JIHOČESKÁ univerzita v Českých Budějovicích

Přírodovědecká fakulta

ANALÝZA DARKNETU SE ZAMĚŘENÍM NA FORENZNÍ ZKOUMÁNÍ

Bakalářská práce

Tomáš Rýc

Vedoucí práce: Ing. Petr Břehovský

České Budějovice 2020

Jihočeská univerzita v Českých Budějovicích
Přírodovědecká fakulta

ZADÁVACÍ PROTOKOL BAKALÁŘSKÉ PRÁCE

Student: Tomáš Rýc

Obor – zaměření studia: Kriminalisticko-technická činnost

Katedra: Ústav aplikované informatiky

Školitel: Ing. Petr Břehovský

Garant z PřF: .....
(jméno, příjmení, tituly, katedra – jen v případě externího školitele)
Školitel – specialista, konzultant: .....
(jméno, příjmení, tituly, u externího š. název a adresa pracoviště, telefon, fax, e-mail)

Téma bakalářské práce:
Analýza DARKNETu se zaměřením na forenzní zkoumání

Úkoly práce :

1. Provést analýzu DARKNETu zaměřenou na možné páchání protiprávního jednání
   - Nastínit způsoby identifikace trestné činnosti v rámci DARKNETu
   - Analyzovat možnosti dokumentace trestné činnosti
   - Analyzovat možnosti identifikace pachatele
   - Definovat možnosti monitoringu protiprávního jednání

Hlavní cíl práce:

1. Vytvořit metodiku řešení kybernetické kriminality v rámci DARKNETu pro použití orgánů činných v trestním řízení.

   Základním kritériem pro splnění či nesplnění hlavního cíle práce bude použitelnost navržené metodiky v praxi. Metodika musí být dostatečně konkrétní a musí obsahovat všechny eventuality, které mohou při vyšetřování páchání trestné činnosti v prostředí DARKNETu nastat. Zároveň metodika musí být dostatečně robustní na to, aby nebylo možné (nebo extrémně obtížné) na základně procesních či technických pochybení napadnout předložený důkazní materiál v rámci soudního řízení.

   Pokud z teoretické části práce vyplyne potřeba tvorby metodiky pro každý typ trestné činnosti zvlášť (tj. nebude možné aplikovat navrženou metodiku na všechny typy trestných činů páchaných v prostředí DARKNet), pak bude pro každý typ či kategorii trestných činů vytvořena metodika samostatně (tzn. „per use case“)

se zohledněním, že mohou vycházet ze stejného základu (např. zabavení zařízení, ze kterého byla s největší pravděpodobností trestná činnost v prostředí DARKNET páchána).

2. Nasimulování aktivního a pasivního útoku pomocí Tor simulátoru Shadow. Výsledky tohoto nasimulovaného útoku budou poté přidány do metodiky a budou sloužit jako možnosti pro identifikaci pachatele.

Základní doporučená literatura :

1. Fratepietro F., Rossetti P., DEFT User Guide, http://www.deftlinux.net/
2. Carrian B., File Systém Forensic Analysis, Addison Wesley Professional, ISBN: 0-32-126817-2
3. Digvijaysinh Rathod, Darknet Forensics, Institute of Forensic Science, Gujarat Forensic Sciences University, Inida, ISSN 2278-6856
4. https://www.dataforensics.org/tor-browser-forensics/

Financování práce: .....

Vedoucí práce: .....podpis: .....

U externích vedoucích fakultní garant práce.....podpis: .....

Vedoucí katedry, kde proběhne obhajoba .....podpis: .....

Případný souhlas vedoucího ústavu AV .....podpis: .....

V Českých Budějovicích dne ..... Podpis studenta: .....

BIBLIOGRAFICKÉ ÚDAJE

Rýc, T., 2020: Analýza Darknetu se zaměřením na forenzní zkoumání. [Analyse the DARKNET with focus on forensic research, Bc. Thesis, in Czech] – Faculty of Science, The University of South Bohemia, České Budějovice, Czech Republic.

Anotace

Bakalářská práce se zaměřuje na analýzu DARKNETu se zaměřením na forenzní zkoumání. V práci jsou popsána protiprávní jednání vyskytující se v síti DARKNET a jejich možný způsob forenzního zkoumání. Jsou navrženy způsoby identifikace trestné činnosti, možnosti její dokumentace a analyzována možnost identifikace pachatele. Pomocí navržené metodiky je navrženo řešení kybernetické kriminality pro použití orgánů činných v trestním řízení.

Klíčová slova

Darknet, Tor, forensic research, cybercrime

Abstract

This bachelor thesis aims to analyse the DARKNET with focus on forensic research. This paper describes a variety of illegal actions which occur in the DARKNET and possible form of their forensic research. A forms of identification of the illegal actions, possibilities of their documentation and analysis of the identification of the perpetrator are suggested. By the suggested methodology is designed a cybercrime solution, which can be used by the authorities active in the criminal proceedings.

Čestné prohlášení

Prohlašuji, že svou bakalářskou práci "Analýza DARKNETU se zaměřením na forenzní zkoumání" jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce nebo v poznámce pod čarou.

Prohlašuji, že v souladu s § 47b zákona č. 111/1998 Sb., v platném znění souhlasím se zveřejněním své bakalářské práce, a to v nekrácené podobě (nebo v úpravě vzniklé vypuštěním vyznačených částí archivovaných Přírodovědeckou fakultou) elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejich internetových stránkách, a to se zachováním mého autorského práva k odevzdávanému textu této kvalifikační práci. Souhlasm dále s tím, aby toutéž elektronickou cestou byly v souladu s uvedeným ustanovením zákona č. 111/1998 Sb., zveřejněny posudky školitele a oponentů práce i záznam o průběhu a výsledku obhajoby kvalifikační práce. Rovněž souhlasím s porovnáním textu mé kvalifikační práce s databází kvalifikačních prací Theses.cz provozovanou Národním registrem vysokoškolských kvalifikačních prací a systémem na odhalování plagiátů.

V Českých Budějovicích dne 18. května 2020 Tomáš Rýc

Poděkování

Touto cestou bych rád poděkoval vedoucímu bakalářské práce panu *Ing. Petru Břehovskému* za odborné vedení, cenné rady, připomínky, náměty, konzultace a svědomité metodické vedení v průběhu zpracování bakalářské práce. Bývalému vedoucímu mé bakalářské práce Ing. *Ing. Jaroslavu Kothánkovi, Ph.D. Paní doc.RNDR. Ivě Dostálkové Ph.D* za pomoc a konzultaci práce. Panu Ing. *Jiřímu Pokornému* za konzultaci z pohledu policejních složek. Panu Ing. *Rudolfovi Vohnoutovi* za připomínky a podněty k dokončení práce. Také děkuji své *rodině* a *snoubence* za trpělivost, morální podporu a rodinné zázemí v průběhu bakalářského studia.

Obsah

ÚVOD

Tématem bakalářské práce je analýza DARKNETu se zaměřením na možnosti forenzního zkoumání. Definuje pojem DARKNET a jeho umístění v kyberprostoru včetně rozdílů mezi DARKNETem, Deepwebem a Surfacewebem a zaměřuje se na možnosti forenzního zkoumání DARKNETu. Téma práce jsem si vybral, protože se zajímám o anonymní prohlížeče v kyberprostoru a to nejen DARKNETu a o možnosti identifikace uživatele.

V současnosti zažívají informační a komunikační technologie a na ně navázané telekomunikační služby obrovský rozvoj. S tímto je spojena vyšší konkurence na trhu, snižování cen informačních a komunikačních technologií a s tím dostupnost technologií pro širokou veřejnost. S touto dostupností technologií a služeb výrazně narůstá znalost ovládání nových technologií a softwaru uživateli. Uživatelé postupně zjišťují, že kyberprostor a v něm konkrétně internet není pouze Surfaceweb, ale i DARKNET a Deepweb. S rostoucím uměním ovládání technologií a softwaru je spojena jak legální, tak nelegální činnost uživatelů v kyberprostoru.

Práce je rozdělena na teoretickou a praktickou část. V praktické části předkládá metodiku řešení kybernetické kriminality v rámci DARKNETu pro typy trestné činnosti. Dále praktická část simuluje útoky a s nimi spojenou identifikaci pachatele za pomoci simulátoru Tor sítě **Shadow**.

Závěrem práce vyhodnocuje přínosy metodiky, a zda je možné aplikovat navrženou metodiku na všechny typy trestných činů páchaných v prostředí DARKNETu, nebo je nutné na každou trestnou činnost vytvořit metodiku samostatně.

Obsah práce je členěn do devíti kapitol, které představují hlavní okruhy DARKNETu. Kapitoly jsou dále členěny do podkapitol, ve kterých je pak daný problém podrobněji rozebrán. **První kapitola** řeší otázky cílů a použitých metod při zpracování bakalářské práce. Ve **druhé kapitole** je rozebrán kybernetický prostor se zaměřením na surfaceweb, deepweb a darkweb. **Třetí kapitola** popisuje trestnou činnost v kybernetickém prostoru. Shrnuje trestné činy proti utajování, integritě a dostupnosti dat a systémů, trestné činy související s počítači, obsahem a porušováním autorských a souvisejících práv. Způsob identifikace trestné činnosti v Dakrnetu je obsažen ve **čtvrté kapitole**, která popisuje technologie, komunikaci v síti Tor, výhody a nevýhody Toru, analýzu informací a nastavení routerů uvnitř Tor sítě. **Pátá část** je zaměřena na analýzu možnosti dokumentace. **Šestá část** klade důraz na zabezpečení při vstupu na Darknet pro orgán činný v trestním

řízení. Analýza možností identifikace pachatele je uvedená v s**edmé části** simulace útoků na Tor síť za pomoci použití simulátoru Shadow je uveden v **osmé části** a popis forenzního software pro získání informací je uveden v **deváté části**. Výsledkem práce je vytvoření metodiky řešení kybernetické kriminality pro použití orgánů činném v trestním řízení, která je uvedena v **desáté části**.

1 Cíl práce a metodika

Cílem práce je vytvořit metodiku řešení kybernetické kriminality v rámci DARKNETu pro použití orgány činných v trestním řízení. Vzhledem k tomu, že v práci je řešena kybernetická kriminalita v trestním řízení dle zákona č. 40/2009 Sb., trestní zákoník, není v práci řešena z hlediska zákona č. 500/2004 Sb., správní řád a zákona č. 89/2012 Sb., občanský zákoníku. Tím ale není řečeno, že v rámci těchto zákonů nemůže v síti DARKNET dojít k protiprávnímu jednání. Dále popisuji způsoby jak se připojit k tomuto prostoru s důrazem na bezpečnost připojení a nastiňuji způsoby identifikace trestné činnosti, možnosti její dokumentace a analyzuji možnosti identifikace pachatele za pomoci simulování útoku v simulátoru Shadow. V neposlední řadě popisuji možnosti monitoringu protiprávního jednání v tomto prostoru.

Bakalářská práce je zpracována s použitím systémového přístupu (*nashromážděním nezbytného množství teoretických podkladů pro zpracování práce a způsobu uspořádání práce*) a aplikací metod:

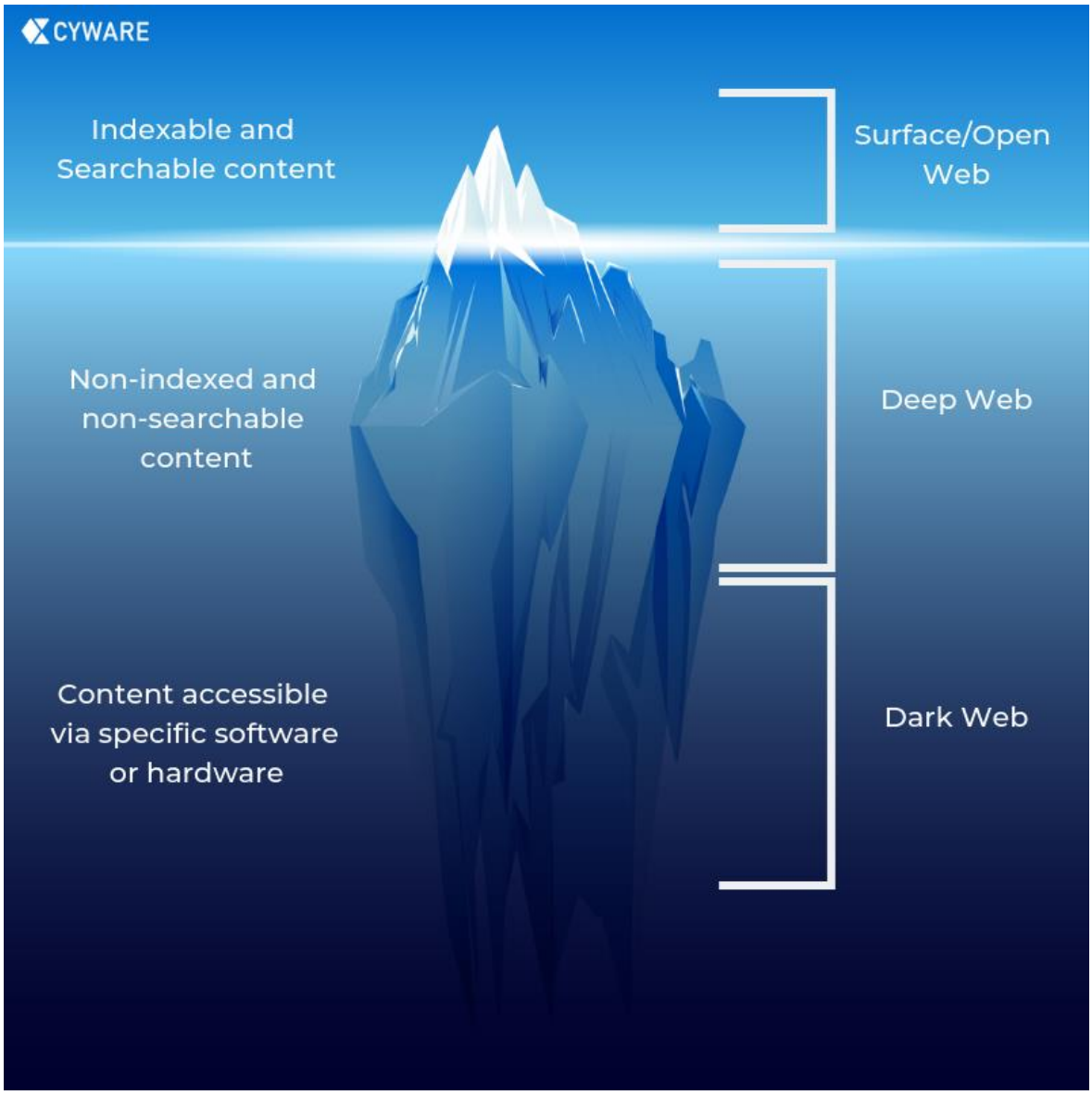
- analýza – k hodnocení kybernetického prostoru,
- syntéza – ke shrnutí jednotlivých kapitol v dílčím závěru v jeden celek,
- indukce – pro stanovení obecných závěrů trestné činnosti v kybernetickém prostoru,
- deskriptivní (*popisný způsob poznávání*) a historické metody (*heuristika – shromáždění relevantních zdrojů, bibliografie*) pro popis dosavadních poznatků z dané oblasti,
- dedukce – při vyvozování závěru na základě obecných a známých skutečností, předpokladů a tvrzení.

