Zákonné odposlechy v moderních sítích

Shrnutí výsledků skupiny pro zákonné odposlechy projektu Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace

Technická zpráva FIT VUT

Libor Polčák, Tomáš Martínek, Radek Hranický, Stanislav Bárta, Martin Holkovič, Barbora Franková, Petr Kramoliš

Technický report č. FIT-TR-2014-07
Fakulta informačních technologií, Vysoké učení technické v Brně

Last modified: 16. ledna 2015

Obsah

Kapitola 1

Úvod

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 [2] a jsou standardizovány Evropským ústavem pro telekomunikační normy (*European Telecommunications Standards Institute* – ETSI). Při získávání dat přenášených v počítačových sítích spolupracují orgány činné v trestním řízení 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.

Cílem této souhrnné technické zprávy je shrnutí výsledků dosažených skupinou pro zákonné odposlechy projektu *Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace* (Sec6Net) financovaného grantem Ministerstva vnitra ČR (VG20102015022). Tato technická zpráva obsahuje aktualizované podstatné informace zveřejněné v rámci předešlých technických zpráv [65, 50, 60] a publikovaných článků souvisejících s projektem [58, 59, 61, 57]. Současně tato zpráva obsahuje podstatné informace pro instalaci a konfiguraci systému, či jeho budoucí rozšíření.

1.1 Definice pojmů

Zákonné odposlechy představují způsob boje s kriminální, pirátskou, podvodnou, či teroristickou činností. Na základě nařízení orgánů činných v trestním řízení je prováděn záznam a analýza síťové komunikace mezi podezřelými osobami.

- **Zákonný odposlech** (*Lawful Interception* – LI) umožňuje sledovat aktivitu podezřelých osob využívajících veřejných komunikačních prostředků jako jsou telefonní sítě nebo Internet. Dle norem ETSI je pojem zákonný odposlech definován [16] jako činnost prováděná poskytovatelem komunikačních služeb (NWO/AP/SvP). Cílem této činnosti je poskytnutí konkrétních informací orgánům činným v trestním řízení.

- **Systém pro zákonné oposlechy** (*Lawful Interception System* – LIS) je nástroj umožňující oprávněným orgánům realizaci zákonných odposlechů.
- **Internetový protokol** (*Internet Protocol* – IP) je dominantní protokol síťové vrstvy [35] pro komunikaci mezi počítači zapojenými na Internetu. Od 80. let 20. století se v rámci Internetu používá verze č. 4 (IPv4) [34]. V 90. letech 20. století se však ukázalo, že především množství IP adres nedostačuje požadavkům nově připojovaných zařízení. Nástupcem IPv4 je protokol IP verze 6 (IPv6) [14], který mimo 4-násobného zvětšení IP adresy přinesl řadu dalších novinek včetně odlišného způsobu generování adres a autokonfigurace [58] (o problémech spojených s generováním IPv6 adres pojednávají sekce 6.4 a 6.5).
- **IP adresa** je identifikátor počítače, či místa připojení k Internetu. Rozlišujeme IP adresy verze 4 a 6, které jsou vzájemně odlišné a jsou přidělovány odlišnými mechanismy. Jeden počítač může mít více IP adres (u protokolu IPv6 je to pravidlem). Pokud nedochází k překladu adres (např. kvůli šetření množství dostupných IP adres pro danou síť) je daná IP adresa koncového zařízení jedinečná v rámci Internetu. U protokolu IPv4 však k překladu adres (*Network Address Translation* – NAT) dochází velmi často.
- **MAC adresa** (*Media Access Control* – MAC) je jedinečný identifikátor síťového rozhraní, který je používán pouze v lokální síti a zpravidla nebývá součásti komunikace mimo lokální síť.

1.2 Zákonné odposlechy z hlediska legislativy ČR a EU

Z hlediska legislativy ČR nutnost poskytování LI plyne ze zákona č. 127/2005 Sb. ve znění pozdějších předpisů, *Zákon o elektronických komunikacích* [71], hlava V, díl 1, odposlech a záznam zpráv: Provozovatel veřejné komunikační sítě, či veřejně dostupné služby elektronických komunikací je povinen na základě vyžádání oprávněného orgánu poskytnout tomuto orgánu provozní a lokalizační údaje. Obdobně je tato povinnost ukotvena také v legislativě Evropské unie [2].

Oba dokumenty uchovávané údaje podrobněji specifikují dle konkrétního typu sítě a povahy přenášených dat. U sítí elektronických komunikací s přepojováním paketů tyto údaje zahrnují identifikaci obou stran, čas zahájení komunikace, dobu přenosu, množství přenesených dat a další údaje.

1.3 Představení systému pro zákonné odposlechy

Systém pro zákonné odposlechy (*Sec6Net Lawful Interception System* – SLIS) byl vytvořen na Vysokém učení technickém v Brně pro potřeby výzkumu v oblasti zákonných odposlechů, analýzy dat a jejich rekonstrukce a hardwarové akcelerace sond určených pro sběr a identifikaci dat. SLIS dokáže spolupracovat se sondami vytvořenými v rámci projektu *Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace.*

1.4 Struktura dokumentu

Tato technická zpráva je zaměřena na popis prototypu zařízení pro realizaci odposlechu na straně poskytovatele služeb navrženého v rámci projektu *Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace.* Tato zpráva je členěna do devíti kapitol:

Kapitola 2 popisuje současné standardy pro zákonné odposlechy a otevřené dokumenty dostupné pro komerční řešení zákonných odposlechů.

Kapitola 3 popisuje zvolenou architekturu použitou pro systém SLIS, jeho nasazení a vstupní a výstupní rozhraní. Tato kapitola také podává motivaci pro specifikaci úrovní odposlechů. Tato kapitola obsahuje základní informace pro práci se systémem.

Kapitola 4 se zabývá životním cyklem odposlechu a vnitřnostmi systému. Tato kapitola se zabývá podrobným popisem jednotlivých funkčních bloků. Tato kapitola je určena pouze zájemcům o detailní architekturu vytvořeného systému.

Kapitola 5 se podrobně zabývá principem dynamické identifikace na obecné úrovni. V této kapitole je představen graf identifikátorů detekovaných v síti, jeho konstrukce a operace nad ním. Navržený graf usnadňuje korelaci identity uživatelů v moderních sítích a dokáže se vypořádat s problémy vznikajícími v souvislosti s IPv6 a odposlechy na základě aplikačních identifikátorů.

Kapitola 6 poskytuje popis podporovaných protokolů pro zjišťování identity uživatelů. Ke každému protokolu, či metodě je uveden základní popis a činnost systému.

Kapitola 7 se zabývá uživatelskou stránkou práce se systémem. SLIS je distribuován jako živá instalovatelná distribuce založená na OS Linux, konkrétně Ubuntu 12.04 LTS. Kromě instalace, kapitola popisuje konfiguraci systému, jeho zabezpečení a ovládání.

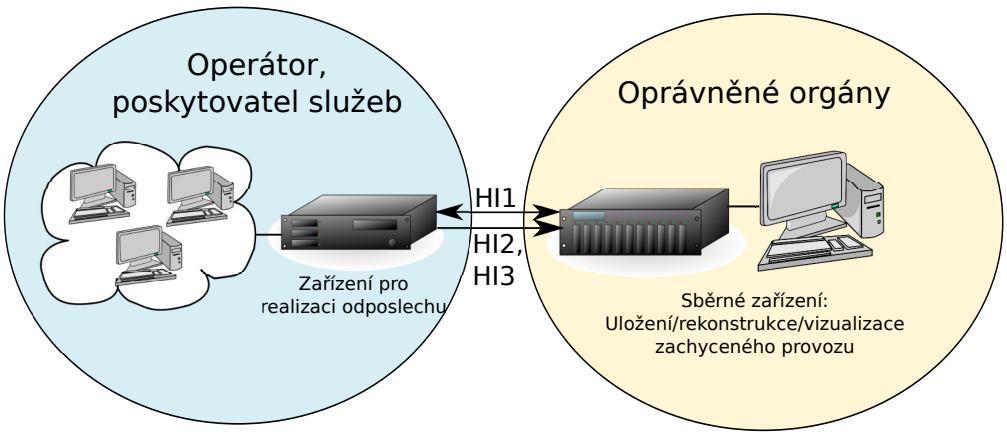
Kapitola 8 slouží programátorům využívajícím SLIS jako platformu pro tvorbu pokročilejších funkcí, či pro tvůrce modulů vytvářející nový modul pro detekci identity. Kapitola obsahuje formát a možný obsah zpráv přenášených na jednotlivých rozhraních systému.

Kapitola 9 shrnuje text této technické zprávy a podává seznam ostatních dosažených výsledků.

Kapitola 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 2.1).



Obrázek 2.1: Obecný model LIS: data jsou zachycena NWO/AP/SvP a zaslána LEA pro další analýzu

2.1 Architektura zákonných odposlechů definovaná ETSI

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 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 [1]. V takovém případě jsou zachytávány veškeré pakety určené odposlouchávanému, nebo odeslané odposlouchávaným včetně aplikačních dat a hlaviček protokolů nižších vrstev modelu ISO/OSI [35]. Odposlouchávaná data jsou zasílaná ve formě zpráv CC.

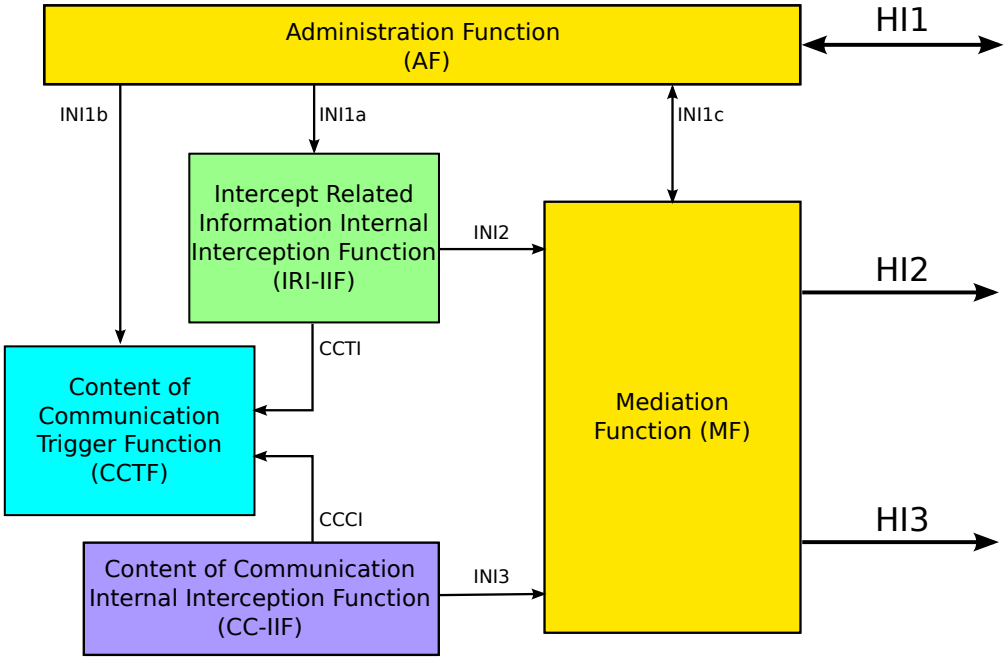
Identifikace odposlouchávaného musí být jednoznačná a neměly by být příliš velké obtíže s určením, zda má být konkrétní síťový tok předmětem odposlechu [22]. Norma nespecifikuje, které identifikátory musí být striktně podporovány, ale navrhuje některé z následujících:

- Uživatelské jméno nebo Network Access Identifier (NAI) [3]. 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 [19] referenční model architektury LIS zachycený na obrázku 2.2. ETSI definoval rozhraní mezi NWO/AP/SvP a vyšetřující LEA (*Handover Interface* – HI). [16, 21, 23, 24, 18], 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.



Obrázek 2.2: Referenční model LIS publikovaný ETSI [19]

Referenční model LIS vytvořený ETSI se skládá z pěti spolupracujících částí (bloků): *Administration Function* (AF), *Intercept Related Information – 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 2.2. 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, 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 [5] je obdobou referenčního modelu ETSI a je používaný převážně v USA. Standard J-STD-025 specifikuje následující tři druhy rozhraní [30]:

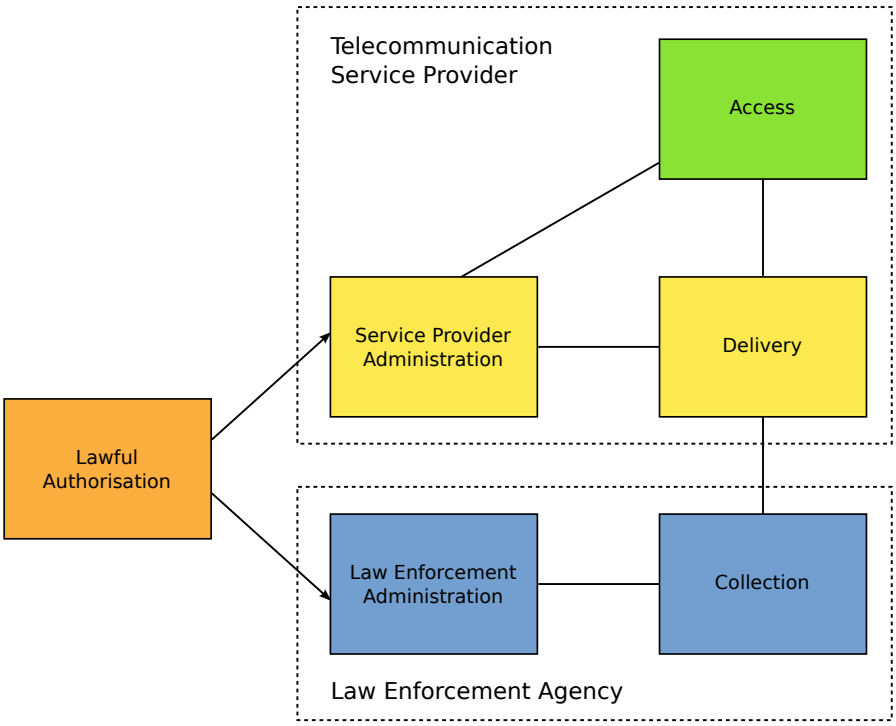
- *Surveillance Administration System* (SAS) zpřístupňuje systém vyšetřujícím orgánům.
- *Call Data Channel* (CDC) poskytuje signalizační zprávy, které obsahují informace spojené s navazováním a ukončováním komunikace.
- *Call Content Channel* (CCC) zprostředkovává úplnou kopii obsahu komunikace.

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 2.3):

- *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 správném 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.



Obrázek 2.3: Architektura LIS a spolupráce s LEA specifikovaná standardem J-STD-025 [5]

2.3 Architektura komerčních systémů pro zákonné odposlechy

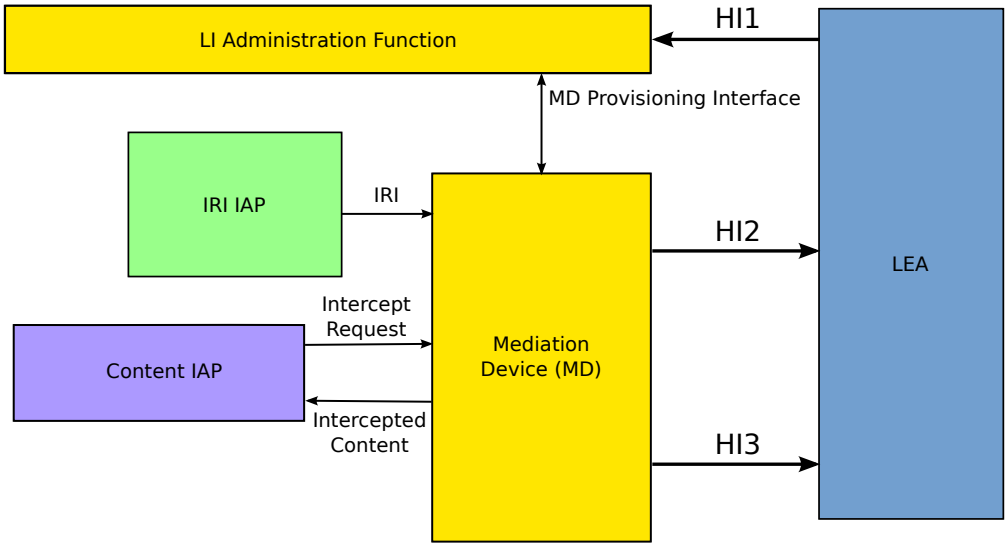
Společnost Cisco vycházela z architektur LIS publikovaných ETSI a v rámci J-STD-025 a navrhlo vlastní architekturu [6]. Rozhraní LIS směrem k LEA je velmi podobné referenční architektuře ETSI, je však konfigurovatelné tak, aby mohlo být kompatibilní i s J-STD-025. Architektura LIS společnosti Cisco se skládá z následujících částí (viz obrázek 2.4):

- *LI Administration Function* je zodpovědná za zpracování odposlechů.
- *Intercept Access Point* (IAP) je místo, ke kterému je připojeno zařízení, které je napojeno na síťovou infrastrukturu NWO/AP/SvP a je schopné získávat odposlouchávaná data. Přeneseně se může jako IAP označovat samotné zařízení provádějící odposlech dat. Rozlišujeme dva druhy IAP:

  1. *Content IAP* získává provoz na síťové vrstvě ISO/OSI modelu [35].

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.



Obrázek 2.4: Architektura LIS publikovaná společností Cisco [6]

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 [4]. 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 zemích. Dokument je však již částečně zastaralý, protože nebere v úvahu bezstavové přidělování adres protokolem IPv6 [55] 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* [37], 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. [4, 75]), že se komerční řešení nezabývají některými oblastmi síťové komunikace dostatečně do hloubky. V následujících částech této technické zprávy jsou bezpečnostní problémy současných LIS popsány.

Kapitola 3

Architektura vytvořeného systému pro zákonné odposlechy

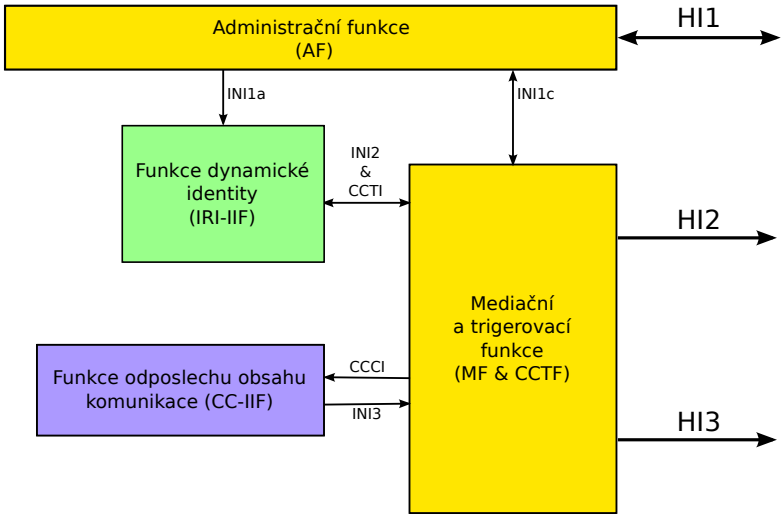
Jedním z cílů projektu *Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace* je vytvoření prototypu systému pro sběr dat pro zákonné odposlechy (Lawful Interception System – LIS), který bude schopen komunikovat s vytvořenou vysokorychlostní sondou a mikro-sondou a poskytnout data pro rekonstrukci a klasifikaci datových toků. Tato kapitola je zaměřena na popis architektury vyvíjeného LIS, která vychází z doporučení ETSI [16, 17, 21, 23, 19, 24, 22, 25] a je také inspirována architekturou LIS firmy Cisco publikovanou v RFC 3924 [6]. Navržený prototyp systému je označen jako SLIS (Sec6net Lawful Interception System)

Všimněte si, že cílem této aktivity nebylo vytvořit zcela nový LIS, který by dokázal konkurovat současným komerčním řešením, ale vytvořit zjednodušený prototyp LIS, který tvořil a tvoří základní prostředí pro: a) vývoj nových technik dynamické identifikace uživatele v prostředí IPv6 sítí, b) sběr dat a vývoj nových metod v oblasti rekonstrukce a vizualizace zachyceného provozu a c) jako základní testovací prostředí pro vývoj mikro-sondy a vysokorychlostní sondy. Z těchto důvodů jsou některé části vyvíjeného LIS zjednodušeny oproti standardu ETSI. Tato zjednodušení jsou v textu podrobněji popsána.

3.1 Blokové schéma architektury

Základní blokový diagram navrhovaného LIS je uveden na obrázku 3.1. Vstupní požadavky k odposlechu jsou přijímány skrze rozhraní HI1 a zpracovávány tzv. *Administrační funkcí* (AF). Blok AF nejdříve provádí kontrolu správného vyplnění povinných a volitelných položek – je zkontrolována unikátnost *Lawful Interception IDentifier* (LIID) [23, 24], korektnost časových údajů a přítomnost odposlouchávající agentury v systému. Pokud je vše v pořádku, je odposlech zařazen do fronty čekajících odposlechů. AF je následně zodpovědná za korektní inicializaci a ukončení odposlechu, tj. konfiguraci ostatních částí systému (skrze rozhraní INI1a 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. Veškeré požadavky na přidání, či odebrání odposlechu jsou navíc z bezpečnostních důvodů v AF zaznamenávány.

Identifikace jednotlivých uživatelů sítě (např. jejich IP adresa) se obecně může v průběhu času měnit. S ohledem na tuto skutečnost je v LIS k dispozici blok označený jako *Funkce dynamické identity* (IRI-IIF). Úlohou bloku IRI-IIF je detekovat v síťovém provozu zprávy, které



Obrázek 3.1: Architektura prototypu systému pro zákonné odposlechy

se vztahují ke změně identity uživatelů (např. protokoly DHCP, RADIUS apod.) a udržovat informace o aktuální identitě odposlouchávaných cílů. Každá změna identity odposlouchávaného cíle je pak neprodleně signalizována ostatním částem LIS (podrobnější informace o způsobu předávání zpráv jsou uvedeny v dalších kapitolách). Blok IRI-IIF dále vytváří informační zprávy (metadata) sledující začátky a konce odposlouchávaných spojení (např. uživateli byla přidělena IPv4 adresa protokolem DHCP nebo platnost adresy vypršela) a odesílá je skrze rozhraní INI2.

Blok označený jako *Funkce odposlechu obsahu komunikace* (CC-IIF) získává síťový provoz cíle odposlechu (CC) a kopíruje veškerý obsah komunikace (CC) vztahující se k některé IPv4 nebo IPv6 adrese sledovaného cíle. Vstupní požadavky na započetí, či ukončení odposlechu jsou zasílány rozhraním CCCI. Zachycená data jsou odesílána rozhraním INI3.

Centrální správa zachycených dat aktuálně probíhajících odposlechů je úlohou *Mediační funkce* (*Mediation Function* - MF). MF zpracovává jak metainformace od bloku IRI-IIF, tak i obsah zachycené komunikace od bloku CC-IIF. Oba tyto typy dat navzájem kombinuje a zasílá oprávněným orgánům skrze rozhraní HI2 resp. HI3. Z důvodu zjednodušení správy odposlechů je MF ve vyvíjeném LIS kombinovaná s trigerovací funkcí (*Content of Communication Trigger Function* - CCTF), která je zodpovědná za konfiguraci CC-IIF sond. Toto rozhodnutí bylo inspirováno architekturou LIS firmy Cisco [6].

3.2 Předpokládané nasazení systému

Systém byl tvořen od začátku pro spolupráci s mikrosondami a vysokorychlostními sondami. Mikrosondy jsou určené pro metalické sítě do rychlostí 1 Gb/s. Vysokorychlostní sondy pracují na optických sítích na rychlostech 10 Gb/s a více (do konce projektu se předpokládá vytvoření sondy pro 100 Gb/s).

Schopnosti mikrosondy byly navrženy s ohledem na zapojení (IAP) v přístupových sítích poskytovatele internetového připojení (NWO/AP). Ideální [13] je zapojení co nejblíže k odposlouchávanému, protože pak je možné:

1. Zachytit všechna data, která tento zákazník odeslal bez ohledu na jejich adresáta. Zejména je potřeba předcházet situacím, kdy si dva zákazníci stejného poskytovatele připojení (NWO/AP) vyměňují data mezi sebou, či možným přenosům části dat jinými linkami v rámci distribuční části sítě NWO/AP.

2. Předejít možným pochybnostem o pravém autorovi dat. Sítě založené na protokolu IP umožňuji podvržení [13] zdrojové adresy, která není nijak verifikována. Odposloucháváním dat co nejblíže k podezřelému je pak možné předejít možným pozdějším nejasnostem při uznávání odposlechnutých dat jako soudního důkazu.

Vysokorychlostní sondy jsou určené pro případy, kdy není možné využít IAP blízko k odposlouchávanému, např. protože přistupuje ze sítě v jiné jurisdikci, či existuje podezření prozrazení odposlechu při jeho realizaci v síti jeho NWO/AP. V takovém případě je možné vysokorychlostní sondy připojit na páteřní linky v Internetu. Je však nutné rozmístit sondy na všechna místa, přes která mohou být zájmová data směrována.

Dále uvažujeme vysokorychlostní sondy zaměřené na aplikační protokoly. Předpokládáme zapojení takových vysokorychlostních sond na hranicích sítě NWO/AP. Cílem odposlechu těchto sond jsou data směřující k nespolupracujícím poskytovatelům služeb (SvP), či ležící v jiné jurisdikci.

Dynamickou identitu uživatelů je vhodné také sledovat přímo v síti, ve které je prováděn odposlech. V takové síti se typicky nacházejí různé autentizační servery, servery pro správu adresního prostoru apod. Na klíčových přístupových linkách k těmto serverům je například možné rozmístit softwarové sondy pro sledování přidělovaných adres a autentizací a tyto informace využívat v rámci IRI-IIF.

Obrázek 3.2 ukazuje možná rozmístění sond v síti. Data zachycená jednotlivými sondami jsou ukládaná v jediném centrálním místě (centrální zařízení), které řídí probíhající odposlechy a konfiguruje jednotlivé sondy. Na obrázku jsou také znázorněny možná umístění sond pro zjišťování dynamické identity uživatelů. Na obrázku 3.2 jsou zobrazeny jednak softwarově řešené sondy umístěné buď na linkách vedoucích ke kritickým serverů spravujících danou síť (RADIUS, autentizace, DHCP, DHCPv6 aj.), či na strategických místech (autokonfigurace IPv6 [58]). Pro aplikační odposlechy byla funkcionalita spojená s dynamickou identifikací integrována do vysokorychlostních sond a předpokládá se umístění na hranicích sítě. Více informací o podporovaných protokolech pro zjišťování dynamické identity obsahuje kapitola 6.

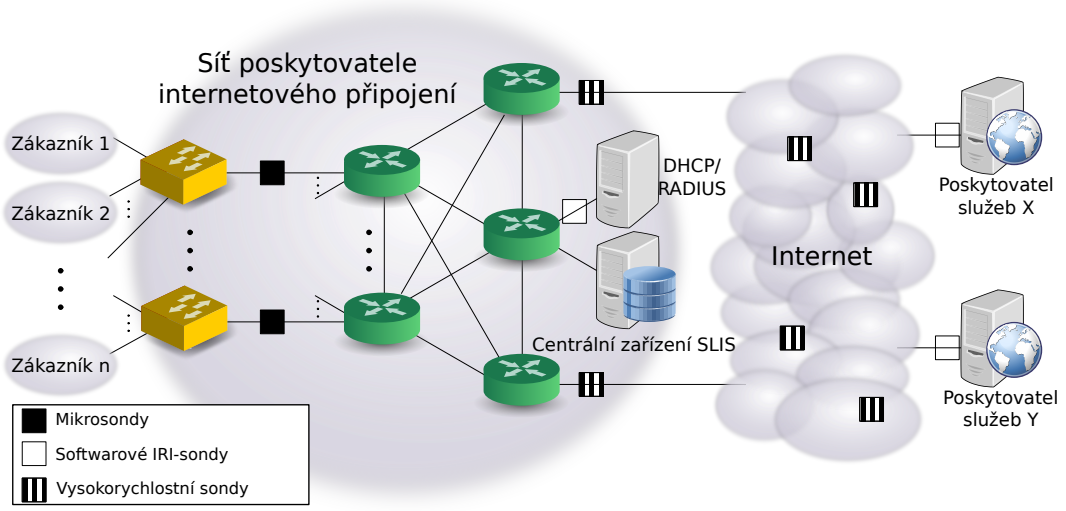
Z obrázku 3.2 je patrné, že dochází k rozdělení funkčních bloků architektury SLIS na několik fyzických zařízení:

Centrální zařízení obsahuje takové části, které je vhodné mít v systému pouze jednou. Jedná se o AF, MF&CCTF a také ústřední část, nebo-li *jádro* IRI-IIF. Jádro IRI-IIF obsahuje sdílené informace získané z jednotlivých sond IRI-IIF, např. všechny zjištěné informace o identitě v síti.

Softwarové sondy IRI-IIF se mohou v rámci jedné instance SLIS vyskytovat v libovolném počtu, mohou být dokonce integrovány v rámci centrálního zařízení. Každá sonda může být specializovaná na určité protokoly, podle toho, jaká data proudí na konkrétní sledované lince.

Mikrosondy se mohou v rámci jedné instance SLIS vyskytovat v libovolném počtu. Každá sonda umožňuje zachytávat CC z dat proudících po připojené lince.

Vysokorychlostní sondy se mohou v rámci jedné instance SLIS vyskytovat v libovolném počtu. Každá sonda umožňuje zachytávat CC, některé vysokorychlostní sondy mohou být také



Obrázek 3.2: Předpokládané nasazení vytvářeného systému pro zákonné odposlechy v síti poskytovatele internetu (NWO/AP/SvP)

vybaveny systémem na rozpoznávání identity v aplikačních protokolech, který je součástí IRI-IIF. Přichází také v úvahu integrace centrálního zařízení přímo na vysokorychlostní sondě.

Obrázek 3.3 zobrazuje možné nasazení systému, kde jsou kromě centrálního zařízení připojeny 2 softwarové sondy IRI-IIF, mikrosonda a vysokorychlostní sonda se schopností dekódování identity z aplikačních protokolů (např. SMTP, XMPP, OSCAR apod.).

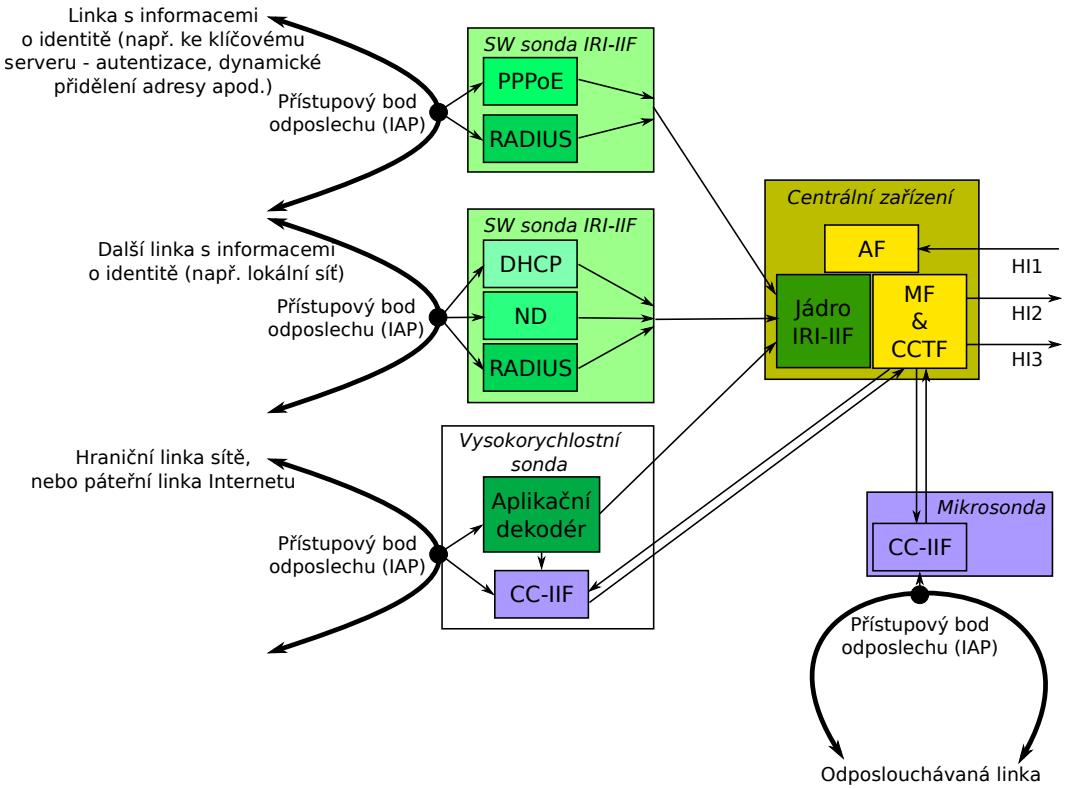
3.3 Vstupní specifikace odposlechu

Administrační funkce (AF) zpracovává požadavky od jedné, či více LEA. Požadavky jsou přijímány skrz rozhraní HI1. Norma ETSI TS 101 671 [23] říká, že rozhraní HI1 může být řešeno jak manuálně, tak automaticky (elektronicky). Z této normy ovšem také vyplývá, že přímá aktivace/deaktivace/modifikace odposlechu by měla být v kompetenci NWO/AP/SvP a tedy neměla být přímo přístupná LEA.

V současnosti se uvažuje manuální rozhraní HI1 - např. dopis poštou, který přijme pověřený pracovník NWO/AP/SvP a požadavek na aktivaci odposlechu do systému SLIS zadá pomocí webového uživatelského rozhraní, které vyobrazeno na obrázku 3.4.

Pracovník NWO/AP/SvP zadá do webového rozhraní požadavek na nový odposlech. Je-li požadavek korektní, je předán AF k dalšímu zpracování. Požadavek na nový odposlech zahrnuje následující údaje:

- *LEA* - název orgánu činného v trestním řízení, či LEA požadujícího odposlech.
- *Lawful Interception Identifier* (LIID): Řetězec alfanumerických znaků, který jednoznačně identifikuje odposlech [23]. LIID je dodáván LEA a je unikátní nejen v rámci LEA, ale

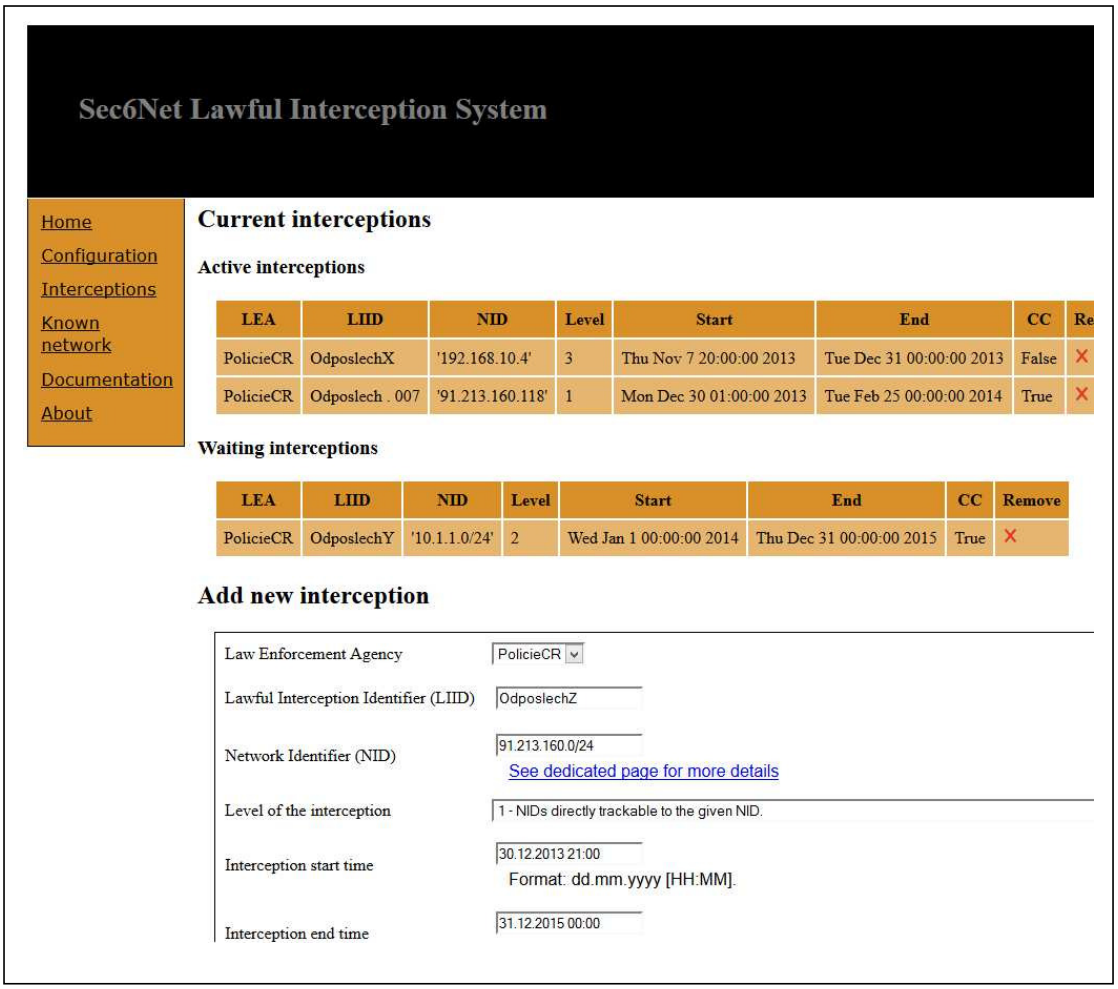


Obrázek 3.3: Rozdělení funkčních bloků SLIS na fyzická zařízení zapojená v síti.

také na mezinárodní úrovni [24] (řetězec obsahuje dvoupísmennou zkratku státu definovanou ISO 3166-1 [36]). Všechna data přenášena rozhraními HI2 a HI3 musí být tímto identifikátorem označena.

- *Network Identifier* (NID)- síťový identifikátor: Identifikátor používaný v síťových protokolech k označení účastníků komunikace nebo k označení konkrétního spojení (např. TCP). V současnosti je LIS schopen označit uživatele na základě MAC adresy, statické IPv4 a IPv6 adresy nebo obecně rozsahem adres definovaným adresou sítě a maskou. Na úrovni protokolů pro autentizaci pak dále podporuje přihlašovací jména protokolů PPP a RADIUS. U protokolů pro dynamické přidělení adresy podporuje SLIS identifikátor klienta protokolu DHCP a DUID protokolu DHCPv6. V prostředí sítí SDN je podporován specifický bod připojení do sítě specifikovaný názvem prvku a výběrem jeho portu. Na úrovni aplikačních protokolů je možné identifikovat uživatele resp. jeho komunikaci na základě emailové adresy nebo loginu některého z protokolů pro předávání zpráv v reálném čase (XMMP, IRC, OSCAR, YMSG). Dále je podporován protokol SMTP, u kterého je cíl odposlechu specifikován e-mailovou adresou. Podrobnosti o těchto protokolech a použitých identifikátorech jsou podrobně popsány v kapitole 5.

- *Úroveň*, nebo-li rozsah odposlechu - nově přidaný parametr, který vyplynul z rozhovorů se zástupci Policie ČR. Přesné znění povolení k odposlechu není dopředu zřejmé, v některých případech je možné kombinovat několik zdrojů o identitě uživatele a tím do odposlechu



Obrázek 3.4: Správa odposlechů systému SLIS přes webové uživatelské rozhraní

zahrnout např. všechny IP adresy jednoho počítače, či všechna zařízení konkrétního uživatele, v jiných případech je odposlech nutné omezit jen na konkrétní předaný identifikátor, např. IP adresu (více o této problematice pojednává sekce 3.4)

- *Datum a čas zahájení odposlechu* - okamžik, od kterého umožňuje soudní povolení získávání dat
- *Datum a čas ukončení odposlechu* - okamžik, od kterého již platné soudní povolení neumožňuje získávání dalších dat a odposlech má být ukončen.
- *CC ano/ne* - informace, zda bude zaznamenáván i obsah komunikace (CC) nebo pouze metadata o komunikaci (IRI).

Protože odposlech může být cílen nejen na některý z podporovaných síťových identifikátorů NID definovaných výše, ale také pomocí obecných identifikačních údajů jako je jméno a příjmení, rodné číslo apod., očekává se, že pověřený pracovník NWO/AP/SvP bude schopen dodanou

identifikaci cíle odposlechu (*Handover Interface 1 Identifier* – HI1ID) jednoznačně konvertovat na některý z výše uvedených identifikátor NID popsaný výše. Pověřený pracovník obvykle využije interní databáze obsahující seznam zákazníků a převede HI1ID na NID.

Zájmový obsah komunikace je sondami detekován na úrovni IP adresy (pokud je to dovoleno) a pro odposlechy aplikačních protokolů jsou podporovány datové toky používající identifikátory síťové a transportní vrstvy. Tyto identifikátory souhrnně označujeme jako $\mathrm{NID_{CC}}$:

- IPv4 nebo IPv6 adresa, či rozsah adres specifikovaný adresou sítě a maskou.
- Trojice IP adresa, port a protokol transportní vrstvy určené pro odposlech konkrétní služby běžící na serveru specifikované IP adresy.
- Pětice zdrojová/cílová IP adresa, zdrojový/cílový port a protokol transportní vrstvy pro odposlech konkrétního aplikačního spojení.

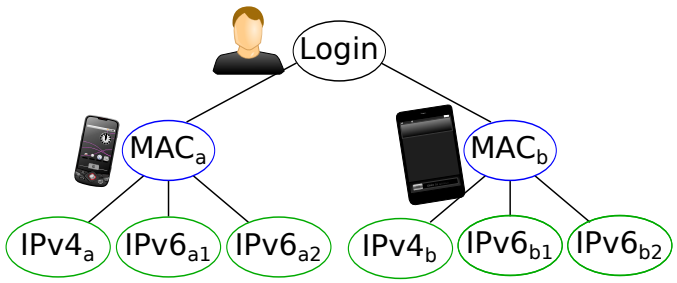
Tyto $\mathrm{NID_{CC}}$ byly vybrány s ohledem na možnost dostatečné granularizace provozu a se zaměřením na takovou identifikaci provozu, která by byla uskutečnitelná na rychlosti linky podporované mikrosondami (1 Gb/s) a vysokorychlostní sondou (v současnosti 10-40 Gb/s, v budoucnu 100 Gb/s).

Protože množina vstupních identifikátorů $\mathrm{NID_{In}}$ zahrnuje identifikátory nepatřící do $\mathrm{NID_{CC}}$, je nutné přesně zjistit spojitosti mezi jednotlivými identifikátory a odvodit odpovídající identifikátory typu $\mathrm{NID_{CC}}$. Vstupní identifikátory $\mathrm{NID_{In}}$ jsou předávány z bloku administrační funkce do bloku IRI-IIF pro sledování dynamické identity uživatele. Blok IRI-IIF sleduje identifikátory používané v síti tak, jak je popsáno v sekci 5 a kapitole 6.

3.4 Podporované úrovně odposlechu

V moderních sítích, ke kterým uživatele přistupuji protokolem IPv6, či využívají několik zařízení může konkrétní znění povolení odposlechu značně ovlivnit rozsah dat, které je možné zachytávat.

Uvažujme následující scénář: uživatel používá dvě bezdrátová zařízení (např. mobilní telefon a tablet) v síti s autentizaci prostřednictvím protokolu RADIUS. Uživatel autentizuje obě zařízení a každé zařízení získá IPv4 adresu a několik IPv6 adres (viz sekce 6.1 a 6.5. Danou situaci zachycuje obrázek 3.5.



Obrázek 3.5: Každé ze zařízení jednoho uživatele používá několik IP adres, obě zařízení však byla autentizována pomocí jediného jména (protokolem RADIUS).

Jak je vidět v uvedeném příkladě, každé zařízení získalo několik IPv6 adres. Na základě MAC adresy je možné určit, že všechny IP adresy patří ke stejnému zařízení. Navíc je možné využít

autentizace obou MAC adres pomocí protokolu RADIUS. V souvislosti s tímto příkladem, lze ze strany orgánů činných v trestním řízení očekávat různé druhy požadavků na rozsah odposlechu:

1. *Odposlech v rozsahu síťové adresy* - Předmětem zájmu je komunikace spojena pouze s konkrétní IP adresou (např. $IPv4_a$). Přestože lze z grafu dohledat informaci, že uživatel používá více IP adres, budou se zachytávat pouze pakety s IP adresou $IPv4_a$.

