VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INFORMAČNÍCH SYSTÉMŮ
DEPARTMENT OF INFORMATION SYSTEMS

DETEKCE PODVODNÝCH VÝSTUPNÍCH UZLŮ SÍTĚ TOR

MALICIOUS TOR EXIT NODE DETECTION

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE **ANTON FIRC**
AUTHOR

VEDOUCÍ PRÁCE **Ing. LIBOR POLČÁK, Ph.D.**
SUPERVISOR

BRNO 2019

Vysoké učení technické v Brně
Fakulta informačních technologií

Ústav informačních systémů (UIFS) Akademický rok 2018/2019

Zadání bakalářské práce

Student: **Firc Anton**
Program: Informační technologie
Název: **Detekce podvodných výstupních uzlů sítě Tor**
Malicious Tor Exit Node Detection
Kategorie: Počítačové sítě

Zadání:

1. Seznamte se s anonymizační sítí Tor.
2. Nastudujte existující způsoby detekce podvodných výstupních uzlů.
3. Po dohodě s vedoucím práce navrhněte novou, či upravenou metodu detekce podvodných výstupních uzlů.
4. Návrh implementujte.
5. Implementaci otestujte a nalezené podvodné uzly reportujte projektu Tor.
6. Zhodnoťte výsledky práce a navrhněte její možná vylepšení.

Literatura:

- POLČÁK Libor. *Základní informace o síti Tor*. FIT-TR-2017-01, Brno, 2017. Dostupná online, URL https://www.fit.vutbr.cz/~polcak/.
- WINTER, Philipp aj. *Spoiled Onions: Exposing Malicious Tor Exit Relays*, Proceedings of the 14th Privacy Enhancing Technologies Symposium (PETS 2014), 2014.
- MCCOY, Damon aj. *Shining Light in Dark Places: Understanding the Tor Network*, Proceedings of the Eighth International Symposium on Privacy Enhancing Technologies (PETS 2008), Leuven, Belgium, 2008, str. 63-76.
- CHAKRAVARTY, Sambuddho aj. *Detecting Traffic Snooping in Tor Using Decoys*, Proceedings of the 14th International Conference on Recent Advances in Intrusion Detection, Menlo Park, CA, USA, 2011, str. 222-241.
- Tor project. *Reporting Bad Relays*, Dostupné online, URL https://trac.torproject.org/projects/tor/wiki/doc/ReportingBadRelays.

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 až 3.

Podrobné závazné pokyny pro vypracování práce viz http://www.fit.vutbr.cz/info/szz/

Vedoucí práce: **Polčák Libor, Ing., Ph.D.**
Vedoucí ústavu: Kolář Dušan, doc. Dr. Ing.
Datum zadání: 1. listopadu 2018
Datum odevzdání: 15. května 2019
Datum schválení: 3. května 2019

Abstrakt

Hlavným cieľom tejto bakalárskej práce je štúdium a implementácia systému pre detekciu podvodných výstupných uzlov siete Tor. Práca obsahuje informácie o už existujúcich riešeniach pre detekciu podvodných výstupných uzlov a techniky použité na detekciu podvodných výstupných uzlov. Ďalej sa práca zaoberá návrhom systému pre detekciu podvodných výstupných uzlov narúšajúcich šifrovanú HTTPS komunikáciu pri prístupe na akúkoľvek z celosvetovo najnavštevovanejších stránok na internete. Popisuje spôsob implementácie a testovania navrhnutého detekčného nástroja. Takisto sa zaoberá detekciou podvodných výstupných uzlov za použitia implementovaného nástroja a popisuje incident, pri ktorom bol odhalený a reportovaný podvodný výstupný uzol.

Abstract

The main goal of this bachelor's thesis is to study and implement detection for malicious Tor exit nodes. This work contains information about existing solutions for detecting malicious exit nodes and techniques used for Tor malicious exit nodes detection. This work also discusses design of system for detecting malicious Tor exit nodes tampering with encrypted HTTPS communication when accessing any of the worldwide most visited websites. It describes the process of implementation and testing of designed detection tool. It also deals with detection of malicious exit nodes using implemented tool and describes incident which led to the detection and reporting of a malicious exit node.

Kľúčové slová

anonymizačná sieť, sieť Tor, detekcia, podvodné výstupné uzly

Keywords

anonymization network, Tor network, detection, malicious exit nodes

Citácia

FIRC, Anton. *Detekce podvodných výstupních uzlů sítě Tor*. Brno, 2019. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Libor Polčák, Ph.D.

Detekce podvodných výstupních uzlů sítě Tor

Prehlásenie

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Libora Polčáka Ph.D. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Anton Firc
7. mája 2019

Poďakovanie

Ďakujem Ing. Liborovi Polčákovi Ph.D. za vedenie tejto bakalárskej práce, odbornú pomoc a rady pri jej vypracovaní. Ďalej by som sa chcel poďakovať rodičom za neustálu podporu v priebehu celého štúdia.

Obsah

Kapitola 1

Úvod

Anonymizačná sieť Tor poskytuje jej užívateľom prístup k službám internetu bez prezrádzania ich IP adresy. K 15.1.2019 má sieť Tor približne dva milióny užívateľov[1] a vyše šesť tisíc smerovačov[2] rozmiestnených po celom svete.

Anonymizácia je dosiahnutá utajením IP adresy užívateľa pred službou, ku ktorej pristupuje. Pri prístupe na internet cez sieť Tor je komunikácia užívateľa smerovaná naprieč niekoľkými uzlami siete Tor tvoriacimi okruh. Okruh bežne tvoria tri uzly: prvý - vstupný uzol, druhý - stredný uzol a tretí - výstupný uzol [5]. Takto je zabezpečené, že služba nepozná IP adresu užívateľa, ale IP adresu výstupného uzla. Komunikácia v rámci siete Tor je šifrovaná a žiadny z uzlov (okrem výstupného) nepozná cieľ komunikácie [2, 5]. IP adresu užívateľa pozná iba vstupný uzol [6]. Ďalšie informácie o sieti Tor sú popísané v kapitole 2.

Výstupný uzol „odlúpne“ poslednú vrstvu šifrovania a tým pádom získava prístup k originálnej komunikácii užívateľa [5]. Tento fakt umožňuje prevádzkovateľovi výstupného uzla sledovať všetku komunikáciu alebo kompromitovať prenos informácií za účelom získania prihlasovacích údajov k určitej službe, prípadne iných osobných údajov [10, 2]. Takéto správanie odporuje hlavnej koncepcii siete Tor, je teda vhodné takéto uzly detekovať a nahlásiť. Existujúce spôsoby detekcie podvodných výstupných uzlov sú detailnejšie popísané v kapitole 3.

Cieľom tejto práce je navrhnúť a implementovať systém pre detekciu podvodných výstupných uzlov siete Tor. Hlavným zmyslom je odhalenie podvodných výstupných uzlov a nahlásenie týchto uzlov projektu Tor. Návrh systému je popísaný v kapitole 4. Popisu implementácie sa venuje kapitola 5.

V priebehu práce bol nájdený jeden podvodný výstupný uzol blokujúci prístup k doméne s pornografickým obsahom. Pri prístupe na doménu prostredníctvom tohto uzlu bol užívateľ presmerovaný na stránku informujúcu o zablokovaní obsahu vzhľadom na federálne zákony. Implementovaný nástroj tento uzol detekoval podľa linkov obsiahnutých v HTML dokumente. Linky smerovali na inú doménu ako bola testovaná a používali protokol HTTP namiesto HTTPS, takže sa objavili v logovacom súbore ako podozrivé. Uzol bol následne reportovaný projektu Tor, výsledok žiaľ zatiaľ nie je známy. Tento incident spolu s postupom testovania implementácie a samotnej detekcie podvodných výstupných uzlov je popísaný v kapitole 6. Kapitola 7 obsahuje zhodnotenie dosiahnutých výsledkov tejto práce.

[1] Počet užívateľov získaný z https://metrics.torproject.org/userstats-relay-country.html

[2] 15.1.2019 bol počet smerovačov 6585, podľa http://torstatus.blutmagie.de/

Kapitola 2

Anonymizačná sieť Tor

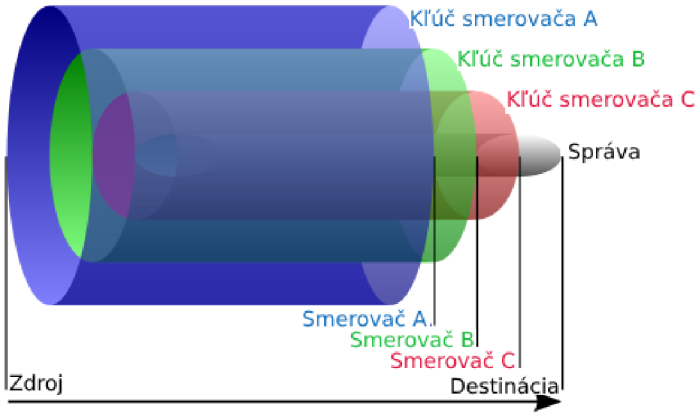
Táto kapitola popisuje architektúru siete Tor. Zaoberá sa spôsobom vytvárania okruhov, bezpečnosťou siete a problematikou podvodných výstupných uzlov.

2.1 Architektúra siete Tor

Sieť Tor sa skladá z viacerých navzájom prepojených bodov - uzlov. Sú to fyzické zariadenia používajúce voľne dostupný software pre plnenie konkrétnych úloh. Uzly môžeme rozdeliť podľa druhu úloh do skupín:

- **Smerovače**
  - **Strážca** - dlho a stabilne bežiace uzly, použiteľné ako vstupné uzly
  - **Most** - verejne neinzerované vstupné smerovače pre pripojenie do siete Tor v krajinách, kde sú bežne dostupné uzly blokované
  - **Výstupný uzol** - umožňujú prístup do internetu zo siete Tor
- **Adresárové služby** - ukladajú informácie o aktívnych uzloch siete Tor a ich aktuálnom stave
  - **Sieťový konsenzus** - dokument obsahujúci informácie o uzloch aktuálne tvoriacich sieť Tor, je pravidelne aktualizovaný adresárovými službami
- **Užívateľské proxy** - bežia na počítači klienta a získavajú informácie o stave siete, podľa ktorých vytvárajú okruhy
- **Skryté služby** - servery dostupné len zo siete Tor pomocou domény *.onion*

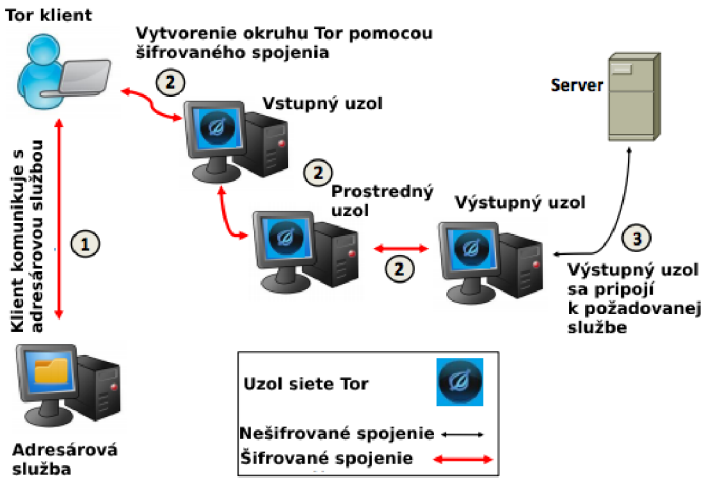
Komunikácia užívateľa je smerovaná cez okruhy, ktoré sú vytvorené perzistentným spojením medzi viacerými uzlami siete Tor. Normálne okruh tvoria práve tri uzly: prvý - vstupný uzol, druhý - prostredný uzol a tretí - výstupný uzol [5, 6]. Komunikácia v rámci siete Tor je šifrovaná [6]. Šifrovanie komunikácie je asymetrické. Klient niekoľkokrát zašifruje posielanú správu verejnými kľúčmi získanými od uzlov v aktuálne používanom okruhu. Pri prechode každým uzlom je správa dešifrovaná a preposlaná ďalšiemu uzlu po ceste k destinácii [2]. Obrázok 2.1 znázorňuje postupné odstraňovanie vrstiev šifrovania. Výstupný uzol odstráni poslednú vrstvu šifrovania a originálnu správu prepošle bežným TCP spojením požadovanej destinácii [2].



Obr. 2.1: Vrstvy šifrovania správy pri prechode okruhom siete Tor[1].

Obrázok 2.2 ukazuje základné kroky pre vytvorenie okruhu pozostávajúceho z troch uzlov siete Tor. Klient získa zoznam dostupných uzlov z adresárovej služby, vytvorí okruh pomocou viacerých uzlov Tor a následne začne vytvoreným okruhom preposielať premávku.

1. Tor klient získa z adresárovej služby zoznam dostupných uzlov (sieťový konsenzus).
2. Klient použije zoznam dostupných uzlov pre vytvorenie okruhov. Štandardne je okruh tvorený tromi uzlami.
3. Klient si zvolí jeden z okruhov a nadviaže so vstupným uzlom TCP spojenie. Premávka je potom smerovaná cez okruh k výstupnému uzlu, ktorý komunikuje priamo s danou destináciou.



Obr. 2.2: Základné kroky tvorby okruhu Tor[2].

[1]Prevzaté z http://en.wikipedia.org/wiki/Image:Onion_diagram.svg
[2]Prevzaté z https://megamindsin.blogspot.com/2016/06/tor-dark-web.html

2.2 Bezpečnosť siete Tor

2.2.1 Sledovanie komunikácie

Komunikácia klienta so vstupným uzlom siete Tor je šifrovaná. Stále je však možné určiť, že daný užívateľ komunikoval so zariadením v sieti Tor a takisto je možné zaznamenať informácie o tokoch dát, ako veľkosť jednotlivých paketov alebo ich počet do a zo siete Tor. Takto isto je možné sledovať komunikáciu medzi výstupným smerovačom a určitou destináciou [3]. Táto skutočnosť môže viesť k prezradeniu identity užívateľa voči službe na internete. Sieťové toky *klient - vstupný uzol* a *výstupný uzol - destinácia* je možné spojiť podľa času a množstva prenesených dát a tak dokázať, že užívateľ komunikoval v danom čase s určitou službou [3].

2.2.2 Podvodné výstupné uzly

Výstupné uzly siete Tor spĺňajú úlohu prostredníka v komunikácii medzi užívateľom a destináciou. Táto úloha umožňuje prevádzkovateľom podvodných výstupných uzlov zneužiť úplný prístup k premávke užívateľov. Podvodný výstupný uzol dokáže odchytiť všetku prichádzajúcu a odchádzajúcu premávku užívateľa medzi výstupným uzlom a destináciou. Odchytávanie premávky je možné jednoducho dosiahnuť použitím vlastných nástrojov implementovaných využitím knižnice `libpcap`[3] alebo použitím už existujúcich riešení ako napríklad `dsniff`[4].

