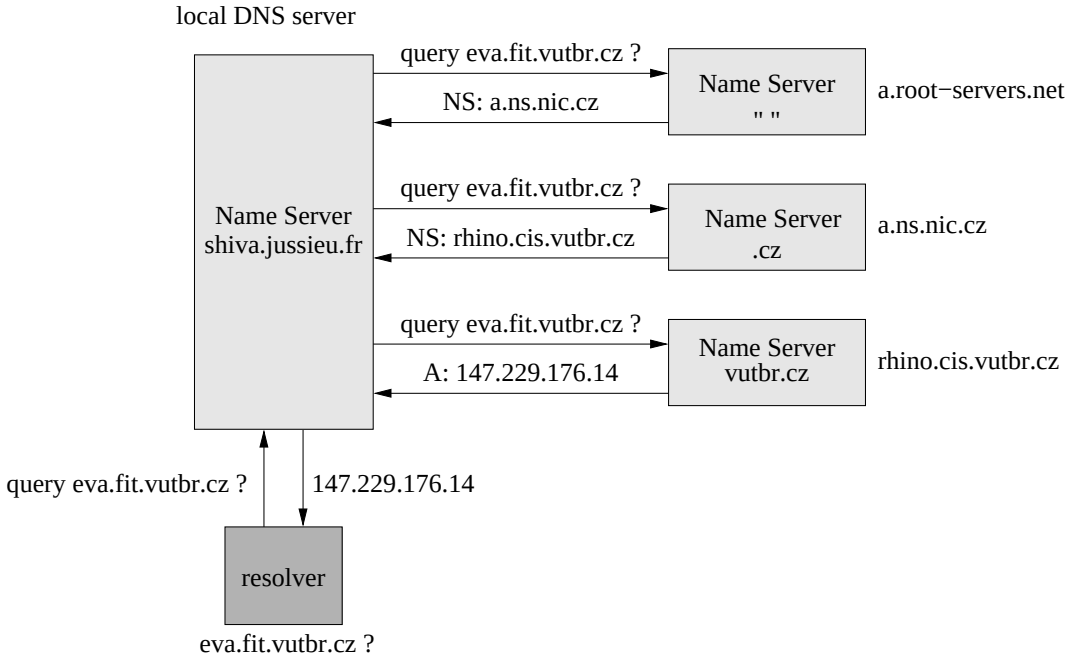
A.2.3 Popis rezoluce

Pojmem rezoluce obecně ve vztahu k systému DNS rozumíme proces, při kterém dochází k vyhledávání určité informace v databázi DNS. K tomu, aby takové vyhledávání mohlo být provedeno, stačí znát pouze adresu IP jednoho z autoritativních serverů kořenové domény. Postupně tak můžeme od kořene stromu databáze procházet k listům a získat tak libovolnou informaci.

Princip komunikace v rámci rezoluce

Rezoluce je základním komunikačním procesem v systému DNS, která je využívaná u většiny hrozeb, jež jsou blíže popsány v kapitole 2.

Na následující příkladu A.2 lze vidět ukázku rezoluce, kdy resolver hledá odpověď k dotazu na záznam A doménového jména `eva.fit.vutbr.cz`, tedy hledá adresu IP tohoto doménového jména.



Obrázek A.2: Schéma komunikace jednotlivých serverů při rezoluci [31].

V rámci této ukázky je uvažováno, že odpověď na tento dotaz nemá resolver a ani žádný ze serverů DNS uloženou ve své paměti cache. Je tedy nutné projít celým stromem databáze DNS.

1. Resolver na začátku přeposílá dotaz na nakonfigurovaný server DNS na záznam A, tedy v tomto případě na server `shiva.jussieu.fr`.

2. Tento server vykonává tzv. rekurzivní dotazování, tudíž se dotazuje do té doby, dokud nezíská odpověď, či případně odpoví tázajícímu, že ji nezná. Na začátku však zkontroluje, jestli pro toto doménové jméno nemá uložený záznam. Jelikož v této situaci nemá, tak se dotáže na záznam A kořenového serveru DNS `a.root-servers.net`.

3. Jelikož tento kořenový server nezná odpověď na toto konkrétní doménové jméno, tak se pokusí vrátit nejlepší možnou odpověď — vykonává tzv. iterativní dotazování. Na toto doménové jméno se tedy podívá odzadu a zjišťuje, že zná adresu autoritativního serveru pro doménu TLD `cz`, kterou vrací serveru `shiva.jussieu.fr`.

4. Server `shiva.jussieu.fr` se tedy pokouší poslat stejný dotaz na záznam A serveru `a.ns.nic.cz`.

5. Autoritativní server `a.ns.nic.cz` pro doménu `cz` však opět nezná odpověď na toto doménové jméno a pošle tedy nejlepší možnou odpověď: adresu autoritativního serveru pro doménu `vutbr`.

