Útoky na systémy pro zákonné odposlechy

FIT VUT Technický report

Libor Polčák, Radek Hranický

Technický report č. FIT-TR-2012-008
Fakulta informačních technologií, Vysoké učení technické v Brně

Last modified: 30. listopadu 2012

Útoky na systémy pro zákonné odposlechy

Libor Polčák, Radek Hranický

Vysoké učení technické v Brně, email:
ipolcak@fit.vutbr.cz,xhrani00@stud.fit.vutbr.cz

Abstrakt Tato technická zpráva popisuje útoky na systémy pro sběr dat pro zákonné odposlechy, možnosti skrývání dat v síti a možná protiopatření. V práci práce jsou představeny normy s standardy pro zákonné odposlechy platné v Evropské unii i Spojených státech amerických a referenční architektura představená společností Cisco. Na základě těchto norem je podán přehled možných útoků, kterými je možné systém pro zákonné odposlechy obelstít a znesnadnit tak analýzu nasbíraných dat. Kromě metod útoků jsou představena i možná protiopatření a je diskutována jejich vhodnost pro účely sběru dat pro zákonné odposlechy. V rámci práce je představen nástroj pro skrytí komunikace a nástroj pro detekci tohoto útoku. Demonstrací běhu tohoto programu je ukázáno podvržení zprávy při komunikaci využívající veřejnou komunikační síť IRC.

1 Úvod

Se zneužitím počítačových sítí pro kriminální účely se setkáváme již od 80. let 20. století [88]. S narůstajícím objemem dat dostupných v rámci počítačových sítí roste motivace pro růst kriminality v počítačovém prostředí. Organizované kriminální skupiny navíc využívají stále se vylepšující možnosti komunikace v reálném čase pomocí počítačové sítě místo dříve používaných způsobů dorozumívání jako je např. telefon.

Zákonné odposlechy umožňují vyšetřovatelům závažné kriminální činnosti sbírat důkazní materiál v rámci počítačových sítí. V Evropské unii byly zákonné odposlechy schváleny Radou Evropské unie [6] a jsou standardizovány Evropským ústavem pro telekomunikační normy (European Telecommunications Standards Institute – ETSI). Při získávání dat přenášených v počítačových sítích spolupracují vyšetřující orgány s provozovateli počítačových síti (Network Operator/Access Provider – NWO/AP), případně poskytovateli služeb (Service Provider – SvP. Pro vlastní sběr dat je v rámci počítačové sítě NWO/AP/SvP instalován systém pro zákonné odposlechy (Lawful Interception System – LIS). LIS sleduje dění v sítí a přijímá požadavky na realizaci zákonných odposlechů povolených soudem. Nasbíraná data jsou předávaná k další analýze vyšetřujícím orgánům.

Povaha decentralizovaného síťového prostřední založeného nad Internetovým protokolem verze 4 (IPv4) [54], či 6 (IPv6) [29] naneštěstí dovoluje využití technik, které umožňují obelstít systém pro sběr dat v rámci zákonných odposlechů.

Pomocí šifrování komunikace je možné utajit obsah přenášených dat. V případě využití anonymizačních sítí (např. Tor [4], I2P [1]) je možné skrýt i identitu komunikujících stran. Další techniky umožňují ukrýt závadnou komunikaci tak, aby odposlouchávající viděl nezávadnou komunikaci. Tyto techniky mohou být zneužity pro provádění trestné činnosti [69,19,22] a vytvářejí tak prostor pro obelstění systémů pro zákonné odposlechy a následně i vyšetřovatele.

Cílem této technické zprávy je podat přehled technik, které je možné využít pro oklamání systému pro sběr dat pro zákonné odposlechy. Druhým cílem zprávy je přinést návrhy metod, kterými by bylo možné obelstění procesu zachytávání dat zabránit, nebo jej alespoň detekovat.

Pro demonstraci použitelnosti popisovaných metod byl v rámci projektu *Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace* [53] vytvořen nástroj LIS Deception Proxy (LDP), který dokáže oklamat systém pro zákonné odposlechy. Kromě demonstrace možného útoku slouží aplikace ke studiu možností, jak útok eliminovat. Pomocí LDP může být prováděno testování a případné zdokonalování současných systémů pro zákonné odposlechy. Vzhledem k implementaci překladu IPv6 adres lze uvažovat i více odesílatelů zpráv ve vnitřní síti. Jako příklad programu, který dokáže v zachycených datech odhalit manipulaci s daty, která může vést až k ukrytí přenášených dat programem LDP byl v rámci projektu *Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace* [53] vyvinut nástroj LIS Noise Cleaner (LNC).

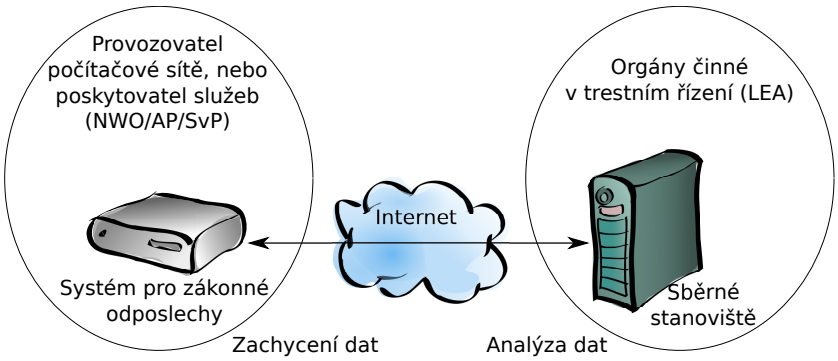
Tato technická zpráva má následující členění. Sekce 2 popisuje současné standardy pro specifikaci činností systémů pro sběr dat pro zákonné účely. Sekce 3 poskytuje přehled technik pro ukrytí dat v počítačových sítích. V sekci 4 jsou představeny současné možnosti pro vyhledávání neobvyklého síťového provozu a je diskutována vhodnost těchto metod pro účely zákonných odposlechů. Aplikace demonstrující jeden z popisovaných útoků je popsána v sekci 5 společně s nástrojem pro detekci tohoto útoku. Závěry technické zprávy jsou shrnuty v sekci 7.

2 Architektura systémů pro zákonné odposlechy

Tato kapitola se zabývá popisem architektur používaných pro tvorbu systémů pro zákonné odposlechy (LIS). Na popis referenční architektury vytvořené v ETSI navazují popisy architektur LIS používané v ostatních částech světa. Na závěr kapitoly jsou představeny architektury používané v současných LIS. Všechny zde popisované architektury LIS předpokládají umístění LIS na straně NWO/AP/SvP. Zachycená data jsou odeslána orgánům činným v trestním řízení (Law Enforcement Agency – LEA), kde jsou dále analyzována vyšetřovateli (obrázek 1).

2.1 Architektura Evropského ústavu pro telekomunikační normy

Pro Evropskou unii standardizoval zákonné odposlechy (Lawful Interception – LI) úřad ETSI. Každý odposlech musí být schválen soudním příkazem, přičemž může být požadováno buď odposlouchávání pouze metainformací vztahujících



Obrázek 1. Obecný model LIS: data jsou zachycena NWO/AP/SvP a zaslána LEA pro další analýzu

se ke komunikacím odposlouchávaného uživatele (Intercept Related Information – IRI), nebo může být povoleno i získávání kompletního síťového provozu odposlouchávaného subjektu (Content of Communication – CC).

V případě odposlechů pouze zpráv IRI získává LEA informace o aktuálním chování odposlouchávaného pomocí čtyř druhů zpráv.

1. Informace o připojení uživatele k síti, vygenerování, či přiřazení nové IP adresy, zahájení nové komunikace apod. jsou signalizovány pomocí zpráv IRI typu *begin*.
2. Informace o odpojení uživatele k síti, ukončení používání IP adresy, ukončení komunikace apod. jsou signalizovány pomocí zpráv IRI typu *end*. Zprávu typu *end* musí vždy předcházet odpovídající zpráva typu *begin*.
3. Potvrzení o probíhajícím připojení uživatele k síti, používaní konkrétní IP adresy odposlouchávaným, probíhající komunikaci apod. jsou signalizovány pomocí zpráv IRI typu *continue*. Zprávu typu *continue* musí vždy předcházet odpovídající zpráva typu *begin*. Zpráva typu *continue* může být využita například pro zpřesnění dříve signalizované informace, např. pokud dojde k prodloužení přidělení IP adresy pomocí mechanismu DHCP.
4. Ostatní zprávy, jako je například signalizace chybových stavů, pokusy uživatele o přihlášení se k síti apod. jsou signalizovány zprávami IRI typu *report*. Tato zpráva může být využita např. pokud se odposlouchávaný pokusí získat novou IP adresu mechanismem DHCP, ale ještě není možné vytvořit zprávu IRI typu *begin*, protože není známo, jaké IP adresy budou uživateli pomocí DHCP nabídnuty, tím spíše není známo jakou IP adresu si odposlouchávaný zvolí.

V případě povolení pro odposlech veškeré komunikace je vyšetřující LEA poskytována kopie všech dat přenášených uživatelem např. ve formě souboru typu PCAP [2]. V takovém případě jsou zachytávány veškeré pakety určené

odposlouchávanému, nebo odeslané odposlouchávaným včetně aplikačních dat a hlaviček protokolů nižších vrstev modelu ISO/OSI [55]. Odposlouchávaná data jsou zasílaná ve formě zpráv CC.

Identifikace odposlouchávaného musí být jednoznačná a neměly by být příliš velké obtíže s určením, zda má být konkrétní síťový tok předmětem odposlechu [39]. Norma nespecifikuje, které identifikátory musí být striktně podporovány, ale navrhuje některé z následujících:

- Uživatelské jméno nebo Network Access Identifier (NAI) [7]. NAI se používá při autentizaci pro přístup k síti.
- IP adresa (IPv4, IPv6)
- MAC adresa
- Identifikace uživatelovi přípojky nebo kabelového modemu
- Další identifikátory, na kterých se NWO/AP/SvP a LEA dohodnou

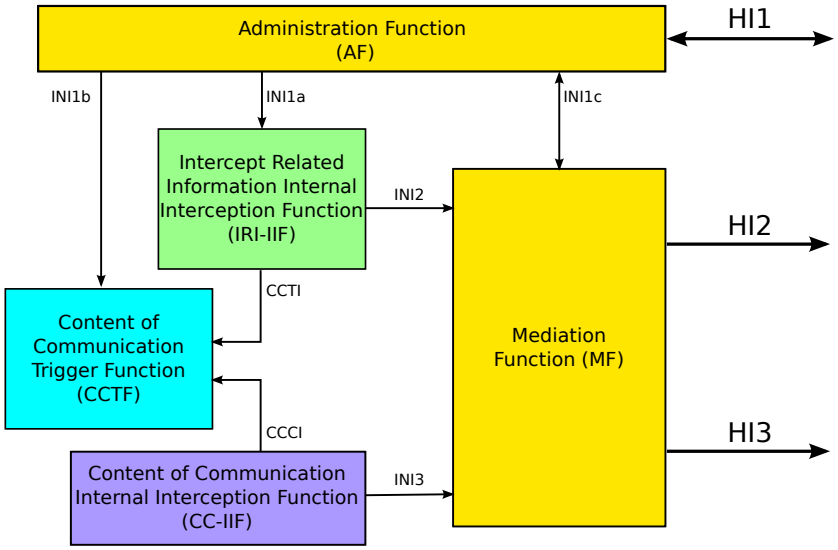
Ze strany LEA může mimo konkrétního síťového uživatele přijít požadavek na odposlech osoby identifikované jménem a dalšími údaji, které uživatele jednoznačně identifikují (např. adresou). V takovém případě bude úkolem pověřeného pracovníka tuto identitu převést na takovou, kterou je možné použít ve zbytku systému. Pověřený pracovník typicky využije interní databáze obsahující seznam zákazníků.

ETSI také vytvořil [37] referenční model architektury LIS zachycený na obrázku 2. ETSI definoval rozhraní mezi NWO/AP/SvP a vyšetřující LEA (Handover Interface – HI) [35,38,40,41,36], které je tvořeno:

- rozhraním HI1, kterým LEA zadávají a odebírají povolené odposlechy. LEA je rozhraním HI1 informována o zahájení a ukončení odposlechů zadaných touto LEA a jakýchkoliv problémech (např. technického rázu) týkajících se odposlechů prováděných pro tuto LEA.
- rozhraním HI2, které slouží pro přenos zpráv IRI do monitorovacího střediska LEA (Law Enforcement Monitoring Facility – LEMF).
- rozhraním HI3, které slouží pro přenos zpráv CC do LEMF.

Referenční model LIS vytvořený ETSI se skládá z pěti spolupracujících částí (bloků): *Administration Function* (AF), *Internal Interception Function Internal Interception Function* (IRI-IIF), *Content of Communication Trigger Function* (CCTF), Content of Communication Internal Interception Function (CC-IIF) a *Mediation Function* (MF). Bloky jsou propojeny rozhraními pojmenovanými jako *Internal Network Interface* (INI), *Content of Cummunication Trigger Interface* (CCTI) a *Content of Communication Control Interface* (CCCI) tak, jak je naznačeno na obrázku 2. V dalším textu rozebereme činnost bloků LIS a obsah dat posílaných jednotlivými rozhraními.

Vstupní požadavky k odposlechu jsou přijímány skrze rozhraní HI1 a zpracovávány blokem AF. Blok AF nejdříve provádí kontrolu správné specifikace odposlechu a jeho povolení soudem. Normy doporučují, aby oprávnění k odposlechu prováděl pověřený zaměstnanec NWO/AP/SvP. Pokud je vše v pořádku,



Obrázek 2. Referenční model LIS publikovaný ETSI [37]

je odposlech zařazen do fronty čekajících odposlechů. Blok AF je následně zodpovědný za korektní inicializaci a ukončení odposlechu, tj. konfiguraci ostatních částí systému (skrze rozhraní INI1a, INI1b a INI1c) tak, aby bylo zajištěno, že budou zachycena všechna data přenášená v povoleném intervalu pro odposlech a zároveň nebudou zaznamenána žádná data mimo platnost odposlechu.