Pri nešifrovanej komunikácii, ako napríklad HTTP, môže útočník priamo získať prihlasovacie údaje, ako užívateľské meno a heslo. Získané údaje môže následne zneužiť pre krádež alebo prevzatie kontroly nad účtom pre danú službu. Pre pripojenie k ukradnutému účtu môže útočník použiť ten istý výstupný uzol, pomocou ktorého prihlasovacie údaje získal, alebo akékoľvek iné zariadenie na internete.

Mimo nešifrovanej komunikácie je možné napadnúť aj riadne šifrovanú komunikáciu, ako napríklad HTTPS pripojenie k sociálnym sieťam alebo do internetového bankovníctva. Pomocou útoku typu *man-in-the-middle* môže útočník narušiť spojenie podvrhnutím vlastného SSL certifikátu [10, 2]. Útočník následne dokáže dešifrovať odpoveď od užívateľa a získať napríklad prihlasovacie údaje. Nástroj `sslstrip`[5] prevádza HTTPS odkazy na HTTP, čím obíde šifrovanie a útočník získa prístup k prihlasovacím údajom rovnako ako pri použití protokolu HTTP [10].

Všetky podvodné uzly by mali byť užívateľmi nahlásené projektu Tor[6]. Hlásenie má obsahovať nasledovné informácie:

- IP adresu alebo *fingerprint* podvodného uzla
- popis činnosti, ktorá je v rozpore s činnosťou siete Tor
- dodatočné informácie potrebné k napodobeniu incidentu

Proces nahlasovania podvodných výstupných uzlov a následný proces spracovania hlásení popisuje Wiki stránka projektu Tor zameraná na nahlasovanie podvodných výstupných uzlov [8]. Po nahlásení uzla sa tím projektu Tor najprv pokúsi o napodobenie nahláseného incidentu. Po úspešnom napodobení incidentu je kontaktovaný majiteľ daného uzla. Pokiaľ

[3] https://www.tcpdump.org/
[4] https://www.monkey.org/ dugsong/dsniff/
[5] https://moxie.org/software/sslstrip/
[6] https://trac.torproject.org/projects/tor/wiki/doc/ReportingBadRelays

problém nie je možné s majiteľom uzla vyriešiť, je uzol v rámci siete Tor označený špeciálnym označením. Podľa závažnosti incidentu existujú tri druhy označenia, podľa ktorých je následne uzol používaný v rámci siete s obmedzením alebo úplne vylúčený z používania:

- **BadExit** - uzol s označením BadExit zostáva v rámci siete používaný, no už nikdy nie ako výstupný uzol
- **Invalid** - vyradený z používania. Uzly s označením Invalid je možné používať po nastavení príznaku *AllowInvalidNodes* ako vstupné alebo stredné uzly.
- **Reject** - uzol je úplne vylúčený z používania v rámci siete Tor

Kapitola 3

Detekcia podvodných výstupných uzlov

Výstupný uzol má priamy prístup k originálnej komunikácii užívateľa, to mu dovoľuje komunikáciu sledovať a narušiť [2, 5, 10]. Sledovanie komunikácie je pasívny proces bez zjavných účinkov. To, že je komunikácia sledovaná, je však možné dokázať pri potenciálnom použití takto získaných dát [2]. Narušením komunikácie dokáže útočník dešifrovať užívateľovu komunikáciu, napríklad podvrhnutím vlastného falošného X.509 certifikátu [10]. Útoky tohto typu sú jednoduchšie detekovateľné, nakoľko je možné ich odhaliť aj pokiaľ útočník nepoužije získané informácie.

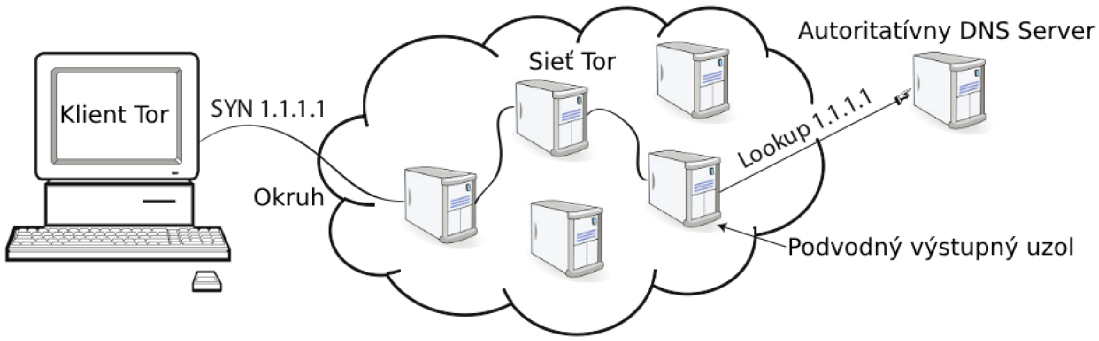
3.1 História detekcie podvodných výstupných uzlov

Prvý pokus o detekciu podvodných výstupných uzlov je z roku 2008 od McCoy aj. [5]. Tento pokus bol zameraný na zisťovanie sledovania komunikácie výstupnými uzlami siete Tor. Sledovacie nástroje, ako napríklad `tcpdump`, sú často nastavené pre spätné vyhľadanie IP adresy pomocou protokolu DNS. Tým dávajú priestor vystopovať spätnú požiadavku k výstupnému uzlu.

Pre tieto účely bol vytvorený vlastný autoritatívny DNS server, ktorý mapoval doménové mená na blok IP adries pod kontrolou autora. Následne bol pomocou klienta Tor vytvorený okruh a týmto okruhom bol odoslaný SYN ping na jednu z IP adries, pre ktoré ponúkali DNS rezolúciu. Táto komunikácia bola smerovaná na niekoľko vybraných portov bežne využívaných pre nešifrovanú komunikáciu. Tento proces je popísaný na obrázku 3.1.

Použitím popísaného postupu bol hneď v prvý deň testovania odhalený výstupný uzol, ktorý sa pokúsil o spätný DNS dotaz okamžite po prenose autorových dát. Bližšia inšpekcia ukázala, že iba komunikácia na porte 110 spúšťala spätné DNS dotazy. To znamená, že bola zaznamenávaná iba komunikácia bežná pre protokol POP3. Tým vzniká podozrenie, že bol tento port špeciálne vybraný pre získavanie prihlasovacích údajov. Pre detekciu krádeže prihlasovacích údajov bola detekcia rozšírená o takzvaný *honeypot*. Jednalo sa o systém, ktorý priradil každému výstupnému uzlu jedinečné prihlasovacie údaje a tie následne použil pre prihlásenie k nezabezpečenej službe. Pokiaľ boli prihlasovacie údaje zneužité, je možné ľahko dohľadať podvodný výstupný uzol, ktorý tieto údaje získal.

Ako sám autor uvádza, takýto systém detekcie má svoje obmedzenia. Spätné DNS požiadavky je možné vystopovať len k DNS serveru daného výstupného uzla. Výstupný uzol môže navyše využívať bezplatné DNS služby, ktoré poskytujú prevažne anonymnú DNS



Obr. 3.1: Technika detekcie podvodných výstupných uzlov[1]

rezolúciu. Ak zoberieme do úvahy, že výstupný uzol vykoná dopyt na DNS server okamžite, tak je možné takýto uzol ľahko odhaliť. Pokiaľ je dopyt na DNS server vykonaný neskôr, je potrebné rozsiahlejšie riešenie pre spojenie výstupného uzla a dotazu na DNS server.

Celá metóda sa zároveň spolieha na vyvolanie spätných DNS dotazov prechádzajúcou komunikáciou. Tie môžu byť vyvolané až po dlhšej časovej dobe, napríklad pri vyťažených uzloch. Prípadne môže byť tento spôsob detekcie obídený zakázaním spätného DNS dotazovania pri logovaní.

3.2 IMAP a SMTP

Táto sekcia popisuje prácu *Detecting Traffic Snooping in Tor Using Decoys* [2]. Všetky informácie obsiahnuté v tejto sekcii sú z tejto práce prevzaté.

Chakravarty aj. [2] sa vo svojej práci zamerali na detekciu sledovania nezabezpečenej mailovej komunikácie protokolov IMAP a SMTP. Protokoly IMAP a SMTP boli zvolené, pretože poskytujú nešifrovanú autentifikáciu cez obyčajný text a sú povolené na veľkej väčšine uzlov. Technika detekcie, ktorú autori zvolili, však nie je obmedzená len na spomínané protokoly. Jej funkcionalitu je možné rozšíriť aj pre iné nešifrované služby nad TCP ako je napríklad FTP alebo Telnet.

Subsystém posielajúci *návnady* je založený na vlastnom klientovi podporujúcom IMAP a SMTP protokoly. Každý deň vytvorí klient spojenie ku každej jednej službe prostredníctvom všetkých dostupných a použiteľných výstupných uzlov siete Tor. Po vytvorení spojenia sa klient autentifikuje voči serveru použitím jedinečných prihlasovacích údajov spárovaných s daným výstupným uzlom. Po úspešnom prihlásení vygeneruje náhodnú aktivitu, ako prehliadanie zložiek pri IMAP alebo odoslanie falošných emailových správ v prípade SMTP.

Za normálnych okolností by mal server zaznamenať jedno prihlásenie za deň. Pokiaľ dôjde k neplánovanému prístupu za použitia predom preposlaných prihlasovacích údajov, je toto spojenie označené za nelegitímne. Tieto spojenia sú označované na základe párovania spojení generovanými klientom a všetkými spojeniami prijatými serverom. Po úspešnom pripojení prepošle server všetky zaznamenané informácie o pripojení klientovi. Klient porovná prijaté údaje vrátane prihlasovacieho mena a hesla. Pokiaľ klient nenájde v záznamoch ním generovaných spojení pár, vygeneruje správu obsahujúcu:

- čas posledného vygenerovaného spojenia používajúceho dané prihlasovacie údaje

[1] Prevzaté z práce *Shining Light in Dark Places: Understanding the Tor Network* [5]

- čas nevyžiadaného pripojenia
- IP adresu z ktorej bolo nevyžiadané pripojenie realizované
- identifikátor výstupného uzla zapojeného do incidentu

Krádež prihlasovacích údajov je možné detekovať v závislosti na tom, či a kedy útočník získané údaje použije. Z toho vyplýva, že pre riadnu detekciu podvodných výstupných uzlov je nutné aby takýto systém bežal dlhšiu dobu. Autori projektu takto detekovali podvodné výstupné uzly v priebehu desiatich mesiacov počínajúc augustom 2010. Počas tejto doby zaznamenali desať incidentov.

Aj keď každý incident súvisel s iným výstupným uzlom, všetky nelegitímne pripojenia boli smerované na IMAP server. Podľa použitých prihlasovacích údajov boli vystopované výstupné uzly zapojené do každého incidentu. Prvé štyri incidenty boli detekované počas prvých troch dní behu systému. Všetky štyri incidenty odpovedali podobnému vzoru: pripojenie na server bolo iniciované z IP adresy výstupného uzla a časový interval medzi posledným prezradením prihlasovacích údajov a pokusom o pripojenie bol takmer totožný. V týchto štyroch prípadoch bol časový interval medzi odhalením prihlasovacích údajov a pokusom o prihlásenie výrazne podobný a zároveň výrazne kratší ako pri zvyšných incidentoch. Tieto skutočnosti viedli autorov k podozreniu, že všetky štyri prípady boli koordinované jednou osobou alebo skupinou, ktorá použila rovnaké nástroje alebo metodológiu v každom prípade.

Spoločné znaky vykazovali ešte posledné dva incidenty. Jednalo sa o výstupné uzly nachádzajúce sa v rámci siete jedného poskytovateľa internetového pripojenia v Indii. Veľká časť pokusov o pripojenie na server pochádzala z IP adries v rovnakej sieti.

Zvyšné incidenty už neniesli žiadne výrazné spoločné znaky. Pokusy o pripojenie na server pochádzali z rôznych zdrojov, nikdy však z výstupného uzla ktorý získal dané prihlasovacie údaje.

3.3 exitmap

Táto sekcia popisuje prácu *Spoiled Onions: Exposing Malicious Tor Exit Relays* [10]. Všetky informácie obsiahnuté v tejto sekcii sú z tejto práce prevzaté.

Táto práca prináša nástroj `exitmap` pre detekciu podvodných výstupných uzlov. Modulárna štruktúra nástroja dovoľuje rýchly a jednoduchý vývoj nových detekčných modulov. Tento nástroj beží na jednom zariadení a vyžaduje knižnicu `Stem` pre Python, ktorá implementuje riadiaci protokol pre sieť Tor.

Po spustení nástroj spustí lokálny Tor proces a získa najnovší sieťový konsenzus, podľa ktorého nájde všetky aktuálne aktívne uzly. Následne sú vybrané výstupné uzly pre analýzu, môže sa jednať o jediný uzol alebo zoznam uzlov so spoločnými vlastnosťami, ako napríklad krajina, v ktorej sa nachádzajú. Zoznam uzlov určených pre testovanie je náhodne zoradený, aby nedochádzalo k testovaniu v rovnakom poradí. Táto vlastnosť je obzvlášť vhodná pre vývoj a testovanie nových detekčných modulov, nakoľko rozkladá rovnomerne záťaž medzi testované uzly.

Po získaní a zoradení uzlov sú vytvorené okruhy, ktoré používajú tieto uzly ako výstupné uzly. O vytvorení okruhu informuje Tor zaslaním asynchrónnej udalosti. Po prijatí udalosti spustí nástroj požadovaný detekčný modul. Činnosť detekčných modulov je popísaná ďalej v tejto sekcii.

Detekčné moduly môžu byť samostatné procesy alebo Python moduly. Procesy sú spúšťané použitím `torsock`, ktorý nahradí systémové volania ako `socket()` a `connect()`, čím ich presmeruje na port, ktorý používa Tor. Pre presmerovanie sieťového API Pythonu na port, ktorý používa Tor, bol rozšírený modul SocksiPy.

Pre zníženie nárokov na sieť Tor v zmysle výpočtovej náročnosti a takisto priepustnosti siete implementovali autori systému niekoľko vylepšení, aby bola zabezpečená rýchlosť a nízka cena detekcie. V prvom rade nástroj vytvára okruhy v sieti Tor za použitia dvoch uzlov namiesto štandardných troch. Ďalej bol ako vstupný uzol použitý vlastný smerovač, ktorý poskytoval dostatočný výkon pre zvládnutie záťaže vytvorenej detekciou. Ostatné vstupné a prostredné uzly teda neboli detekciou zaťažované.