6. Po obdržení adresy autoritativního serveru domény `vutbr` `rhino.cis.vutbr.cz` je na něj opět odeslán dotaz na záznam A.

7. Tento server již zná odpověď, respektive má uložený záznam typu A, který vrací serveru `shiva.jussieu.fr`.

8. Při obdržení odpovědi v podobě záznamu A svoje dotazování server `shiva.jussieu.fr` končí a vrací tuto odpověď resolveru.

V případě, že by jakýkoliv ze serverů při procesu rezoluce měl dotazovanou informaci uloženou v paměti cache, tak ji využije a vrátí tak odpověď, aniž by bylo nutné procházet všechny patřičné úrovně stromu databáze DNS. Jak již ale bylo zmíněno v kapitole A.2.1, tak taková odpověď by byla považována za neautoritativní, a tedy nemusí být aktuální. U útoků DNS k tomuto jevu však obvykle nedojde, jelikož je doménové jméno téměř vždy unikátní.

Způsob získávání informací

Jelikož naprostá většina síťových aplikací potřebuje ke svému provozu informace z databáze DNS, nejčastěji kvůli překladu doménového jména na adresu IP, je nutné, aby využily služeb resolveru implementovaného přímo v operačním systému. Tyto aplikace však dotazy pokládají samy a uživatel tak o nich mnohdy ani nemusí vědět.

Pokud však chceme získat informaci v rámci systému DNS přímo, lze k tomu využít nástrojů jako například `nslookup`, `dig` či `host`. Při útocích, které tento systém zneužívají, jsou mnohdy tyto nástroje využívány k odesílání či přijímají dat.

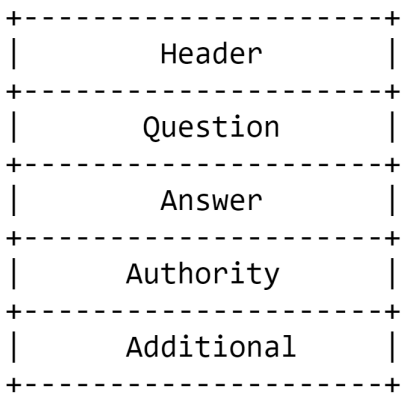
A.3 Specifikace protokolu DNS

Systém DNS ke komunikaci využívá protokolu DNS, jehož pakety jsou základním nositelem informací z databáze DNS. Pro funkci dotazování byl protokol DNS navržen tak, aby využíval transportního protokolu UDP, jelikož většina informací se vejde právě do jednoho paketu UDP, tedy 512 bajtů. Protokol DNS však podporuje i využití transportního protokolu TCP v případě, že je potřeba v rámci komunikace přenést větší množství dat, například u přenášení informací mezi autoritativními servery.

Jelikož transportní protokol UDP nezajišťuje spolehlivost komunikace, tak v případě ztráty paketu je nutné jeho opětovné odeslání.

A.3.1 Položky paketu DNS

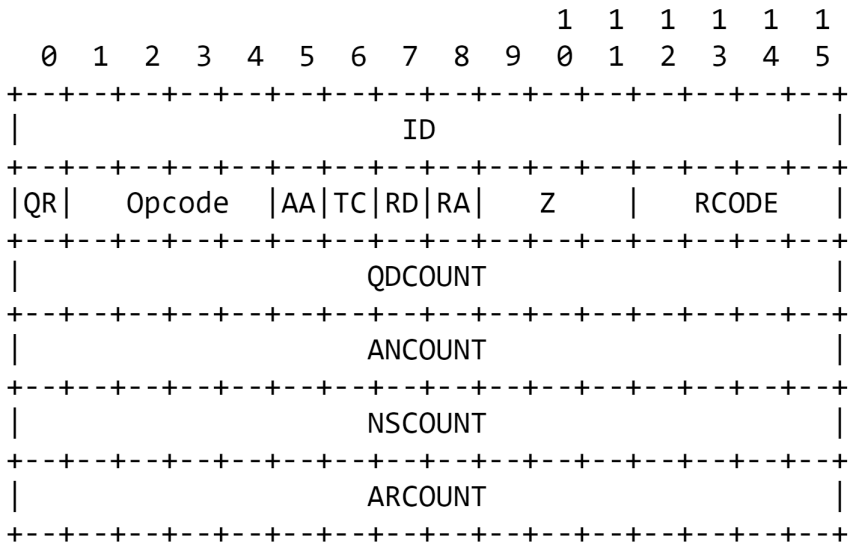
Na obrázku A.3 je možné vidět pět základních složek, které tvoří strukturu paketu DNS.



Obrázek A.3: Struktura paketu DNS [32].

Zde následuje popis těchto jednotlivých části a jejich význam v rámci paketu DNS.