2. *Odposlech v rozsahu rozhraní nebo počítače* - Předmětem zájmu je veškerá komunikace v rámci jednoho síťového rozhraní nebo počítače. Oprávněné orgány mohou svůj požadavek na odposlech zadat formou NIDu na konkrétní síťovou adresu (např. $IPv4_a$) nebo adresu rozhraní (např. MAC adresy M_a). Jádro IRI-IIF je schopen v grafu dohledat všechny IP adresy související se stejným rozhraním (tj. adresy $IPv4_a$, $IPv6_{a1}$ a $IPv6_{a2}$). Přes úroveň rozhraní však nezasahuje (tj. adresy $IPv4_b$, $IPv6_{b1}$ a $IPv6_{b2}$ nejsou předmětem odposlechu).

3. *Odposlech v rozsahu uživatele* - Předmětem zájmu je veškerá komunikace daného uživatele. Oprávněné orgány mohou svůj požadavek na odposlech zadat formou NIDu na konkrétní síťovou adresu (např. $IPv4_a$) nebo adresu rozhraní (např. MAC adresy M_a) nebo na jiný identifikátor (např. PPP login PL_a). Jádro IRI-IIF je schopen v grafu dohledat všechny IP adresy související se stejným uživatelem (tj. adresy $IPv4_a$, $IPv6_{a1}$, $IPv6_{a2}$, ale i $IPv4_b$ a $IPv6_{b1}$ a $IPv6_{b2}$).

Formálně je postup stanovení odposlechu popsán v sekci 5.

3.5 Výstupy související s uskutečněným odposlechem

Jak již bylo napsáno výše, implementovaný LIS je určen především pro testovací a vědecké účely a detailní implementace rozhraní [23] nebyla předmětem aktivit skupiny.

Výstupy definované normami je možné rozdělit do dvou skupin: zjištění metadat o odposlechu a zachycení samotného provozu přeneseného podezřelým. V prvním případě jde o zprávy IRI přenášené rozhraním HI2, ve druhém o data CC přenášená rozhraním HI3.

Konkrétní komunikace je vždy identifikována pomocí *Communication Identifier* (CID) [23, 24]. CID se skládá z:

- unikátního ID operátora přiděleného LEA,
- NID, kterého se odposlech týká,
- *Communications Identity Number* (CIN), které identifikuje sezení, nebo komunikaci v rámci jednoho odposlechu, který je identifikován pomocí LIID, v případě SLIS se vždy jedná o číslo,
- *Delivery Country Code* (DCC) Dvoupísmenné označení země dle ISO 3166-1[36], kde se nachází MF.

Výstupem rozhraní HI2 a HI3 jsou soubory, které jsou ukládány do podadresářů v adresáři úložiště. Adresář úložiště je specifikován v souboru `mf.ini`. Názvy jednotlivých podadresářů odpovídají názvům příslušných LEA.

Rozhraní HI2 slouží k předávání metadat o detekovaných síťových spojeních. Z důvodu zjednodušení nejsou ve vyvíjeném LIS tato metadata odesílána oprávněným orgánům přímo ve formě proudového zpracování, ale jsou ukládaná do textových souborů. Každý textový soubor obsahuje metadata pro jeden odposlech identifikovaný pomocí LIID. Název souboru je ve formátu

`LIID.hi2`. Na každém řádku souboru s meta informacemi je uložena jedna událost. U každé události je evidováno následující:

- čas výskytu,
- typ (*IRI begin*, *end*, *report*, nebo *continue*),
- CID [23, 24],
- typ dynamické detekce (identifikátor IRI modulu),
- popis události,
- seznam souvisejících NIDů.

Rozhraní HI3 slouží pro přeposílání obsahu komunikace podezřelých osob. Z důvodu zjednodušení se kopie zachycených dat ukládá do souboru ve formátu PCAP [1]. Pro usnadnění případné další rekonstrukce zachycených dat je v rámci jednoho LIID vytvořen separátní soubor pro každou komunikaci. Název souboru je ve formátu `LIID_CIN.pcap`.

Kapitola 4

Scénář odposlechu

LEA požádá soud o odposlech specifického uživatele identifikovaného pomocí HI1ID. Žádost o odposlech musí mít definované časové období, ve kterém je možné data podezřelého odposlouchávat, tj. počátek odposlechu (t_s) a konec odposlechu (t_e). Součástí specifikace odposlechu je i požadavek, zda má LIS zachytávat pouze metainformace o spojeních realizovaných podezřelou osobou (IRI), nebo zda bude povolen sběr veškeré komunikace uživatele (IRI+CC). Po schválení odposlechu jsou jeho parametry předány pověřenému pracovníku NWO/AP/SvP, ke kterému je podezřelý připojen.

Pověřený pracovník NWO/AP/SvP je taková osoba, která má oprávnění ke konfiguraci LIS. Po přijetí požadavku k odposlechu zkontroluje platnost odposlechu. S využitím interní databáze převede dodaný HI1ID na NID_{In} a provede zadání odposlechu do LIS. Vytvořený SLIS umožňuje zadávat odposlechy skrze webové rozhraní nebo pomocí specializovaných nástrojů z příkazové řádky.

4.1 Inicializace odposlechu

Abychom si blíže ukázali činnost SLIS, uvažujme příklad, ve kterém LEA_1 zažádá o IRI zprávy (označené LIID = X) týkající se konkrétního uživatele identifikovaného jménem a adresou. Uvažujme, že pověřený zaměstnanec na základě databáze uživatelů zjistí, že tento uživatel se do sítě připojuje pomocí zařízení s MAC adresou *00:11:22:aa:bb:cc*. Oproti tomu LEA_2 má zájem o odposlech veškeré komunikace počítače s konkrétní IPv4 adresou *10.0.0.1* a přeje si zachycená data označovat LIID = Y. Pověřený zaměstnanec nakonfiguruje SLIS na odposlech adresy *10.0.0.1*. Parametry obou odposlechů, které zaměstnanec zadal do SLIS jsou uvedeny v tabulce 4.1.

AF v sobě uchovává dvě prioritní fronty. První frontu tvoří čekající odposlechy, kde nejvyšší prioritu má odposlech s nejbližším časem zahájení. Druhou frontou je fronta aktivních odposlechů, kde nejvyšší prioritu má odposlech s nejbližším časem ukončení. Při přijetí nového požadavku porovná AF jeho časové údaje s aktuálním časem a zařadí jej do jedné z front dle následujících

LEA	LIID	NID_{In}	t_s	t_e	typ
LEA_1	X	MAC: 00:11:22:aa:bb:cc	1.1.2011	1.1.2012	IRI
LEA_2	Y	IPv4: 10.0.0.1	2.1.2011	15.5.2012	IRI+CC

Tabulka 4.1: Příklad konfigurace SLIS, kterou zadává pověřený pracovník NWO/AP/SvP

LIID	t_s	t_e	typ	LEA	Soubor pro HI2	CIN	Soubor pro HI3
X	1.1.2011	1.1.2012	IRI	FBI	X.hi2	1	X1.pcap
						2	X2.pcap
						3	X3.pcap
Y	2.1.2011	15.5.2012	IRI+CC	BIS	Y.hi2	5	Y5.pcap
						6	Y6.pcap

Tabulka 4.2: Příklad obsahu tabulky LIID uvnitř bloku MF&CCTF

kritérií:

1. Pokud ještě nenastala doba, od které je platné soudní povolení na získávání dat. Systém tedy vloží požadavek do speciální fronty čekajících odposlechů.
2. Pokud doba platnosti soudního povolení již nastala. Odposlech je ihned aktivován a vložen do fronty aktivních odposlechů.

AF konfiguruje IRI-IIF a MF&CCTF 10 sekund před stanoveným začátkem odposlechu. Ke konfiguraci dalších částí LIS dochází co nejpozději, abychom minimalizovali dobu, po kterou je možné získat informace o odposlechu ze sond SLIS. Na druhou stranu probíhá konfigurace s krátkým časovým předstihem, aby se zamezilo ztrátě dat přenášených těsně po začátku odposlechu, ke které by došlo v průběhu konfigurace SLIS.

Podobně při ukončení odposlechu čeká AF 10 sekund, než odebere odposlech ze zbytku systému. Toto zpoždění bylo zavedeno, aby bylo zaručeno bezproblémové zpracování dat zachycených těsně před povoleným koncem odposlechu. Blok MF&CCTF je zodpovědný za odstranění dat nasbíraných sondami mimo povolené rozmezí odposlechu.

Zprávy posílané skrz rozhraní INI1a a INI1c mají specifický formát. Rozhraním INI1a se posílají výše uvedené údaje z požadavku na odposlech a navíc je k nim přidán CID s dosud neinicializovanou hodnotou CIN. Skrz rozhraní INI1c se posílají pouze nejnutnější informace: LIID, čas začátku a konce odposlechu a informace, zda bude zaznamenáván také obsah komunikace.

4.2 Aktivace odposlechu

AF posílá MF&CCTF pro každý odposlech jeho LIID, období platnosti (t_s a t_e) a informaci, zda odposlech povoluje odchycení celého obsahu komunikace. MF&CCTF si tyto informace ukládá do tzv. *tabulky LIID*. Podle této tabulky MF&CCTF ukládá odchycená data do správných souborů a zahazuje data nasbíraná mimo interval platnosti odposlechů. Metadata související s odposlechem jsou ukládána do textových souboru s příponou `.hi2`. Tento soubor je jeden pro každý odposlech. Samotný obsah komunikace (odchycené pakety) je pro daný odposlech ukládán separátně do binárních `.pcap` souborů, přičemž každý tento soubor odpovídá jedné komunikaci (jednomu CINu). Pro jeden odposlech tedy existuje tolik `.pcap` souborů, kolik pro tento odposlech existuje CINů. Dále AF posílá MF&CCTF posílá identifikaci LEA ve formě unikátního řetězce, který je pro každou spolupracující LEA umístěn v konfiguraci systému, tzv. *Law Enforcement Agency Identifier* (LEAID). MF&CCTF tentp řetězec používá pro ukládání dat IRI a CC. Příklad obsahu tabulky LIID je znázorněn v tabulce 4.2.

Současně s konfiguraci bloku MF&CCTF posílá AF do bloku IRI-IIF pro každý odposlech jeho LIID a šablonu pro CID (CID, ve kterém není vyplněn CIN). Tabulka 4.3 ukazuje konfigurační data, která jsou poslána pro odposlechy *X* a *Y*.

LIID	Šablona pro CID (LEA, NID_{In}, CIN, DCC)
X	(LEA_1, MAC: 00:11:22:aa:bb:cc, -, CZ)
Y	(LEA_2, IPv4: 10.0.0.1/32, -, CZ)

Tabulka 4.3: Příklad konfigurace zasílané z AF do IRI-IIF

4.3 Minimalizace dat přenášených ze síťové sondy do centrální části SLISu

Jeden podezřelý může být předmětem odposlechů několika různých LEA. Aby nedocházelo k duplicitnímu přeposílání stejných dat z CC-IIF do MF&CCTF (označených pouze různými LIID a CID), přiřazuje vyvíjený LIS každému odposlechu nově zavedený identifikátor SID, který zastupuje množinu odposlechů (identifikovaných pomocí LIID). Díky tomu je možné zachycená data označit příslušným identifikátorem SID a jejich kopírování a rozesílání různým LEA realizovat až v rámci bloku MF&CCTF. Komunikace mezi blokem CC-IIF a MF&CCTF se tak minimalizuje a vyžaduje jen nezbytně nutnou šířku pásma.

System Identifier (SID) je 32-bitový celočíselný identifikátor množiny odposlechů vztažených k rozsahu odposlouchávaných NID_{CC}. Mapování SID na množinu LIID se uchovává v blokuMF&CCTF, který také zajišťuje přidělování nových SID. Výhodou použití SID je jeho pevná velikost, nicméně při použití v mikrosondě se ukázalo, že pro označení zachycených dat není možné použít 32-bitové číslo. Vzhledem k požadavkům na schopnosti LIS dodaných Policií ČR na počet souběžných odposlechů a nebezpečí využití mnoha adres [55] souběžně nakonfigurovaných na jednom rozhraní jsme se rozhodli ponechat tento identifikátor ve 32-bitové verzi a pro mikrosondu vytvořit 16-bitový identifikátor. Při praktickém nasazení se v krajních případech předpokládá distribuce zachytávaných adres mezi několik sond.

Reason Identifier (RID) je 16-bitový celočíselný identifikátor využívaný sondami CC-IIF. Mimo snížení paměťové náročnosti původního identifikátoru SID z 32 bitů na 16 bitů umožňuje RID každé sondě CC-IIF uplatnění vlastní politiky pro přiřazení identifikátorů zadaným odposlechům (např. na základě umístění pravidla odposlechu ve filtrační tabulce apod.). Mechanismus mapování dvojice (ID sondy, RID) $\rightarrow$ SID je pak umístěn v bloku MF&CCTF, opět z důvodů snížení paměťově náročnosti na straně sond CC-IIF.

Jelikož se případná duplikace zachycených dat a jejich zasílání jednotlivým LEA provádí až MF&CCTF, je v témž bloku také realizován samotný algoritmus vytvářející relaci přiřazující LIID k SID a uložena tzv. *tabulka SID*, která tuto relaci reprezentuje. Pokaždé, když je do systému přidán nový požadavek na odposlech statické IP adresy nebo dojde k dynamické změně IP adresy odposlouchávaného cíle (např. skrze protokol DHCP), může obecně dojít ke změně relace přiřazující LIID a SID. Blok IRI-IIF, který tyto události sleduje, informuje blok MF&CCTF a ten upraví odpovídajícím způsobem relaci přiřazující LIID a SID. Podrobně je tento princip mapování SID na LIID popsán níže v této kapitole.

Postup získání identifikátoru RID odpovídajícího danému identifikátoru SID probíhá v následujících krocích:

1. Blok MF&CCTF zašle jedné nebo více CC-IIF sondám požadavek o přidání nového filtračního pravidla a doplní jej příslušným identifikátorem SID.
2. Každá CC-IIF sonda si vloží nové pravidlo do své filtrační tabulky a vrátí bloku MF&CCTF odpovídající RID identifikátor (obvykle index do filtrační tabulky).
3. Blok MF&CCTF si uloží relaci (ID sondy, RID) $\rightarrow$ SID.

SID	NID_{CC}	LIID	CID
1	IPv4: 10.0.0.1/32	X	(LEA_1, MAC: 00:11:22:aa:bb:cc, 1, CZ)
		Y	(LEA_2, IPv4: 10.0.0.1/32, 5, CZ)
2	IPv6: 2001:db8::5/128	X	(LEA_1, MAC: 00:11:22:aa:bb:cc, 2, CZ)

Tabulka 4.4: Příklad obsahu tabulky SID uvnitř bloku MF&CCTF. Tabulka vyjadřuje vztah mezi SID a odpovídajícím NID_{CC}. Pro každý SID je navíc v tabulce uložena množina LIID a pro každý LIID odpovídající CID, který byl konkrétní komunikaci přiřazen.

Sondy CC-IIF následně zachycené pakety označují identifikátory RID (na místo identifikátorů SID). Úlohou bloku MF&CCTF je pak tyto identifikátory RID přeložit na identifikátory SID, identifikátory SID přeložit na seznam příslušejících identifikátorů LIID a uložit zachycené pakety do příslušných souborů ve formátu PCAP.

Pro názornou ukázku uvedeného principu uvažujme následující posloupnost událostí, která by mohla nastat v případě našeho demonstračního příkladu:

1. Do SLIS jsou vloženy dva požadavky na odposlech X a Y definované výše.
2. Dne 1.1.2011 zahájí AF odposlech požadavku X a předá příslušné zprávy blokům IRI-IIF a MF&CCTF.
3. Sledováním protokolu DHCP detekuje blok IRI-IIF, že IPv4 adresa *10.0.0.1* byla přiřazena počítači s MAC adresou *00:11:22:aa:bb:cc*. Dále bylo zjištěno, že počítač začal používat IPv6 adresu *2001:db8::5* (např. sledováním protokolu DHCPv6).
4. IRI-IIF informuje MF&CCTF, že v odposlechu s LIID = X došlo k přiřazení IPv4 adresy *10.0.0.1* a že bude tato komunikace označována CID = *(LEA_1, MAC: 00:11:22:aa:bb:cc, 1, CZ)*.
5. MF&CCTF přiřadí pro tento odposlech SID_1.
6. Dále IRI-IIF ohlásí MF&CCTF, že v odposlechu s LIID = X došlo k přiřazení IPv6 adresy *2001:db8::5* a že bude používán CID = *(LEA_1, MAC: 00:11:22:aa:bb:cc, 2, CZ)*.
7. Jelikož nedochází ke shodě s některou z již existujících IP adres v tabulce SID, přiřadí MF&CCTF tomuto odposlechu SID_2.
8. Dne 2.1.2011 zahájí AF odposlech požadavku Y a předá příslušné zprávy blokům IRI-IIF a MF&CCTF.
9. IRI-IIF může okamžitě ohlásit odposlech IPv4 adresy *10.0.0.1* s CID např. *(LEA_2, IPv4: 10.0.0.1/32, 5, CZ)*.
10. Jelikož dochází ke shodě s již odposlouchávanou IPv4 adresou z požadavku X, přidělí MF&CCTF pro tento odposlech SID_1.

Obsah tabulky MF&CCTF bloku vyjadřující mapování identifikátorů LIID na SID po provedení výše uvedených kroků je znázorněn v tabulce 4.4. Dále transformace identifikátorů NID_{In} na NID_{CC} prováděné uvnitř bloku IRI-IIF jsou uvedeny v tabulce 4.5.

Pro každý přijatý paket MF&CCTF zjistí ID sondy, která jej zaslala a dvojici (jeden pro každý směr) RIDů, kterou je paket označen. Pro oba tyto RIDy provede MF&CCTF následující operace:

LIID	NID_{In}	NID_{CC}	CIN
X	MAC: 00:11:22:aa:bb:cc	IPv4: 10.0.0.1	1
		IPv6: 2001:db8::5	2
Y	IPv4: 10.0.0.1	IPv4: 10.0.0.1	5

Tabulka 4.5: Příklad transformace NID_{In} na NID_{CC} uvnitř IRI-IIF

ID sondy	RID	Soubor pro HI3	t_s	t_e
1	3	A4.pcap	1. 1. 2015	10. 4. 2016
		A6.pcap	1. 1. 2015	10. 4. 2016
		B7.pcap	1. 2. 2015	7. 3. 2015
1	5	C1.pcap	13. 5. 2013	1. 2. 2017
		C3.pcap	13. 5. 2013	1. 2. 2017
2	8	A6.pcap	1. 1. 2015	10. 4. 2016
		D3.pcap	5. 9. 2015	31. 12. 2015
		D4.pcap	5. 9. 2015	31. 12. 2015

Tabulka 4.6: Příklad dodatečné rychlé vyhledávací tabulky uvnitř MF&CCTF

- Na základě výše zmíněné relace pro danou dvojici (ID sondy, RID) určí SID.
- Pro každý SID na základě tabulky 4.4 určí množinu dvojic (LIID, CIN).
- Ke každé dvojici z této množiny určí na základě tabulky 4.2 množinu trojic (soubor pro HI3, t_s, t_e).
- Pokud čas zachycení paketu spadá do intervalu (t_s, t_e), zapíše paket do PCAP souboru pro HI3.

Protože tato činnost je výpočetně náročná, bylo provedeno několik optimalizací. Aby nebylo nutné provádět pro každý paket všechny výše uvedené operace, byl v rámci MF&CCTF implementován zrychlený překlad, který pro každou příchozí dvojici (ID sondy, RID) vrátí přímo množinu trojic (soubor pro HI3, t_s, t_e) bez dalších mezikroků. Pro tuto operaci se využívá dodatečné rychlé vyhledávací tabulky, kterou je potřeba udržovat s každou změnou v některé z odstíněných tabulek. Její příklad ukazuje tabulka 4.6.

Další provedenou optimalizací je paralelizace tohoto řešení. Pro každou připojenou CC sondu (mikrosondu, či vysokorychlostní sondu) existuje samostatné obslužné vlákno, které zpracovává pakety přijaté na rozhraní INI3 právě od této sondy. Ačkoli jednotlivá vlákna potřebují přistupovat ke sdíleným zdrojům (interním tabulkám v MF&CCTF, apod.), díky použitému synchronizačnímu mechanismu mohou časově nejnáročnější operace probíhat paralelně. Mezi tyto patří čtení přijatého paketu a zápis do výstupního souboru.

V jeden okamžik mohou v MF&CCTF různá vlákna zpracovávat různé pakety. Ke každému paketu existuje množina výstupních souborů, do kterých má být zapsán. Tyto množiny se sice mohou překrývat, ale implementovaný mechanismus tento problém efektivně řeší. Vlákno nejprve paket zapíše do těch souborů, o které se jiné vlákno neuchází a poté se postupně střídá s ostatními v zápisu do konfliktních souborů.

4.4 Zobecnění algoritmu mapování LIID na SID pro rozsahy IP adres

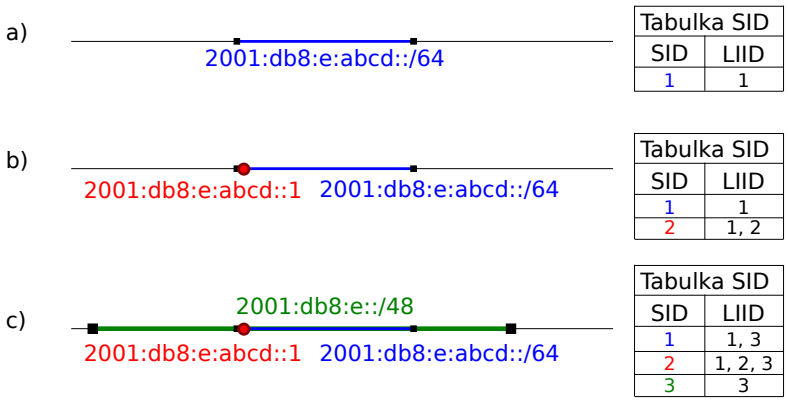
V předchozím příkladě jsme demonstrovali použití identifikátorů SID v okamžiku, kdy je sledovaný cíl předmětem různých požadavků na odposlech z různých LEA. Pro jednoduchost jsme uvažovali situaci, kdy dochází k překryvu konkrétních IP adres. Při reálném nasazení LIS však může docházet i k situacím, kdy se překrývají nejen jednotlivé adresy, ale i rozsahy IP adres. Může se tak například stát, že jeden požadavek na odposlech sleduje rozsah IP adres, který je podrozsahem IP adres jiného požadavku. V následující části textu je proto relace mapování LIID na SID zobecněna na rozsahy a následně demonstrována na příkladě.

Při požadavku ze strany IRI-IIF na odposlech nového rozsahu adres R_N se kontroluje, zda již v tabulce SID neexistuje odpovídající větší nebo menší rozsah, a podle potřeby se vygeneruje nový SID a upraví informace uložené v tabulce SID. Konkrétně se přiřazení SID řídí následujícími pravidly:

- Pokud není žádná adresa z rozsahu adres R_N odposlouchávána, pak je vygenerován nový SID a odposlech je uložen do tabulky SID.
- Pokud je R_N vlastní podmnožinou jiného rozsahu adres, např. rozsahu R_V (tzn. R_N obsahuje nějakou část adres obsažených v R_V, ale ne všechny), pak se vygeneruje nový SID a do tabulky SID se k nově přidávanému odposlechu uloží i odposlechy vztahující se k R_V.
- Pokud se již odposlouchává stejný rozsah adres, pak se pouze nový odposlech uloží do tabulky SID ke stávajícím odposlechům pro daný rozsah a všechny podrozsahy adres.
- Při existenci odposlechů pro menší rozsahy adres se vytvoří nový SID pro nově vkládaný rozsah a zároveň se informace o nově přidávaném odposlechu uloží i ke všem odposlouchávaným podrozsahům adres.

LIS podporuje pouze rozsahy specifikované maskou, tzn. pro všechny platné rozsahy adres A, B platí $A \cap B = \emptyset \vee A \subseteq B \vee B \subseteq A$ a nemůže tedy nastat jiný případ než zmíněné. Mechanismus přidělování SID a správy *tabulky SID* je demonstrován na následujícím příkladu:

1. Předpokládejme, že je v systému nejdříve aktivován odposlech $LIID_1$ vztahující se na adresový rozsah *2001:db8:e:abcd::/64* a MF&CCTF přidělí např. SID_1. MF&CCTF uloží do tabulky SID informaci, že data označená SID_1 se vztahují k odposlechu $LIID_1$. Situaci znázorňuje část a) obrázku 4.1.
2. V případě, že je následně aktivován odposlech $LIID_2$ vztahující se na jedinou IPv6 adresu *2001:db8:e:abcd::1*, pak je přidělen nový SID. Předpokládejme, že má nově přidělený SID pro odposlechnutá data vztahující se k této IPv6 adrese hodnotu SID_2. Pak MF&CCTF uloží do tabulky SID, že data označená SID_2 se vztahují k odposlechům $LIID_1$ a $LIID_2$. Změna je zachycena v části b) obrázku 4.1.
3. Pokud by byl následně aktivován odposlech $LIID_3$ vztahující se na celý rozsah *2001:db8:e::/48* a uvažujme, že pro odposlechnutá data vztahující se k tomuto IPv6 rozsahu byl přidělen SID_3, pak MF&CCTF uloží do tabulky SID, že data označená SID_3 se vztahují k odposlechu $LIID_3$. MF&CCTF dále doplní, že data označená SID_1 i SID_2 se vztahují také k odposlechu $LIID_3$. Tento stav je zachycen v části c) obrázku 4.1.



Obrázek 4.1: Příklad postupného přidávání odposlechů a vytváření vazeb mezi rozsahy IPv6 adres, SID a LIID

4.5 Zobecnění algoritmu mapování LIID na SID pro spojení transportní vrstvy

Mechanismy mapování LIID identifikátoru na SID popsané v předchozích kapitolách mohou být dále zobecněny na jednotlivá spojení definovaní skrze trojici (IP adresa, port a protokol transportní vrstvy) nebo pětici (zdrojová/cílová IP adresa, zdrojový/cílový port a protokol transportní vrstvy). Pro jednoduchost uvažujme v následujícím textu pouze použití transportního protokolu TCP.

Zaveďme nyní binární relaci *in* tak, že A *in* B platí, pokud je splněna právě jedna z následujících situací:

- A i B jsou totožné NID (rozsah IP adres, 3-jice, 5-tice).
- A představuje IP adresu. B představuje rozsah IP adres. A patří do rozsahu B.
- A představuje rozsah IP adres. B představuje rozsah IP adres. A je podrozsahem B.
- A představuje 3-jici (IP_X, port, TCP). B představuje rozsah IP adres. Platí IP_X *in* B.
- A představuje 5-tici (IP_X, port_X, IP_Y, port_Y, TCP). B představuje rozsah IP adres. Platí IP_X *in* B nebo IP_Y *in* B.
- A představuje 5-tici (IP_X, port_X, IP_Y, port_Y, TCP). B je 3-jice (IP_Z, port_Z, TCP). Platí (IP_X *in* $\mathrm{IP}_Z \wedge \mathrm{port}_X = \mathrm{port}_Z$) $\vee$ (IP_Y *in* $\mathrm{IP}_Z \wedge \mathrm{port}_Y = \mathrm{port}_Z$)

Všimněte si prosím, že relace *in* je reflexivní a tranzitivní, avšak není symetrická. Dokud jsme neuvažovali 3-jice a 5-tice, platilo, že pokud A *in* B a zároveň A *in* C, musí platit buď B *in* C nebo C *in* B. Pokud je ale A např. 5-tice představující TCP spojení a B a C např. IP adresy, výše uvedený výrok platit nemusí.

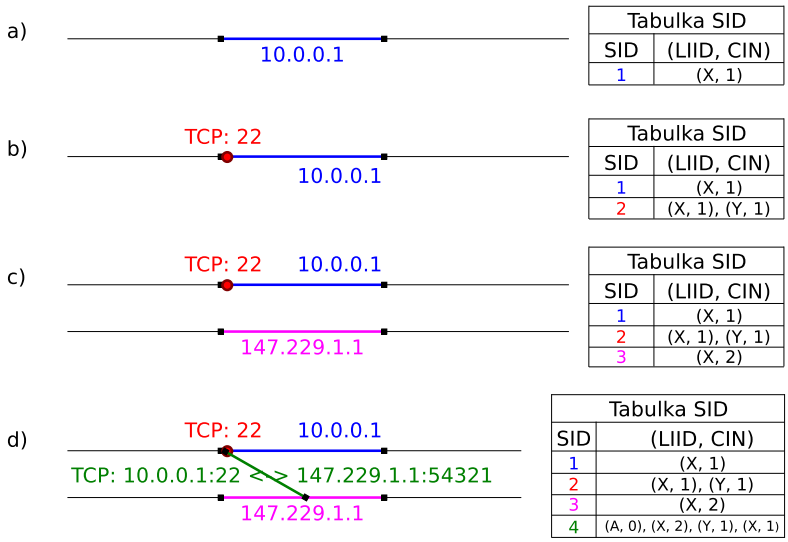
Vysvětleme nyní nově zavedený algoritmus přidělování SID zobecněný pro podporu 3-jic a 5-tic. Blok IRI-IIF oznámí nový odposlech s LIID_N cílený na NID_N. Pro NIDCC_N v rámci s LIID_N odposlechu přiřadil blok IRI-IIF CIN_N. Postupujeme následovně:

1. Pokud neexistuje žádný odposlech cílený na NID, kde NID *in* NID_N nebo NID_N *in* NID, pak je vygenerován nový SID: SID_N. K SID_N se v tabulce přiřadí dvojice (LIID_N, CIN_N).

2. Pokud existuje nějaký odposlech cílený na NID_V, takový, že NID_N *in* NID_V, přičemž NID_N a NID_V nejsou totožné, pak je vygenerován nový SID: SID_N. K SID_N se v tabulce přiřadí dvojice (LIID_N, CIN_N). K SID_N se ovšem také přiřadí všechny dvojice, které představují odposlech cílený na NID_V takový, že NID_N *in* NID_V.

3. Pokud již existuje odposlech cílený na NID, který je totožný s NID_N, pak nevytváříme nový SID. Dvojice (LIID_N, CIN_N) je přidána ke všem SID, ke kterým patří odposlechy cílené na NID_M takové, že NID_M *in* NID_N.

4. Pokud existuje odposlech na NID_M takový, že NID_M *in* NID_N a zároveň NID_M a NID_N nejsou totožné, pak je vygenerován nový SID: SID_N. K SID_N se v tabulce přiřadí dvojice (LIID_N, CIN_N). Dvojice (LIID_N, CIN_N) se však přidá také ke všem SID, ke kterým patří odposlechy cílené NID_M takové, že NID_M *in* NID_N.



Obrázek 4.2: Ukázka mapování LIID na SID pro rozsahy aplikační protokoly

Uvedený postup nemusí být na první pohled příliš jasný, proto jej demonstrujme na příkladu. Ukázkový postup je znázorněn na obrázku 4.2. Uvažujeme následující kroky:

1. Byl přidán odposlech s LIID X, jehož cílem je adresa *10.0.0.1*. Uvažujme, že blok IRI-IIF této adrese v rámci LIID X přidělil CIN 1. MF&CCTF v *tabulce SID* vytvoří nový záznam: SID 1 a do tabulky uloží, že data označená SID 1 se budou vztahovat k LIID X (pro CIN 1), jak je znázorněno v části a) obrázku 4.2.

2. Poté byl přidán odposlech s LIID Y, jehož cílem je 3-jice (*10.0.0.1*, 22, TCP). Uvažujme, že blok IRI-IIF této 3-jici v rámci LIID Y přidělil CIN 1. Blok MF & CCTF tedy vytvoří nový SID 2. Do *tabulky SID* uloží, že data označená SID 2 se budou vztahovat k odposlechu s LIID X (pro CIN 1) a také k odposlechu s LIID Y (pro CIN 1). Situaci znázorňuje část b) obrázku 4.2.

3. Blok IRI-IIF detekoval, že k odposlechu s LIID X nově patří také NID_{CC} v podpobě adresy *147.229.1.1*. Uvažujme, že blok IRI-IIF této adrese v rámci LIID X přidělil CIN 2. Blok MF & CCTF tedy vytvoří nový SID 3. Do */tabulky SID* uloží, že data označená SID 3 se budou vztahovat k odposlechu s LIID X (pro CIN 2). Protože neexistuje žádný NID, který by byl v relaci *in* s nově přidanou adresou *147.229.1.1*, nebude provedena žádná další činnost. Situaci znázorňuje část c) obrázku 4.2.

4. Nakonec byl přidán nový odposlech s LIID A, jehož cílem je 5-tice (*10.0.0.1*, 22, *147.229.1.1*, 54321, TCP). Uvažujme, že blok IRI-IIF této 5-tici v rámci LIID A přidělil CIN 0. Blok MF&CCTF v *tabulce SID* vytvoří nový záznam: SID 4 a do tabulky uloží, že data označená SID 4 se budou vztahovat k LIID A (pro CIN 0). Zároveň však blok MF&CCTF přidá k SID 4 všechny odposlechy cílené na takové NID, pro které platí (*10.0.0.1*, 22, *147.229.1.1*, 54321, TCP) *in* NID. Do *tabulky SID* tedy blok MF&CCTF uloží informaci, že data označená SID 4 se budou vztahovat také k LIID X (pro CIN 1), k LIID X (pro CIN 2) a k LIID Y (pro CIN 1). Situaci znázorňuje část d) obrázku 4.2.

4.6 Zachycení odposlouchávaných dat

Pokud sonda CC-IIF zachytí paket vztahující se k některé sledované IP adrese, či vztahující se k některému sledovanému spojení, je potřeba zjistit, jakým RID mají být data označena tak, aby mohl být RID v rámci MF&CCTF převeden na SID. Postupuje se podle následujících pravidel:

- Pokud se jedná o konkrétní sledované spojení (pětice), pak se použije RID pro tuto pětici.
- Jestliže spojení neodpovídá žádné pětici, ale některé z trojic, pak je použit RID vztahující se k této trojici.
- Jinak, pokud se odposlouchává konkrétní IP adresa nebo daná IP adresa spadá do jediného rozsahu, je použit RID platný pro tuto IP adresu, či rozsah.
- Pokud je nalezeno dokonce několik rozsahů (navzájem vnořených) s různými RID, potom je vždy nutné nalézt ten nejmenší z nich (nejvíce zanořený) a jeho RID použít pro označení zachycených dat.

V příkladě uvedeném v sekci 4.4 jsou data z rozsahu *2001:db8:e::/48* označována SID_3, respektive RID_3, až na podrozsah adres *2001:db8:e:abcd::/64*. Data vztahující se ke komunikaci stroje s IPv6 adresou *2001:db8:e:abcd::1* jsou označována SID_2, respektive RID_2. Ostatní data vztahující se ke komunikaci s alespoň jednou IPv6 adresou z rozsahu *2001:db8:e:abcd::/64* jsou označována SID_1, respektive RID_1.

V případě, že daný paket byl odposlechnut na základě celé pětice (protokol transportní vrstvy, zdrojová a cílová IP adresa, zdrojový a cílový port), pak je nutné paket označit jen daným RID vztahujícím se k dané pětici. Jinak je potřeba aplikovat algoritmus zvlášť na zdrojovou IP adresa (a port) a cílovou IP adresu (a port). Daný paket může náležet do různých odposlechů na základě jak svého odesilatele, tak svého adresáta.

Sondy CC-IIF jsou konfigurovány pomocí IP adres, rozsahu IP adres, trojic a pětic transportní vrstvy, které mají odposlouchávat, a SID, kterým odposlechnutá data označovat. Sonda pro každý obdržený SID okamžitě vygeneruje unikátní RID tak, aby pro danou sondu byla dodržena bijekce mezi nakonfigurovanými SID a používanými RID.

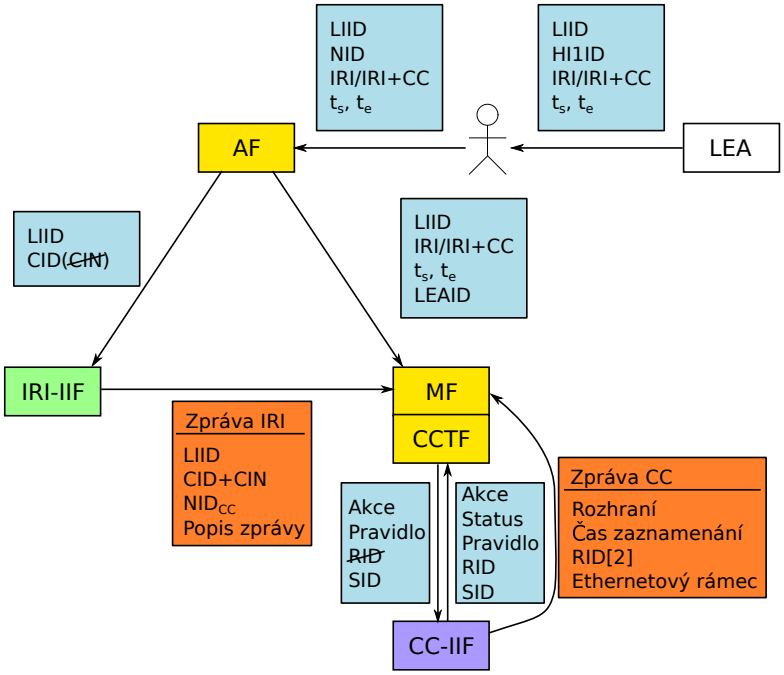
U sond CC-IIF se předpokládá zapojení za tapem, hubem, nebo SPAN portem. Z kopie provozu síťové linky sondy odstraňují pakety, které nejsou předmětem některého z odposlechů. Na

svůj výstup propustí pouze rámce, které obsahují IP datagramy spadající k některému z aktuálních odposlechů.

Všimněte si, prosím, že zachycená data mohou sondy CC-IIF označit až dvěma identifikátory RID, protože se odposlech může vztahovat jak ke zdrojové, tak k cílové IP adrese (a portu).

4.7 Identifikátory předávané uvnitř systému

Tato kapitola popisovala scénář odposlechu a význam zpráv předávaných mezi jednotlivými částmi systému SLIS. Dynamickou identifikací podezřelých se blíže zabývá kapitola 5 a přesnou strukturou zpráv kapitola 8. Obrázek 4.3 shrnuje přenášené zprávy v rámci SLIS a ukazuje výměnu identifikátorů mezi částmi systému.



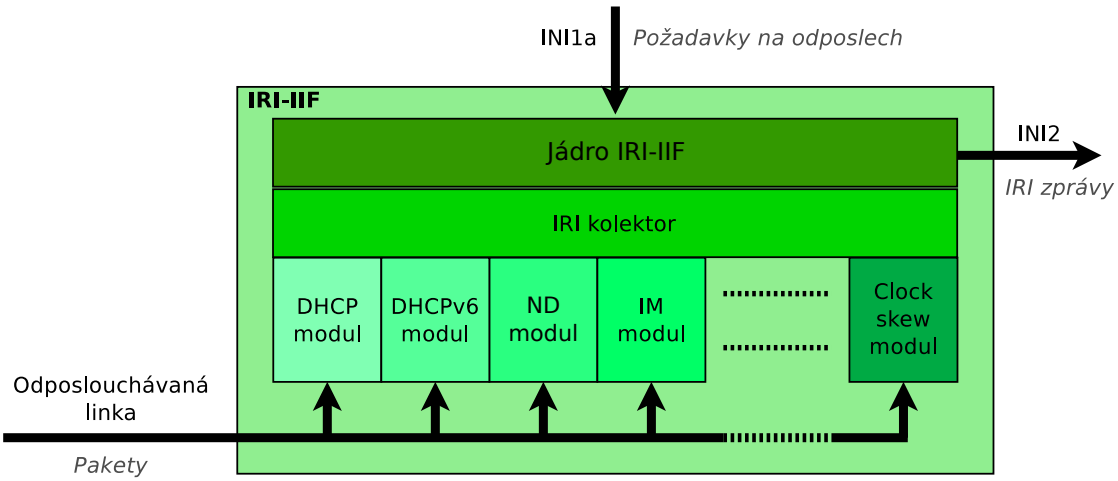
Obrázek 4.3: Výměna zpráv v rámci SLIS vytvořeného v rámci projektu *Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace*

Kapitola 5

Dynamická identifikace uživatele

5.1 Architektura bloku IRI-IIF

Identita odposlouchávaného uživatele (IP adresa, případně další identifikátory používané v síťovém prostředí) se může na straně poskytovatele dynamicky měnit např. skrze protokoly DHCP, RADIUS, ND apod. Úlohou bloku IRI-IIF je sledovat tuto identitu a informovat MF&CCTF o změnách v identifikátorech určených k odposlechu tak, aby mohly být včas překonfigurovány napojené sondy CC-IIF. Architektura bloku IRI-IIF navrženého v rámci projektu *Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace*je znázorněna na obrázku 5.1.



Obrázek 5.1: Architektura bloku IRI-IIF

Vstupy bloku tvoří:

1. Požadavky na odposlechy (rozhraní INI1a) zahrnující jednoznačný identifikátor odposlechu LIID a jednoznačný identifikátor (NID) odposlouchávaného uživatele (např. MAC adresa nebo přihlašovací jméno protokolu RADIUS).

2. Informace o změně identity uživatele, které mohou být ve formě síťové komunikace (např.

DHCP nebo RADIUS provoz) nebo ve formě logovacích souborů popř. jiných datových struktur.

Výstupy bloku tvoří tzv. *IRI zprávy* informující o identitě odposlouchávaného uživatele, o její změně popř. další doplňující informace. Jednotlivé typy a formát těchto zpráv je definován normou ETSI [24]. Jejich stručný přehled je uveden v tabulce 5.1. Aby bylo možné zjistit, ke kterému odposlechu příslušná IRI zpráva patří, je jejím povinným parametrem jednoznačný identifikátor odposlechu LIID, jež byl zadán jako součást vstupního požadavku na odposlech.

IRI zpráva	Popis
Begin	Oznamuje okamžik přidělení identity, vzniká vazba mezi NID a NID_{CC} (např. MAC adresou a IP adresou, nebo aplikačním identifikátorem a konkrétním spojením identifikovaným skrze 5-tici)
End	Oznamuje ukončení období pro přidělení identity. Vazba mezi NID a NID_{CC} zaniká.
Continue	Oznamuje pokračování platnosti identity - Vazba mezi NID a NID_{CC} stále platí.
Report	Jde o čistě informativní zprávu. Identita se nemění.

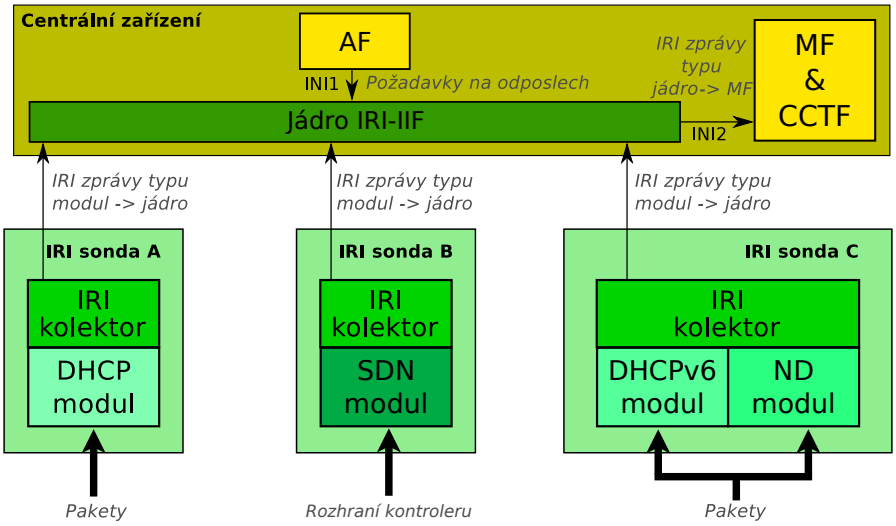
Tabulka 5.1: IRI zprávy vytvářené blokem IRI IIF

5.1.1 Modulární architektura

Architektura bloku IRI-IIF je navržena modulárně. Skládá se z tzv. *jádra IRI-IIF* a *modulů* pro analýzu jednotlivých protokolů (DHCP, ND, DHCPv6 apod.). Mezi jádrem a moduly existuje mezivrstva, tzv. *IRI kolektor*. Cílem kolektoru je přijímat zprávy od jednotlivých modulů a přeposílat je jádru. Díky použití této mezivrstvy je možné jednotlivé části IRI-IIF provozovat i jako samostatná zařízení v síti, tzv. *IRI sondy*, jak bylo popsáno v sekci 3.2. IRI sonda sestává z vlastního kolektoru a jednoho, či více modulů. Kolektor zprávy od modulů přeposílá do centrálního zařízení, kde se nachází jádro IRI-IIF. Tento princip je znázorněn na obrázku 5.2.

Modul analyzuje a zpracovává informace o změně identity uživatelů v rámci sledované sítě. Sleduje zejména požadavky o přidělení IP adresy, odpovědi na tyto požadavky a pro každého uživatele si udržuje stav, ve kterém se v rámci procesu přidělení IP adresy nachází. Získané informace modul následně předává jádru IRI-IIF, které je schopno na jejich základě snadno generovat výstupní IRI zprávy. Všimněte si prosím, že modul nemá informace o tom, kteří uživatelé jsou odposlouchávání a udržuje si proto seznam přidělených IP adres pro všechny uživatele sítě.