Z vlastních poznatků získaných studiem byla použita metoda zobecnění problému a snaha upozornit na důvody zabezpečení při vstupu na Darknet, včetně nastavení prohlížeče a sociální inženýrství. Dále byly využity poznatky a připomínky expertů činných v trestním řízení, vědomosti získané při řádném studiu a ostatní praktické zkušenosti spojené v rámci využívání sociálních sítí.

2 Kybernetický prostor

Ještě do roku 2014 nebyl v České republice pojem kybernetický prostor legislativně definován. To se změnilo 23. července 2014, kdy byl schválen zákon č. 181/2014 Sb., o kybernetické bezpečnosti a o změně souvisejících zákonů (dále jen ZKB). Pojem kybernetický prostor je vymezen v §2, písm. a), ZKB takto: „*Kybernetickým prostorem je digitální prostředí umožňující vznik, zpracování a výměnu informací, tvořené informačními systémy, a službami a sítěmi elektronických komunikací*.“

Z výše uvedené definice je nutné si uvědomit, že kybernetickým prostorem jsou všechny informační systémy s jejich službami a sítěmi elektronických komunikací, které zabezpečují jejich propojení. V praxi se můžeme setkat s informačními systémy nepřipojenými do globální sítě Internet (dále jen Internet). Zpravidla se jedná o utajované sítě (státní, ale i nestátní) či privátní sítě společností, které z hlediska jimi definované bezpečnostní politiky nepřipouští fyzické připojení do Internetu. Pak jsou informační systémy, které jsou připojeny do Internetu. Tyto informační systémy můžeme chápat podle velikosti od minimálních (např. domácí PC a chytrý mobilní telefon) připojených pomocí wifi/routeru do internetu až po složité informační systémy (desítky či stovky PC se servery a službami) připojených komplexními demilitarizovanými zónami s bezpečnostními prvky k Internetu. Všechny tyto informační systémy tvoří neomezenou, v čase se měnící síť Internet, která nezná mezinárodní hranice. Množství informací a dat, které je obsaženo v Internetu má obrovský ekonomický potenciál. To přináší zájem uživatelů a s tím spojenou jak legální tak nelegální činnost uživatelů v tomto kybernetickém prostoru. Některé publikace přirovnávají kybernetický prostor k ledovci [13], který je rozdělen na tři části: Surfaceweb, Deepweb a Darkweb a to podle aplikační vrstvy v rámci sítí a služeb. Často se uvádí, že Deepweb a Darkweb společně tvoří Darknet, toto tvrzení ovšem není pravdivé, Darknet je podsítí Deepwebu.



Obrázek 1 Kybernetický prostor

2.1 Surfaceweb

Pojmy surface a deep web jako první uvedl Michael Bergman ve studii The Deep Web: Surfacing Hidden Value [14] v roce 2001. Jedná se o stránky (Youtube, Facebook, Google), které jsou indexované a je možné se k nim připojit bez autentizace pomocí nám známých prohlížečů (*Firefox, Google Chrome, Internet Explorer, Opera*).

2.2 Deepweb

Jedná se o různé stránky, které nejsou z jakýchkoli technických příčin indexovány, nebo se jedná o soukromé stránky, kde si vlastník nepřeje indexaci anebo o speciální stránky, kam je možné se dostat jen po splnění určitých podmínek např. autentizačních. Např. účet na stránce (Facebooku, Youtube), e mail účet (gmail, seznam), fotka přidaná na Facebooku. Deepweb tvoří odhadem 96% veškerého obsahu Internetu.

2.3 Darkweb

Darkweb jsou stránky nepřístupné pomocí klasických prohlížečů (*Google Chrome, Mozilla Firefox, Opera, Internet Explorer*). Jde o anonymní šifrované sítě fungující uvnitř Internetu, na kterých běží vlastní služby. Těmto službám se říká hidden service. K přístupu na Darkweb je zapotřebí použít speciální software – prohlížeč (Tor, I2P, Freenet, DN42).[1]

Pojem DARKNET je v této práci chápán jako Darkweb kdy se jedná o prostředí Internetu, do kterého je možný přístup pouze prostřednictvím speciálního softwaru – TOR. Tato práce se bude zaměřovat na hidden service a služby HTTP/S.

3 Trestná činnost v kybernetickém prostoru

Úmluva Rady Evropy č. 185 ze dne 8. listopadu 2001 (dále jen Úmluva) o kyberkriminalitě sjednocuje národní právní úpravy v oblasti kyberkriminality. Česká republika ratifikovala Úmluvu 22. srpna 2013 s platností k 1. prosinci 2013. Na základě této Úmluvy byly do právních řádů České republiky implementovány takové nástroje, aby bylo možné postihovat kybernetické trestné činy (dále jen TČ). Tato kapitola vychází z této Úmluvy [35].

Úmluva definuje čtyři základní skupiny TČ v kyberprostoru:

3.1 Trestné činy proti utajování, integritě a dostupnosti počítačových dat a systémů

Jedná se o TČ, který pachatel způsobí úmyslně s cílem neoprávněně přistoupit k celému počítačovému systému nebo k jeho části. Tento TČ je definován v §230 odst. 1 TZK **neoprávněný přístup a zásah do počítačového systému** a nosiče informací a jedná se zpravidla o hacking, cracking nebo computer trespass.

V případě, že pachatel využije k přístupu do systému malware, jako prostředek útoku, lze tento neoprávněný přístup přiřadit k § 230 odst. 2 TZK.

TČ je i skutek kdy pachatel úmyslně provádí odposlech neveřejných zpráv. Tento trestný čin je definován v §182 TZK **porušení tajemství dopravovaných zpráv**, metodou sniffingu.

Dalším TČ je zásah do dat a to úmyslným **poškozením nebo vymazáním** za předpokladu že došlo k závažné škodě. TČ je definován v § 230 odst. 2 písm. a) a b) TZK a je způsobován útoky malware, DoS a hackingem.

V případě **výroby, prodeje zařízení případně i softwaru vytvořeného za účelem úmyslného spáchání TČ** uvedených v bodě 3.1 je možné tyto činnosti postihnout dle § 230 odst. 2 a odst. 3 TZK.

3.2 Trestné činy související s počítači

Tyto TČ souvisí s **padělaním** a podvodem (pozměňováním) dat uložených v počítači. Musí být vykonány úmyslně pachatelem. Padělání dat uložených v počítači je postihováno dle § 230 odst. 2 písm. c) TZK.

U **podvodu** musí jít o úmysl s cílem získat sobě nebo jinému majetkový prospěch, kdy se zpravidla jedná o phishing, pharming a spear phishing. Podvod v případě podvržení stránek s cílem získání majetkového prospěchu je možné řešit § 209 TZK.

3.3 Trestné činy související s obsahem

Do těchto TČ v kybernetickém prostoru spadá držení, výroba a šíření nezákonných materiálů prostřednictvím Internetu. Z hlediska obsahu se z velké většiny jedná o TČ související s dětskou pornografií. TČ související s dětskou pornografií jsou postihovány především dle § 192 výroba a jiné nakládání s dětskou pornografií.

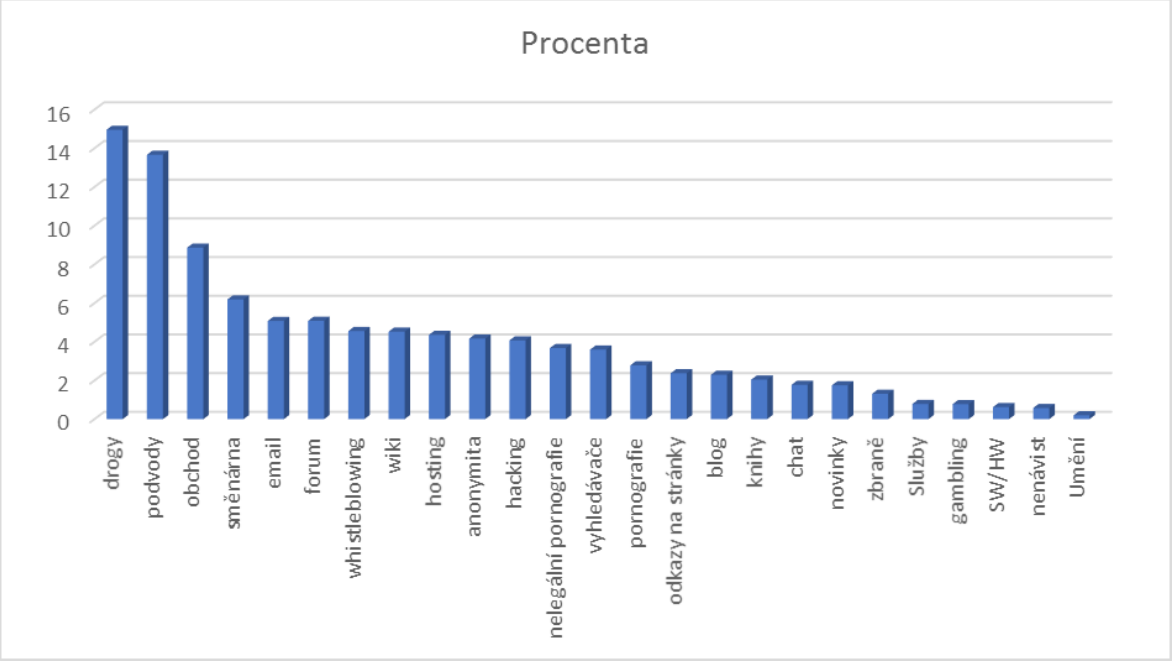
Do TČ souvisejících s obsahem dále spadá problematika šíření rasismu a xenofobie. V případě rasisticky a xenofobně motivované pohrůžky je uplatněn § 352 TZK, při rasisticky a xenofobně motivované urážce § 355 TZK. Šíření rasistických a xenofobních materiálů v kybernetickém prostoru je postiženo v § 356 a 403 TZK. Samostatnou kapitolou je popírání, hrubé snižování, schvalování nebo ospravedlňování genocidy nebo zločinů proti lidskosti, které je postihováno dle § 405 TZK.

3.4 Trestné činy související s porušením autorských práva a práv souvisejících

V kyberprostoru je velmi rozšířeným TČ porušování autorských práv formou internetovým pirátstvím, crackingem a warezem. Trestní právo postihuje tyto činnosti podle § 270 TZK porušení autorského práva, práv souvisejících s právem autorským a práv k databázi. U těchto TČ je nutné vyhodnotit účinek, tedy jak bylo zasaženo do chráněných práv a nikoliv nepatrně. Zásah do chráněných práv tedy musí být nikoli nepatrný, aby byla naplněna skutková podstata TČ.

3.5 Trestné činy vyskytující se na DARKNETu

V oblasti DARKNETu, jehož rozsah jsem pro tuto práci objasnil v bodě 1.3, je četnost činností ať již legálních či nelegálních uvedena v procentuálním vyjádření v Grafu 1. Ke zjištění TČ vyskytujících se na Darknetu jsem použil pět prací [2][3][4][5][6] ve kterých jsou statistické údaje nejen o TČ v DARKNETu. V těchto pracích se v podobě grafu objevují hidden services (dále jen HS), což jsou vlastně servery uvnitř DARKNETu, které jsou přístupné pouze pomocí speciálních prohlížečů. Tyto HS v těchto pracích jsem zprůměroval, abych došel ke zjištění kolik procent HS s určitým obsahem se vyskytuje uvnitř TOR sítě. Z tohoto důvodu je možné, že se některé HS mohou opakovat. Dohromady se vyskytovalo 53 412 HS. Některé z těchto prací neměli přesný počet serverů, takže jejich počet byl stanoven podle procentuálního grafu. Jedná se pouze o přibližný procentuální počet, ale pro představu o obsahu činností v DARKNETu je postačující.



Graf 1 - odhadovaný procentuální výskyt HS uvnitř Tor sítě

Z grafu je patrné, že na DARKNETu se vyskytuje tato TČ:

a) Trestné činy proti utajování, integritě a dostupnosti počítačových dat a systémů, konkrétně – hacking, prodej nelegálního SW a HW určeného k úmyslnému spáchání trestných činů.
b) Trestné činy související s obsahem v podobě nelegální pornografie a šíření rasistických a xenofobních materiálů – nelegální pornografie, nenávist atd.
c) Trestné činy související s porušováním autorských práv a práv prodejem nelegálního softwaru – Whistleblowing, SW/HW obchod.

4 Způsob identifikace trestné činnosti v DARKNETu

Před samotným popsáním způsobu identifikace TČ v DARKNETu je nutné pochopit technologii TOR včetně komunikace v síti a zabezpečení technologie před případným útokem.

Následně při dostatečném zabezpečení vlastních technologií je možné přistoupit do sítě DARKNET a provádět identifikaci trestné činnosti v postupných krocích. Nejdříve je nutné provést vyhledání maximálního počtu serverů v síti. Po jejím vyhledání následuje vyhodnocení obsahu podle stanovených závadných slov na jednotlivých serverech. Díky tomuto kroku se podaří snížit počty serverů s možným závadným obsahem a tyto vyfiltrované servery budou podrobeny hloubkové analýze, která odhalí servery s aplikacemi na prodej nezákonného materiálu, fóra, blogy s nezákonným obsahem apod. Tato hloubková analýza by zároveň měla odhalit, zda informace na serveru naplňují skutkovou podstatu trestných činů, a to jednoho či více skutků. Poté je možné zahájit dokumentaci trestné činnosti, která bude popsána v kapitole 5.