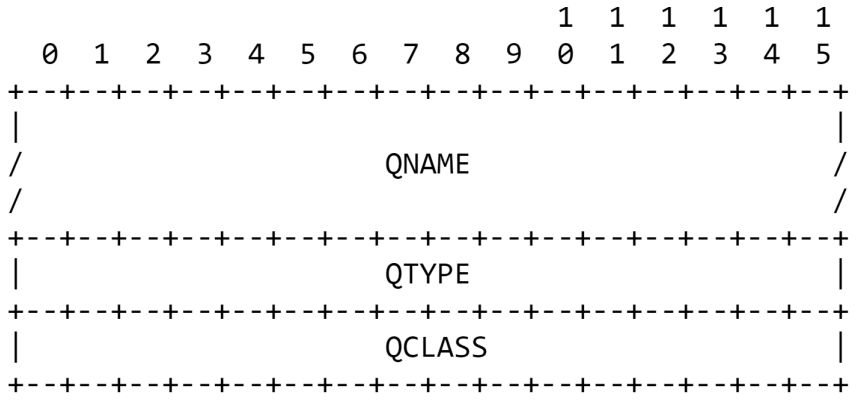
1. Hlavička (Header) – je základním prvkem paketu DNS, bez které se nelze obejít. Její strukturu je možné vidět na obrázku A.4. Každá hlavička začíná jednoznačným identifikátorem `ID`, díky kterému může resolver poznat odpověď na daný dotaz. Dále obsahuje řadu příznaků jako například `QR` pro rozlišení dotazu a odpovědi, příznak `TC` (TrunCation), jež se hojně využívá při zahajování komunikace přes TCP, příznaky `RD` a `RA` pro identifikaci podpory rekurzivního dotazování či `RCODE` pro určení případné chybovosti odpovědi.



Obrázek A.4: Struktura hlavičky paketu DNS [32].

2. Dotaz (Question) – obsahuje, jak je možno vidět na obrázku A.5, pole `QNAME`, ve kterém se nachází doménové jméno, na které je kladen dotaz. Dalšími elementy jsou `QTYPE` pro označení žádaného záznamu pro toto doménové jméno a `QCLASS` pro zvolení

třídy dotazu. Ve většině případů v tomto posledním poli najdeme hodnotu jedna označující třídu Internet.



Obrázek A.5: Struktura dotazu paketu DNS [32].

3. Odpověď (Answer) – je předmětem následující kapitoly A.4, kde je blíže popsána její obecná struktura a jednotlivé typy záznamů.
4. Sekce pro informace o záznamech NS (Authority)
5. Sekce pro doplňující informace (Additional)

A.3.2 Komprese v paketech DNS

K docílení největšího možného využití paketu DNS zasílaného nad transportním protokolem UDP je zavedena v rámci protokolu DNS tzv. komprese. Ta využívá skutečnosti, že se v paketu DNS mnohdy nachází stejné doménové jméno, nebo alespoň jeho část, několikrát. Zneužitím níže popsaného principu komprese však může dojít, jak je zmíněno v kapitole 2, k pádu napadeného systému či v krajních případech až zmocnění se nad daným zařízením.

V první řadě je však důležité zmínit, jakým způsobem jsou uložena doménová jména v paketu DNS. Oddělovače, o kterých byla řeč v kapitole A.1, v doménovém jméně se z čitelné podoby převádí na číselnou hodnotu vyjadřující počet znaků subdomény, respektive počet znaků do dalšího oddělovače. Každé doménové jméno je pak zakončeno bajtem o hodnotě nula vyjadřující délku názvu kořenové domény. Pokud tedy vezmeme příklad doménového jména z kapitoly A.1

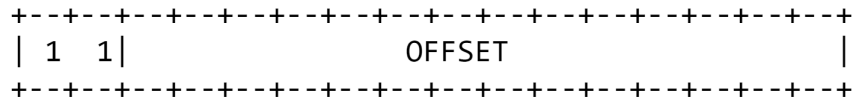
`www.fit.vutbr.cz`,

dostáváme, vyjádřeno v ASCII podobě, po převedení do formy v rámci paketu DNS:

`3www3fit5vutbr2cz0`.

Aby bylo možné již jednou definované doménové jméno v paketu DNS znovu použít nebo alespoň jeho část, tak je využíván ukazatel, jehož formát lze vidět na obrázku A.6.