Identifikace jednotlivých uživatelů sítě (např. jejich IP adresa) se obecně může v průběhu času měnit. Blok IRI-IIF detekuje v síťovém provozu zprávy, které se vztahují ke změně identity uživatelů. Každá změna identity odposlouchávaného cíle je pak neprodleně signalizována ostatním částem LIS rozhraním CCTI. Blok IRI-IIF dále vytváří zprávy IRI a odesílá je skrze rozhraní INI2.

Rozhraním INI1b přijímá blok CCTF statickou konfiguraci odposlechů typu CC a rozhraním CCTI přijímá dynamicky se měnící konfiguraci odposlechů typu CC. Úkolem bloku CCTF je konfigurace bloku CC-IIF. Protože může být blok CC-IIF tvořen sadou sond, udržuje si blok CCTF tabulku rozmístění jednotlivých sond a podle potřeby vytváří blok CCTF specifickou konfiguraci pro různé sondy.

Blok CC-IIF sleduje síťový provoz a kopíruje veškerý obsah komunikace vztahující se k odposlouchávanému. Vstupní požadavky na započetí, či ukončení odposlechu jsou zasílány rozhraním CCCI. Zachycená data jsou odesílána rozhraním INI3.

Blok MF může korelovat data odesílaná rozhraními HI2 a HI3, překódovávat zprávy IRI a CC do formátu, kterému rozumí konkrétní LEMF apod. V nejjednodušším případě blok MF pouze přeposílá zprávy IRI z rozhraní INI2 rozhraním HI2 a CC data z rozhraní INI3 rozhraním HI3.

2.2 Standard J-STD-025

Standard J-STD-025 [8] je obdobou referenčního modelu ETSI a je používaný převážně v USA. Standard J-STD-025 specifikuje následující tři druhy rozhraní [52]:

- *Surveillance Administration System* (SAS) zpřístupňuje systém vyšetřujícím orgánům.
- *Call Data Channel* (CDC) poskytuje signalizační zprávy, které obsahují informace spojené s navazováním a ukončováním komunikace.
- *Call Content Channel* (CCC) zprostředkovává úplnou kopii obsahu komunikace.

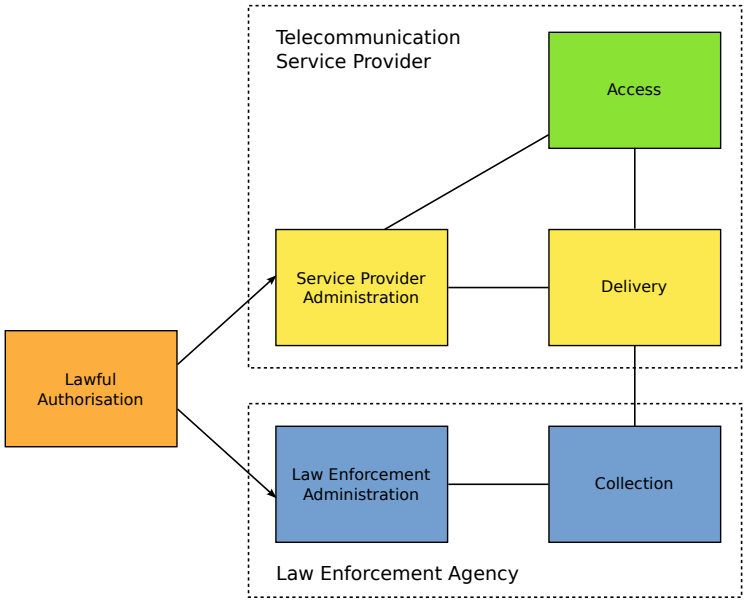
S drobnými odchylkami můžeme říct, že SAS je obdobou HI1 rozhraní, CDC obdobou HI2 rozhraní a CCC obdobou HI3 rozhraní. Komunikace mezi vyšetřovateli a subjektem provádějícím odposlech je tedy velmi podobná.

LIS spolupracuje se sběrným centrem na straně LEA a celá architektura se skládá z následujících částí (viz obrázek 3):

- *Lawful Authorisation* má na starosti schvalování odposlechů soudní cestou. Pokud je odposlech povolen, je informován příslušný poskytovatel telekomunikačních služeb.
- *Telecommunication Service Provider* je poskytovatel telekomunikačních služeb, který bude konkrétní odposlech provádět. LIS instalovaný v jeho síti se skládá z následujících částí:
  - *Service Provider Administration* provádí správu odposlechů a konfiguraci jednotlivých zařízení.
  - *Access* poskytuje pasivním systémům určených pro odposlech dat přístup k síťovým prvkům přenášejících provoz uživatelů využívajících síť operátora.
  - *Delivery* zodpovídá za doručení nasbíraných dat ve správním formátu příslušným LEA.
- *Law Enforcement Agency* je zodpovědná za sběr odchycených dat od poskytovatelů telekomunikačních služeb. V síti LEA jsou umístěny dvě části sběrného centra:
  - *Law Enforcement Administration* provádí správu odposlechů a konfiguraci zařízení pro sběr odposlechnutých dat.
  - *Collection* je zodpovědná za sběr signalizačních zpráv i kopie obsahu komunikace.

2.3 Architektura komerčních systémů pro zákonné odposlechy

Společnost Cisco vycházela z architektur LIS publikovaných ETSI a v rámci J-STD-025 a navrhlo vlastní architekturu [10]. Rozhraní LIS směrem k LEA je velmi podobné referenční architektuře ETSI, je však konfigurovatelné tak, aby mohlo být kompatibilní i s J-STD-025. Architektura LIS společnosti Cisco se skládá z následujících částí (viz obrázek 4):

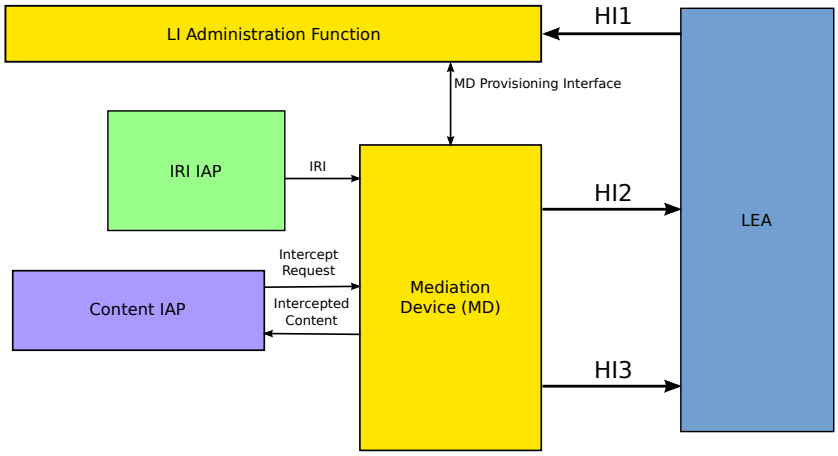


Obrázek 3. Architektura LIS a spolupráce s LEA specifikovaná standardem J-STD-025 [8]

- *LI Administration Function* je zodpovědná za zpracování odposlechů.
- *Intercept Access Point* (IAP) je zařízení, které je napojeno na síťovou infrastrukturu NWO/AP/SvP a je schopné získávat odposlouchávaná data. Rozlišujeme dva druhy IAP:
  1. *Content IAP* získává provoz na síťové vrstvě ISO/OSI modelu [55].
  2. *IRI IAP* je zodpovědná za získávání zpráv IRI (metainformací o provozu odposlouchávaného).
- *Mediation Device* (MD) konfiguruje IAP, replikuje data v případě odposlouchávání jednoho subjektu více LEA, odposlechnutá data převádí do formátu očekávaném LEA a zasílá rozhraními HI2 a HI3.

Společnost Cisco také na trh dodává některé síťové prvky, které dokáží poskytovat data pro LI, založené na výše uvedené architektuře. Obsah zpráv přenášených vnitřními rozhraními LIS (*MD Provisioning Interface*, *IRI*, *Intercept Request* a *Intercepted Content*) však není volně dostupný.

Francouzská společnost Aqsacom dodává na trh LIS pojmenovaný *Aqsacom real time Lawful Interception System* (ALIS) a své řešení LI částečně popsala a zveřejnila [9]. Publikovaný *White Paper* popisuje z větší části obecně LI. Částečně se dotýká amerických odposlechů, ale protože jde o francouzský produkt, nejvíce se zabývá modelem ETSI. *White Paper* popisuje některé problémy LI, mimo jiné upozorňuje na případy, kdy jsou omylem zachytávána data uživatelů, na které se odposlech nevztahuje a nejednoznačnost zákonů v některých



Obrázek 4. Architektura LIS publikovaná společností Cisco [10]

zemích. Dokument je však již částečně zastaralý, protože nebere v úvahu bezstavové přidělování adres protokolem IPv6 [74] a předpokládá adresy přidělované administrativně.

Společnost IP Fabrics poskytuje komplexní řešení pro LIS a systémy pro preventivní uchovávání provozních a lokalizačních údajů o elektronické komunikaci (Data Retention – DR). IP Fabrics dodává sondy pro LIS pojmenované *Deep Probe* [56], které jsou stavěné na zpracování 1 Gb/s nebo 10 Gbps provozu. Mimo sond, které mezi sebou navzájem nespolupracují dodává IP Fabrics mediační zařízení, které zajišťuje komunikaci s LEA.

2.4 Shrnutí architektur systémů pro zákonné odposlechy

V současné době ve světě existují dva významné standardy pro LI: referenční architektura publikovaná ETSI a americký standard J-STD-025. Oba standardy jsou si podobné, takže komerční zařízení dokáží podporovat oba způsoby komunikace s LEA provádějící odposlech.

Informace o komerčních nástrojích jsou volně přístupné pouze z části a o současných LIS není často možné volně získat podrobné informace. Volně dostupných materiálů však naznačují (např. [9]), že se komerční řešení nezabývají některými oblastmi síťové komunikace dostatečně do hloubky. V následujících částech této technické zprávy jsou bezpečnostní problémy současných LIS popsány.

3 Techniky ukrytí dat před systémy pro zákonné odposlechy

Tato kapitola se zabývá možnostmi, kterými mohou pachatelé trestné činnosti oklamat vyšetřovatele. Šifrování představuje nejběžnější možnost utajení obsahu

komunikace, se kterou se setkáváme, např. pokud pracujeme s internetovým bankovnictvím. Dále si představíme anonymizační sítě, které kromě utajení obsahu komunikace umožňují utajit i účastníky komunikace. Na závěr si představíme možnosti využití maskovacích technik, které využívají skrytých kanálů v síťových protokolech, snaží se obelstít odposlouchávajícího nebo se snaží utajit samotný výskyt podezřelé komunikace.

3.1 Šifrování dat

Kryptografie, nebo-li využití šifrování dat, představuje poměrně běžnou cestu, jak ochránit přenášená data před možností prozrazení obsahu. Uživatel má na výběr z několika aplikací a protokolů, které dokáží šifrování dat zajistit. Z těch nejčastěji se vyskytujících zmiňme alespoň Transport Layer Security (TLS) [32], Secure Shell (SSH) [95], IPsec [58] a Pretty Good Privacy (PGP) [17].

Pokud chce uživatel využít šifrování, je nutná spolupráce obou komunikujících stran. Iniciátor komunikace (A) kontaktuje druhého účastníka komunikace (B) a podle konkrétního použitého šifrovacího mechanismu si mohou A a B ověřit identitu druhé strany a obvykle se obě komunikující strany dohodnou na šifrovacím algoritmu a symetrickém klíči, které budou použity po inicializační fázi komunikace.

Obrázek 5. Přenos dat mezi oběma účastníky komunikace probíhá šifrovaně

ETSI požaduje, aby odposlouchávající organizace dešifrovala zachycenou komunikaci, nebo dodala použité šifrovací klíče v případě, že je použité šifrování v režii odposlouchávající organizace [37]. Pokud není ani jeden z účastníků komunikace pod kontrolou odposlouchávající organizace, pak není možné jednoduchou cestou získat klíče použité pro šifrování [30,31]. LIS na lince mezi účastníky komunikace A a B je však schopen odhalit probíhající komunikaci (a může např. vytvořit příslušné zprávy IRI), může identifikovat účastníky komunikace (např. jejich IP adresy) [48,57], ale bez znalosti klíče není LIS schopen předat vyšetřovatelům obsah komunikace v čitelné podobě. I z těchto dat je možné odvodit, kdo spolu komunikoval a získané informace využít pro další analýzu, např. [18,94].

3.2 Anonymizační techniky

Šifrovací techniky popisované v sekci 3.1 neumožňují utajit identitu komunikujících stran. Tato sekce popisuje metody, kterými je možné zamezit prozrazení

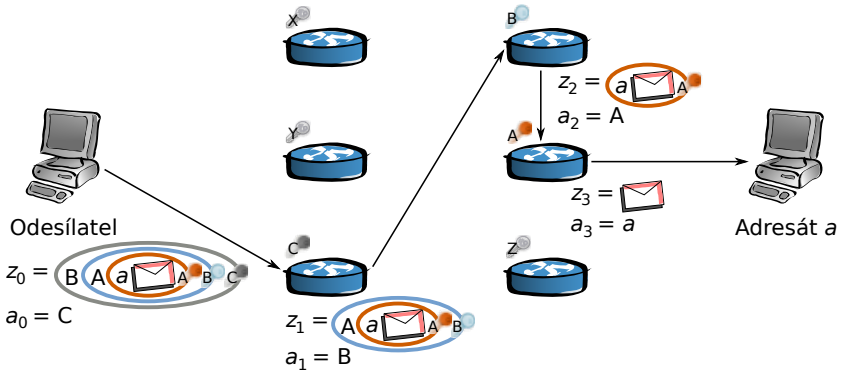
identity cíle komunikace, případně i zdroje, za předpokladu, že vyšetřující LEA provádějící odposlech má přístup pouze k omezené části Internetu. V kombinaci s využitím šifrování odchozího a příchozího síťového provozu podezřelého umožňují techniky popisované v této sekci omezit informací získanou zákonným odposlechem na pouhé časové periody, kdy mohl podezřelý komunikovat.