Nástroj poskytoval päť detekčných modulov, každý zameraný na špecifický typ MitM útoku. Modul pre testovanie HTTPS spojenia stiahol z pripravenej stránky X.509 certifikát a vypočítal jeho fingerprint. Vypočítaný fingerprint bol následne porovnaný s očakávaným fingerprintom, ktorý bol pevne daný v zdrojovom kóde modulu. Pokiaľ došlo pri porovnávaní k nezhode, systém vypísal varovanie. Certifikát bol zo stránky získaný poslaním hello packetu, ktorý je používaný prehliadačom TorBrowser, čo zabezpečilo menšiu odhaliteľnosť detekcie. Útočník mohol nadobudnúť podozrenie po zistení, že užívateľ získal iba X.509 certifikát bez akéhokoľvek prehliadania danej stránky. Napriek tomu v čase, kedy by útočník nadobudol podozrenie, už nástroj získal potrebný certifikát pre kontrolu. Na tomto princípe fungoval aj modul pre testovanie XMPP a IMAPS spojenia.

Ďalší modul testoval SSH spojenie. Modul porovnával verejný kľúč získaný z SSH serveru s verejným kľúčom zapísaným v zdrojovom kóde.

Modul pre detekciu použitia nástroja `sslstrip` získal web stránku obsahujúcu HTTPS odkazy a kontroloval, či získaný HTML dokument obsahuje očakávané HTTPS hlavičky alebo hlavičky *degradované* na HTTP.

Posledný modul testoval DNS servery používané výstupnými uzlami. Na vstupe detekčného modulu bola sada doménových adries s im odpovedajúcimi IP adresami. Pokiaľ pri prístupe na akúkoľvek z domén bola vrátená iná IP adresa, vypísal systém varovanie. Testované domény spadali do viacerých kategórií ako financie, sociálne siete, politické aktivity a pornografia.

Dokopy `exitmap` detekoval 38 podvodných výstupných uzlov. Väčšina detekovaných uzlov vykonávala HTTPS MitM útoky. Takisto boli odhalené uzly prevádzkujúce nástroj `sslstrip`. Jeden z detekovaných uzlov vkladal do odpovede vlastný HTML kód. Tento uzol bol odhalený použitím sslstrip detekčného modulu, ktorý kontroloval, či vrátený HTML dokument presne odpovedá očakávanému tvaru. Výstupné uzly nachádzajúce sa v Malajzii, Hong Kongu a Turecku cenzurovali DNS záznamy. Mnohé stránky, ako napríklad *torproject.org*, *facebook.com* alebo *youtube.com*, vracali neplatné IP adresy. Štyri z detekovaných uzlov obsahovali chybu v konfigurácii, nakoľko používali politiku OpenDNS, ktorá cenzurovala stránky s pornografickým obsahom. Dva z uzlov prevádzkovali antivírový softvér, ktorý narúšal IMAPS sedenia, pravdepodobne pre inšpekciu. Všetky uzly zúčastňujúce sa HTTPS, SSH, a XMPP MitM útokov vymenili originálny certifikát za vlastnoručne podpísaný. Tieto certifikáty však neboli vydané overenou autoritou uloženou v úložisku certifikátov prehliadača TorBrowser. Užívateľ je teda v prípade MitM útoku presmerovaný na stránku varujúcu pred neznámym certifikátom.

Kapitola 4

Návrh

Táto kapitola sa zaberá detailným popisom návrhu systému pre detekciu podvodných výstupných uzlov narúšajúcich šifrovanú HTTPS komunikáciu. Postupne vysvetľuje motiváciu tvorby detekčného nástroja, navrhovanú metódu detekcie a takisto aj možné nedostatky navrhnutej metódy.

4.1 Motivácia

Winter aj. [10] vo svojej práci, popísanej v kapitole 3, predstavili metódy detekcie podvodných výstupných uzlov, ktoré podvrhávajú falošné certifikáty alebo degradujú šifrované HTTPS spojenie použitím nástroja `sslstrip`.

Metódy detekcie popísané v sekcii 3.3 pracovali s bližšie nešpecifikovanými stránkami. Tieto stránky však nemuseli byť pre potenciálneho útočníka dostatočne zaujímavé. Nakoľko útočník pri každom zásahu do komunikácie riskuje odhalenie, je možné že si útočník vytipuje stránky, pri ktorých je najvyššia pravdepodobnosť získania osobných údajov užívateľa. Tým čiastočne minimalizuje riziko odhalenia a výrazne zvýši pravdepodobnosť úspešnosti útoku. S minimalizáciou rizika však prichádza aj nižšia pravdepodobnosť detekcie podvodného výstupného uzla. Systém popísaný v tejto práci je navrhnutý s ohľadom na tieto skutočnosti.

Systém bude kontrolovať bezpečnosť šifrovaného pripojenia pre N[1] celosvetovo najnavštevovanejších stránok na internete podľa portálu *Alexa*[2]. Vychádzam teda z predpokladu, že čím navštevovanejšia stránka je, tým je zaujímavejšia pre potenciálneho útočníka. Tento postup zvýši šance pre detekciu podvodných výstupných uzlov, nakoľko sa bude jednať o reálne stránky denne navštevované veľkým počtom užívateľov.

4.2 Navrhovaný princíp detekcie

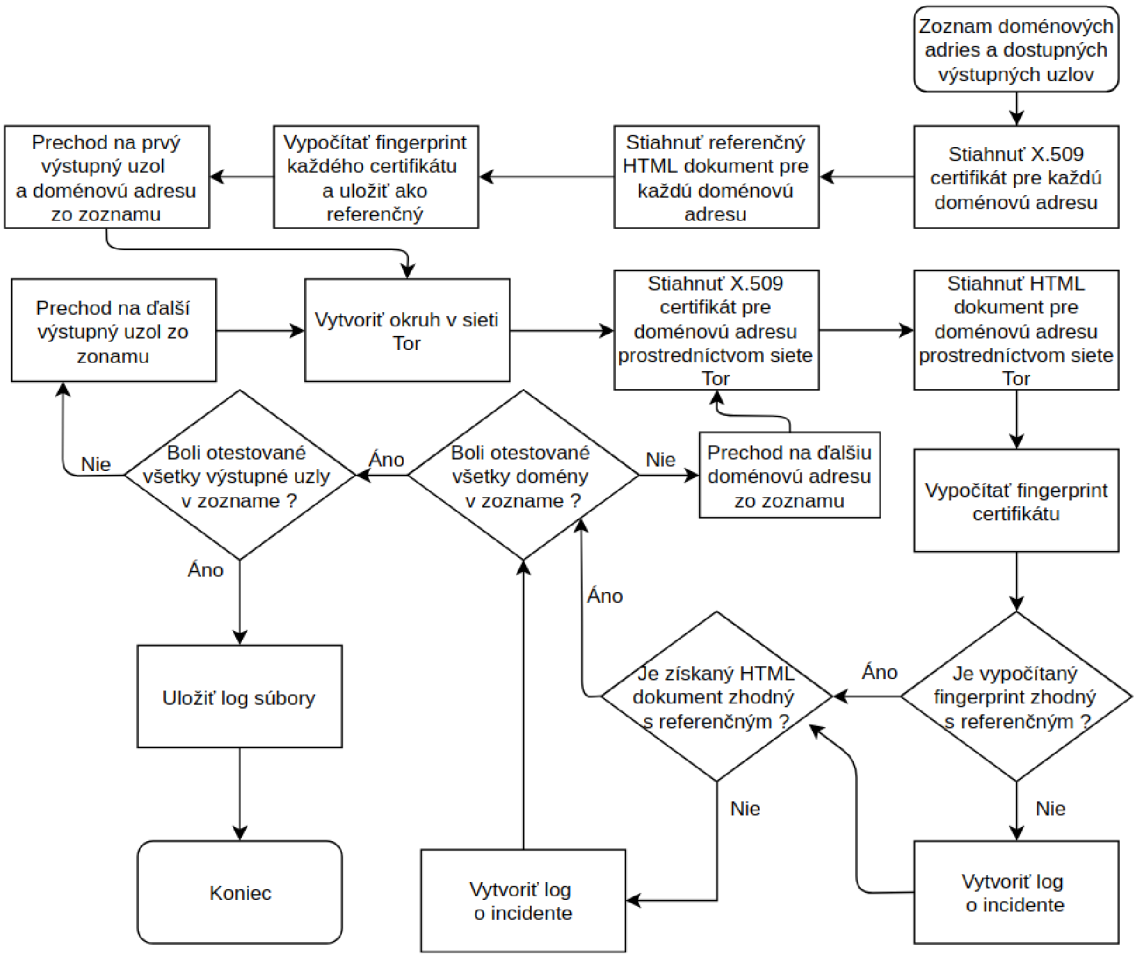
Systém má na vstupe zoznam doménových adries. Zo všetkých domén nachádzajúcich sa v zozname stiahne X.509 certifikát a uloží ho ako referenčný. Z každého referenčného certifikátu vypočíta fingerprint a uloží ho pre neskoršie porovnanie s certifikátmi získanými pri prístupe cez sieť Tor. Následne vytvorí okruh za použitia prvého výstupného uzla, pre každú doménu zo zoznamu stiahne X.509 certifikát a vypočíta jeho fingerprint. Fingerprint vypočítaný z certifikátu získaného prostredníctvom siete Tor je následne porovnaný s fingerprintom vypočítaným z referenčného certifikátu. Pokiaľ sa fingerprinty pre danú do-

[1] Presnú hodnotu bude možné určiť až počas implementácie a testovania.
[2] https://www.alexa.com/topsites

ménovú adresu zhodujú, pokračuje systém v porovnávaní. V prípade, že sa fingerprinty pre danú doménovú adresu nezhodujú, vypíše systém varovné hlásenie a vytvorí log o incidente. Podrobnejší popis vygenerovaného logu je v sekcii 4.3.

Súbežne s kontrolou certifikátov bude kontrolované, či daný výstupný uzol nedegraduje šifrovanú komunikáciu nástrojom `sslstrip`. Pre každú doménovú adresu stiahne HTML dokument a uloží ho ako referenčný. Po vytvorení okruhu a kontrole X.509 certifikátu stiahne prostredníctvom siete Tor HTML dokument pre každú doménovú adresu a porovná ho s referenčným HTML dokumentom. Pokiaľ je nájdená nezhoda medzi týmito dokumentmi, vypíše varovanie a vytvorí log o incidente. Podrobnejší popis vygenerovaného logu je v sekcii 4.3.

Týmto spôsobom je zaistená detekcia v prípade, že má originálny dokument obsahovať HTTPS linky, no získaný dokument obsahuje len HTTP linky. Takisto je týmto spôsobom možné detekovať, či do HTML kódu nebol výstupným uzlom vložený vlastný kód.



Obr. 4.1: Algoritmus testovania výstupných uzlov.

4.3 Log o incidente

HTTPS MitM

Pre každý incident s podvrhnutím vlastného falošného certifikátu log obsahuje nasledovné informácie:

- fingerprint daného výstupného uzla
- IP adresa daného výstupného uzla
- doménová adresa
- dátum a čas zachytenia incidentu

Výpis 4.2 znázorňuje príklad logu o MitM incidente obsahujúceho viac doménových adries.

```
-----------------------------------------------------
Fingerprint: 920F636A9AD647E409DF6149D48D0D6BC02C4FAC
IP address : 51.77.0.81
Records    : google.com - 2019-01-21-15:30:24
facebook.com - 2019-01-21-15:31:01
-----------------------------------------------------
```

Výpis 4.2: Príklad logu o MitM incidente.

sslstrip

Pre každý incident s podvrhnutím vlastného falošného certifikátu log obsahuje nasledovné informácie:

- fingerprint daného výstupného uzla
- IP adresa daného výstupného uzla
- doménová adresa
- dátum a čas zachytenia incidentu
- výpis rozdielov medzi HTML dokumentmi

Výpis 4.3 znázorňuje príklad logu o sslstrip incidente obsahujúceho viac doménových adries.

Po ukončení behu detekcie budú logy uložené do súborov. Názov logovacieho súboru bude odpovedať formátu {`mitm|sslstrip`}`-dátum-čas` kde `mitm` označuje logovací súbor pre incidenty typu HTTPS MitM a `sslstrip` pre incidenty typu sslstrip, dátum a čas predstavujú dátum a čas vytvorenia logovacieho súboru.

```
------------------------------------------------
Fingerprint: 920F636A9AD647E409DF6149D48D0D6BC02C4FAC
IP address : 51.77.0.81
Records    :
google.com - 2019-01-21-15:30:24
< https://myexamplepage.com
---
> http://myexamplepage.com
************************************************
facebook.com - 2019-01-21-15:35:01
> <script src="nebezpecnySkript.js"></script>
************************************************
------------------------------------------------
```

Výpis 4.3: Príklad logu o sslstrip incidente.

4.4 Možné nedostatky

Počas implementácie a testovania tohto systému bude nutné zvážiť úskalia, ktoré s vybraným postupom detekcie prichádzajú.

V prvom rade sa jedná o veľké množstvo spojení k rôznym destináciám, ktoré prichádzajú z rovnakého uzla v rámci siete Tor za krátky časový interval. Komunikácia odpovedajúca tomuto vzoru je podozrivá, nakoľko nie je bežné, aby obyčajný užívateľ v priebehu pár sekúnd navštívil desiatky stránok. Výstupný uzol teda môže implementovať mechaniky, ktoré kontrolujú prechádzajúcu komunikáciu a pokiaľ ju vyhodnotí ako podozrivú (mohla by viesť k odhaleniu), tak ju ponechá nedotknutú. Tento problém je možné odstrániť vytvorením nového okruhu k testovanému výstupnému uzlu pre každú doménu. Vytváranie nového okruhu pre každú doménu však môže výrazným spôsobom zaťažiť sieť Tor.

Ďalším možným problémom je, že systém zo stránky získa iba bezpečnostný certifikát. Je teda hneď možné zistiť, že sa nejedná o normálneho užívateľa. Pokročilejší systém podvrhávajúci falošné certifikáty teda môže pozastaviť svoju činnosť a tak sa vyhnúť odhaleniu. Tento problém je možné aspoň čiastočne vyriešiť simulovaním aktivity užívateľa.

4.5 Rozšírenie existujúceho riešenia

Tento systém plánujem implementovať ako rozšírenie už existujúceho nástroja pre detekciu podvodných výstupných uzlov `exitmap`[3] [10]. Tento nástroj implementuje určité funkcie potrebné pre činnosť navrhnutého systému.

Nástroj získa najnovší sieťový konsenzus a následne dokáže filtrovať uzly podľa označení. Ako výstupné uzly je možné zvoliť uzly nachádzajúce sa len v určitej krajine alebo výhradne uzly s označením *BadExit*. Takisto umožňuje špecifikovať vstupný uzol a jeden alebo viac špecifických výstupných uzlov pre vytvorenie okruhov.

Okruhy vytvára za použitia len dvoch uzlov - vstupného a výstupného, čím znižuje nároky kladené na sieť Tor pri detekcii.

Vďaka už implementovaným funkciám zabezpečujúcim vytváranie okruhov a smerovanie tokov do siete Tor môžem upriamiť pozornosť na implementáciu a testovanie detekčného modulu.