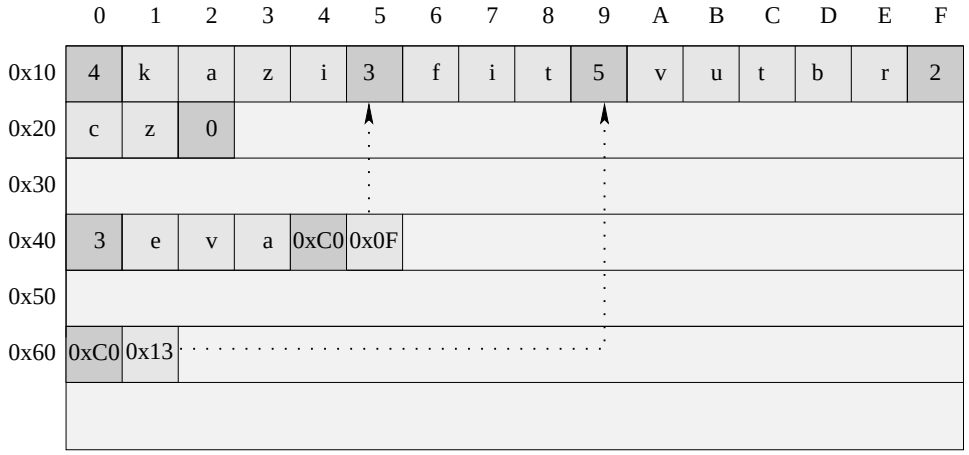
Jedná se o 16bitový identifikátor, který má vždy první dva nejvíce významné bity nastavené na hodnotu jedna a pole `OFFSET`, které drží hodnotu vzdálenosti od začátku paketu



Obrázek A.6: Struktura ukazatele v paketu DNS [32].

DNS, kde se nachází doménové jméno k znovupoužití. Právě díky dvou prvním bitům o hodnotě jedna je možné rozlišit ukazatel od běžného bajtu s délkou subdomény.

Praktická ukázka využití ukazatelů ke kompresi paketu DNS je možná vidět na příkladu A.7.



Obrázek A.7: Příklad komprese v paketu DNS [31].

A.4 Záznamy DNS a jejich typy

Základním nositelem informací z databáze DNS jsou záznamy DNS. Jak již bylo popsáno výše v kapitole A.3, záznamy DNS jsou součástí paketu DNS nacházející se přímo za sekcí s dotazy.

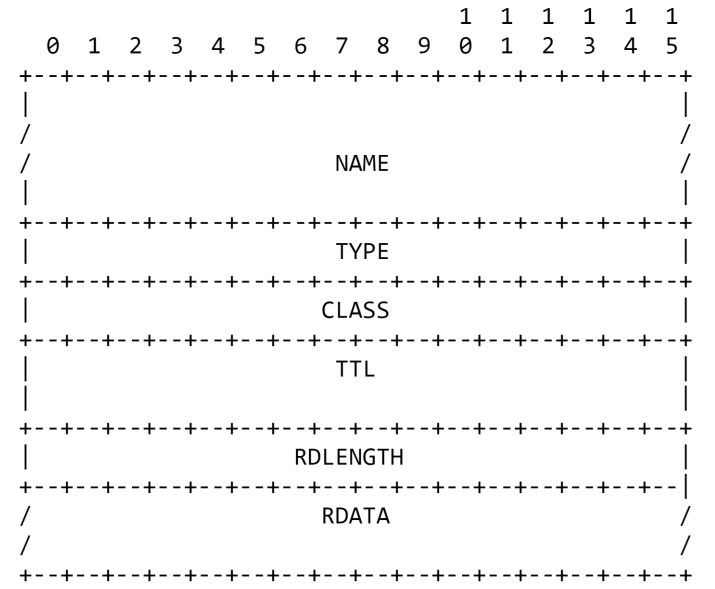
V rámci hrozeb v systému DNS mohou níže specifikované záznamy nést různé informace relevantní k danému útoku či přímo citlivá data, škodlivé skripty atd.

A.4.1 Základní struktura záznamu

Obecnou strukturu záznamu DNS v rámci paketu DNS je možné vidět na obrázku A.8.

Z tohoto obrázku je patrné, že se záznam vždy skládá z položky `NAME`, která určuje doménové jméno, pro kterou je poskytnut daný záznam. Dále pak vždy obsahuje položku `TYPE` určující typ záznamu, položku `CLASS` identifikující třídu záznamu, která má ve většině případů hodnotu jedna – Internet, a položku `TTL`, která značí dobu, po kterou může být daný záznam uložen v paměti cache. Tato hodnota může být i nulová, z čehož vyplývá, že se záznam DNS nemá uložit do této paměti.

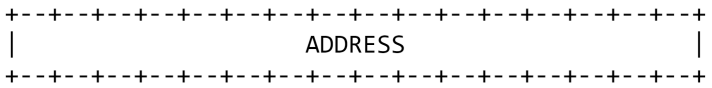
Na závěr každý záznam obsahuje další dvě pole a to `RDLENGTH`, který udává délku obsahu záznamu, a `RDATA` se samotným obsahem.



Obrázek A.8: Obecná struktura záznamu paketu DNS [32].