Jádro IRI-IIF přijímá vstupní požadavky na odposlechy a udržuje si tabulku aktuálně probíhajících odposlechů. Dále přijímá zprávy od jednotlivých modulů, provádí jejich filtraci (na základě aktuálně probíhajících odposlechů) a generuje výstupní IRI zprávy. Kromě filtrace je jádro zodpovědné i za propojení informací z různých protokolů, které se týkají odposlouchávaného uživatele (bude podrobněji vysvětleno níže v kapitole 5.2). V neposlední řadě si jádro také udržuje aktuální seznam identit všech uživatelů v síti, aby bylo schopné reagovat na případy, kdy přijde požadavek na odposlech již komunikujícího uživatele.



Obrázek 5.2: Modulární architektura IRI-IIF

5.1.2 Rozhraní mezi moduly a jádrem IRI-IIF

Jedním z cílů této práce bylo navrhnout rozhraní mezi moduly a jádrem tak, aby mělo jádro s případnou úpravou předávaných informací co nejmenší práci. Navržen byl proto takový způsob, kdy jednotlivé moduly zasílají informace o pokusu nebo přidělení identity ve formě velmi blízké výstupním IRI zprávám. Jádro pak provádí pouze jejich filtraci a případnou úpravu (např. doplnění jednoznačného identifikátoru odposlechu LIID).

Jednotlivé typy zpráv zasílaných mezi modulem a jádrem jsou uvedeny v tabulce 5.2. U zpráv typu *IRI Begin*, *End* a *Continue* se obvykle jedná o označení začátku, ukončení nebo prodloužení období, kdy byla přiřazena IP adresa, či kdy bylo probíhalo určité TCP spojení. Při obecném návrhu jádra IRI-IIF je však potřeba zpracovávat ze strany modulů nejen informace o přidělení IP adres a o probíhajících TCP spojeních, ale také informace o autentizaci.

U některých modulů (např. modul pro sadu Instant Messaging protokolů) lze z IRI zpráv vyčíst jak informace o autentizaci (např. IRC Login), tak indentifikaci konkrétního TCP spojení či IP adresy komunikujících stran. Pokud tedy budeme odposlech cílit na konkrétní IRC Login, můžeme ihned určit související TCP spojení a odposlouchávat jej.

V případě modulů pro protokol PPP nebo RADIUS však nemůžeme odposlouchávat cíl čistě na základě informací z těchto modulů. Potřebujeme dodatečné informace z jiného modulu, na základě kterých můžeme určit IP adresu cíle nebo TCP spojení v rámci nějž komunikuje.

Tento princip však platí i opačně: Bez údajů o autentizaci z modulů pro protokol PPP nebo RADIUS však není možné efektivně spojovat informace o propojení některých protokolů v grafu a realizovat odposlechy zadané na určité druhy NIDů (např. na RADIUS Login - viz následující příklad).

Příklad:

Uvažujme síť, kde se uživatelé nejprve autentizují skrze RADIUS a výslednou IP adresu získají až na základě protokolů DHCP nebo ND. Odposlech konkrétního uživatele je systému pro zákonné odposlechy zadán skrze *RADIUS login*. V bloku IRI IIF jsou k jádru připojeny 3 moduly (pro zpracování protokolů RADIUS, DHCP a ND). První modul sleduje autentizaci uživatele pomocí protokolu RADIUS a vytváří zprávy o spojení *RADIUS login - MAC adresa*.

Další dva moduly sledují protokoly pro přidělení IP adresy a vytváření zprávy o spojení *MAC adresa - IP adresa*. Pouze kombinací obou těchto vazeb lze realizovat uvedený odposlech, zadaný na *RADIUS login*.

IRI zpráva	Popis
Begin	Oznamuje úspěšnou autentizaci nebo přidělení identity
End	Ukončení období pro autentizaci nebo přidělení identity
Continue	Obnova identity
Report	Informativní zpráva (přidělení identity nenastává)

Tabulka 5.2: IRI zprávy předávané mezi moduly a jádrem bloku IRI IIF

5.2 Architektura jádra IRI-IIF

Jednou z hlavních úloh jádra IRI-IIF je zpracovávat vstupní požadavky na odposlech a uchovávat si o nich potřebné informace dokud daný odposlech neskončí (tzv. *management požadavků na odposlech*). Druhou z klíčových úloh tohoto bloku tvoří sledování příchozích zpráv ze strany jednotlivých modulů, provádění jejich filtrace na základě aktivních odposlechů a na závěr jejich případná transformace do požadovaného formátu IRI zpráv. V rámci této úlohy je také nezbytné, aby si blok uchovával informace o již aktivních spojeních a byl schopen reagovat na příchod požadavku na odposlech týkající se již aktivní komunikace (tzv. *management aktivních spojení*). V neposlední řadě musí jádro IRI-IIF správně identifikovat rozsah odposlechu a případně zreplikovat výstupní IRI zprávu pro všechny požadavky, kterých se týká. Všechny tyto úlohy jádra IRI-IIF jsou podrobněji popsány v následujících podkapitolách a na závěr je uvedeno celkové schéma činnosti tohoto bloku.

5.2.1 Management požadavků na odposlechy

Všechny potřebné pro správu požadavků na odposlechy si jádro IRI-IIF uchovává v tzv. *Tabulce odposlechů*, která obsahuje následující položky:

1. Jednoznačný identifikátor odposlechu LIID.
2. Jednoznačný identifikátor odposlouchávaného uživatele (NID), např. MAC adresa, IPv6 adresa, RADIUS login apod.
3. Identifikátor (CIN) sezení nebo komunikace v rámci jednoho odposlechu zadaného skrze LIID. Poznámka: Ve skutečnosti je ukládán celý CID identifikátor, avšak pro funkci bloku IRI-IIF je využit pouze CIN.
4. Čas začátku/konce odposlechu (požadavky na začátek/konec odposlechu jsou zasílány dopředu s časovou rezervou).
5. Příznak, zda se má uchovávat kompletní obsah komunikace nebo pouze IRI zprávy.
6. Příznak úrovně odposlechu (tak jak bylo specifikováno v sekci 3.4, podrobněji jsou úrovně specifikovány dále v této sekci, především v částech 5.2.3, 5.2.4 a 5.2.5.

Jádro IRI-IIF sleduje zprávy na rozhraní INI1a a rozlišuje příchozí požadavky na:

1. *Založení nového odposlechu* - Při příchodu požadavku na nový odposlech si jádro IRI-IIF uloží potřebné informace do *Tabulky odposlechů*. Pokud se požadavek na nový odposlech týká již probíhající komunikace, potom blok vygeneruje zprávu *IRI Begin* (více informací viz *Management aktivních spojení*)

2. *Zrušení odposlechu* - Při příchodu požadavku na zrušení odposlechu odstraní jádro IRI-IIF příslušnou položku z *Tabulky odposlechů*.

3. *Aktualizace informací o odposlechu* - Skrze požadavek na aktualizaci může být např. upraven začátek nebo konec odposlechu popř. i provedena změna v typu zachytávaných dat (zprávy IRI vs. obsah komunikace CC).

Příklad:
Uvažujme scénář, kdy IRI blok přijme dva požadavky na odposlech. První bude označen jednoznačným identifikátorem LIID=X, bude se vztahovat na komunikaci počítače s MAC: 00:25:90:0f:81:37 (pouze IRI zprávy) a bude probíhat od 13:37 1.1.2012. Druhý odposlech bude označen jednoznačným identifikátorem LIID=Y, bude se vztahovat na komunikaci počítače s IPv4: 192.168.1.63 (IRI zprávy i obsah komunikace) a bude probíhat od 12:00 3.6.2012. Obsah *Tabulky odposlechů* je znázorněn v tabulce 5.3. Pozn.: bližší informace o úrovni odposlechu budou uvedeny níže.

LIID	NID	CIN	CC	úroveň	začátek odposlechu
X	MAC: 00:25:90:0f:81:37	1	ne	1	13:37 1.1.2012
Y	IPv4: 192.168.1.63	4	ano	1	12:00 3.6.2012

Tabulka 5.3: Obsah tabulky odposlechů

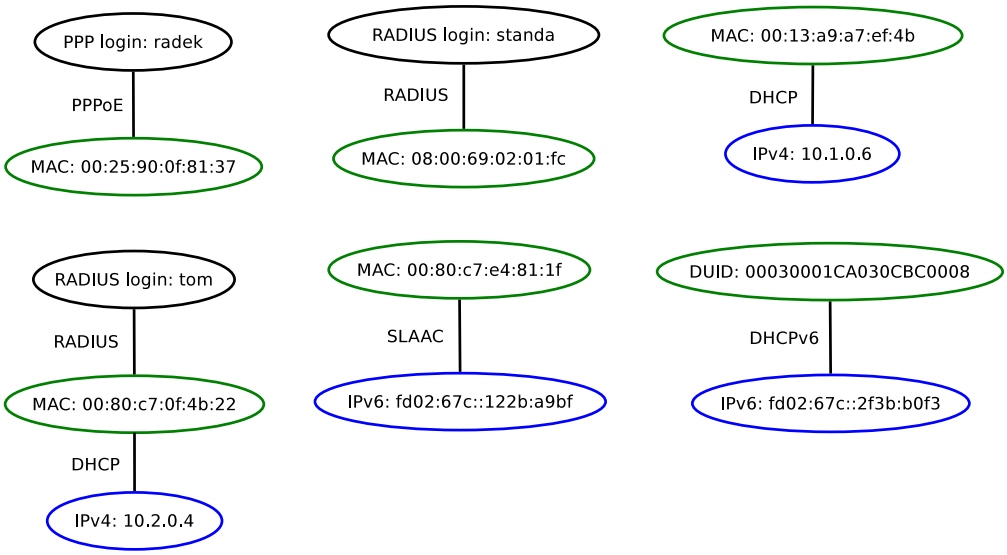
5.2.2 Management aktivních spojení

Hlavní úlohou jádra IRI-IIF je zpracovávat informace z jednotlivých modulů, provádět jejich filtraci s ohledem na aktivní odposlechy a transformovat je do požadovaného tvaru IRI zpráv. Mimo to musí také při příchodu nového požadavku na odposlech ověřit, zda se týká již aktivní komunikace či nikoliv. Pokud odposlouchávaný cíl již aktivně komunikuje, vygeneruje blok odpovídající zprávu *IRI Begin*. Aby bylo možné tyto činnosti realizovat, musí si jádro IRI-IIF uchovávat informace o aktuálně přidělených adresách všech počítačů v rámci sledované sítě.

Informace o detekovaných identifikátorech jsou uchovávány v grafu. Uzly grafu reprezentují detekované NIDy a hrany zachycují vazbu mezi vrcholovými NIDy známou z konkrétního síťového protokolu. Formálně je zachycený graf definován jako $G = (V, E, p)$, kde:

- V je množina vrcholů. Každý vrchol reprezentuje konkrétní NID detekovaný jedním z modulů.
- $E \subseteq V \times V$ je symetrická relace hran, kde každá hrana spojuje právě dva vrcholy.
- $p : E \to (\mathcal{P}(P) \setminus \emptyset)$ je úplná funkce, která každé hraně přiřazuje jeden nebo i více protokolů z množiny P (množiny všech protokolů), konkrétně ty, ze kterých byla vazba mezi NIDy naučena. Např. $p((K.L.M.N, x@y))$ vrací množinu obsahující SMTP pokud informace o použití e-mailové adresy $x@y$ na počítači s IP adresou $K.L.M.N$ byla nalezena modulem pro SMTP. Ve většině případů je vazba mezi identifikátory nalezena právě jedním modulem a pak je výstupní množina jednoprvková.

Výše definovaný graf nikdy neobsahuje hrany mezi vrcholem představujícím IP adresu serveru poskytující služby a spojením TCP/UDP k tomuto serveru. Všimněte si také, že mezi množinou detekovaných identifikátorů a vrcholy grafu existuje bijekce. Konkrétnímu identifikátoru i tedy vždy náleží právě jediný vrchol a naopak každý z vrcholů grafu reprezentuje právě jeden identifikátor.



Obrázek 5.3: Příklad grafu v jádře IRI-IIF

Příklad:

1. Modul PPPoE: počítači s MAC adresou 00:25:90:0f:81:37 byla udělena autorizace na základě PPP loginu „radek“ a hesla.
2. Modul RADIUS: počítači s MAC adresou 08:00:69:02:01:fc byla udělena autorizace na základě RADIUS loginu „standa“ a hesla.
3. Modul DHCP: počítači s MAC adresou 00:13:a9:a7:ef:4b byla přidělena IPv4 adresa 10.1.0.6.
4. Modul RADIUS: počítači s MAC adresou 00:80:c7:0f:4b:22 byla udělena autorizace na základě RADIUS loginu „tom“ a hesla a současně mu byla také přidělena IPv4 adresa 10.2.0.4.
5. Modul ND: počítači s MAC adresou 00:80:c7:e4:81:1f byla přidělena IPv6 adresa fd02...
6. Modul DHCPv6: počítači s DUID identifikátorem 0003.. byla přidělena IPv6 adresa fd02...

Na základě těchto událostí může v jádře IRI-IIF vzniknout graf dle obrázku 5.3.

Obecně jádro IRI-IIF na základě IRI zpráv od modulů spravuje *Graf NIDů* dle následujících pravidel:

1. Při přijetí *IRI Begin* nebo *IRI Continue* je do grafu vložena nová hrana mezi vrcholy popsanými ve zprávě. Pokud tyto vrcholy v grafu dosud neexistují, přidají se do něj.

2. Při přijetí *IRI End* je z grafu odstraněna hrana mezi vrcholy popsanými ve zprávě. Pozn.: příslušná hrana v grafu může být odstraněna pouze modulem, který inicializoval jeho založení.

3. Z grafu jsou automaticky odebírány vrcholy, mezi nimiž není žádná hrana.

Poslední úlohou v rámci managementu aktivních spojení je udržování a aktualizace identifikátorů CIN u aktivních odposlechů. Jinými slovy blok IRI-IIF musí dle norem s každým nově přiděleným NID_{CC} inkrementovat hodnotu CIN u daného odposlechu. IRI-IIF blok si tedy musí uchovávat tabulku přiřazených CIN identifikátorů vyjadřující vztah mezi NID_{CC}, CIN a LIID (viz příklad v tabulkce 5.4). Tyto změny se promítají také do tabulky odposlechů, kde je u každého odposlechu uchováván vždy nejvyšší přiřazený CIN pro odpovídající LIID.

NID_{CC}	LIID	CIN
MAC: 00:25:90:0f:81:37	X	1
IPv4: 192.168.1.63	Y	4
IPv6: 2001:0db8:3c4d::abcd:ef12	Y	3

Tabulka 5.4: Příklad tabulky CIN v bloku IRI-IIF

5.2.3 Typy NID

Jak již bylo naznačeno v sekci 3.4, orgány činné v soudním řízení a následně soudní příkaz může požadovat dodání rozdílných dat v případě, že konkrétní počítač, či uživatel využívají souběžně několik komunikaci pomocí několika síťových protokolů.

Analýzou požadavků ETSI [17, 26, 22, 25, 27, 20] jsme dospěli k závěru, že LIS si musí poradit s komunikacemi na různých vrstvách síťového modelu [35]. Nicméně detekce identifikátorů NID na různých úrovních modelu přestavuje rozdílné typy informace. Proto jsme podporované NID rozdělili do následujících čtyř kategorií představujících typ NIDu z hlediska jeho významu v síti:

- **typ** α - Aplikační identifikátor, identifikátor aplikačních spojení
  - 5-tice (IP klienta, IP serveru, port klienta, port serveru, typ transportního protokolu)
  - 3-jice (IP, port, typ transportního protokolu)
- **typ** β - Adresa síťové vrstvy
  - IPv4 adresa
  - IPv6 adresa
- **typ** γ - Adresa síťového rozhraní, nebo identifikátor konkrétního počítače
  - MAC adresa
  - DHCP client ID [15]
  - DHCPv6 DUID [7]
- **typ** δ - Ostatní identifikátory (především pro autentizaci)
  - RADIUS login [67]

  - PPP login [73]
  - Číslo PPP sezení [73]

- **typ** λ - Identifikátor používaný v rámci protokolu aplikační vrstvy
  - login IRC, název kanálu v rámci serveru IRC [56],
  - login login XMPP [69],
  - login YMSG,
  - login OSCAR,
  - login SIP [68],
  - e-mailová adresa [66] apod.

I když současná implementace jádra IRI-IIF výrazně nerozlišuje mezi kategorii λ a α, jak je ukázáno v článku *On Identities in Modern Networks* [61], je takové dělení výhodné.

5.2.4 Princip vyhledávání v grafu

Cílem bloku IRI-IIF je ke vstupnímu NID, který je součástí zadání odposlechu, nalézt všechny související NID. Ze souvisejících NID jsou podstatné především NID_{CC}, které se používají ke konfiguraci sondy CC-IIF. Ostatní související NID mají pouze informativní hodnotu a jsou posílány na výstup jako metadata skrz rozhraní HI2.

Jak bylo uvedeno v sekci 5.2.2, jsou získané informace o nalezených identitách ukládány v jádře IRI-IIF. Úlohu nalezení NID související s odposlechem je proto možné řešit pomocí algoritmů nad grafy. Vstupní zadání odposlechu cílí na jeden konkrétní NID. Výstupem hledání je množina NID_{CC}, na které má být veden odposlech.

V této části zprávy budeme rozlišovat následující čtyři úrovně odposlechu:

I. úroveň - zachytává jen data, která se přímo vztahují ke vstupnímu NIDu. Např. pokud dostaneme autentizační login, chceme zachytávat jen a pouze pakety obsahující IP adresu rozhraní autentizované tímto loginem. Pokud je vstupní NID IP adresa, pak nemáme zájem o jiné IP adresy, i kdyby byly použité na stejném rozhraní jako vstupní IP adresa.

II. úroveň - zachytává data vztahující se ke konkrétnímu počítači, nebo alespoň konkrétnímu síťovému rozhraní. Pro vstupní NID ve formě IP adresy jsou předmětem data adresovaná nejen vstupní IP adresou, ale i jinými IP adresami používaných na stejném rozhraní a je-li to možné, i všechny IP adresy tohoto počítače.

III. úroveň - zachytává jen data, která jistě pocházejí od konkrétního uživatele, nebo která jsou mu zasílaná.

IV. úroveň - je nejobecnější a zachytává data na základě informací o všech identifikátorech patřící do té komponenty grafu, ve které se nachází vstupní NID. Na rozdíl od III. úrovně, může při použití této úrovně potenciálně dojít k zachycení dat jiného uživatele sdílejícího stejný počítač.

Pro vysvětlení principu vyhledávání je potřeba definovat několik pomocných struktur, které jsou dále využívány v rámci prohledávání v grafu. Následující text byl původně publikován ve článku *On Identities in Modern Networks* [61].

Začněme definicí množiny $C = \{\alpha, \beta, \gamma, \delta, \lambda\}$ (podporované kategorie NID). Dále zavedeme úplnou funkci $l : V \to C$, která přiřazuje každý vrchol do jedné z kategorií C. Nad množinou kategorií Cdále definujme částečné uspořádání $\delta > \gamma > \beta > \alpha$ a $\lambda > \alpha$.

Smyslem uspořádání $>$ nad množinou C je postihnout vztahy mezi jednotlivými skupinami identifikátorů. Konkrétní tok z kategorie α se vždy vztahuje ke konkrétnímu počítači v případě 3-jice a ke dvěma počítačům v případě 5-tice. Ke konkrétní IP adrese z kategorie β je však možné potenciálně najít mnoho toků kategorie α. Podobně pro jeden počítač, či jeho jedno rozhraní (označené identifikátorem z kategorie γ) je možné v síti nalézt více IP adres (typicky jedna IPv4 a mnoho IPv6 [55]). Obdobně je možné autentizovat více počítačů pomocí jediného identifikátoru kategorie δ.

Kategorie λ přináší jiný pohled na toky kategorie α a jejich aplikační obsah. Při využívání jedné aplikace je možné využívat několik toků pro jedno sezení, např. řídící a datový kanál SIP. Vztah $\lambda > \alpha$ postihuje právě tento poznatek.

Definujme funkci capture : $V \to 2^V$, která pro daný identifikátor (reprezentovaný vrcholem grafu z množiny V) vrací množinu známých identifikátorů spadajících do dané úrovně odposlechů.

Ještě před definováním capture budeme potřebovat definovat pomocné binární relace. Vztah 5.1 definuje antisymetrickou binární relaci covers $\subseteq V \times V$. Vrcholy x a y jsou ve vztahu covers v případě, že jsou přímo propojené a y patří do specifičtější kategorie C.

$$\text{covers} = \{(x, y) \in V^2 : (x, y) \in E \wedge l(x) > l(y)\}. \tag{5.1}$$

Vztah 5.2 definuje binární relaci $\text{linked}_c \subseteq V \times V$, one for each $c \in C$. Vrchol x je v relaci linked_c s vrcholem y pokud jsou přímo propojené a vrchol y patří do kategorie c nebo specifičtější. Protože graf zachycující známý stav objevených identifikátorů nesmí obsahovat hrany mezi IP adresou serveru a tokem kategorie α, není možné se pomocí relace linked_c dostat do domény jiného počítače. To je chtěná a podstatná vlastnost, protože zaručuje, že tranzitivní a reflexivní uzávěr linked_c^* nepropojuje identity různých komunikujících.

$$\text{linked}_c = \{(x, y) \in V^2 : (x, y) \in E \wedge l(y) \leq c\}. \tag{5.2}$$

A nyní již definujme relaci capture pomocí vztahu 5.3 za využití reflexního a tranzitivního uzávěru nad relacemi covers a linked_c for $c \in \{\gamma, \delta, \lambda\}$.

$$\text{capture}(v) = \begin{cases} \{x \in V \ : v \ \text{covers}^* \ x\} & : \text{ I. úroveň,} \\ \{x \in V \ : v \ \text{linked}_\gamma^* \ x\} & : \text{ II. úroveň,} \\ \{x \in V \ : v \ \text{linked}_{l(v)}^* \ x\} & : \text{ III. úroveň, } l(v) \in \{\delta, \lambda\}. \\ \{x \in V \ : v \ (\text{linked}_\delta \cup \text{linked}_\lambda)^* x\} & : \text{ IV. úroveň} \end{cases} \tag{5.3}$$

V rámci implementace SLIS jsou podporovány pouze úrovně I, II a IV. Úroveň III není podporována. Podporované úrovně jsou v rámci uživatelského rozhraní značeny arabskými číslicemi 1, 2 a 3. Úroveň IV je označována jako úroveň 3.

Identifikátory určené pro konfiguraci sondy (NID_{CC}) je možné získat pomocí vztahu 5.4.

$$\text{capture}_{CC}(v) = \{x \in V \ : x \in \text{capture}(v) \wedge l(x) \in \{\alpha, \beta\}\}. \tag{5.4}$$

Článek *On Identities in Modern Networks* [61] dále obsahuje rozšíření modelu pro sítě s překladem adres. Toto rozšíření nebylo v rámci SLIS implementováno.

Vlastní implementace vyhledávání v grafu pracuje podle následujícího algoritmu:

1. Inicializuj množinu r tak, že obsahuje pouze vrchol představující vstupní NID x.
2. Postupně procházej hrany v grafu vedoucí z uzlů v množině r. Pokud hrana vede do nově nalezeného uzlu patřícího do capture x, přidej jej do množiny r.

3. V hledání pokračujeme, dokud se množina r rozrůstá. Pokud již nemůžeme najít nový uzel patřící do r, pak capture$(x) = r$.

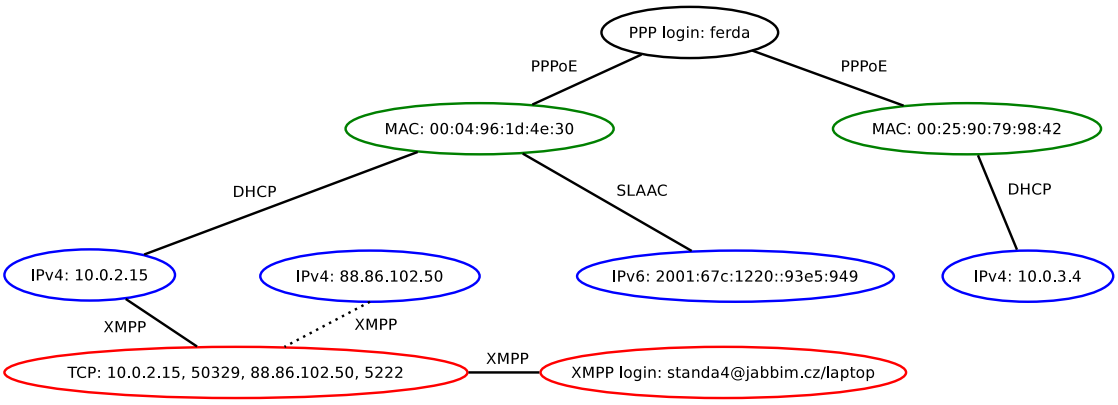
5.2.5 Ukázka vyhledávání v grafu

Ukázkový příklad prohledávání grafu uvažuje situaci uvedenou na obrázku 5.4. Máme zde uživatele, identifikovaného pomocí loginu PPP: *ferda*. Tímto loginem jsou autentizována dvě síťová rozhraní s MAC adresami *00:04:96:1d:4e:30* a *00:25:90:79:98:12*. Prvnímu rozhraní je přiřazena IPv4 adresa *10.0.2.15* a IPv6 adresa *2001:67c:1220::93e5:949*. Druhému rozhraní je přiřazena IPv4 adresa *10.0.3.4*.

Uživatel komunikuje pomocí protokolu XMPP přes vzdálený server s počítačem s přiřazenou IP adresou *88.86.102.50*. Tato komunikace probíhá v rámci TCP spojení uvedeného na obrázku. Uživatel v rámci dané komunikace požívá login XMPP (spojený s informacemi o místě použití za znakem '/': *standa4@jabbim.cz/laptop*. Všimněte si, že IP adresa serveru není v grafu propojená hranou z množiny E, jak bylo specifikováno v části 5.2.2.

Uvažujme tři různé odposlechy:

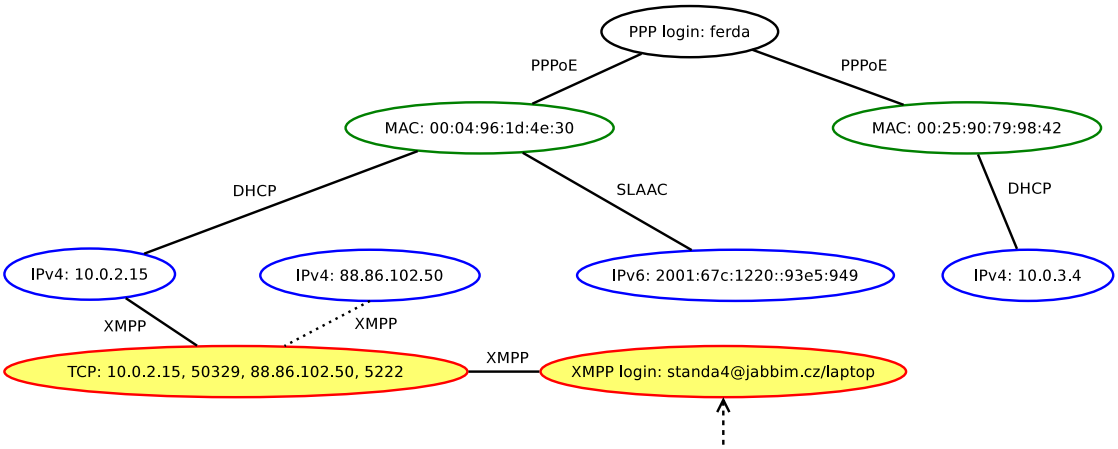
- **Odposlech X** - Cílem je NID představující XMPP login: *standa4@jabbim.cz/laptop*. Úroveň odposlechu je 1.
- **Odposlech Y** - Cílem je také NID představující XMPP login: *standa4@jabbim.cz/laptop*. Úroveň odposlechu je 2.
- **Odposlech Z** - Cílem je také NID představující XMPP login: *standa4@jabbim.cz/laptop*. Úroveň odposlechu je 3.



Obrázek 5.4: Ukázkový příklad grafu v IRI-IIF

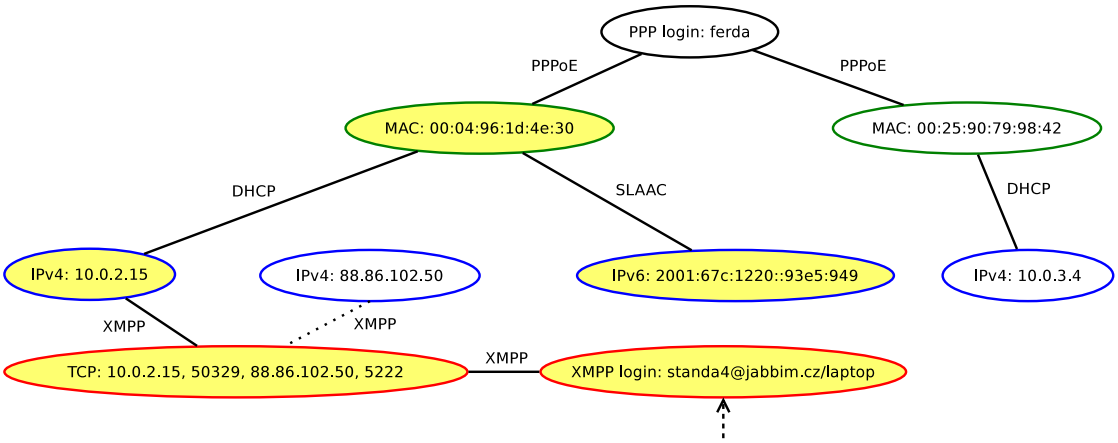
Odposlech X má nastavenou **úroveň 1**. Dle postupu specifikovaného v části 5.2.4 bude do výsledné množiny uzlů zahrnut pouze samotný vstupní uzel (NID) typu λ a přímo propojené uzly typu α. Výsledek hledání je vyobrazen na obrázku 5.5. Uzly patřící do nalezené množiny jsou zvýrazněny žlutě. NID související s odposlechem jsou tedy XMPP login a TCP spojení v rámci něhož XMPP komunikace probíhá.

Odposlech Y má nastavenou **úroveň 2**. Na této úrovni patří do oblasti zájmu veškerá komunikace probíhající přes dané síťové rozhraní. Dle postupu specifikovaného v části 5.2.4 budou do výsledné množiny uzlů propojené uzly až do úrovně γ Výsledek hledání je vyobrazen na



Obrázek 5.5: Odposlech 1. úrovně cílený na XMPP login

obrázku 5.6. K nalezeným uzlům tedy přibyla MAC adresa *00:04:96:1d:4e:30* daného rozhraní a obě IP adresy, které jsou tomuto rozhraní přiřazeny.

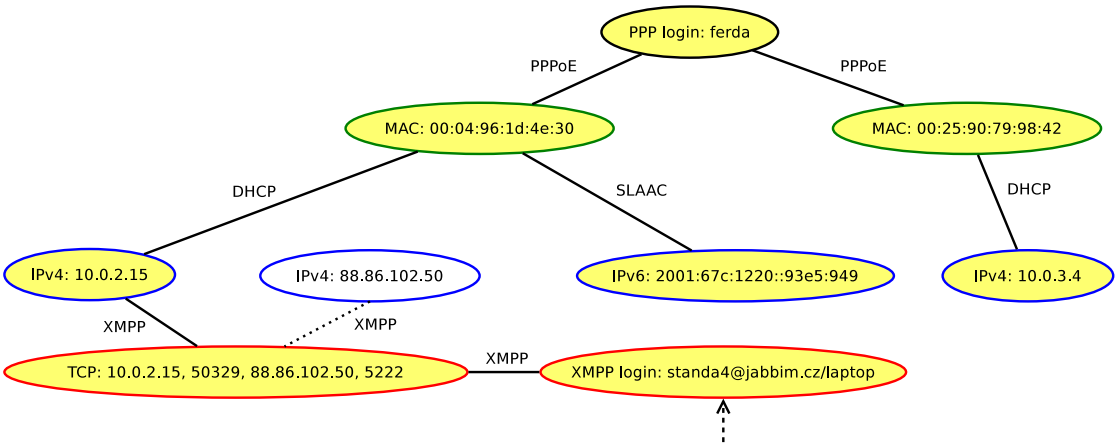


Obrázek 5.6: Odposlech 2. úrovně cílený na XMPP login

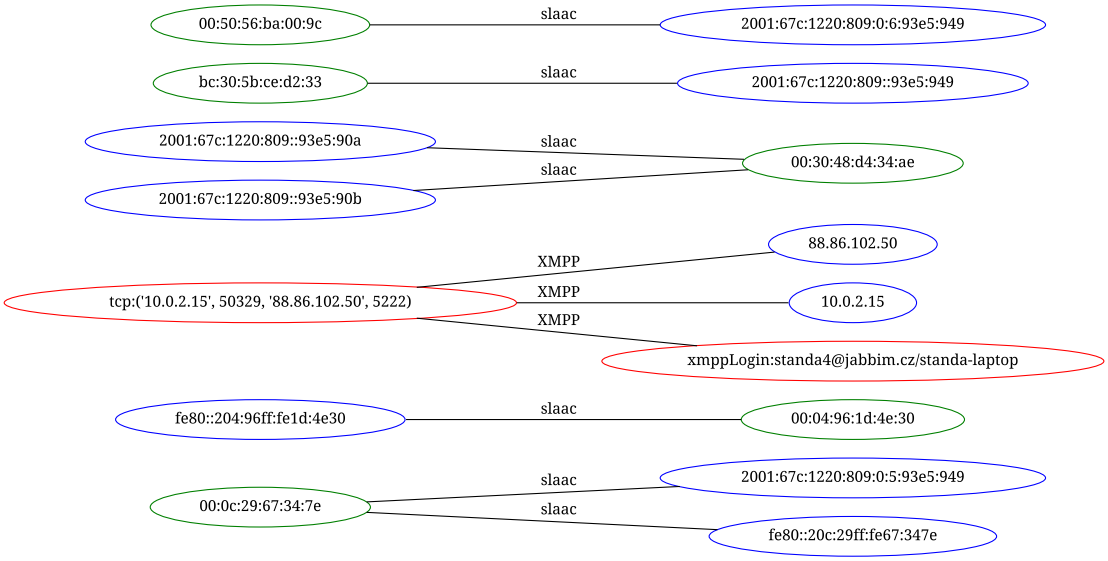
Odposlech Z má nastavenou **úroveň 3**. Na této úrovni se snažíme pokrýt veškerou komunikaci spojitelnou s aktivitami daného uživatele. Dle postupu specifikovaného v části 5.2.4 budou do výsledné množiny uzlů zahrnuty všechny uzly v dané komponentě grafu. Výsledek hledání je vyobrazen na obrázku 5.7. K nalezeným uzlům tedy přibyla login PPP *ferda*, MAC adresa *00:25:90:79:98:12* druhého rozhraní patřícího danému uživateli a také IPv4 adresa, která je tomuto rozhraní přiřazena.

5.2.6 Zobrazení grafu skrz webové rozhraní

Skrz webové rozhraní systému SLIS je možné celý graf vizualizovat a poskytnout tak operátorovi obsluhujícímu systém lepší představu o aktuálním stavu sítě. Ukázka reálného grafu, která byla získána z webového rozhraní systému, je na obrázku 5.8.



Obrázek 5.7: Odposlech 3. úrovně cílený na XMPP login



Obrázek 5.8: Ukázka reálného grafu NID zobrazitelného skrz webové rozhraní

Obrázek ukazuje, že graf detekovaných identifikátor typicky není spojitý. Počet komponent v síti bez autentizace uživatelů a dat z aplikačních protokolů by měl odpovídat počtu počítačů. V síti s autentizaci uživatelů jsou propojeny všechny stroje přihlášené konkrétním uživatelem a pokud při detekci dat z aplikačních protokolů může počet komponent dále klesat v důsledku vazeb přes identifikátory kategorie γ.

Na obrázku 5.8 vidíme přidělení IPv6 adres bezstavovou konfigurací IPv6. Např. počítač se síťovým rozhraním s MAC adresou *00:30:48:d4:34:ae* si vygeneroval dvě různé IPv6 adresy: *2001:67c:1220:809::93e5:90a* a *2001:67c:1220:809::93e5:90b*.

V prostřední části vidíme komunikaci pomocí protokolu XMPP [69]. Červený uzel v levé části představuje zmíněnou 5-tici, kde protokolem transportní vrstvy je TCP. V grafu vidíme asociace TCP spojení s uzly představujícími IP klienta a serveru. Z obrázku je také zřejmé, že

pro komunikaci uživatel používal XMPP login: *standa4@jabbim.cz/standa-laptop.*

5.3 Souhrn činnosti jádra IRI-IIF

Jádro IRI-IIF je realizováno jako událostmi řízený program. Událostmi se rozumí jednak příchozí požadavky od AF z rozhraní INIa, jednak IRI zprávy ze strany modulů.

- Obdrží-li jádro IRI-IIF z rozhraní INI1a požadavek na nový odposlech:
  1. Přidá odposlech jako novou položku do tabulky odposlechů.
  2. Analyzuje obsah tabulky NID a identifikuje všechny NID_{CC}, které s daným odposlechem souvisí.
  3. Pro každý NID_{CC} vygeneruje zprávu *IRI BEGIN* (odposlech na již aktivní komunikaci) a pošle ji na rozhraní INI2.
- Obdrží-li jádro IRI-IIF z rozhraní INI1a požadavek na zrušení odposlechu:
  1. Odstraní odpovídající položku z tabulky odposlechů.
- Obdrží-li jádro IRI-IIF zprávu IRI ze strany modulů:
  1. Zkontroluje, zda se některý z NID ve zprávě netýká některého z aktivních odposlechů. Pokud ano, potom:
     (a) Identifikuje všechny odposlechy (LIID identifikátory), ke kterým se IRI zpráva vztahuje.
     (b) Rozkopíruje IRI zprávu pro všechny příslušné LIID a všechny kopie pošle na rozhraní INI2.
  2. Vloží/odstraní/aktualizuje záznam v tabulce NID. Konkrétní činnost se liší podle typu zprávy:
     – *IRI-BEGIN*
       (a) Vloží do tabulky NID nový záznam obsahující ID modulu a příslušné NID.
       (b) Přidá nový záznam do tabulky CIN pro všechny LIID, kterých se NID týká. U dotčených odposlechů aktualizuje hodnoty CIN také v tabulce odposlechů na nejvyšší CIN pro dané LIID.
     – *IRI END*
       (a) Odstraní z tabulky NID záznam vytvořený odpovídající zprávou *IRI BEGIN*.
     – *IRI CONTINUE*
       (a) Pokud tento ještě neexistuje, vloží do tabulky NID nový záznam obsahující ID modulu a příslušné NID.
     – *IRI REPORT*
       (a) Žádné další akce

Kapitola 6

Podporované přístupy pro hledání identity uživatele

Následující části budou věnovány podrobnému popisu návrhu jednotlivých modulů bloku IRI-IIF pro analýzu protokolů přidělujících IPv4 nebo IPv6 adresy. Popis protokolů DHCP, RADIUS, PPPoE, DHCPv6 a Neighbor Discovery je převzat z dřívější technické zprávy [50]. Postupně byla v rámci projektu *Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace* přidávána podpora dalších zdrojů pro zjišťování identity [8, 39, 28, 31], některé zjištěné poznatky byly publikovány i formou vědeckých publikací [64, 57, 58, 61]. U každého přístupu je vždy nejprve uveden základní popis protokolu, či metody a následně podrobně popsán návrh modulu včetně stavového diagramu zobrazujícího jeho činnost.

Protokol DHCP (viz část 6.1) je stále dominantním způsobem pro zjištění konfigurace počítače po připojení počítače do lokální sítě. Protokoly RADIUS (viz část 6.2) a PPPoE (viz část 6.3 se používají často při připojení domácností pomocí technologií z rodiny (*Digital Subscriber Line* – DSL), protokol RADIUS se dále používá při autentizaci v rámci firemních sítí a v bezdrátových sítích. Bezestavová autokonfigurace adres (SLAAC, viz část 6.5) a protokol DHCPv6 (viz části 6.4 a 6.5) se používají v sítích založených na technologii IPv6. Jejich společným znakem je využívání mechanismu objevování sousedů (*Neighbor Discovery* – ND).

Z modulů pro protokoly pracující na aplikační vrstvě jsme se zabývali především protokoly pro komunikaci. Z protokolů pro komunikaci v reálném čase jsme vybrali protokoly XMPP (část 6.6), IRC (část 6.7) a proprietární OSCAR (část 6.8) a YMSG (část 6.9). Pro e-mailovou komunikaci podporujeme protokol SMTP (část 6.10), podpora tohoto protokolu byla implementována i v rámci rychlého zpracování aplikačních protokolů na vysokorychlostní sondě založené na diplomové práci [9].

Experimentálně jsme se pokoušeli identifikovat počítač bez informaci z lokální sítě, autentizačních systémů i znalosti aplikačního protokolu. Využili jsme princip založený na dříve publikované metodě založené na unikátní odchylce měření času [47], která je způsobená nepřesností výrobního procesu na atomární úrovni a je jedinečná pro každý počítač. Měření ukázala, že tato metoda je sice v principu použitelná [39, 28], ale ve skutečném prostředí nemusí být obelstěna, či nedostatečně přesná [64, 57]. Metodu a nástroj pro zjišťování odchylky v měření času popisuje část 6.11.

V poslední době jsou ve vědeckých kruzích často diskutovány softwarem řízené sítě (*Software Defined Networking* – SDN). Sítě v rámci SDN jsou řízené pomocí kontroléru. Část 6.12 popisuje možnosti nalezení identity z kontroleru OpenDaylight a Pox. Zákonnými odposlechy v prostředí SDN se také zabývá diplomová práce jednoho z členů týmu – Barbory Frankové. Očekáváme, že

diplomová práce bude dokončena a obhájena v první polovině roku 2015.

Tuto kapitolu uzavírá sekce 6.13, která shrnuje podporované identifikátory IRI-IIF a potažmo SLIS.

6.1 DHCP

6.1.1 Popis protokolu

Dynamic Host Configuration Protocol (DHCP) [15] se používá pro dynamické přidělování IPv4 adres. Toto přidělování je založeno na komunikaci mezi klientem a DHCP serverem. DHCP server (dále jen server) je zpravidla umístěn ve stejné podsíti jako klient. Server má k dispozici určitý rozsah adres, ze kterého přiděluje jednotlivým klientům. Proces přidělení adresy probíhá obvykle v následujících krocích:

1. Klient požádá o přidělení adresy zasláním zprávy *DHCP Discover* (skrze všesměrové vysílání).
2. Server zašle klientovi odpověď s navrhovanou adresou skze zprávu *DHCP Offer*.
3. Klient všesměrovým vysíláním požádá o navrhovanou adresu zasláním zprávy *DHCP Request*.
4. Server zašle klientovi potvrzení o přidělení adresy skrze zprávu *DHCP Ack*.

Platnost přidělené IPv4 adresy je časově omezena na hodnotu uvedenou v položce *Lease time*. Po vypršení této doby již nesmí klient přidělenou adresu používat, pokud si před jejím vypršením úspěšně nepožádal o její prodloužení. Žádost o prodloužení adresy probíhá v následujících krocích:

1. Klient požádá o prodloužení adresy zasláním zprávy *DHCP Request* (všesměrové vysílání). Klient jinými slovy žádá o stejnou adresu, kterou měl doposud přidělenu.
2. Server zašle klientovi buď potvrzení o přidělení adresy (zasláním zprávy *DHCP Ack*) nebo odmítnutí tohoto požadavku (zasláním zprávy *DHCP Nack*).