4.1 Technologie Darkentu – TOR

Pro přístup do sítě DARKNETu musíme použít speciální software nebo konfiguraci. Jedná se o speciální prohlížeče jako např. Freenet, I2P a TOR. Tato práce je zaměřena na nejrozšířenější speciální prohlížeč TOR. TOR byl vytvořen za účelem uchránění anonymity uživatele.

4.1.1 Technologie Darkentu – TOR

Komunikace po sítí funguje pomocí Onion Routeru (dále jen OR), každý OR funguje jako user level process (má svoji adresu a fyzickou paměť). OR komunikuje s dalšími OR v síti přes TLS protokol. Uživatel, když chce navázat spojení, spustí software, který se nazývá Onion Proxy (dále jen OP). OP slouží k načtení adresářů, vytvoření obvodu skrze síť a zpracovává spojení. OP přijímají Transmition control protocol (dále jen TCP), díky kterému můžou mezi sebou vytvořit spojení.

Každý OR poté podepíše TLS certifikaci, router descriptor (souhrn klíčů, adres, šířek pásma a další) a adresáře. Tyto podpisy jsou podepsány dlouhým identifikačním klíčem. OR používá ještě krátký onion klíč, který slouží k dešifrování žádosti, vytvoření obvodu a vyjednání ephemeral klíče (z důvodu komunikace mezi OR pomocí Diffie-Hellman handshake).

OP vyjedná s každým OR v obvodu symetrický klíč, s každým jednotlivě. Pro vytvoření obvodu OR pošle create cell šifrovanou onion klíčem za použití polovičního Diffie-Hellman handshake prvnímu uzlu v obvodu. Pro zvýšení obvodu OP klienta pošle za použití poloviční Diffie-Hellman handshake router extend cell přes první uzel v obvodu OR1. OR1 poté zkopíruje poloviční Diffie-Hellman handshake a pošle create cell přes druhý uzel OR2. OR2 zná pouze OR1, klient ho nepotřebuje znát. Jakmile OR2 odpoví OR1 s vytvořením cell. OR1 poté přepošle pomocí router extended cell klientovi. Pro vytvoření OR3 stačí klientovi, aby vydal poslednímu uzlu příkaz k vytvoření jednoho hopu navíc. Obvod tvoří implicitně 3 OR, z důvodu bezpečnosti klienta, ale je možné zvýšit počet routerů.

Každý OR (OR1, OR2 a OR3) v obvodu má svůj onion klíč, kterým může šifrovat nebo dešifrovat komunikaci, OR1 zašifruje svým klíčem a pošle OR2, OR2 zašifruje svým klíčem a pošle OR3, OR3 zašifruje a pošle dál. Při odpovědi nazpět OR3 dostane odpověď a dešifruje ji svým klíčem a přeposílá OR2, OR2 dešifruje svým klíčem a pošle OR1, OR1 dešifruje svým klíčem a přepošle klientovi.[10]

Dešifrování komunikace při spojení se serverem probíhá v 7 krocích:

1. OR1 dešifruje pomocí K1 a předá tuto zprávu dál na OR2, tato zpráva je stále šifrovaná.
2. OR2 dešifruje pomocí K2 a pošle na OR3.
3. OR3 dešifruje zprávu pomocí K3 a může přečíst, co se v ní nachází, protože už není šifrovaná. OR3 přečte zprávu, která bude obsahovat např. „spoj mě se serverem X“. OR3 se může spojit se serverem X. Nyní pošle odpověď zpět pomocí opačného postupu.
4. OR3 zašifruje pomocí K3 a pošle OR2.
5. OR2 zašifruje pomocí K2 a pošle OR1.
6. OR1 zašifruje pomocí K1 a pošle ji zpět klientovi.
7. Klient má všechny 3 klíče, takže může dešifrovat zprávu. [8]

Kdyby poté chtěl útočník napadnout komunikaci, viděl by na prvním hopu pouze to, že se uživatel přihlásil do Tor sítě. Pokud by zaútočil na druhém hopu, viděl by, že pouze probíhá nějaká šifrovaná komunikace. Pokud by zaútočil na posledním hopu, viděl by, že se nějaký router připojuje na server, ale nebude vědět, kdo to je.

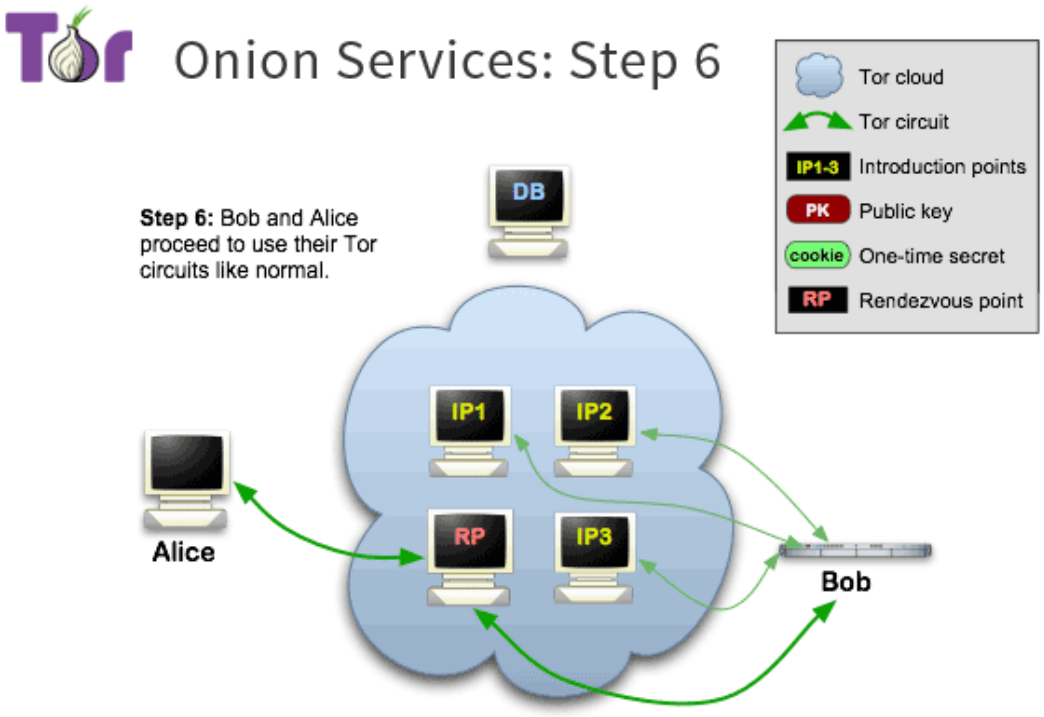
4.1.2 Stránky - Hidden services

V rámci TOR sítě se vytváří HS což jsou stránky uvnitř této sítě s koncovkou .onion. K těmto stránkám je možné se připojit jedině tak, že budeme uvnitř této sítě.

Když chce klient navázat spojení s HS, vytvoří obvod 3 OR, 3. OR kontaktuje Directory server (dále jen DS), aby mu sdělil informace o HS včetně adres introduction pointů (dále jen InP). InP jsou serverem vybrané routery, které znají jeho adresu. Klient poté vybere router, aby se choval jako Rendezvous point (dále jen RP), který bude fungovat jako spojka mezi klientem a serverem (Klient se připojí k RP přes OR1 a OR2, RP se poté chová jako OR3. Server se připojí k RP pomocí svých vytvořených OR). Poté RP kontaktuje InP, aby server věděl, že se s ním chce klient spojit přes RP. InP přepošlou tuto zprávu serveru, který se rozhodne, jestli naváže s RP spojení nebo ne. Pokud ano, tak server kontaktuje RP, že chce navázat spojení. RP poté přepošle žádost klientovi. Nyní může klient komunikovat se serverem. [28]

Spojení mezi klientem a stránkou se vytvoří následovně:

1. Server vytvoří 3 náhodné OR, které se nazývají InP. Sdělí o těchto InP Directory serveru, kdyby se k němu někdo chtěl připojit, aby věděl, kde ho můžeš kontaktovat.
2. Pokud chce klient navštívit server, musí znát jeho onion adresu, která se skládá z 16 písmen odvozených z veřejného klíče patřícímu serveru. Klient posílá žádost na Directory server.
3. Po zadání adresy onion serveru bude klient znát veřejný klíč a InP. Klient vytvoří pomocí náhodného routeru RP v síti a dostane jednorázové cookie.
4. Klient sestaví přivítací zprávu, ve které bude obsahovat RP a jednorázové cookie, zašifruje pomocí veřejného klíče a zašle na náhodný InP.
5. Dešifruje se klientova přivítací zpráva a vytvoří se obvod do RP na kterou zašle jednorázové cookie.
6. Klient a server mohou pomocí svého obvodu navázat komunikaci přes RP. [8][9]



Obrázek 2 Konečná komunikace serveru a klienta

4.1.3 Výhody a nevýhody TORu

Výhodou je, že jde o úplnou anonymitu, jelikož informace jde přes 3 routery v síti a útočník nemá možnost tyto data napadnout (výjimky budou popsány dále v této práci).[11]

Nevýhodou je, že za cenu anonymity je komunikace pomalá. Vždy, když chci navštívit nějaký server, musím projít přes 3 routery, což samozřejmě zpomalí komunikaci.[11]

4.2 Vyhledávání v TOR síti

Orgán činný v trestním řízení může na TČ narazit pomocí vyhledávačů uvnitř Darknetu nahlášením oběti nebo svědka. Zároveň je možné informace získat pomocí routerů uvnitř sítě. Vstupní a výstupní routery jsou největším ziskem informací. Vstupní router získá informaci o uživateli, který se připojil do Tor sítě. Výstupní router získává dotaz na vstup do serveru .onion. Pokud by orgán činný v trestním řízení vytvořil více výstupních routerů, mohl by monitorovat servery, které se na Darknetu objevují, protože by věděl, kam se OR připojuje. Tyto servery by poté mohl prozkoumat, zda se na nich nenachází trestná činnost.

4.2.1 Vyhledávací služby

Pro vyhledávání HS uvnitř Darknetu slouží vyhledávací služby určené přímo pro Tor, tyto služby fungují tak, že sbírá .onion URL z Tor sítě, pokud si HS nepřeje být indexovaný může to napsat do robots.txt. Mezi nejznámější patří například *DuckDuckGo, Torch, Ahmia*. Po zadání vyhledávaného řetězce vyhledávací služba nabídne servery ze surface webu, ale také HS.

4.2.2 Analýza informací

Po získání .onion stránek je možné stáhnout obsah webu do počítače (viz dokumentace stránek), poté vypsat klíčová slova TČ. Je možné použít software Mallet[29] nebo uClassify[30] (text classifier) pro automatické rozdělení témat stránek [4]. Je možné použít Support Vector Machine (SVM) [5], neuronová síť, která rozpozná, co se na stránce objevuje pomocí text classifier. Darknet Usage Text Addresses (DUTA) obsahuje pouze HS, získává informace ze stránek pouze s portem 80 (http) [6].

4.2.3 Nastavení routeru uvnitř Tor sítě

Router se může nastavit buď jako vstupní/střední nebo výstupní. Vstupní router je router mezi klientem a středním routerem. Střední router je mezi vstupním a výstupním routerem. Vstupní routery potřebují mít stabilní rychlost připojení alespoň 2 MB za sekundu jinak z nich budou střední routery. Výstupní router je mezi druhým routerem a serverem. Tor vyžaduje, aby měl výstupní router připojení více, než 100 Mb za sekundu. Podrobný způsob nastavení vstupního nebo výstupního routeru viz [28].

Po nastavení výstupních routerů, pokud by se někdo připojoval nešifrovaně, na výstupním routeru by byl čistý text a tím můžeme získat stránku pouhým poslouchánímkomunikace.

5 Analýza možnosti dokumentace

5.1 Dokumentace TČ

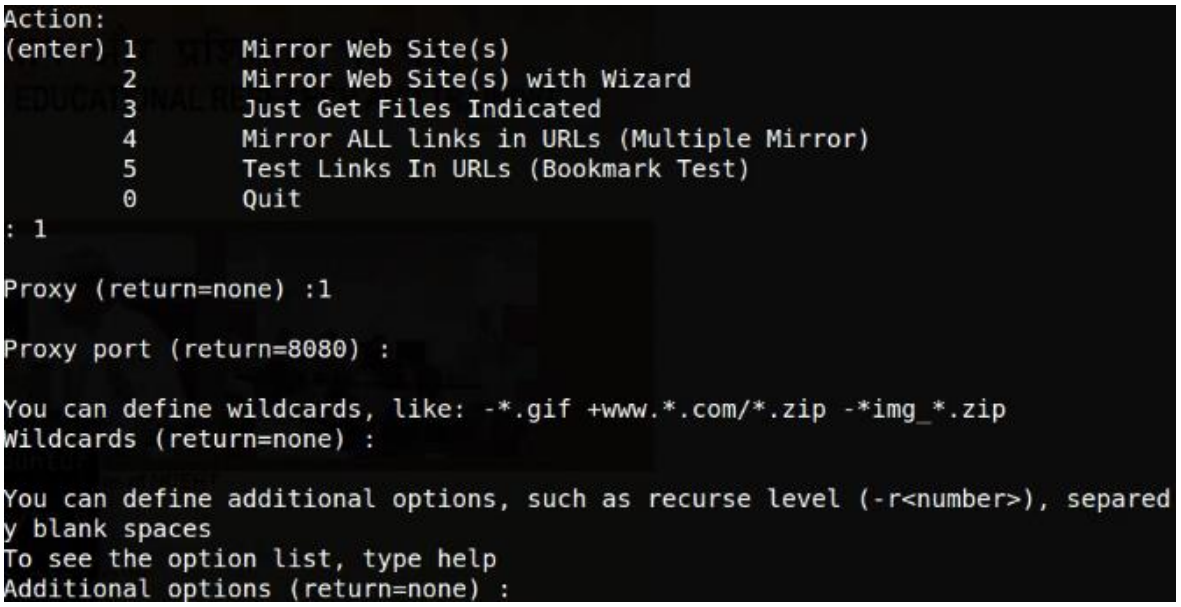
Při nalezení stránky, na které se nachází nelegální obsah je zapotřebí tuto stránku nahlásit orgánu činnému v trestním řízení, který potřebuje tuto stránku uložit lokálně. Pokud by se útočník pokusil smazat nebo změnit obsah stránky, bude uložená na počítači a může sloužit jako důkazní materiál. K tomuto účelu slouží software pro stažení http a https stránek.