A.4.2 Záznam typu A a AAAA

Záznam typu A či AAAA nese informaci o namapování daného doménového jména na adresu IP. V případě typu A se jedná o adresu IPv4 a u typu AAAA o adresu IPv6.

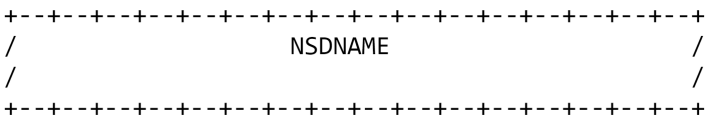


Obrázek A.9: Struktura záznamu typu A [32].

Na obrázku A.9 lze vidět strukturu tohoto záznamu, kde v poli `ADDRESS` se nachází 32bitová hodnota v případě adresy IPv4 nebo 128bitová hodnota u adresy IPv6.

A.4.3 Záznam typu NS

Záznam typu NS obsahuje informaci o doménovém jméně autoritativního serveru pro danou doménu. Využívá se především při procesu rezoluce, kdy iterativní servery DNS pouze vrátí informaci, kde je možné se dále doptat.

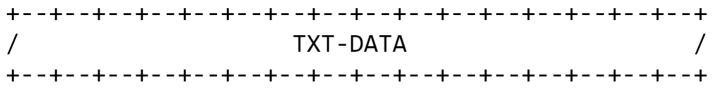


Obrázek A.10: Struktura záznamu typu NS [32].

V poli `NSDNAME`, které je možné vidět na obrázku A.10 tohoto záznamu, se nachází doménové jméno autoritativního serveru.

A.4.4 Záznam typu TXT

V záznam typu TXT jsou uložena libovolná textová data, která se týkají dodatečných informací o různých údajích spojených se samotnou doménou.

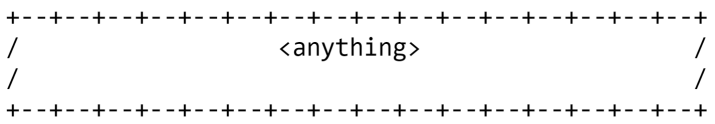


Obrázek A.11: Struktura záznamu typu TXT [32].

Dle struktury z obrázku A.11 tento záznam obsahuje pole `TXT-DATA`, ve kterém se nachází textový řetězec.

A.4.5 Záznam typu NULL

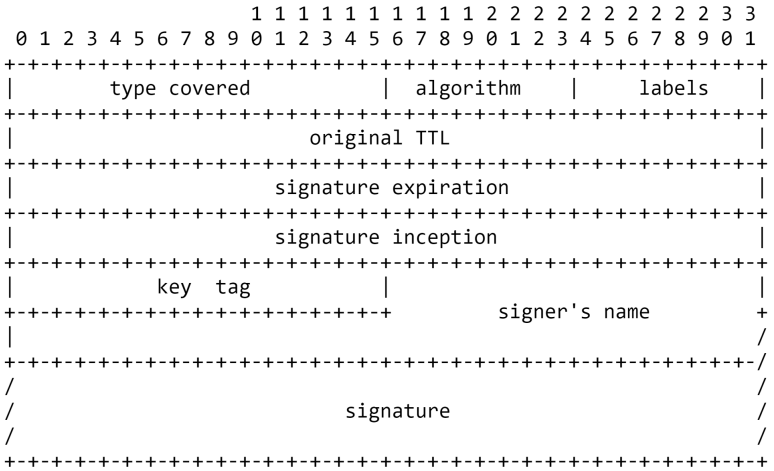
Dalším zmíněným je záznam typu NULL, který dle struktury na obrázku A.12 může obsahovat libovolná data. Jejich použití je spíše ojedinělé.



Obrázek A.12: Struktura záznamu typu NULL [32].

A.4.6 Záznam typu SIG

Posledním detailněji zmíněným typem záznamu je SIG. Pomocí něho jsou autentizována data zasílaná v systému DNS v rámci bezpečnostního rozšíření DNSSEC. Dle struktury, která je možná vidět na obrázku A.13, je patrné, že obsahuje velké množství různých polí spojenými s digitálním podpisem.



Obrázek A.13: Struktura záznamu DNS typu SIG dle [18].

V současné době je záznam SIG nahrazen novějším typem záznamu RRSIG. V rámci této práce je však zdůrazněn vzhledem kvůli jeho využití při jedné z bezpečnostních hrozeb.

A.4.7 Další typy záznamů

V rámci této práce byly zmíněny pouze takové typy záznamů, které jsou relevantní vzhledem k zmíněným bezpečnostním hrozbám blíže popsaných v kapitole 2. Systém DNS však definuje několik desítek typů záznamů, mezi které například dále patří typ SOA obsahující informace o uložení autoritativních dat v dané zóně, typ MX nesoucí informaci o poštovním serveru pro danou doménu, typ PTR pro zpětné mapování adresy IP na doménové jméno a mnoho dalších.