[3] Popísaný v sekcii 3.3

Kapitola 5

Implementácia

Táto kapitola sa zaoberá spôsobom implementácie detekčného nástroja. Popisuje všeobecné detaily implementácie, odchýlky od pôvodného návrhu, postupne činnosť všetkých modulov detekčného nástroja a spôsob ošetrenia prípadných chybových stavov.

5.1 Všeobecné implementačné detaily

Pre implementáciu nástroja je použitý jazyk Python. Mimo prostredia pre beh Python aplikácií je hlavnou prerekvizitou softvér Tor. Implementácia prebehla za použitia verzie 0.3.2.10. Všetky externé nástroje a ich verzie použité pri implementácii, testovaní a potrebné pre beh nástroja sú popísané v nasledovnom zozname:

- **Python** - verzia 2.7.15rc1 (vychádza z exitmap, nemožné použiť Python 3)
- **Tor** - verzia 0.3.2.10
- **OpenSSL** - verzia 1.1.0g
- **sed** - verzia 4.4

5.1.1 Odchýlky od návrhu

Počas implementácie a testovania jednotlivých častí sa ukázalo, že je nutné pozmeniť pôvodný návrh. Všetky zmeny a dôvody ktoré viedli k vykonaniu týchto zmien sú popísané v nasledujúcich podčastiach.

Detekcia použitia nástroja sslstrip

V pôvodnom návrhu popísanom v časti 4.2 nástroj porovnáva kompletné HTML dokumenty získané prostredníctvom siete Tor a bežnou cestou. Toto riešenie sa ukázalo ako nevhodné, nakoľko obsah a štruktúra HTML dokumentov jednotlivých domén sa líši v závislosti na geografickej polohe klienta (užívateľa). Hlavné odlišnosti v obsahu sa prejavovali v linkoch. Napríklad doména *google.com* prispôsobuje takmer všetky linky v závislosti na geografickej polohe klienta. V závislosti na krajine linky obsahujú iné domény najvyššej úrovne a takisto obsahujú rozdielne jazykové značky v URI. Príklad týchto rozdielov je ukázaný vo výpise 5.1. Tieto rozdiely vytvárali pri použití navrhnutej metódy veľké množstvo informácií, ktoré neniesli žiadnu hodnotu v rámci detekcie degradácie HTTPS spojenia na HTTP.

Slovensko: `href="https://www.google.sk/imghp?hl=sk&tab=wi"`

Francúzsko: `href="https://www.google.fr/imghp?hl=fr&tab=wi"`

Spojené štáty americké: `href="https://www.google.com/imghp?hl=en&tab=wi"`

Výpis 5.1: Rozdiely v URL adresách v závislosti na geografickej polohe klienta pre rovnaký element HTML dokumentu domény *google.com*.

Vzhľadom na tieto skutočnosti bola metóda detekcie upravená. Po novom je HTML dokument získaný, rozparsovaný a sú z neho uložené linky obsahujúce hlavičku protokolu HTTP alebo HTTPS. Linky získané prostredníctvom siete Tor a referenčné linky sú následne porovnávané a párované v troch krokoch:

- krok 1 - hľadá sa presná zhoda
- krok 2 - hľadá sa zhoda doménového mena
- krok 3 - hľadá sa zhoda hlavičky protokolu

Porovnávaním linkov v týchto troch krokoch sme schopní eliminovať rozdiely v závislosti na lokalizácii obsahu stránky. Zároveň zostáva zachovaná spoľahlivosť detekcie, nakoľko počet linkov s rovnakými hlavičkami protokolu sa musí zhodovať. Inak nástroj všetky nezrovnalosti zaznamená do logovacieho súboru pre analýzu užívateľom.

Slabinou tejto metódy je, že sa v poslednom rade opiera len o počet linkov. Pokiaľ referenčná stránka obsahuje výhradne HTTPS odkazy, je možné jedine pozmeniť URL adresy, aby smerovali na webový priestor v správe útočníka. Akákoľvek zmena protokolu z HTTPS na HTTP sa v takomto prípade okamžite zaznamená do logovacieho súboru. Pokiaľ ale referenčná stránka obsahuje aspoň jeden HTTP link, je možné takýto útok zrealizovať. Keď útočník zachová rovnaký počet HTTP linkov ako je v referenčnom dokumente, môže sa vyhnúť detekcii. Viď príklad 5.2.

Akokoľvek nepravdepodobný takýto scenár je, nastať môže. Preto je táto metóda doplnená ešte o *poistný mechanizmus*. V každom kroku je zaznamenávaný počet odstránených linkov. Po ukončení testovania všetkých výstupných uzlov sa zozbierajú počty zaznamenané každým jedným. Následne dochádza k porovnaniu týchto počtov. Pokiaľ nejaký výstupný uzol odstránil nezvyčajný počet linkov, nástroj informuje užívateľa. Pokiaľ by sme o tento mechanizmus doplnili príklad 5.2 a neuvažovali zmeny HTML dokumentu v závislosti na lokalizácii, útok by bol odhalený. Pri kontrole bežného výstupného uzla by boli všetky linky odstránené v kroku 1, pri kontrole podvodného uzla by však bol v kroku 1 odstránený iba jeden link. Zvyšné dva linky by boli odstránené až v kroku 2. Takéto správanie je pri väčšej vzorke testovaných uzlov podozrivé. Užívateľ by bol nástrojom notifikovaný a pri bližšom skúmaní by bolo možné tento uzol odhaliť.

Ani takto rozšírená metóda však nemusí vždy spoľahlivo fungovať. Vychádza z predpokladu, že prevažná väčšina testovaných uzlov nevykonáva záškodnícku činnosť. Takisto sa predpokladá, že budú naraz testované všetky použiteľné výstupné uzly, čím bude zabezpečená dostatočne veľká vzorka pre porovnávanie.

- **Príklad zlyhania detekcie sslstripingu**

  Referenčná stránka obsahuje 3 linky:

  1. HTTPS odkaz na inú stránku
  2. HTTP link načítavajúci obrázok
  3. HTTPS odkaz na stránku s prihlásením

  Útočník chce napadnúť stránku prihlasovania, aby získal prihlasovacie údaje užívateľov. Zmení teda link odkazujúci na stránku s prihlásením na HTTP. Útočník zároveň pozná špecifiká detekčného nástroja, takže zmení pôvodný HTTP link odkazujúci na obrázok na HTTPS. Stránka po úprave vyzerá nasledovne:

  1. HTTPS odkaz na inú stránku
  2. HTTPS link načítavajúci obrázok
  3. HTTP odkaz na stránku s prihlásením

  Detekčný nástroj následne pri kontrole porovná všetky linky, vzhľadom na to, že sedí počet HTTP a HTTPS linkov a ich doménové adresy, zostane takýto zásah nepovšimnutý.

Výpis 5.2: Príklad scenára, v ktorom nebude sslstriping detekovaný.

Postup overenia správneho správania sa výstupného uzla

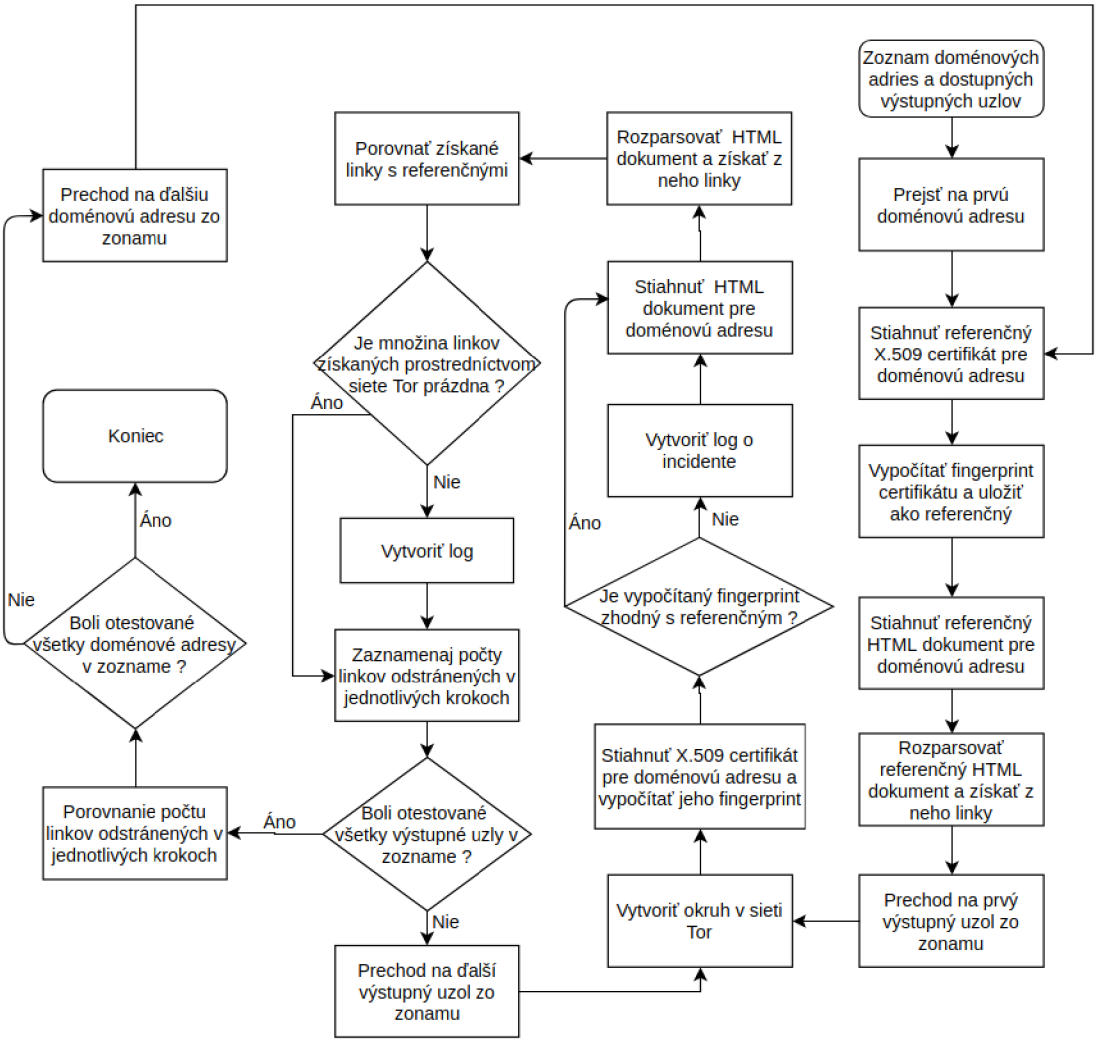
Podľa návrhu nástroj vytvorí okruh v sieti Tor a následne skontroluje všetky zadané domény. Počas testovania implementácie sa tento prístup ukázal ako nespoľahlivý. Veľká väčšina okruhov v priebehu testovania zlyhala. Okruhy nezvládali veľký nápor dát v tak malom časovom intervale.

Ďalším problémom takéhoto prístupu k detekcii je možnosť odhalenia. Tento problém bol popísaný už v časti 4.4. Vzhľadom na tieto skutočnosti bol postup upravený.

Nástroj testuje jednu doménu naprieč všetkými výstupnými uzlami. Po otestovaní jednej domény sa proces začína odznova. Pri tomto postupe dochádza k nižšiemu počtu zlyhaní okruhov. Takisto je dosiahnuté, že než výstupný uzol nadobudne podozrenie, je komunikácia ukončená. Zároveň je pre testovanie každej domény spustený nový Tor proces, takže si výstupný uzol testovaciu komunikáciu nemôže spojiť.

Tento postup síce kladie vyššie nároky na sieť Tor, ale dodáva dôveryhodnejšie výsledky. Takisto sa predchádza neočakávaným ukončeniam okruhov v priebehu testovania viacerých domén. Pri zlyhaní okruhu je test ukončený a daný uzol „preskočený“. Takto je vynechaný len jeden uzol z testovania jednej domény namiesto kompletného ukončenia testovania pre zvyšné domény. Zároveň nápor na sieť Tor je len nárazový, detekcia nebeží dlhodobo.

Diagram znázorňujúci nový postup testovania je na obrázku 5.3.



Obr. 5.3: Nový algoritmus testovania výstupných uzlov.

Logovacie súbory

Vzhľadom na zmenu procesu detekcie bolo nutné upraviť štruktúru logovacích súborov. Logovacie súbory sú teraz tvorené pre každú testovanú doménu, nie pre výstupný uzol. Štruktúra logu pre detekciu sslstrip je vo výpise 5.4 a pre detekciu podvrhnutia X.509 certifikátu vo výpise 5.5.

```
**********************************************************************
Domain    :
Exit node :
Time      :
-----------------Unmatched links from Tor-------------------------

-----------------Unmatched reference links-----------------------

**********************************************************************
```

Výpis 5.4: Štruktúra logovacieho súboru o detekcii sslstripingu. `Domain` - testovaná doménová adresa, `Exit node` - URL presne identifikujúca výstupný uzol, `Time` - čas vytvorenia záznamu a časti obsahujúce postupne nepriradené linky získané prostredníctvom siete Tor a nepriradené referenčné linky.

```
**************************************************************
Domain
Exit node
Time
---BEGIN CERTIFICATE---
..text certifikátu..
----END CERTIFICATE----
**************************************************************
```

Výpis 5.5: Štruktúra logovacieho súboru o podvrhnutom X.509 certifikáte. `Domain` - testovaná doménová adresa, `Exit node` - URL presne identifikujúca výstupný uzol, `Time` - čas vytvorenia záznamu a obsah stiahnutého certifikátu.

5.2 torcheck

Tento modul je vstupným bodom detekčného nástroja. Zabezpečuje spustenie ostatných modulov, interakciu s nástrojom `exitmap` a výsledné spracovanie získaných dát. Grafické znázornenie štruktúry nástroja je na obrázku 5.6.

Modul je možné spustiť s prepínačom `-D`(`--domain`) `domain`, ktorým užívateľ zadá testovanú doménovú adresu, alebo s prepínačom `-F`(`--file`) `filename`, ktorým užívateľ zadá súbor obsahujúci zoznam doménových adries určených na testovanie. Súbor obsahuje na každom riadku jednu doménovú adresu. Argumenty sú navzájom exkluzívne. Pokiaľ bol použitý prepínač `-D`, je priamo volaná funkcia `run_test(domain)`, ktorej je predaná doménová adresa. Pokiaľ bol použitý prepínač `-F`, je otvorený súbor a v cykle volaná funkcia `run_test(domain)` pre každú doménovú adresu.