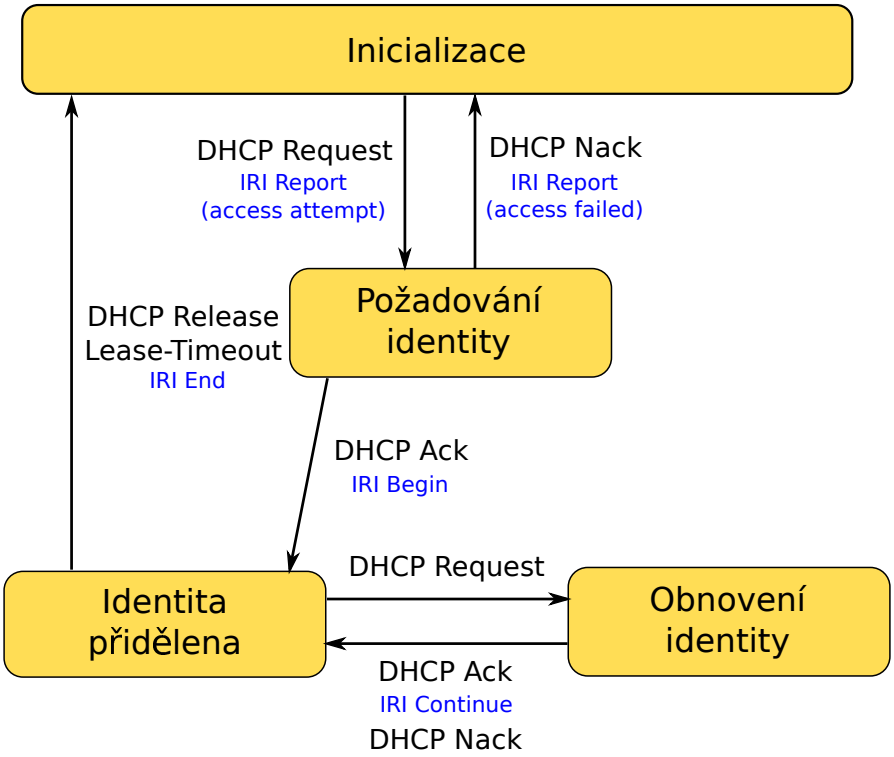
Uvedený scénář týkající se prodloužení platnosti adresy se může opakovat vícekrát. V případě odmítnutí požadavku o prodloužení adresy (*DHCP Nack*) nemá klient jinou možnost, než znovu požádat server o nabídku dostupných adres (zasláním zprávy *DHCP Discover*) a na základě nové nabídky (*DHCP Offer*) si vybrat adresu jinou. Pokud již klient nebude adresu dále používat, může (volitelně) sám předat serveru informaci o uvolnění adresy (zasláním zprávy *DHCP Release*), čímž se adresa vrátí zpět mezi nepřiřazené adresy a DHCP server jí může přidělit ostatním klientům.

Pro centrální správu adresového prostoru v několika podsítích lze využít *DHCP relay*. Jedná se o model, kde DHCP proxy servery v dané podsíti (tzv. *relay agenti*) nepřidělují adresy, ale pouze přeposílají DHCP zprávy na centrální DHCP server. DHCP server tyto požadavky obsluhuje a ve své odpovědi vyplní adresu *relay agenta*, který zprávu přepošle klientovi ve své podsíti. Pro klienta je komunikace plně transparentní (klient nepozná, zda komunikuje s *relay agentem* nebo přímo serverem).

6.1.2 Činnost IRI-IIF

Blok IRI-IIF analyzuje uvedené DHCP zprávy a na jejich základě udržuje aktuální tabulku přidělených IPv4 adres společně s jejich dobou platnosti a generuje příslušné IRI zprávy. Stavový diagram funkce IRI-IIF pro zpracovaní protokolu DHCP je uveden na obrázku 6.1. Každá klientská stanice prochází při získávání adresy následujícími stavy:

- *Inicializace* - stanice nemá přidělenu žádnou adresu (výchozí stav)
- *Požadování* identity - stanice se pokouší o získání adresy
- *Identita přidělena* - stanici byla přidělena adresa (IPv4)
- *Obnovení identity* - pokus o obnovení adresy



Obrázek 6.1: Stavový diagram funkce IRI-IIF protokolu DHCP

Činnost bloku IRI-IIF se řídí podle následujících pravidel:

- Počátečním stavem je *Inicializace*, ve kterém klientská stanice prozatím nemá přidělenu žádnou adresu.
- Příchodem požadavku o přidělení adresy (zpráva *DHCP Request*) je proveden přechod do stavu *Požadování identity* a je generována zpráva typu *IRI Report (access attempt)*. Jako odpověď na tento požadavek může přijít buď potvrzující zpráva *DHCP Ack* nebo odmítnutí *DHCP Nack*. Při potvrzení je proveden přechod do stavu *Identita přidělena* a současně je na výstupu generována zpráva *IRI Begin*. V opačném případě je proveden přechod zpět do stavu *Inicializace* a je generována zpráva typu *IRI Report (access failed)*.
- Ve stavu *Identita přidělena* může klientská stanice sama ukončit platnost přidělené adresy (zasláním zprávy *DHCP Release*) nebo si může adresu ponechat, dokud nevyprší její *Lease*

Událost	Použité identifikátory	Typ zprávy IRI
Hledání DHCP serveru (DHCP discover)	MAC nebo DHCPClientID	REPORT
Požadavek o IP adresu (DHCP request)	MAC nebo DHCPClientID	REPORT
Přidělení IP adresy (DHCP ack)	MAC nebo DHCPClientID, IPv4	BEGIN
Prodloužení *lease time* pro IP adresu (DHCP ack)	MAC nebo DHCPClientID, IPv4	CONTINUE
Přiřazení IP adresy k jiné MAC než na kterou je aktivní záznam (DHCP ack)	MAC nebo DHCPClientID, IPv4	END pro starou MAC, BEGIN pro novou MAC
Zamítnutí přidělení IP adresy (DHCP nack)	MAC nebo DHCPClientID	REPORT
Vypršení *lease time* pro IP adresu (založeno na interním časovači (timer) v modulu)	MAC nebo DHCPClientID, IPv4	END
Uvolnění IP adresy (DHCP release)	MAC nebo DHCPClientID, IPv4	END

Tabulka 6.1: Zprávy IRI generované modulem DHCP.

time. V obou těchto případech se generuje zpráva typu *IRI End*. Alternativně může stanice požádat o prodloužení stávající adresy (zpráva *DHCP Request*). V tomto případě je proveden přechod do stavu *Obnovení identity*.

- V případě, že pokus o obnovení identity dopadne kladně (zpráva *DHCP Ack*), je proveden přechod do stavu *Identita přidělena* a je generována zpráva typu *IRI Continue*. V případě, že stanici nebyla adresa prodloužena, má stanice právo si tuto adresu ponechat alespoň do okamžiku, než vyprší její platnost. Provede se proto přechod do stavu *Identita přidělena* avšak bez jakéhokoliv generování IRI zpráv.

Tabulka 6.1 zachycuje zprávy IRI generované modulem pro DHCP.

6.2 RADIUS

6.2.1 Popis protokolu

Protokol *Remote Authentication Dial In User Service* (RADIUS) [67] slouží pro autentizaci, autorizaci a účtování. Z pohledu IRI-IIF je zajímavá především autentizace, která může být v některých případech doplněna i o přidělení IP adresy klientovi. Autentizace se provádí buď pomocí uživatelského jména a hesla nebo dle portu, ze kterého přišla žádost o připojení. RADIUS zprávy jsou často zasílány skrze *Remote Access Service* (RAS), přičemž komunikace mezi RADIUS serverem a RAS je realizována pomocí PPP protokolu (například PPPoE). V závislosti na připojení systému pro zákonné odposlechy pak blok IRI-IIF může sledovat buď zprávy od koncových klientů nebo zprávy zasílané mezi RAS a RADIUS serverem. Protokol RADIUS provádí autentizaci a případné přidělení IP adresy v následujících krocích:

1. Klient (nebo RAS) zašle požadavek o přístup (*Access Request*) přímo na adresu RADIUS serveru. Součástí tohoto požadavku jsou obvykle přihlašovací údaje v podobě RADIUS loginu a hesla.

2. RADIUS server zašle klientovi zprávu o povolení přístupu (*Access Accept*) s konfiguračními údaji popř. IP adresou, maskou sítě apod.

V případě zamítnutí přístupu zašle RADIUS server zprávu *Access Reject*. Za neúspěšnou autentizaci je považována také situace, kdy RADIUS server neodpoví klientovi ani po opakovaném zaslání požadavku (*Access Request*).

Platnost autentizace nebo přidělené adresy zde není časově omezena, neboť se předpokládá, že klient musí vždy pro přístup do sítě žádat povolení u RAS nebo RADIUS serveru.

RADIUS server může v průběhu autentizace po klientovi požadovat i dodatečné informace jako jsou např. sekundární heslo, pin apod. skrze opakované zasílání zprávy *Access Challange* a až na základě těchto informací zaslat klientovi *Access Accept* nebo *Access Reject*.

6.2.2 Činnost IRI-IIF

Blok IRI-IIF sleduje komunikaci RADIUS protokolu a páruje zprávy *Access Request* s odpověďmi *Access Accept* nebo *Access Reject*. Přidělené adresy se zapisují do tabulky a generují se odpovídající IRI zprávy. Přidělené adresy v tabulce zůstávají do té doby, dokud se daná stanice opět nepokusí o autentizaci a přidělení adresy nebo danou adresu nezíská jiná stanice. V rámci modulu IRI-IIF prochází každá klientská stanice následujícími stavy (viz diagram na obrázku 6.2):

- *Inicializace* - stanice není autentizována a také nemá přidělenu žádnou adresu (výchozí stav)

- *Požadování identity* - stanice se pokouší o autentizaci a případné získání IP adresy

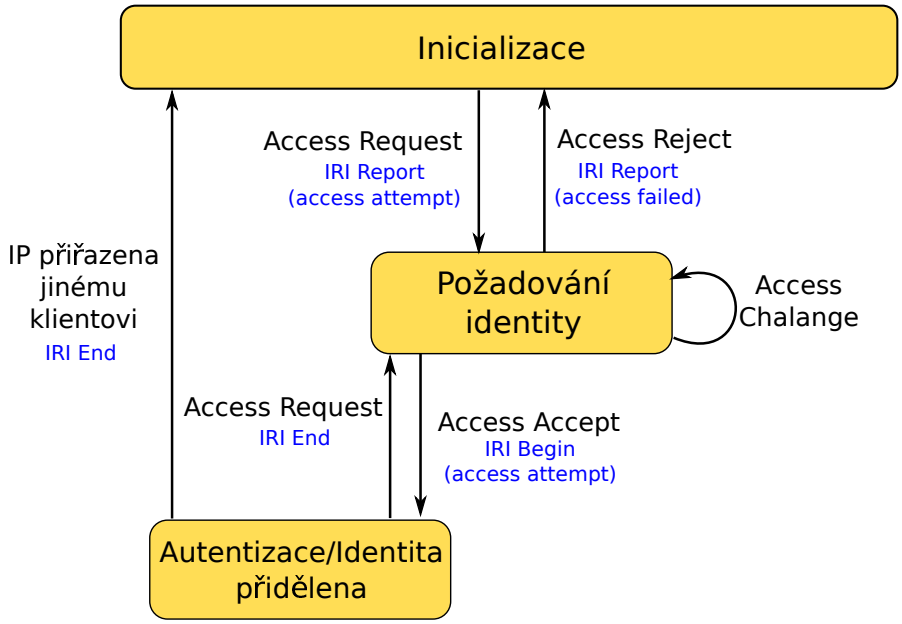
- *Autentizace/Identita přidělena* - stanici byla udělena autentizace popř. přidělena adresa (IPv4)

Činnost bloku IRI-IIF se řídí podle následujících pravidel:

- Počátečním stavem je *Inicializace*, ve kterém klientská stanice prozatím nemá přístup do sítě.

- Příchodem požadavku o přidělení přístupu do sítě (zpráva *Access Request*) je proveden přechod do stavu *Požadování identity* a je generována zpráva typu *IRI Report(access attempt)*. Následně se čeká na zachycení odpovědi a to zprávu *Access Accept* nebo *Access Reject*. V případě kladné odpovědi se generuje zpráva *IRI-Begin* a klientská stanice přechází do stavu *Autentizace/Identita přidělena*. V opačném případě je zaslána zpráva *IRI Report (access failed)* a proveden přechod do stavu *Inicializace*. Pokud byla v rámci autentizace klientovi přidělena i IP adresa, je tato adresa součástí zprávy *IRI-Begin*. Naopak zaslání zprávy *IRI-Begin* bez IP adresy znamená, že klientovi byla udělena pouze autentizace.

- Ve stavu *Autentizace/Identita přidělena* klientská stanice zůstává dokud není daná adresa přidělena jinému klientovi nebo dokud klient opět nepožádá o autentizaci. V obou těchto případech zašle modul zprávu *IRI End*. V závislosti na tom, zda byla klientovi v rámci autentizace přidělena i IP adresa je tato adresa součástí i zaslané zprávy *IRI End*.



Obrázek 6.2: Stavový diagram funkce IRI-IIF protokolu RADIUS

Tabulka 6.2 zachycuje zprávy IRI generované modulem pro RADIUS.

Samotný protokolu RADIUS nemá žádné povinné zprávy spojené s ukončením autentizace. To vede k těžkostem spojených se zasíláním IRI zprávy typu *end*. K problémům může docházet především při spojení přiřazování IP adres v kombinaci několika mechanismů. Uvažujme např. následující scénář:

1. Uživatel A se autentizuje a pomocí protokolu RADIUS je mu přiřazena IP adresa 192.168.1.1. Modul pro RADIUS ohlásí pomocí zprávy IRI begin počátek sezení jádru IRI-IIF.

2. Uživatel A přestane komunikovat, ale nevyšle žádnou zprávu protokolem RADIUS. Z pohledu jádra IRI-IIF však nedojde k žádné změně, protože od modulu zkoumajícího zprávy protokolu RADIUS nedostane žádnou zprávu.

3. Uživatel B dostane přiřazenu IP adresu pomocí protokolu DHCP. Modul protokolu DHCP toto přiřazení detekuje a oznámí jádru IRI-IIF. Jádro IRI-IIF vytvoří vazbu mezi IP adresou a MAC adresou síťového rozhraní uživatele B.

Tím však chybně vznikne zprostředkovaná vazba mezi přihlašovacím jménem protokolu RADIUS uživatele A a MAC adresou protokolu B. To může mít za následek neoprávněné zachycení dat uživatele B v rámci odposlechu uživatele A.

Pro předcházení výše uvedené situace a obdobných scénářů je nutné upravit ukončovací podmínku protokolu RADIUS na základě znalosti chování konkrétní sítě. Současná implementace je spolehlivá v sítích, ve kterých jsou IP adresy jednou přidělené protokolem RADIUS vždy přiděleny pouze tímto protokolem.

Událost	Použité identifikátory	Typ zprávy IRI
Požadování přístupu (ACCESS_REQUEST)	RADIUS_Login, MAC	REPORT
Povolení přístupu s přidelením IPv4 nebo IPv6 (ACCESS_ACCEPT)	RADIUS_Login, MAC, IP nebo IPv6	BEGIN
Povolení přístupu BEZ přidělení IPv4 nebo IPv6 (ACCESS_ACCEPT)	RADIUS_Login, MAC	BEGIN
Zamítnutí přidělení IP adresy (ACCESS_REJECT)	RADIUS_Login, MAC	REPORT

Tabulka 6.2: Zprávy IRI generované modulem RADIUS.

6.3 PPPoE

6.3.1 Popis protokolu

Point-to-Point Protocol over Ethernet (PPPoE) [49] je variantou protokolu *Point-to-Point Protocol* (PPP), který slouží pro vytváření spojení mezi právě dvěma uzly v síti. Protokol PPP se nejčastěji používá na síťové spoje, ve kterých se více než dva uzly nemohou vyskytnout, např. sériové linky. Primární úlohou protokolu PPPoE je vytváření spojení typu point-to-point skrze sdílené médium. V některých konfiguracích je však tento protokol použit i pro přidělování adres, a proto je nezbytné jej v rámci systému pro zákonné odposlechy analyzovat a zpracovávat. PPPoE protokol podporuje přidělování jak IPv4, tak i IPv6 adres, postup přidělení se však pro jednotlivé verze mírně liší (podrobněji bude popsáno dále). Analýzou tohoto protokolu je možné získat kromě případné adresy i přihlašovací údaje klienta jako je PPP login. V případě, že jsou tyto přihlašovací údaje pro každého klienta unikátní, mohou sloužit i jako jednoznačný identifikátor pro jeho dohledání v síti.
Postup při navázání PPPoE spojení je následující:

1. Nejprve pomocí *Link Control Protocol* (PPP LCP) klient kontaktuje *Broadband Remote Access Server* (BRAS) a dohodne se na způsobu přenosu dat a případné autentizaci. (BRAS je síťové zařízení na straně poskytovatele, které vytváří s klientem spojení typu point-to-point a pro tento účel může vyžadovat autentizaci klienta.)

2. Pokud je autentizace BRASem vyžadována, klient se autentizuje. Na výběr jsou obvykle autentizační metody PAP [48] a CHAP [74]. U obou metod je od klienta vyžadován PPP login a heslo. V případě neúspěšné autentizace se spojení PPPoE okamžitě ukončí.

3. Klient může požádat o přidělení IP adresy. Tento krok se liší pro IPv4 a IPv6:

   (a) *IPv4* - Pomocí protokolu *PPP IP Control Protocol* (IPCP) BRAS sdělí klientovi svoji IP adresu a případně i IP adresu přidělenou klientovi. Proces přidělení adresy klientovi začíná tak, že klient zašle zprávu *IPCP Configure-Request*, ve které žádá o jím zvolenou adresu. Může se jednat o libovolnou adresu nebo speciální případ adresy s hodnotou 0.0.0.0 (v situaci, kdy klient neví jakou adresu zvolit). BRAS na tento požadavek reaguje zprávou *IPCP Configure-Nak* nebo *Configure-Ack* v závislosti na tom, zda s danou adresou souhlasí či nikoliv. V případě, že nesouhlasí odpoví zprávou *Configure-Nak*, jejíž součástí je i IP adresa, kterou BRAS klientovi navrhuje. Klient pak tuto

adresu odešle s novou zprávu *Configure-Request* a BRAS tento požadavek potvrdí srkze *Configuration-Ack*. V případě, že BRAS pomocí protokolu PPP adresy nepřiděluje, doporučí klientovi použití adresy 0.0.0.0, čímž mu dává najevo, aby se pokusil získat IP adresu až po dokončení navázání PPP spojení skrze protokol vyšší vrstvy (např. DHCP).

(b) *IPv6* - Pro IPv6 se využívá protokol *PPP IPv6 Control Protocol* (IPv6CP). Na rozdíl od IPCP se klient s BRASem nedohadují na IP adrese, ale předají si pouze identifikátory rozhraní. Pomocí těchto identifikátorů si pak odvodí linkovou IPv6 adresu druhého uzlu. Identifikátory rozhraní jsou přenášeny pomocí zpráv *IPCP Configure-Request* stejně jako IPv4 adresy v případě IPCP. Na rozdíl od IPCP ale obě strany vždy souhlasí s navrhovaným identifikátorem rozhraní a potvrdí si jej pomocí zprávy *IPv6CP Configure-Ack*.

Po těchto krocích je PPPoE spojení mezi BRASem a klientem úspěšně navázáno. V případě, že klient nezískal IPv4 nebo IPv6 adresu a vyžaduje ji, tak právě v tomto okamžiku může využít protokoly vyšších vrstev. Adresy získané z vyšší vrstvy jsou pak nezávislé na protokolu PPPoE, a proto i při ukončení PPPoE spojení zůstávají nadále platné.

Posledním krokem v rámci PPPoE relace je samotné ukončení spojení. Vlastností protokolu PPPoE je, že obě strany periodicky odesílají zprávy pro udržení spojení (*keepalive*), na které si vzájemně odpovídají. V případě, že klientovi nebo BRASu nepřijde odpověď na *keepalive* zprávu, odešle zprávu typu *PPPoE Active Discovery Termination (PADT)* a ukončí spojení. Stejná zpráva se odesílá i při běžném (vyžadovaném) ukončení spojení nebo při ukončení spojení z důvodu neúspěšné autentizace apod.

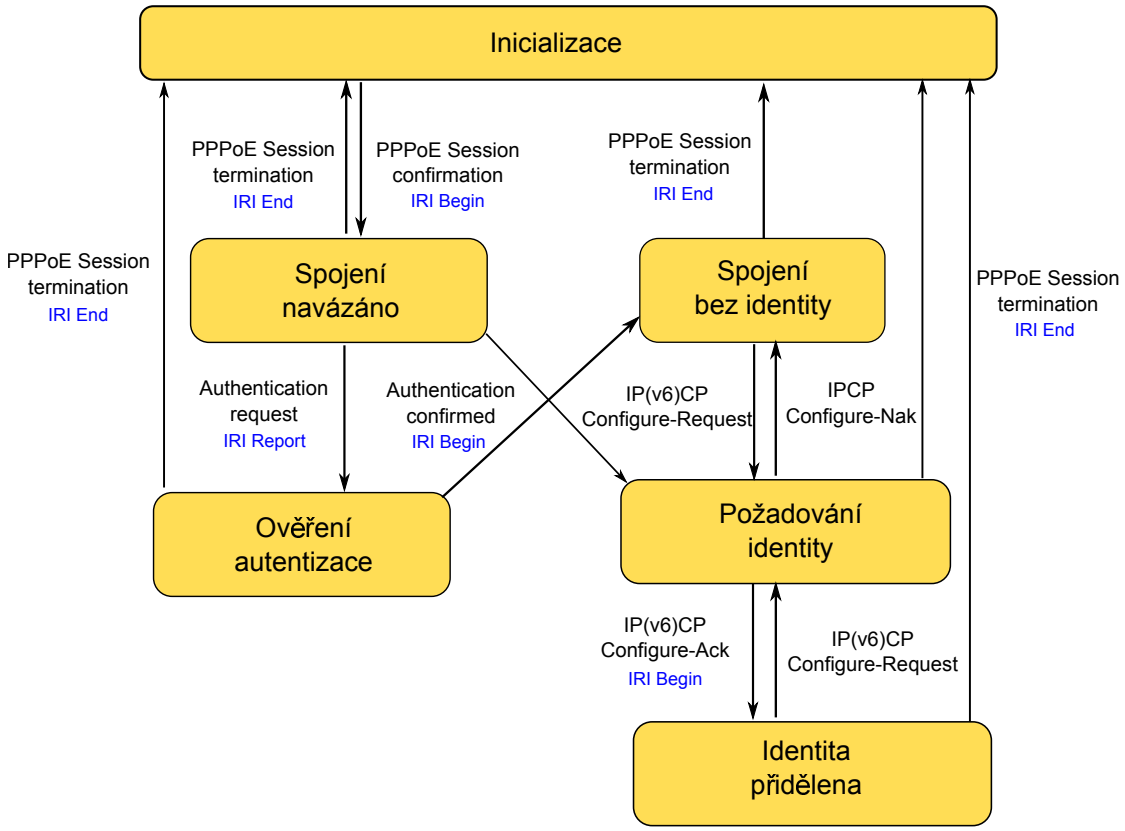
6.3.2 Činnost IRI-IIF

Modul IRI-IIF analyzuje zprávy protokolu PPPoE a na jejich základě si udržuje tabulku aktivních spojení spolu s přiřazenými IP adresami. Stavový diagram funkce IRI-IIF je uveden na obrázku 6.3. Každá klientská stanice prochází při navazování PPPoE spojení následujícími stavy:

- *Inicializace* - stanice nemá navázáno žádné PPP spojení ani přidělenu IPv4 nebo IPv6 adresu
- *Spojení navázáno* - stanice navázala spojení s BRASem
- *Ověření autentizace* - stanice odeslala přihlašovací údaje na BRAS a probíhá jejich ověřování
- *Spojení bez identity* - stanice navázala spojení s BRASem, ale ještě nezískala IP adresu
- *Požadování identity* - stanice žádá BRAS o přidělení IPv4 nebo IPv6 adresy
- *Identita přidělena* - stanici byla přidělena IPv4 nebo IPv6 adresa (popř. obojí)

Činnost bloku IRI-IIF se řídí podle následujících pravidel:

- Počátečním stavem je *Inicializace*, ve kterém klientská stanice ještě nenavázala PPPoE spojení, a proto nemá ani přiřazenu žádnou IP adresu.
- Po přijetí zprávy *PPPoE Active Discovery Session confirmation (PADS)* se detekuje úspěšné navázaní spojení mezi klientem a BRASem a provede se přechod do stavu *Spojení navázáno*.



Obrázek 6.3: Stavový diagram funkce IRI-IIF protokolu PPPoE

- Následně se BRAS a klient dohadují na způsobu autentizace a některých dalších parametrech daného PPPoE spojení. Když se obě strany nedohodnou zašle některá ze stran zprávu *PPPoE Active Discovery Session termination (PADT)*, spojení se ukončí a v rámci modulu se provede přechod zpět do stavu *Inicializace*.
- Pokud BRAS vyžaduje autentizaci (viz předchozí bod), musí mu klient zaslat požadované informace obvykle v podobě přihlašovacího jména a hesla. Pro přenos těchto údajů se využije protokol PAP [48] nebo CHAP [74]. Zachycením zprávy *Authentication* request (*PPP PAP Authenticate-Request* pro PAP, *PPP CHAP Response* pro CHAP) získá modul přihlašovací jméno klienta. Modul toto jméno považuje za typ identifikátoru uživatele, odešle jej společně se zprávou *IRI Report (access-attempt)* a přechází do stavu *Ověření autentizace*.
- Úspěšnou autentizaci modul detekuje přijetím zprávy *Authentication confirmed* (*PPP PAP Authenticate-ACK* pro PAP, *PPP CHAP Success* pro CHAP). Tímto způsobem se také potvrdí, že zaslané přihlašovací údaje jsou platné a modul odešle první zprávu typu *IRI Begin* informující o úspěšné autentizaci (bez přidělené IP adresy). Modul následně provede přechod do stavu *Spojení bez identity*.
- Pokud nebude autentizace úspěšná (klient např. zadá nesprávné přihlašovací údaje), potom BRAS vynutí ukončení spojení skrze zprávu *PADT* a modul provede přechod zpět do stavu

Událost	Typ zprávy IRI	Popis události	Použité identifikátory
Navázání spojení PPP	BEGIN	Klient navázal spojení s BRASem	PPP Session, MAC
Uživatel se pokouší autentizovat	REPORT	Zjištění přihlašovacího jména uživatele	PPP Session, MAC, PPP Login
Uživatel se úspěšně autentizoval	BEGIN	Uživatel se úspěšně autentizoval	PPP Session, MAC, PPP Login
Zachycení zprávy IP(v6)CP obsahující platnou IP adresu pro klienta	BEGIN	Uživatelovi byla přiřazena IP adresa	PPP Session, IP adresa
Konec IP(v6)CP	END	IP adrese skončila platnost	PPP Session, IP adresa
Zachycení PPPoED - session termination	END	Konec sezení	PPP Session, MAC, pokud proběhla autentizace PPP Login

Tabulka 6.3: Zprávy IRI generované modulem PPPoE.

Inicializace.

- V případě, že autentizace nebyla vyžadována nebo byla a dopadla úspěšně se klient pokusí získat IPv4 nebo IPv6 adresu. Proces přidělování obou těchto adres probíhá zcela nezávislé skrze protokoly IPCP a IPCPv6. Pokus o získání IPv4 nebo IPv6 adresy modul detekuje přijetím zprávy *IP(v6)CP Configure-Request*. Úspěšné přidělení adresy detekuje přijetím zprávy *IP(v6)CP Configure-Ack*, zatímco zpráva *IPCP Configure-Nak* informuje o neúspěšném přidělení adresy (pouze u IPv4) a nabízí klientovi jinou IP adresu, o kterou by mohl při příštím *IPCP Configure-Reguest* požádat. Při úspěšném přidělení IP adresy přechází modul do stavu *Identita přidělena* a generuje zprávu *IRI Begin* obsahující danou IP adresu. Při neúspěšném přidělení se modul vrací do předchozího stavu a posílá zprávu *IRI Report (access-reject)*. Výjimku k tomuto scénáři tvoří pouze situace, kdy je detekována zpráva *IPCP Configuration Ack* s IPv4 adresou nastavenou na hodnou 0.0.0.0. BRAS touto zprávou dává najevo, že adresy nepřiděluje a klient o ni bude moci požádat pouze protokolem vyšší vrstvy. Zprávu *IRI Begin* v těchto případech modul neposílá. V rámci protokolu IPCPv6 pak není součástí zasílaných zpráv přímo IPv6 adresa, ale pouze identifikátor rozhraní, který slouží k odvození lokální IPv6 adresy.
- Ve stavu *Identita přidělena* se může klient pokusit získat i druhou IP adresu. Čili, když nejprve získal IPv4, může se pokusit získat IPv6 a naopak. Postup zůstává stejný.
- Posledním možným krokem stavového automatu je ukončení PPPoE spojení, které modul detekuje zachycením zprávy *PADT*. Při ukončení spojení odešle modul ke každé dříve zaslané zprávě *IRI Begin* příslušnou zprávu *IRI End*. Nejvíce tedy může poslat až tři *IRI End* zprávy odpovídající úspěšné autorizaci, přidělení IPv4 adresy a přidělení IPv6 adresy. Pro každé PPPoE spojení si tedy musí modul uchovávat navíc informaci o tom, které z IP adres byly přiřazeny a zda byla udělena autorizace.

Tabulka 6.3 zachycuje zprávy IRI generované modulem pro PPPoE.

6.4 DHCPv6

6.4.1 Popis protokolu

Dynamic Host Configuration Protocol for IPv6 (DHCPv6) [7] se používá pro dynamické přidělování IPv6 adres a dalších parametrů sítě. Tento proces probíhá obdobně jako u protokolu DHCP. Jedná se tedy o časově omezené přidělení adresy klientovi centrálním prvkem (DHCPv6 serverem). DHCPv6 server má k dispozici rozsah, ze kterého přiřazuje adresy klientům. Na rozdíl od klasického DHCP se v DHCPv6 nepřiřazuje adresa na základě MAC adresy, ale je zde využit nový identifikátor *DHCPv6 Unique Identifier* (DUID). V závislosti na konfiguraci rozlišujeme několik typů DUID identifikátorů. Primární vlastností všech DUID identifikátorů však je, že jejich hodnota musí být pro daný počítač jedinečná (nicméně není zajištěná stálost a DUID se může měnit např. v závislosti na provozovaném operačním systému), a proto je i unikátní v rámci monitorované sítě. Přiřazení adresy protokolem DHCPv6 probíhá v následujících krocích:

1. Klient zašle zprávu *DHCPv6 Solicit* na adresu DHCPv6 severů (skupinová adresa ff02::1:2).
2. Server odpoví klientovi zprávou *DHCPv6 Advertise* s nabízenou adresou.
3. Klient požádá o přidělení nabízené adresy zasláním zprávy *DHCPv6 Request* (na skupinovou adresu ff02::1:2).
4. Server zašle klientovi zprávu *DHCPv6 Reply*, která obsahuje buď kladnou nebo zápornou odpověď.

Klientovi je nejčastěji přiřazena IPv6 adresa s časovým omezením a to hned dvěma intervaly. První interval (*Preferred lifetime*) udává plnohodnotné přidělení adresy, tj. klient není v průběhu této doby v používání adresy nijak omezen. Po vypršení tohoto intervalu by však již klient neměl vytvářet nová spojení s danou IPv6 adresou, ale stále může adresu používat pro již otevřená spojení. Druhý interval (*Valid lifetime*) pak označuje celkovou dobu pro přidělení IPv6 adresy, po které již klient nesmí adresu používat. Interval vzniklý mezi *Preffered lifetime* a *Valid lifetime* je dán klientovi proto, aby se připravil na situaci, kdy mu bude adresa odebrána, popřípadě, aby informoval ostatní, že přechází na jinou adresu. Pro udržení adresy po delší dobu si klient může požádat o její prodloužení (obou intervalů). Žádost o prodloužení doby přidělení adresy probíhá v následujících krocích:

1. Klient zašle serveru zprávu *DHCPv6 Renew* (na skupinovou adresu ff02::1:2).
2. Server, který klientovi adresu přidělil odpoví zprávou *DHCPv6 Reply* s potvrzením prodloužení času přidělené adresy.

Pokud server klientovi na *DHCPv6 Renew* neodpoví (nebo zašle *DHCPv6 Reply* s negativní odpovědí), nabízí protokol DHCPv6 klientovi ještě možnost zažádat o prodloužení času přidělené adresy u jiného serveru. Tento proces probíhá následovně:

1. Klient zašle zprávu *DHCPv6 Rebind* na adresu DHCPv6 serverů (na skupinovou adresu ff02::1:2).
2. Server schopný vyhovět požadavku odpoví zprávou *DHCPv6 Reply* s potvrzením o prodloužení času přidělené adresy.

V případech, kdy se klient přepojí na jinou síť (např. z bezdrátové na pevnou linku), provede restart počítače nebo se probudí z úsporného režimu a není si jistý, zda může stále využít přidělenou IPv6 adresu, potom je vhodné, aby zaslat serveru zprávu *DHCPv6 Confirm*. Na základě této zprávy mu server zašle odpověď v podobě *DHCPv6 Reply*.

Protokol DHCPv6 umožňuje také distribuci prefixů ostatním prvkům sítě. Zde se předpokládá hlavní využití při konfiguraci směrovačů u koncových uživatelů tak, aby poskytovatel nemusel manuálně konfigurovat každý směrovač umístěný u koncového uživatele. Po získání takového prefixu může koncový směrovač začít přidělovat adresy s daným prefixem pomocí protokolu SLAAC nebo DHCPv6. Přiřazení prefixu probíhá ve stejných krocích jako přidělení IP adresy a je dokonce možné současně získat jak samostatnou IP adresu, tak i prefix. Podobně jako u IP adresy i platnost přiděleného prefixu je časově omezena, přičemž tato platnost může být opět prodloužena.

Obdobně jako DHCP má i DHCPv6 možnost přidělovat adresy pomocí *relay* agentů komunikujících s centrálním DHCPv6 serverem. Pro klienta je komunikace plně transparentní (klient nepozná, zda komunikuje s *relay agentem* nebo přímo serverem). Zpráva zasílaná mezi *relay agentem* a serverem zahrnuje původní DHCPv6 zprávu, ke které je přidána nová hlavička obsahující informace potřebné pro *relay*. Modul IRI-IIF pro analýzu protokolu DHCPv6 proto postupuje dle stejného algoritmu, ať už je zapojen přímo mezi klientem a *relay agentem* nebo mezi *relay agentem* a serverem.

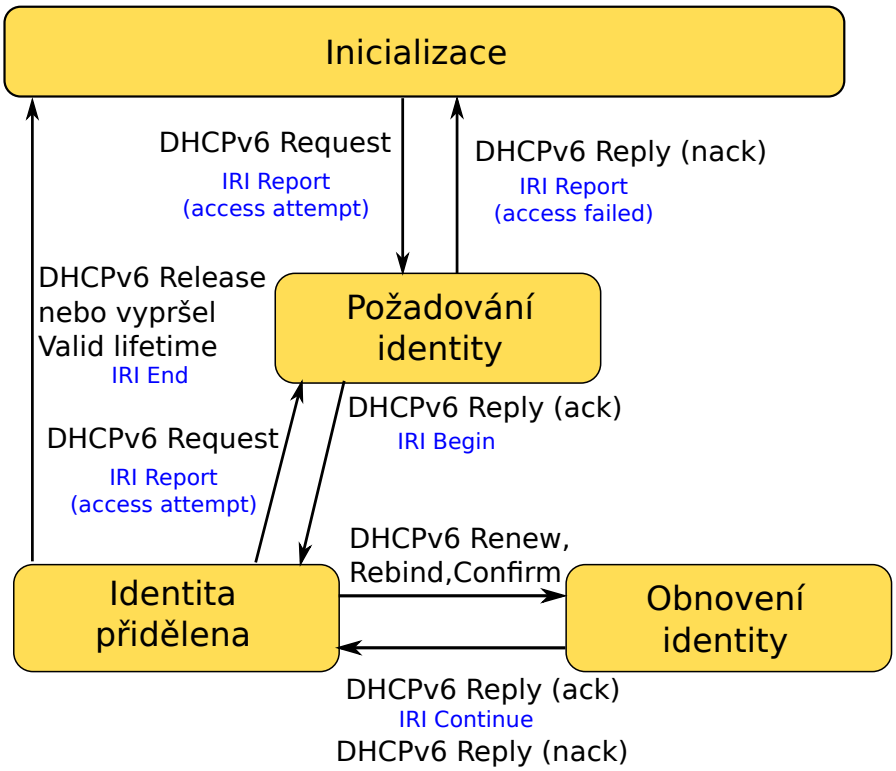
6.4.2 Činnost IRI-IIF

Modul IRI-IIF analyzuje příchozí DHCPv6 zprávy a na jejich základě si udržuje aktuální tabulku přidělených IPv6 adres společně s jejich dobou platnosti. S identifikátorem DUID modul pracuje jako s hexadecimální hodnotou a jeho obsah resp. typ dále nezkoumá. Protože je distribuce prefixů velmi podobná jako distribuce adres, zpracovává modul obě možnosti stejným způsobem. Stavový diagram pro analýzu protokolu DHCPv6 je znázorněn na obrázku 6.4. Každá klientská stanice prochází při získávání IPv6 adresy následujícími stavy:

- *Inicializace* - stanice nemá prozatím přidělenu žádnou adresu (výchozí stav)
- *Požadování identity* - stanice se pokouší o získání adresy
- *Identita přidělena* - stanici byla přidělena IPv6 adresa
- *Obnovení identity* - pokus o prodloužení adresy

Činnost bloku IRI-IIF se řídí podle následujících pravidel:

- Počátečním stavem je *Inicializace*, ve kterém klientská stanice prozatím nemá přidělenu žádnou IPv6 adresu nebo prefix.
- Příchodem požadavku o přidělení adresy (zpráva *DHCPv6 Request*) je proveden přechod do stavu *Požadování identity* a je generována zpráva typu *IRI Report(access attempt)*. Jako odpověď je zaslána zpráva *DHCPv6 Reply*, která obsahuje buď potvrzení nebo odmítnutí tohoto požadavku. Při potvrzení je proveden přechod do stavu *Identita přidělena* a současně je na výstupu generována zpráva *IRI Begin*. V opačném případě je proveden přechod zpět do stavu *Inicializace* a je generována zpráva *IRI Report(access failed)*.
- Ve stavu *Identita přidělena* může klientská stanice sama ukončit platnost přidělené adresy (zasláním zprávy *DHCPv6 Release*) nebo si může adresu ponechat, dokud nevyprší její platnost. V obou těchto případech se generuje zpráva typu *IRI End*. Alternativně může stanice



Obrázek 6.4: Stavový diagram funkce IRI-IIF protokolu DHCPv6

požádat o potvrzení přidělené adresy (skrze zprávu *DHCPv6 Confirm*) nebo o její prodloužení (zprávou *DHCPv6 Renew* popř. *DHCPv6 Rebind*). V těchto případech je proveden přechod do stavu *Obnovení identity*.

- Pokud dopadne pokus o potvrzení/prodloužení adresy kladně, potom je proveden přechod do stavu *Identita přidělena* a je generována zpráva *IRI Continue*. V opačném případě se opět provede přechod do stavu *Identita přidělena*, avšak na výstupu není generována žádná IRI zpráva (klient si může stávající adresu ponechat až do vypršení její platnosti).

Na přidělení prefixu se nahlíží stejným způsobem jako na přidělení adresy. V závislosti na možnostech připojení systému pro zákonné odposlechy v rámci sítě poskytovatele pak bude možné odposlouchávat jednotlivé adresy nebo celé prefixy popř. obojí.

Tabulka 6.4 zachycuje zprávy IRI generované modulem pro DHCPv6.

6.5 Objevování sousedů (ND) včetně bezstavové autokonfigurace adres (SLAAC)

6.5.1 Popis protokolu

Bezstavová autokonfigurace adres (*Stateless Address Autoconfiguration* – SLAAC) [55] slouží pro automatické přidělení IPv6 adresy koncové stanici. Komunikace probíhá mezi klientem a sme-

Událost	Použité identifikátory	Typ zprávy IRI
Hledání DHCP serveru (DHCPv6 solicit)	DUID	REPORT
Požadování IP adresy (DHCPv6 request)	DUID	REPORT
Potvrzení adresy ze zprávy DHCPv6 confirm (DHCPv6 confirm)	DUID	REPORT
Přidělení IP adresy (DHCPv6 reply)	DUID, IPv6	BEGIN
Prodloužení „valid lifetime“ pro IPv6 adresu (DHCPv6 reply)	DUID, IPv6	CONTINUE
Potvrzení adresy ze zprávy DHCPv6 confirm (DHCPv6 reply)	DUID, IPv6	CONTINUE
Zamítnutí přidělení IP adresy (DHCPv6 reply)	DUID	REPORT
Vypršení *valid lifetime* pro IP adresu (založeno na interním časovači v modulu)	DUID, IPv6	END
Uvolnění IP adresy (DHCPv6 release)	DUID, IPv4	END
Uvolnění IP adresy na základě zjištění duplicity (DHCPv6 decline)	DUID, IPv4	END

Tabulka 6.4: Zprávy IRI generované modulem DHCPv6.

rovačem pomocí protokolu ICMPv6. Na rozdíl od jiných protokolů pro přiřazování adres však není klientovi adresa přidělena přímo SLAAC serverem, ale jsou mu zaslány pouze konfigurační udaje, na jejichž základě si klient IPv6 adresu vygeneruje sám. Mezi tyto konfigurační údaje patří zejména prefix dané sítě (horní část IPv6 adresy) a doba platnosti tohoto prefixu. Postup pro přidělení IPv6 adresy probíhá v následujících krocích:

1. Klient si nejprve vygeneruje tzv. *lokální IPv6 adresu* tak, že si sám zvolí (např. náhodně) identifikátor rozhraní (spodní část IPv6 adresy o velikosti 64 bitů) a horní část nastaví na prefix fe80::/64. Po zvolení identifikátoru si na jeho základě odvodí i tzv. *solicited-node multicast* adresu a přihlásí se do této skupiny. Zmíněná skupinová adresa je vytvořena spojením prefixu ff02::1:ff00:0000/104 se spodními 24-mi bity IPv6 adresy. Důvod pro vytvoření adresy a přihlášení klienta do této skupiny spočívá ve využití mechanizmu objevování sousedů (*Neighbor Discovery* – ND), kdy se klient dotazuje všech uživatelů dané skupiny, zda některý z nich již nepoužívá takto vygenerovanou adresu. Podrobnější popis protokolu ND je uveden níže.

2. Aby si klient mohl přiřadit také globální IPv6 adresu, musí nejprve znát prefix podsítě, ve které se nachází. O tuto informaci může požádat nejbližší směrovač(e) opět skrze protokol ND. Získání informace o prefixu probíhá v následujících krocích:

   (a) Klient zašle žádost o konfigurační údaje skrze zprávu *Router Solicitation*. Tato zpráva je odeslána na skupinovou (multicast) adresu ff02::2, kde jsou zařazeny všechny směrovače v dané podsíti.

   (b) Směrovač(e) odpovídají zprávou *Router Advertisement* obsahující informace o prefixu podsítě a době jeho platnosti.

3. Jakmile má klient k dispozici prefix podsítě, vygeneruje si globální adresu buď na základě lokální adresy - záměnou prefixu lokálního adresy za prefix dané podsítě nebo zcela nezávisle na lokální adrese. V prvním případě se identifikátor rozhraní (spodní část) globální adresy shoduje s identifikátorem rozhraní lokální adresy. Ve druhém případě se identifikátory rozhraní navzájem liší. Pro takto vytvořenou globální adresu je nezbytné se opět přihlásit do skupiny odvozené z *solicited-node multicast* adresy a ověřit, zda již není používána jiným klientem sítě.

Ověření lokální i globální IPv6 adresy (*Duplicate Address Detection - DAD*) pomocí protokolu ND probíhá v následujících krocích:

1. Klient vloží vygenerovanou adresu do zprávy *Neighbor Solicitation* a tuto zprávu zašle na skupinovou (multicast) adresu odvozenou od vygenerované IPv6 adresy.

2. Pokud klientovi do určitého časového intervalu nepřijde odpověď ve formě zprávy *Neighbor Advertisement* oznamující, že vygenerovaná adresa je již používána, považuje klient adresu za unikátní a začne ji používat.

Uvedeným způsobem si může klient vygenerovat i více globálních IPv6 adres. Doba platnosti přiřazené globální adresy je omezena na hodnotu zaslanou v rámci zprávy *Router Advertisement (RA)* v položce *Valid lifetime*. Klient si však může platnost přiřazené adresy prodlužovat po dobu, kdy směrovače periodicky zasílají *RA*.

Mechanismus, kterým by klient oznámil ostatním síťovým uzlům, že již svoji adresu nebude používat není bohužel v protokolu definován. Pro účely monitorování sítě je však potřeba znát informaci, kdy již stanice danou adresu nepoužívá. Jeden ze způsobů řešení tohoto problémů

spočívá ve využití principu multicastu, kdy je stanice přihlášena do skupiny jen tak dlouho, má-li to pro ni význam. Pro protokol SLAAC to tedy znamená, že klient je přihlášen v *solicited node multicast* skupině jen tak dlouho, dokud aktivně využívá aspoň jednu adresu náležící do této skupiny. Jinými slovy, za konec platnosti IPv6 adresy lze považovat okamžik, kdy již klient není součástí multicastové skupiny. Pro detekci tohoto stavu lze využít chování směrovačů, které se sami automaticky dotazují na všechny multicastové skupiny a zjišťují, zda je v nich přihlášen alespoň jeden klient. Tyto zprávy jsou zasílány na skupinu adresu ff::16 reprezentující všechny směrovače podporující multicast. Trvalým přihlášením se do této skupiny a následným odposloucháváním zasílaných zpráv lze tak nepřímo odvodit platnost jednotlivých IPv6 adres.