5.2 Možnost dokumentace

Existuje více programů, které jsou schopny zadokumentovat stránku na internetu. Tyto programy fungují i uvnitř Darknetu, ale je potřeba, aby tyto programy věděly, že budou komunikovat přes proxy server, přes který komunikuje Tor. Je potřeba nastavit proxy server na 127.0.0.1:8080, na kterém komunikuje Tor proxy, tím budou programy na dokumentaci vědět, kde komunikovat s Torem.

Speciální program pro dokumentaci stránek je např. httrack.[46] Tento program je možné nainstalovat na Windows i Linux.

Poté je potřeba v terminálu použít příkaz pro nastavení nastavení localhost na 9050: `polipo socksParentProxy=localhost:9050` [47], localhost se nastaví na 9050 což je implicitní port pro Tor a můžeme dokumentovat stránku. Použít příkaz httrack, kde je potřeba poté napsat jméno projektu, vybrat cestu projektu, zadat adresu serveru xyz.onion vybrat akci 0-5 (1 – Mirror stránky, 2 – Mirror stránky s Wizardem, 3 – Pouze označit soubory, 4 – Mirror všech linků v URL, 5 – Testovat linky v URL, 0 – exit), nastavit Proxy na localhost:8080, definovat další nastavení, a nakonec zadokumentovat stránku.



Obrázek 3- dokumentace obrázku pomocí httrack 1 – Mirror stránky, 2 - Mirror stránky s Wizardem, 3 – Označit pouze soubory, 4 – Mirror všech linků v URL, 5 – Testovat linky v URL, 0 – exit, nastavení Proxy portu a přidání dalších možností

Linux má přímo příkaz v terminálu, který dokáže zadokumentovat stránku, pomocí příkazu `torify wget --mirror xyz.onion`.

6 Zabezpečení při vstupu na Darknet pro orgán činný v trestním řízení

Pohybovat se v Darknetu znamená být také neviditelný pro ostatní. Toho musí být samozřejmě schopni i vyšetřovatelé, kteří hledají v dané části Internetu osoby páchající protiprávní jednání.

6.1 Důvod zabezpečení

Tor síť je jedna z nejlepších anonymních sítí, vyskytují se v ní však chyby, které snižují její bezpečnost. Útočník může zaútočit a zjistit vaši IP adresu. Také může napadnout účet, popřípadě napadnout počítač pomocí viru. Ze všech těchto důvodů je důležité k Darknetu přistupovat anonymně a chránit si své soukromí a bezpečí.

Základním předpokladem pro to být neviditelný je skrytí IP adresy, která je viditelná i v Darknetu.

K tomu lze využít následující varianty:

a) Pořízení veřejné IP adresy přes prostředníka (firmu)
b) Pořízení a připojení přes VPN službu
c) Připojení přes vlastní server

6.2 Pořízení veřejné IP adresy

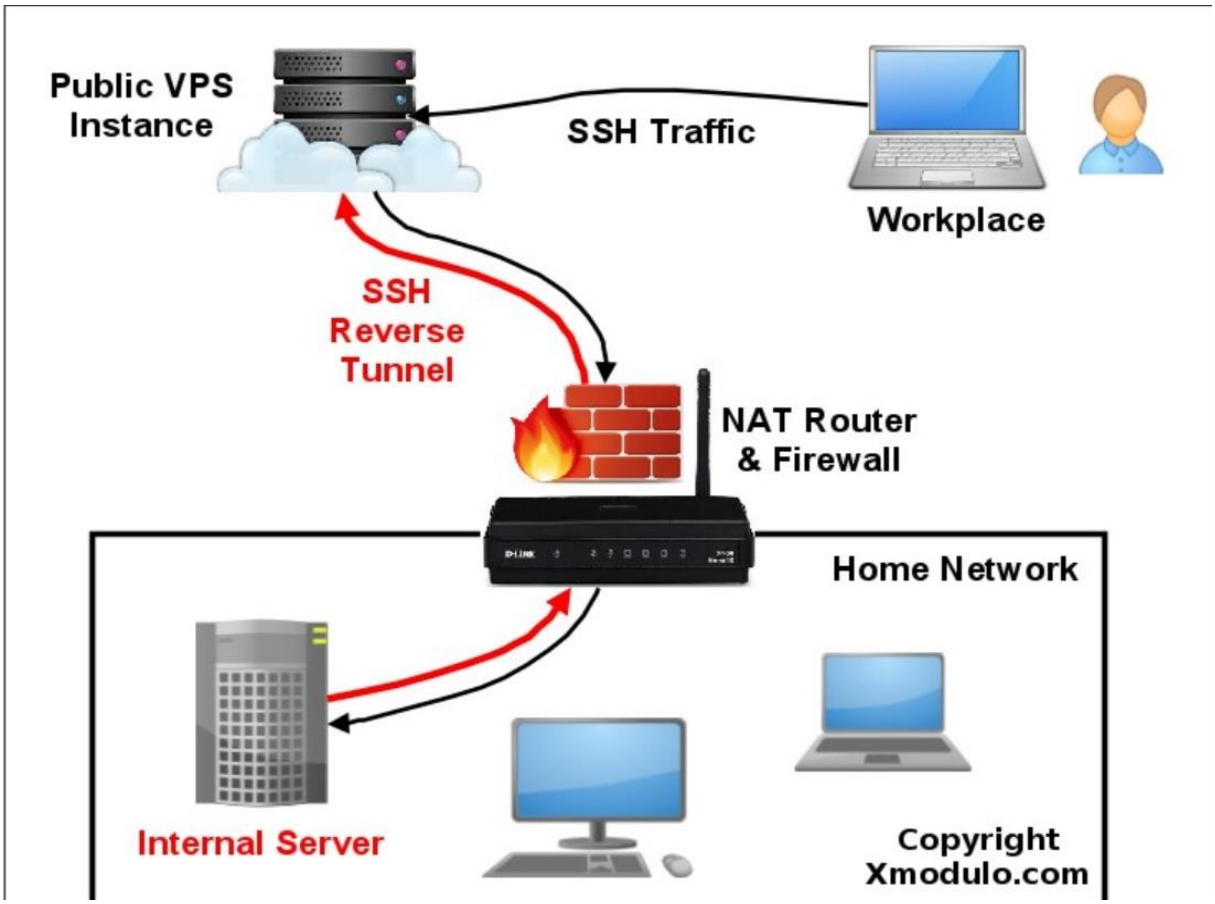
Jedná se o nejjednodušší způsob, jak zakrýt svou identitu v rámci Darknetu. Pokud se totiž využije a zaregistruje jiná osoba než ta, která dané připojení používá, pak v rámci služby whois bude dohledatelná pouze tato osoba. Jako příklad můžeme uvést doménu pcr.cz, kdy se dozvíme výpisem whois, že je zaregistrována na Pavla Smrže, k registraci došlo 10. 8. 2001 a doména je vedena pro ministerstvo vnitra. Z výše popsaného vyplývá, že používanou IP adresu PČR by si mohl prověřit kdokoliv z prostředí Darkentu a následně vyvinout sadu otázek a při nich by mohl odhalit, že se nejedná o osobu, která je vedená pod výpisem whois. Proto je potřeba, aby orgán činný v trestním řízení použil další ochrany své identity a nepřistupoval ze svého rozsahu IP adres. Tato metoda je asi nejméně bezpečná z hlediska zakrytí identity, ale zároveň nejrychleji realizovatelná.

6.3 Pořízení a připojení přes VPN službu

V dnešní době existuje mnoho služeb, které nám pomohou s přístupem na Internet a také nám pomohou na něm bezpečně a zakrytě surfovat. Jednou z takových služeb je VPN, kdy k přístupu na Internet dochází tzv. přes prostředníka, v tomto případě VPN službu. VPN služby nabízí mnoho společností a ceny se liší dle počtu serverů, na kterých daná služba běží, ale i taky za kolik zemí se lze schovat. V tomto případě se z výpisu whois osoba z prostředí Darkentu nedozví nic víc, než že používáme VPN od daného poskytovatele. To se samozřejmě jeví z pohledu identifikace jako ideální, ale má to také svá úskalí. Jako největší vidím to, že veškerá komunikace je vedena skrz prostředí, o kterém PČR nic neví, takže lze provést na dané spojení tzv. man-in-the-middle útok. Spojení mezi klientem (PČR) a serverem (služba VPN) je sice zašifrované a zároveň i dál je komunikace šifrovaná, ale certifikát, kterým došlo k zašifrování, vydala a vlastní firma, která nám spojení realizuje. Je důležité si uvědomit, že v rámci Darkentu se dlouhou dobu buduje důvěra, někdy trvá i roky, než se člověk dostane k osobě, která je hlavním pachatelem, a proto bych ani tuto variantu neviděl jako nejvhodnější, protože právě čas může způsobit, že služba přestane fungovat, případně dojde k narušení integrity dat a tím pádem veškerá posbíraná data budou dále nepoužitelná v rámci důkazního břemene.

6.4 Připojení přes vlastní server

Asi nejbezpečnější, avšak také nejvíc časově náročné, je připojení přes vlastní prostředky, ideálně server. Na něm jsme schopni spouštět další virtuální servery a s nimi i spojené služby, tudíž nám dává relativně velkou volnost. Základním předpokladem je, že server bude připojen v rámci infrastruktury, ke které má přístup pouze omezený počet lidí z PČR. Ideální pro tyto potřeby by se hodil pronajatý byt s připojením od místního poskytovatele služby internet. Tento server by sloužil jako prostředník mezi klientem a přístupem do Darknetu. Nejprve by se na něm samozřejmě musela spustit služba VPN, tak aby byla vždy komunikace šifrována a nemohla být narušena integrita dat. Další služba, které by na daném serveru měla být spuštěna, by pak byl bnc bouncer, případně server, který bude realizovat spojení do Darknetu. Výhoda bnc bounceru je, že si můžete zvolit libovolný dns název místo IP adresy, tudíž při výpisu whois nikdo neví, přes co jste připojen. Připojení pak probíhá tak, že uživatel se přihlásí nejprve na vlastní server (v pronajatém bytě), kde si spustí virtuální PC, které se následně přihlásí do požadované části Darkentu, viz. obr. X níže.



Obrázek 4 - Návrh připojení PČR do DARKNETu

Díky tomu, že je vše v naší správě, je pak možné se přihlásit odkudkoliv na světě a následně se tvářit jako že jsem doma a klidně se zájmovou osobou diskutovat, jako bych byl pořád na jednom místě. Výhoda tohoto řešení je, že navázaní důvěry, která je v tomto případě nejdůležitější pro odhalení trestné činnosti, je jednodušší, než kdybychom byli připojeni přes různé VPN služby či proxy servery. A protože pachatel bude chtít znát dřív nebo později naší identifikaci a zeptá se na něco, co si bude moc ověřit, pak je důležité být stále konstantní, což nám tento způsob připojení umožní.

6.5 Zabezpečení koncového zařízení

Při vstupu na Darknet používejte antivir, protože je to nejlepší způsob zabránění stáhnutí viru do počítače a ochránění před útočníkem. Ani antivir nezachytí všechny nové viry, proto je potřeba ke zvýšení bezpečnosti používat také Virtual machine (dále jen VM), který zajistí, že při stáhnutí nějakého viru stačí přeinstalovat VM a můžete pokračovat dále.

Z důvodu bezpečnosti je rovněž důležité nezapínat Javascript, Flash. Zakázat vyskakovací okna a soubory cookies. Při používání Tor browseru neměňte rozlišení, nechte ho na implicitní velikosti z důvodu otisku prohlížeče.

Poté, co jste se ochránili dostatečně v softwaru, je důležité ochránit se i na Darknetu. Nikomu neříkat osobní informace, nezveřejňovat nikde heslo, nepřihlašovat se na žádné stránky, které se vyskytují na internetu (Facebook, Twitter, Youtube, Google… tyto stránky o mně mají cookie záznamy, a tudíž okamžitě co se na ně přihlásím, ví, jaký jsem počítač, a kdo jsem).

7 Analýza možnosti identifikace pachatele

7.1 Možnosti identifikace pachatele

Pachatel může být, jak klient, který páchá TČ (sleduje dětskou pornografii, šíří extrémistické názory), tak správce serveru, na kterém se nachází TČ. Pachatele lze dopadnout útokem na klienta, síť Tor nebo server (HS). Dále se tyto útoky můžou dělit na aktivní a pasivní útoky.

U aktivních útoků se aktivně podílíme na modifikaci útoku. Pasivní útoky jsou ty, u nichž odposloucháváme na síti a zjistíme IP adresu (klienta, HS).

7.1.1 Aktivní útok

Útoky, při kterých je potřeba napadnout HS/OR, abychom mohli zjistit, jestli klient komunikuje s HS.

1. Denial of service (dále jen DoS) útok

Tento typ útoku je vytvořen k zahlcení služby. Může být použit proti OR tím, že útočník pošle obrovský počet packetů na OR, čímž ho zahltí a OR přestane fungovat.

Sniper attack – Je určený k identifikaci HS. V tomto útoku je potřeba, aby útočník kontroloval klienta a jeden z routerů mezi HS a RP. Útočník poté potřebuje udělat z routeru, který kontroluje HS vstupní uzel, aby zjistil lokaci HS. Aby tohoto dosáhl, musí zrušit všechny HS vstupní uzly serveru, dokud HS nevybere útočníkův router jako vstupní uzel. Jakmile se útočníkův router stane vstupním uzlem, zná lokaci HS. Toho dosáhne tím, že při komunikaci se serverem skrze OR začne posílat packet SENDME na výstupní OR, přestane odpovídat výstupnímu OR a pouze posílá SENDME packety, tím blokujeme čtení paketů a HS si vybere jiný vstupní uzel. Tento postup opakuje, dokud si HS nevybere útočníkem ovládaný HS. [12]

Cellflood attack – Klient při spojení s HS posílá packety CREATE. Pokud chce klient zvýšit svou bezpečnost a použít více než implicitní 3 OR, pošle se packet RELAY_EXTEND. Když má klient zvýšený počet OR, je packet CREATE 4x větší, než CREATE packet při implicitním nastavení. Tohoto se dá využít k napadení OR. Tento útok začne posílat velké množství packetů CREATE s využitím více OR. Napadený OR se zahltí a začne odpovídat packety DESTROY, tento OR se přestane vybírat a vybere se nový OR, tento OR může patřit útočníkovi. Funguje za stejným účelem, jako Sniper attack.[36]

2. Congestion attack

Útoky směřované na zjištění OR, které se nacházejí uvnitř obvodu.