Hlavné telo tvorí funkcia `run_test(domain)`. V prvom rade je spustený skript `sslcheck_clear.py` ktorý stiahne referenčný X.509 certifikát a HTML dokument. Tento skript je popísaný v časti 5.3. Skript je spustený ako samostatný proces za využitia knižnice `subprocess` a metódy `Popen()`. Po ukončení behu skriptu je skontrolovaná návratová hodnota a buď vypísaná chyba pri získavaní referenčných dát alebo sa pokračuje spustením nástroja `exitmap`. Nástroju `exitmap` je predaná ako argument testovaná doménová adresa prepínačom `-D`. Spustenie procesu obsluhujúceho `exitmap` znovu zabezpečuje metóda `Popen()`. Po jeho ukončení je načítaný obsah súboru `stages_count`, z ktorého sú získané

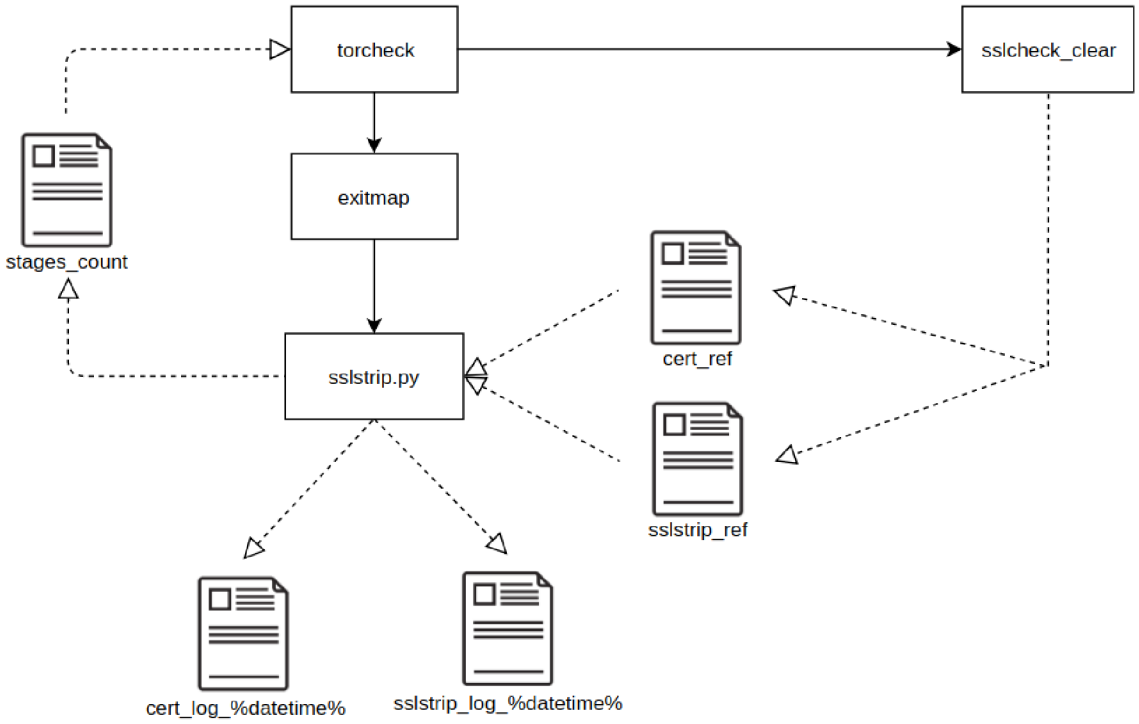
informácie o počte linkov odstránených v jednotlivých krokoch a identifikátor výstupného uzla. Formát zápisu je nasledovný:

```
X Y Z~<https://metrics.torproject.org/rs.html#details/FINGERPRINT>
```

kde X/Y/Z predstavujú počet linkov odstránených v krokoch 1/2/3. Následne je vypočítaný priemerný počet odstránených linkov v jednotlivých krokoch. Tento priemer je porovnaný s počtom odstránených linkov pre každý výstupný uzol. Pokiaľ sa počet odstránených linkov líši o viac než 10% nástroj vypíše nasledovné varovanie:

```
*WARNING* exit node <https://metrics.torproject.org/rs.html#details
/FINGERPRINT> removed suspisious amount of links in STAGE X - Y
while average is Z
```

kde `X` predstavuje číslo kroku 1/2/3, `Y` počet linkov odstránených v danom kroku, `Z` priemerný počet linkov odstránených v danom kroku. Po skončení kontroly všetkých uzlov je súbor `stages_count` odstránený a beh nástroja ukončený.



Obr. 5.6: Grafické znázornenie štruktúry nástroja.

5.3 sslcheck_clear

Tento modul získava referenčný fingerprint X.509 certifikátu a HTML dokument danej domény. Na vstup dostáva ako jediný argument doménovú adresu. Po spustení začne modul získaním referenčného X.509 certifikátu. Získanie certifikátu zabezpečuje nástroj `OpenSSL`, jeho výstup je upravený pomocou nástroja `sed` aby sme získali iba reťazec certifikátu, ktorý je oddelený značkami `-BEGIN CERTIFICATE-` a `-END CERTIFICATE-`. Reťazec je následne

uložený do súboru `cert_ref.pem`. Celý proces získania, parsovania a uloženia certifikátu je obsluhovaný z príkazového riadku v jazyku `Bash`. Spustenie príkazu zabezpečuje metóda `Popen()` z Python knižnice `subprocess`. Po úspešnom získaní a uložení certifikátu je pomocou `OpenSSL` knižnice pre Python načítaný a je vypočítaný jeho `sha1` fingerprint. Ako algoritmus bol zvolený `sha1`, nakoľko oproti `sha2` pracuje s menším počtom bitov, čím znižuje výpočtové nároky na beh detekčného nástroja. Kvalita a bezpečnosť algoritmu bola zanedbaná, nakoľko sa jedná len o porovnanie získaných certifikátov. Následne je vytvorený súbor obsahujúci referenčný fingerprint certifikátu danej domény. Po ukončení práce s certifikátom je stiahnutý HTML dokument z danej domény. Tento dokument je následne za použitia knižnice `BeautifulSoup4` rozparsovaný, sú v ňom vyhľadané linky spĺňajúce kritériá popísané v časti 5.3.1 obsahujúce HTTP alebo HTTPS hlavičky a všetky tieto linky sú priebežne zapisované do súboru obsahujúceho referenčné linky.

5.3.1 Výber linkov pre kontrolu

Projekt HTTPLeaks [4] sa zameriava na vytvorenie zoznamu možností, pomocou ktorých môže web stránka sťahovať externý obsah. Z vytvoreného zoznamu som vybral osem prípadov, v ktorých môže najpravdepodobnejšie dôjsť k pokusu o útok. Jedná sa o HTML elementy, ktoré zabezpečujú napríklad interakciu s užívateľom, ako je zadávanie vstupu alebo presmerovanie na inú doménu/časť stránky. Tieto elementy sú tým pádom najnáchylnejšie na napadnutie, nakoľko môžu prijímať prihlasovacie údaje alebo presmerovať užívateľa na nezabezpečenú (HTTP) verziu stránky.

Pre stopercentnú presnosť detekcie by bolo vhodné zahrnúť všetky možnosti úniku informácií prostredníctvom HTTP dotazov. Kontrola všetkých elementov vedie k výraznému predĺženiu trvania detekcie. Tento časový nárast však nie je úmerný dosiahnutému kvalitatívnemu zlepšeniu. Ďalším dôvodom, prečo kontrolovať menej elementov, je vznik takzvaných *false positives*[1]. Tieto výsledky sú spôsobené zmenami v HTML dokumentoch stránok napríklad v závislosti na lokalizácii obsahu podľa lokality klienta (užívateľa). Výskyt false positives v logovacích súboroch sťažuje kontrolu výsledkov. Takisto predpokladáme, že pri použití nástroja `sslstrip` budú ovplyvnené všetky linky obsiahnuté v HTML dokumente. Z týchto dôvodov boli vybrané len určité elementy a ich atribúty, pri ktorých je najväčšia pravdepodobnosť degradácie HTTPS linkov na HTTP:

- atribút `href` elementu `<a>`
- atribút `href` elementu `<link>`
- atribút `action` elementu `<form>`
- atribút `src` elementu `<img>`
- atribút `src` elementu `<image>`
- atribút `href` elementu `<image>`
- atribút `src` elementu `<input>`
- atribút `src` elementu `<script>`

[1] Z angličtiny, výsledok testu, ktorý nesprávne ukazuje kladné splnenie podmienky

5.4 Modul sslstrip

Beh modulu začína jeho načítaním a inicializáciou globálnych premenných. Po inicializácii je volaná funkcia `setup(args)`, ktorá dostane ako argument premennú obsahujúcu všetky argumenty, s ktorými bola spustená aplikácia `exitmap` a do globálnej premennej `destinations` uloží užívateľom zadané doménové meno.

Po úspešnom načítaní a nastavení modulu je volaná funkcia `fetch_page(exit_desc)`, ktorá prijme ako argument objekt obsahujúci informácie o testovanom výstupnom uzle. Z tohto objektu je následne získaný fingerprint testovaného výstupného uzla a vytvorená URL odkazujúca na informácie o danom uzle. Detekcia začína získaním X.509 certifikátu prostredníctvom siete Tor pre testovanú doménovú adresu a vypočítaním `sha1` fingerprintu. Získanie certifikátu a výpočet fingerprintu prebieha rovnakým spôsobom ako je popísaný v časti 5.3. Po výpočte fingerprintu je zo súboru `cert_ref` načítaný referenčný fingerprint. Pokiaľ sa fingerprint certifikátu získaného prostredníctvom siete Tor nezhoduje s referenčným fingerprintom, vypíše nástroj nasledovné varovanie:

```
Mismatching X.509 certificate fingerprint for (DESTINATION)
<https://metrics.torproject.org/rs.html#details/FINGERPRINT>
```

Následne je do logovacieho súboru zapísaný záznam o incidente.

Ďalším krokom je kontrola linkov získaných prostredníctvom siete Tor. Modul získa HTML dokument patriaci danej doméne prostredníctvom knižnice `urllib2`. Timeout spojenia je nastavený na 10 sekúnd. V prípade neúspechu vypíše chybovú hlášku a ukončí testovanie pre daný výstupný uzol. Následne je volaná funkcia `load_links()`, ktorá do poľa `links_ref[]` načíta zo súboru `sslstrip_ref` všetky referenčné linky. Po úspešnom načítaní linkov je pomocou knižnice `BeautifulSoup4` rozparsovaný získaný HTML dokument a sú v ňom nájdené všetky linky spĺňajúce rovnaké kritéria ako tie referenčné. Kritériá pre získavanie referenčných linkov sú popísané v časti 5.3.1. Pokiaľ link začína `http://` alebo `https://`, je volaná funkcia `check_link(link)` a ako parameter je jej predaný aktuálne kontrolovaný link vo formáte textového reťazca.

Funkcia `check_link(link)` prijíma ako parameter textový reťazec obsahujúci link určený na kontrolu. Kontrola linkov získaných prostredníctvom siete Tor prebieha v troch krokoch. Link je postupne porovnávaný s množinou referenčných linkov získaných bežnou cestou. Pokiaľ je v niektorom z krokov nájdená dvojica *link získaný cez Tor - referenčný link*, tak je referenčný link odstránený z množiny referenčných linkov a kontrola pokračuje odznova ďalším linkom získaným cez Tor. V každom kroku je zaznamenávaný počet odstránených linkov. Počet linkov odstránených v jednotlivých krokoch je následne spolu s URL identifikujúcou daný výstupný uzol zapísaný do súboru `stages_count`. Význam týchto informácií a spôsob ich spracovania je popísaný v časti 5.2. Presný popis krokov 1, 2, 3 sa nachádza v nasledujúcich častiach:

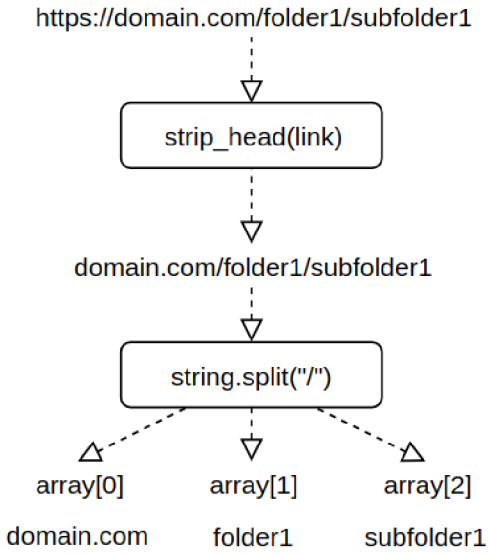
- **Krok 1**
  Prvý prechod poľom referenčných linkov. Hľadá sa presná zhoda medzi kontrolovaným linkom a jedným z referenčných linkov. Pri nájdení zhody je referenčný link odstránený z poľa referenčných linkov, inkrementované počítadlo pre krok 1 a funkcia ukončená. Pokiaľ nebola nájdená presná zhoda, pokračuje prehľadávanie krokom 2.

- **Krok 2**
  Druhý prechod poľom referenčných linkov hľadá zhodu v doménovom mene. Najprv

sa kontroluje, či vybraná dvojica linkov obsahuje rovnaký protokol HTTP/HTTPS. Pokiaľ je táto podmienka splnená, je link rozparsovaný pre extrakciu doménového mena. Rozparsovanie linku začína odstránením hlavičky protokolu, čo je zabezpečené funkciou `strip_head(link)`.

Táto funkcia prijíma ako parameter textový reťazec obsahujúci link začínajúci `https://` alebo `http://`. Funkcia následne vráti vstupný reťazec bez prvých 8 alebo 7 znakov podľa protokolu v hlavičke.

Po odstránení hlavičky protokolu oboch z dvojice linkov sú linky rozdelené na časti podľa oddeľovača "/". Využitá je funkcia `string.split(separator, max)`, ktorá po rozdelení vracia pole textových reťazcov. Prvý element získaného poľa následne obsahuje doménové meno. Postup parsovania je graficky znázornený na obrázku 5.7. Pri zhode doménových mien je referenčný link odstránený z poľa referenčných linkov, inkrementované počítadlo pre krok 2 a funkcia ukončená. Pokiaľ nebola nájdená zhoda doménových mien, pokračuje prehľadávanie krokom 3.



Obr. 5.7: Grafické znázornenie postupu parsovania linku v kroku 2.

- **Krok 3**
  Tretí prechod poľom referenčných linkov hľadá zhodu v hlavičke protokolu. Hlavičky sú kontrolované pomocou regulárnych výrazov. Ak sa hlavičky protokolov zhodujú, referenčný link je odstránený z poľa referenčných linkov, je inkrementované počítadlo pre krok 3 a funkcia ukončená. Pokiaľ nebola zhoda nájdená ani v treťom kroku je link zapísaný do poľa `links_unmatched[]` pre neskoršie spracovanie.

Po kontrole všetkých linkov získaných prostredníctvom siete Tor nasleduje vytvorenie logovacieho súboru obsahujúceho výpis referenčných linkov a linkov získaných cez Tor, ktorým nebol pri kontrole priradený pár. Pokiaľ pole `links_unmatched[]` neobsahuje žiadny prvok, do logovacieho súboru sa nič nezapisuje. Takto sa ušetrí množstvo informácií obsiahnutých v logovacom súbore. Táto praktika zároveň nemá žiadny negatívny vplyv na výsledok detekcie. Pokiaľ boli priradené všetky linky získané cez Tor k referenčným linkom, nemáme žiadnu dvojicu linkov na porovnanie, či došlo k degradácii HTTPS spojenia na HTTP. Pri

zápise do logovacieho súboru sú linky kódované do formátu `utf-8`, aby sa predišlo problémom so zápisom a zobrazovaním špeciálnych znakov obsiahnutých v URL adresách. Jedná sa napríklad o znaky & alebo #.