6.5.2 Činnost IRI-IIF

Blok IRI-IIF analyzuje ICMPv6 pakety a na jejich základě udržuje aktuální tabulku přidělených IPv6 adres společně s jejich stavem. Oproti jiným protokolům pro přidělování IP adres jako jsou např. DHCP, RADIUS nebo PPP je SLAAC specifický tím, že každá klientská stanice může mít přiděleno několik IPv6 adres současně. V rámci každé takto přidělené adresy probíhá nezávisle proces ověřování skrze protokol ND. Úlohou modulu IRI-IIF je sledovat tento protokol a uchovávat si tak stav pro každou jednotlivou IPv6 adresu (na místo stavu celé klientské stanice). Každá IPv6 adresa prochází následujícími stavy (viz diagram na obrázku 6.5):

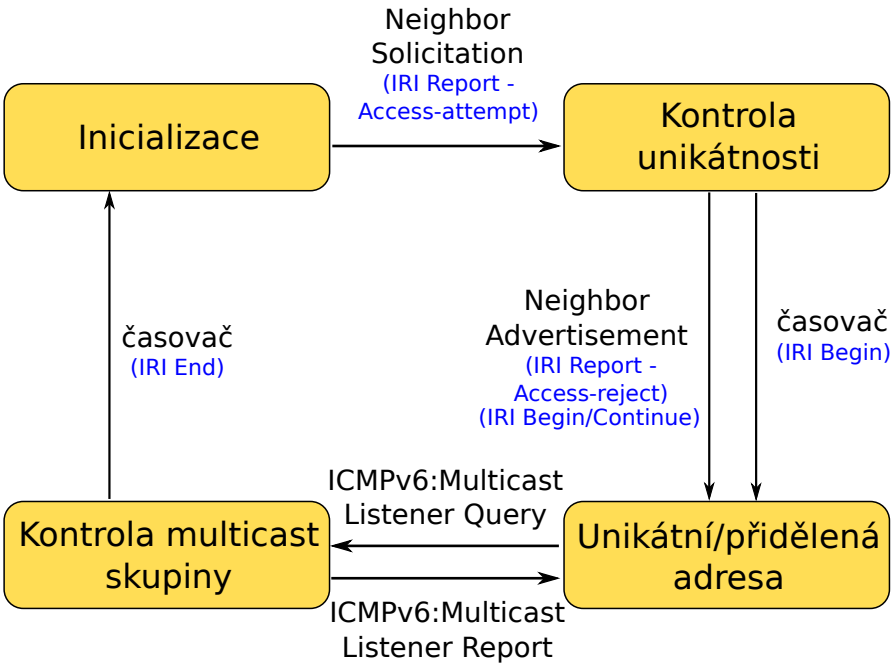
- *Inicializace* - adresa není přidělena žádnému klientovi (výchozí stav)
- *Kontrola unikátnosti* - stanice si vytvořila IPv6 adresu a zjišťuje její unikátnost
- *Unikátní/přidělená adresa* - tento stav reprezentuje dvě situace: 1) na kontrolu duplicity adresy nikdo neodpověděl a tak jí může klient začít používat a 2) na kontrolu duplicity se ozvala jiná klientská stanice, která danou adresu již používá.
- *Kontrola multicast skupiny* - směrovač se dotazuje, zda se ještě v dané skupině nachází některý klient.

Všechny zprávy protokolu SLAAC jsou zasílány na skupinové adresy. Aby byl modul schopen tyto zprávy přijímat a analyzovat je nezbytné, aby byl v daných (multicastových) skupinách také přihlášen. Nejprve musí být modul přihlášen ve skupině ff::16, do které jednotlivý klienti zasílají zprávy týkající se přihlašování do skupin. Následně, když modul detekuje, že se určitý klient přihlašuje do některé *solicited node multicastové* skupiny, okamžitě se tam přihlásí také. Pouze tímto způsobem je modul schopen přijímat a analyzovat zprávy, které jsou uvedeny v diagramu.

Důležitá poznámka: Pokud je striktně vyžadováno, aby se modul choval pouze pasivně tj. nevkládat žádné pakety uvnitř sledované sítě (včetně zpráv pro přihlášení do multicastových skupin), potom je nezbytné zajistit, aby modul získal přístup k uvedené komunikaci jiným způsobem (např. vhodným zapojením v rámci infrastruktury poskytovatele nebo konfigurací aktivních prvků).

Činnost bloku IRI-IIF se řídí podle následujících pravidel:

- Počátečním stavem je *Inicializace*. V tomto stavu není IPv6 adresa přiřazena žádnému klientovi, přesněji řečeno, modul IRI-IIF o takovém přiřazení doposud neví.
- Přijetím zprávy *Neighbor Solicitation* modul detekuje situaci, kdy si klient vygeneroval vlastní IPv6 adresu a snaží o její ověření. V souvislosti s touto událostí se provede přechod do stavu *Kontrola unikátnosti* a modul vygeneruje zprávu *IRI Report (access-attempt)*.
- V rámci kontroly unikátnosti adresy mohou nastat dvě situace:



Obrázek 6.5: Stavový diagram funkce IRI-IIF protokolu SLAAC

1. Jako odpověď na *Neighbor Solicitation* přijde zpráva *Neighbor Advertisement*, což znamená, že danou adresu již používá nějaká jiná stanice. Vůči klientovi, který si tuto duplicitní adresu vytvořil se jedná o neúspěšný pokus o přidělení adresy a modul proto zašle zprávu *IRI Report (access-reject)*. Naopak z pohledu stanice, která již adresu používá se jedná o potvrzení její platnosti a modul vygeneruje zprávu *IRI Continue* nebo *IRI Begin* v závislosti na tom, zda již má modul ve své tabulce o stanici záznam či nikoliv.
2. Do 2 sekund nepřijde žádná odpověď na kontrolu duplicity, předělení adresy se tímto potvrdí, počítač si nastaví své rozhraní a modul odešle zprávu *IRI Begin*.

V obou případech provede modul přechod do stavu *Unikátní/přidělená adresa*. Prosím všimněte si, že diagram znázorňuje stavy pro jednotlivé adresy a nikoliv klientské stanice. Zatímco pro jednu stanici se může jednat o neúspěšný pokus, pro jinou stanici reprezentuje stejná událost úspěšné přidělení nebo prodloužení adresy.

- Před použitím vygenerované adresy se klient musel přihlásit do multicastové skupiny odvozené z IPv6 adresy (*solicited-node* adresa). Směrovač se periodicky dotazuje skrze zprávu *ICMPv6 Multicast Listener Query*, zda-li je v dané skupině přihlášena nějaká stanice. Přijetím uvedené zprávy provede modul přechod do stavu *Kontrola multicast skupiny*.
- Pokud do časového intervalu uvedeného v předcházející ICMPv6 zprávě nepřijde od klienta odpověď, znamená to, že se již nenachází v dané skupině a IPv6 adresu přestal používat. Dokonce, pokud nepřijde odpověď od žádného klienta, jsou všechny IPv6 adresy mapované na danou skupinu považovány za již neplatné. Modul vygeneruje pro tyto neplatné adresy zprávu *IRI End* a přechází do stavu *Inicializace*.

- Naopak, pokud v rámci kontroly multicastové skupiny odpoví alespoň jeden klient, potvrdí tím i platnost pro všechny IPv6 adresy náležící do daného rozsahu a modul aplikuje přechod zpět do stavu *Unikátní/přidělená adresa*.

Při analýze chování protokolu SLAAC na různých operačních systémech bylo bohužel zjištěno, že některé z nich nedodržují předepsané pravidla a např. zcela ignorují jak zprávu *Neighbor Solicitation* (klient nereaguje na požadavek o ověření adresy), tak i zprávu *Neighbor Advertisement* (klient nereaguje na informaci o již používané adrese). Pro modul IRI-IIF se jedná o nepříjemný problém, neboť nelze spolehlivě párovat požadavek a odpověď na ověření adresy. Podrobnější informace o výsledcích testování protokolu SLAAC na různých operačních systémech a postupy, jak reagovat na nestandardní chování některých z nich byly publikovány v článku *A New Approach for Detection of Host Identity in IPv6 Networks* [58] a v rozšířené verzi tohoto článku nazvané *Host Identity Detection in IPv6 Networks* [59] a tento dokument se tím nebude zabývat.

Tabulka 6.5 zachycuje zprávy IRI generované modulem pro ND/SLAAC.

6.6 Extensible Messaging and Presence Protocol (XMPP)

6.6.1 Popis protokolu

Extensible Messaging and Presence Protocol (XMPP) je protokol pro komunikaci v reálném čase využívající transportní protokol TCP. Je to textový protokol využívající jazyka XML. Zprávy jsou zasílány přes datový proud v podobě XML elementů.

Jádro protokolu a podpora pro zasílání zpráv (Instant Messaging) jsou specifikovány v podobě RFC dokumentů (RFC 6120 [69], RFC 6121 [70], RFC 6122 [70]). Protokol XMPP je dále rozšiřitelný a jednotlivá rozšíření jsou vydávána v podobě XMPP Extension Protocol (XEP) dokumentů.

Uživatel je v síti identifikován pomocí Jabber ID (JID), který je ve formátu:

```
login_uzivatele@domena/zdroj
```

Část identifikátoru *zdroj* určuje odkud je uživatel přihlášen, aby bylo možno rozlišit současné přihlášení k jednomu účtu z více míst nebo zařízení. Komunikace s XMPP serverem probíhá nejčastěji pomocí TCP spojení na portu 5222.

Typy zasílaných zpráv

Zprávy jsou zasílaný v podobě XML elementů, které jsou zasílané přes datový proud a rozlišují se tři základní typy.

- **Message** - Slouží k zasílání klasických zpráv mezi jednotlivými entitami účastnící se komunikace.
- **Presence** - Využívá se k zasílání informací o dostupnosti klienta. Typicky se zasílá pomocí broadcastu mezi všemi uživateli v seznamu kontaktů, ale lze definovat určeného příjemce.
- **Info/Query (IQ)** - Mechanismus požadavku a odpovědi při komunikaci klienta se serverem.

Příklad zprávy:

Událost	Zpráva IRI	Popis události	Identifkátory
Zachycení NA odeslané z jiné MAC adresy než byla dříve známa pro danou IPv6 adresu	IRI END	IP adresa již není připojena v siti	Původní MAC adresa, zdrojová IPv6 adresa NA
Zachycení NA pro novou adresu	IRI BEGIN	Uživatel si vygeneroval novou IP adresu	MAC adresa odesilatele, zdrojová IPv6 adresa NA
Zachycení NA (dvojice MAC:IPv6) pro kterou už existuje záznam	IRI CONTINUE	Uživatel potvrzuje, že stále používá IP adresu	MAC adresa odesilatele, zdrojová IPv6 adresa NA
Cca 1 sekundu po zachycení NS z jiné zdrojové MAC , než bylo dříve zjištěno pro zdrojovou IPv6 adresu v původní NS	IRI END	IP adresa již není připojena v síti	Původní MAC adresa a IPv6 adresa
Cca 1 sekundu po zachycení NS se zdrojovou IPv6 adresou pro položku (MAC:IPv6), která ještě nebyla uložena (DAD)	IRI BEGIN	Uživatel si vygeneroval novou IP adresu	Zdrojová MAC adresa, dotazovaná IPv6 adresa (v rámci DAD)
Cca 1 sekundu po zachycení NS se zdrojovou IPv6 adresou :: pro položku (MAC:IPv6) která již je uložena (DAD)	IRI CONTINUE	Uživatel potvrzuje, že stále používá IP adresu	Zdrojová MAC adresa, dotazovaná IPv6 adresa (v rámci DAD)
Na všechny adresy patřící, ke kterým patří *solicited node* adresa, do jejíž skupiny klient odeslal zprávu *Multicast Listener Done* (132)	IRI END	Uživatel již není připojen v síti	Původní MAC adresa a všechny IPv6 adresy, ke kterým patří daná solicited multicastová skupina
Na všechny adresy patřící, ke kterým patří *solicited node* adresa, ve které nepřišla odpověď na zprávu *Multicast Listener Query* (130) od směrovače	IRI END	Uživatel již není připojen v síti	Všechny MAC adresy a všechny IPv6 adresy, ke kterým patří daná solicited multicastová skupina

Tabulka 6.5: Zprávy IRI generované modulem ND/SLAAC.

```
<message xmlns="jabber:client" type="chat"
         to="standa4@core.im" id="aabca">
   <body>Pozdrav</body>
   <active xmlns="http://jabber.org/protocol/chatstates"/>
</message>
```

Významné zprávy protokolu XMPP

Pro tvorbu zpráv IRI jsou významné zprávy protokolu XMPP, které slouží k přihlášení k síti a autentizaci a také zprávy měnící status uživatele a zprávy přenášející vlastní obsah komunikace.

- **Přihlášení a autentizace** - Na počátku je vytvořen datový proud zasláním počátečních elementů `<stream:stream>` mezi klientem a serverem a naopak. Server zašle seznam podporovaných mechanizmů autentizace pomocí elementu `<stream:features>`. Klient zahájí autentizaci elementem `<auth>`. Jako odpověď zašle server element `<challenge>`, na který klient zašle odpověď v podobě elementu `<response>`. Podle úspěšnosti autentizace vrátí server odpověď. V případě úspěchu je zaslán element `<success>`, jinak je zaslán element `<failure>` a datový proud je uzavřen.

  Serverem je zahájeno připojení zdroje klienta. Klient pomocí zprávy IQ typu set zašle element `<bind>` obsahující pod-element `<resource>` s informací o připojovaném zdroji. V případě schválení zašle server zprávu IQ typu result s elementem `<bind>` a pod-elementem `<jid>` obsahujícím celé uživatelské ID a klient je tímto připojen.

- **Úpravy statusu** - Výměna informací o statusu klienta se provádí elementem `<presence />`. V základní verzi se jedná o dvě hodnoty dostupný/nedostupný. Dostupný kontakt zasílá element `<presence />` bez atributu type a nedostupný zasílá atribut type s hodnotou `unavailable`.

  S informací o statusu mohou být zasílány další informace, které jsou rozděleny do tří typů elementů. Element `<show />` smí být pouze jeden a slouží k detailnějšímu určení stavu. Text s popisem stavu pro koncového uživatele je obsažen v elementu `<status />`. Může jich být více v rozdílných jazycích, kdy je jazyk specifikován příslušným atributem. Element `<priority />` určuje prioritu v případě, že má klient připojených více zdrojů. Priorita je určena hodnotou -128 až +127 a v případě, že není element zaslán, je priorita brána jako hodnota nula. Větší číslo znamená vyšší prioritu.

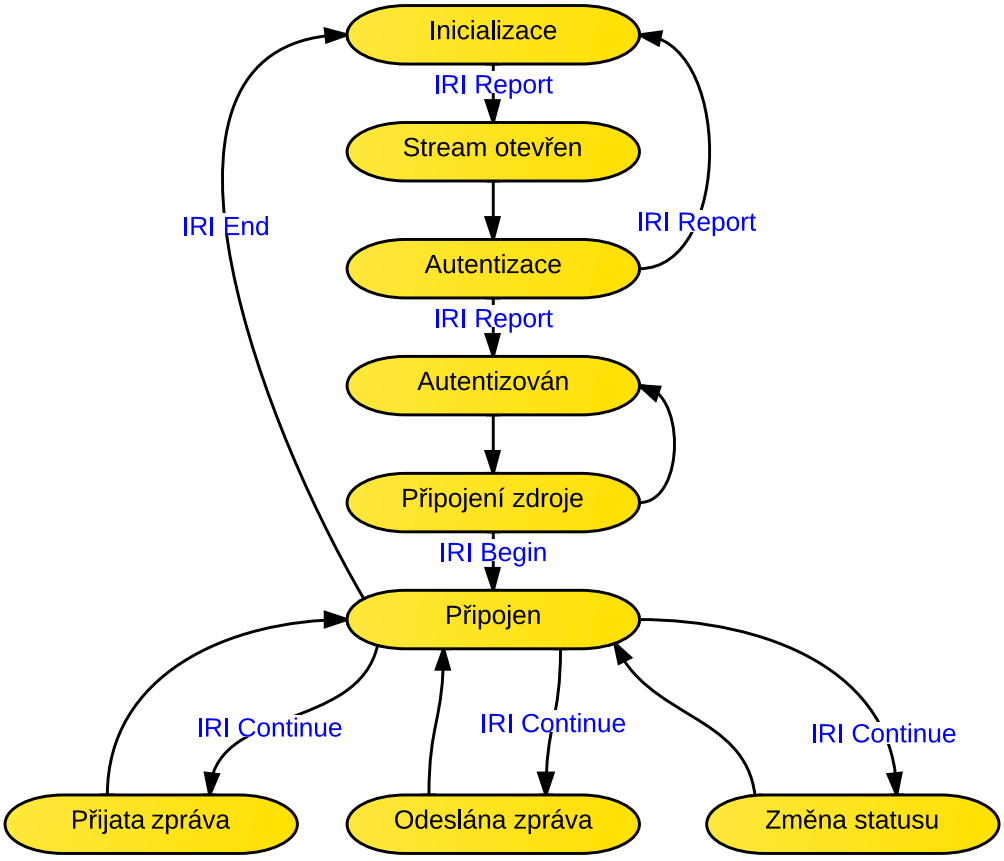
- **Zprávy s obsahem komunikace** - Zprávy mezi uživateli XMPP sítě jsou zasílány pomocí elementů `<message />` Úvodní zpráva by měla obsahovat element `<thread />` určující, k jakému vláknu komunikace zpráva patří. Podle určeného vlákna jsou zprávy rozdělovány do skupin souvisejících komunikací. Tímto je zaručena větší přehlednost pro koncového uživatele.

  Zprávy jsou podle jejich významu rozlišovány na *chat*, *groupchat*, *headline*, *normal* a *error*. Na zprávu typu *normal* je možno zaslat odpověď, ale zprávy jsou zasílány osamoceně a nejsou ukládány do historie konverzace. Zpráva typu *headline* slouží k zasílání varování, upozornění nebo dalších informací, na které není očekávána odpověď. Typy *chat* a *groupchat* jsou využívány pro komunikaci, kdy jsou zprávy zasílány v rámci určitého kontextu a je ukládána historie proběhnuté komunikace. Groupchat se využívá pro skupinové komunikace, kdy spolu komunikují více než dva uživatelé.

  Samotný text zprávy je obsažen v elementu `<body />`, kterých může existovat více v rozdílných jazycích nebo nemusí být vůbec zadán. Předmět určující obsah zprávy je pro uživatele specifikováno v elementu `<subject />`, který také může být zaslán v rozdílných jazycích.

6.6.2 Činnost IRI-IIF

IM modul pro protokol XMPP zpracovává celé TCP spojení a analyzuje XML elementy tvořící datový proud na aplikační úrovni. Po úspěšném otevření datového proudu je vytvořena zpráva *IRI Report* informující o připojení k serveru. Po dokončení autentizace je vytvořena zpráva *IRI Report* informující o jejím úspěchu či neúspěchu. Při neúspěchu je datový proud uzavřen a pro další pokus je potřeba nového spojení. Jakmile je připojen zdroj uživatele, je vytvořena zpráva *IRI Begin* informující o začátku komunikace. Pro běžnou komunikaci pomocí zpráv a pro zprávy upravující status klienta jsou vytvářeny zprávy *IRI Continue* informující o nastalé události. Ukončující zpráva *IRI End* je vytvářena poté, co se uživatel odpojí nebo poté, co je uzavřeno TCP spojení.



Obrázek 6.6: Stavový diagram funkce IRI-IIF protokolu XMPP

Tabulka 6.6 zachycuje zprávy IRI generované modulem pro XMPP, tabulka 6.7 pak doplňuje identifikátory detekované pro tyto zprávy.

Událost	Zpráva IRI
Dokončení autentizace	IRI Report
Úspěšné připojení zdroje	IRI Begin
Byly zjištěny všechny potřebné údaje z probíhající komunikace a IRI Begin zatím nebyla vytvořena	IRI Begin
Odeslaná nebo přijatá zpráva	IRI Continue
Změna statusu	IRI Continue
Odpojení od IM sítě	IRI End
Reakce na sledované události v případě, že IRI Begin nebyla vytvořena	IRI Report
Začátek šifrované komunikace	IRI Report

Tabulka 6.6: Zprávy IRI generované modulem XMPP.

Událost	Identifikátory
Přijmutí nebo odeslání zprávy	Identifikátory XMPP_LOGIN obou komunikujících stran obalené identifikátorem XMPP
Ostatní zprávy	Identifikátor XMPP_LOGIN sledovaného klienta
Zprávy kdy není znám identifikátor sledovaného klienta	Identifikátory XMPP_LOGIN ani XMPP nejsou uváděny

Tabulka 6.7: Identifikátory detekované modulem XMPP.

6.7 Internet Relay Chat (IRC)

6.7.1 Popis protokolu

Internet Relay Chat (IRC) je protokol pro komunikaci v reálném čase využívající transportní protokol TCP. Je to textový protokol a na rozdíl od protokolů XMPP a OSCAR neumožňuje zasílání zpráv obsahujících hlas či video. Tento protokol typicky slouží pro komunikaci mezi větším množstvím uživatelů.

Bylo publikováno mnoho specifikací popisujících fungování protokolu IRC (RFC 2810 [41], RFC 2811 [42], RFC 2812 [43], RFC 2813 [44]), ale doposud neexistuje žádná oficiální specifikace.

Pro identifikaci uživatelů připojených k síti IRC se využívá přezdívky, která je tvořena řetězcem znaků. Komunikace pomocí protokolu IRC je ve většině případů anonymní a tak si uživatel může zvolit libovolnou aktuálně volnou přezdívku a tu si ponechá, dokud si ji nezmění nebo neukončí komunikaci.

IRC komunikace probíhá pomocí TCP spojení a přestože IRC protokolu byl přidělen port 194, téměř vždy byly a jsou pro komunikaci využívány porty 6665-6669 (nejčastěji 6667).

Typy zasílaných zpráv

Zprávy jsou předávány v podobě textových příkazů. Server zasílá odpovědi jako reakci na některé zaslané příkazy. Každá zpráva se skládá ze tří částí: prefixu, příkazu a jeho parametrů. Prefix je volitelný, určuje původní zdroj zprávy a jeho přítomnost je oznámena počátečním znakem ':', na který prefix přímo navazuje. Příkaz určuje význam zprávy a pro bližší upřesnění za ním následují parametry. Parametrů může být u každého příkazu maximálně 15. Jednotlivé části jsou odděleny

jednou mezerou. Odpověď je stejného formátu jako normální zpráva, ale klíčové slovo určující příkaz je tvořeno třemi číslicemi.

Příklad příkazů a odpovědi:

```
NICK standa
USER standa standa irc.europnet.org :Stanislav Barta
:free.chat-fr.europnet.org 001 standa
:Welcome to the EuropNet IRC Network
 standa!standa@dhcps240.fit.vutbr.cz
```

Významné zprávy protokolu IRC

Pro tvorbu zpráv IRI jsou významné příkazy protokolu IRC, které slouží pro připojení k serveru, pro připojení a odpojení od kanálů, nastavení a zrušení statusu *AWAY* a příkazy pro zasílání zpráv s obsahem komunikace mezi uživateli. Stejně důležité jsou také odpovědi zaslané jako reakce na tyto zprávy.

- **Připojení k síti** - Připojení k síti IRC může probíhat dvěma způsoby. Ve většině případů se uživatel může k serveru připojit bez znalosti hesla a vybere si libovolnou volnou přezdívku. Druhou možností je mít na serveru registrovaný účet a pro připojení je poté vyžadováno heslo. Při přihlašování pomocí registrovaného účtu musí být příkaz s heslem *PASS* zaslán jako první a teprve poté může být odeslán příkaz *NICK* obsahující přezdívku a příkaz *USER* s uživatelským a reálným jménem. Po úspěšném připojení zasílá server odpověď *RPL_ WELCOME*, která má číselnou hodnotu 001. Přihlášení k serveru s libovolnou přezdívkou bez nutnosti znát heslo probíhá stejným způsobem, pouze není odesláno heslo a rovnou se zasílá příkaz *NICK*.

- **Připojení nebo odpojení od kanálu** - Pro připojení ke kanálu se použije příkaz *JOIN* obsahující název kanálu, ke kterému se připojuje. V případě úspěšného připojení je uživateli zaslán příkaz *JOIN* jako potvrzení akce a následně téma kanálu v odpovědi *RPL_ TOPIC*, která má číselnou hodnotu 332. Odpojení od kanálu se provádí příkazem *PART*.

- **Nastavení nebo zrušení statusu AWAY** - Nastavení i zrušení statusu se provádí stejným příkazem a tím je příkaz *AWAY*. Lze ho zadat s parametrem určujícím zasílaný text nebo bez parametrů. Pokud je parametr zadán, status je aktivován. Jako potvrzení tohoto stavu musí server zaslat odpověď *RPL_ NOWAWAY*, která má číselnou hodnotu 306. Status se ruší zasláním příkazu *AWAY* bez zadaných parametrů. Server jako odpověď zašle *RPL_ UNAWAY* s číselnou hodnotou 305.

- **Zprávy s obsahem komunikace** - K zasílání zpráv mezi uživateli je možno použít tří různých příkazů a jsou jimi *PRIVMSG*, *NOTICE* a *SQUERY*.

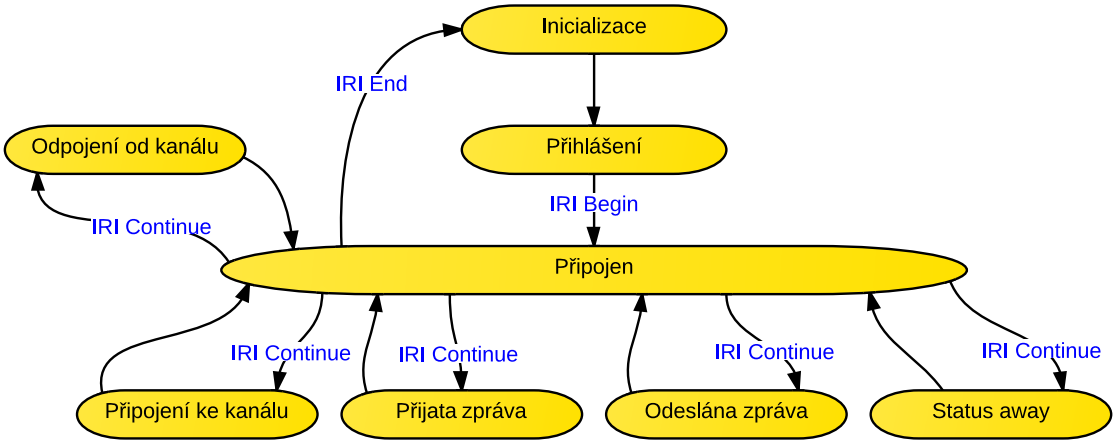
  Příkaz *PRIVMSG* slouží k zasílání zpráv mezi uživateli i k zasílání zpráv kanálům. Příkaz *NOTICE* se používá obdobně jako *PRIVMSG*. Rozdíl je v tom, že na příkaz *NOTICE* nesmí být zasílány automatické odpovědi a to platí i pro servery, které nesmí zasílat informace o případné chybě. Tento příkaz je použitelný jak pro uživatele, tak pro služby, ale většinou je využíván službami. Poslední možností zasílání zprávy je příkaz *SQUERY*. Tento příkaz se používá podobně jako *PRIVMSG*, ale jeho příjemcem musí být služba a je jediným příkazem použitelným pro zasílání textových zpráv službám.

Událost	Zpráva IRI
Úspěšné připojení k IM síti	IRI Begin
Byly zjištěny všechny potřebné údaje z probíhající komunikace a IRI Begin zatím nebyla vytvořena	IRI Begin
Odeslaná nebo přijatá zpráva	IRI Continue
Změna statusu AWAY	IRI Continue
Připojení ke kanálu nebo odpojení od kanálu	IRI Continue
Odpojení od IM sítě	IRI End
Reakce na sledované události v případě, že IRI Begin nebyla vytvořena	IRI Report

Tabulka 6.8: Zprávy IRI generované modulem IRC.

6.7.2 Činnost IRI-IIF

IM modul pro protokol IRC zpracovává celé TCP spojení a analyzuje jednotlivé řádky tvořící příkazy protokolu na aplikační úrovni. Po úspěšném připojení k IRC serveru je vytvořena zpráva *IRI Begin* informující o začátku komunikace. Dále jsou tvořeny zprávy *IRI Continue* podle zachycených příkazů pro připojení nebo odpojení od kanálu. Zprávy *IRI Continue* jsou také vytvářeny po přijmutí nebo odeslání zpráv s obsahem komunikace a při nastavení nebo zrušení statusu AWAY. Po odpojení klienta od sítě nebo poté, co je uzavřeno TCP spojení je vytvořena zpráva *IRI End*.



Obrázek 6.7: Stavový diagram funkce IRI-IIF protokolu IRC

Tabulka 6.8 zachycuje zprávy IRI generované modulem pro IRC, tabulka 6.9 pak doplňuje identifikátory detekované pro tyto zprávy.

Událost	Identifikátory
Přijmutí nebo odeslání zprávy	Identifikátory IRC_LOGIN obou komunikujících stran obalené identifikátorem IRC
Ostatní zprávy	Identifikátor IRC_LOGIN sledovaného klienta
Zprávy kdy není znám identifikátor sledovaného klienta	Identifikátory IRC_LOGIN ani IRC nejsou uváděny
Připojení ke kanálu nebo odpojení od kanálu	Identifikátor CHANNEL

Tabulka 6.9: Identifikátory detekované modulem IRC.

6.8 Open System for Communication in Realtime (OSCAR)

6.8.1 Popis protokolu

Open System for Communication in Realtime (OSCAR) je protokol pro komunikaci v reálném čase využívající transportní protokol TCP. Je to binární protokol využívající pro zapouzdření přenášených dat dvou aplikačních protokolů *Frame Layer Protocol* (FLAP) a *Simple Network Atomic Communication* (SNAC), kdy protokol SNAC je přenášen v datové části protokolu FLAP.

Protokol OSCAR je proprietárním protokolem a je využíván v systémech ICQ a AIM. Přestože je protokol proprietární, díky reverznímu inženýrství a také díky tomu, že v roce 2008 společnost AOL uveřejnila části dokumentace, je dnes identifikována struktura většiny zasílaných zpráv.

Pro identifikaci uživatelů připojených k síti OSCAR se využívá identifikační číslo UIN, které je tvořeno pouze číslicemi. Původně bylo UIN tvořeno pouze pěti číslicemi, ale s přibývajícím počtem uživatelů se rozsah hodnot postupně zvětšoval.

Komunikace probíhá pomocí TCP spojení na některém z portů 5190, 5191, 5192, 5193.

Typy zasílaných zpráv

Zprávy jsou zasílány v binární podobě s využitím dvou aplikačních protokolů.

FLAP

Zapouzdřuje všechny zprávy zasílané protokolem OSCAR. Pro rozdělení zpráv do tříd podle typu obsahu slouží tzv. kanály. Aktuálně je využíváno pět kanálů.

- 0x01 - slouží pro navázání nového spojení
- 0x02 - pouze přes tento kanál jsou přenášeny zprávy obsahující vlastní data
- 0x03 - slouží pro přenášení zpráv o chybách
- 0x04 - slouží k úspěšnému dokončení navazování spojení
- 0x05 - slouží k udržování otevřeného spojení

SNAC

Protokol zapouzdřující přenášená data. Zprávy jsou děleny do skupin, tzv. rodin, podle toho jaké služby poskytují.

- 01 - základní správa služeb
- 02 - správa online dat uživatele (např. profil)
- 03 - správa seznamu kontaktů a dostupnosti uživatele (nastavení statusu)
- 04 - zasílání zpráv s vlastním obsahem komunikace mezi uživateli
- 06 - zaslání emailu s pozvánkou služby AIM
- 07 - správa účtu služby AIM
- 08 - zobrazení vyskakovacího okna na straně klienta (využíváno službou AIM)
- 09 - správa seznamů upravujících soukromí uživatele
- 0a - nalezení uživatele služby AIM podle emailu
- 0b - sběr statistik
- 0d - správa skupinové komunikace služby AIM
- 0e - skupinová komunikace služby AIM
- 0f - vyhledání uživatelů služby AIM
- 10 - správa uživatelských ikon uložených na serveru
- 13 - správa seznamů kontaktů uložených na serveru
- 15 - zajištění kompatibility se starším Mirabilis ICQ databázovým serverem
- 17 - autentizace nebo registrace nového uživatele

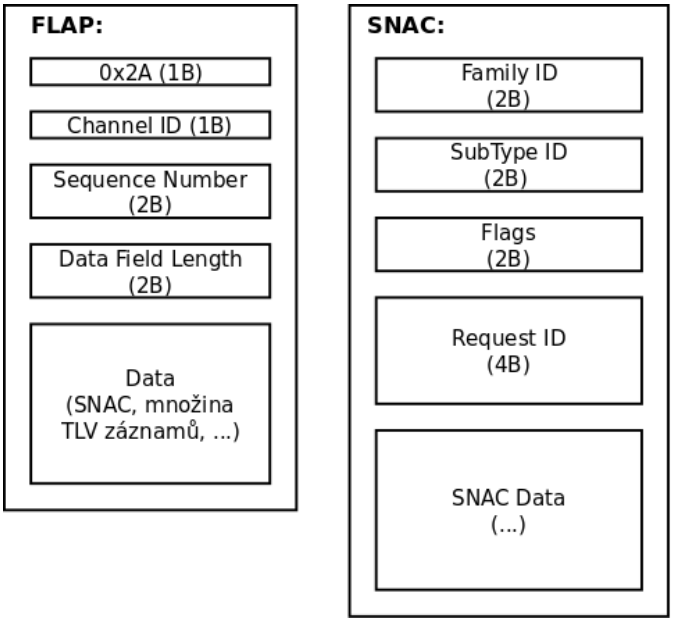
Data jsou organizována pomocí struktury TLV (Type, Length, Value), která určuje typ dat, jejich délku a dále vlastní data. Struktury TLV by měly být zapouzdřeny v protokolu SNAC, ale často jsou uloženy jako data přímo v protokolu FLAP.

Významné zprávy protokolu IRC

Pro tvorbu zpráv IRI jsou významné zprávy protokolu OSCAR, které souvisí s autentizací a přihlášením k síti, zprávy sloužící k základní správě služeb a také k nastavení statusu uživatele. Dále jsou důležité zprávy, pomocí kterých jsou zasílány zprávy s vlastním obsahem komunikace mezi uživateli.

- **Přihlášení a autentizace** - Autentizace může probíhat dvěma způsoby. Při prvním způsobu zašle klient přes *kanál 0x01* na server požadavek, který obsahuje informace o uživateli. Server zašle odpověď přes *kanál 0x04*. Odpověď obsahuje *TLV* položky a podle jejich typu lze určit úspěšnost autentizace. Pokud proběhla úspěšně, odpověď obsahuje mimo jiné *TLV typu 0x0006* obsahující autentizační cookie a *TLV typu 0x0005* obsahující IP adresu *BOS* serveru. Naopak v případě neúspěchu obsahuje mimo jiné *TLV typu 0x0008* obsahující chybový kód.

  Při využití druhé možnosti klient zašle požadavek na získání šifrovacího klíče pomocí zprávy *SNAC(17,06)* a server mu ho zašle pomocí zprávy *SNAC(17,07)*. Požadavek na autentizaci je zaslán ve zprávě *SNAC(17,02)* klientem. Server odpoví zprávou *SNAC(17,03)*, z které lze



Obrázek 6.8: Struktura rámce protokolů FLAP a SNAC

určit úspěšnost autentizace a to podle *TVL* struktur, které obsahuje. Platí stejná pravidla jako v případě autentizace přes kanál 0x01, tedy pokud odpověď obsahuje *TLV typu 0x0006*, byla autentizace úspěšná. Pokud osahuje *TLV typu 0x0008*, byla autentizace neúspěšná.

Po úspěšném získání cookie a IP adresy BOS serveru, klient na tento server zašle pomocí *kanálu 0x01* zprávu obsahující *TLV* strukturu jejímž obsahem je dříve získaná cookie. Jako odpověď od serveru je přijata zpráva *SNAC(01,03)*, která obsahuje seznam podporovaných služeb. Klient zprávou *SNAC(01,17)* požádá o zjištění verzí služeb a server mu poskytne odpověď ve zprávě *SNAC(01,18)*. Pomocí zpráv *SNAC(01,06)* a *SNAC(01,07)* získá klient informaci a rychlostních limitech a ve zprávě *SNAC(01,08)* je potvrdí a tím je připojení připraveno.

Dále jsou zjištěny a potvrzeny nebo přenastaveny limity jednotlivých služeb. To je provedeno skrze zprávy *SNAC(XX,02)*, *SNAC(XX,03)* a *SNAC(XX,04)*, kde *XX* nahrazuje identifikátor skupiny nastavované služby. Skrze zprávu *SNAC(XX,02)* klient požádá server o poskytnutí informací o službě, ve zprávě *SNAC(XX,03)* mu server zašle odpověď a zprávou *SNAC(XX,04)* klient zašle svou odpověď. V posledním kroku jsou odeslány klientem dvě zprávy, které dokončí přihlášení k síti. Zpráva *SNAC(01,1E)* obsahuje informace o DC a statusu klienta. Zpráva *SNAC(01,02)* informuje o připravenosti klienta.

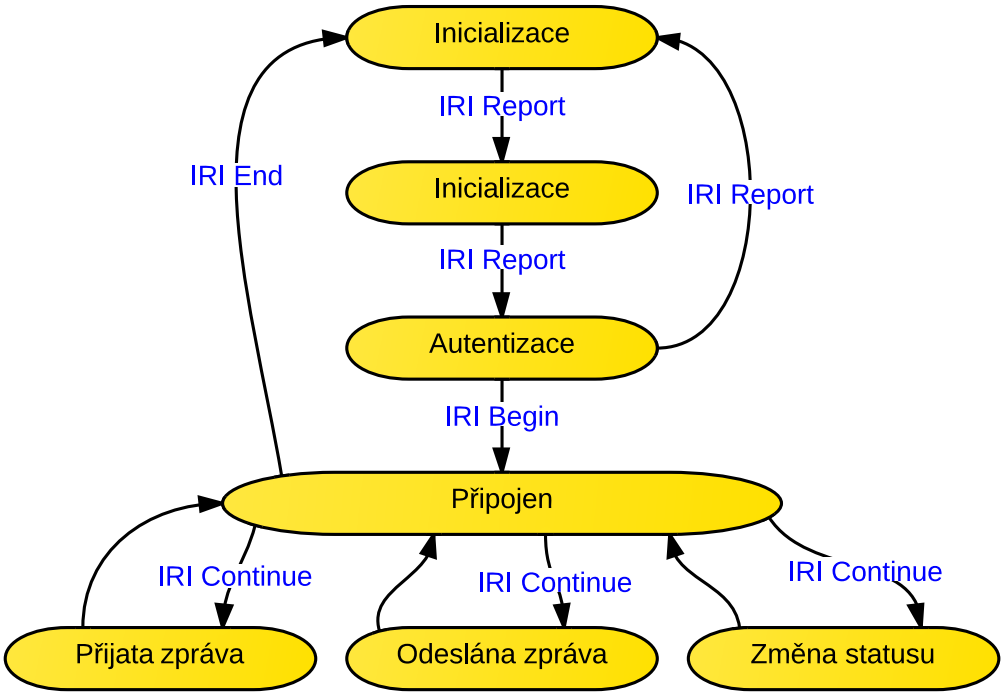
- **Úpravy statusu** - Svůj status zasílá klient ve zprávě *SNAC(01,1E)*. Informace o změně statusu svých kontaktů získává klient od serveru ve zprávách *SNAC(03,0B)* a *SNAC(03,0C)*. Zpráva *SNAC(03,0B)* je zasílána pokud kontakt přejde do stavu online nebo změní svůj status a zpráva *SNAC(03,0C)* je zaslána, když kontakt přejde do stavu offline.
- **Zprávy s obsahem komunikace** - Odchozí zpráva je zasílána v podobě zprávy *SNAC(04,06)* a příchozí zpráva v podobě *SNAC(04,07)*. Pomocí zprávy *SNAC(04,14)* je

přenášena informace o tom, že uživatel zrovna píše. V případě, že je uživateli zaslána zpráva, ale nemůže být doručena, uživatel je o této události informován zprávou *SNAC(04,0A)*.

Pokud je některý uživatel obtěžován zprávami jiného uživatele, může tohoto uživatele nahlásit pomocí zprávy *SNAC(04,08)*. Tento požadavek musí server potvrdit zprávou *SNAC(04,09)*.

6.8.2 Činnost IRI-IIF

IM modul pro protokol OSCAR zpracovává celé TCP spojení a analyzuje binární protokoly FLAP a SNAC tvořící datový proud na aplikační úrovni. Po otevření spojení mezi klientem a serverem je vytvořena zpráva *IRI Report* informující o připojení k serveru. Dále je vytvořena zpráva *IRI Report* informující o úspěšnosti či neúspěšnosti autentizace. Jakmile dá klient vědět, že je připraven ke komunikaci, je vytvořena zpráva *IRI Begin* informující o začátku komunikace. Pro běžnou komunikaci pomocí zpráv a pro zprávy upravující status klienta jsou vytvářeny zprávy *IRI Continue* informující o nastalé události. Ukončující zpráva *IRI End* je vytvářena poté co se uživatel odpojí nebo poté, co je uzavřeno TCP spojení.



Obrázek 6.9: Stavový diagram funkce IRI-IIF protokolu OSCAR

Tabulka 6.10 zachycuje zprávy IRI generované modulem pro OSCAR, tabulka 6.11 pak doplňuje identifikátory detekované pro tyto zprávy.

Událost	Zpráva IRI
Dokončení autentizace	IRI Report
Úspěšné připojení k IM síti	IRI Begin
Byly zjištěny všechny potřebné údaje z probíhající komunikace a IRI Begin zatím nebyla vytvořena	IRI Begin
Odeslaná nebo přijatá zpráva	IRI Continue
Změna statusu	IRI Continue
Odpojení od IM sítě	IRI End
Reakce na sledované události v případě, že IRI Begin nebyla vytvořena	IRI Report

Tabulka 6.10: Zprávy IRI generované modulem OSCAR.

Událost	Identifikátory
Přijmutí nebo odeslání zprávy	Identifikátory OSCAR_LOGIN obou komunikujících stran obalené identifikátorem OSCAR
Ostatní zprávy	Identifikátor OSCAR_LOGIN sledovaného klienta
Zprávy kdy není znám identifikátor sledovaného klienta	Identifikátory OSCAR_LOGIN ani OSCAR nejsou uváděny

Tabulka 6.11: Identifikátory detekované modulem OSCAR.

6.9 Yahoo! Messenger Protocol (YMSG)

6.9.1 Popis protokolu

Yahoo"! Messenger Protocol (YMSG) je binární protokol pro komunikaci v reálném čase využívající transportní protokol TCP. Přenášená data jsou uložená v textové podobě. Protokol YMSG je proprietárním protokolem firmy Yahoo! a je využíván v jejím komunikátoru Yahoo! Messenger. Pro identifikaci uživatelů připojených k síti YMSG se využívá uživatelské jméno, které je tvořeno řetězcem ASCII znaků. Komunikace probíhá pomocí TCP spojení na portu 5050.

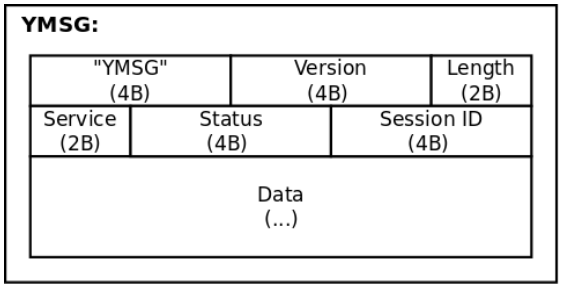
Typy zasílaných zpráv

Zasílané zprávy jsou tvořeny binární hlavičkou a přenášenými daty v plain-textové podobě. Z obsahu hlavičky lze určit typ aktuálně využívané služby protokolu YMSG. Vlastní data jsou tvořena dvojicemi hodnot. První položka z dvojice určuje její typ a druhou položkou je určená hodnota. Jednotlivé položky jsou odděleny dvojicí hexadecimálních hodnot C0 a 80.

Významné zprávy protokolu IRC

Pro tvorbu zpráv IRI jsou významné zprávy protokolu YMSG, které souvisí s autentizací a přihlášením k síti a zprávy sloužící k nastavení statusu uživatele. Dále jsou důležité zprávy, pomocí kterých jsou zasílány zprávy s vlastním obsahem komunikace mezi uživateli.

- **Přihlášení a autentizace** - Přihlášení k YMSG síti sestává z kroků autentizace a získání seznamu kontaktů od serveru. Autentizace je provedena pomocí zpráv služeb *Authentication*



Obrázek 6.10: Formát rámce protokolu YMSG

určené hexadecimálním číslem 57 a *Authentication Response* určené hexadecimálním číslem 54. Pomocí zpráv služby *Authentication* klient zašle svoje uživatelské jméno a server mu zašle odpověď obsahující řetězec pro zašifrování přihlašovacích údajů. Přihlašovací údaje zašle klient ve zprávě služby *Authentication Response*. Po úspěšné autentizaci získá klient svoje osobní údaje ve zprávě služby *List* určené hexadecimálním číslem 55. Seznam kontaktů je klientovi zaslán ve zprávě služby *List v15* určené hexadecimálním číslem f1.