Systémy s velkým zpožděním doručení dat První metodou, která umožňovala skrytí identity komunikujících stran, je Mix-Net [21]. Při e-mailové komunikaci umožňuje skrýt identitu obou nebo jedné komunikující strany. Aby toho mohlo být dosaženo, spojil Mix-Net kryptografii se sítí serverů, které jsou nazývaný *mixy* a slouží k přeposílání zpráv. Doručení zprávy Z skrze anonymizační síť adresátovi a probíhá následovně:

1. Odesílatel si zvolí délku cesty v anonymizační síti k pro danou zprávu. Čím delší cesta, tím je složitější vystopovat cestu konkrétní zprávy anonymizační síti. S délkou cesty však také roste střední doba zpoždění doručení zprávy od odesílatele k adresátovi zprávy.
2. Odesílatel vybere náhodně k *mixů* a vytvoří cestu $c = (m_1, m_2 \dots m_k, a)$, kde $m_1 \dots m_k$ jsou zvolené *mixy* a a je adresát zprávy.
3. Odesílatel ze zprávy Z postupně vytváří zprávy $Z_0 \dots Z_k$. Zprávu Z_0 odešle odesílatel. Každému *mixu* m_i pro $i \in \{1, \dots, k\}$ bude doručena zpráva Z_{i-1}, kterou *mix* m_i dešifruje svým soukromým klíčem a získá zprávu Z_i a adresáta (m_{i+1}, či a v případě $i = k$).
   (a) Odesílatel inicializuje zprávu Z_k, kterou bude odesílat *mix* m_k na $Z_k = Z$ a adresáta a_k zprávy Z_k nastaví na $a_k = a$.
   (b) Odesílatel pro $i \in \{0, \dots, k-1\}$ vytvoří zprávu Z_i (postupuje postupně od $i = k-1$ po $i = 0$) tak, že zašifruje dvojici (Z_{i+1}, a_{i+1}) veřejným klíčem *mixu* m_{i+1} a nastaví $a_i = m_{i+1}$.
   (c) Výsledná odesílaná zpráva Z_0 je tedy nakonec tvořena několika vrstvami šifrovaných zpráv.
4. Odesílatel odešle a_0 zprávu Z_0.
5. Zpráva je doručena přes specifikované *mixy* m_1 až m_k tak, že každý *mix* m_i na cestě od odesílatele k příjemci z obdržené zprávy Z_{i-1} rozšifruje Z_i a a_i. Zprávu Z_i ale nepřepošle a_i okamžitě, ale s odesláním počká předem neznámou dobu. Během této doby jsou *mixu* m_i doručeny další zprávy posílané skrze anonymizační síť. Pro zvýšení anonymity každý *mix* mění pořadí odesílaných zpráv oproti pořadí, ve kterém mu byly zprávy doručeny.

Tím, že každý z *mixů* odstraňuje jednu vrstvu šifrování a zároveň odesílá zprávy v jiném pořadí, než ve kterém mu byly doručeny, se znemožňuje možná korelace zpráv při odposlechu komunikace v rámci anonymizační sítě a na jejich okrajích. Postup zasílání e-mailové zprávy anonymizační sítí je zachycen na obrázku 6.

Popsaný způsob doručení zpráv v systému Mix-Net zaručuje, že kromě odesílatele a adresáta zná každý uzel na cestě jen předchozí a následující uzel v anony-



Obrázek 6. Zasílání e-mailových zpráv anonymizační síti Mix-Net

mizační síti. Přeposílající uzly tedy nemají možnost zjistit identitu komunikujících stran. Poslední uzel v anonymizační síti, ale může odhadnout, že je poslední a pozorovat obsah zprávy Z.

Pro anonymní doručení odpovědi systém Mix-Net umožňuje, aby odesílatel vytvořil strukturu popisující zpáteční cestu. Za využití asymetrické kryptografie je možné zajistit, že i na zpáteční cestě zná každý z přeposílajících uzlů pouze identitu předchozího a následujícího uzlu v anonymizační síti, ale nezná adresy ostatních uzlů ani obsah přenášené zprávy.

Systémy *Mixmaster* [70], *Babel* [49] a *Miximinion* [28] se snažily vylepšit řešení anonymní komunikace navržené pro Mix-Net. Goldberg et al. [47] shrnují stav anonymizačních technik pro e-mailovou komunikaci na konci 90. let 20. století. Mazières a Kaashoek [66] popsali zkušenosti s provozem anonymizační e-mailové služby včetně několika případů, kdy došlo ke zneužití pro kriminální účely. V literatuře se objevují i pokusy zabývající se možnostmi získání identity komunikujících stran používající anonymizační techniky. Pfitzmann a Pfitzmann [79] nalezli možnost prozrazení identity v řešení založeném na Mix-Netu. Další studii možných hrozeb zpracoval Cottrell [24].

LIS umístěný na lince připojující odesilatele zprávy k Internetu vidí pouze to, že sledovaný počítač komunikoval s anonymizační sítí. Několikanásobné šifrování přenášené zprávy, brání zachycení obsahu zprávy, nebo alespoň jeho adresáta. Oproti tomu LIS umístěný na lince připojující adresáta a k Internetu vidí obsah zprávy, ale nemůže určit, kdo byl skutečným odesílatelem zprávy Z.

Systémy s nízkým zpožděním doručení dat Hlavní nevýhodou Mix-Netu a odvozených technik a služeb je velká latence. V některých případech (např. *anon.penet.fi*) trvalo několik dnů, než se zpráva dostala k adresátovi. Proto není možné uplatnit stejný postup na protokoly, u kterých záleží na poměrně rychlém doručení, např. při prohlížení WWW stránek. Postupně se začala objevovat nová řešení anonymizace, která berou v úvahu rychlost doručování.

Onion routing (OR) [48,81,90,91] představuje obecné řešení pro anonymizaci spojení řady aplikačních protokolů včetně HTTP, FTP, SMTP, telnet aj. OR předpokládá existenci známé sítě anonymizačních stanic (směrovačů) s dostupnými veřejnými kryptografickými klíči. Zdroj komunikace nejdříve vytvoří virtuální okruh pomocí struktury *onion* (cibule), která specifikuje ve vrstvách cestu k cíli. Princip rezervace vychází z postupu zasílání zpráv Mix-Netem. Jednotlivé směrovače na cestě vytvořenou cibuli loupou a rezervují dopřednou a zpětnou cestu pro vytvářený virtuální okruh. Kromě počátečního směrovače zná každý směrovač pouze identitu svých sousedů. Po vytvoření virtuálního okruhu může zdroj komunikace komunikovat s cílem pomocí brány proxy, kterou realizuje první směrovač v anonymizační síti. V rámci anonymizační sítě se posílají pakety pevné velikosti a veškerá komunikace je šifrovaná. Aby nebylo možné odhadnout, že v rámci sítě aktuálně proudí data, posílají jednotlivé směrovače vycpávkový provoz.

Autoři uznávají, že v případě monitorování vstupních i výstupních bran by bylo možné odhadnout, které stanice mezi sebou komunikují, ale pokud je anonymizační síť dostatečně velká, je těžké monitorovat všechny vstupní a výstupní brány.

Tor [33,4] navazuje na OR. Tor umožnil používat anonymizační sítě, aniž by využíval patenty, kterými byl zatížen OR. Tor přinesl následující vylepšení a změny:

- Vylepšené vytváření cesty. Cesta se vytváří postupně přidáváním dalšího uzlu k cestě vytvořené dříve. Tím se Tor brání před možností, kdy si některý z uzlů mohl ukládat provoz a později využít další uzly na původní cestě k rozšifrování obsahu přenášeného v původním spojení.
- Podporou pro využití proxy na bázi rozhraní SOCKS [59] se zvětšilo množství aplikací, které je možné anonymizovat.
- Neprobíhá přeuspořádávání zpráv před odesláním ve směrovačích v anonymizační síti, aby bylo sníženo zpoždění potřebné k průchodu síti Tor na minimum.
- Neposílají se prázdné zprávy sloužící jako výplň a neprobíhá řízení provozu, aby se minimalizovalo množství dat přenášených v rámci sítě Tor.
- Více TCP spojení může sdílet jeden virtuální okruh nebo jeho část. Tím pádem není nutné pro každé nové spojení potřeba vytvářet nový virtuální okruh. Vytvoření nového okruhu je výpočetně náročné a především pro krátká spojení HTTP představovalo vytváření vlastního okruhu velkou režii.
- Detekce zahlcení umožnila zvýšení stability sítě.
- Byla zavedena kontrola integrity přenášených zpráv. Při technice OR mohl uzel v rámci cesty pozměnit obsah a externí pozorovatel sledovat dění na výstupu sítě [28].
- Randez-vous body řeší problém pozdějšího zpětného navázání komunikace. OR obsahoval tzv. odpovědní cibule, ale po vypnutí uzlu nebo změně klíče na některém z uzlů obsažených v odpovědní cibuli nebylo možné zpětnou komunikaci navázat.

– Identita většiny uzlů v anonymizační síti je veřejně dostupná. Někteří poskytovatele připojení blokují přístup ke všem zveřejněným uzlům. Aby se k anonymizační sítí mohli připojit i zákaznicí takových poskytovatelů, byl zaveden speciální typ uzlu – *bridge*. Tor umožňuje každému klientovi sítě zjistit pouze několik uzlů typu *bridge* za časovou jednotku.
– Skrytá služba (Hidden Service) umožňuje v rámci sítě Tor provozovat anonymizovaný server (např. je možné provozovat internetový obchod s nabídkou ilegálních produktů [22]).

Protože Tor umožňuje anonymizovat pouze data přenášená protokolem TCP, je možné sledovat DNS dotazy zasílané protokolem UDP [33]. Autoři Toru doporučují používat při využití sítě Tor lokální proxy, která zabezpečí utajenou komunikaci s DNS servery.

Základní princip anonymizačních sítí typu OR a Tor je založený na fungování sítě Mix-Net, proto jsou i důsledky zapojení podezřelého do těchto anonymizačních sítí pro zákonné odposlechy podobné. LIS umístěný na lince připojující účastníka anonymizační sítě, je možné, že tento uživatele komunikuje s anonymizační sítí, ale není možné jednoduše zjistit skutečnou druhou stranu komunikace. Protože jsou data šifrována, není možné zjistit ani jejich obsah. Při odposlechu internetové služby (serveru) je možné odposlechnou komunikaci, která může být podle povahy služby i nešifrovaná, ale není možné zjistit IP adresu skutečného iniciátora komunikace. V případě, kdy je komunikace nešifrovaná, je možné identitu komunikujícího zjistit např. z přenášené e-mailové adresy v případě protokolu POP3.

3.3 Ukrývání informací

Uživatel sítě, který má motivaci znesnadnit odhalení své komunikace, může využít i další prostředky pro maskování svého síťového provozu. V této sekci se zaměříme na komunikační techniky využívající pro komunikaci takový způsob komunikace, který původně nebyl pro komunikaci navržen.

Skryté kanály Do skupiny ukrývání informací patří využití skrytých kanálů [96]. Skryté kanály byly původně popsány Lampsonem [61] jako možnosti úniku informací od procesu s vysokým stupněm ochrany k procesu s nízkým stupněm ochrany pomocí technik, které nebyly původně určeny ke komunikaci (např. přítomnost souboru s určitým jménem v souborovém systému). Ministerstvo obrany USA upravilo tuto definici [5] a považuje za skrytý kanál jakoukoliv možnost komunikace, při které se porušuje systémová bezpečnostní politika. V síťovém prostředí skryté kanály používají jako nosiče síťové protokoly [46]. Někdy jsou v literatuře skryté kanály popisovány i jako steganografické techniky (např. [43]).

Zander et al. [96] zpracovali přehled technik pro ukrytí komunikace. Je možné využít různé položky v hlavičkách protokolů IP, TCP a dalších k přenosu informací [73]. Například je možné přenášet informace v rámci položky TTL, v každém paketu 1 bit. Příjemce pak interpretuje nižší hodnoty jako 0 a vyšší hodnoty

jako 1 [83]. Je však možné využít většinu položek v hlavičkách včetně zdrojové a cílové adresy [46,64]. Informace je možné přenášet i délkou paketů [78], pořadím paketů [60], či uměle vyvolanou ztrátou paketů [85]. K přenosu informací je možné využít i tunely postavené nad běžně používanými protokoly, které jsou primárně určeny k jiným účelům (např. HTTP, ICMP).

Gianvecchio et al. [45] navrhli framework pro ovlivňování časových charakteristik provozu, pokud bude takový provoz sledován pomocí technik, jako je IPFIX, či Netflow, bude se takový provoz jevit, jako by patřil jiné třídě aplikací. Podobný efekt dokáže vyvolat i SkypeMorph [71], který maskuje provoz v síti Tor za provoz aplikace Skype.