Beh detekčného modulu končí zápisom počtu linkov odstránených v jednotlivých krokoch a URL presne identifikujúcej daný výstupný uzol do súboru `stages_count`. Každý riadok v súbore reprezentuje informácie o jednom výstupnom uzle. Každý riadok sa delí na štyri časti ktoré postupne udávajú:

1. počet odstránených linkov v kroku 1
2. počet odstránených linkov v kroku 2
3. počet odstránených linkov v kroku 3
4. URL presne identifikujúca výstupný uzol

Jednotlivé časti sú od seba oddelené znakom medzera (`" "`). Tento znak sa nikdy nenachádza v URL adresách, takže slúži ako vhodný oddeľovač pri parsovaní obsahu tohto súboru [1]. Príklad záznamu o jednom výstupnom uzle je vo výpise 5.8.

```
2 6 15 <https://metrics.torproject.org/rs.html#details/917AA1D942C36
B6E7D39E8A496CC25C253B36F55>
```

Výpis 5.8: Príklad záznamu o jednom výstupnom uzle v súbore `stages_count`.

5.5 exitmap

Táto časť sa zaoberá popisom implementácie už existujúceho detekčného nástroja `exitmap`. Nakoľko je tento nástroj dielom iného autora a sám o sebe nie je predmetom tejto práce, zahŕňa popis jeho implementácie iba stručný popis jeho činnosti.

Po spustení nástroj spustí proces Tor potrebný pre komunikáciu v rámci siete Tor. Vytvorenie a následnú prácu s procesom zabezpečuje knižnica `stem`. Aby nástroj dokázal vytvárať vlastné okruhy a kontrolovať pripájanie dátových tokov k okruhom, je potrebné proces spustiť s nasledovnými parametrami:

- `"__DisablePredictedCircuits": "1"` - dovoľuje užívateľovi tvorbu vlastných okruhov
- `__LeaveStreamsUnattached": "1"` - zabráni procesu Tor pripojenie dátového toku k okruhu

Následne dochádza k načítaniu detekčného modulu, tento proces začína konfiguráciou modulu - volaním funkcie `setup(args)`, ak existuje. Táto funkcia bola oproti pôvodnému nástroju modifikovaná. Bližší popis sa nachádza v časti 5.5.1. Ďalej dochádza k výberu výstupných uzlov, ktorých výstupná politika dovoľuje prístup k testovanej doméne alebo doménam. Informácie o testovaných doménach sú uložené v globálnej premennej modulu, kam boli v našom prípade zapísané funkciou `setup(args)`. Po získaní zoznamu použiteľných výstupných uzlov je tento zoznam ešte náhodne premiešaný, aby nedochádzalo k testovaniu výstupných uzlov stále v rovnakom poradí. Následne dochádza k vytvoreniu okruhov. Okruhy sú tvorené len z dvoch uzlov. Vstupný uzol je buď zadaný užívateľom alebo náhodne vybraný a použitý pre všetky okruhy. Po vytvorení okruhu notifikuje Tor `exitmap` a

ten spustí novú inštanciu detekčného modulu. Detekčné moduly teda bežia ako samostatné procesy, čím sa výrazne skracuje doba potrebná pre kontrolu všetkých výstupných uzlov.

Detekčný modul po spustení volá funkciu `run_python_over_tor(function, arguments[])`. Ako argumenty prijíma názov funkcie a následne zoznam argumentov pre zadanú funkciu. Zabezpečuje smerovanie všetkej sieťovej premávky danej funkcie cez sieť Tor. Týmto spôsobom je volaná funkcia `fetch_page(exit_desc)`, ktorá obsluhuje funkcionalitu detekčného modulu.

Po ukončení posledného procesu obsluhujúceho detekčný modul sú vypísané štatistiky. Obsahujú počet spustených modulov a pomer počtu okruhov, ktoré zlyhali, k všetkým okruhom.

5.5.1 Modifikácie nástroja exitmap

Funkcia `setup(args)`

Pôvodná funkcia `setup()` bola upravená, aby prijímala ako argument argumenty, s ktorými bol nástroj spustený. Doménová adresa určená pre testovanie sa nástroju zadáva ako argument pri spustení. Keďže detekčný modul nemá prístup k argumentom, bolo potrebné nájsť spôsob ako túto informáciu predať.

Funkcia `setup()` ponúka dobré riešenie, nakoľko takto detekčný modul pozná testovanú doménu ešte pred spustením detekcie. Toto správanie je nutné, nakoľko nástroj filtruje výstupné uzly pred spustením detekcie tak, aby ich výstupná politika dovoľovala prístup k testovanej doméne. Zároveň pokiaľ bude nástroj rozšírený o nové argumenty, nie je potrebné vykonať žiadne zmeny v kóde, aby boli tieto argumenty pre detekčný modul prístupné.

Argumenty

Oproti pôvodnému nástroju bol pridaný prepínač `-D` alebo `--domain`, ktorý umožňuje zadať doménovú adresu určenú na testovanie. Takto je možné opakovane spúšťať testy pre rozličné domény bez potreby meniť zdrojový kód. Takýto prístup je takisto bezpečnejší a spoľahlivejší ako predávanie doménovej adresy napríklad prostredníctvom dočasného súboru.

5.6 Ošetrenie chybových stavov

Pri výskyte chyby sa nástroj snaží navrátiť späť do konzistentného stavu, aby nebol narušený priebeh detekcie. Všetky chybové stavy sú ošetrené tak, aby v najhoršom prípade došlo k vyradeniu jednej doménovej adresy z testovania.

5.6.1 Zlyhanie okruhu

Ošetrenie tejto chyby je riešené už v rámci nástroja `exitmap`. Pri zlyhaní okruhu je vypísaná chybová hláška a všetky procesy komunikujúce prostredníctvom tohto okruhu ukončené. Tvorba ďalších okruhov ani práca s už otvorenými nie je nijako narušená.

5.6.2 Neprístupná doménová adresa

Získavanie dokumentov má nastavený timeout na 10 sekúnd, aby sa predišlo uviaznutiu. Po uplynutí tohto času je vypísaná chybová hláška pre užívateľa a chyba ošetrená v závislosti od modulu, v ktorom sa vyskytla. V prípade modulu získavajúceho referenčné dáta je vrátený

chybový kód. Spracovanie chyby pri získavaní referenčných dát je popísané ďalej v tejto podkapitole.

V prípade chyby pri získavaní X.509 certifikátu cez Tor je preskočený proces kontroly certifikátu a pokračuje sa stiahnutím HTML dokumentu. Pokiaľ dôjde k chybe pri získavaní HTML dokumentu, je proces kontroly danej doménovej adresy pre okruh ukončený. Chyba pri získavaní dát cez jeden okruh nemá vplyv na ostané okruhy.

5.6.3 Chyba pri získavaní referenčných dokumentov

Pokiaľ nastala akákoľvek chyba pri získavaní referenčných dokumentov, vráti modul chybový kód. Bez referenčných dokumentov detekcia nemôže pokračovať, takže je pre danú doménu rovno ukončená ešte pred spustením nástroja `exitmap`. Detekcia následne pokračuje bežným spôsobom.

5.6.4 Chyba pri otváraní súborov

Pokiaľ dôjde pri otváraní akéhokoľvek súboru k chybe, preskočí nástroj prácu s takýmto súborom a pokračuje ďalej. Výnimkami sú súbory obsahujúce referenčné dáta a súbor s doménovými adresami určenými na testovanie. V prípade problému s referenčnými súbormi je ukončená detekcia pre danú doménovú adresu, nakoľko nemá zmysel pokračovať ďalej bez dát pre porovnanie výsledku. Pri probléme so súborom obsahujúcim doménové adresy je nástroj rovno ukončený s chybovou hláškou.

Kapitola 6

Testovanie a nasadenie

Táto kapitola popisuje spôsoby testovania funkčnosti detekčného nástroja. Samostatne sa venuje testovaniu detekcie použitia nástroja `sslstrip` a detekcie podvrhnutia falošného X.509 certifikátu. Popisuje aj spôsob a výsledky testovania výstupných uzlov siete Tor.

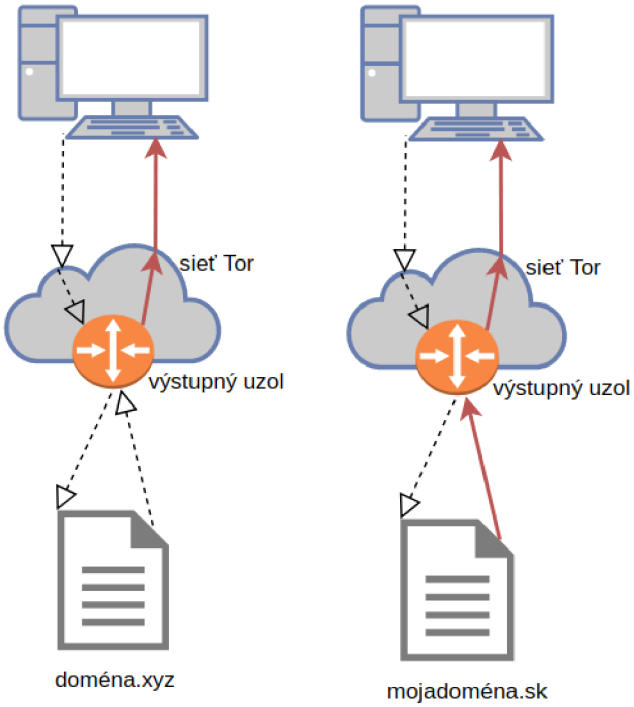
6.1 Metóda testovania

Ako prvá možnosť pre testovanie funkčnosti nástroja sa ponúka vytvorenie vlastného výstupného uzla siete Tor, následné nakonfigurovanie uzla pre použitie nástroja `sslstrip` a podvrhnutie falošného X.509 certifikátu. Táto metóda však prináša riziká a možné komplikácie spôsobené premávkou iných užívateľov siete Tor. Pri prevádzkovaní výstupného uzla z domu alebo školskej siete môžeme napríklad čeliť obvineniam z porušenia autorského zákona. Prístup k obsahu chránenému autorským zákonom prostredníctvom nami prevádzkovaného výstupného uzla nemôžeme s istotou vylúčiť ani pri krátkodobej prevádzke uzla len pre účely testovania.

Ďalším problematickým bodom je prevádzka nástroja `sslstrip` a nástroja pre podvrhnutie falošných X.509 certifikátov. Pri prevádzke týchto nástrojov môže dôjsť k odhaleniu takejto činnosti, čo môže viesť k poškodeniu nášho mena alebo mena fakulty.

Ako ďalšia možnosť sa ponúka vytvorenie *privátnej* siete Tor, ktorá bude bežať buďto v rámci virtuálnych počítačov alebo počítačov v uzavretej sieti bez prístupu do internetu. Pri použití tejto metódy však nedokážeme s istotou zaručiť, že výsledok testovania v takto vytvorenej sieti bude zhodný s výsledkom v reálnej sieti. Pre použitie detekčného nástroja v takomto prostredí by boli nutné modifikácie, nakoľko získavanie zoznamu použiteľných výstupných uzlov je závislé od pripojenia na internet. Táto metóda takisto poskytuje veľa priestoru pre nesprávnu alebo odlišnú konfiguráciu siete od tej reálnej.

Ako výsledné riešenie pre testovanie som s ohľadom na popísané skutočnosti zvolil postup, pri ktorom bude využitá reálna sieť Tor a podvodná činnosť výstupného uzla bude simulovaná v rámci kontrolovaného webového priestoru. Princíp tejto metódy je znázornený na obrázku 6.1, presný popis metódy pre testovanie detekcie použitia nástroja `sslstrip` v časti 6.2 a podvrhnutia falošného X.509 certifikátu v časti 6.3. Týmto spôsobom sa vyhneme spomenutým právnym, konfiguračným a validačným problémom.



(a) Reálny útok vykonaný výstupným uzlom.

(b) Simulovaný útok vykonaný web stránkou.

Obr. 6.1: Rozdiel medzi reálnym útokom a simulovaným útokom. Prerušované čiary znázorňujú nekompromitovanú (originálnu) komunikáciu, plné čiary kompromitovanú komunikáciu.

6.2 Testovanie detekcie použitia nástroja sslstrip

Pre testovanie detekcie použitia nástroja `sslstrip` som zvolil metódu, ktorá využíva webový priestor s PHP kódom modifikujúcim jeho obsah v závislosti na klientskej IP adrese. Podľa prvých 8 bitov klientskej IP adresy generuje HTML dokument:

- ak IP adresa začína číslom z rozsahu 0-127, obsahuje výhradne linky začínajúce *https://*
- ak IP adresa začína číslom z rozsahu 128-255, obsahuje výhradne linky začínajúce *http://*

Pri testovaní spadala IP adresa počítača, teda aj adresy prostredníctvom ktorej bol získavaný referenčný dokument, do rozsahu 0-127. Získané referenčné linky teda všetky obsahovali HTTPS hlavičky. Pri získaní linkov cez sieť Tor za použitia výstupného uzla ktorého prvých 8 bitov spadalo do rozsahu 128-255 sme dostali iba linky s HTTP hlavičkou. Toto správanie simulovalo nástroj sslstrip v závislosti na IP adrese. Z logovacieho súboru vytvoreného pri testovaní, je už na prvý pohľad jasný úspech detekcie a samotného testovania. Obsah logovacieho súboru je znázornený vo výpise 6.2. Linky získané prostredníctvom siete Tor, až na hlavičku protokolu presne zodpovedajú referenčným linkom. Takýto obsah logovacieho súboru vzorovo znázorňuje detekciu podvodného výstupného uzla degradujúceho HTTPS spojenie na HTTP.

```
**********************************************************************
Domain    : stud.fit.vutbr.cz/~xfirca00/IBT/
Exit node : <https://metrics.torproject.org/rs.html#details/917AA1D942C36
B6E7D39E8A496CC25C253B36F55>
Time      : 2019-04-18T18:09:49.772841
------------------Unmatched links from Tor-------------------------

http://www.facebook.com
http://leaking.via/conditional-comment-1
http://leaking.via/form-action

-----------------Unmatched reference links-------------------------

https://www.facebook.com
https://leaking.via/conditional-comment-1
https://leaking.via/form-action
**********************************************************************
```

Výpis 6.2: Obsah logovacieho súboru usvedčujúci výstupný uzol z degradácie HTTPS komunikácie na HTTP.