Příloha B

Dostupná detekční pravidla IDS

V této příloze je uveden výčet nalezených pravidel dvou systému IDS, konkrétně Suricata a Snort. V rámci této přílohy jsou tato pravidla členěna dle útoků zmiňovaných v kapitolách 2 a taktéž kapitole 4 zabývající se monitorováním komunikace DNS a návrhem detekce.

Pravidla Emerging Threats od společnosti Proofpoint

V této části jsou vypsány pravidla spadající pod systém Suricata, konkrétně se jedná o sadu Emerging Threats Open od společnosti Proofpoint[1].

Detekce útoku Alina POS

```
alert dns $HOME_NET any -> any any (msg:"ET MALWARE AlinaPOS Exfiltration v
ia DNS"; dns.query; content:".analytics-akadns.com"; nocase; endswith; pcre
:"/^[A-Z0-9_-]+\.analytics-akadns\.com$/i"; reference:url, blog.centurylink
.com/alina-point-of-sale-malware-still-lurking-in-dns/; classtype:command-a
nd-control; sid:2030440; rev:1; metadata:created_at 2020_07_02, deployment
Perimeter, former_category MALWARE, malware_family AlinaPOS, performance_im
pact Low, signature_severity Major, updated_at 2020_07_02;)

alert dns $HOME_NET any -> any any (msg:"ET MALWARE AlinaPOS Exfiltration v
ia DNS"; dns.query; content:".akamai-analytics.com"; nocase; endswith; pcre
:"/^[A-Z0-9_-]+\.akamai-analytics\.com$/i"; reference:url,blog.centurylink.
com/alina-point-of-sale-malware-still-lurking-in-dns/; classtype:command-an
d-control; sid:2030441; rev:1; metadata:created_at 2020_07_02, deployment P
erimeter, former_category MALWARE, malware_family AlinaPOS, performance_imp
act Low, signature_severity Major, updated_at 2020_07_02;)

alert dns $HOME_NET any -> any any (msg:"ET MALWARE AlinaPOS Exfiltration v
ia DNS"; dns.query; content:".akamai-information.com"; nocase; endswith; pc
re:"/^[A-Z0-9_-]+\.akamai-information\.com$/i"; reference:url,blog.centuryl
ink.com/alina-point-of-sale-malware-still-lurking-in-dns/; classtype:comman
d-and-control; sid:2030442; rev:1; metadata:created_at 2020_07_02, deployme
nt Perimeter, former_category MALWARE, malware_family AlinaPOS, performance
```

[1] viz https://rules.emergingthreats.net/open/suricata-5.0/

```
_impact Low, signature_severity Major, updated_at 2020_07_02;)

alert dns $HOME_NET any -> any any (msg:"ET MALWARE AlinaPOS Exfiltration v
ia DNS"; dns.query; content:".akamai-technologies.com"; nocase; endswith; p
cre:"/^[A-Z0-9_-]+\.akamai-technologies\.com$/i"; reference:url,blog.centur
ylink.com/alina-point-of-sale-malware-still-lurking-in-dns/; classtype:comm
and-and-control; sid:2030443; rev:1; metadata:created_at 2020_07_02, deploy
ment Perimeter, former_category MALWARE, malware_family AlinaPOS, performan
ce_impact Low, signature_severity Major, updated_at 2020_07_02;)

alert dns $HOME_NET any -> any any (msg:"ET MALWARE AlinaPOS Exfiltration v
ia DNS"; dns.query; content:".sync-akamai.com"; nocase; endswith; pcre:"/^[
A-Z0-9_-]+\.sync-akamai\.com$/i"; reference:url,blog.centurylink.com/alina-
point-of-sale-malware-still-lurking-in-dns/; classtype:command-and-control;
sid:2030444; rev:1; metadata:created_at 2020_07_02, deployment Perimeter, f
ormer_category MALWARE, malware_family AlinaPOS, performance_impact Low, si
gnature_severity Major, updated_at 2020_07_02;)
```

Detekce útoku DNSMessenger

```
alert udp any 53 -> $HOME_NET any (msg:"ET MALWARE DNSMessenger Payload (TX
T base64 gzip header)"; content:"|00 10 00 01|"; content:"H4sIA"; distance:
7; within:5; fast_pattern; reference:url,blog.talosintelligence.com/2017/10
/dnsmessenger-sec-campaign.html; classtype:trojan-activity; sid:2024840; re
v:1; metadata:affected_product Windows_XP_Vista_7_8_10_Server_32_64_Bit, at
tack_target Client_Endpoint, created_at 2017_10_13, deployment Perimeter, f
ormer_category TROJAN, malware_family DNSMessenger, performance_impact Mode
rate, signature_severity Major, updated_at 2017_10_13;)
```