Využití aplikačních protokolů přináší možnosti využití kanálů s velkou přenosovou kapacitou. Např. Mazurczyk et al. [68] nedávno představili způsob, kdy se účastníci telefonního hovoru navenek domluví na využití určitého kodeku, ale ve skutečnosti využijí jiný s nižším datovým tokem. Tím vznikne v datovém kanálu místo, které je možné využít pro další telefonní hovor nebo výměnu jiných dat.

Zmatení odposlouchávajících I bez spolupráce obou komunikujících stran je možné zmást odposlouchávajícího. Cronin et al. [25] popsali hlavní problémy odposlouchávání v rámci IP sítí:

- Nejednoznačnosti ve specifikacích protokolů, v implementacích a v konfiguraci. Decentralizovaná kontrola a různorodé implementace způsobují nepředvídatelné chování komunikujících stran při neočekávaném formátu vyměňovaných zpráv.
- Maximální snaha o doručení (*best effort delivery*) umožňuje ztrátu, zpřeházení a duplikaci přenášených paketů.
- Sdílený stav a kontext mezi komunikujícími stranami způsobuje komplikace pro odposlouchávající zařízení, které si tento stav musí také udržovat.
- Dynamické a asymetrické směrování, které se může změnit během času trvání komunikace, může způsobit, že odposlouchávající uvidí jen část komunikace, případně jen jeden ze směrů komunikace.
- Neexistuje autentizace zdroje ani příjemce, takže odposlouchávající nemá jistotu, že adresy uložené v hlavičkách jsou pravé.

Popsané vlastnosti nabízejí různorodé možnosti zneužití k oklamání odposlouchávajícího. Cronin et al. [25] popsali útoky vedoucí k oklamání nebo zmatení LIS.

Oklamání nastává, pokud LIS vůbec nedokáže detekovat to, že byla nějaká zpráva odeslána. Příkladem takového útoku je např. využití nízké tolerance LI zařízení oproti síťovému zařízení (přepínač, směrovač apod.), které paket zpracovává jako další na cestě.

Ke *zmatení* LIS dochází, pokud LIS vidí jinou zprávu, než jaká byla ve skutečnosti odeslána, nebo pokud LIS vidí velké množství zpráv a nemůže se rozhodnout, která z nich je pravá a která je podvržená. Příkladem takového útoku je odeslání několika TCP segmentů se stejným sekvenčním číslem, ale různým obsahem. To, který ze segmentů se dostane k cíli, může být ovlivněno hodnotu

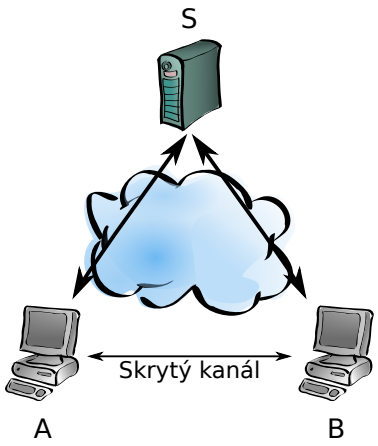
TTL v IP hlavičce (Hop Limit v IPv6 hlavičce), velikostí paketu (některé segmenty mohou překračovat MTU) apod. Pokud se k cíli dostane více segmentů s různým obsahem, pak není definováno, který obsah se TCP/IP zásobníkem dostane do cílové aplikace [86].

Pro zmatení LIS je možné využít i softwarové chyby. Například Bhargavan et al. [13] popsali chyby poštovních systémů, které umožňují dostat mailového agenta do stavu, který nevyhovuje normě. V takovém případě by mohlo dojít k odeslání elektronické pošty, aniž by LIS vygeneroval příslušné zprávy IRI.

Při použití technik pro skrývání informací je LIS buďto obelstěn takovým způsobem, že přenášená data vůbec nezachytí, i když jsou ve skutečnosti přenášená po lince, kterou monitoruje (oklamání), nebo LIS data zachytí, ale jejich analýza neodhalí pravý obsah komunikace (skryté kanály, zmatení).

3.4 Možnosti nepřímé komunikace

V literatuře jsou popsány i způsoby utajené komunikace mezi uživateli A a B využívající třetího subjektu tak, že není zřejmé, že se nějaká komunikace mezi A a B odehrává (viz obrázek 7). Pro tyto účely je možné využít nespecifikované vlastnosti běžně používaných protokolů, jako je např. IP, HTTP [27].



Obrázek 7. Uživatele A a B mění stav serveru S a tím mezi sebou nepozorovaně komunikují

Jednou z možností [27], kterou mohou účastníci tajné komunikace využít, je pole *Identifier* obsažené v hlavičce IPv4 [54]. Hodnota přenášená v tomto

poli musí být unikátní v rámci paketů přenášených síti v rámci jednoho spojení. Některé OS používají globální čítač společný pro všechna spojení. Pokud se účastníci komunikace domluví na využití konkrétního serveru, může odesílatel zprávy v určitých časových okamžicích ovlivňovat hodnotu globálního čítače serveru a příjemce zprávy si v určitých okamžicích může zkontrolovat aktuální hodnotu globálního čítače a dekódovat přenášený symbol.

Další z možností [27,65] je využití počítadla navštívených stránek, které se objevují na některých webových stránkách. Podobně jako v předchozím případě odesílatel zprávy ovlivňuje v určitých časových okamžicích svými požadavky hodnotu počítadla a tím odesílá jeden, nebo více symbolů a příjemce zprávy v určitých časech kontroluje hodnotu počítadla.

Při odposlechu podezřelého, který využívá možnosti nepřímé komunikace, nedokáže LIS identifikovat skutečné komunikující počítače. Při odposlechu uživatele A, B, či obou (z obrázku 7) se dá předpokládat umístění LIS na přístupové lince vedoucí od uživatele A, respektive B k Internetu. Odposlechnutá data sice budou pravděpodobně obsahovat komunikaci se serverem S, ale protože tento server bude záměrně volen tak, aby byl co nejméně nápadný, nemusí být ani po důkladné analýze odposlechnutých dat zřejmé, že se nějaká komunikace mezi uživatelem A a B uskutečnila.

3.5 Shrnutí anonymizačních metod

V této kapitole byly představeny techniky, které je možné využít pro ukrytí komunikace před LIS. Tyto techniky jsou shrnuty v tabulce 1. Každá technika je uvedená v jednom řádku a tabulka ve sloupcích obsahuje:

- **Šifrovaný přenos dat** označuje metody anonymizačních technik, které ke svému fungování inherentně používají šifrování dat. Ve všech případech však může odesílatel před odesláním obsah šifrovat.
- **Utajení komunikujících stran** označuje takové metody, které se primárně zaměřují na utajení komunikujících uživatelů sítě.
- **Přímá komunikace** se odehrává pokud jsou data přenášena přímo od odesílatele k adresátovi zasílané zprávy.
- **Nutnost spolupráce komunikujících** je potřebná, pokud musí obě strany používat určitý software či jeho upravenou verzi.
- **Maskování komunikace** nastává, pokud odesílatel zasílá kromě skutečné komunikace i jiná data, která ve skutečnosti pouze kryjí skutečný obsah přenášené zprávy.

4 Současné možnosti rozpoznání anonymizovaného provozu

Obsahem této kapitoly je popis současných možností pro analýzu anonymizačních technik popsaných v kapitole 3.

	Šifrovaný přenos dat	**Utajení komunikujících stran**	**Přímá komunikace**	**Nutnost spolupráce komunikujících**	**Maskování komunikace**
Šifrování dat	Ano	Ne	Ano	Ano	Ne
Anonymizační sítě	V rámci sítě	Ano	Ne	Ne	Ne
Skryté kanály	Ne	Ne	Ano	Ano	Ano
Zmatení odposlouchávajících	Ne	Pouze při oklamání	Ano	Ne	Ano
Ovlivňování globálních počítadel	Ne	Ano	Ne	Ano	Ano

Tabulka 1. Shrnutí možností anonymizačních technik

4.1 Anonymizační sítě

Prvním problémem, který je potřeba řešit je samotná detekce provozu v anonymizační síti. Znalost toho, že je odposlouchávaný subjekt zapojen do anonymizační sítě přináší odposlouchávající agentuře dvě výhody: 1) agentuře stačí ukládat pouze metainformace o provozu, protože k šifrovaných obsahu nebude mít k dispozici klíče a tím pádem jej nemůže rozšifrovat; 2) může využít některou z níže uvedených technik, které řeší problém identifikace komunikujících stran v anonymizační síti.

V současnosti je Tor [4] pravděpodobně nejpoužívanější anonymizační sítí. V době psaní této technické zprávy bylo do této sítě zapojeno přibližně 3 000 směrovačů sítě Tor [3]. Na začátků roku obsahovala anonymizační síť přibližně 2 400 směrovačů. Velikost sítě a tedy i poskytovaná anonymita výrazně rostou. Kvůli velikosti sítě Tor se v souvislosti s anonymizací síťového provozu mluví především o této anonymizační síti. Cílem projektu spravujícího síť Tor však není ochrana uživatelů před globální pozorovatelem, ale snaží se zachovat anonymitu před pozorovatelem s pouze částečnou kontrolou nad sítí Tor [4,72].

Provoz v síti Tor bývá přenášen v buňkách o velikosti 586 B pomocí protokolu TCP. Pokud není naplno využitá přenosová kapacita mezi komunikujícími uzly sítě Tor, je možné na síti pozorovat zvýšené množství paketů s délkou payloadu TCP o velikosti 586 B, či v jejich násobcích. Je možné, že v budoucích verzích bude tato možnost detekce odstraněna a provoz sítě Tor se bude maskovat za jiný, běžný síťový provoz [71].

Šifrovaná data jsou charakteristická tím, že mají vysokou entropii [42]. Další pomůckou pro odhalení provozu v síti Tor je veřejný seznam směrovačů zapojených do sítě [3].

Studie zkoumající využití sítě Tor [63,69,19] ukazují, že je tato síť často používaná v kombinaci s protokoly přenášející nešifrovaný text, které mohou pomoci s odhalením identity uživatele. Výstupní uzel z anonymizační sítě může sledovat odchozí provoz a v případě, že se mu podaří zjistit identitu uživatele alespoň jediného toku z virtuálního okruhu vytvořeného v rámci sítě Tor, pak je tento uživatel také odesílatelem dat pro všechna ostatní TCP spojení uskutečněná v rámci tohoto virtuálního okruhu [62]. Toho je možné využít i v kombinaci s jinými aplikačními protokoly, jako je například HTTP [34], BitTorrent [62] a další [69].

Autoři sítě Tor [4] doporučují používat ve spojení s touto sítí další anonymizační techniky, jako je například anonymizační proxy `privoxy` [1]. V současné literatuře je však možné nalézt případy, kdy uživatelé anonymizační síť používají takovým způsobem, který případnému sledujícímu usnadňuje možnosti identifikace. Např. Chaabane et al. [19] zjistili, že někteří uživatelé využívají v síti cestu tvořenou jediným uzlem. Tento uzel tedy zná jak zdroj, tak cíl realizované komunikace.

Další techniku umožňující zjištění identity uživatele v síti Tor popsali Chakravarty et al. [20]. Popisovaná technika mění charakteristiku provozu posílaného ze strany serveru směrem k výstupním uzlu ze sítě Tor. Pokud je vytvořená charakteristika provozu dostatečně reprezentativní, pak může odposlouchávající pozorovat, zda anonymizovaný provoz této charakteristiky dorazil až ke sledované osobě a tím pádem se přesvědčit, zda tato osoba opravdu komunikovala s konkrétním serverem. Tento scénář má následující nedostatky:

1. Odposlouchávající potřebuje mít přístup přímo k serveru, ke kterému se připojuje sledovaná osoba, nebo musí mít možnost ovlivňovat linky mezi tímto serverem a výstupními uzly ze sítě Tor.
2. Odposlouchávající ovlivňuje charakteristiku provozu, což je v rozporu s normami pro LI.

Denezis [26] studoval možnosti korelace provozu vstupujícího do anonymizační sítě založené na mixech s provozem vystupujícím z této sítě. Johnson et al. [75] tuto metodu s drobnými změnami vyzkoušeli v malé síti Tor skládající se ze čtyřech uzlů. Otázkou však je, jestli by uvedená metoda byla použitelná i v sítí s mnohem vyšším počtem uzlů.

V literatuře nalezneme i další techniky zaměřené na anonymizační sítě obecně. Raymod [80] neformálně popsal několik možných slabin anonymizačních sítí. Berthold et al. [12] popsali několik možností, jak sledovat průchod dat anonymizační sítí a zjišťovat totožnost komunikujících. Navržené možnosti předpokládají možnost značkování provozu a tedy nejsou využitelné v síti Tor, která kontroluje integritu zasílaných dat. Další nevýhodou je nutnost kontroly nad Mixy, které zjišťování identity provádějí. Výchozí délka cesty v síti Tor jsou 3 skoky [4].

[1] http://www.privoxy.org

Při přibližném počtu 3 000 směrovačů by bylo potřeba kontrolovat 1 655 z nich, aby pravděpodobnost, že virtuální tunel bude procházet alespoň jedním z námi kontrolovaných uzlů, byla alespoň 0,9.

Další skupina technik umožňujících útoky na anonymizační sítě se zaměřuje na nasazení upravených uzlů do sítě. Bauer et al. [11] zkoumali možnosti při kompromitaci vstupních a výstupních uzlů sítě Tor. Navržená technika využívá faktu, že jsou preferovány uzly, které jsou stabilně dostupné a inzerují velkou dostupnou kapacitu. Výsledky ukazují, že i s poměrně malým podílem přenosové kapacity kompromitovaných uzlů z celkové přenosové kapacity sítě, je možné získat informace o velkém množství spojení. Ze své povahy však tato forma útoku není vhodná pro LIS.