6.3 Testovanie detekcie podvrhnutia certifikátu

Pre testovanie detekcie podvrhnutia falošného X.509 certifikátu som zvolil metódu, ktorá využíva zmenu vystaveného certifikátu počas behu detekcie. Po spustení detekcie bol získaný referenčný X.509 certifikát a uložený jeho fingerprint. Následne počas behu detekcie bol pôvodný Let'sEncrypt certifikát deaktivovaný a vystavený nový. Nástroj na túto zmenu okamžite reagoval, vypísal varovanie o nezhodujúcich sa fingerprintoch a vytvoril log o udalosti. Čiastočný obsah logovacieho súboru je zobrazený vo výpise 6.3, kompletný obsah sa nachádza v prílohe B. Z logu je jasné o ktorý výstupný uzol sa jedná a zároveň obsahuje získaný podvodný certifikát pre následnú analýzu.

```
**************************************************************
Domain    kavickovo.eu
Exit node <https://metrics.torproject.org/rs.html#details/69E06EBB
2573A4F89330BDF8BC869794A3E10E4D>
Time      2019-04-19T13:08:02.199128
-----BEGIN CERTIFICATE-----
MIIGYzCCBUugAwIBAgISA+qUaahxCjLAwksuANs0R1lzMA0GCSqGSIb3DQEBCwUA
...
/TXrau/l9g==
-----END CERTIFICATE-----
**************************************************************
```

Výpis 6.3: Čiastočný obsah logovacieho súboru usvedčujúci výstupný uzol z podvrhnutia falošného certifikátu.

6.4 Testovanie výstupných uzlov

Predmetom tejto práce nie je len návrh a implementácia nástroja pre detekciu podvodných výstupných uzlov, ale aj samotná detekcia a prípadné reportovanie výsledkov projektu Tor.

Po dosiahnutí funkčného stavu som začal s testovaním výstupných uzlov. Prvé testy prebiehali výhradne za použitia uzlov s označením *BadExit*. Tieto uzly predstavujú najvyššiu pravdepodobnosť detekcie záškodníckej činnosti. Spolu bolo otestovaných približne prvých 50 doménových adries zo zoznamu *Alexa Top 1 milion*. Počas tohto testovania nebol odhalený žiadny uzol podvrhávajúci falošné X.509 certifikáty alebo degradujúci komunikáciu z HTTPS na HTTP. Pri hlbšom pátraní sa ukázalo, že takmer všetky výstupné uzly s označením *BadExit* blokovali prístup k rozličným stránkam použitím vlastného DNS servera.

Takýto výsledok ale nie je vôbec prekvapivý. Označenie *BadExit* dostavajú uzly, ktoré výrazne neporušujú pravidlá siete Tor - napríklad tie, ktoré sa nachádzajú za proxy, ktorá nedovoľuje prístup k určitým doménam. Podvrhnutie falošného X.509 certifikátu alebo degradácia HTTPS spojenia na HTTP je vážny a nebezpečný zásah do komunikácie. Odhalené uzly sú teda okamžite vyradené z prevádzky. Ponechanie takýchto uzlov v sieti predstavuje riziko aj v prípade, že nebudú použité ako výstupné.

Ďalej som pokračoval v testovaní za použitia všetkých dostupných výstupných uzlov. V priemere sa počet použiteľných výstupných uzlov pohyboval okolo čísla 900. Pri takomto množstve testovaných uzlov trvá jeden sken približne 45 minút. Veľká väčšina času je spotrebovaná na oneskorenie medzi tvorbou okruhov. Percentuálny pomer okruhov, ktoré zlyhali pri testovaní jednej domény sa pohyboval okolo 7%. Spolu bolo otestovaných približne prvých 300 doménových adries zo zoznamu *Alexa Top 1 milion*. Testovanie prebiehalo postupne v priebehu troch týždňov. V priebehu testovania boli postupne kontrolované logovacie súbory a hľadané nepriradené HTTP linky získané prostredníctvom siete Tor.

Napriek tomu, že sa v logovacích súboroch objavovali nepriradené HTTP linky získané cez Tor, nebol odhalený žiadny prípad degradácie HTTPS spojenia na HTTP. Tieto linky nikdy nemali odpovedajúci pár obsahujúci HTTPS hlavičku protokolu. To znamená, že tieto linky nevznikli zmenou HTTPS linku na HTTP. Takisto tieto linky nikdy nesmerovali na inú doménu ako bola testovaná. Pri podrobnejšej analýze sa vždy ukázalo že dané HTTP linky nepredstavujú hrozbu pre užívateľa. Väčšinou sa jednalo o linky smerujúce na články, alebo podobný obsah.

Aj keď nebol odhalený žiadny podvodný uzol vykonávajúci MitM útoky, narazil som na uzol blokujúci prístup k doméne s pornografickým obsahom. Pri skúmaní logovacieho súboru (viď príloha D) z testovania doménovej adresy *bongacams.com* som zistil, že všetky linky získané prostredníctvom siete Tor obsahujú protokol HTTP a odkazujú mimo testovanú doménu. Tieto linky odkazovali na stránky ruského ministerstva spravodlivosti informujúcich o blokovaní extrémistického obsahu alebo obsahu, ktorý porušuje autorské práva. Následne som pomocou prehliadača Tor Browser a podozrivého výstupného uzla pristúpil na danú doménu. Miesto očakávaného obsahu som bol presmerovaný na stránku ktorá informovala, že sa pokúšame o prístup na stránku, ktorej obsah je v rozpore s ruskými federálnymi zákonmi. Táto stránka podľa doménovej adresy patrí ruskej spoločnosti zaoberajúcej sa riešeniami pre kontrolu a analýzu internetovej premávky. Na základe tohto faktu predpokladám, že za blokovanie prístupu nebol zodpovedný výstupný uzol, ale poskytovateľ internetového pripojenia. Nakoľko je takéto správanie v rozpore s pravidlami siete Tor [9], nahlásil som tento výstupný uzol projektu Tor.

Ďalej som objavil zvláštne správanie pri testovaní doménovej adresy *ebay.com*. Pri testovaní tejto domény sa objavilo niekoľko varovaní o nezhodujúcom sa fingerprinte X.509

certifikátu. Po preskúmaní certifikátu uloženého v logovacom súbore som zistil, že sa nejedná o podvrhnutý certifikát ako dokazuje aj tabuľka 6.4. Táto doména z nezistených dôvodov poskytla viac než polovici výstupných uzlov certifikát rozdielny od toho referenčného. Z tabuľky je zjavné, že sa nejedná o podvrhnutý certifikát, ale certifikát vystavený pre subdoménu *pages.ebay.com*. Pri pokuse o prístup na túto subdoménu je užívateľ presmerovaný na doménu *ebay.com*. Prvé podozrenie padlo na vystavovanie odlišného certifikátu vzhľadom na lokáciu výstupného uzla. Toto podozrenie sa ale nepotvrdilo, nakoľko rovnaký certifikát bol získaný výstupnými uzlami nachádzajúcimi sa v Severnej Amerike, Európe a Ázii. Kompletný záznam z logovacieho súboru vrátane získaného certifikátu sa nachádza v prílohe C.

	Referenčný certifikát
sha-1 fingerprint	F6:AC:99:99:F8:69:EB:25:91:6C:F3:66:5A:1F:AB:34:1B:39:F0:CE
platný do	06.06.2019
Common Name (CN)	www.ebay.com
Organizational Unit (OU)	Site Operations san1-v7 s
Organization (O)	eBay\, Inc.
Issuer	CN = DigiCert SHA2 Secure Server CA,O = DigiCert Inc,C = US

	Certifikát získaný cez Tor
sha-1 fingerprint	83:46:ED:F6:07:9C:8C:4D:29:64:4A:1A:55:64:E7:79:35:70:80:60
platný do	29.05.2019
Common Name (CN)	pages.ebay.com
Organizational Unit (OU)	Site Operations
Organization (O)	eBay\, Inc.
Issuer	CN = DigiCert SHA2 Secure Server CA,O = DigiCert Inc,C = US

Tabuľka 6.4: Rozdiely medzi referenčným certifikátom a certifikátom získaným prostredníctvom siete Tor pre doménovú adresu *ebay.com*

6.5 Zhodnotenie

Pri testovaní bola úspešne overená funkčnosť nástroja. Nástroj dokáže spoľahlivo detekovať zmenu poskytnutého X.509 certifikátu. Takisto dokáže detekovať degradáciu HTTPS spojenia na HTTP v atribútoch HTML elementov. Detekcia degradácie spojenia je však v konečnom dôsledku závislá na užívateľovi, ktorý kontroluje logovacie súbory.

Počas detekcie bol odhalený výstupný uzol jednajúci v rozpore s pravidlami siete Tor. Uzol blokoval prístup k doméne s pornografickým obsahom. Tento uzol bol nahlásený projektu Tor. Výsledok tohto hlásenia žiaľ zatiaľ nie je známy.

Nástroj taktiež odhalil predávanie rôznych certifikátov doménou *ebay.com*. Oba certifikáty boli podpísané certifikačnou autoritou a vystavené pre spoločnosť eBay, takže sa nejednalo o falošný certifikát.

Pri ďalšom testovaní je potrebné zvážiť niekoľko aspektov, ktoré môžu ovplyvniť jeho výsledok. V prvom rade môže byť problematické trafiť *správne* doménové adresy pre testovanie. Počet registrovaných doménových adries každým dňom rastie. Vzhľadom na počet

presahujúci 500 miliónov je takmer nemožné otestovať všetky. Útok môže byť cielený len na jednu konkrétnu doménu, ktorá sa nenachádza v zoznamoch *Alexa Top*. V takomto prípade je vysoko nepravdepodobné, že bude pri normálnom testovaní uzol odhalený.

Ďalej je možné, že pri kontrolovaní logovacích súborov z detekcie degradácie HTTPS spojenia na HTTP dôjde k chybe. Niektoré logovacie súbory obsahujú veľké množstvo linkov z rozdielnych výstupných uzlov. Je preto možné, že pri kontrole dôjde k chybe ľudského faktora a linky vedúce k odhaleniu podvodného výstupného uzla budú jednoducho prehliadnuté.

Takisto treba počítať s možnosťou, že sa v čase detekcie v sieti žiadne podvodné výstupné uzly nenachádzajú. Výstupné uzly sú pravidelne projektom Tor monitorované a testované, či nepraktikujú nejaký druh záškodníckej činnosti [7]. V prípade odhalenia sú z bežného používania vyradené.

Kapitola 7

Záver

Cieľom tejto práce bolo zoznámenie sa s anonymizačnou sieťou Tor, štúdium existujúcich spôsobov detekcie podvodných výstupných uzlov, návrh novej alebo vylepšenej metódy pre detekciu podvodných výstupných uzlov, implementácia tejto metódy, následné testovanie a prípadné reportovanie výsledkov projektu Tor.

Potrebné poznatky o sieti Tor som získal z dokumentácie projektu Tor a technických správ riešiacich problematiky týkajúce sa siete Tor. Existujúce riešenia detekcie podvodných výstupných uzlov som naštudoval z technických správ jednotlivých riešení. Na základe získaných znalostí a návrhov pre vylepšenia z naštudovaných existujúcich riešení pre detekciu podvodných výstupných uzlov som navrhol vylepšenú metódu detekcie MitM útokov na HTTPS spojenia výstupnými uzlami siete Tor.

Počas implementácie bolo potrebné vyriešiť niekoľko problémov a pozmeniť pôvodný návrh. Najväčšie problémy spôsobovala zmena linkov v závislosti na lokalizácii obsahu stránok. Kvôli tomuto problému bola výrazne upravená metóda detekcie degradácie HTTPS spojenia.

Testovanie dokázalo, že nástroj je schopný detekovať MitM útok s podvrhnutím certifikátu alebo degradáciou HTTPS spojenia na HTTP. Počas prevádzky nástroja boli zaznamenané dva incidenty. Prvý incident zahŕňal výstupný uzol, ktorý blokoval prístup na stránku s pornografickým obsahom. Tento výstupný uzol bol následne nahlásený projektu Tor. Druhý incident detekoval predanie rozdielnych X.509 certifikátov, nejednalo sa však o falošné certifikáty.

V ďalšom pokračovaní považujem za prvoradé nástroj čo najviac automatizovať. Vylúčenie ľudského faktora z detekcie tento proces výrazne urýchli a poskytne dôveryhodnejšie výsledky. Takisto je nutné nájsť vhodnejší spôsob porovnávania URL adries s ohľadom na lokalizáciu.

Literatúra

[1] Berners-Lee, T.; W3C/MIT; Fielding, R.; aj.: *Uniform Resource Identifier (URI): Generic Syntax*. [Online; navštíveno 03.05.2019].
URL https://tools.ietf.org/html/rfc3986#section-2

[2] Chakravarty, S.; Portokalidis, G.; Polychronakis, M.; aj.: Detecting Traffic Snooping in Tor Using Decoys. In *Proceedings of the 14th International Conference on Recent Advances in Intrusion Detection*, RAID'11, Berlin, Heidelberg: Springer-Verlag, September 2011, s. 222–241.

[3] Coufal, Z.: *Korelace dat na vstupu a výstupu sítě Tor*. Diplomová práca, Vysoké učení technické v Brně, Fakulta informačních technologií, 2014.
URL http://www.fit.vutbr.cz/study/DP/DP.php?id=15819

[4] Cure53 Team: HTTPLeaks. https://github.com/cure53/HTTPLeaks, 2019.

[5] McCoy, D.; Bauer, K.; Grunwald, D.; aj.: Shining Light in Dark Places: Understanding the Tor Network. In *Proceedings of the Eighth International Symposium on Privacy Enhancing Technologies (PETS 2008)*, editácia N. Borisov; I. Goldberg, Springer, July 2008, s. 63–76.

[6] Polčák, L.: *Základní informace o síti Tor*. FIT VUT v Brně, 2017, [Online; navštíveno 15.01.2019].
URL https://www.fit.vutbr.cz/~polcak/

[7] Tor project: *How to report bad relays*. [Online; navštíveno 01.05.2019].
URL https://blog.torproject.org/how-report-bad-relays

[8] Tor project: *Reporting Bad Relays*. [Online; navštíveno 14.01.2019].
URL
https://trac.torproject.org/projects/tor/wiki/doc/ReportingBadRelays

[9] Tor project: *Tor FAQ - traffic filtering*. [Online; navštíveno 01.05.2019].
URL https://2019.www.torproject.org/docs/faq.html.en#OutgoingFirewall

[10] Winter, P.; Köwer, R.; Mulazzani, M.; aj.: Spoiled Onions: Exposing Malicious Tor Exit Relays. In *Proceedings of the 14th Privacy Enhancing Technologies Symposium (PETS 2014)*, Júl 2014.