- **Úpravy statusu** - Úpravy statusu jsou prováděny pomocí zpráv služby *Status Update*, která je určena hexadecimální hodnotou C6. Nastavovaný stav je určen podle položky v datové části určené číslem 10. Informace o změně statusu klienta ze seznamu kontaktů je prováděna pomocí stejné služby, kdy je status služby nastaven na *Server Ack*.
- **Zprávy s obsahem komunikace** - Zprávy s vlastním obsahem komunikace jsou předávány pomocí zpráv služby *Message* určené hexadecimálním číslem 6. Odchozí zprávy mají status služby nastaven na Offline a příchozí zprávy mají tento status na staven na *Server Ack*.

6.9.2 Činnost IRI-IIF

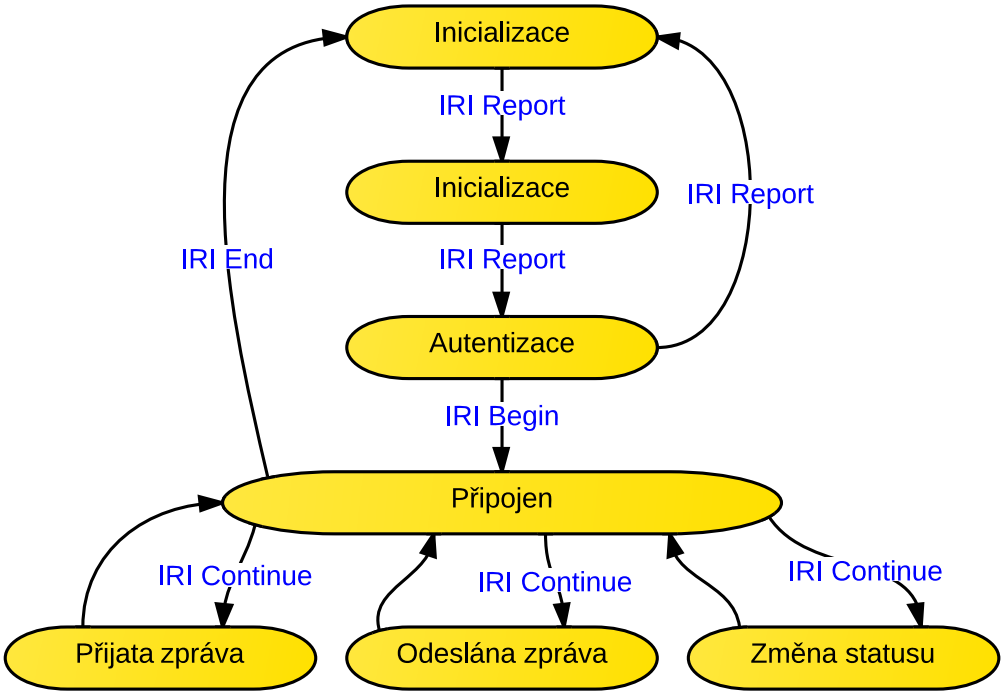
IM modul pro protokol YMSG zpracovává celé TCP spojení a analyzuje jednotlivé zprávy protokolu na aplikační úrovni. Po úspěšném potvrzení zahájení autentizace je vytvořena zpráva *IRI Report* informující o připojení k serveru. Po dokončení autentizace je vytvořena zpráva *IRI Report* informující o jejím úspěchu či neúspěchu. Jakmile klient přijme seznam kontaktů ze serveru, je vytvořena zpráva *IRI Begin* informující o začátku komunikace. Dále jsou tvořeny zprávy *IRI Continue* informující o přijetí nebo odeslání zpráv s obsahem komunikace. Zprávy *IRI Continue* jsou také vytvářeny podle zachycených zpráv upravujících status klienta. Po odpojení klienta od sítě nebo po uzavření TCP spojení je vytvořena zpráva *IRI End*.

Tabulka 6.12 zachycuje zprávy IRI generované modulem pro YMSG, tabulka 6.13 pak doplňuje identifikátory detekované pro tyto zprávy.

6.10 Simple Mail Transfer Protocol (SMTP)

6.10.1 Popis protokolu

Protokol *Simple Mail Transfer Protocol* (SMTP) [45] je aplikační protokol a zároveň internetový standard na odesílaní elektronické pošty (e-mail). Slouží pro přenos pošty mezi odesílatelem a adresátem, odesílatelem a jeho poštovním serverem nebo mezi dvěma různými poštovními



Obrázek 6.11: Stavový diagram funkce IRI-IIF protokolu YMSG

servery. I když je možné odesílat poštu přímo adresátovi, v drtivé většině se pošta odesílá na poštovní server adresáta.

Architektura protokolu SMTP je znázorněna na obrázku 6.12, které popis je následující:

- *Mail User Agent* (MUA) - Program na správu e-mailů (minimálně příjem a odesílaní), který je umístěn na koncových stanicích.
- *Mail Submission Agent* (MSA) - Proces na serveru, který se stará o příjem e-mailů od MUA. Mezinárodní organizace IANA, která se kromě jiného stará o přiřazování čísel portů na transportní vrstvě aplikačním protokolem, přiřadila procesu MSA port 587. Z historických důvodů ale proces MSA může naslouchat i na portu číslo 25.
- *Mail Transfer Agent* (MTA) - Proces na serveru, který se stará o přenos doručeného e-mailu na jiný (cílový) SMTP server. Procesu MTA byl organizací IANA přiražený port 25.
- *Mail Delivery Agent* (MDA) - Proces na serveru, který se stará o doručení e-mailu do schránky adresáta umístěném na serveru, na kterém běží cílový MSA. Přenos e-mailu mezi procesem schránkou adresáta a MUA již není zabezpečen protokolem SMTP, ale k tomu vyhrazenými protokoly (POP3, IMAP).

Činnost jednotlivých bloků architektury při posílaní e-mailu je následující:

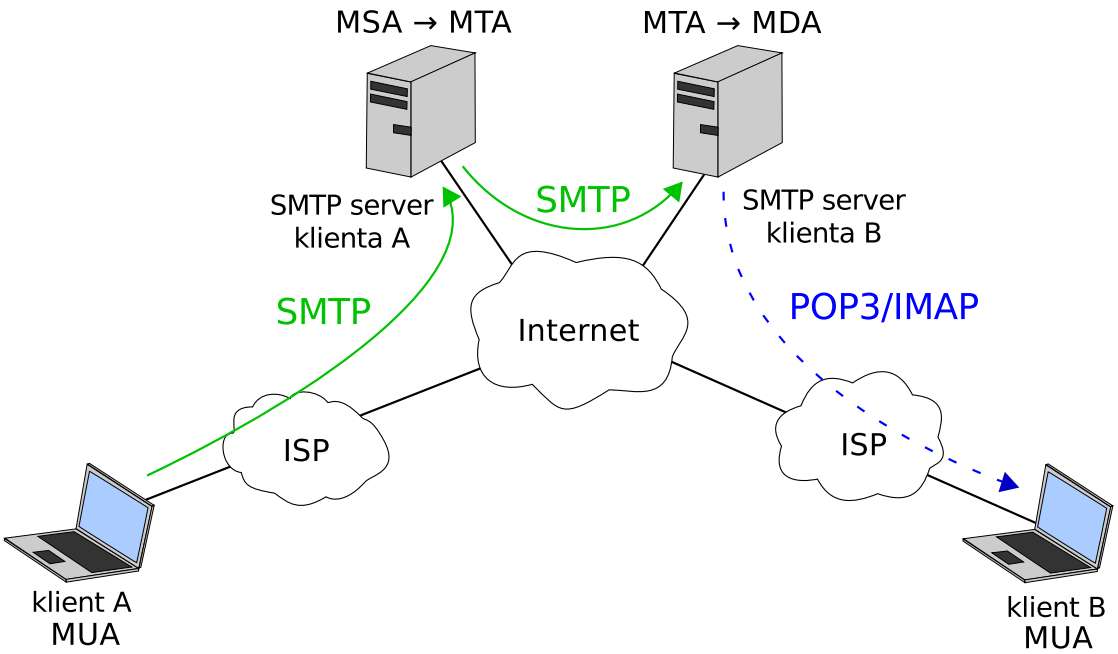
1. Odesilatel zprávy pomocí programu MUA napíše zprávu a odešle ji pomocí protokolu SMTP na svůj poštovní server.

Událost	Zpráva IRI
Dokončení autentizace	IRI Report
Úspěšné připojení k IM síti	IRI Begin
Byly zjištěny všechny potřebné údaje z probíhající komunikace a IRI Begin zatím nebyla vytvořena	IRI Begin
Odeslaná nebo přijatá zpráva	IRI Continue
Změna statusu	IRI Continue
Odpojení od IM sítě	IRI End
Reakce na sledované události v případě, že IRI Begin nebyla vytvořena	IRI Report

Tabulka 6.12: Zprávy IRI generované modulem YMSG.

Událost	Identifikátory
Přijmutí nebo odeslání zprávy	Identifikátory YMSG_LOGIN obou komunikujících stran obalené identifikátorem YMSG
Ostatní zprávy	Identifikátor YMSG_LOGIN sledovaného klienta
Zprávy kdy není znám identifikátor sledovaného klienta	Identifikátory YMSG_LOGIN ani YMSG nejsou uváděny

Tabulka 6.13: Identifikátory detekované modulem YMSG.



Obrázek 6.12: Zobrazení architektury SMTP protokolu

2. Na poštovním serveru odesilatele běží proces MSA, který se stará o přijetí zprávy od MUA.

3. Na tom stejném serveru jako se nachází proces MSA se může nacházet i proces MTA starající se o přenos e-mailu na další poštovní server protokolem SMTP. Typicky je dalším poštovním serverem server příjemce, na kterém se e-mail příjme taktéž MTA procesem. Nepoužije se MSA proces, protože ten se stará pouze o příjem e-mailu od MUA. V případě, že je poštovní server odesilatele a příjemce stejný, není nutné využít proces MTA.

4. Po přijetí e-mailu na poštovní server příjemce je e-mail lokálně uložen. Po připojení příjemce e-mailu (proces MUA) na server se spustí proces MDA, který se dále postará o doručení adresátovi.

SMTP ke své činnosti využívá výhradně transportní protokol TCP. Protokol TCP sám zabezpečuje spolehlivý přenos dat mezi oběma účastníky tak, aby nedošlo k ztrátě paketů nebo k změně jejich pořadí. Tím pádem se protokol SMTP nemusí starat o zabezpečení spolehlivého přenosu.

Komunikace protokolem SMTP funguje na principu příkaz - odpověď. Každý příkaz nebo odpověď je poslán v textovým formátu a ukončen koncem řádku (znaky CRLF). Klient odešle příkazem parametr komunikace (např. adresu příjemce) a server pomocí číselných kódů reaguje na daný příkaz (např. kódem 250 - potvrzení parametru). Součástí kterékoli odpovědi ze strany serveru může kromě čísla kódu být i libovolný text, oddělený od kódu mezerou a ukončený koncem řádku. Tento text je volitelný a slouží na zpřesnění významu dané zprávy, např. doména serveru při připojení na server.

Původně normou definované příkazy [45], se rozšířili o tzv. *Extended SMTP (ESMTP)* [46] příkazy. Tyto nové příkazy zavedli např. příkaz *AUTH*, který reprezentuje autentifikaci iniciátora přenosu. V dnešní době se dá říct, že minimálně z důvodu autentifikace, všichni klienti a servery podporují a zároveň preferují tyto ESMTP příkazy, proto ve zbytku této sekce budu vždy předpokládat využití těchto příkazů. Protokol SMTP je navržený tak, že formát zpráv posílaných mezi uživatelem a serverem nebo mezi dvěma servery je úplně stejný. Jiná je jenom forma ověřovaní odesílatele e-mailu.

1. **SMTP server odesílatele** ověřuje, jestli uživatel odesílá e-mail ze správné zdrojové adresy v rámci dané (svojí) domény. V případě domény "domain.com" je tím zajištěno, aby uživatel s adresou "user@domain.com" nemohl odeslat e-mail z adresy "administrator@domain.com". Server může odesilatele ověřit buď na základě autentifikace pomocí přihlašovacího jména a hesla nebo podle zdrojové IP adresy.

2. **Zbývající SMTP servery** již neověřují jestli uživatel použil správnou zdrojovou e-mailovou adresu v rámci domény. Uživatelé se totiž ověřují pouze na svých vlastních serverech. Zbývající SMTP servery ověřují jenom to, zda je SMTP server odesílatele oprávněn odesílat e-maily z příslušné domény. K ověření SMTP serveru slouží protokol *Domain Name System (DNS)* záznam typu *SPF*, který by měla obsahovat každá doména. V záznamu *SPF* je uvedeno, ze kterých IP adres je možné odesílat e-maily z dané domény. Jedná se o IP adresy SMTP serverů, ne uživatelů. V případě adresy "user@domain.com" tak každý server kontroluje, zda je zdrojová IP adresa obsažena v *SRT* záznamu domény "domain.com".

Zjednodušený postup při přenášení zprávy je následující:

1. Klient se serverem naváže komunikaci a dohodnou se na metodě autentifikace.

2. Klient se v případě nutnosti podle zvolené metody autentifikuje.

3. Následuje odeslání identifikátorů (e-mailových adres) odesílatele a příjemce (resp. příjemců).

4. Klient odešle na server odesílanou zprávu.

5. Posledním krokem je ukončení spojení, které inicializuje klient.

Samotný obsah odesílané zprávy je kódovaný ve formátu *Internet Message Format (IMF)* [66], který kromě čistého textu a předmětu zprávy obsahuje i hlavičku e-mailu. Obsahem hlavičky je např. položka odesilatel, adresát, dále čas odeslání, unikátní identifikátor dané zprávy atd. Unikátní identifikátor zprávy je zároveň jediný identifikátor, který se zjišťuje z obsahu zprávy. Další identifikátory protokolu SMTP jsou e-mail odesílatele a email všech příjemců. Avšak tyto identifikátory nejsou získávány z obsahu zprávy, ale z příkazů odesílatele.

```
> 220 smtp.server.com Simple Mail Transfer Service Ready
  < EHLO client.example.com
> 250-smtp.server.com Hello client.example.com
> 250-SIZE 1000000
> 250 AUTH LOGIN PLAIN CRAM-MD5
  < AUTH LOGIN
> 334 VXNlcm5hbWU6
  < adlxdkej
> 334 UGFzc3dvcmQ6
  < lkujsefxlj
> 235 2.7.0 Authentication successful
  < MAIL FROM:<mail@samlogic.com>
> 250 OK
  < RCPT TO:<john@mail.com>
> 250 OK
  < DATA
> 354 Send message content; end with <CRLF>.<CRLF>
  < obsah odesílanej zprávy
  < .
> 250 OK, message accepted for delivery: queued as 12345
  < QUIT
> 221 Bye
```

Obrázek 6.13: Příklad komunikace mezi klientem a serverem

V případě, že je zpráva odesílaná prostřednictvím poštovních serverů, je nutné, aby servery věděli kam mají zprávu poslat dále. K tomu využívají protokol DNS [51], konkrétně záznam typu MX [52]. Server si z e-mailu adresáta zjistí doménu (část e-mailu za znakem "@"), ve které vyhledá MX záznam. Ten obsahuje název serveru, na který se daná zpráva odešle. Způsob, jak si adresát stáhne poštu ze svého poštovního serveru již není součástí protokolu SMTP, ale jiných protokolů (např. POP3 [54] a IMAP [12]).

Kromě nešifrované komunikace umožňuje protokol SMTP i šifrovanou komunikaci [29]. První možností šifrování je na aplikační úrovni. Její podporu dává server najevo v odpovědi na příkaz

EHLO, která obsahuje seznam podporovaných rozšiřujících funkcí. Když se v tomto seznamu nachází parametr STARTTLS, je šifrovaná komunikace podporována. Když klient šifrovanou komunikaci podporuje a preferuje, namísto příkazu *AUTH* odešle příkaz *STARTTLS*. Server na daný příkaz odpoví kódem 220 (*Go ahead*). Všechna následující komunikace bude probíhat šifrovaně a ukončení spojení je možné detekovat jenom na úrovni ukončení TCP spojení. Další možností použití šifrované komunikace je pomocí mezivrstvy umístěné mezi transportním protokolem TCP a aplikačním SMTP. Jedná se o mezivrstvu SSL/TLS, jejíž úlohou je zabezpečit šifrovaný přenos dat bez nutnosti změny zpráv aplikačního protokolu. Hlavní výhodou tohoto řešení je, že celá komunikace od vytvoření spojení až po ukončení je šifrována. Tím pádem není vůbec možné zjistit, zda-li se jedná o komunikaci protokolem SMTP nebo kterýmkoliv jiným. Na základě skutečnosti, že nad TCP vrstvou se nenachází protokol SMTP, ale SSL/TLS je použito i jiné číslo portu. Organizací IANA bylo komunikaci skrz SSL/TLS přiřazeno číslo 465.

6.10.2 Činnost IRI-IIF

Protokol SMTP je aplikačním protokolem, který vnáší do analýzy protokolu jeden velmi významní problém. Aplikační protokol je možné nastavit tak, aby využívali libovolné porty (ne jenom ty přiřazené organizací IANA), a proto není snadné říci, je-li některá komunikace SMTP komunikací. Navzdory tomu, že organizace IANA přidělila protokolu SMTP TCP porty číslo 25 a 587, nejsou tyto údaje příliš brány do úvahy. Modul IRI-IIF proto musí analyzovat každé TCP spojení, ve kterém vyhledává zprávy ve formátu SMTP protokolu.

Každé spojení je identifikováno IP adresami a porty obou účastníků komunikace spolu s typem použitého transportního protokolu (v tomto případě vždy TCP). Analýza každého spojení začíná výhradně při navazovaní TCP spojení. Pokud by modul náhodou zachytil zprávy z již probíhající komunikace, tak o danou komunikaci nebude mít zájem.

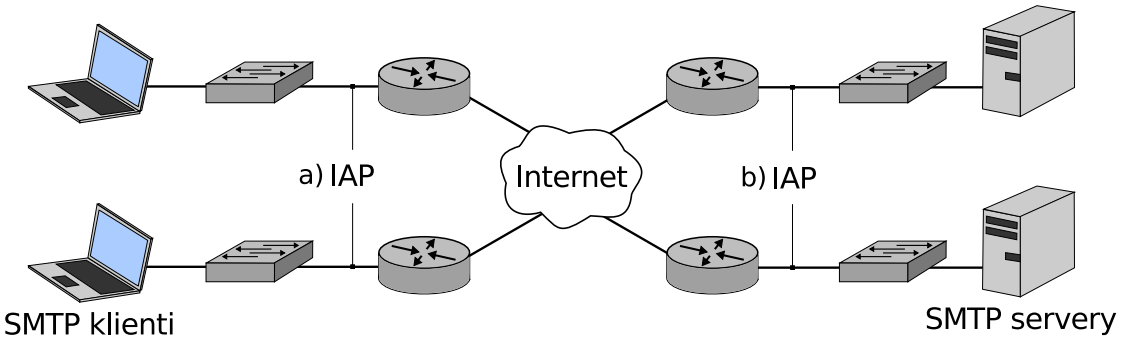
V průběhu analýzy komunikace se modul snaží detekovat e-mail odesilatele, e-mail příjemce (resp. příjemců) a počet příjemců. Výsledný stavový automat protokolu se nachází na obrázku 6.16. Na obrázku 6.15 je zobrazena zjednodušená verze tohoto automatu, jehož stavy jsou následující:

- *Inicializace* - v síti není žádné začínající TCP spojení
- *Spojení navázáno* - odesílatel navázal spojení se SMTP serverem
- *Uživatel přivítán* - odesílatel přijal od serveru seznam podporovaných rozšiřujících funkcí
- *Pokus o autentifikaci* - odesílatel se pokouší autentifikovat
- *Autentifikace úspěšná* - odesílatel se úspěšně autentifikoval
- *Specifikování příjemců* - odesílatel odesílá na server seznam příjemců e-mailu
- *Odesílaní zprávy* - odesílatel začal s přenosem e-mailové zprávy
- *Zpráva úspěšně odeslána* - odesílatel úspěšně přenést e-mail na SMTP server

Topologie spolu s předpokládaným umístěním IRI-sondy je zobrazená na následujícím obrázku 6.14.

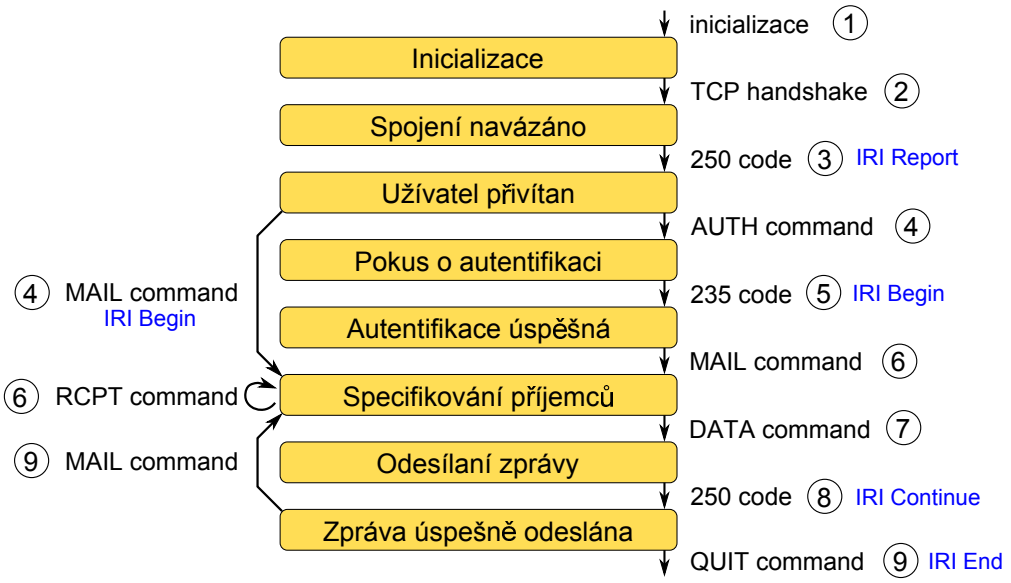
Vzhledem k tomu že v rámci TCP komunikace může být odesílatelem zprávy odesílatel e-mailu i SMTP server, použil sem při popisu modulu IRI-IIF následující konvenci:

- **klient** - odesilatel e-mailu v rámci TCP spojení



Obrázek 6.14: Ukázková topologie protokolu SMTP s předpokládaným umístěním IRI-sondy: a) na straně ISP, b) na straně poskytovatele služby

- **server** - příjimce e-mailu v rámci TCP spojení (vždy se jedná o SMTP server)
- **odesílatel** - uživatel, který odesílá e-mail



Obrázek 6.15: Stavový diagram IRI-IIF protokolu SMTP

Činnost bloku IRI-IIF se řídí podle následujících pravidel:

1. Počátečním stavem je *Inicializace*. Modul IRI-IIF se pokouší detekovat novou, ještě neprobíhající TCP komunikaci.
2. Nové TCP spojení se detekuje zachycením TCP handshake, které je nutné vykonat na začátku každého spojení. Klient se se serverem dohaduje na sekvenčních číslech, které budou nadále používat. Následně se modul přesune do stavu *Spojení navázáno*.

3. Po úspěšném navázaní spojení odešle server uvítací zprávu s kódem 220 (*Service ready*). Klient po přijetí zprávy odešle příkaz *EHLO*, kterým žádá server o identifikace a inicializaci SMTP spojení. Server na příkaz *EHLO* odpoví seznamem parametrů označených kódem 250 (*Requested mail action okay, completed*) a oddělených konci řádků. V tomto seznamu parametrů se nachází identifikace SMTP serveru a podporované rozšiřující funkce. Po zachycení zprávy obsahující kód 250 se vykoná přechod do stavu *Uživatel přivítán* a odešle se zpráva *IRI Report*.

4. V závislosti na konfiguraci klienta se může začít autentifikace klienta nebo specifikace e-mailové adresy odesilatele.

   - V případě komunikace mezi odesílatelem e-mailu a jeho poštovním serverem se klient většinou pokusí autentizovat. Jako první zprávu odešle klient příkaz *AUTH* s parametrem obsahujícím typ autentifikace. Modul se přesune do stavu *Pokus o autentizaci*.
   - Druhá možnost je v případě komunikace mezi dvěma SMTP servery nebo když je odesílatel e-mailu na svém poštovním serveru autentizovaný jiným způsobem (např. zdrojová IP adresa) případně vůbec. Za těchto okolností klient odešle příkaz *MAIL* s parametrem obsahujícím zdrojovou e-mailovou adresu odesílatele. E-mail odesílatele je ze zprávy dekódovaný a odeslaný zprávou *IRI Report* do IRI-Core. Současně s odesláním zprávy IRI se vykoná přesun do stavu *Specifikace příjemců*.

5. Samotný průběh autentifikace ve stavu *Pokus o autentizaci analyzovaný*. Na závěr tohoto procesu server odpoví klientovi buď s kódem 235 (*Authentication successful*) reprezentujícím úspěšnou autentizaci na základě kterého se modul přesune do stavu *Autentifikace úspěšná* a odešle se zpráva *IRI Begin*, nebo s kódem 535 (*Authentication failed*) reprezentující neúspěšnou autentizaci, po kterém se odešle zpráva *IRI Report*.

6. Po úspěšné autentizaci odešle odesilatel zprávy příkaz *MAIL*, ve kterém uvede e-mail odesilatele. Po specifikování odesilatele e-mailu klient začne odesílat seznam příjemců e-mailu. Počet příjemců je minimálně jeden, maximum není protokolem SMTP specifikováno. Klient odešle příkaz *RCPT* s prvním příjemcem zprávy a následně čeká na přijetí potvrzení příjemce serverem. K potvrzení příjemce server opět využije odpověď s kódem 250 (*Requested mail action okay, completed*). Až po potvrzení příjemce může klient odeslat dalšího příjemce pomocí stejného příkazu *RCPT*. Formát parametru zprávy RCPT může mít vícero podob. Nejzákladnější podoba je zadání plné e-mailové adresy příjemce spolu s doménou. Další možností je specifikování e-mailu příjemce bez domény. Doména příjemce je tak stejná jako doména odesilatele zprávy a modul proto k emailu příjemce tuto doménu přidá. Poslední možností specifikace příjemce je zadání hodnoty "postmaster". Hodnota "postmaster" reprezentuje správce poštovního serveru odesilatele, přičemž při této hodnotě nejsou nerozlišované velké a malé písmena. Hodnota "postmaster" se bez přidání domény později odešle zprávou *IRI Continue*.

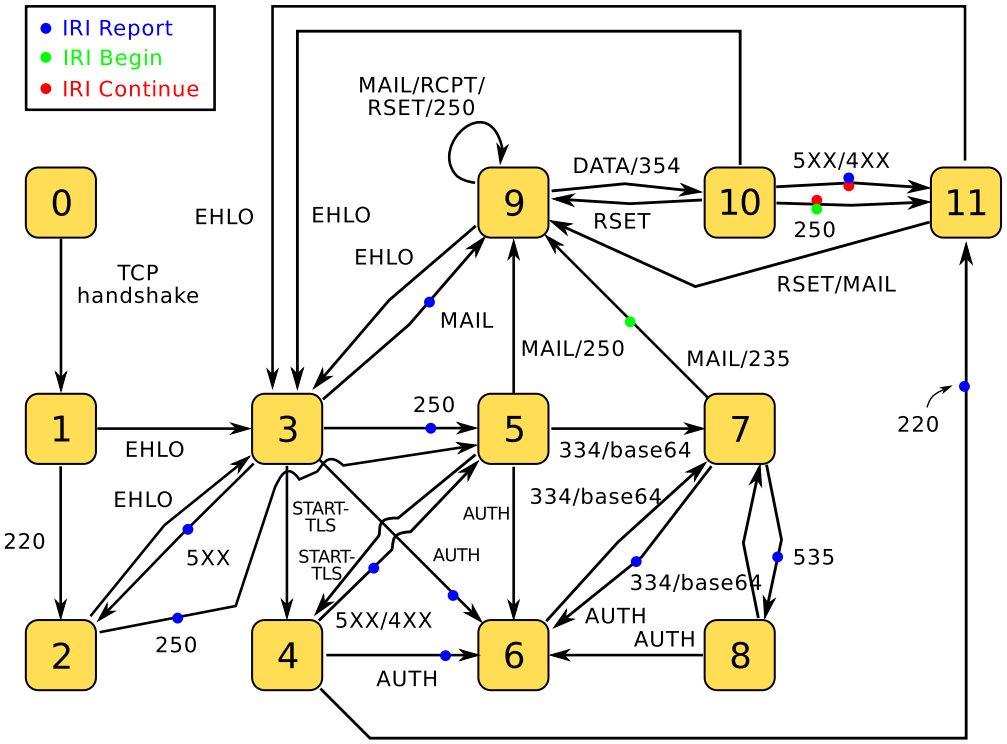
7. Po zadání posledního příjemce zprávy příkazem *RCPT* následuje samotný přenos obsahu zprávy. Klient tento stav dává najevo odesláním zprávy *DATA*. Po potvrzení zprávy serverem začne klient odesílat zprávu a modul IRI-IIF přejde do stavu *Odeslání zprávy*.

8. Konec odesílaní zprávy je detekovaný odesláním znaku "." ze strany klienta. Server následně odpoví buď s kódem 250 (*Requested mail action okay, completed*) reprezentujícím, že všechno proběhlo v pořádku nebo s kódem reprezentujícím číslo chyby. Při odeslání kódu 250 modul odešle zprávu *IRI Continue* se všemi parametry, které dokázal zjistit (e-mail

odesilatele, e-maily příjemců, atd.). Důvod proč nejsou e-mailové adresy po zadaní příkazu *MAIL* nebo *RCPT* odesílány okamžitě je popsán v posledním bodu - příkaz *RSET*. Zachycením kódu 250 se modul přesune do stavu *Zpráva úspěšně odeslaná*.

9. Po úspěšném odeslaní zprávy může klient příkazem *QUIT* ukončit SMTP spojení, po kterém bude následovat ukončení TCP spojení. Modul tak odešle zprávu *IRI End* a vrátí se do stavu *Inicializace*. Avšak když má klient další zprávy na odeslaní, nemusí ukončovat spojení a vytvářet nové. Namísto toho může znovu odeslat zprávu *MAIL* s platnou hodnotou e-mailu odesilatele čímž inicializuje přenos další zprávy. Modul na touto zprávu zareaguje odesláním správy *IRI Continue* a přechodem do stavu *Specifikovaní příjemců*.

10. Podél celé doby komunikace může klient odeslat příkaz *RSET* nebo *QUIT*. Při příkazu *QUIT* modul detekuje ukončení spojení a v případě, že během spojení byla odeslaná zpráva *IRI Begin*, tak se odešle zpráva *IRI End*. V opačném případě (zpráva *IRI Begin*nebyla odeslána) se odešle zpráva *IRI Report*. Příkaz *RSET* dává smysl jenom po autentizaci uživatele a slouží k vynucení smazání všech parametrů zprávy (e-mail odesilatele a příjemců) na straně serveru. Tyto parametry je tak nutné zadat znovu. Z tohoto důvodu se identifikátory neposílají hned po detekci. Při detekovaní příkazu *MAIL* po příkaze *RSET* se modul přesune do stavu *Specifikace příjemců*.



Obrázek 6.16: Úplný stavový diagram IRI-IIF protokolu SMTP

Tabulka 6.14 zachycuje zprávy IRI generované modulem pro SMTP.

Událost	Zpráva IRI	Popis události	Identifkátory
Zachycení odpovědi č. 250, po navázání TCP spojení	IRI REPORT	Uživatel se připojil na SMTP server	IP adresy, TCP porty
Zachycení odpovědi č. 235	IRI BEGIN	Uživatel se úspěšně autentifikoval	Původní MAC adresa a IPv6 adresa
Zachycení odpovědi č. 535	IRI REPORT	Uživatel se neúspěšně autentifikoval	Původní MAC adresa a IPv6 adresa
Zachycení odpovědi č. 220	IRI CONTINUE	Mezi odesielatelem a serverem bylo vytvořené spojení SSL.	IP adresy, TCP porty
Zachycení odpovědi č. 250	IRI CONTINUE	Uživatel úspěšně odeslal e-mail.	
Zachycení odpovědi č. 4XX	IRI CONTINUE	Uživatel se neúspěšně pokusil odeslat e-mail.	
Zachycení příznaku TCP FIN nebo příkazu QUIT nebo zachycení odpovědi č. 221	IRI REPORT / IRI END	Uživatel ukončuje/ukončil spojení se SMTP serverem	IP adresy, TCP porty

Tabulka 6.14: Zprávy IRI generované modulem SMTP.

Aktuální verze modulu pro zpracování SMTP je implementována pouze pro verzi IRI-IIF spustitelnou na vysokorychlostní sondě. Bez vysokorychlostní sondy není funkcionalita tohoto modulu v rámci SLIS dostupná.

6.11 Identifikace počítače pomocí odchylky v měření času

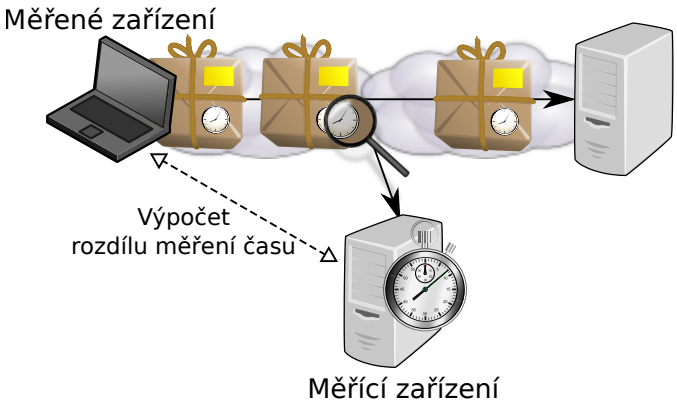
6.11.1 Popis metody

Každé zařízení v počítačové síti má vlastní vnitřní hodiny. Signál hodin je produkován krystalovým oscilátorem, který kmitá na určité frekvenci. Tato frekvence je závislá na parametrech výroby a typu použitého krystalu. Mechanické nepřesnosti vznikající během výroby způsobují, že i krystaly stejného typu, řady i výrobního data mohou mít odlišné frekvence a systémový čas, který se pomocí vnitřních hodin nastavuje, nebude nikdy úplně přesný [47].

Identifikace [47, 64, 57] počítače založena na odchylce systémového času měřeného a měřícího zařízení je znázorněna na obrázku 6.17. Měřící zařízení hledá ve všech paketech procházejících sítí časová razítka. Po přijetí dostatečného počtu paketů se vypočítá odchylka hodin, tedy změna rozdílu (první derivace) mezi časem udaným časovými razítky z měřeného zařízení a systémovým časem měřícího zařízení [47] odchylku. Vypočtená odchylka by měla být unikátní [47] pro každé zařízení. Posun hodin se měří v počtu mikrosekund o které se hodiny rozejdou každou sekundu (počet částí za milión částí, *parts per million – ppm*). Přesnost měření se pohybuje okolo 1 ppm [47, 72].

K získání časové informace ze vzdáleného zařízení lze použít následující zdroje dat:

- Protokol TCP může volitelně obsahovat pole TCP Timestamp Option (TSopt) [38] v hla-



Obrázek 6.17: Metoda odhadu posunu měření času je založena na sběru časových razítek z paketů pozorovaných v síti. Pomocí metody je možné odhadnout odchylku při měření času koncových stanic i serverů.

vičce každého segmentu. Frekvence změny hodnoty časového razítka závisí [47, 57] na operačním systému a hardwarové specifikaci počítače. Je tedy nutné frekvenci zpětně dopočítat. Nevýhodou tohoto zdroje je, že se časové značky nevyskytují ve výchozím stavu ve spojeních TCP iniciovaných z počítačů s OS Windows [47, 57]. OS systémy odvozené od Linuxu, BSD a Apple OS časová razítka používají [57]. U těchto operačních systémů je možná pasivní identifikace bez vědomí měřeného.

- Protokol ICMP definuje zprávy Timestamp (typ 13, kód 0) a Timestamp Reply (typ 14, kód 0). Výhodou je pevná frekvence 1 kHz, nevýhodou nutnost spolupráce měřeného zařízení, filtrace zpráv ICMP firewally, průchod přes překladač adres (NAT). ICMPv6 zprávy typu Timestamp nedefinuje a metoda je tedy dostupná jen pro IPv4.
- Vkládání časové informace na aplikační vrstvě popsali poprvé Ding-Jie a kol. [33]. Časové značky se získávají od klienta na základě kódu v jazyce JavaScript dodaného spolupracujícím serverem. Klient poté periodicky zasílá časové značky získané z funkce getTime() technikou AJAX a tím umožňuje výpočet odchylky. Nevýhodou metody je nutná spolupráce s webovým serverem a klient musí daný server navštívit a setrvat na něm po dobu alespoň několika minut [57]. Výhodou je nezávislost na OS [57], prohlížeči [57] a pevná frekvence 1 kHz.
- Samotné aplikační protokoly obsahují časové značky, např. HTTP (čas vygenerování stránky na serveru), XMPP, SMTP, RTP aj. [57]. Nevýhodou těchto značek je nízká frekvence a tedy nutnost delšího času pro změření odchylky [53, 76].

V rámci činnosti skupiny jsme zkoumali [39, 28, 64, 57] tuto metodu a její aplikaci v prostředí pro zákonné odposlechy. Bohužel se ukázalo, že metoda není dostatečně spolehlivá a ve větších sítích není použitelná [57]. I přesto je možné její experimentální nasazení v malých sítích, případně pod expertním dohledem, který by vyloučil chybnou identifikaci zařízení.

6.11.2 Činnost IRI-IIF

Pro výpočet odchylky časových značek jsme vyvinuli program PC Fingeprinter (*PCF*) [63]. Program analyzuje přijaté pakety zda obsahují časovou známku a po přijetí dostatečného počtu

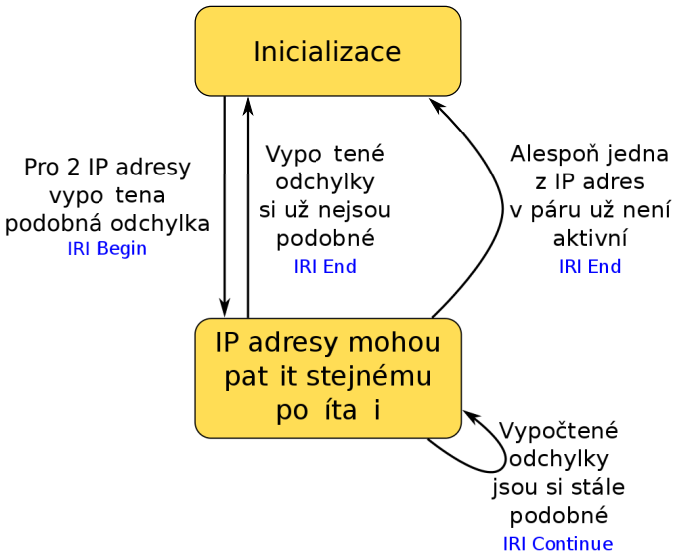
paketů vypočte odchylku hodin. Konkrétní počet paketů nutný pro výpočet je uveden v konfiguračním souboru a výchozí hodnota je 50.

V PCF je pro každou IP adresu a metodu sběru časových známek uložen seznam IP adres, u kterých bylo spočtena podobná odchylka (rozdíl spočtených odchylek je menší než 1 ppm). PCF informace o podobnosti vypočtené odchylky v měření času měřených zařízení dále předává do jádra IRI-IIF pomocí konverzního skriptu. PCF společně s konverzním skriptem tvoří modul IRI-IIF pojmenovaný *clockskew*.

Modul *clockskew* zjišťuje pro které IP adresy byla spočtena podobná odchylka, takové IP adresy mohou patřit ke stejnému počítači. Za pomoci modulu *clockskew* je tedy možné zjistit IP adresy jednoho počítače i pokud nejsou k dispozici informace pro činnost modulů pro ND, DHCP apod. (popsané v předchozích sekcích této kapitoly). Navíc je možné identifikovat i IP adresy patřící k různým rozhraním stejného počítače. Nevýhodou však je to, že odchylka v měření času není dostatečně unikátní [57] pro použití ve velkých sítích.

Diagram činnosti modulu je zobrazen na obr. 6.18. Při přidání nově objeveného páru IP adres do vnitřního seznamu modulu *clockskew* se odesílá zpráva *IRI Begin* spolu s dvojicí IP adres, které mají podobnou odchylku. Další kontrola proběhne opět po přijetí dostatečného počtu paketů nebo uplynutí 5 minut. Pokud jsou hodnoty odchylky stále podobné, odesílá se *IRI Continue*.

Zprávy IRI se odesílají vždy pro dvojici podobných adres s podobným *clockskew*. V případě, že se odchylka změnila a vypočtené odchylky hodin počítačů identifikovaných pomocí IP adresy si přestaly být podobné (rozdíl se zvětšil nad 1 ppm), modul odešle zprávu *IRI End* s touto dvojicí adres. Zpráva *IRI End* může být odeslána také v případě, že po předem daný časový limit nebyl přijat žádný paket (přesná hodnota se také nastavuje v konfiguračním souboru). Modul *clockskew* analyzuje přijaté pakety zda obsahují časovou známku a po přijetí dostatečného počtu paketů vypočte odchylku hodin. Konkrétní počet paketů nutný pro výpočet je uveden v konfiguračním souboru a výchozí hodnota je 50.



Obrázek 6.18: Stavový diagram modulu IRI-IIF pro časové značky.

Tabulka 6.15 zachycuje zprávy IRI generované modulem pro detekci identity na základě časo-

Událost	Typ zprávy IRI
Spárování IP adres na základě podobné odchylky	BEGIN
Přepočítání odchylky, IP adresy se stále podobají	CONTINUE
Změna odchylky, IP adresy se přestaly podobat	END
Vypršení časového limitu pro aktivní počítače	END

Tabulka 6.15: Zprávy IRI generované modulem pro detekci identity pomocí odhadu nepřesnosti měření hodin počítače.

vých značek. Modul se identifikuje vždy identifikuje řetězcem `clockskew(_source)`, kde `source` je zdroj odhadu nepřesnosti měření, v současnosti jeden z trojice `tcp`, `javascript`, `icmp`.

V případě spuštění modulu v konfiguraci NAT (není možné spustit při detekci odchylky pomocí protokolu ICMP) modul vyhledává konkrétní spojení na úrovni TCP, která mají shodný odhad odchylky. V ostatních případech modul inzeruje podobnost odchylky u dvou IP adres.

6.12 Zjišťování identity z kontrolérů SDN

6.12.1 Popis metody

SDN (softwarově definované sítě) jsou jedním z moderních přístupů k počítačovým sítím. Jedná se o dynamickou, adaptivní architekturu s nízkými náklady, která odděluje kontrolní a datovou část síťových zařízení. Jednotlivá síťová zařízení jsou programovatelná a infrastruktura abstrahovaná pro aplikace a síťové služby.

Architektura síťových zařízení

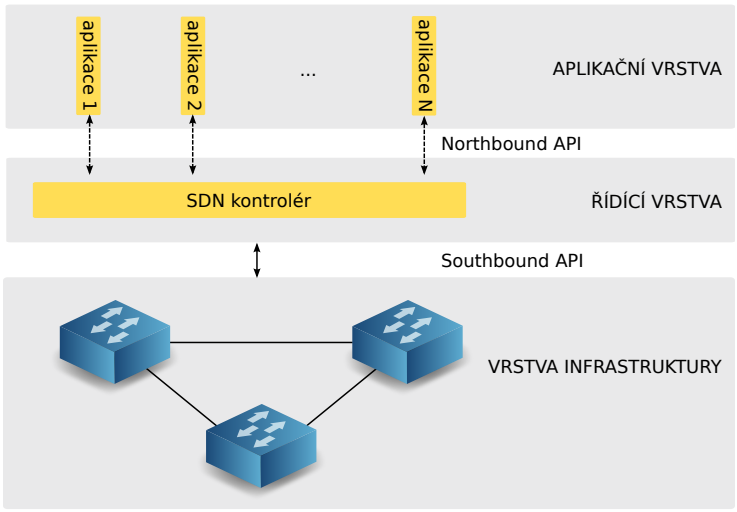
Klasické síťové zařízení obsahuje vlastní hardware pro přeposílání paketů (data plane) a operační systém v software pro management a logiku (control plane). Všechna tato zařízení mají pevně danou funkcionalitu od výrobce, složitý operační systém a jsou uzavřená k inovacím.

Řídící část (Control Plane) Funkce řídící části zahrnují konfiguraci systému, vysokoúrovňový management a statistiky. Ve směrovačích slouží ke správě směrovací a ppřepínací tabulky. Vzhledem k tomu, že funkce v řídící části se neprovádí nad každým paketem, nejsou obvykle limitovány rychlostí ani časem a proto mohou být implementovány v software.