Herrmann et al [51] popsali vytvoření otisků jednotlivých internetových stránek složených z velikosti přenášených paketů, jejich četnosti. Tyto otisky je pak možné využít pomocí vícehodnotových naivních Bayesových klasifikátorů především pro identifikaci šifrovaných spojení. V omezené míře je tento postup možný použít i pro anonymizační sítě. Pro toto využití je však potřeba další výzkum.

4.2 Ukrývání informací

Normalizaci dat [50] je možné využít také pro odstraňování možných skrytých kanálů [84]. Normalizaci přenášených dat je však nutné kombinovat s dalšími opatřeními. Jedním z nich je odstranění časových skrytých kanálů, čehož je možné docílit např. náhodným pozdržením požadavků. Tím pádem není potřeba skryté kanály přímo detekovat, ale stačí preventivně odstraňovat prostředky umožňující samotný vznik skrytého kanálu, případně podniknutými opatřeními alespoň omezit přenosovou kapacitu skrytého kanálu. Podobný způsob popisují i Zhiyong et al. [97]. Tyto přístupy však nelze použít v rámci LIS, protože LIS je pouze pasivní prvek, který nemá možnost zasahovat do komunikace.

Browne [14] přinesl myšlenku, že je možné měřit entropii systému a pokud změřená entropie neodpovídá očekávané, pak se v systému nachází skrytý kanál, který tuto entropii ovlivňuje. Počítání entropie se jeví jako poměrně častý a slibný způsob detekce skrytých kanálů.

Využitím entropie při hledání skrytých časových kanálů se zabýval i Stillman [87]. Navržený přístup je založen na zkoumání rozestupů mezi odesílanými pakety. Ze získaného histogramu se odvodí řetězec bitů, které by mohly být odeslanými pakety reprezentovány. Odvozený řetězec je následně vyhledán v paměťovém prostoru procesu, který pakety do sítě odeslal. V případě LIS však není možné nahlížet do operační paměti odposlouchávaného uživatele.

Cabout et al. [16] vytvořili metriku nazývanou ε-podobnost. Metrika je založená na sledování délky mezirámcových mezer. Pokud je v seřazené posloupnosti naměřených mezirámcových mezer patrná schodovost, jedná se pravděpodobně o jeden z typů časových skrytých kanálů.

Gianvecchio a Wang [44] ukázali, že ε-podobnost je vhodná pouze pro určité typy časových skrytých kanálů. Kromě ε-podobnosti porovnali i další možnosti pro detekci skrytých kanálů: Kolmogorovův-Smirnovův test [77], test pravidel-

nosti [16] a metody založené na entropii. Detekovány byly různé typy časových kanálů a z testovaných metod nebyl nalezen jednoznačný vítěz testu.

Mazurczyk et al. [67] zkoumali skryté kanály využívající mechanismus opětovného zasílání nepotvrzených paketů v TCP. Navrhují sledovat četnost opakovaného přenosu segmentů a porovnání obsahu. Upozorňují na problém týkající se chyb v sítí (přibližně 0,09 % paketů [89]). LIS v případě odposlechů veškeré komunikace (CC) ukládá všechna data, takže porovnání obsahu není v případě LIS problém. Toto porovnání je však potřeba navrhnout dostatečně efektivně, aby příliš nezdržovalo zpracování dat.

Tumoian a Anikeev [92] zkoumali generování počátečního sekvenčního čísla protokolem TCP a možnosti předcházení jeho zneužití pro vytváření skrytého kanálu [82]. Pro detekci navrhují použít neuronové sítě a jsou schopni detekovat skryté kanály s vysokou přesností.

5 Příklad nástroje pro ukrytí síťového provozu a detekci tohoto útoku

V této sekci si představíme nástroj LIS Deception Proxy (LDP) pro oklamání LIS využívající metodu pro oklamání nástrojů pro rekonstrukci provozu inspirovanou prací Paxsona [76] a Cronina et al. [25] aplikovanou v prostředí Internetového protokolu verze 6 (IPv6). Doplňkem k tomuto nástroji je aplikace LIS Noise Cleaner (LNC), která slouží k čištění souboru ve formátu libpcap, které byly zašuměné útokem na bázi nástroje LDP. Obě aplikace vznikly jako součást projektu *Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace* [53].

5.1 Nástroj pro oklamání LIS

Cílem LDP je zajistit přenos tajné zprávy k příjemci takovým způsobem, aby z dat získaných na základě odposlechu byla zpráva obtížně analyzovatelná. Pro realizaci tohoto typu útoku není nutná spolupráce vzdáleného příjemce. Útok je pro komunikující subjekty transparentní.

Princip útoku spočívá v úmyslném vytváření kolizních TCP segmentů (více segmentů se stejným sekvenčním číslem). Z nich vždy pouze jeden tvoří bajt (znak) skutečné zprávy a zbytek představuje šum. Snaha o rekonstrukci dat zachycených LIS pak vede k nejednoznačnosti – v případě kolizních segmentů není jasné, který znak má být použit pro rekonstrukci přenášené zprávy.

LDP pracuje jako překladač NPTv6 [93] kombinovaný s možností segmentace procházejících toků TCP. Společně se segmentovanými daty TCP je možné kromě šumu zasílat podvrženou zprávu, což dále znesnadňuje pozdější analýzu dat. Pomocí dynamické modifikace pole Hop Limit v hlavičce IPv6 [29] je zajištěno zahození datagramů se šumem a falešnou zprávou na dříve než dorazí k příjemci. Na základě experimentů Cronina et al. [25] je tento princip útoku velmi účinný proti dnešním nástrojům pro rekonstrukci datového toku.

Pro úspěšnou realizaci útoku je nutné znát počet skoků k příjemci. Navíc může být použito asymetrické směrování – počet skoků na cestě k příjemci a zpět může být různý (pakety jsou směrovány odlišnou cestou). Z tohoto důvodu je detekce počtu skoků prováděna dvěma způsoby:

- Na základě Hop Limitu získaného z prvního paketu přijatého od vzdáleného příjemce je určen předpokládaný počet skoků (*Hop Count – HC*). LDP vychází z předpokladu že ve většině Unixových systémů je výchozí Hop Limit u odesílaných IPv6 datagramů roven hodnotě 64 a v systémech Microsoft Windows hodnotě 128. Pokud tedy Hop Limit zprávy zachycené od vzdáleného počítače označíme jako H_{msg}, můžeme říci, že:
  Pokud $H_{msg} \leq 64$, pak $HC_{est} = 64 - H_{msg}$
  a pokud $64 < H_{msg} \leq 128$, pak $HC_{est} = 128 - H_{msg}$
  Kde HC_{est} je předpokládaný počet skoků.
- Předpokládaný počet skoků je poté ověřen pomocí ICMPv6 zpráv typu *echo* [23]. Vždy je poslána zpráva, jejíž Hop Limit je nastaven na určitou hodnotu a je ověřováno, zda přijde odpověď. Princip určení počtu skoků znázorňuje vývojový diagram na obrázku 8. Cyklus ověřování je zahájen s předpokládaným počtem skoků (HC_{est}). Pokud přicházejí odpovědi, je počet skoků snižován, pokud ne, je zvyšován. Jakmile je nalezena hranice, kde přestávají, resp. začínají přicházet odpovědi, je zjišťování ukončeno.

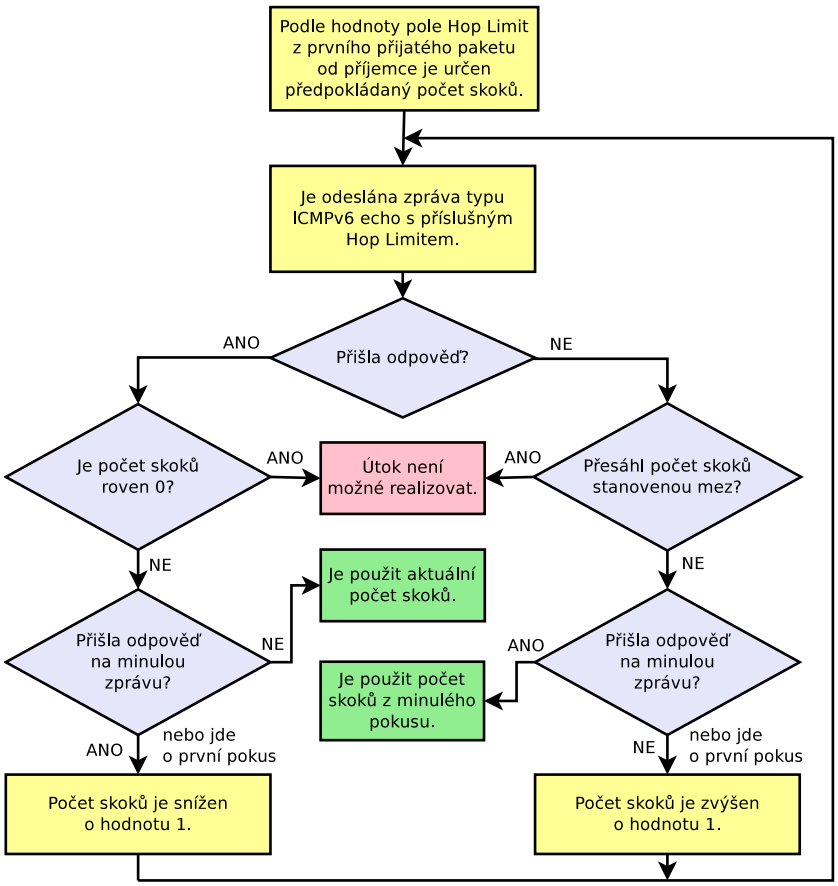
Samozřejmě existují situace, kdy útok není možné realizovat. Jednou z nich je případ, kdy je příjemce přímo připojen do sítě, ve které běží i LDP (na cestě není žádný směrovač, který by paket zahodil). Dalším případem je směrování s vyvažováním zátěže (*load balancing*), kdy jsou pakety odesílané směrem ke vzdálenému příjemci střídavě směrovány různými cestami. Pokud je počet skoků na těchto cestách různý, nastává situace, kterou LDP nedokáže řešit. Nikdy totiž nelze předem odhadnout, která cesta bude v daný moment pro posílaný paket vybrána.

Podsystém LDP pro fragmentaci zprávy a generování šumu přenášenou zprávu je dělí (segmentuje) do jedno-bajtových segmentů a každý tento bajt je odesílán v samostatném datagramu IPv6. Kromě odesílání datagramů, které obsahují původní zprávu, jsou odesílány i takové, které tvoří šum. Cílem vysílení šumu je ukrytí přenášené tajné zprávy, což výrazně zvyšuje šance na oklamání systému pro zákonné odposlechy.

Nechť φ je minimální hodnota pole Hop Limit v hlavičce IPv6 datagramu, která je nutná k úspěšnému dosažení cíle. HL_{msg} je Hop Limit datagramů obsahujících segment přenášené zprávy. Hop Limit datagramů obsahujících šum HL_{noise} je volen tak, aby platilo:

$$HL_{msg} \geq \varphi \ \wedge \ HL_{noise} < \varphi$$

Tím je zajištěno zahození šumu směrovači na cestě k příjemci. Techniku fragmentace zprávy a generování šumu naznačuje příklad na obrázku 9. Původní šesti-bajtová zpráva je fragmentována na jednotlivé bajty. Každý bajt je přenášen v samostatném TCP segmentu s příslušně stanoveným sekvenčním číslem.

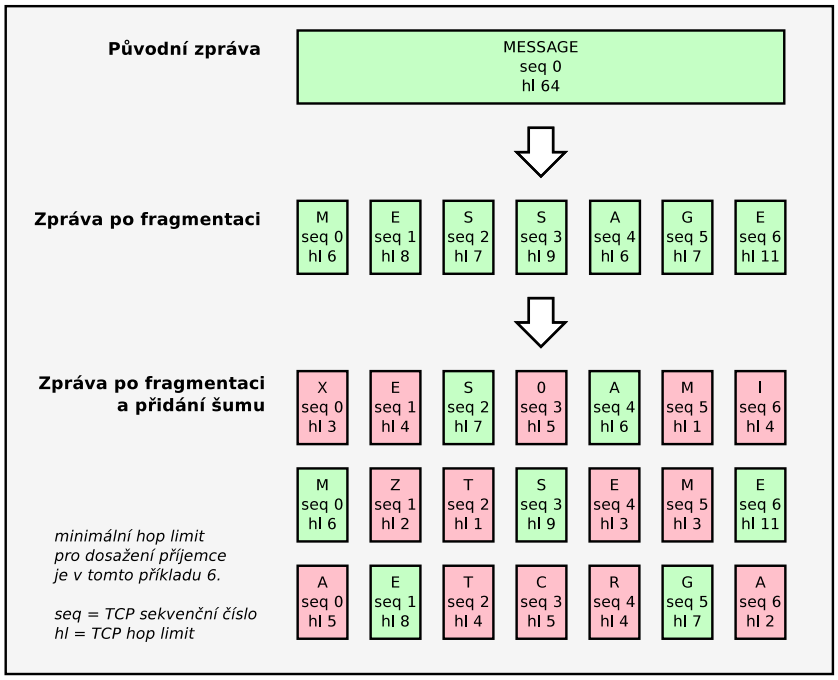


Obrázek 8. Určení počtu skoků od LDP k cílovému počítači

TCP segment je poté zapouzdřen v IPv6 datagramu s vhodně zvolenou hodnotou Hop Limit.

S každým segmentem původní zprávy jsou na obrázku 9 posílány dva segmenty obsahující šum. Každá segment z trojice segmentů v jednom sloupci ve spodní části obrázku obsahuje kolizní sekvenční čísla. Při rekonstrukci přenášené zprávy tedy není jednoznačné, který znak se má použít, což je podstatou útoku realizovaného LDP.