Príloha A

Obsah priloženého pamäťového média

Adresárová štruktúra:

- **Zdrojové súbory**
  Adresár obsahuje zdrojové súbory nástroja
- **Virtuálny disk**
  Adresár obsahuje virtuálny disk s predinštalovaným nástrojom a súborom README s inštrukciami pre spustenie
- **Text bakalárskej práce**
  Adresár obsahuje zdrojové súbory textovej časti

Príloha B

Obsah log súboru z testovania

Kompletný záznam z logovacieho súboru z testovania detekčného nástroja zmenou X.509 certifikátu počas detekcie z časti 6.3.

```
**************************************************************
Domain    kavickovo.eu
Exit node <https://metrics.torproject.org/rs.html#details/69E06EBB
2573A4F89330BDF8BC869794A3E10E4D>
Time      2019-04-19T13:08:02.199128
-----BEGIN CERTIFICATE-----
MIIGYzCCBUugAwIBAgISA+qUaahxCjLAwksuANs0R1lzMA0GCSqGSIb3DQEBCwUA
MEoxCzAJBgNVBAYTAlVTMRYwFAYDVQQKEw1MZXQncyBFbmNyeXB0MSMwIQYDVQQD
ExpMZXQncyBFbmNyeXB0IEF1dGhvcml0eSBYMzAeFw0xOTA0MTkwOTQyNTVaFw0x
OTA3MTgwOTQyNTVaMBcxFTATBgNVBAMTDGthdmlja292by5ldTCCAiIwDQYJKoZI
hvcNAQEBBQADggIPADCCAgoCggIBALuA+soRAo1gslDuZoGbkYKtbJDHFuC0okSy
HRAFvIipG3MfJxAy40b7Q1AKjQHdqkOHI6oxPLU0l1tLL7NxfNqtPjSOhhIcQg+Q
JYs42eMzJ7CxdBNlrClft4YanSOQ4n74XgDEtqztasWTycRh2Mdo9NmA8inbl0Gh
6P7QfrDS8VoHMMTZx5dCLqFzgX33EJSg56NqNAXYSr9i0G0NWl/Gp1y5+zUzyQjl
dzVE2bYNUm+sHM7tVoBRaNlgyYzLFR977nUQkdMy+wPCavAdskUjfTOngushTbQy
kVXEhSDfXsTpJ2V6j6LAXxzX4G/Q3O43NMWHo9t26cnX9rxT3drG4okKBingvLVm
fV5qd3wMJSsd1RbSCMBUDr3S7/YGjKQOctvRh/Ai4fNoImKPaGOPE9AziZ61MEEC
Yv4oy9typS9e/9iAla+8sY0iykMBXLsNxhn4hfPJXBJ2yAt3UQYnKcv+F5Q85z/s
NcOTPXWnwO+y11rYeXePLNjtkNEbxV+YIx49HpI51Wq85kWfG+z7sh017vw1yor0
m7OsUyMtHmC871e9rETLgLCbjXnXHu1RhnPhlptOoD/jzEq0G12Kw2OD0pdxf7G8
FPJrTO9Q3vOYVJ4Q6L2EpF0vk6mz7KxMWxEIeUFA8FyrEGKmhtROQEbxV39+mRQM
nIdc8BofAgMBAAGjggJ0MIICcDAOBgNVHQ8BAf8EBAMCBaAwHQYDVR0lBBYwFAYI
KwYBBQUHAwEGCCsGAQUFBwMCMAwGA1UdEwEB/wQCMAAwHQYDVR0OBBYEFLSXKR/y
nzaRvr+ZaMbmGvTattUUMB8GA1UdIwQYMBaAFKhKamMEfd265tE5t6ZFZe/zqOyh
MG8GCCsGAQUFBwEBBGMwYTAuBggrBgEFBQcwAYYiaHR0cDovL29jc3AuaW50LXgz
LmxldHNlbmNyeXB0Lm9yZzAvBggrBgEFBQcwAoYjaHR0cDovL2NlcnQuaW50LXgz
LmxldHNlbmNyeXB0Lm9yZy8wKQYDVR0RBCIwIIIMa2F2aWNrb3ZvLmV1ghB3d3cu
a2F2aWNrb3ZvLmV1MEwGA1UdIARFMEMwCAYGZ4EMAQIBMDcGCysGAQQBgt8TAQEB
MCgwJgYIKwYBBQUHAgEWGmh0dHA6Ly9jcHMubGV0c2VuY3J5cHQub3JnMIIBBQYK
KwYBBAHWeQIEAgSB9gSB8wDxAHcA4mlLribo6UAJ6IYbtjuD1D7n/nSI+6SPKJMB
nd3x2/4AAAFqNTEo8wAABAMASDBGAiEA03oKtmxD/AXhjRXEpeTFDa1BEb3Iw+IG
Hj6T+HDzJxACIQDWgWZmkxjLkgC9h25V3R/tNH5i3/wCPJz/mzEy3YQyrAB2AGPy
```

```
283oO8wszwtyhCdXazOkjWF3j711pjixx2hUS9iNAAABajUxK1IAAAQDAEcwRQIg
I4AqGUUgG5F5kY7cX8wGBsusONfcx+B1lVmTqgWlTMYCIQDu8NlLbkW3evTJXEZq
2e7HB1C98pz3u7il8lPvu8//LjANBgkqhkiG9w0BAQsFAAOCAQEAmAQGaKvGoflU
Sk65godk83k8S9l6j41l3ohydfnHlMaPPNDMnasaKN2mt+ZUSFYrjuD7M2Oj+acl
ASSCr6yE6hqhsAB80iVAvWcbZu8x9LrScAi2r/Ds/B0MN0f4bzRJtBcCJVseau4k
MhMYVkKoBF+Cg2EvYYw54lw6SmZT7hFMcMNG+X2N+rJ7aoyHkYEX7CV2SO9Ro44q
22KedjS7IzWRE60zLG24HW6T5jQBNQ1kGuhkh6BiHIZl5+tlHu8PIdvelNrf+yyI
8EQUhHMYp18hWLmh0cii8lt912Ol/K7OoWEwe+q7vyaUL7KJG0jiqb7mgQ1zh2Xu
/TXrau/l9g==
-----END CERTIFICATE-----
*****************************************************************
```

Príloha C

Log súbor o podozrivom certifikáte pre doménu ebay.com

Neskrátený obsah logovacieho súboru z incidentu s rozdielnymi certifikátmi popísanom v časti 6.4.

```
**************************************************************
Domain     ebay.com
Exit node <https://metrics.torproject.org/rs.html#details/683A66
          8EBD5E275889B510CAEA45752016E3DE30>
Time       2019-04-25T17:59:29.536591
-----BEGIN CERTIFICATE-----
MIIJIzCCCAugAwIBAgIQAQRN/4Rlz+fqCmvEWvPejTANBgkqhkiG9w0BAQsFADBN
MQswCQYDVQQGEwJVUzEVMBMGA1UEChMMRGlnaUNlcnQgSW5jMScwJQYDVQQDEx5E
aWdpQ2VydCBTSEEyIFNlY3VyZSBTZXJ2ZXIgQ0EwHhcNMTkwMTAyMDAwMDAwWhcN
MTkwNTI5MTIwMDAwWjB9MQswCQYDVQQGEwJVUzETMBEGA1UECBMKQ2FsaWZvcm5p
YTERMA8GA1UEBxMIU2FuIEpvc2UxEzARBgNVBAoTCmVCYXksIEluYy4xGDAWBgNV
BAsTD1NpdGUgT3BlcmF0aW9uczEXMBUGA1UEAxMOcGFnZXMuZWJheS5jb20wggEi
MA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQDcvJGeJhhq3jErJ8Y03LuMHcff
etsXULcCCQ00GIIhGRPQQR7RKBI5EF5R7abRgLn2HixBqC/b37IF4E/Ll0w1U9cT
F/m1ruJzV+8/GvIiAlgho01XOBnn+4ULMnZnF+eeSeCUjW7Ro/x9k4dVpoRWHK4S
ROxZvwG1FdgariXYaEexiKC9zIsqwrCCsH0jkIZjHI3BHSu3M2AAG80JnDQ60Vyh
QSKIPG8vqP01C8X2eM1zkGAiBQxj92EMFWVd5ErQEc5lLcHyLmySeBMA1Y+FODVZ
VQ5r+AvVvz1mF42Hxx/Kfydcfvq1v5qQ/KR+SWTdkkJW60EDtgWWYA6atvtnAgMB
AAGjggXNMIIFyTAfBgNVHSMEGDAWgBQPgGEcgjFh1S8o541GOLQs4cbZ4jAdBgNV
HQ4EFgQUgb3wLpFEQxVGfw+bGTMvJVZFZCUwggMIBgNVHREEggL/MIIC+4IOcGFn
ZXMuZWJheS5jb22CDGJlZnIuZWJheS5iZYIMYmVubC5lYmF5LmJlggxjYWZyLmVi
YXkuY2GCB2ViYXkuYXSCB2ViYXkuYmWCB2ViYXkuY2GCB2ViYXkuY2iCCmViYXku
Y28udWuCCGViYXkuY29tggtlYmF5LmNvbS5hdYILZWJheS5jb20uaGuCC2ViYXku
Y29tLm15ggtlYmF5LmNvbS5zZ4IHZWJheS5kZYIHZWJheS5lc4IHZWJheS5mcoIH
ZWJheS5pZYIHZWJheS5pboIHZWJheS5pdIIHZWJheS5ubIIHZWJheS5waIIHZWJh
eS5wbIIHZWJheS5ydYIScGFnZXMuYmVmci5lYmF5LmJlghJwYWdlcy5iZW5sLmVi
YXkuYmWCEnBhZ2VzLmNhZnIuZWJheS5jYYINcGFnZXMuZWJheS5hdIINcGFnZXMu
ZWJheS5iZYINcGFnZXMuZWJheS5jYYINcGFnZXMuZWJheS5jaIIQcGFnZXMuZWJh
eS5jby51a4IRcGFnZXMuZWJheS5jb20uYXWCEXBhZ2VzLmViYXkuY29tLmhrghFw
YWdlcy5lYmF5LmNvbS5teYIRcGFnZXMuZWJheS5jb20uc2eCDXBhZ2VzLmViYXku
```

```
ZGWCDXBhZ2VzLmViYXkuZXOCDXBhZ2VzLmViYXkuZnKCDXBhZ2VzLmViYXkuaWWC
DXBhZ2VzLmViYXkuaW6CDXBhZ2VzLmViYXkuaXSCDXBhZ2VzLmViYXkubmyCDXBh
Z2VzLmViYXkucGiCDXBhZ2VzLmViYXkucGyCDXBhZ2VzLmViYXkucnWCEnBvcmNl
bGFpbi5lYmF5LmNvbYISc29sdXRpb25zLmViYXkuY29tghJ0ZXN0LXBhZ2VzLmVi
YXkuY2GCC3d3dy5lYmF5LmJlghZ3d3cuc29sdXRpb25zLmViYXkuY29tghVwYWdl
cy5tb3RvcnMuZWJheS5jb22CE3d3dy5tb3RvcnMuZWJheS5jb20wDgYDVR0PAQH/
BAQDAgWgMB0GA1UdJQQWMBQGCCsGAQUFBwMBBggrBgEFBQcDAjBrBgNVHR8EZDBi
MC+gLaArhilodHRwOi8vY3JsMy5kaWdpY2VydC5jb20vc3NjYS1zaGEyLWc2LmNy
bDAvoC2gK4YpaHR0cDovL2NybDQuZGlnaWNlcnQuY29tL3NzY2Etc2hhMi1nNi5j
cmwwTAYDVR0gBEUwQzA3BglghkgBhv1sAQEwKjAoBggrBgEFBQcCARYcaHR0cHM6
Ly93d3cuZGlnaWNlcnQuY29tL0NQUzAIBgZngQwBAgIwfAYIKwYBBQUHAQEEcDBu
MCQGCCsGAQUFBzABhhhodHRwOi8vb2NzcC5kaWdpY2VydC5jb20wRgYIKwYBBQUH
MAKGOmh0dHA6Ly9jYWNlcnRzLmRpZ2ljZXJ0LmNvbS9EaWdpQ2VydFNIQTJTZWN1
cmVTZXJ2ZXJDQS5jcnQwCQYDVR0TBAIwADCCAQYGCisGAQQB1nkCBAIEgfcEgfQA
8gB3AGPy283oO8wszwtyhCdXazOkjWF3j711pjixx2hUS9iNAAABaBCV9zgAAAQD
AEgwRgIhAL6s8LkyXuRMKKfmP7DjBvgqekvSAZ6h81YZaSS4MSaaAiEAmX4ANeBd
RCAOpBtkGpSOPBIJCcTK4qs6DgHuxKgd0FsAdwCHdb/nWXz4jEOZX73zbv9WjUdW
Nv9KtWDBtOr/XqCDDwAAAWgQlff7AAAEAwBIMEYCIQClrhgN7PVUcX6N8DdWKdqq
cVqcAQk2/dxmPPlN/XvijwIhANFOOObFl3Z4LU8EXA7xZJP9FORbeM6y0T91qzG7
cVFCMA0GCSqGSIb3DQEBCwUAA4IBAQDQtiWZ1jng9zheGPJaz8+9P3jn8yf0jjiV
EYBjkQamILIiHa4HQEN+0e/W/x1Go6q1lKO2BgGlBZnDOL0OxTsO4gIDD3WPG9Jf
XgB7IJCyr0OVFi+wZS4MJvFBTr3muJnacAcCJAz2Yho5RoVkdlR8WUVwF985YRIQ
BvVj7j5i0ooF9HAwiVrcHcco5A4wAUZgDC1pCpasnVUZsyAU+tIoXxTo4gmgZepe
v2fxoYDuXs2aiuvxbViDjtODg3IAdzmU3DXcrcU0km9uDUk/5e+BO/kUuWv0aGKR
pkyU9GLsrJtIj2l0SWhga4kh6ydiRbImF7qjlY4TsSNO91leAUPb
-----END CERTIFICATE-----
**************************************************************
```

Príloha D

Log súbor vedúci k odhaleniu blokovania prístupu

Obsah logovacieho súboru, ktorý vrhol podozrenie na výstupny uzol z blokovania obsahu. Popis incidentu sa nachádza v časti 6.4.

```
**********************************************************************
Domain     : bongacams.com
Exit node  : <https://metrics.torproject.org/rs.html#details/AF8E8939D67
             1BC3ED6FD5C5C6C58D9C4F29AE0FD>
Time       : 2019-04-25T19:16:47.012102
------------------Unmatched links from Tor-------------------------

http://nap.rkn.gov.ru/reestr/
http://minjust.ru/ru/extremist-materials
http://www.minjust.ru/nko/fedspisok
http://mc.yandex.ru/watch/21144079

-----------------Unmatched reference links-------------------------

https://sk.bongacams.com/members/join
https://sk.bongacams.com/members/join
https://sk.bongacams.com/members/join
https://blog.bongacams.com/bongacams-mobile
https://bongacash.com
https://twitter.com/bongacams
https://www.instagram.com/bongacams.media
https://blog.bongacams.com
https://t.me/bongacamsofficial
https://sk.bongacams.com/?page=2
https://sk.bongacams.com/
https://www.google.com/recaptcha/api.js?onload=recaptchaInit&render=exp
licit&hl=sk&t=1556212576
**********************************************************************
```