Congestion attack by modulating traffic – Tento útok není aktuální, protože v jeho době bylo uvnitř Toru pouze pár OR, nyní je tato síť mnohem větší a proto jen zmíním, jak tento útok funguje. Útočník kontroluje server a jeden OR. Útok je vytvořen k identifikaci OR, které tvoří komunikaci mezi serverem a klientem. Aby útok proběhl úspěšně, musí se klient připojit k útočníkovu serveru. Útočníkův server posílá náhodně generované data klientovi (mezi 10-25 sekundami) a přestane posílat (mezi 30-75 sekundami), tímto se vytvoří specifický vzor. Útočníkem ovládaný OR poté vytvoří připojení s ostatními OR a zjistí jeho rychlost spojení při tomto specifickém vzoru. Pokud se rychlost spojení OR shoduje, útočník bude vědět, že OR pravděpodobně navázal spojení se serverem. Tato technika může vést k odhalení všech OR v obvodu. [26]

A practical Congestion attack – Útočník kontroluje výstupní uzel. Útočník napadne kódem Javascriptu HTML odpověď na výstupním uzlu. Javascript kód nechá klienta poslat http žádost s intervalem 1 sekundy. Tato žádost bude obsahovat čas, kdy byla poslaná. Díky tomu bude moci útočník zjistit čas na konečném uzlu, protože bude vědět, kdy přišla žádost s časem od klienta. Poté vypočítá průměrnou dobu odezvy připojení, díky tomu bude vědět, že klient poslal žádost v přesný čas, než se tato žádost objevila na výstupním OR. Útočník bude opakovat, aby měl dostatek vzorků a byl si jistý, že se jedná o pozorovaného klienta. Pokud bude odezva zpomalena s intervalem 1 sekundy a čas odpovídá, bude vědět, který vstupní uzel se připojil k výstupnímu uzlu a díky tomu zjistí, který klient se připojil k HS. [27]

7.1.2 Pasivní útok

Útoky, při kterých stačí poslouchat komunikaci. Tím můžeme zjistit, jestli se klient připojil k HS.

1. Correlation attack

Korelační útoky jsou takové útoky, kdy útočník odposlouchává na vstupním uzlu (klient vstupuje do Toru) a výstupním uzlu (připojení k serveru). V těchto útocích se sleduje korelaci provozu mezi prvním a posledním uzlem. Pokud zjistí korelaci, bude vědět, že k serveru se připojil tento klient.

Relay early traffic confirmation attack – K tomuto útoku potřebuje útočník mít přístup k HS directory routeru a vstupnímu uzlu klienta. Klient při vstupu na HS musí

nejprve požádat InP, který se nachází v HS directory. Útočník poté zjistí, že proběhla komunikace mezi klientem a HS a ví, že klient se připojil k tomuto serveru. [13]

Replay attack – Útočník si vybere packet ve vstupním uzlu a duplikuje ho. Duplikovaný packet pošle následně na stejný druhý uzel. Útočník poté může detekovat packet na posledním uzlu, který také kontroluje. Duplikovaný packet způsobí, že šifrovací a Counter Encrypted data se dostanou ze synchronizace a tím vznikne šifrovací error. Útočník může tento error vidět na posledním uzlu, pro jistotu by útočník měl zkontrolovat, jestli se error objevil až poté, co poslal packet a čas odpovídá cestě z vstupního uzlu do výstupního uzlu. Jestli se objevil error a čas odpovídá, útočník ví, že se klient připojil k HS. [14]

Cellcounter based attack – Útočník z výstupního uzlu vytvoří provoz mezi klientem a serverem. Vybere náhodný signál (např. binární sekvenci). Změní packet counter klienta a nahradí ho náhodným signálem. Útočník poté rozpozná na vstupním uzlu jeho náhodný signál. Pokud se náhodný signál shoduje, útočník ví, že klient se snažil kontaktovat HS. [15]

Low resource routing attack – Útočník kontroluje klienta a výstupní uzel k HS. Kontrolováním klienta může zjistit jeho vstupní uzel. Útočník zaútočí na vstupní uzel (DoS útokem) a tím ho udělá nedostupným. Vybere se nový vstupní uzel s možností vybrat útočníkem kontrolovaný vstupní uzel. Když je útok úspěšný a vybere se útočníkův kontrolovaný vstupní uzel, má možnost zjistit proběhlou komunikaci přes vstupní uzel, který vede přes kontrolovaný výstupní uzel a tím zjistí korelaci mezi klientem a serverem

Další příklady útoků:

Correlation based traffic attack [16]

HTTP based application level attack [18]

Bad apple attack [37]

Raptor attack [19].

2. Časové útoky

Časové útoky jsou takové útoky, které ke zjištění klienta využívají rychlosti spojení. Klient se připojuje k HS s určitou rychlostí spojení, tato rychlost spojení bude viditelná na vstupním a výstupním OR. Tento útok se může řadit jak k aktivnímu, tak k pasivnímu útoku, protože můžeme pouze odposlouchávat a zjistit rychlost spojení nebo můžeme napadnout a přetížit vstupní OR a toto přetížení sledovat na výstupním OR.

Bandwith estimation attack – Používá rychlost spojení ke zjištění klienta, identitu HS nebo identitu OR. Útok spočívá v podobě rychlosti spojení. Útočník musí ovládat HS, technologii ke zjištění rychlosti spojení a mapu obsahující vstupní a výstupní routery autonomního systému. Útočník umístí technologii ke zjištění rychlosti spojení poblíž vstupních a výstupních routerů. Pokud se klient z vstupního routeru připojí k HS přes výstupní router, vytvoří se vzor rychlosti spojení. Tento vzor je poté možné sledovat a zjistit shodu v rychlosti spojení, pomocí které dokážeme zjistit, zda se klient připojil k serveru.[25]

The indirect rate reduction attack – Tento útok sleduje změnu rychlosti spojení tím, že zaútočí na výstupní OR a sníží tak rychlost spojení mezi výstupním OR a HS. Toto snížení rychlosti spojení je poté pozorovatelné zároveň na vstupním OR. Tento rozdíl lze sledovat a zjistit, kdo se ze vstupního OR připojil přes výstupní OR k HS.[24]

3. Fingerprint attack

Útoky otisku využívají toho, že provoz má nějakou specifickou charakteristiku. Tyto útoky mohou identifikovat, kterou stránku klient chce navštívit nebo jestli se klient připojuje k HS.

Website fingerprint – Útočník kontroluje pouze vstupní uzel. Nejdříve útočník odposlouchává packety z různých HS, které by mohl klient navštívit. Útočník takto získá informace, které posílá server v packetu. Útočník kontroluje klientův příchozí a odchozí provoz. Provoz proběhne a útočník poté porovná klientovy packety a packety všech jeho vybraných serverů. Útok je úspěšný, když je ve vzoru shoda. [20]

Další příklady útoků:

Circuit fingerprint [21]

Throughput fingerprint [22]

7.2 Zastavení hidden services

Blokování HS – HS nevydrží dlouho proti technickému útoku. Je možné blokovat HS:

1. Napadením všech OR, které vlastní HS. Použít na ně Sniper attack nebo Cellflood attack, když se vybere útočníkův OR, je možné odmítat spojení, avšak k úplnému zastavení HS je potřeba ovládat všechny OR, přes které je možné se k HS připojit. Výhodou zde je, že díky ovládání OR, který komunikuje s HS známe IP adresu a lokaci HS. Nevýhodou je, že musíme ovládat všechny OR komunikující s HS.

2. Operátoři Toru mohou sami rozhodnout, že zablokují, co se na HS nachází, tím že vytvoří patch, který zakáže žádosti do HS. K tomuto je zapotřebí spolupracovat s operátory Toru. Nevýhodou je, že nezjistíme lokaci a IP adresu HS, takže pachatel může založit nový HS.

Tor může zablokovat informace, které se nacházející na HS jednoduše tím, že modifikují Tor program, aby určité stránky a klienti nemohli přijímat žádosti. Tor může využít svého postavení, aby zveřejnil jména klientů nebo zablokoval stránky. [2]

8 Simulace Tor útoků

Tato práce obsahuje simulaci aktivního a pasivního útoku. Jelikož tyto útoky naplňují skutkovou podstatu páchání TČ (popsané v kapitole 3.1), není možné tyto útoky nasimulovat uvnitř Tor sítě kvůli jejímu fungování, např. nemohu odposlouchávat výstupní OR uvnitř TORu, protože mi nepatří. Tím bych spáchal TČ. Z tohoto důvodu je potřeba nasimulovat Tor síť za použití simulátoru Shadow. Tyto simulace jsou náročné jak časově a výpočtem, tak i pamětí, a proto jsem se rozhodl použít virtuální počítače, které nabízí například Microsoft a Google.

8.1 Virtuální počítače

Spousta firem nabízí za peníze výpočetní techniku. Já jsem k této práci potřeboval rychlejší výpočet a paměť, proto jsem si vybral virtuální počítače.

8.1.1 Azure Microsoft

Azure Microsoft [38] nabízí free trial na měsíc v hodnotě 170 dolarů.

Na tomto VM jsem si vybral operační systém Linux – Ubuntu 18.04, 32 GB RAM a disk s kapacitou 100 GB za 220 dolarů na měsíc.

Microsoft nabízí připojení k VM přes SSH nebo RDP protokol. V této práci jsem použil komunikaci pomocí vzdálené plochy přes RDP protokol. Vytvořil jsem klienta pro vzdálenou plochu a povolil na VM připojení vzdálené plochy.

8.1.2 Google Cloud

Google Cloud [39] nabízí free trial na rok v hodnotě 300 dolarů.

Na tomto VM, jsem si vybral operační systém Linux – Ubuntu 18.04, 30 GB RAM a disk s kapacitou 100 GB za 220 dolarů na měsíc.

Ke Google Cloud se lze připojit přes SSH, také nabízí stažení/nahrání souborů, proto jsem nebyl nucen k používání vzdálené plochy.

8.2 Simulátor Tor sítě

Pro nasimulování útoku na Tor síť je potřeba použít simulátor Tor sítě, na výběr je Chutney [43] nebo Shadow [40], v této práci na simulování útoků byl použit simulátor Shadow.

8.2.1 Shadow

Shadow je simulátor sítě, kde lze TOR síť simulovat pomocí Tor plug-inu. Tento simulátor se používá většinou pro větší síť, ale může být použit i na malou síť. Shadow může být použít v jakémkoliv Linux prostředí bez použití roota. Komunikace klientů a serverů uvnitř Shadow je generována pomocí Traffic Generatoru Tgen. Výsledky simulace se ukládají do log souborů. Každý prvek sítě (klient, server, vstupní OR, střední OR, výstupní OR…) má vlastní log, kde se nachází informace o spojení.

Log soubory vytváří:

-tgen log, kde se nachází všechny informace o spojení:

-Server obsahuje počet čtených bytů a počet zapsaných bytů.

-Klient obsahuje počet poslaných bytů, počet čtených bytů, celkový počet navázaných spojení, počet úspěšných spojení. Také se u log souboru klienta dá zjistit, **který server navštívil**, jestli bylo spojení úspěšné nebo zda se objevil error při navazování spojení.

-tor log, kde se nachází informace o tom, jak se budoval obvod a spojení uvnitř obvodu a zda vše proběhlo úspěšně.

-OR, kde se vyskytují vstupní, střední a výstupní OR. Obsahují budování obvodu a dokončení obvodu, packety, které byly poslané na OR (například dokončení posílání packetu CREATE OR 3), názvy OR, které s nimi sousedí (OR 2 ví například, že *3FB0BD7827C760FE7F9DD810FCB10322D63AB4CF* patří OR 1).

-torctl log, tento log obsahuje informace o rychlosti spojení, port localhosta a bližší informace o obvodu OR (viz. Obrázek 5 a Obrázek 6).

8.2.2 Generování sítě

K vygenerování sítě je potřeba mít nainstalovaný Shadow [40], Shadow-plugin-tor [41] a Tgen [42].

Pro vygenerování sítě je potřeba mít příklady serverů, které chci generovat, aby Shadow věděl, jak mají tyto servery vypadat. V této práci je použito top 1000 alexa serverů.[48] Dále potřebuje znát Consensus[49], který je vytvořen každou hodinu DB, kde se vyskytují informace o OR pro všechny klienty. Tímto se Shadow dozví, jak pracují OR. Byl použit Consensus březen 2020. Server-descriptors[51] obsahuje informace, které o sobě poskytnou OR, toto chování poté Shadow aplikuje ve vygenerované síti pro všechny klienty. Byl použit Server-descriptors březen 2020. Extra-infos[50] poskytuje informace, které klient nepotřebuje znát, jsou zveřejněné vlastníkem OR, tyto informace Shadow

použije ve vygenerované síti. Byl použit Extra-infos březen 2020. Userstats-relay-country, který poskytuje statistiku, jíž stát poskytuje nejvíce OR, Shadow poté vygeneruje OR, které budou patřit procentuálně pod daný stát.

K vygenerování sítě se používá python skript generate.py, který se nachází v Tor plug-inu. Pokud chci vygenerovat síť s 3 OR, které se budou chovat jako DB. (nauths), 50 OR (nrelays), 50 klienty (nclients) a 50 servery (nservers) je potřeba použít tento příkaz:

```
python ~/shadow-plugin-tor/tools/generate.py --nauths 3 --nrelays 50 --nclients 200 --nservers 50 --fweb 1.0 --fbulk 0.0 ../../alexa-top-1000-ips.csv ../../consensuses-2020-03/27/2020-03-27-03-00-00-consensus ../../server-descriptors-2020-03/ ../../extra-infos-2020-03/ ../../userstats-relay-country.csv
```

Po vygenerovaní síť obsahuje:

-soubor shadow.config.xml ve kterém jsou uložené všechny IP adresy DB, OR, klientů a serverů, také obsahuje, jak dlouho bude trvat simulace.

-adresář shadow.data.template, ve kterém se nachází vzor sítě. Jsou v ní uložené adresáře každého OR, klienta, serveru.

-adresář conf, který slouží ke konfiguraci sítě, obsahuje torrc soubory, ve kterých se dá změnit chování sítě.

8.2.3 Simulace sítě

Poté, co máme vygenerovanou síť, je potřeba tuto síť nakonfigurovat uvnitř adresáře conf a poté spustit simulaci pomocí příkazu `shadow shadow.config.xml`, aby simulace byla rychlejší, je možné nastavit počet pracovníků a zároveň uložit do logu např. `shadow -w 3 shadow.config.xml > shadow.log`

Po dokončení tohoto příkazu, který může trvat několik hodin, se vytvoří adresář shadow.data, kde se nachází nasimulovaná síť.