Co by se stalo,pokud by skutečně došlo k přijetí kolizních TCP segmentů? Implementace TCP/IP v různých operačních systémech vykazují odlišné typy chování. Některé využívají k sestavení první přijatý segment [15], některé segment nejčastěji se vyskytující [15], některé naopak poslední přijatý [86]. V praxi mohou být navíc pro rekonstrukci dat na základě odposlechu použity specializované nástroje pro analýzu datového toku. Není také vyloučeno hledání výrazů



Obrázek 9. Příklad fragmentace zprávy a přidání šumu. Přenášená zpráva je znázorněna zeleně a šum červeně.

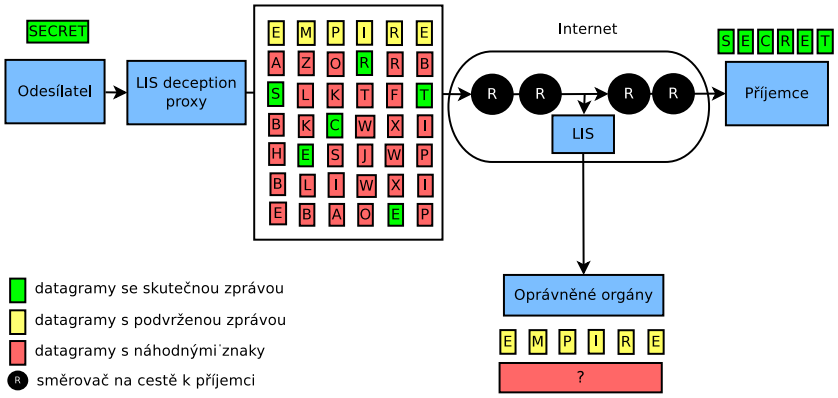
v určitém jazyce na základě lexikální analýzy zachycených dat. V takovýchto případech by mohl být útok velmi efektivní např. v kombinaci s využitím kryptografie, či steganografie – tyto techniky popisuje sekce 3.

Vzhledem k různým možnostem přístupu ke kolizním TCP segmentům umožňuje nástroj LIS Deception Proxy použít jako šum kromě náhodných znaků také jednu, či dvě **podvržené zprávy**. Přičemž byte jedné z nich je odeslán vždy jako první a druhé vždy jako poslední z kolizních paketů s daným sekvenčním číslem. Pokud by tedy v případě kolize byl použit vždy první (resp. poslední) ze zachcených segmentů, byla by interpretována jedna (resp. druhá) podvržená zpráva.

Příklad útoku s použitím jedné podvržené zprávy ilustruje obrázek 10.

Bod v rámci kterého je odposlech prováděn se nazývá Intercept Access Point (IAP) [35]. Umístění IAP v bezprostřední blízkosti proxy serveru je nejvýhodnější z hlediska útočníka. Důvodem je největší zachycené množství šumu ze všech uvedených případů. Nejvýhodnější z hlediska odposlouchávajícího je umístit IAP co nejblíže k příjemci. Čím blíže cíli totiž budeme, tím méně šumu by bude zachyceno.

Pro dosažení maximální efektivity útoku je pro určování Hop Limitu datagramů se šumem použit generátor pseudonáhodných čísel s rozložením $f(x) =$



Obrázek 10. Demonstrace činnosti programu LDP.

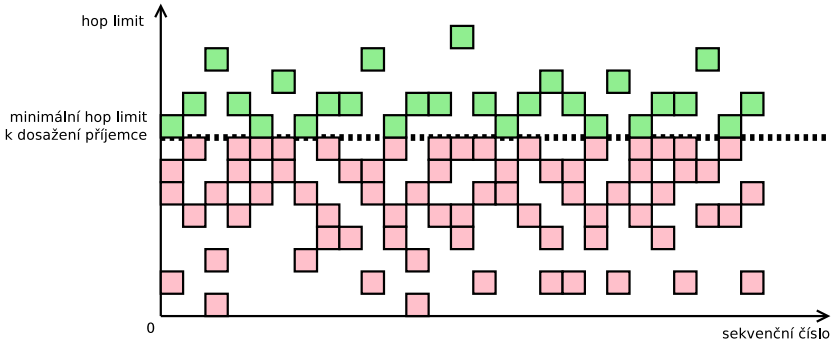
$\frac{2x}{max^2}$. Hodnota max představuje nejvyšší Hop Limit, kdy bude datagram ještě zahozen. Vlastností tohoto rozložení je, že datagramy obsahují častěji vyšší hodnotu Hop Limit, než nižší. Samozřejmě by bylo možné stanovit Hop Limit datagramů se šumem vždy na nejvyšší (max) hodnotu. Tato skutečnost by ale mohla vést k jednoduchému zjištění počtu skoků na základě dat zachycených LIS. Stačilo by potom odfiltrovat pouze datagramy s touto hodnotou.

5.2 Nástroje pro odstranění šumu

Aplikace LNC pro odstranění šumu pracuje se soubory ve formátu libpcap [2]. Spektrum paketů odeslaných LDP zachycených LIS vypadá přibližně jako na obrázku 11. Červeně označené jsou pakety se šumem a podvrženými zprávami, zeleně označené tvoří přenášenou zprávu. Lze s jistotou říci, že z hlediska hodnoty Hop Limit bude v případě tohoto typu útoku vždy existovat hranice (označená přerušovanou čarou), za kterou se již nebude vyskytovat žádný šum. Navíc bude pro každé sekvenční číslo existovat pouze jeden segment s vyšší hodnotou Hop Limit než je minimální hodnota pro dosažení příjemce.

V praxi však dochází k situacím, kdy je původní TCP segment znovu odeslán odesilatelem a LDP tak vygeneruje pro každé sekvenční číslo několik sad kolizních segmentů.

LNC při čištění souboru ve formátu libpcap pracuje dvou-průchodově. V prvním průchodu LNC hledá pro každý TCP segment identifikovaný zdrojovou a cílovou IP adresou a číslem portu a sekvenčním číslem maximální Hop Limit a přenášený obsah. V druhém průchodu pak LNC vytváří soubor ve formátu pcap s odstraněným šumem. Do výstupního souboru se dostane pouze první segment přenášející stejný obsah jako ten s nejvyšším Hop Limitem. Tím je zajištěno to, že segment s konkrétním sekvenčním číslem a obsahem bude ve výstupním souboru co nejdříve, čímž se zabrání problémům některých aplikací, pokud detekují



Obrázek 11. Příklad hodnot pole Hop Limit přenášených v zachycených paketech

tok TCP se segmenty přenášenými mimo pořadí. Obrázek 12 ukazuje příklad nesprávné rekonstrukce TCP streamu pomocí funkce *Follow TCP Stream*.

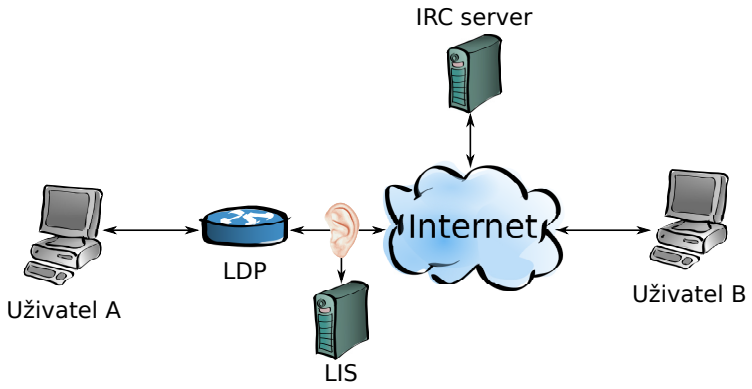
```
GET / HTTP/1.1
Accept: image/gif, image/jpeg, image/pjpeg, image/pjpeg, application/x-shockwave-flash, app
application/msword, application/xaml+xml, application/vnd.ms-xpsdocument, application/x-ms-
silverlight, */*
Accept-Language: cs
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0; .NET CLR 1.1.43
InfoPath.1; .NET CLR 3.0.04506.648; .NET CLR 3.0.4506.2152; .NET CLR 3.5.30729)
Accept-Encoding: gzip, deflate
If-Modified-Since: Tue, 10 Apr 2012 02:29:15 GMT
If-None-Match: "941edc-a-4bd49e39a3156"
Host: [fd02:1111::eeed:0:0:2]
Connection: Keep-Alive

H[2 bytes missing in capture file]P/1.1[1 bytes missing in capture file]200[3 bytes missing
[3 bytes missing in capture file]te: T[1 bytes missing in capture file]e, 1[1 bytes missing
[1 bytes missing in capture file]2[1 bytes missing in capture file]0[2 bytes missing in cap
missing in capture file]
[2 bytes missing in capture file]er[2 bytes missing in capture file]r[4 bytes missing in ca
bytes missing in capture file]([1 bytes missing in capture file]eb[1 bytes missing in captu
missing in capture file]-[1 bytes missing in capture file]o[1 bytes missing in capture file
in capture file] [1 bytes missing in capture file]u[3 bytes missing in capture file]10[2 by
capture file]2[1 bytes missing in capture file]12[1 bytes missing in capture file]02[2 byte
capture file]
```

Obrázek 12. Ukázka problému při rekonstrukci toku se segmenty mimo pořadí v programu Wireshark

6 Příklad ukrytí dat při komunikaci protokolem IRC

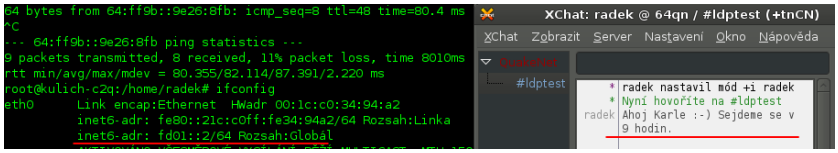
Cílem tohoto experimentu [53] bylo ověřit účinnost LDP a LNC při utajené komunikaci při využívání protokolu IRC pro komunikací v reálném čase. Síťová

topologie použitá v rámci experimentu je znázorněna na obrázku 13. Uživatelé A a B jsou připojeni k IRC serveru QuakeNet a vzájemně spolu komunikují v rámci kanálu `#ldptest`. Data odesílaná uživatelem A jsou před zachycením zašuměná programem LDP.



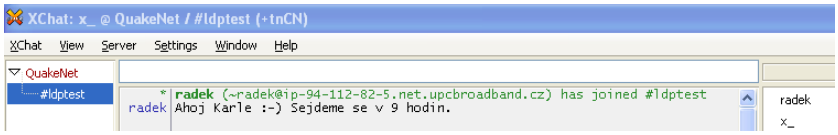
Obrázek 13. Síťová topologie pro experimentální ověření vlastností LDP a LNC pro komunikaci protokolem IRC

Pro konfiguraci LDP byly použity dvě podvržené zprávy, úroveň šumu pět náhodných znaků na každý bajt skutečné zprávy. V každém z pěti opakování experimentů byla fragmentovaná zpráva společně se šumem odeslána a IRC serveru byla úspěšně doručena pouze původní zpráva. Vzájemná komunikace s uživatelem B skrz server QuakeNet probíhala bez problémů. Na obrázku 14 je snímek obrazovky uživatele A (útočníka), který zaslal uživateli B zprávu „Ahoj Karle :-) Sejdeme se v 9 hodin“. Na obrázku 15 je snímek obrazovky uživatele B, který úspěšně zobrazil zprávu přenesenou na server.



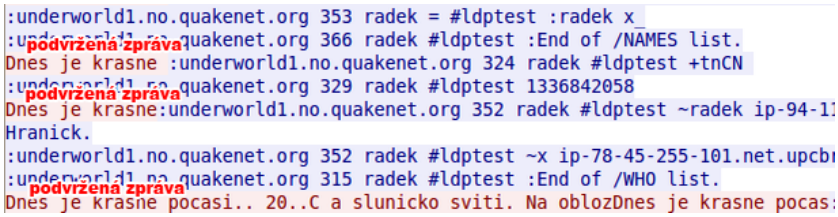
Obrázek 14. Snímek obrazovky uživatele A (útočníka)

Ze zachycených dat byla provedena rekonstrukce TCP toku pomocí nástroje Wireshark. Jak lze vidět na obrázku 16, byla zrekonstruována podvržená zpráva. Počet použitých znaků samozřejmě závisel na délce přenášené zprávy. V praxi



Obrázek 15. Snímek obrazovky uživatele B

by bylo zřejmě vhodnější použít dynamicky generovaný text namísto statického, nicméně pro účely experimentu a demostraci útoku je tento způsob dostačující.



Obrázek 16. Pokus o rekonstrukci dat se šumem pomocí nástroje Wireshark

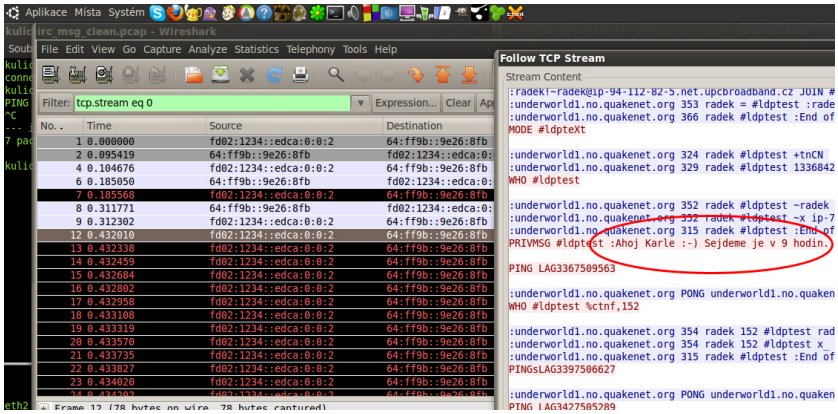
Data se šumem byla následně vyčištěna programem LNC. Ve všech případech se podařilo úspěšně zrekonstruovat průběh původní komunikace. Výsledek ukazuje obrázek 17.