Datová část (Data Plane) Mezi funkce datové části patří spíše jednoduché úkony, které se týkají především přijímání, zpracování a řízení paketů. Funkce datové části se provádí nad každým paketem, což vyžaduje vysokou výkonnost v reálném čase a implementaci v hardware.

Princip SDN

SDN odděluje rozhodovací logiku a rychlé přepínání paketů v hardware. Zatímco data plane zůstává na síťovém zařízení, vyšší logika se přesouvá do odděleného kontroleru. Síťové zařízení tedy obsahuje rychlý HW přo přepínání paketů a jednoduchý operační systém, který sám o sobě musí umět pouze komunikovat s kontrolerem a na základě pokynů od kontroleru přeposílat pakety mezi porty. Architektura SDN je znázorněna na obrázku 6.19.



Obrázek 6.19: Architektura SDN.

Síťové zařízení V SDN je síťové zařízení abstrahováno od specifické činnosti jako je router, směrovač, firewall nebo load-balancer. Má za úkol pouze provádět určité operace nad daty a rychle přeposílat pakety na základě instrukcí od kontroleru.

SDN kontrolér SDN kontrolér je software, který kontrolu nad množinou zdrojů jednotlivých datových částí. Může být implementován jako větší množství programových částí, které mohou být rozmístěny v několika fyzických zařízeních. Všechny části si udržují synchronizovaný a konzistentní pohled na topologii a stav sítě.

- NOX/POX je široce rozšířený open source kontrolér implementovaný v Pythonu, který byl vyvinut ze staršího NOXu. Poskytuje asynchronní, událostmi řízené programovací rozhraní. Jádro NOXu poskytuje pomocné metody a API pro interakci s OpenFlow přepínači. K dispozici jsou i další komponenty jako sledování hostů, směrování nebo zjišťování topologie. V porovnání s dalšími kontroléry je pomalejší a méně výkonný.
- OpenDaylight je také open source projekt, který má podporu i v komerční sféře. Jeho součástí je GUI a poskytuje northbound rozhraní pro Java a HTTP API.

Komunikační kanál Komunikační kanál je rozhraní, které propojuje síťové zařízení s kontrolérem. Přes komunikační kanál kontrolér konfiguruje síťová zařízení a přijímá nebo zasílá pakety.

Programové rozhraní Síťová zařízení jsou konfigurována kontrolérem přes programové rozhraní (API). V rámci SDN se využívají Southbound a Northbound API. Southbound API je nízkoúrovňové programovací rozhraní, které obvykle obsahuje pouze nezbytnou funkcionalitu na programování síťových zařízení. Northbound API je programovací rozhraní, které zapouzdřuje nízkou úroveň instrukcí southbound API a umožňuje programovat složitější síťové funkce. Příkladem northbound rozhraní může být například JSON nebo Thrift.

Událost	Typ zprávy IRI
Nalezeno nové koncové zařízení	IRI Begin
Obnovení informace o aktivních koncových zařízeních	IRI Continue
Koncové zařízení se odpojilo	IRI End

Tabulka 6.16: Zprávy IRI generované moduly pro OpenDaylight a Pox.

6.12.2 Činnost IRI-IIF

Základním principem SDN je oddělení kontroléru od datových funkcí. Kontrolér si musí udržovat aktuální a konzistentní obraz topologie sítě, čehož lze využít při identifikaci koncového zařízení. Byly vytvořeny moduly pro SLIS zpracovávající topologií kontroleru OpenDaylight a Pox. Z kontroleru lze zjistit název/ID přepínače, port a pak identifikátory 2. a 3. vrstvy, tedy MAC adresu a IP adresy připojeného zařízení.

OpenDaylight modul

Modul zpracovává informace o topologii z kontroleru SDN OpenDaylight. S využitím HTTP GET se v předem nastavených intervalech pravidelně dotazuje na aktuální topologii sítě. Odpovědí kontroleru je seznam všech aktivních koncových zařízení (dynamicky zjištěných přes síť i nakonfigurovaných pomocí PUT API). Nový seznam se porovná s předchozím a změny jsou oznamovány IRI Core odesláním zprávy BEGIN, CONTINUE nebo END. Textový řetězec identifikující modul je "SDN-opendaylight". V tabulce 6.16 jsou shrnuty typy IRI zpráv spolu s událostmi, které vedou k jejich vytvoření.

POX modul

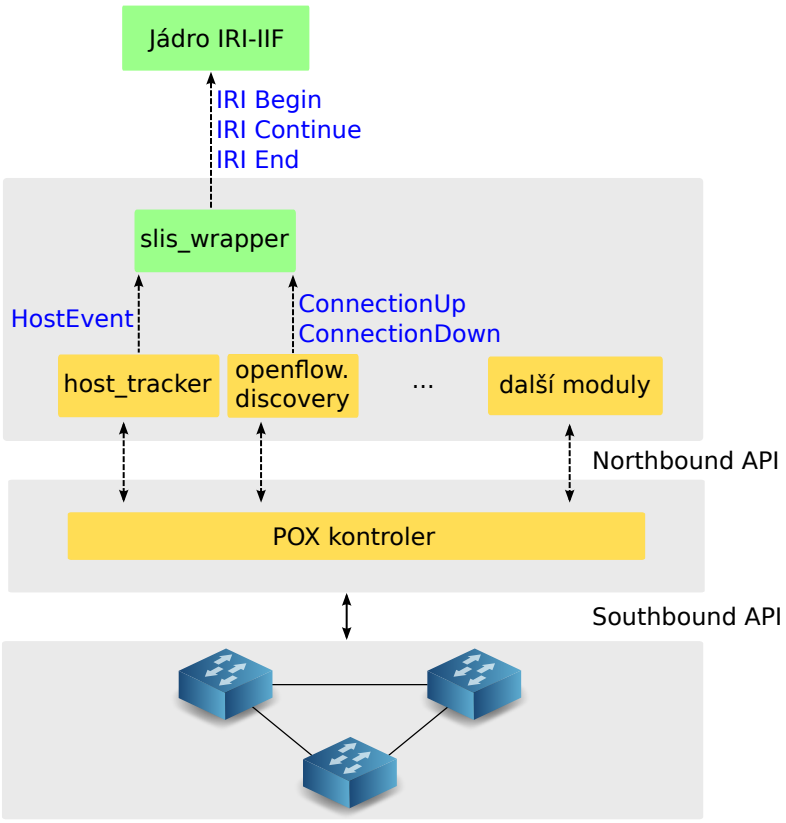
Modul zpracovává informace z kontroleru SDN Pox a je naimplementován jako modul pro Pox, který odesílá zprávy IRI Core. Změna topologie se detekuje automaticky v modulech `openflow.discovery` a `host_tracker`. Tyto moduly při změně topologie nebo stavu koncového zařízení vytvoří událost, která se zachytí v modulu `slis_wrapper`. Na základě této události se vygenerují zprávy pro IRI Core. Zachytáváním vhodných událostí (ConnectionUp, ConnectionDown, LinkEvent, HostEvent) je možné získat kompletní obraz celé topologie. Schéma modulů je uvedeno na obrázku 6.20. Textový řetězec identifikující modul je "SDN-pox".

Typy IRI zpráv a události, které vedou k jejich vytvoření jsou stejné jako pro OpenDaylight modul uvedené v tabulce 6.16.

6.13 Podporované identifikátory

Předešlé sekce popisovaly protokoly podporované IRI-IIF systému SLIS. Téměř každý modul je spojen s jedním, čí více specifickým identifikátorem, který může být použitý jako vstupní NID. Tato sekce obsahuje shrnutí NIDů podporovaných systémem SLIS.

- MAC adresa identifikující rozhraní počítače, např. síťové karty protokolu Ethernet, Wi-Fi apod. Vstupní rozhraní SLIS rozpoznává MAC adresy zapsané v textové reprezentaci 6-ti hexadecimálních dvojic oddělěných dvojtečkou, či pomlčkami, např. 20:00:00:00:00:00, 40-50-60-70-80-90. MAC adresa může být detekována moduly DHCP, RADIUS, PPPoE, ND/SLAAC a SDN.



Obrázek 6.20: Zapojení modulu slis_wrapper pro POX kontroler.

- IPv4 adresa identifikující rozhraní počítače. Vstupní rozhraní SLIS rozpoznává MAC adresy zapsané v textové reprezentaci čtyř oktetů oddělených tečkami. SLIS podporuje jak konkrétní IP adresy jednoho počítače, tak prefixy adres zadané adresou sítě a délkou prefixu. Např. 127.0.0.1, 10.0.1.0/24. IPv4 adresa může být detekována moduly DHCP, RADIUS, PPPoE, XMPP, IRC, OSCAR, YMSG, SMTP, clockskew, SDN.
- IPv6 adresa identifikující rozhraní počítače. Vstupní rozhraní SLIS rozpoznává MAC adresy zapsané v textové reprezentaci8-mi hexadecimálních čtveřic oddělených dvojtečkou, či zkrácené formě. SLIS podporuje jak konkrétní IP adresy jednoho počítače, tak prefixy adres zadané adresou sítě a délkou prefixu. Např. 2001:db8::1, 2001:db8:a::/48. IPv6 adresa může být detekována moduly RADIUS, PPPoE, DHCPv6, ND/SLAAC, XMPP, IRC, OSCAR, YMSG, SMTP, clockskew, SDN.
- IPv6 DUID je identifikátor používaný DHCPv6 pro identifikaci počítače. V praxi je však téměř vždy generován během instalace konkrétního operačního systému. Do systému SLIS se DUID vkládá pomocí prefixu *duid:* následovaným řetězcem hexadecimálních znaků, např. duid:0:3:0:1:38:72:c0:9:cb:55. IPv6 DUID je detekován jen modulem DHCPv6. SLIS nepodporuje podrobné zpracování DUID např. pro získávání MAC adresy zakódované v určitých typech DUID (mimojiné proto, že daná síťová karta nemusí být v daném počítači již přítomna).
- DHCP Client ID je identifikátor používaný protokolem DHCP pro identifikaci počítače. Typicky se generuje při instalaci operačního systému. Často je možné jej změnit v konfiguraci klienta DHCP. Do systému SLIS se DHCP Client ID vkládá pomocí prefixu *dhcpClientID:* následovaného hexadecimálním číslem, např. dhcpClientID:0x010025907d5615. DHCP Client ID je detekováno jen modulem DHCP.
- PPP Login je přihlašovací jméno protokolu PPP. Do systému SLIS se vkládá pomocí prefixu *ppp:* následovaného samotným přihlašovacím jménem, např. ppp:adam. PPP Login je detekován jen protokolem PPPoE.
- PPP Session je číslo sezení protokolu PPPoE. Typicky není vhodné specifikovat odposlech navázaný na sezení PPPoE. Pokud by to však bylo třeba je možné použít prefix *ppps* následovaný číslem sezení, apř. ppps:0001. PPP Session je podporováno jen protokolem PPPoE.
- RADIUS Login je přihlašovací jméno protokolu RADIUS. Do systému SLIS se vkládá pomocí prefixu *radius:* následovaného samotným přihlašovacím jménem, např. radius:barbora. RADIUS Login je detekován jen protokolem RADIUS.
- TCP spojení může specifikovat buď obě strany komunikace, či jen jednu z nich.
  - Prefix *tcp:* uvozuje obousměrné spojení následované čtveřicí skládající se ze dvou páru IP adresy a č. portu, např. tcp:(10.0.0.1, 4112, 83.11.26.4, 80). Obousměrné spojení TCP je detekováno moduly XMPP, IRC, OSCAR, YMSG, SMTP.
  - Prefix *tcp3:* uvozuje TCP spojení se specifikovanou jednou stranou a skládá se z IP adresy a čísla portu oddělených podtržítkem, např. tcp3:2001:db8::1_4112. TCP spojení se specifikovanou jednou stranou komunikace je detekováno modulem clockskew v režimu detekce spojení procházejících NAT.
- XMPP Login je přihlašovací jméno protokolu XMPP uvozené prefixem *xmppLogin:* následované samotným přihlašovacím jménem, např. xmppLogin:adam. XMPP Login je detekován pouze modulem XMPP.

- YMSG Login je přihlašovací jméno protokolu YMSG uvozené prefixem *ymsgLogin:* následované samotným přihlašovacím jménem, např. ymsgLogin:barbora. YMSG Login je detekován pouze modulem YMSG.
- OSCAR Login je přihlašovací jméno protokolu OSCAR uvozené prefixem *oscarLogin:* následované samotným přihlašovacím jménem, např. oscarLogin:cecilie. OSCAR Login je detekován pouze modulem OSCAR.
- IRC Login je přihlašovací jméno protokolu IRC uvozené prefixem *ircLogin:* následované samotným přihlašovacím jménem, např. ircLogin:david. IRC Login je detekován pouze modulem IRC.
- IRC Channel je název místnosti pro komunikaci protokolem IRC uvozený prefixem *ircChannel:* následovaný samotným jménem místnosti, např. ircChannel:mychannel. IRC Channel je detekován pouze modulem IRC.
- E-mail Address je uvozena prefixem *email:* následovaným samotnou e-mailovou adresou, např. email:adam@example.net. E-mailová adresa je detekována jen protokolem SMTP.
- Port přepínače SDN je uvozen prefixem *sdnConnector:* a specifikuje typ protokolu SDN, identifikaci přepínače a jeho portu oddělené podtržítkem, např. OF_00:00:00:00:00:00:00:03_1. Port přepínače SDN je podporován jen modulem pro SDN.

Kapitola 7

Instalace a používání vytvořeného systému pro zákonné odposlechy

Sec6Net Lawful Interception System (SLIS) je k dispozici na instalovatelných live DVD založených na linuxové distribuci Ubuntu 12.04 LTS. Odpadá tak nutnost manuální kompilace a konfigurace systému spolu s nutnými závislostmi. Systém je také možné otestovat bez nutnosti instalace na disk. V takovém případě je však nutné pamatovat na to, že veškeré změny nebudou uloženy a zachycená data budou ztracena.

SLIS Live DVD obsahuje předinstalované všechny potřebné balíčky a knihovny nutné pro jeho provoz. Základem DVD je linuxová distribuce Ubuntu 12.04 LTS. Některé standardně dostupné balíčky (např. LibreOffice, Samba apod.), které v rámci účelu DVD nejspíše nebudou použity byly ze systému odstraněny. Systém však obsahuje funkční balíčkovací systém a potřebné balíčky je možné doinstalovat standardními nástroji dostupnými v Ubuntu.

7.1 Zavedení systému SLIS z live DVD

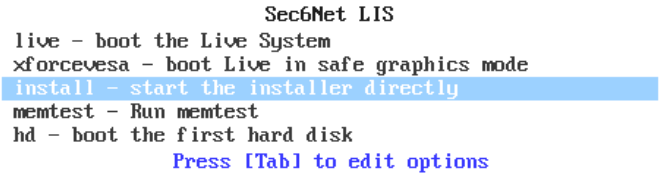
Pro zavedení systému SLIS z live DVD je na cílovém počítači potřeba nastavit zavedení z DVD mechaniky. Postup je závislý na cílové platformě. Typicky je třeba po spuštění počítače přejít do programu BIOS (obvykle stiskem klávesy `F2` či `DEL`). Zde je nutné v menu zavedení systému (typicky `boot`, či `boot options`) nastavit prioritu CD/DVD mechaniky oproti pevnému disku, či ostatním zařízením. Toto nastavení typicky bývá v sekci `Boot device priority`, či `Boot device order`. Poté uložíme nastavení BIOSu, vložíme SLIS Live DVD do DVD mechaniky a restartujeme počítač.

Při bootování ze SLIS live DVD nabízí systém uživateli dvě možnosti (viz obrázek 7.1):

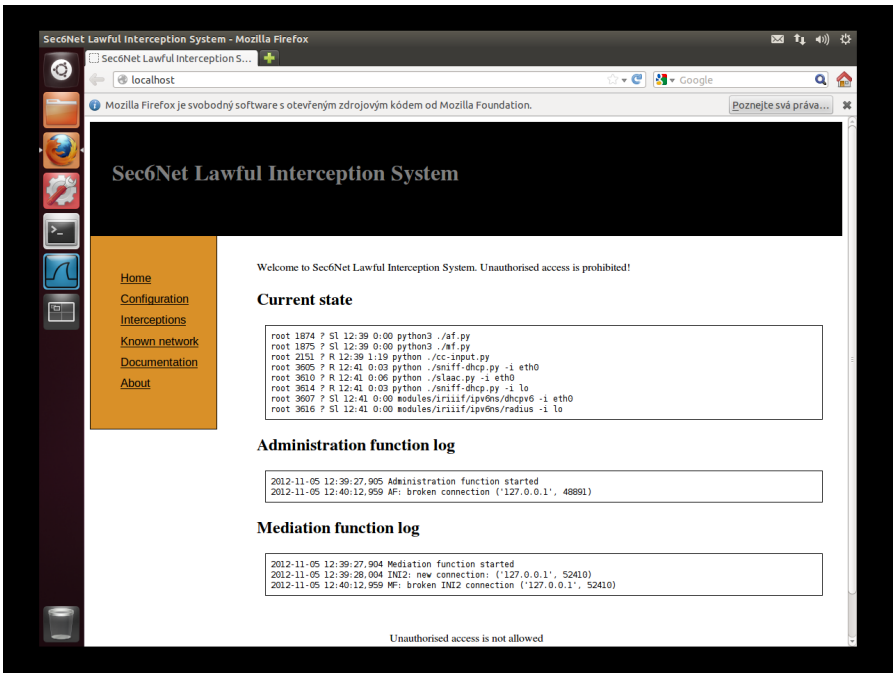
- Vyzkoušet Ubuntu 12.04 LTS se systémem SLIS bez nutnosti instalace.
- Nainstalovat Ubuntu 12.04 LTS se systémem SLIS na cílový počítač.

Pokud vybereme volbu `live - Boot the Live System` bude zaveden systém Ubuntu 12.04 s předinstalovaným systémem SLIS. Po zavedení systému se zobrazí okno prohlížeče `Mozilla Firefox` s webovým rozhraním systému SLIS. V této fázi je systém plně připraven k provozu (viz obrázek 7.2).

Na ploše se také nachází ikona Nainstalovat Sec6Net 12.04, která spustí instalátor systému. Pokud chceme instalátor spustit po startu počítače, vybereme v úvodním menu (obrázek 7.1) volbu `install - Start the installer directly`.



Obrázek 7.1: Úvodní nabídka SLIS Live DVD.

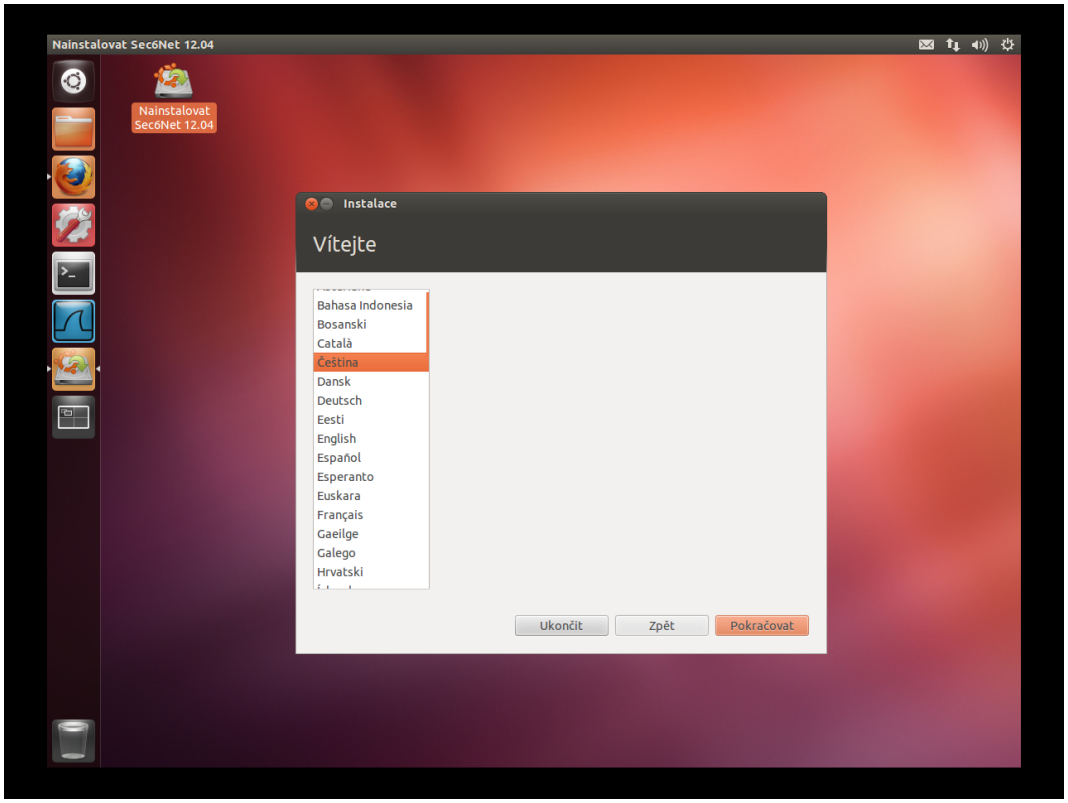


Obrázek 7.2: Úvodní stránka systému SLIS.

7.2 Instalace systému SLIS

Tuto volbu použijeme, chceme-li trvale nainstalovat systém SLIS na cílový počítač. Instalátor je možné spustit jak přímo z úvodního menu (obrázek 7.1), tak kdykoli po zavedeném systému. Je využit standardní instalátor systému Ubuntu (viz obrázek 7.3). Pro podrobnější postup doporučujeme využít existující návody pro instalaci distribuce Ubuntu jako je například tento: `http://wiki.ubuntu.cz/instalace/pr%C5%AFvodce_instalac%C3%AD`.

Pokud je dostupné připojení k Internetu, je možné při instalaci aktualizovat balíčky distribuce Ubuntu. Po Ukončení instalace budete vyzváni k restartování počítače. Poté můžete zavést systém Ubuntu 12.04 LTS se SLIS přímo z pevného disku počítače bez nutnosti přítomnosti SLIS live DVD v CD/DVD mechanice.



Obrázek 7.3: Instalace SLISu probíhá pomocí standardního instalátoru systému Ubuntu 12.04.

7.3 Součásti systému SLIS

Samotný systém SLIS se na DVD, či v nainstalovaném systému nachází v adresáři `/opt/slis`. Inicializační skript se nachází v `/etc/init.d/slis` a automaticky systém startuje po spuštění systému.

O zobrazení GUI se stará server Apache běžící na portu 80. K ovládání systému je možné použít webový prohlížeč (lokálně i vzdáleně). Pro vzdálený přístup je možné využít SSH server běžící na portu 5715. Z bezpečnostních důvodů je server po instalaci neaktivní. Pro aktivaci je potřeba vytvořit nového uživatele, nebo přiřadit stávajícímu heslo (např. pomocí programu `passwd`) a postupovat podle níže uvedeného návodu.

7.4 Ovládání SLIS

Po nastartování systému Ubuntu je spuštěno prohlížeč Firefox a jsou otevřeny stránky umožňující sledovat stav SLIS a přidávat, či odebírat odposlechy. Na úvodní stránce (zobrazené na obrázku 7.2) je možné zkontrolovat, které části systému skutečně běží.

Pro ukončení SLIS je možné využít inicializační skript:

```
# /etc/init.d/slis stop
```

Pro nové spuštění SLIS je možné využít inicializační skript:

```
# /etc/init.d/slis start
```

Před startem systému po jeho ukončení je vhodné chvíli počkat. Např. je možné využít příkaz:

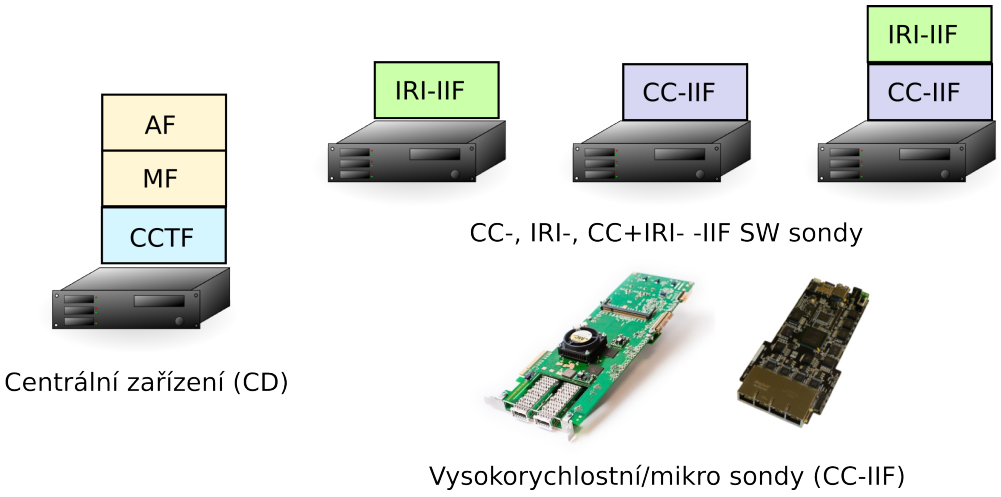
```
# /etc/init.d/slis stop; sleep 10; /etc/init.d/slis start
```

7.4.1 Konfigurace SLIS

Před prvním použitím systému je vhodné systém nakonfigurovat. V konfiguračním souboru `/opt/slis/slis.config`, ve kterém je možné povolit, které bloky systému se mají spouštět. Ve výchozím stavu se spouští centrální zařízení (AF, MF, CCTF) i sonda pro mapování sítě (IRI-IIF) a sonda pro odposlech dat (CC-IIF).

Popsaným způsobem se aktivují pouze softwarové sondy. Pro aktivací hardwarové sondy je potřeba použít konfigurační postup konkrétní sondy. V případě, že bude ke SLIS připojena hardwarová sonda, není nutné spouštět i softwarovou.

Obrázek 7.4 ukazuje příklady konfigurace počítačů a sond zapojených do zákonných odposlechů. Více informaci o architektuře systému SLIS a zapojení sond hledejte v části *Architektura vytvořeného systému pro zákonné odposlechy.*



Obrázek 7.4: Systém je možné nakonfigurovat, aby na konkrétním počítači běžely jen určité části systému. Případně je k systému možné připojit vysokorychlostní, či mikrosondu.

Dodatečná konfigurace jednotlivých bloků se nachází v konfiguračních souborech s příponou `.ini` v adresáři `/opt/slis/light`.

- Konfigurace pro software spouštěný na centrálním zařízení:

  af.ini konfiguruje porty pro vnitřní komunikaci s jádrem IRI-IIF a vkládají a mažou odposlechy rozhraním HI1. Dále specifikuje *Delivery Country Code* [24, 36].

  lea.ini obsahuje informace o oprávněných orgánech činných v trestním řízení a LEA (včetně LEAID), kteří mohou využívat danou instalaci SLIS.

mf.ini, cctf.ini konfiguruje porty pro komunikaci se sondami CC-IIF a pro vnitřní komunikaci s jádrem IRI-IIF. V `mf.ini` je možné nastavit adresář úložiště výstupních souborů pro rozhraní HI2 a HI3.

iri-core.ini konfiguruje porty pro vnitřní komunikaci s AF a MF&CCTF a pro síťovou komunikaci se sondami IRI-IIF. Dále specifikuje NIDy, které nejsou v dané síti unikátní a nesmí být použity pro odvozování identity pachatele (např. u NWO/AP/SvP, který u protokolu PPPoE poskytuje stejné přístupové jméno pro všechny zákazníky).

- Konfigurace pro software spouštěný na sondě IRI-IIF:

  iri-collector.ini řídí spouštění modulů IRI-IIF, na kterých rozhraních budou poslouchat a získávat informace. Dále tento soubor obsahuje konfiguraci portu pro komunikaci s centrálním zařízením, konkrétně jádrem IRI-IIF a jméno sondy (slouží k identifikaci zdroje metadat o komunikaci).

- Konfigurace pro software spouštěný na sondě CC-IIF:

  cc-iif.ini konfiguruje porty pro komunikaci s MF&CCTF a nastavuje ID sondy pro spárování konfigurace rozhraním CCCI a zachycených dat rozhraním INI3. V případě, že je použitá mikrosonda podporující pouze protokol UDP, je tento soubor použit i pro její konfiguraci. V takovém případě musí být nastavena volba `SLIS_UDP=True` v souboru `/opt/slis/slis.config`

7.4.2 Vzdáleny přístup ke SLIS a zabezpečení provozu mezi síťovými prvky SLIS

V rámci SLIS je dostupný server SSH, který však není aktivní. Pro jeho zprovoznění je potřeba postupvat následujícím způsobem:

1. Zkontrolujte konfiguraci v souboru `/etc/ssh/sshd_config`, povolte jen takové volby, které potřebujete pro svou práci.
2. Spusťte `# dpkg-reconfigure openssh-server`

Ve výchozím stavu nyní běží SSH server na portu 5715. Zdůvodnění využití protokolu SSH naleznete ve starší technické zprávě *Spolehlivá a zabezpečená komunikace v rámci systému pro zákonné odposlechy* [40] Pokud chcete, aby sondy komunikovaly s centrálním zařízením šifrovaně, řiďte se následujícím postupem:

1. Pro zvýšení bezpečnosti přidejte do SLIS s centrálním zařízením nového uživatele, např. příkazem `# useradd -m -s /bin/false` *`secureprobe`*. Tento uživatel se nebude moct do systému přihlásit k terminálu, ale bude moct vytvořit tunel pro přenos dat pomocí SSH.
2. Na počítači se sondou vytvořte soukromou a veřejnou část klíče který bude využit pro přístup ke SLIS, např. příkazem `ssh-keygen -t rsa -b 4096`.
3. Na počítači s centrální části SLIS vytvořte adresář `/home/`*`secureprobe`*`/.ssh` a do souboru `/home/`*`secureprobe`*`/.ssh/authorized\_keys` přidejte veřejnou část klíče (zpravidla obsah souboru `.ssh/id\_rsa.pub` na počítači se sondou).
4. Na sondě spusťte příkaz:

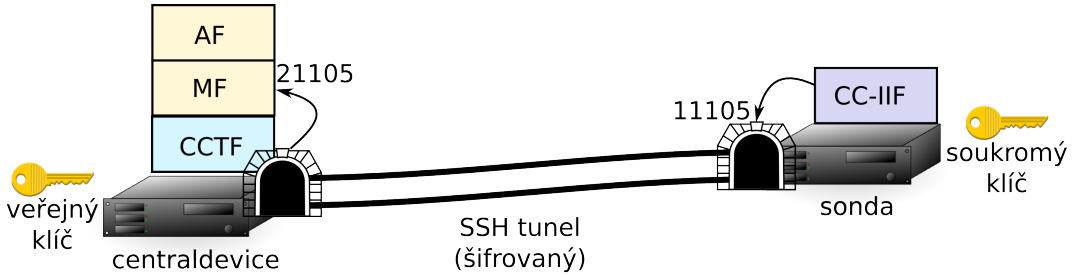
```
ssh <centraldevice> -l secureprobe -p 5715 -Nf \
   -L <vstupni port tunelu>:localhost:<tcp port rozhrani>
```

Např. `ssh centraldevice -l secureprobe -p 5715 -Nf -L 11105:localhost:21105`. Pro trvale umístěnou sondu nezapomeňte zajistit vytvoření tunelu po restartu systému (např. úpravou startovacího skriptu `/etc/init.d/slis`).

5. Na sondě nakonfigurujte přístup k centrálnímu zařízení s parametry:

```
server= localhost,  port=< vstupní port tunelu>
```

Obrázek 7.5 ukazuje tunel vytvořený ve výše uvedeném předchozím příkladě.



Obrázek 7.5: Zabezpečený přenos dat pomocí SSH [40].

7.4.3 Správa odposlechů

Přes webové rozhraní je možné vkládat nové odposlechy, či odstraňovat probíhající odposlechy. Odposlech dat je možný na základě některého z podporovaných *NID*. Podporovány jsou protokoly a metody popsané v části *Dynamická identifikace uživatelů*.

Zachycená data jsou ukládána do nakonfigurovaného adresáře v následující struktuře adresářů a souborů:

```
základní adresář/

   agentura1/

      liidx.hi2
      liidy.hi2
      liidx_1.pcap
      liidx_2.pcap
      liidy_1.pcap

   agentura2/
```

```
        liida.hi2
        liida_7.pcap
        liida_8.pcap
        ...

    ...
```

Základní adresář lze zadat pomocí relativní nebo absolutní cesty v souboru `mf.ini`. Implicitně jde o adresář s názvem `storage` v adresáři, kde je nainstalován SLIS. Názvy textových souborů pro rozhraní HI2 jsou ve formátu `LIID.hi2`. Názvy PCAP souborů pro HI3 jsou ve formátu `LIID_CIN.pcap`.

Kapitola 8

Rozhraní vytvořeného systému pro zákonné odposlechy

Tato kapitola se zabývá popisem rozhraní vytvořeného systému SLIS. Jak rozhraní vnitřními, tak rozhraními pro komunikaci se sondami. Tato kapitola je určená programátorům, kteří by chtěli vytvořený systém dále rozšířit. Pro běžné použití systému nejsou zde obsažené informace podstatné.

Názvy rozhraní vycházejí z doporučení ETSI popisované v kapitole 2. Protože vytvořený systém SLIS referenční architekturu modifikuje, jak bylo popsáno v kapitole 3, nejsou některá rozhraní přítomna.

Není-li v popisu rozhraní určeno jinak, jsou rozhraní realizována jako textová, kde každá zpráva je přenášena na samostatném řádku.

8.1 Vnější rozhraní systému

8.1.1 Rozhraní HI1

Vytvářený systém pro LI by podporuje pouze manuální ověření [23] pověření k odposlechu. Na pověřeném zaměstnanci je případná transformace předaného identifikátoru odposlechu na $\mathrm{NID_{In}}$). Odposlechy jsou přidávány pomocí programu *hi1.py*: buď přímo z příkazové řádky, nebo pomocí webového rozhraní. Komunikace s AF probíhá pomocí požadavků a odpovědí tak, jak je ukázáno na obr. 8.1. Pověřený zaměstnanec zadá požadavek na přidání, či odebrání odposlechu. AF odpoví, zda se akce podařila, nebo ne. Zpráva z odpovědi se zobrazuje v příkazové řádce i ve webovém rozhraní. Vnitřními rozhraními systému se navíc v pravidelných intervalech přenášejí zprávy obsahující řetězec `'HEARTBEAT'` pro detekci živosti spoje.

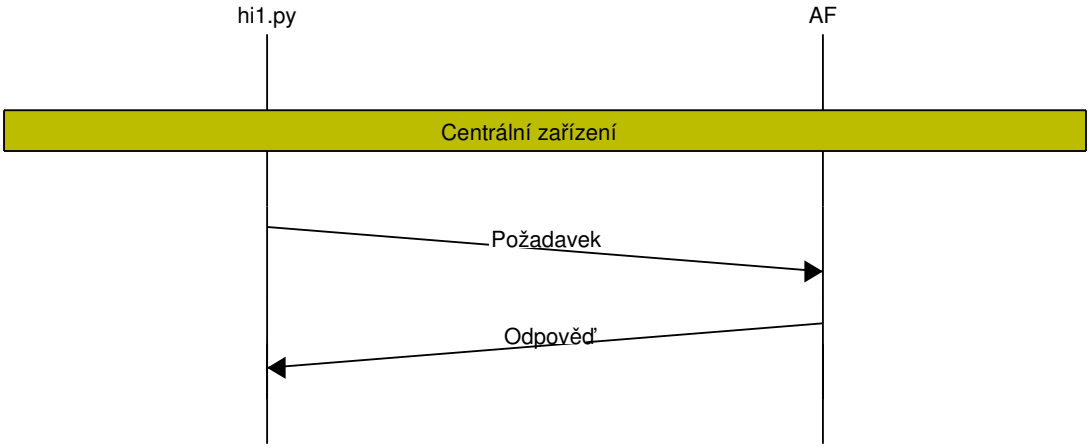
Přidání odposlechu

Formát zprávy

`('new_intercept', HI1Intercept(`*LIID*`,` *NID*`,` *start odposlechu*`,` *konec odposlechu*`,` *zasílání CC*`))`

Například

```
('new_intercept', HI1Intercept('LIID2', '192.168.1.1', '1303470480',
'1335045600', 'CC'))
```



Obrázek 8.1: Komunikace při vkládání a odebírání odposlechů ze systému.

```
('new_intercept', HI1Intercept('LIID2', '192.168.1.1', '1303470480',
'1335045600', ))
```

Odebrání odposlechu

AF okamžitě odebírá odposlechy označené předaným LIID a informuje zbytek systému, pokud je odposlech aktivní.

Formát zprávy

('delete_intercept', *LIID*)

Například

```
(delete_intercept, 'cz1')
```

Odpovědi

Formát zprávy

('ack'/*řetězec popisující chybu*, *popis odposlechu*)

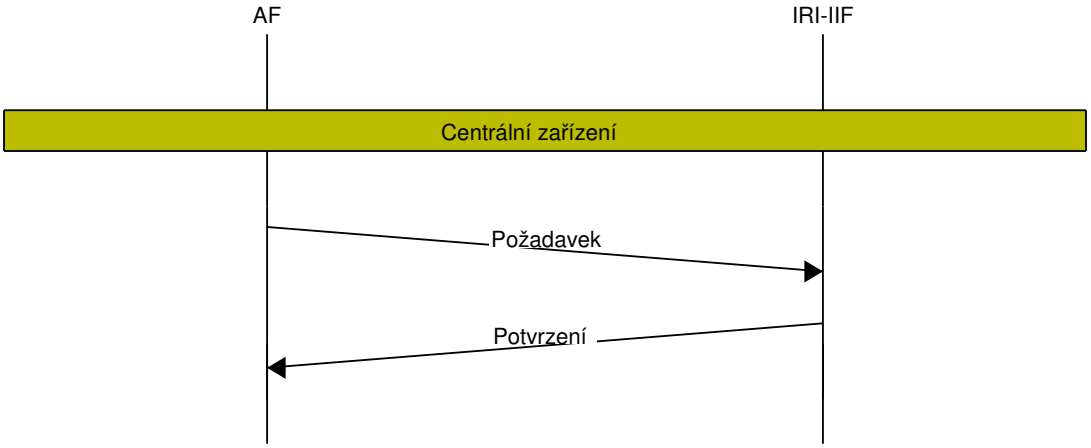
Například

```
("Cannot add interception from the past", HI1Intercept('LIID2',
'192.168.1.1', '1303470480', '1335045600', 'CC'))
("Corrupted interception - duplicate LIID", HI1Intercept('LIID2',
'192.168.1.1', '1303470480', '1335045600', 'CC'))
("ack", HI1Intercept('LIID2', '192.168.1.1', '1303470480',
'1335045600', 'CC'))
```

8.2 Správa odposlechů

8.2.1 Rozhraní INI1a

Komunikace je iniciovaná ze strany AF. V rámci komunikace AF zasílá požadavky a očekává jejich potvrzení ze strany jádra IRI-IIF, jak ukazuje obr. 8.2.



Obrázek 8.2: Komunikace na rozhraní INI1a.

Přidání odposlechu

Formát zprávy

`('new_intercept', 'iri-iif', INI1AIntercept(`*CID*`, `*NID*`, `*LIID*`, `*Odchytávání CC*`))`

Například

```
('new_intercept', 'iri-iif', INI1AIntercept(CID('Operator ID',
NIDIPv4('10.10.10.1'), 'None', 'DCC'), '192.168.1.1', 'LIID', True))
```

Odebrání odposlechu

IRI-IIF okamžitě odebírá odposlechy označené předaným LIID

Formát zprávy

`('delete_intercept', 'iri-iif', `*LIID*`)`

Například

```
('delete_intercept', 'iri-iif', 'LIID')
```

Potvrzovací zprávy

Formát zprávy

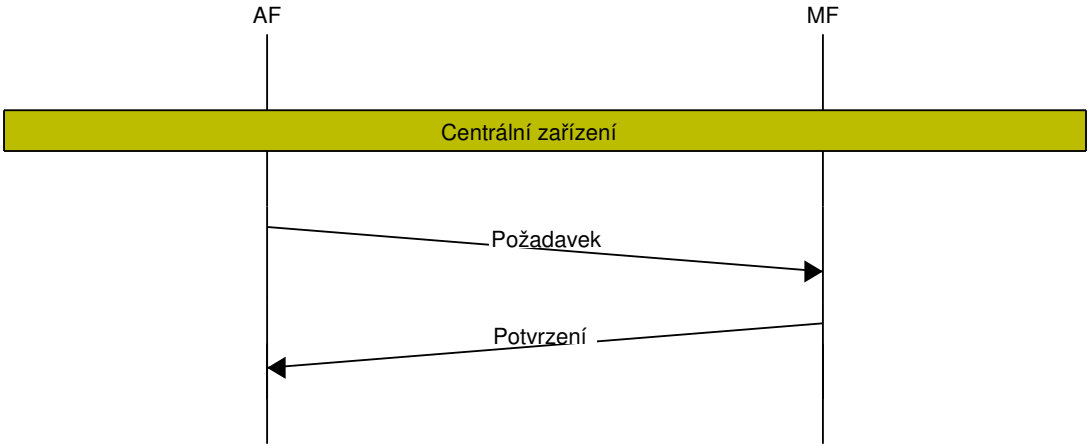
`('ack', `*Původní zpráva*`)`

Například

```
('ack', ('new_intercept', 'iri-iif', INI1AIntercept(CID('Operator
ID', '10.10.10.1', 'None', 'DCC'), '192.168.1.1', 'LIID',
(1302818400.0, 1304114400.0))))
```

8.2.2 Rozhraní INI1c

Komunikace iniciovaná ze strany AF. V rámci komunikace AF zasílá požadavky a očekává jejich potvrzení ze strany MF, jak ukazuje obr. 8.3.



Obrázek 8.3: Komunikace na rozhraní INI1c.

Přidání odposlechu

Formát zprávy

`('new_intercept', 'mf', INI1CIntercept(`*LIID*`,` *LEA*`, (`*start odposlechu*`,` *konec odposlechu*`),` *Boolean určující zasílání CC*`))`

Například

```
('new_intercept', 'mf', INI1CIntercept('LIID-ABC', 'LEA',
(1302818400.0, 1304114400.0), True))
```

Odebrání odposlechu

MF okamžitě odebírá odposlechy označené předaným LIID

Formát zprávy

`('delete_intercept', 'mf',` *LIID*`)`

Například

```
('delete_intercept', 'mf', 'LIID-ABC')
```

Potvrzovací zprávy

Formát zprávy

`('ack',` *původní zpráva*`)`

Například

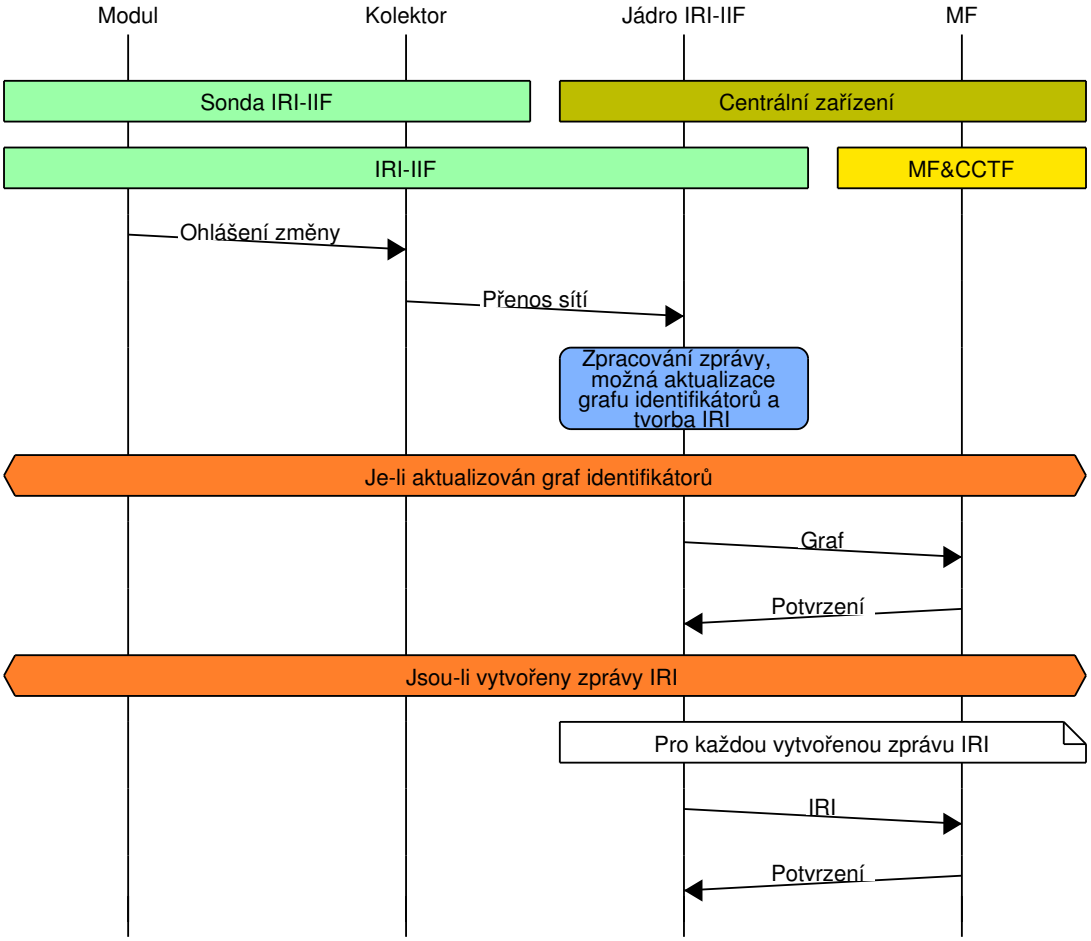
```
('ack', ('new_intercept', 'mf', INI1CIntercept('LIID-ABC', 'LEA',
(1302818400.0, 1304114400.0), True)))
```

8.3 Vytváření zpráv IRI

Jak bylo ukázáno na obr. 3.3 IRI-IIF je rozdělena na centrální jádro IRI-IIF a jednotlivé sondy IRI-IIF. Každá ze sond IRI-IIF může spouštět několik modulů zjišťujících identifikátory dostupné v dané lokaci sondy IRI-IIF.