8.3 Simulace útoků

Tato práce simuluje útoky proti Tor síti, zároveň pokrývá pasivní a aktivní útok. Aktivní a pasivní útoky jsou úzce spojeny, protože samotný aktivní útok mi nestačí, potřebuji také odposlouchávat síť a zjistit, zda se klient připojil k HS.

8.3.1 Pasivní útok

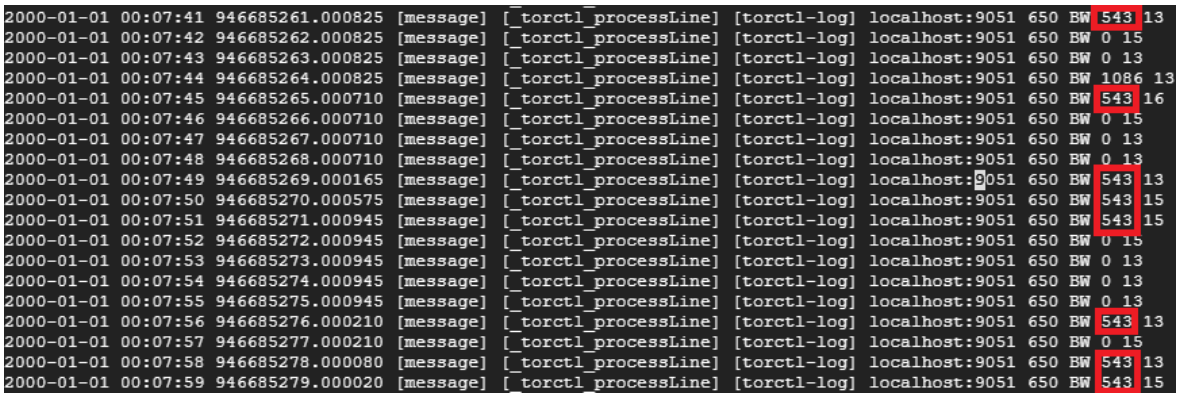
Pasivní útoky pouze sledují komunikaci a podle toho dokáží zjistit, kudy komunikace vedla.

Časový útok

Řekněme, že ovládáme vstupní OR a výstupní OR, poté dokážeme pomocí torctl logů uvnitř shadow.data zjistit, jestli komunikace probíhala mezi těmito dvěma OR podle jejich rychlosti spojení, na Obrázku 5 dole, kde se nachází vstupní OR, můžeme vidět, že přišly 3 packety za sebou s rychlostí spojení 543 a tyto 3 packety se stejnou rychlostí spojení 543 se zároveň objevily na výstupním OR uprostřed (Obrázek 6). Zároveň by se mohlo jednat o Korelační útok, jelikož najednou přišly packety v čase 00:07:29, 00:07:30 a 00:07:31 ze vstupního OR a objevily se i na výstupním OR v čase v čase 00:07:49, 00:07:50 a 00:07:51, každý packet dělila jedna sekunda a čas mezi vstupním a výstupním OR odpovídá trvání simulace.

```
2000-01-01 00:07:10 946685230.000045 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 543 1104
2000-01-01 00:07:11 946685231.000910 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 543 17
2000-01-01 00:07:12 946685232.000020 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 1086 1101
2000-01-01 00:07:13 946685233.000020 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 0 561
2000-01-01 00:07:14 946685234.000020 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 0 14
2000-01-01 00:07:15 946685235.000020 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 0 13
2000-01-01 00:07:16 946685236.000020 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 0 556
2000-01-01 00:07:17 946685237.000020 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 0 14
2000-01-01 00:07:18 946685238.000845 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 543 1099
2000-01-01 00:07:19 946685239.000680 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 1086 560
2000-01-01 00:07:20 946685240.000235 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 1629 1646
2000-01-01 00:07:21 946685241.000575 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 7818 8389
2000-01-01 00:07:22 946685242.000970 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 31299 31351
2000-01-01 00:07:23 946685243.000730 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 21538 21015
2000-01-01 00:07:24 946685244.000970 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 20995 21015
2000-01-01 00:07:25 946685245.000490 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 19876 21044
2000-01-01 00:07:26 946685246.000355 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 20381 19958
2000-01-01 00:07:27 946685247.000730 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 19280 19871
2000-01-01 00:07:28 946685248.000730 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 0 20
2000-01-01 00:07:29 946685249.000970 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 543 13
2000-01-01 00:07:30 946685250.000020 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 543 15
2000-01-01 00:07:31 946685251.000620 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 543 15
2000-01-01 00:07:32 946685252.000620 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 0 558
2000-01-01 00:07:33 946685253.000620 [message] [_torctl_processLine] [torctl-log] localhost:9051 650 BW 0 1100
```

Obrázek 5 - Nasimulovaný vstupní OR při nejmenší nasimulované síti



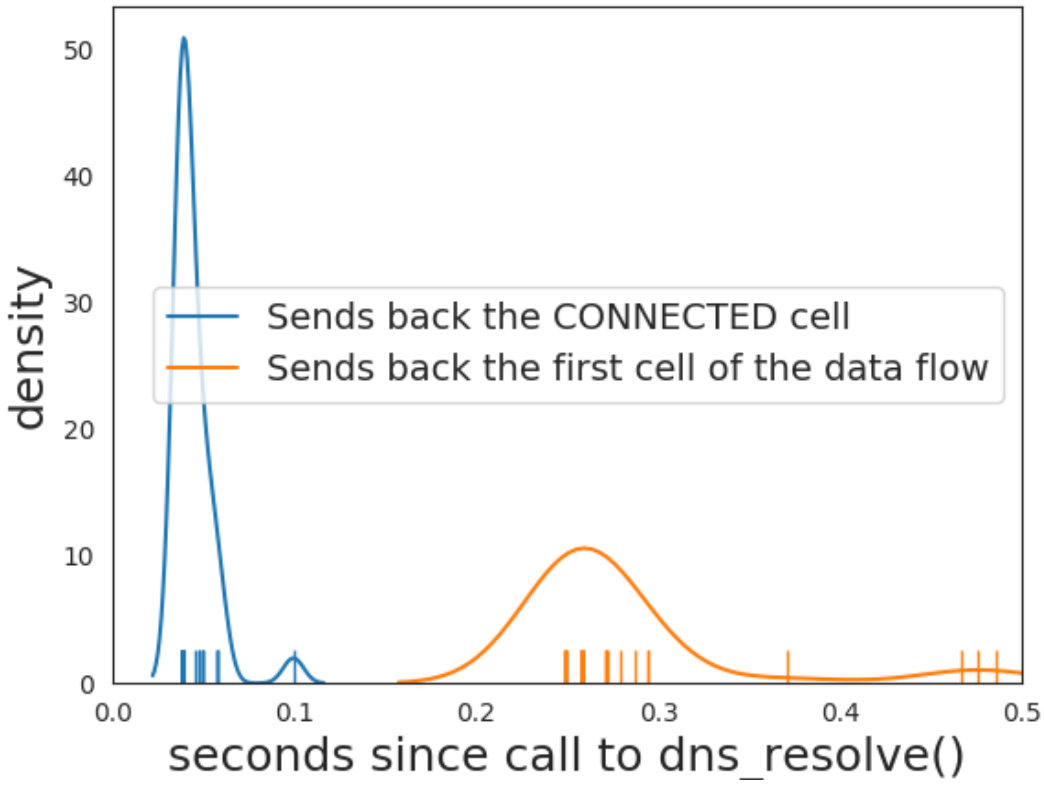
Obrázek 6 - Nasimulovaný výstupní OR při nejmenší nasimulované síti

Tato síť je malá a vytvořená pouze k praktické ukázce. Tor síť obsahuje mnohem více klientů a OR, ale tato ukázka nám ukazuje, že můžeme zjistit, pomocí rychlosti spojení nebo pomocí korelace, zda vstupní a výstupní OR tvoří obvod. Tímto je možné odhalit klienta, který se připojil přes vstupní a výstupní OR k HS.

8.3.2 Aktivní útok

Dropping on the Edge: Flexibility and Traffic Confirmation in Onion Routing Protocols [44]

Tento útok zasílá relay drop packety skrze obvod přes vstupní OR klienta, tyto packety nejsou logovány, proto jsou „neviditelné“ na „okraji“ (vstupní OR klienta, výstupní OR, RP, InP), pro zjednodušení označíme „okraj“ jako výstupní OR. Vždy, když se klient chce připojit k serveru, tak se pošle packet BEGIN, tím se zavolá dns_resolve() na výstupním OR, které přesměruje klienta k serveru. Když proběhne přesměrování dns_resolve() úspěšně výstupní OR pošle packet CONNECTED klientovi.



Obrázek 7- počet packetů CONNECTED poslaných klientovi a první packety toku dat. Tento obrázek byl vytvořen z vygenerované sítě. Tyto data se vyskytovaly na výstupním OR uvnitř nasimulované sítě Shadow. Čas než byl zavolán dns_resolve() je 0,5 sekundy.

Při každém zavolání dns_resolve() je potřeba uložit IP adresu serveru do logu a poslat 3 packety klientovi (ideálně packety, které nejsou logované). Tyto 3 packety budeme nazývat „Dropmark“. Po zaslání Dropmarku je možné tyto packety sledovat na vstupním OR u klienta a tím zjistit, že se klient připojil k serveru.

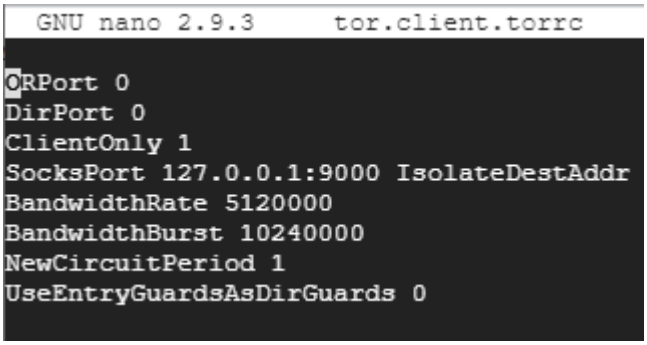
Pro nasimulování tohoto útoku byl použit speciálně upravený Tor, autorem tohoto útoku.[45] Uvnitř Toru je vytvořen tento útok. Při instalaci Tor plug-inu v Shadow je potřeba zadat cestu k tomuto Toru pomocí příkazu:

```
./setup build --tor-prefix /cesta/k/Toru
```

Pro vytvoření útoku vygenerujeme síť (viz. 8.2.2). Pro tento útok byla vygenerována síť s 3 DB routery, 50 OR, 100 web klienty a 50 servery.

Po vygenerování sítě je potřeba prvně nakonfigurovat síť, přidáním příkazů uvnitř adresáře conf na konec souboru:

tor.client.torrc – soubor vytvořený pro konfiguraci klienta uvnitř simulace

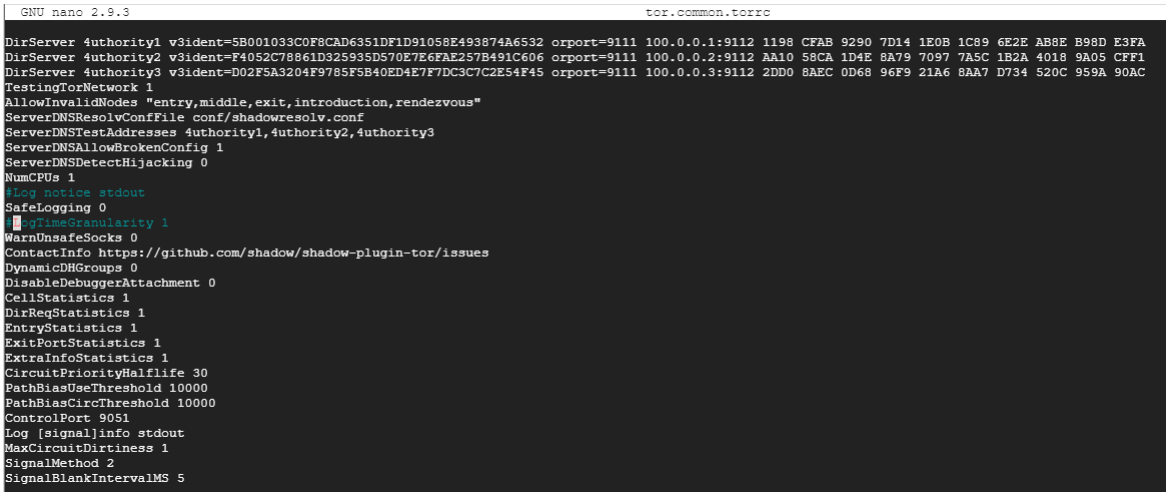


Obrázek 8 - konfigurace klienta

NewCircuitPeriod 1 – každou sekundu nastaví nový obvod, pokud to bude potřeba.

UseEntryGuardsAsDirGuards 1 – vstupní OR, je použit jako OR pro stahování informací o HS.

tor.common.torrc – soubor vytvořený pro konfiguraci simulace



Obrázek 9 - konfigurace simulace

Log [signal]info stdout – speciálně vytvořený log pro nalezení nebo nenalezení „vodoznaku“.

MaxCircuitDirtiness 1 – Může použít obvod znovu, pokud byl maximálně 1 sekundu starý.

SignalMethod 2 – signál vytvořený speciálně pro vytvoření nalezení/nenalezení „vodoznaku“.