7 Shrnutí

Tato technická zpráva se zabývala analýzou útoků na systémy pro zákonné odposlechy včetně ukázky realizace jedné z představených metod. Popisovanými útoky je možné obelstít pomocí systémů pro zákonné odposlechy vycházejících ze dvou ze současných hlavních standardů pro zákonný sběr dat v počítačových sítích. Ani evropské normy ETSI, ani americký standard J-STD-025 a dokonce ani architektura pro zákonné odposlechy firmy Cisco se možností ukrytí dat před systémy pro zákonné odposlechy nezabývají.

V současné době je pravděpodobně nejčastější metodou pro ukrytí dat využití šifrování. Běžně používané nástroje jako je SSH, SSL, PGP apod. zajišťují vysokou míru utajení přenášené informace a bez dostupných klíčů není možné přenášený obsah v přijatelné době dešifrovat. Samotné šifrování dat však neumožňuje komunikujícím stranám přenos dat utajit. Pokročilé anonymizační techniky dovolují komunikujícím přenášet tajné informace pomocí skrytých kanálů, či dalších technik.

Prostorové skryté kanály využívají často nejednoznačnosti použitých síťových protokolů. Jedním z příkladu prostorového skrytého kanálu je využití nevýznamných nebo nevyužívaných polí v hlavičce některého z použitých protokolů.



Obrázek 17. Úspěšná rekonstrukce původní komunikace po vyčištění programem LNC

Dalsím příkladem může být využití délky paketu pro přenos skryté informace. Časové skryté kanály využívají pro přenos dat časové relace mezi pakety, jako je např. čas odeslání, rozestup mezi pakety apod.

Další možností pro utajenou komunikaci je využití anonymizační sítě, jako je např. Tor, nebo I2P. Kromě šifrování přenášených dat anonymizační sítě značně znesnadňují možnost identifikace komunikujících stran.

Pro odhalení neobvyklého provozu v počítačových sítí je možné využít celou řadu metod. Některé však nejsou vhodné pro účely zákonných odposlechů, protože zasahují do síťové komunikace. Prozatím nebyla nalezena žádná univerzální metoda umožňující detekci libovolného útoku.

V rámci projektu *Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace* [53] vznikly nástroje LIS Deception Proxy (LDP) a LIS Noise Cleaner (LNC) pro demonstraci a odhalení konkrétního útoku na LI systémy. Aplikace LDP ukrývá data přenášených protokolem TCP v prostředí sítí založených na protokolu IPv6. Implementovaná metoda útoku byla inspirována prací Cronina et al. [25], kteří nenašli žádnou aplikaci, která by uměla v přenášených datech automaticky nalézt skutečně přenášenou zprávu. Pro účely předcházení útokům provedených aplikací LDP, nebo útoků založených na podobném principu byla v rámci této práce představena aplikace LNC, která dokáže ze souboru obsahujícího pakety zachycené na síti odstranit šum, který je podstatou útoku realizovaného aplikací LDP.

Reference

1. I2P Anonymous Network - I2P. [[online]], citováno 2011-12-27.
   URL http://www.i2p2.de/

2. TCPDUMP/LIBPCAP public repository. [[online]], citováno 2011-10-24.
URL http://www.tcpdump.org/
3. Tor Network Status. [[online]], citováno 2012-01-12.
URL http://torstatus.blutmagie.de
4. Tor Project: Anonymity Online. [[online]], citováno 2011-12-28.
URL https://www.torproject.org/
5. Department of Defense Trusted Computer System Evaluation Criteria. Technická Zpráva DoD 5200.28-STD, prosinec 1985.
6. COUNCIL RESOLUTION of 17 January 1995 on the lawful interception of telecommunications (96/C 329/01). Listopad 1996.
7. Aboba, B.; Aboba, B.; Arkko, J.; aj.: *RFC 4282 The Network Access Identifier*. Prosinec 2005.
8. Alliance for Telecommunications Industry Solutions/Telecommunications Industry Association Joint Standard: *Lawfully Authorized Electronic Surveillance. J-STD-025-B*. Červenec 2006.
9. Aqsacom: *Lawful Interception for Internet Protocol (IP) Networks*. Březen 2010, aqsacom Document No. 040451, V4.0.
10. Baker, F.; Foster, B.; Sharp, C.: *RFC 3924 Cisco Architecture for Lawful Intercept in IP Networks*. Říjen 2004.
11. Bauer, K.; McCoy, D.; Grunwald, D.; aj.: Low-resource routing attacks against tor. In *Proceedings of the 2007 ACM workshop on Privacy in electronic society*, WPES '07, New York, NY, USA: ACM, 2007, ISBN 978-1-59593-883-1, s. 11–20.
12. Berthold, O.; Pfitzmann, A.; Standtke, R.: The Disadvantages of Free MIX Routes and How to Overcome Them. In *Designing Privacy Enhancing Technologies*, *Lecture Notes in Computer Science*, ročník 2009, editace H. Federrath, Springer Berlin / Heidelberg, 2001, ISBN 978-3-540-41724-8, s. 30–45.
13. Bhargavan, K.; Gunter, C. A.: Network Event Recognition. *Formal Methods in System Design*, ročník 27, 2005: s. 213–251, ISSN 0925-9856.
14. Browne, R.: An entropy conservation law for testing the completeness of covert channel analysis. In *Proceedings of the 2nd ACM Conference on Computer and communications security*, CCS '94, New York, NY, USA: ACM, 1994, ISBN 0-89791-732-4, s. 270–281.
15. Burns, B.: IDS/IPS Evasion with Overlapping TCP Segments. [online], 2010, citováno 2012-10-30.
URL http://forums.juniper.net/t5/Security-Mobility-Now/IDS-IPS-Evasion-with-Overlapping-TCP-Segments/ba-p/52410
16. Cabuk, S.; Brodley, C. E.; Shields, C.: IP covert timing channels: design and detection. In *Proceedings of the 11th ACM conference on Computer and communications security*, CCS '04, New York, NY, USA: ACM, 2004, ISBN 1-58113-961-6, s. 178–187.
17. Callas, J.; Donnerhacke, L.; Finney, H.; aj.: *RFC 4880 OpenPGP Message Format*. Listopad 2007.
18. Carley, K. M.; Dombroski, M.; Tsvetovat, M.; aj.: Destabilizing dynamic covert networks. In *Proceedings of the 8th International Command and Control Research and Technology Symposium*, Washington, DC, 2003.

19. Chaabane, A.; Manils, P.; Kaafar, M. A.: Digging into Anonymous Traffic: A Deep Analysis of the Tor Anonymizing Network. In *Proceedings of the 2010 Fourth International Conference on Network and System Security*, NSS '10, Washington, DC, USA: IEEE Computer Society, 2010, ISBN 978-0-7695-4159-4, s. 167–174.
20. Chakravarty, S.; Stavrou, A.; Keromytis, A.: Identifying Proxy Nodes in a Tor Anonymization Circuit. In *IEEE International Conference on Signal Image Technology and Internet Based Systems*, prosinec 2008, s. 633–639.
21. Chaum, D. L.: Untraceable electronic mail, return addresses, and digital pseudonyms. *Commun. ACM*, ročník 24, únor 1981: s. 84–90, ISSN 0001-0782.
22. Christin, N.: Traveling the Silk Road: A measurement analysis of a large anonymous online marketplace. Technická Zpráva CMU-CyLab-12-018, CyLab, Carnegie Mellon University, Pittsburgh, USA, červenec 2012.
23. Conta, A.; Deering, S.; Gupta, M.: *RFC 4443 Internet Control Message Protocol (ICMPv6) Internet Control Message Protocol (ICMPv6)*. Březen 2006.
24. Cottrell, L.: *Mixmaster & Remailer Attacks*. Citováno 2011-11-22. URL http://www.dia.unisa.it/~ads/corso-security/www/NEW/remailer-essay
25. Cronin, E.; Sherr, M.; Blaze, M.: On the (un)reliability of eavesdropping. *Int. J. Secur. Netw.*, ročník 3, únor 2008: s. 103–113, ISSN 1747-8405.
26. Danezis, G.: The Traffic Analysis of Continuous-Time Mixes. In *Privacy Enhancing Technologies*, *Lecture Notes in Computer Science*, ročník 3424, editace D. Martin; A. Serjantov, Springer Berlin / Heidelberg, 2005, ISBN 978-3-540-26203-9, s. 742–746.
27. Danezis, G.: Covert Communications Despite Traffic Data Retention. In *Security Protocols XVI*, *Lecture Notes in Computer Science*, ročník 6615, editace B. Christianson; J. Malcolm; V. Matyas; M. Roe, Springer Berlin / Heidelberg, 2011, ISBN 978-3-642-22136-1, s. 198–214.
28. Danezis, G.; Dingledine, R.; Mathewson, N.: Mixminion: design of a type III anonymous remailer protocol. In *2003 Symposium on Security and Privacy.*, květen 2003, ISSN 1081-6011, s. 2–15.
29. Deering, S.; Hinden, R.: *RFC 2460 Internet Protocol, Version 6 (IPv6) Specification*. Prosinec 1998.
30. Denning, D.; Smid, M.: Key escrowing today. *Communications Magazine, IEEE*, ročník 32, č. 9, září 1994: s. 58 –68, ISSN 0163-6804.
31. Denning, D. E.: The future of cryptography. In *The governance of cyberspace: politics, technolgy and global restructuring*, editace B. D. Loader, kapitola 11, Taylor & Francis e-Library, 2005, s. 173–188.
32. Dierks, T.; Rescorla, E.: *RFC 5246 The Transport Layer Security (TLS) Protocol Version 1.2*. Srpen 2008.
33. Dingledine, R.; Mathewson, N.; Syverson, P.: Tor: the second-generation onion router. In *Proceedings of the 13th conference on USENIX Security Symposium - Volume 13*, SSYM'04, Berkeley, CA, USA: USENIX Association, 2004.
34. Eckersley, P.: How Unique Is Your Web Browser? In *Privacy Enhancing Technologies*, *Lecture Notes in Computer Science*, ročník 6205, editace M. Atallah; N. Hopper, Springer Berlin / Heidelberg, 2010, ISBN 978-3-642-14526-1, s. 1–18.

35. European Telecommunications Standards Institute: *ETSI TR 101 943: Telecommunications security; Lawful Interception (LI); Concepts of Interception in a generic Network Architecture*. Červenec 2001, version 1.1.1.
36. European Telecommunications Standards Institute: *ETSI ES 201 158: Telecommunications security; Lawful Interception (LI); Requirements for network functions*. Duben 2002, version 1.2.1.
37. European Telecommunications Standards Institute: *ETSI TR 102 528: Lawful Interception (LI); Interception domain Architecture for IP networks*. Říjen 2006, version 1.1.1.
38. European Telecommunications Standards Institute: *ETSI TR 101 331: Lawful Interception (LI); Requirements of Law Enforcement Agencies*. Říjen 2009, version 1.3.1.
39. European Telecommunications Standards Institute: *ETSI TR 102 232-3: Lawful Interception (LI); Handover Interface and Service-Specific Details (SSD) for IP delivery; Part 3: Service-specific details for internet access services*. Říjen 2009, version 2.2.1.
40. European Telecommunications Standards Institute: *ETSI TR 101 671: Lawful Interception (LI); Handover interface for the lawful interception of telecommunications traffic*. Srpen 2010, version 3.6.1.
41. European Telecommunications Standards Institute: *ETSI TR 102 232-1: Lawful Interception (LI); Handover Interface and Service-Specific Details (SSD) for IP delivery; Part 1: Handover specification for IP delivery*. Srpen 2010, version 2.5.1.
42. Fawcett, T. W.: *Exfild: A Tool For The Detection of Data Exfiltration Using Entropy And Encryption Characteristics of Network Traffic*. Diplomová práce, University of Delaware, 2010.
43. Fisk, G.; Fisk, M.; Papadopoulos, C.; aj.: Eliminating Steganography in Internet Traffic with Active Wardens. In *Information Hiding*, *Lecture Notes in Computer Science*, ročník 2578, editace F. Petitcolas, Springer Berlin / Heidelberg, 2003, ISBN 978-3-540-00421-9, s. 18–35.
44. Gianvecchio, S.; Wang, H.: Detecting covert timing channels: an entropy-based approach. In *Proceedings of the 14th ACM conference on Computer and communications security*, CCS '07, New York, NY, USA: ACM, 2007, ISBN 978-1-59593-703-2, s. 307–316.
45. Gianvecchio, S.; Wang, H.; Wijesekera, D.; aj.: Model-Based Covert Timing Channels: Automated Modeling and Evasion. In *Proceedings of the 11th international symposium on Recent Advances in Intrusion Detection*, RAID '08, Berlin, Heidelberg: Springer-Verlag, 2008, ISBN 978-3-540-87402-7, s. 211–230.
46. Girling, C. G.: Covert channels in LAN's. *IEEE Transactions on Software Engineering*, ročník 13, č. 2, 1987: s. 292–296.
47. Goldberg, I.; Wagner, D.; Brewer, E.: Privacy-enhancing technologies for the Internet. In *Compcon '97. Proceedings, IEEE*, únor 1997, s. 103–109.
48. Goldschlag, D.; Reed, M.; Syverson, P.: Hiding Routing information. In *Information Hiding*, *Lecture Notes in Computer Science*, ročník 1174, editace R. Anderson, Springer Berlin / Heidelberg, 1996, ISBN 978-3-540-61996-3, s. 137–150.