Detekce hrozby SIGRed

```
alert tcp any 53 -> any any (msg:"ET EXPLOIT Possible Windows DNS Integer O
verflow Attempt M1 (CVE-2020-1350)"; flow:established,from_server; byte_tes
t:2,>=,0xfeea,0; content:"|00 00 18|"; within:76; content:"|00 00 18|"; dis
tance:12; within:64; fast_pattern; content:"|c0|"; distance:2; within:1; co
ntent:"|00 18|"; distance:1; within:2; reference:cve,2020-1350; reference:u
rl,research.checkpoint.com/2020/resolving-your-way-into-domain-admin-exploi
ting-a-17-year-old-bug-in-windows-dns-servers/; classtype:attempted-admin;
sid:2030533; rev:4; metadata:affected_product Windows_DNS_server, created_a
t 2020_07_14, former_category EXPLOIT, performance_impact Significant, sign
ature_severity Critical, updated_at 2020_07_16;)

alert tcp any any -> any 53 (msg:"ET EXPLOIT Possible Windows DNS Integer O
verflow Attempt M2 (CVE-2020-1350)"; flow:established,to_server; byte_test:
2,>=,0xfeea,0; content:"|00 00 18|"; within:76; fast_pattern; content:"|c0|
"; distance:2; within:1; content:"|00 18|"; distance:1; within:2; reference
:cve,2020-1350; reference:url,research.checkpoint.com/2020/resolving-your-w
ay-into-domain-admin-exploiting-a-17-year-old-bug-in-windows-dns-servers/;
classtype:attempted-admin; sid:2030532; rev:4; metadata:affected_product Wi
```

```
ndows_DNS_server, created_at 2020_07_14, former_category EXPLOIT, performan
ce_impact Significant, signature_severity Critical, updated_at 2020_07_16;)
```

Pravidla pro předplatitele systému IDS Snort

V druhé části jsou uvedeny pravidla systému Snort[2] pro předplatitele sady pravidel. Jedná se o verzi pro registrované uživatele bez poplatku.

Detekce útoku DNSMessenger

```
alert udp $HOME_NET any -> $EXTERNAL_NET 53 ( msg:"MALWARE-CNC Win.Trojan.D
NSMessenger outbound connection"; flow:to_server; content:"|01 00 00 01 00
00 00 00 00 00 0A|",depth 11,offset 2; content:"|05|stage",within 6,distanc
e 10,nocase; content:"|00 10 00 01|",within 45; metadata:impact_flag red; s
ervice:dns; reference:url,blog.talosintelligence.com/2017/10/dnsmessenger-s
ec-campaign.html; classtype:trojan-activity; sid:44595; rev:3; )
```

Detekce hrozby SIGRed

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 53 ( msg:"SERVER-OTHER Microsoft W
indows DNS server remote integer overflow attempt"; flow:to_server,establis
hed; content:"|FF|",depth 1; byte_test:1,=,5,4,bitmask 0x78; content:"|00 1
8|",depth 40,offset 22; metadata:policy balanced-ips drop,policy max-detect
-ips drop,policy security-ips drop; service:dns; reference:cve,2020-1350; r
eference:cve,2021-26897; reference:url,portal.msrc.microsoft.com/en-us/secu
rity-guidance/advisory/CVE-2020-1350; reference:url,portal.msrc.microsoft.c
om/en-us/security-guidance/advisory/CVE-2021-26897; classtype:attempted-use
r; sid:54518; rev:5; )

alert tcp $EXTERNAL_NET 53 -> $HOME_NET any ( msg:"SERVER-OTHER Microsoft W
indows DNS server remote integer overflow attempt"; flow:to_client,establis
hed; content:"|FF|",depth 1; byte_test:1,=,5,4,bitmask 0x78; content:"|00 1
8|",depth 40,offset 22; metadata:policy balanced-ips drop,policy max-detect
-ips drop,policy security-ips drop; service:dns; reference:cve,2020-1350; r
eference:url,portal.msrc.microsoft.com/en-us/security-guidance/advisory/CVE
-2020-1350; classtype:attempted-user; sid:54575; rev:4; )

alert tcp $EXTERNAL_NET any -> $HOME_NET 53 ( msg:"SERVER-OTHER Microsoft W
indows DNS server remote integer overflow attempt"; flow:to_server,establis
hed; content:"|FF|",depth 1; byte_test:1,=,0,4,bitmask 0x78; content:"|00 1
8|",depth 40,offset 22; metadata:policy balanced-ips drop,policy max-detect
-ips drop,policy security-ips drop; service:dns; reference:cve,2020-1350; r
eference:url,portal.msrc.microsoft.com/en-us/security-guidance/advisory/CVE
-2020-1350; classtype:attempted-user; sid:54576; rev:4; )

alert tcp $EXTERNAL_NET 53 -> $HOME_NET any ( msg:"SERVER-OTHER Microsoft W
indows DNS server remote integer overflow attempt"; flow:to_client,establis
```

[2] viz https://www.snort.org/downloads/#rule-downloads