SignalBlankIntervalMS 5 – speciálně vytvořený signál, který vytvoří pauzu 5 milisekund.

tor.guard.torrc – soubor vytvořený pro konfiguraci vstupního OR uvnitř simulace

```
GNU nano 2.9.3   tor.guard.torrc

ORPort 9111
SocksPort 0
DirPort 0
ExitPolicy "reject *:*"
ActivateSignalAttackListen 1
```

Obrázek 10 - konfigurace vstupního OR

ActivateSignalAttackListen 1 – speciálně vytvořený signál pro poslech útoku.

tor.middle.torrc – soubor vytvořený pro konfiguraci středního OR uvnitř simulace

```
GNU nano 2.9.3   tor.middle.torrc

ORPort 9111
SocksPort 0
DirPort 0
ExitPolicy "reject *:*
```

Obrázek 11 - konfigurace středního OR

tor.exit.torrc – soubor vytvořený pro konfiguraci výstupního OR uvnitř simulace

```
GNU nano 2.9.3    tor.exit.torrc

ORPort 9111
SocksPort 0
DirPort 0
ExitPolicy "accept *:80"
ExitPolicy "reject *:*"
```

Obrázek 12 - konfigurace výstupního OR

tor.exitguard.torrc – soubor vytvořený pro konfiguraci speciálního výstupního OR uvnitř simulace

```
GNU nano 2.9.3 tor.exitguard.torrc

ORPort 9111
SocksPort 0
DirPort 0
ExitPolicy "accept *:80"
ExitPolicy "reject *:*"
```

Obrázek 13 - konfigurace speciálního výstupního OR

Spuštění simulace pomocí příkazu `shadow -w 8 shadow.config.xml > shadow.log` trvá zhruba několik hodin. Nyní můžeme na Obrázku 14 zjistit, kolik se nám povedlo zachytit vodoznaků.

```
Cauwin@Ubuntu:~/shadow/shadow-plugin-tor/resource/DroppingOnTheEdge/sit$ grep "S
potted watermark" shadow.data/hosts/relayguard*/stdout-tor-1000.log | wc -l
3
Cauwin@Ubuntu:~/shadow/shadow-plugin-tor/resource/DroppingOnTheEdge/sit$ grep tr
ansfer-error shadow.data/hosts/webclient*/stdout-tgen-1002.log | wc -l
0
Cauwin@Ubuntu:~/shadow/shadow-plugin-tor/resource/DroppingOnTheEdge/sit$ grep tr
ansfer-complete shadow.data/hosts/webclient*/stdout-tgen-1002.log | wc -l
3296
Cauwin@Ubuntu:~/shadow/shadow-plugin-tor/resource/DroppingOnTheEdge/sit$ "No wat
ermark" shadow.data/hosts/relayguard*/stdout-tor-1000.log | wc -l
No watermark: command not found
0
Cauwin@Ubuntu:~/shadow/shadow-plugin-tor/resource/DroppingOnTheEdge/sit$ grep "N
o watermark" shadow.data/hosts/relayguard*/stdout-tor-1000.log | wc -l
Cauwin@Ubuntu:~/shadow/shadow-plugin-tor/resource/DroppingOnTheEdge/sit$ grep "N
o watermark" shadow.data/hosts/relayguard*/stdout-tor-1000.log | wc -l
3279
```

Obrázek 14 - síť nasimulovaná pouze pro odposlech

„Spotted watermark“ znázorňuje počet nalezených vodoznaků. „Transfer-error“ znázorňuje počet errorů. „Transfer-complete“ popisuje počet úspěšných záznamů. „No wattermark“ je počet nenalezených vodoznaků.

Jelikož jsme zatím jen poslouchali, tak jsme nemohli najít žádné vodoznaky. Na obrázku 14 jsme ale zachytili 3 vodoznaky, to znamená, že 3 vodoznaky jsou falešně úspěšné. 3279 vodoznaků jsme správně nenašli. K nalezení vodoznaků je potřeba zapnout zapsání útoku. Je potřeba přidat do souborů:

tor.exit.torrc a **tor.exitguard.torrc**

ActivateSignalAttackWrite 1 – speciální příkaz vytvořený k zapisování útoku.

```
 grep "No watermark" shadow.data/hosts/relayguard*/stdout-tor-1000.log | wc -l
0
Cauwin@Ubuntu:~/shadow/shadow-plugin-tor/resource/DroppingOnTheEdge/sit_aktivni$
 grep "Spotted watermark" shadow.data/hosts/relayguard*/stdout-tor-1000.log | wc
 -l
3254
```

Obrázek 15- síť nasimulovaná s aktivním útokem

Nyní si můžeme na Obrázku 15 všimnout, že počet nenalezených vodoznaků je 0 a počet nalezených vodoznaků je 3254, což je 100% úspěšnost, avšak musíme mít na mysli, že jsme vygenerovali malou síť pouze se 100 klienty, 50 OR a 50 servery. Ve větších sítích bude úspěšnost menší, autor odhaduje úspěšnost zhruba na 99.86% z toho falešně úspěšné na 0.03%. Nám vyšla falešná úspěšnost zhruba 0,091%.

9 Forenzní software

Jakmile známe IP adresu pachatele, je potřeba najít důkazy uvnitř jeho zařízení. Je zapotřebí prohledat jeho PC z důvodu nalezení stop. To je možné za pomocí vytvoření bitové kopie PC pomocí forenzního softwaru.

Je možné použít forenzní software na prohledání PC jako IEF, Axiom nebo Belkasoft všechny zmíněné softwary jsou certifikované, takže je může použít orgán činný v trestním řízení. Může vytvořit bitovou kopii a vyhledat uvnitř počítače důležité programy, které se pojí s Darknetem.

Uvnitř souboru torrc je napsaná cesta k lokálnímu serveru, na kterou HS ukazuje. Tento soubor také obsahuje důležité informace o konfiguraci HS. Pokud je soubor nalezen a vyskytuje se v něm cesta k nelegálnímu HS, nalezli jsme podezřelého.[31]
Implicitně:
1. V macOS se soubor torrc s informacemi o HS nachází v ~/Library/Application Support/TorBrowser-Data/Tor/torrc
2. V Linuxu ~/[path_to_tor_browser]/Browser/TorBrowser/Data/Tor/torrc
3. Ve Windows \Desktop\TorBrowser\Browser\TorBrowser\Data\Tor\torrc[32]
Pokud nebyl nalezen soubor torrc, je možné použít forenzní software, který může nalézt tento soubor smazaný nebo přejmenovaný. Zároveň je důležité nalézt soubor tor.log, který obsahuje informace o komunikaci klienta uvnitř Toru. (v Linuxu je implicitně nastavený: /usr/local/etc/tor/tor.log).

V případě, že byl spáchán TČ pachatelem, který nevlastní HS, je potřeba pachatele zastihnout se zapnutým počítačem, pokud se připojil k HS, bude možné tyto data zjistit v RAM paměti. Vypnout internet, aby nebylo možné smazat data na dálku a vytvořit bitovou kopii.

10 Metodiky řešení kybernetické kriminality pro použití orgánem činným v trestním řízení

Žádná metodika pro odsouzení pachatele u policie pro tento komplexní problém neexistuje, postup se může lišit případ od případu. Myslím si, že postup při spáchání TČ uvnitř Darknetu by mohl být následující:

1. Pří spáchání TČ je zapotřebí, aby trestný čin byl nalezen. Toho lze docílit nalezením TČ orgánem činným v trestním řízení. Toho je možné dosáhnout využitím Tor vyhledávačů nebo pomocí rychlých OR, které komunikují s DB, nebo výstupním routerem, který se připojuje k HS. Tyto informace nás odkážou na HS, které je poté možné prohlédnout a zjistit, zda se na něm nenachází TČ. Popřípadě nahlášením TČ poškozeným nebo oznamovatelem.
2. Při nalezení TČ uvnitř DARKNETu je potřeba definovat, kdo je poškozený a kdo pachatel, popřípadě zda se znají. Následně zjistit odhadovanou cenu podle toho, jak velká je odhadovaná škoda.
3. Určit, jak je možné dopadnout pachatele, který spáchal TČ. K dopadení pachatele, který spáchal svým konáním TČ:

3.1 sledováním trestné činnosti, sdílením extrémistického názoru, nelegálního obsahu. Je zapotřebí napsat žádost soudu o použití § 88 TŘ, který pojednává o odposlechu.

3.1.1 Pokud soud vyhoví žádosti o odposlechu, je možné začít odposlouchávat podezřelého. Odposlouchávat je možné, pokud vlastníme vstupní a výstupní OR. Díky vstupnímu OR zjistíme IP adresu pachatele a díky výstupnímu OR zjistíme, ke kterému HS se připojil. Také je možné použít fingerprint útok, kde budeme komunikovat s HS a zjistíme charakteristiku tohoto provozu, poté nám stačí vlastnit pouze vstupní OR. V tuto chvíli je potřeba:

3.1.1.1 Pasivní útok: Začít odposlouchávat vstupní OR a zároveň vlastnit a odposlouchávat výstupní OR vedoucí k HS tohoto obvodu. Pomocí metadat, které bychom získali, můžeme zjistit, zda se podezřelý připojil k HS (viz 8.3.1).

Začít komunikaci s HS a zjistit charakteristiku provozu. Odposlouchávat na vstupním OR, ze kterého zjistíme, jestli se charakteristika provozu shoduje. Pokud ano, je velká pravděpodobnost, že se klient připojil k odposlouchávanému HS.

3.1.1.2 Aktivní útok: Na vstupní OR použít např. nasimulovaný útok v této práci Dropping On The Edge: Flexibility and Traffic Confirmation in Onion Routing

protocols. Tento útok dokáže odhalit klienta, který se připojil k HS, že mu pošle na vstupní OR 3 packety, které budou vidět na výstupním OR. Z úspěšně nasimulovaného útoku víme, že tento útok odhalí klienta, co se připojil k HS v malé síti na 100% s 0,09% falešnou pozitivitou na velkou síť auto odhaduje zhruba 99,86% s 0,03% falešnou pozitivitou.

3.2 Pachatel vlastní HS s nelegálním obsahem. V tomto případě je potřeba zadokumentovat HS, např. pomocí torify wget --mirror xyz.onion. Aby pachatel nemohl smazat nebo změnit HS. Jakmile máme zadokumentovaný HS, můžeme:

3.2.1 Použít například útok Sniper attack, který zasílá velké množství packetů CREATE a tím zahltí HS OR, obvod si poté vybere nový OR. Tento útok se opakuje tak dlouho, dokud není úspěšný, a nový OR bude patřit orgánu činnému v trestním řízení. Potom může orgán činný v trestním řízení zjistit IP adresu HS, protože vlastní OR, který komunikuje přímo s HS.

3.2.2 Kontaktovat operátory Toru, kteří sami můžou zastavit tento HS pomocí vytvoření patche, který zakáže všechny žádosti na HS. Tímto zastavíme HS, ale s velkou pravděpodobností pachatel založí nový HS. Další nevýhoda je, že nedopadneme pachatele.

4. Jakmile je pachatel nalezen, je potřeba vytvořit právní posouzení trestné činnosti. Dále je potřeba zakázat připojení k HS na které se vyskytuje TČ, tohoto můžeme dosáhnout ovládnutím všech OR, které komunikují s HS a na všech zakážeme přijímat žádosti nebo kontaktováním operátorů Toru.

5. Poté je potřeba povolení soudu k provedení domovní prohlídky, podle § 83 odst. 1 TŘ je potřeba podat návrh státním zastupitelem. Pro prohlídku jiných prostor je potřeba použít § 83a odst. 1 TŘ, postup je stejný jako u domovní prohlídky, navíc podle § 83a odst. 2 TŘ je možné provést prohlídku, jestliže věc nesnese odkladu, avšak poté je potřeba si dodatečně zažádat o povolení k prohlídce státním zástupcem.

6. Pokud soud vyhoví této žádosti, je potřeba využít momentu překvapení, aby nedošlo k vyplašení pachatele, který by mohl zničit stopy a důkazy. Počkat, až se pachatel připojí k Toru, vypnout připojení k internetu a využít momentu překvapení, podezřelý nesmí vypnout PC, protože je potřeba bitová kopie RAM paměti, kde se nachází informace o komunikaci s HS. Zajistit výpočetní techniku a digitální data pachatele pomocí forenzního nástroje. Pokud vlastní HS, je potřeba zjistit, kde se HS nachází pomocí torrc souboru a zajistit ho. Pokud jsou zajištěný HS a HS, který jsme stáhli pomocí dokumentace totožné, s velkou pravděpodobností jsme dopadli pachatele.

7. Napsat otázky soudnímu znalci a posoudit v nich rizika, které mohou vzniknout. Zaslat mu bitovou kopii RAM paměti (popřípadě HS), které byly nalezeny na výpočetní technice u pachatele. Soudní znalec vytvoří znalecký posudek soudu.
8. Soudní spor s pachatelem.

Závěr

Vyšetřování trestné činnosti v DARKNETu by mělo být prováděno specializovanými týmy složenými z odborníků na problematiku. V této práci jsme si popsali trestné činy uvnitř Darknetu:

a) Trestné činy proti utajování, integritě a dostupnosti počítačových dat a systémů, konkrétně – hacking, prodej nelegálního SW a HW určeného k úmyslnému páchání trestných činů.
b) Trestné činy související s obsahem v podobě nelegální pornografie a šíření rasistických a xenofobních materiálů – forum, blogy atd.
c) Trestné činy související s porušováním autorských práv a práv prodejem nelegálního softwaru – obchod.

Všechny výše uvedené trestné činy lze vyšetřovat pomocí jedné z metodik uvedených v kapitole 9. Není nutné vytvářet speciální metodiky pro jednotlivé trestné činy. Co je však nutné řešit, je oprávněnost provádění útoků orgány v trestním řízení v prostředí DARKNETu, tak aby nashromážděná dokumentace mohla být využita v trestním řízení. Zároveň z toho důvodu, že Internet je celosvětová síť, je problematické určit, kde došlo k trestnému činu. Vzhledem k celosvětovosti Internetu je více než pravděpodobné, že k vyšetřování bude potřeba mezinárodní justiční spolupráce prostřednictvím Europolu nebo Interpolu.

Vzhledem k tomu, že vyhledávání a potírání trestné činnosti v rámci DARKNETu vyžaduje vysoce specializované odborníky, bylo by vhodné koordinovat tuto činnost i s jinými státními subjekty nejen z Ministerstva vnitra, ale i z např. Vojenského zpravodajství a útvarů Ministerstva obrany. Toto by však vyžadovalo úpravu legislativy České republiky.

Úpravu legislativy by také vyžadoval odposlech uvnitř DARKNETu (viz kapitola 4.2.2 a 7.1) je potřeba vytvořit spousty OR a následně odposlouchávat, kdo navštívil zakázané weby. Toto orgán činný v trestním řízení provést nemůže kvůli § 88 TŘ, nezískal by informace legálním způsobem, takže by je soud zamítl a musely by být zničeny. Stejně tak informace, které byly získané při odposlechu osoby, která se připojila do DARKNETu (viz kapitola 7.1.2) by byly zamítnuté soudem a musely by být zničeny. Z tohoto důvodu orgán činný v trestním řízení neřeší TČ uvnitř DARKNETu. Při změně legislativy by to možné bylo pomocí metodiky popsané v kapitole 10.