49. Gulcu, C.; Tsudik, G.: Mixing E-mail with Babel. In *Proceedings of the Symposium on Network and Distributed System Security*, únor 1996, s. 2–16.
50. Handley, M.; Paxson, V.; Kreibich, C.: Network Intrusion Detection: Evasion, Traffic Normalization, and End-to-End Protocol Semantics. In *Proceedings of the 10th conference on USENIX Security Symposium - Volume 10*, SSYM'01, 2001.
51. Herrmann, D.; Wendolsky, R.; Federrath, H.: Website fingerprinting: attacking popular privacy enhancing technologies with the multinomial naïve-bayes classifier. In *Proceedings of the 2009 ACM workshop on Cloud computing security*, CCSW '09, New York, NY, USA: ACM, 2009, ISBN 978-1-60558-784-4, s. 31–42.
52. Hoffman, P.; Terplan, K.: *Intelligence Support Systems: technologies for lawful intercepts*. Auerbach Publications, 2006, ISBN 0-8493-2855-1.
53. Hranický, R.: *Podvržená data v počítačových sítích*. Bakalářská práce, Vysoké učení technické v Brně, 2012.
54. Information Sciences Institute University of Southern California: *RFC 791 Internet Protocol*. Září 1981.
55. International Organization for Standardization: *ISO/IEC international standard 7498-1:1994 Information Technology – Open Systems Interconnection – Basic Reference Model: The Basic Model*. 1994.
56. IP Fabrics, Inc: *DeepProbe: 1Gbps and 10Gbps IP Data Collection Probes*. 2011, citováno 2012-01-18.
URL http://www.ipfabrics.com/pdf/DeepProbe.pdf
57. K., M.; Sparrow: The application of network analysis to criminal intelligence: An assessment of the prospects. *Social Networks*, ročník 13, č. 3, 1991: s. 251 – 274, ISSN 0378-8733.
58. Kent, S.; Seo, K.: *RFC 4301 Security Architecture for the Internet Protocol*. Prosinec 2005.
59. Koblas, D.; Koblas, M. R.: Socks. In *Proceedings of the UNIX Security III Symposium*, září 1992, s. 77–83.
60. Kundur, D.; Ahsan, K.: Practical Internet Steganography: Data Hiding in IP. In *Texas Wksp. Security of Information Systems*, duben 2003, str. 5.
61. Lampson, B. W.: A note on the confinement problem. *Commun. ACM*, ročník 16, říjen 1973: s. 613–615, ISSN 0001-0782.
62. Le Blond, S.; Manils, P.; Chaabane, A.; aj.: One bad apple spoils the bunch: exploiting P2P applications to trace and profile Tor users. In *Proceedings of the 4th USENIX conference on Large-scale exploits and emergent threats*, LEET'11, Berkeley, CA, USA: USENIX Association, 2011.
63. Loesing, K.; Murdoch, S.; Dingledine, R.: A Case Study on Measuring Statistical Data in the Tor Anonymity Network. In *Financial Cryptography and Data Security*, *Lecture Notes in Computer Science*, ročník 6054, editace R. Sion; R. Curtmola; S. Dietrich; A. Kiayias; J. Miret; K. Sako; F. Sebé, Springer Berlin / Heidelberg, 2010, ISBN 978-3-642-14991-7, s. 203–215.
64. Lucena, N.; Lewandowski, G.; Chapin, S.: Covert Channels in IPv6. In *Privacy Enhancing Technologies*, *Lecture Notes in Computer Science*, ročník 3856, editace G. Danezis; D. Martin, Springer Berlin / Heidelberg, 2006, ISBN 978-3-540-34745-3, s. 147–166.

65. Luo, X.; Chan, E.; Chang, R.: Crafting Web Counters into Covert Channels. In *New Approaches for Security, Privacy and Trust in Complex Environments*, *IFIP International Federation for Information Processing*, ročník 232, editace H. Venter; M. Eloff; L. Labuschagne; J. Eloff; R. von Solms, Springer Boston, 2007, ISBN 978-0-387-72366-2, s. 337–348.
66. Mazières, D.; Kaashoek, M. F.: The design, implementation and operation of an email pseudonym server. In *Proceedings of the 5th ACM conference on Computer and communications security*, CCS '98, New York, NY, USA: ACM, 1998, ISBN 1-58113-007-4, s. 27–36.
67. Mazurczyk, W.; Smolarczyk, M.; Szczypiorski, K.: Retransmission steganography and its detection. *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, ročník 15, 2011: s. 505–515, ISSN 1432-7643.
68. Mazurczyk, W.; Szaga, P.; Szczypiorski, K.: Using Transcoding for Hidden Communication in IP Telephony. *CoRR*, ročník abs/1111.1250, listopad 2011: str. 17.
69. McCoy, D.; Bauer, K.; Grunwald, D.; aj.: Shining Light in Dark Places: Understanding the Tor Network. In *Privacy Enhancing Technologies*, *Lecture Notes in Computer Science*, ročník 5134, editace N. Borisov; I. Goldberg, Springer Berlin / Heidelberg, 2008, ISBN 978-3-540-70629-8, s. 63–76.
70. Moeller, U.: *Mixmaster Protocol Version 3*. Citováno 2011-11-22. URL http://www.eskimo.com/~rowdenw/crypt/Mix/draft-moeller-v3-01.txt
71. Moghaddam, H. M.; Li, B.; Derakhshani, M.; aj.: SkypeMorph: Protocol Obfuscation for Tor Bridges. In *Proceedings of the 2012 ACM conference on Computer and communications security*, New York, NY, USA: ACM, říjen 2012, ISBN 978-1-4503-1651-4.
72. Murdoch, S.; Danezis, G.: Low-cost traffic analysis of Tor. In *Security and Privacy, 2005 IEEE Symposium on*, květen 2005, ISSN 1081-6011, s. 183–195.
73. Murdoch, S.; Lewis, S.: Embedding Covert Channels into TCP/IP. In *Information Hiding*, *Lecture Notes in Computer Science*, ročník 3727, editace M. Barni; J. Herrera-Joancomartí; S. Katzenbeisser; F. Pérez-González, Springer Berlin / Heidelberg, 2005, ISBN 978-3-540-29039-1, s. 247–261.
74. Narten, T.; Draves, R.; Krishnan, S.: *RFC 4941 Privacy Extensions for Stateless Address Autoconfiguration in IPv6*. Září 2007.
75. Nick Johnson, J. T., Steve McLaughlin: Path Tracing In Tor Networks. In *18th European Signal Processing Conference*, srpen 2010, s. 1856–1860.
76. Paxson, V.: Bro: a system for detecting network intruders in real-time. *Computer Networks*, ročník 31, č. 23-24, 1999: s. 2435 – 2463.
77. Peng, P.; Ning, P.; Reeves, D.: On the secrecy of timing-based active watermarking trace-back techniques. In *2006 IEEE Symposium on Security and Privacy*, květen 2006, ISSN 1081-6011, str. 15.
78. Perkins, M. C.: *Hiding out in Plaintext: Covert Messaging with Bitwise Summations*. Diplomová práce, Iowa State University, 2005.
79. Pfitzmann, B.; Pfitzmann, A.: How to Break the Direct RSA-Implementation of Mixes. In *Advances in Cryptology — EUROCRYPT '89*, *Lecture Notes in Computer Science*, ročník 434, editace J.-J. Quisquater; J. Vandewalle, Springer Berlin / Heidelberg, 1990, ISBN 978-3-540-53433-4, s. 373–381.

80. Raymond, J.-F.: Traffic Analysis: Protocols, Attacks, Design Issues, and Open Problems. In *Designing Privacy Enhancing Technologies*, *Lecture Notes in Computer Science*, ročník 2009, editace H. Federrath, Springer Berlin / Heidelberg, 2001, ISBN 978-3-540-41724-8, s. 10–29.
81. Reed, M.; Syverson, P.; Goldschlag, D.: Anonymous connections and onion routing. *Selected Areas in Communications, IEEE Journal on*, ročník 16, č. 4, květen 1998: s. 482 –494, ISSN 0733-8716.
82. Rutkowska, J.: The Implementation of Passive Covert Channels in the Linux Kernel. In *Chaos Communication Congress*, prosinec 2004.
83. S. Zander, P. B., G. Armitage: Covert Channels in the IP Time to Live Field. In *Australian Telecommunication Networks and Applications Conf.*, prosinec 2006, ISBN 0-97-758610-3.
84. Schear, N.; Kintana, C.; Zhang, Q.; aj.: Glavlit: Preventing Exfiltration at Wire Speed. In *Record of the Fifth Workshop on Hot Topics in Networks: HotNets V*, Beckman Center, University of California, Irvine, California, USA: ACM, listopad 2006, s. 133–138.
85. Servetto, S.; Vetterli, M.: Communication using phantoms: covert channels in the Internet. In *IEEE International Symposium on Information Theory*, červen 2001, str. 229.
86. Shankar, U.; Paxson, V.: Active mapping: resisting NIDS evasion without altering traffic. In *Symposium on Security and Privacy*, květen 2003, ISSN 1081-6011, s. 44–61.
87. Stillman, R.: Detecting IP covert timing channels by correlating packet timing with memory content. In *Southeastcon, 2008. IEEE*, duben 2008, s. 204–209.
88. Stoll, C.: Stalking the Wily Hacker. *Commun. ACM*, ročník 31, květen 1988: s. 484–497, ISSN 0001-0782.
89. Stone, J.; Partridge, C.: When the CRC and TCP checksum disagree. *SIGCOMM Comput. Commun. Rev.*, ročník 30, srpen 2000: s. 309–319, ISSN 0146-4833.
90. Syverson, P.; Reed, M.; Goldschlag, D.: Onion routing access configurations. In *DARPA Information Survivability Conference and Exposition, 2000. DISCEX '00. Proceedings*, ročník 1, 2000, s. 34 –40 vol.1.
91. Syverson, P.; Tsudik, G.; Reed, M.; aj.: Towards an Analysis of Onion Routing Security. In *Designing Privacy Enhancing Technologies*, *Lecture Notes in Computer Science*, ročník 2009, editace H. Federrath, Springer Berlin / Heidelberg, 2001, ISBN 978-3-540-41724-8, s. 96–114.
92. Tumoian, E.; Anikeev, M.: Network Based Detection of Passive Covert Channels in TCP/IP. In *The IEEE Conference on Local Computer Networks, 2005. 30th Anniversary*, listopad 2005, ISSN 0742-1303, s. 802–809.
93. Wasserman, M.; Baker, F.: RFC 6296 IPv6-to-IPv6 Network Prefix Translation. červen 2011.
94. Xu, J.; Marshall, B.; Kaza, S.; aj.: Analyzing and Visualizing Criminal Network Dynamics: A Case Study. In *Intelligence and Security Informatics*, *Lecture Notes in Computer Science*, ročník 3073, editace H. Chen; R. Moore; D. Zeng; J. Leavitt, Springer Berlin / Heidelberg, 2004, ISBN 978-3-540-22125-8, s. 359–377.

95. Ylonen, T.; Lonvick, C.: *RFC 4251 The Secure Shell (SSH) Protocol Architecture.* Leden 2006.
96. Zander, S.; Armitage, G.; Branch, P.: A survey of covert channels and countermeasures in computer network protocols. *Communications Surveys Tutorials, IEEE*, ročník 9, č. 3, 2007: s. 44–57, ISSN 1553-877X.
97. Zhiyong, C.; Yong, Z.: Integrated Covert Channel Countermeasure Model in MLS Networks. In *Information Engineering and Computer Science, 2009. ICIECS 2009. International Conference on*, prosinec 2009, s. 1–4.

A Seznam zkratek

- AF – *Administration Function* – Administrační funkce
- AP – *Access Provider* – Provozovatel veřejné přístupové komunikační sítě
- CC – *Content of Communication* – Obsah komunikace zachycené LIS
- CC-IIF – *Content of Communication - Internal Interception Function* – *Funkce odposlechu obsahu komunikace*
- CCC – *Call Content Channel* – Americká obdoba HI3
- CDC –*Call Data Channel* – Americká obdoba HI2
- CCTF – *Content of Communication Trigger Function* – Trigerovací funkce
- CCTI – *Content of Communication Trigger Interface*
- DHCP – *Dynamic Host Configuration Protocol*
- ETSI – *European Telecommunications Standards Institute* – Evropský ústav pro telekomunikační normy
- HI *Handover Interface*
- IAP – *Intercept Access Point* – Bod, ze kterého se pro odposlech získávají síťová data
- INI – *Internal Network Interface*
- IP – *Internet Protocol* (bez rozlišení verze)
- IPv4 – *Internet Protocol version 4*
- IPv6 – *9*Internet Porotocol version 6
- IRI – *Intercept Related Information Function*
- IRI-IIF – *Intercept Related Information - Internal Interception Function* – Funkce dynamické identity
- ISO/OSI – *International Organization for Standardization/Open Systems Interconnection*
- LEA – *Lawful Enforcement Agency* – Orgány činné v trestním řízení
- LEMF – *Law Enforcement Monitoring Facility* – Monitorovací stanoviště LEA
- I2P – *The Invisible Internet Project* – Anonymizační síť
- LDP – *LIS Deception Proxy* – Program pro realizaci útoku skrytí dat před LIS
- LIS – *Lawful Interception System* – Systém pro sběr dat pro zákonné odposlechy
- LNC – *LIS Noise Cleaner* – Program pro nalezení skryté komunikace ukryté např. programem LDP

– MD – *Mediation Device*
– MF – *Mediation Function*
– MTU – *Maximal Transmission Unit* – Maximální velikost přenášeného paketu
– NAI – *Network Access Identifier*
– NPTv6 – *IPv6-to-IPv6 Network Prefix Translation*
– NWO – *Network Operator* – Provozovatel veřejné komunikační sítě
– SAS – *Surveillance Administration System* – Americká obdoba HI1
– SvP – *Service Provider* – Poskytovatel služeb
– TCP – *Transmission Control Protocol*
– TTL – *Time to Live* – Položka v hlaviččce protokolu IPv4