```
hed; content:"|FF|",depth 1; byte_test:1,=,0,4,bitmask 0x78; content:"|00 1
8|",depth 40,offset 22; metadata:policy balanced-ips drop,policy max-detect
-ips drop,policy security-ips drop; service:dns; reference:cve,2020-1350; r
eference:url,portal.msrc.microsoft.com/en-us/security-guidance/advisory/CVE
-2020-1350; classtype:attempted-user; sid:54577; rev:4; )
```

Příloha C

Obsah přiloženého média

Na přiloženém paměťové médiu v podobě karty SD, je možné v kořenovém adresáři nalézt tyto následující podadresáře a soubor `readme.txt`:

`/tex` – zdrojové soubory technické zprávy včetně souboru PDF s její finální verzí,

`/detection_tool` – zdrojové soubory implementace detekčního nástroje,

`/threat_signatures` – navržené signatury k vybraným hrozbám pro detekční nástroj,

`/alina_pos_sim` – původní skript simulující hrozbu Alina POS,

`/dns_vut_dataset` – komunikace zachycená v páteřní síti VUT,

`/threat_datasets` – původní vytvořené datové sady pro vybrané hrozby,

`/readme.txt` – návod k zprovoznění detekčního nástroje a simulačního skriptu.

Příloha D

Manuál k detekčnímu nástroji

V této příloze je popsán způsob spuštění detekčního nástroje a informace k programu samotnému. Obecná forma příkazu pro spuštění nástroje vypadá následovně:

```
detection_tool.py [-h] [-f FLOWMON_SOUBORY] [-n NFDUMP_SOUBORY] [-b BIND_SO
UBORY] -d ADRESAR_SE_SIGNATURAMI [-s].
```

Význam jednotlivých parametrů je následující:

`-h, --help` – vytiskne nápovědu k programu a ukončí se,

`-f, --flowmon` – definuje cesty k souboru typu CSV sondy Flowmon,

`-f, --flowmon` – definuje cesty k souboru typu CSV zpracovaných nástrojem `nfdump`,

`-b, --bind` – definuje cesty k souboru typu LOG serveru BIND 9,

`-d, --signatures-dir` – definuje cestu k adresáři se signaturami útoků,

`-s, --strict` – aktivuje striktní režim při detekci, viz kapitola 5.2.

Návratové kódy nástroje

Pro různé případy ukončení nástroje vrací program jeden z následujících návratových kódů:

0 – program skončil úspěchem,

10 – nastal problém s otevíráním souboru,

11 – vznikl problém při zpracovávání vstupných dat,

12 – vyskytl se problém s přístupem do adresáře.

Příklad spuštění a výstupu

Pro následující spuštění nástroje s těmito argumenty:

```
python3 detection_tool.py -f 'sigred-generated_23022023_01-flowmon.csv' -d
'threat_signatures/'
```

dostáváme takovýto výstup:

```
================================================================================
Potential threat logs
================================================================================
24-Feb-2023 09:11:22.704 | SIGRed: vulnerability in the Windows DNS server
| client: 10.0.0.10#36113 <--> server: 10.0.0.234#53 (TCP)| query: 9.skodli
va-domena.cz
    Checked conditions:
        [MANDATORY] DNS question is for record type SIG
        [MANDATORY] TCP is used for transmission
        [MANDATORY] Byte stream is from server to client is bigger than 65,
                    500
    Uncheckable conditions:
        [MANDATORY] DNS answer is for record type SIG
        [ADDITIONAL] DNS Signer's name field in SIG record contains bigger
                     pointer value than 0x0C

================================================================================
Program statistics
================================================================================
+-----------------------------+------------------------------------------
| Number of input files       | 1
+-----------------------------+------------------------------------------
| Number of loaded signatures | 4
+-----------------------------+------------------------------------------
| Timespan of logs            | 24-Feb-2023 09:11:22 - 24-Feb-2023 09:11:22
|                             | (0 days, 0 hours, 0 minutes, 1 seconds)
+-----------------------------+------------------------------------------
| Strict mode                 | OFF
+-----------------------------+------------------------------------------

================================================================================
Detection statistics
================================================================================
+-----------------------------------------+---------------+
| Number of parsed DNS logs               | 3             |
+-----------------------------------------+---------------+
| Number of flagged logs as threads       | 1/3 (33.33 %) |
| - from which contained blacklisted domain | 0/1 (0.0 %)   |
+-----------------------------------------+---------------+
+--------------+--------------------------+
| Threat name  | Number of flagged logs   |
+==============+==========================+
| SIGRed       | 1/1 (100.0 %)            |
+--------------+--------------------------+
```