Seznam použitých zdrojů

[1] Darknet vs Dark Web vs Deep Web vs Surface Web — Different Parts Of The World Wide Web. *Https://techlog360.com/* [online]. Tamilnádu: Sabarinath, ©2019 [cit. 2019-04-05]. Dostupné z: https://techlog360.com/darknet-vs-dark-web-vs-deep-web-vs-surface-web/

[2] *OWEN, Gareth a Nick Sav. Global Commission on Internet Governance* [online]. 2015, **20**(8) [cit. 2019-04-05]. Dostupné z: https://www.cigionline.org/sites/default/files/no20_0.pdf

[3] MOORE, Daniel a Thomas RID. Cryptopolitik and the Darknet. *Survival: Global Politics and Strategy*[online]. 2016, **58**(1), 7-38 [cit. 2019-04-05]. DOI: 10.1080/00396338.2016.1142085. Dostupné z: https://doi.org/10.1080/00396338.2016.1142085

[4] BIRYUKOV, Alex, Ivan PUSTOGAROV, Fabrice THILL a Ralph-Philipp WEINMANN. *Content and popularity analysis of Tor hidden services* [online]. Washington, DC: ICDCS, 2014 [cit. 2019-04-05]. ISBN 978-1-4799-4181-0. Dostupné z: https://arxiv.org/pdf/1308.6768.pdf

[5] AVARIKIOTI, Georgia a kol. ZINDROS3. Structure and Content of the Visible Darknet. In: *Https://arxiv.org* [online]. Mountain View, 2018 [cit. 2019-04-05]. Dostupné z: https://arxiv.org/pdf/1811.01348.pdf

[6] Classifying Illegal Activities on Tor Network Based on Web Textual Contents. *EACL* [online]. 2017, **17**(1004), 35-43 [cit. 2019-04-05]. Dostupné z: https://www.aclweb.org/anthology/E17-1004

[7] Onion Services: Step 6. In: *Torproject* [online]. Seattle: The Tor Project, 2019 [cit. 2019-04-05]. Dostupné z: https://2019.www.torproject.org/docs/onion-services

[8] Onion Routing - Computerphile. *Youtube* [online]. San Bruno: YouTube, 2017 [cit. 2019-04-05]. Dostupné z: https://www.youtube.com/watch?v=QRYzre4bf7I

[9] Tor: Onion Service Protocol. *Torproject* [online]. Seattle: The Tor Project, 2019 [cit. 2019-04-05]. Dostupné z: https://2019.www.torproject.org/docs/onion-services.html.en

[10] Dingledine, Roger & Mathewson, Nick & Syverson, Paul. (2004). Tor: The Second-Generation Onion Router. Paul Syverson. 13. Dostupné z: https://www.onion-router.net/Publications/tor-design.pdf

[11] What are the Pros and Cons of Using Tor Browser?... *Deepweb-sites* [online]. ©2019 [cit. 2019-04-05]. Dostupné z: https://www.deepweb-sites.com/pros-and-cons-of-using-tor-browser/

[12] JANSEN, Rob, Florian TSCHORSCH, Aaron JOHNSON a Bjorn SCHEUERMANN. The Sniper Attack: Anonymously Deanonymizing and Disabling the Tor Network. *NDSS Symposium* [online]. 2014 [cit. 2019-04-05].

DOI: 10.14722/ndss.2014.23288. Dostupné z: https://www.robgjansen.com/publications/sniper-ndss2014.pdf
[13] Tor security advisory: "relay early" traffic confirmation attack. *Torproject* [online]. Seattle: The Tor Project, 2014 [cit. 2019-04-05]. Dostupné z: https://blog.torproject.org/tor-security-advisory-relay-early-traffic-confirmation-attack
[14] RYAN Pries, Wei Yu, Xinwen Fu a Wei Zhao. A New Replay Attack Against Anonymous Communication Networks. *IEEE ICC* [online]. 2008 [cit. 2019-04-05]. DOI: 10.1109/ICC.2008.305. ISSN 1938-1883. Dostupné z: http://www.cs.ucf.edu/~xinwenfu/paper/ICC08_Fu.pdf
[15] LING, Zhen a kol. A new cell counter based attack against tor. *ACM CCS* [online]. 2009 [cit. 2019-04-05]. DOI: 10.1145/1653662.1653732. Dostupné z: http://web.cse.ohio-state.edu/~xuan.3/papers/09_ccs_llyfxj.pdf
[16] YE, Zhu a kol. Correlation-Based Traffic Analysis Attacks on Anonymity Networks. *IEEE Transactions on Parallel and Distributed Systems* [online]. 2010, **21**(7), 954 - 967 [cit. 2019-04-05]. DOI: 10.1109/TPDS.2009.146. Dostupné z: https://engagedscholarship.csuohio.edu/cgi/viewcontent.cgi?article=1053&context=enece_facpub
[17] BAUER, Kevin a kol. *Low-resource routing attacks against tor* [online]. Virginia: WPES, 2007 [cit. 2019-04-05]. ISBN 978-1-59593-883-1. Dostupné z: https://cs.gmu.edu/~mccoy/papers/wpes25-bauer.pdf
[18] A potential HTTP-based application-level attack against Tor. *Future Generation Computer Systems*[online]. 2011, **27**(1), 67-77 [cit. 2019-04-05]. DOI: 10.1016/j.future.2010.04.007. Dostupné z: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.710.6952&rep=rep1&type=pdf
[19] *YIXIN, Sun a kol. RAPTOR: Routing Attacks on Privacy in Tor* [online]. Washington, D.C.: USENIX Security, 2015 [cit. 2019-04-05]. ISBN 978-1-931971-232. Dostupné z: https://www.usenix.org/system/files/conference/usenixsecurity15/sec15-paper-sun.pdf
[20] Website fingerprinting in onion routing based anonymization networks. In: PANCHENKO, Andriy, Lukas NIESSEN, Andreas ZINNEN a Thomas ENGEL. *WPES* [online]. Chicago: ACM, ©2011, s. 103-114 [cit. 2019-04-05]. DOI: 10.1145/2046556.2046570. ISBN 978-1-4503-1002-4. Dostupné z: https://www.freehaven.net/anonbib/cache/wpes11-panchenko.pdf
[21] KWON, Albert a kol. *Circuit Fingerprinting Attacks: Passive Deanonymization of Tor Hidden Services* [online]. Washington, D.C.: USENIX Security, 2015 [cit. 2019-04-05]. ISBN 978-1-931971-232. Dostupné z: https://www.usenix.org/system/files/conference/usenixsecurity15/sec15-paper-kwon.pdf

[22] MITTAL, Prateek a kol. Stealthy Traffic Analysis of Low-Latency Anonymous Communication Using Throughput Fingerprinting. In: *CCS* [online]. Chicago: ACM, 2011, s. 215-226 [cit. 2019-04-05]. DOI: 10.1145/2046707.2046732. ISBN 978-1-4503-0948-6. Dostupné z: https://arxiv.org/pdf/1109.0597.pdf
[23] BARBERA, Marco, Vasileios KEMERLIS, Vasilis PAPPAS a Angelos KEROMYTIS. CellFlood: Attacking Tor Onion Routers on the Cheap. In: *ESORICS* [online]. Springer-Verlag Berlin Heidelberg, 2013, s. 664-681 [cit. 2019-04-05]. DOI: https://doi.org/10.1007/978-3-642-40203-6_37. ISBN 978-3-642-40203-6. Dostupné z: http://wwwusers.di.uniroma1.it/~barbera/papers/barbera-esorics13.pdf
[24] Spying in the dark: TCP and tor traffic analysis. In: GILAD, Yossi a Amir HERZBERG. *PETS* [online]. Vigo: Department of Computer Science, 2012, s. 100-119 [cit. 2019-04-05]. DOI: 10.1007/978-3-642-31680-7_6. ISBN 978-3-642-31679-1. Dostupné z: https://www.freehaven.net/anonbib/cache/tcp-tor-pets12.pdf
[25] CHAKRAVARTY, Sambuddho, Angelos STAVROU a Angelos KEROMYTIS. Traffic analysis against low-latency anonymity networks using available bandwidth estimation. In: *ESORICS* [online]. Řecko: Computer Security Research, 2010, s. 249-267 [cit. 2019-04-05]. ISBN 3-642-15496-4. Dostupné z: https://www.cs.columbia.edu/~angelos/Papers/2010/esorics.pdf
[26] MURDOCH, Steven a George DANEZIS. *Low-Cost Traffic Analysis of Tor* [online]. Oakland: IEEE, 2005 [cit. 2019-04-05]. ISBN 0-7695-2339-0. Dostupné z: https://www.cs.ucy.ac.cy/courses/EPL682/papers/anon-2.pdf
[27] EVANS, Nathan, Roger DINGLEDINE a Christian GROTHOFF. A Practical Congestion Attack on Tor Using Long Paths. *USENIX* [online]. 2009, **09**, 33-50 [cit. 2019-04-05]. Dostupné z: https://www.freehaven.net/anonbib/cache/congestion-longpaths.pdf
[28] The Tor Relay Guide. *Torproject* [online]. Seattle: The Tor Project, 2018 [cit. 2019-04-05]. Dostupné z: https://trac.torproject.org/projects/tor/wiki/TorRelayGuide
[29] MALLET. *Http://mallet.cs.umass.edu/* [online]. ©2018 [cit. 2019-04-05]. Dostupné z: http://mallet.cs.umass.edu/download.php
[30] UClassify. *UClassify* [online]. Stockholm, 2008 [cit. 2019-04-05]. Dostupné z: https://www.uclassify.com/browse
[31] Configuring Onion Services for Tor. *Torproject* [online]. Seattle: The Tor Project, 2019 [cit. 2019-04-05]. Dostupné z: https://2019.www.torproject.org/docs/tor-onion-service

[32] Connecting to an authenticated Onion service. *GitHub* [online]. San Francisco, 2007 [cit. 2019-04-05]. Dostupné z: https://github.com/AnarchoTechNYC/meta/wiki/Connecting-to-an-authenticated-Onion-service

[33] What is Surface Web and how is it different from Dark Web?. In: *Cyware* [online]. New York, 2019 [cit. 2019-04-05]. Dostupné z: https://cyware.com/educational-guides/cyber-threat-intelligence/how-is-surface-web-intelligence-different-from-dark-web-intelligence-393c

[34] Návrh připojení PČR do DARKNETu. In: *Xmodulo* [online]. USA: creative commons [cit. 2019-04-05]. Dostupné z: http://xmodulo.com/access-linux-server-behind-nat-reverse-ssh-tunnel.html

[35] KOLOUCH, Jan. *Cybercrime*. Praha: CZ.NIC, 2016, s. 85-133. ISBN 978-80-88168-15-7.

[36] BARBERA, Marco V., Vasileios P. KEMERLIS, Vasilis PAPPAS a Angelos D. KEROMYTIS. *CellFlood: Attacking Tor Onion Routers on the Cheap* [online]. Sapienza University, Rome, Italy, 2013 [cit. 2020-05-18]. Dostupné z: https://cs.brown.edu/~vpk/papers/cellflood.esorics13.pdf. Sapienza University, Rome, Italy.

[37] BLOND, Stevens Le a kol. *One Bad Apple Spoils the Bunch:* [online]. 2011, 8 [cit. 2020-05-18]. Dostupné z: https://arxiv.org/pdf/1103.1518.pdf

[38] Azure Microsoft. *Microsoft Azure* [online]. Redmond, Washington, USA: Microsoft, 2016 [cit. 2020-05-18]. Dostupné z: https://azure.microsoft.com/cs-cz/

[39] Cloud Google. *Google Cloud* [online]. Mountain View, Kalifornie, USA: Google, 2008 [cit. 2020-05-18]. Dostupné z: https://cloud.google.com/

[40] JANSEN, Rob. Shadow. *Shadow* [online]. 2011 [cit. 2020-05-18]. Dostupné z: https://github.com/shadow/shadow

[41] JANSEN, Rob. Shadow plugin tor. *Shadow-plugin-tor* [online]. 2014 [cit. 2020-05-18]. Dostupné z: https://github.com/shadow/shadow-plugin-tor

[42] JANSEN, Rob. Traffic generator. *Tgen* [online]. 2019 [cit. 2020-05-18]. Dostupné z: https://github.com/shadow/tgen

[43] MATHEWSON, Nick. Chutney. *Chutney* [online]. 2011 [cit. 2020-05-18]. Dostupné z: https://github.com/torproject/chutney

[44] ROCHET, Florentin a Olivier PEREIRA. Dropping on the Edge: Flexibility and Traffic Confirmation in Onion Routing Protocols. *Proceedings on Privacy Enhancing Technologies* [online]. UCLouvain, Louvain-la-Neuve, Belgium, 2018, **2018**(B-1348), 20 [cit. 2020-05-18]. Dostupné z: https://content.sciendo.com/view/journals/popets/2018/2/article-p27.xml

[45] Dropping on the Edge: Flexibility and Traffic Confirmation in Onion Routing Protocols. *Dropping on the Edge* [online]. Louvain-la-Neuve, Belgium, 2017 [cit. 2020-05-18]. Dostupné z: https://github.com/frochet/dropping_on_the_edge/tree/master/dropmark/tor

[46] ROCHE, Xavier. Httrack. *Httrack* [online]. 2017 [cit. 2020-05-18]. Dostupné z: https://www.httrack.com/

[47] CHROBOCZEK, Juliusz. Polipo. *Polipo* [online]. 2014 [cit. 2020-05-18]. Dostupné z: https://www.irif.fr/~jch/software/polipo/manual/Web-interface.html

[48] Alexa servers. *Alexa servers* [online]. 2020 [cit. 2020-05-18]. Dostupné z: http://s3.amazonaws.com/alexa-static/top-1m.csv.zip

[49] Consensus. *Consensus* [online]. 2020 [cit. 2020-05-18]. Dostupné z: https://collector.torproject.org/archive/relay-descriptors/consensuses/consensuses-2020-03.tar.xz

[50] Extra infos. *Extra infos* [online]. 2020 [cit. 2020-05-18]. Dostupné z: https://collector.torproject.org/archive/relay-descriptors/extra-infos/extra-infos-2020-03.tar.xz

[51] Relay descriptors. *Relay descriptors* [online]. 2020 [cit. 2020-05-18]. Dostupné z: https://collector.torproject.org/archive/relay-descriptors/server-descriptors/server-descriptors-2020-03.tar.xz

Seznam obrázků

Seznam grafů