Zprávy z každého modulu jsou do jádra IRI-IIF předávány skrze tzv. kolektory, běžící na každé ze sond, které zajišťují jednotnou konfiguraci modulů a případně šifrování přenosu.

Obrázek 8.4 ukazuje příklad tvorby zpráv v rámci IRI-IIF a jejich následovné předávání do MF jako zprávy IRI. Formát a účel zpráva je popsán ve zbytku této sekce věnující se tvorbě zpráv IRI.



Obrázek 8.4: Komunikace v rámci IRI-IIF a tvorba zpráv IRI.

8.3.1 Rozhraní modulů IRI-IIF

Z modulů IRI-IIF se do jádra IRI-IIF přenášejí informace nutné pro tvorbu grafu identifikátorů identifikátorů používaných v síti a zpráv IRI.

Ohlášení změny v síti

Změny jsou hlášeny pomocí zpráv typu *begin* (nalezena nová vazba mezi identifikátory, *continue* (vazba mezi identifikátory stále trvá), *end* (vazba mezi identifikátory není dále platná) a *report*

(pomocná informace neměnící dříve zaslané vazby).

Formát zprávy

(*Jméno modulu*, *Časová značka*, *Typ zprávy: BEGIN/CONTINUE/END/REPORT*, *Popis zprávy*, *Seznam NIDů, jejichž vazeb se zpráva týká*)

Například

```
('pppoe', 1302818400.0, 'BEGIN', 'Client made connection with
BRAS', [('PPP SESSION', '1234'), ('MAC', '00:0c:29:11:7c:14'), ('PPP
Login', 'user'), ('IP', '192.168.0.1')])
```

Za seznamem NIDů, jejichž vazeb se zprává týká mohou volitelně následovat další 2 seznamy. V takovém případě je rozdělení NIDů do seznamů následující:

- 1. seznam: NIDy určené pro umístění do grafu spravovaného jádrem IRI-IIF. Tyto NIDy jsou při odeslání zprávy typu *end* z grafu jádra IRI-IIF odstraněny.
- 2. seznam: NIDy, které nejsou určeny pro umístění do grafu jádra IRI-IIF, ale pouze do případných zpráv IRI.
- 3. seznam: NIDy určené pro umístění do grafu spravovaného jádrem IRI-IIF, jejichž vazba by měla být uchovávána i po odeslání zprávy *end* a neměla by být tedy z grafu jádra IRI-IIF odstraněny.

8.3.2 Rozhraní INI2

INI2 přenáší informace z IRI-IIF do MF. Jde jednak o samotné zprávy IRI (založené na zprávách z modulů IRI-IIF nebo grafu identifikátorů používaných v síti). Dále se při změně grafu identifikátorů používaných v síti přenáší celý graf ve formě tabulky jednotlivých hran.

Zprávy IRI

Formát zprávy

("iri_report", 'mf', (*Typ zprávy: BEGIN/CONTINUE/END/REPORT*, *LIID*, *CID*, NID_{CC}, (*Jméno modulu, který vytvořil IRI zprávu*, *timestamp*, *Typ zprávy vytvořené modulem, obvykle stejný jako typ zprávy*, *Popis zprávy*, *Samotná zpráva z modulu (struktura popsaná výše)*)), *Výsledek funkce* capture *(definované v části 5.2.4)*)

Například

```
('iri_report', 'mf', ('BEGIN', 'pokus', CID('VUTBR', '0.0.0.0/0',
'0', 'CZ'), '192.168.1.64', ('in progress', 1368037519.679987,
'BEGIN', 'Intercept activated', [('IPv4', '0.0.0.0/0')]),
('00:0c:29:6e:7a:fe', '00:0c:29:67:34:92', '00:0c:29:98:d4:df',
'192.168.1.68', '192.168.15.5', '192.168.1.64')))
```

Přenos grafu identifikátorů používaných v síti

Pro účely monitorování sítě, či ladění systému je vhodné zkoumat detekované identifikátory a vyzby z jednotlivých modulů IRI-IIF. Proto jsme do výstupních zpráv IRI-IIF přidali i zasílání samotného grafu po každé jeho změně.

Formát zprávy

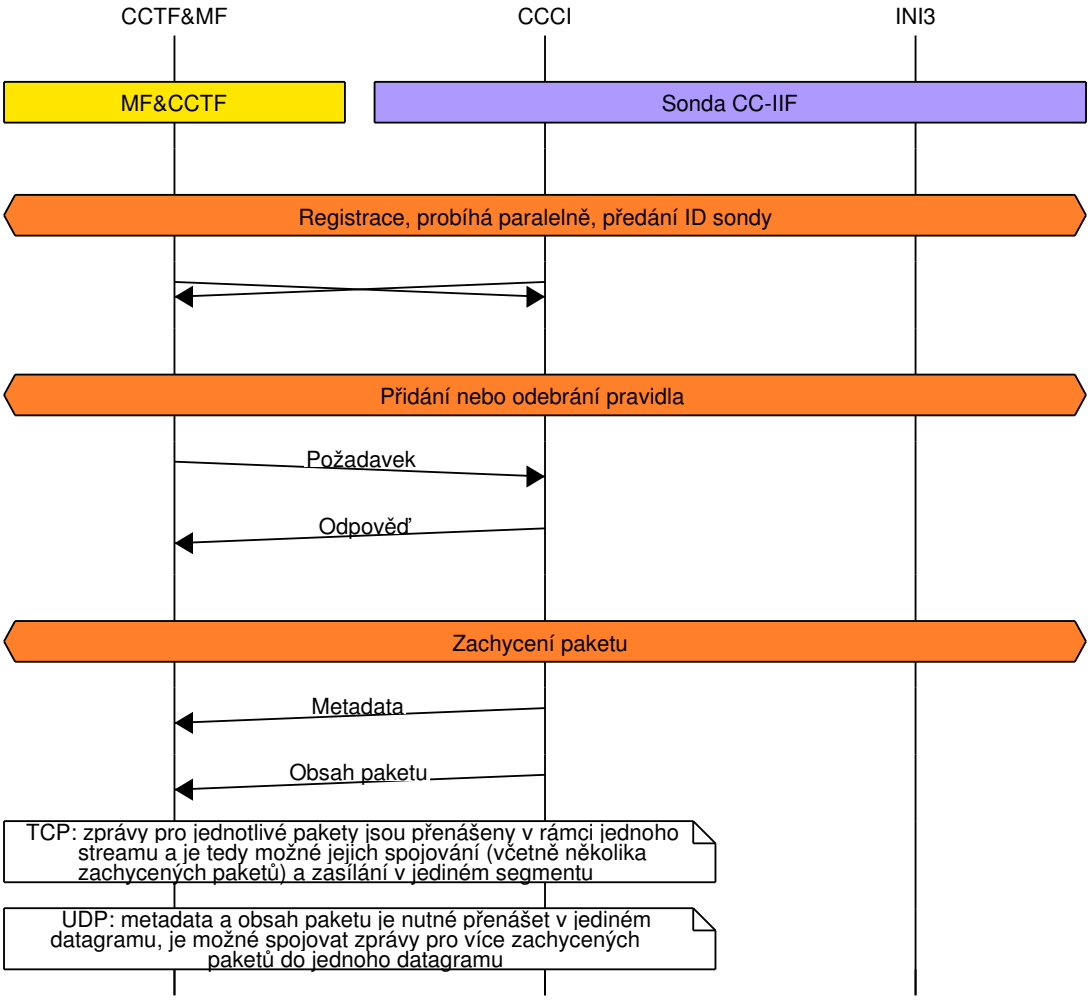
("iri_table", "mf", *Textový řetězec obsahující graf. Každý řádek popisuje jednu vazbu mezi 2 identifikátory a protokolem, ze kterého byla vazba naučena*)

Například

```
("iri_table", "mf", "dhcp\t00:0c:29:98:d4:df\t192.168.1.64\n")
```

8.4 Rozhraní sond CC-IIF

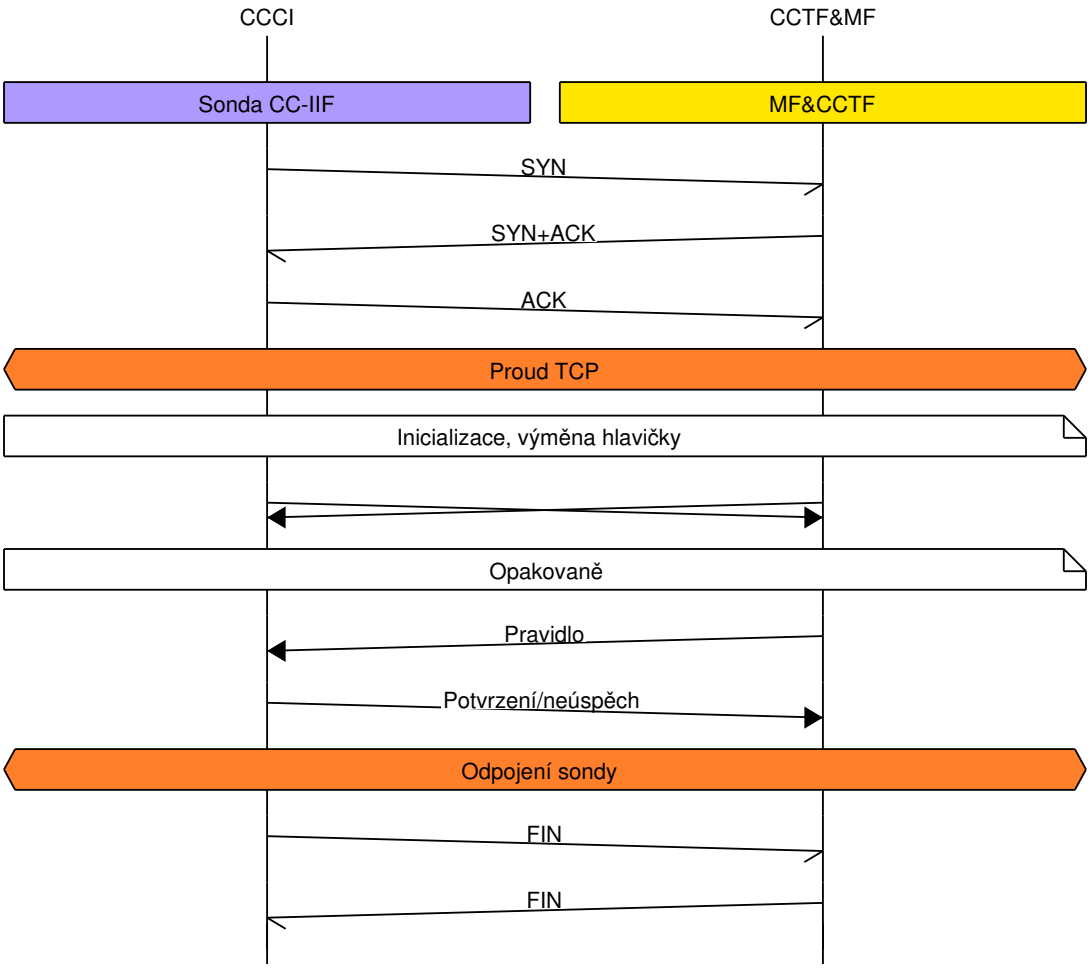
Sond CC-IIF může být k centrálnímu zařízení libovolné množství. Protože má každá z nich k centrálnímu zařízení otevřeny dva proudy TCP, je nutná identifikace sondy pomocí 32-bitového čísla. Obrázek 8.5 znázorňuje komunikaci mezi sondou CC-IIF a centrálním zařízením (konkrétně MF&CCTF). Každá sonda se nejdříve přihlásí na rozhraní CCCI a po navázání komunikace se naváže komunikace rozhraním INI3. Formát zpráv a bližší popis rozhraní je obsahem zbytku sekce.



Obrázek 8.5: Komunikace mezi sondou CC-IIF a centrálním zařízením (MF&CCTF).

8.4.1 Rozhraní CCCI

Rozhraní CCCI je realizované pomocí binárního protokolu nad TCP. CCTF&MF naslouchá na nakonfigurované IP adrese a portu TCP. Obr. 8.6 ukazuje pořadí zpráv předávaných v rámci proudu TCP, který slouží jako komunikační kanál pro rozhraní CCCI.

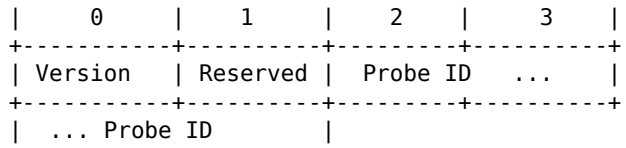


Obrázek 8.6: Proud TCP pro rozhraní CCCI.

Všechny více-bajtové položky budou v rámci TCP streamu posílané v bajtovém pořadí little endian. Předpokládá se trvale CCCI spojení. V případě přerušení a obnovení spojení je MF povinna znovu nakonfigurovat všechny odposlechy, aby se předešlo možné nekonzistenci v případě změny pravidel v době, kdy bylo spojení přerušeno.

Hlavička CCCI

Hlavička se posílá právě jednou na začátku TCP streamu jak ve směru od sondy, tak v opačném směru. Hlavička slouží k předání verze protokolu CCCI (budoucí kompatibilita) a k předání čísla sondy. Hlavička je definovaná na obr. 8.7 a jednotlivé položky popsány níže.



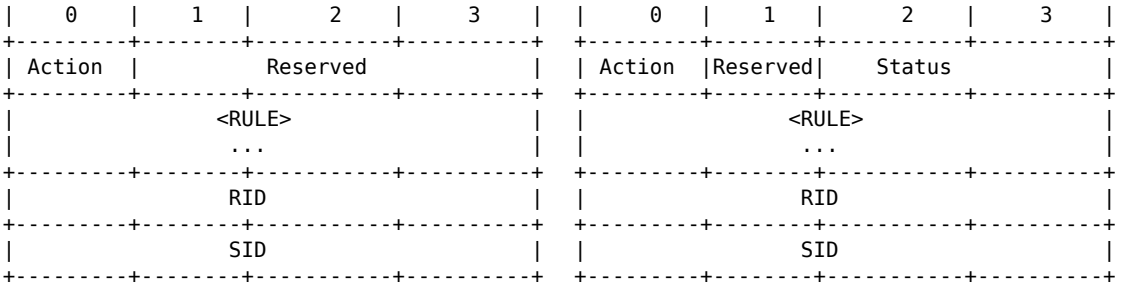
Obrázek 8.7: Formát hlavičky CCCI.

Version (1B) definuje formát protokolu. V současné době předpokládáme v obou směrech verzi 1. Jiná verze protokolu nebyla definována a není podporována systémem SLIS.

Probe ID (4B) jedinečně označuje sondu v rámci instance SLIS. Předaná hodnota musí být shodná na rozhraní CCCI a INI3, aby mohla být komunikace v rámci MF&CCTF spárována. Z MF&CCTF se v rámci ProbeID posílá 0. Sonda CC-IIF si musí svůj identifikátor zjistit z konfigurace.

Konfigurace CCCI

Po výměně hlaviček už může nastat komunikace, ve které jsou jednotlivá pravidla obalována do zpráv typu požadavek (z MF&CCTF do sondy) a odpověď (ze sondy do MF&CCTF). Strukturu těchto zpráv ukazuje obr. 8.8.



Obrázek 8.8: Formát požadavku a odpovědi v rámci CCCI.

Action udává, zda se jedná o přidání, nebo odebrání pravidla:

- 0 = přidání
- 1 = odebrání

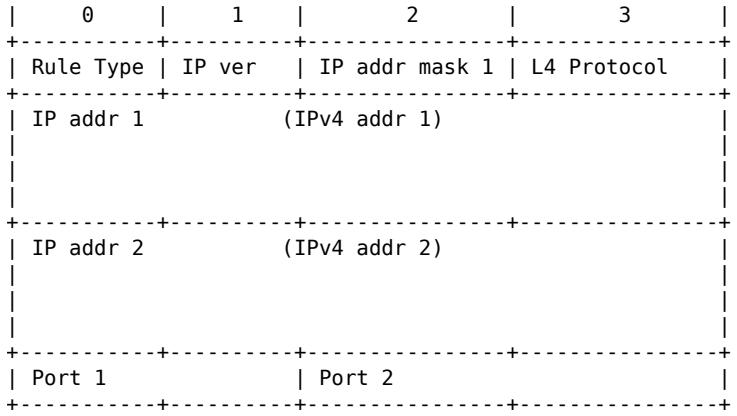
Status (2B) udává návratový kód provedené operace, záporná hodnota a nula značí neúspěch. Zasílá se jen jako součást odpovědi.

RULE (40B) je struktura pro popis pravidla definovaná níže.

SID (4B) Session ID číslo pravidla přiřazeného MF danému požadavku/pravidlu

RID (4B) Číslo pravidla přiřazeného sondou k zaslanému požadavku/pravidlu. V rámci požadavku je hodnota nastavena na shodnou se SID. Je povinností sondy zajistit, aby dané RID bylo unikátní v rámci pravidel nakonfigurovaných v daný okamžik a v průběhu odposlechu se nezměnilo.

Struktura *RULE* popisuje jednotlivá pravidla. Podle typu pravidla mohou být významné pouze některé položky. Pokud některá položka nemá pro daný typ pravidla význam, očekává se, že její hodnota bude 0. Příjemce by však neměl hodnotu těchto polí brát v potaz. Formát struktury je uveden na obr. 8.9.



Obrázek 8.9: Struktura pro popis pravidla používaná v rámci CCCI.

Rule Type (1B), typ pravidla

- 0 - Konkrétní IP adresa, nebo rozsah IP adres specifikovaný adresou sítě a maskou.
- 1 - Odposlech konkrétní komunikace (spojení), pětice TCP/UDP.
- 2 - Přístup k pevně zadanému portu na všech adresách (není podporováno SLIS, ale může být podporováno konkrétní sondou).
- 3 - Přístup k pevně zadanému portu na konkrétní IP adrese, trojice TCP/UDP.

IP ver (1B), verze IP protokolu (položka platná pro pravidla typu 0, 1, 3)

- 4 - IPv4 protokol
- 6 - IPv6 protokol

IP addr mask 1 (1B), maska IP adresy 1 v rozsahu 0 - 128 (položka platná pro pravidla typu 0, pro pravidla typu 1 a 3 musí být nastavená na 32 pro IPv4 a 128 pro IPv6)

IP addr 1 (16B), IPv4 nebo IPv6 adresa dle položky *IP ver* (položka platná pro pravidla typu 0, 1, 3). V případě IPv4 je adresa přenášena na pozici 8-11.

IP addr 2 (16B), IPv4 nebo IPv6 adresa dle položky *IP ver* (položka platná pro pravidla typu 1). V případě IPv4 je adresa přenášena na pozici 24-39 .

Proto (1B), číslo transportního protokolu (položka platná pro pravidla typu 1, 2, 3).

Port 1 (2B), port trasportního protokolu (položka platná pro pravidla typu 1, 2, 3).

Port 2 (2B), port trasportního protokolu (položka platná pro pravidla typu 1).

Při filtrování se uvažuje vždy obousměrná komunikace. Tzn. buď nastane situace, kdy se čtveřice (IP addr 1, IP addr 2, Port 1, Port 2) porovnává se (zdrojovou, cílovou IP adresou, zdrojovým, cílovým portem), nebo se (IP addr 1, IP addr 2, Port 1, Port 2) porovnává s (cílovou, zdrojovou IP adresou, cílovým , zdrojovým portem).

8.4.2 Rozhraní INI3

Rozhraní INI3 je realizované pomocí binárního protokolu nad TCP, či UDP. CCTF&MF naslouchá na nakonfigurované IP adrese a portu. Sonda se připojuje k centrálnímu zařízení a identifikuje se stejným identifikátorem, který použila pro rozhraní CCCI. V případě protokolu UDP se hlavička nezasílá a zasílají se až vlastní odchycená data. V takovém případě se na centrálním zařízení rozlišují data INI3 pomocí portu UDP a správné číslo sondy je potřeba nakonfigurovat v konverzním programu.

Obr. 8.10 ukazuje pořadí zpráv předávaných v rámci proudu TCP, pokud je použit jako komunikační kanál pro rozhraní CCCI.

V případě využití TCP, jsou zprávy pro jednotlivé pakety přenášeny v rámci jednoho streamu a je tedy možné jejich spojování (včetně několika zachycených paketů) a zasílání v jediném segmentu. V případě UDP, je nutné metadata a obsah paketu přenášet v jediném datagramu. Zprávy tedy není možné dělit ani na rozmezí metadat a vlastních dat ani v rámci jednoho z bloků. Je však možné spojovat zprávy pro více zachycených paketů do jednoho datagramu.

Stejně jako v případě CCCI jsou všechny více-bajtové položky přenášeny jako little endian.

Hlavička INI3

Hlavičku INI3 posílá sonda právě jednou na začátku TCP streamu. Hlavička slouží k předání verze protokolu INI3 (budoucí kompatibilita) a k předání čísla sondy. Hlavička je definovaná na obr. 8.11 a jednotlivé položky popsány níže.

Version (1B) definuje formát protokolu. V současné době předpokládáme v obou směrech verzi 1. Jiná verze protokolu nebyla definována a není podporována systémem SLIS.

Probe ID (4B) jedinečně označuje sondu v rámci instance SLIS. Předaná hodnota musí být shodná na rozhraní CCCI a INI3, aby mohla být komunikace v rámci MF&CCTF spárována.

Odposlechnutá data

Jak bylo naznačeno výše, ke každému síťovému paketu (typicky rámec Ethernetu, dále označován jako blob) jsou sondou přidávána metadata popisující daný paket. Kontrolní součet (např. Ethernetové FCS) není součásti blobu, předpokládá se odposlouchávání pouze platných PDU.

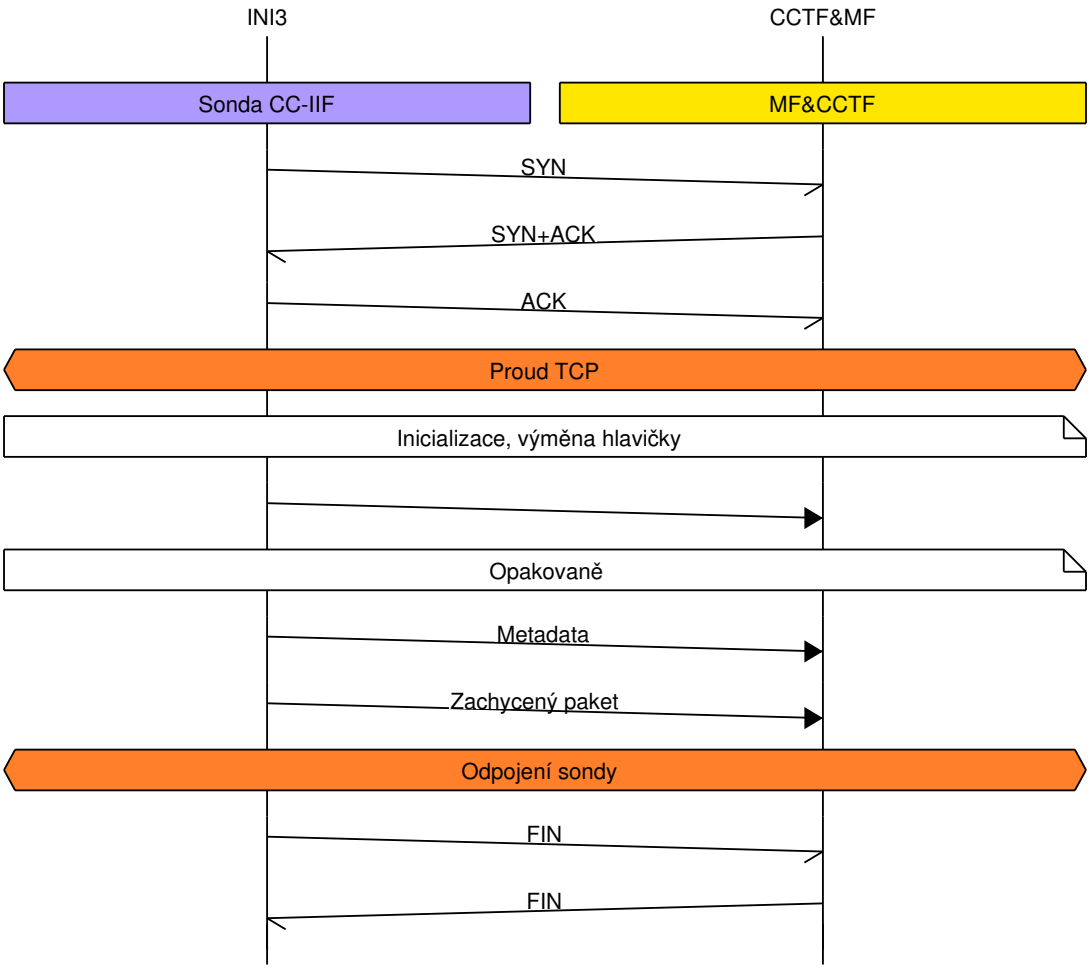
Formát metadat je zachycen na obr. 8.12.

Blob Size (2B) určuje délku následného *blobu* v bajtech .

Interface (1B) obsahuje číslo rozhraní sondy, na kterém byl blob původně přijat.

Timestamp (4B a 4B) je časová značka zachycení paketu ze sítě na vstupním rozhraní sondy. Značka je rozdělená do dvou 4-bajtových částí:

- 0: Počet sekund od začátku Unixové éry



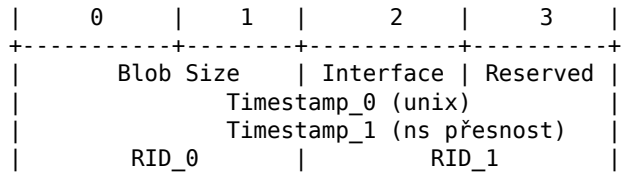
Obrázek 8.10: Proud TCP pro rozhraní INI3.

```
|     0     |    1     |    2    |     3    |
+-----------+----------+---------+----------+
| Version   | Reserved |  Probe ID   ...    |
+-----------+----------+---------+----------+
|  ... Probe ID        |
```

Obrázek 8.11: Formát hlavičky INI3.

- 1: Počet nanosekund

RID Reason ID přiřazený sondou k nejvíce specifickému pravidlu. RID_0 identifikuje zdroj, RID_1 cíl. Pokud je RID_0 anebo RID_1 neplatný (vždy je platný alespoň jeden, jinak je rámec na sondě zahozen), potom platí RID_0 = RID_1. Dvojice identifikátorů RID a jejich platnost je potřeba řešit z důvodu možné vzájemné komunikace dvou (potenciálně nesouvisejících) podezřelých osob. Typicky však je odposloucháván pouze zdroj, nebo cíl komunikace a oba identifikátory RID jsou si rovny.



Obrázek 8.12: Formát metadat posílaných přes INI3 pro každý zachycený blob.

Po bloku metadat následuje blob se samotným obsahem zachyceného rámce.

Kapitola 9

Závěr

Cílem této technické zprávy bylo zdokumentovat hlavní výsledek činnosti skupiny pro zákonné odposlechy působící v rámci projektu *Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace*. Skupina vytvořila prototyp systému pro zákonné odposlechy (LIS) pojmenovaný SLIS. Cílem vytvořeného LIS není nahrazení komerčních nástrojů, ale tento prototyp slouží jako základní prostředí pro: a) vývoj nových technik dynamické identifikace uživatele v prostředí IPv6 sítí, b) sběr dat a vývoj nových metod v oblasti rekonstrukce a vizualizace zachyceného provozu a c) jako základní testovací prostředí pro vývoj mikro-sondy a vysokorychlostní sondy včetně platformy pro zpracování provozu na základě aplikačních identifikátorů.

Zákonnými odposlechy se zabývá celá řada standardů. Technická řešení komerčních firem často implementuji vlastní varianty. Architektura vytvořeného SLIS je založena na standardech ETSI a je rovněž inspirována architekturou firmy Cisco publikovanou v RFC 3924 [6]. S ohledem na použití navrhovaného systému nebylo nutné implementovat všechna doporučení dle norem ETSI a rozhraní HI1, HI2 a HI3 byla proto v rámci prototypu zjednodušena. Kapitoly 2 a 3 se zabývají diskuzemi nad architekturou systému.

V průběhu návrhu a implementace architektury bylo nezbytné vyřešit několik důležitých problémů mezi které patří např. způsob předávání zpráv mezi jednotlivými bloky systému tak, aby nedocházelo k duplicitním přenosům odposlouchávaných dat. Metoda založená na mapování požadavků na jednoznačné identifikátory typu SID je podrobněji popsána v kapitole 4.

Zvláštní pozornost byla věnována problematice dynamické identifikace odposlouchávaného cíle v prostředí IPv4 a IPv6 sítí a také pro aplikační protokoly. Architektura bloku IRI-IIF, který tuto funkci v LI systému plní, byla navržena modulárně s ohledem na podporu různých protokolů používaných v síťovém prostředí. Kapitola 5 se zabývá základními principy implementovaného bloku pro zjišťování dynamické identity uživatele. Na tuto kapitolu navazuje kapitola 6 s popisem jednotlivých podporovaných protokolů a s postupem zjišťování identity.

Pro koncové uživatele systému je určena kapitola 7, která se zabývá popisem ovládání vytvořeného SLIS, jeho instalací, konfigurací a provozováním. Systém se ovládá pomocí webového rozhraní.

Pro možné budoucí implementátory rozšíření systému, či využití implementovaných částí vznikla kapitola 8 popisující strukturu jednotlivých zpráv předávaných v rámci systému. Tato kapitola je určena programátorům.

Kromě popisovaného SLIS vznikla v rámci činnosti skupiny řada dalších výsledků, některé byly použity v samotném systému a jiné tvořili samostatnou větev činností. Jedná se zejména o tyto výsledky:

- *Systém pro dynamickou správu identity* (*Sec6Net Identity Management System* – SIMS)

(2014) vychází z implementace IRI-IIF v rámci SLIS a zpřístupňuje graf identifikátorů. Nástroj je určen pro obecné využití a není omezen pro oblast zákonných odposlechů (LI). Při jeho nasazení je však vždy nutné dbát dodržení platné legislativy a nepřekračovat získáváni dat nad rámec zákona.

- *ndtrack* [32] je samostatný modul IRI-IIF určený pro analýzu zpráv *Neighbor Discovery*. Nástroj je možné uplatnit i pro monitoring a administraci sítě. Funkcionalita nástroje byla prezentována [58] v rámci konference *4th International Conference on Data Communication Networking*. Rozšířená verze článku tohoto článku vyšla [59] na pozvání organizátorů konference v knize *E-Business and Telecommunications*.

- *pcf* [63] je samostatný program pro detekci odchylky časových značek. Tato metoda byla zkoumána jako experimentální zdroj dat pro SLIS, jak je popsáno v sekci 6.11. Průběžné výsledky byly prezentovány na vědecké konferenci *11th International Conference on Security and Cryptography* [57] a ve vědeckém časopise *IEEE Transactions on Dependable and Secure Computing* [64]. Další článek je momentálně v posuzování.

- *immc* [10] je poslední z nástrojů, který se stal modulem IRI-IIF. Tento nástroj se zaměřuje na analýzu protokolů pro komunikaci v reálném čase.

- Několik výsledků projektu souvisí s možnosti oklamání a obelstění systémů pro zákonné odposlechy (jak komerčních, tak SLIS). Metody pro oklamání byly popsány v technické zprávě [60]. Navazující technická zpráva [11] se pak podrobněji věnuje technickým principům anonymizační sítě Tor. Nástroj LDP [60, 62] vznikl jako demonstrátor možného útoku na systém pro zákonné odposlechy. Nástroj LNC [60, 62] se takovou třídu útoku snaží detekovat.

- Spolehlivé a zabezpečené komunikaci v rámci LIS jsme se věnovali samostatné technické zprávě [40]. Její výsledky jsme aplikovali využitím protokolu SSH, tak jak je doporučeno v části 7.4.2.

Samotný SLIS, způsob nasazení a metody pro korelaci identit zveřejněné v rámci SIMS byly publikovány na konferenci *6th Inetrnational Conference on Digital Forensics & Cyber Crime* [61].

Kapitola 10

Literatura

[1] TCPDUMP/LIBPCAP public repository. [[online]], citováno 2011-10-24.
URL `http://www.tcpdump.org/`

[2] COUNCIL RESOLUTION of 17 January 1995 on the lawful interception of telecommunications (96/C 329/01). 1996.

[3] Aboba, B.; Aboba, B.; Arkko, J.; aj.: *The Network Access Identifier*. IETF, 2005, RFC 4282 (Proposed Standard).

[4] AQSACOM: Lawful Interception for IP Network. 2012, White Paper.

[5] ATIS/TIA: *Lawfully Authorized Electronic Surveillance. J-STD-025-B*. Alliance for Telecommunications Industry Solutions/Telecommunications Industry Association Joint Standard, 2006.

[6] Baker, F.; Foster, B.; Sharp, C.: *Cisco Architecture for Lawful Intercept in IP Networks*. IETF, 2004, RFC 3924 (Informational).

[7] Bound, J.; Volz, B.; Lemon, T.; aj.: *Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*. IETF, 2003, RFC3315 (Proposed Standard).

[8] Bárta, S.: Tvorba metadat při odposlechu komunikace v realném čase. 2012, bakalářská práce, Vysoké učení technické v Brně.

[9] Bárta, S.: Rychlé zpracování aplikačních protokolů. 2014, diplomová práce, Vysoké učení technické v Brně.

[10] Bárta Stanislav, P. L.: IMMC - Tvorba metadat pro odposlech komunikace v reálném čase. 2012, `http://www.fit.vutbr.cz/~ibartas/prods.php?id=257&notitle=1`.

[11] Coufal, Z.; Polčák, L.: Anonymizační síť Tor. Technical Report FIT-TR-2014-02, Faculty of Information Technology, Brno University of Technology, 2014.

[12] Crispin, M. R.: *Internet Message Access Protocol - Version 4rev1*. IETF, 2003, RFC 3501 (Proposed Standard).

[13] Cronin, E.; Sherr, M.; Blaze, M.: On the (un)reliability of eavesdropping. *International Journal of Secure Networking*, ročník 3, 2008: s. 103–113, ISSN 1747-8405.

[14] Deering, S.; Hinden, R.: *Internet Protocol, Version 6 (IPv6) Specification*. IETF, 1998, RFC 2460 (Draft Standard).

[15] Droms, R.: *Dynamic Host Configuration Protocol*. IETF, 1997, RFC 2131 (Draft Standard).

[16] ETSI: *ETSI TR 101 943: Telecommunications security; Lawful Interception (LI); Concepts of Interception in a generic Network Architecture*. European Telecommunications Standards Institute, 2001, version 1.1.1.

[17] ETSI: *ETSI TR 101 944: Telecommunications security; Lawful Interception (LI); Issues on IP Interception*. European Telecommunications Standards Institute, 2001, version 1.1.2.

[18] ETSI: *ETSI ES 201 158: Telecommunications security; Lawful Interception (LI); Requirements for network functions*. European Telecommunications Standards Institute, 2002, version 1.2.1.

[19] ETSI: *ETSI TR 102 528: Lawful Interception (LI); Interception domain Architecture for IP networks*. European Telecommunications Standards Institute, 2006, version 1.1.1.

[20] ETSI: ETSI TS 102-232-6: Lawful Interception (LI); Handover Interface and Service-Specific Details (SSD) for IP delivery; Part 6: Service-specific details for IP Multimedia Services. 2008, version 2.3.1.

[21] ETSI: *ETSI TR 101 331: Lawful Interception (LI); Requirements of Law Enforcement Agencies*. European Telecommunications Standards Institute, 2009, version 1.3.1.

[22] ETSI: *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*. European Telecommunications Standards Institute, 2009, version 2.2.1.

[23] ETSI: *ETSI TR 101 671: Lawful Interception (LI); Handover interface for the lawful interception of telecommunications traffic*. European Telecommunications Standards Institute, 2010, version 3.6.1.

[24] ETSI: *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*. European Telecommunications Standards Institute, 2010, version 2.5.1.

[25] ETSI: *ETSI TR 102 232-4: Lawful Interception (LI); Handover Interface and Service-Specific Details (SSD) for IP delivery; Part 4: Service-specific details for Layer 2 services*. European Telecommunications Standards Institute, 2010, version 2.3.1.

[26] ETSI: ETSI TS 102 232-2: Lawful Interception (LI); Handover Interface and Service-Specific Details (SSD) for IP delivery; Part 2: Service-specific details for E-mail services. 2010, version 2.5.1.

[27] ETSI: ETSI TS 102-232-5: Lawful Interception (LI); Handover Interface and Service-Specific Details (SSD) for IP delivery; Part 5: Service-specific details for PSTN/ISDN services. 2010, version 2.5.1.

[28] Franková, B.: Určování identity počítače pomocí odchylky vnitřních hodin. 2013, bakalářská práce, Vysoké učení technické v Brně.

[29] Hoffman, P.: *SMTP Service Extension for Secure SMTP over Transport Layer Security*. IETF, 2002, RFC 3207 (Proposed Standard).

[30] Hoffman, P.; Terplan, K.: *Intelligence Support Systems: Technologies for Lawful Intercepts*. Auerbach Publications, U.S., 2006, ISBN 978-0-8493-2855-8.

[31] Holkovič, M.: Detekce identity na různých vrstvách architektury TCP/IP. 2013, bakalářská práce, Vysoké učení technické v Brně.

[32] Holkovič, M.; Polčák, L.: ndtrack. 2013, `http://www.fit.vutbr.cz/~ipolcak/prods.php?id=308`.

[33] Huang, D.-J.; Yang, K.-T.; Ni, C.-C.; aj.: Clock Skew Based Client Device Identification in Cloud Environments. In *Advanced Information Networking and Applications*, 2012, ISSN 1550-445X, s. 526–533.

[34] Information Sciences Institute University of Southern California, IETF: *Internet Protocol*. 1981, RFC 791 (Internet Standard).

[35] International Organization for Standardization: *ISO/IEC international standard 7498-1:1994 Information Technology – Open Systems Interconnection – Basic Reference Model: The Basic Model*. 1994.

[36] International Organization for Standardization: *International Standard ISO 3166-1, Codes for the representation of names of countries and their subdivisions - Part 1: Country codes, ISO 3166-1: 2006 (E/F)*. 2006.

[37] IP Fabrics, Inc: *DeepProbe: 1Gbps and 10Gbps IP Data Collection Probes*. 2011, citováno 2012-01-18.
URL `http://www.ipfabrics.com/pdf/DeepProbe.pdf`

[38] Jacobson, V.; Braden, B.; Borman, D.: *TCP Extensions for High Performance*. IETF, 1992, RFC 1323 (Proposed Standard, Obsoleted by RFC 7323).

[39] Jirásek, J.: Využití časových informací pro identifikaci počítače. 2012, diplomová práce, Vysoké učení technické v Brně.

[40] Kajan, M.; Koranda, K.; Polčák, L.: Spolehlivá a zabezpečená komunikace v rámci systému pro zákonné odposlechy. Technical Report FIT-TR-2012-007, Faculty of Information Technology, Brno University of Technology, 2012.

[41] Kalt, C.: *Internet Relay Chat: Architecture*. IETF, 2000, RFC 2810 (Informational).

[42] Kalt, C.: *Internet Relay Chat: Channel Management*. IETF, 2000, RFC 2811 (Informational).

[43] Kalt, C.: *Internet Relay Chat: Client Protocol*. IETF, 2000, RFC 2812 (Informational).

[44] Kalt, C.: *Internet Relay Chat: Server Protocol*. IETF, 2000, RFC 2813 (Informational).

[45] Klensin, J. C.: *Simple Mail Transfer Protocol*. IETF, 2008, RFC 5321 (Draft Standard).

[46] Klensina, J.; Freed, N.; Rose, M. T.; aj.: *SMTP Service Extensions*. IETF, 1995, RFC 1869 (Internet Standard).

[47] Kohno, T.; Broido, A.; Claffy, K.: Remote physical device fingerprinting. *IEEE Transactions on Dependable and Secure Computing*, ročník 2, č. 2, 2005: s. 93–108, ISSN 1545-5971.

[48] Lloyd, B.; Simpson, W. A.: *PPP Authentication Protocols*. IETF, 1992, RFC 1334 (Proposed Standard, Obsoleted by RFC 1994).

[49] Mamakos, L.; Lidl, K.; Evarts, J.; aj.: *A Method for Transmitting PPP Over Ethernet (PPPoE)*. IETF, 1999, RFC 2516 (Informational).

[50] Martínek, T.; Kramoliš, P.; Holkovič, M.; aj.: Dynamická identifikace uživatelů v prostředí sítí IPv4 a IPv6. Technical Report FIT-TR-2012-006, Faculty of Information Technology, Brno University of Technology, 2012.

[51] Mockapetris, P.: *Domain Names - Concepts And Facilities*. IETF, 1987, RFC 1034 (Internet Standard).

[52] Mockapetris, P.: *Domain Names - Implementation And Specification*. IETF, 1987, RFC 1035 (Internet Standard).

[53] Murdoch, S. J.: Hot or Not: Revealing Hidden Services by Their Clock Skew. In *Computer and Communications Security*, New York, NY, USA: ACM, 2006, ISBN 1-59593-518-5, s. 27–36.

[54] Myers, J. G.; Rose, M. T.: *Post Office Protocol - Version 3*. IETF, 1996, RFC 1939 (Internet Standard).

[55] Narten, T.; Draves, R.; Krishnan, S.: *Privacy Extensions for Stateless Address Autoconfiguration in IPv6*. IETF, 2007, RFC 4941 (Draft Standard).

[56] Oikarinen, J.; Reed, D.: *Internet Relay Chat Protocol*. IETF, 1993, RFC 1459 (Experimental).

[57] Polčák, L.; Franková, B.: On Reliability of Clock-Skew-Based Remote Computer Identification. In *Proceedings of the 11th International Conference on Security and Cryptography*, SciTePress - Science and Technology Publications, 2014, ISBN 978-989-758-045-1, s. 291–298.

[58] Polčák, L.; Holkovič, M.; Matoušek, P.: A New Approach for Detection of Host Identity in IPv6 Networks. In *Proceedings of the 4th International Conference on Data Communication Networking, 10th International Conference on e-Business and 4th International Conference on Optical Communication Systems*, SciTePress - Science and Technology Publications, 2013, ISBN 978-989-8565-72-3, s. 57–63.

[59] Polčák, L.; Holkovič, M.; Matoušek, P.: Host Identity Detection in IPv6 Networks. *Communications in Computer and Information Science*, ročník 2014, č. 456, 2014: s. 74–89, ISSN 1865-0929.

[60] Polčák, L.; Hranický, R.: Útoky na systémy pro zákonné odposlechy. Technical Report FIT-TR-2012-008, Faculty of Information Technology, Brno University of Technology, 2012.

[61] Polčák, L.; Hranický, R.; Martínek, T.: On Identities in Modern Networks. *The Journal of Digital Forensics, Security and Law*, ročník 2014, č. 2, 2014: s. 9–22, ISSN 1558-7215.

[62] Polčák, L.; Hranický, R.; Matoušek, P.: Hiding TCP Traffic: Threats and Counter-measures. In *Security and Protection of Information 2013, Proceedings of the Conference*, Brno University of Defence, 2013, ISBN 978-80-7231-922-0, s. 83–96.

[63] Polčák, L.; Jirásek, J.; Franková, B.: PC Fingerprinter – pcf. 2012-2014, https://github.com/polcak/pcf.

[64] Polčák, L.; Jirásek, J.; Matoušek, P.: Comment on "Remote Physical Device Fingerprinting". *IEEE Transactions on Dependable and Secure Computing*, ročník 11, č. 5, 2014: s. 494–496, ISSN 1545-5971.

[65] Polčák, L.; Kramoliš, P.; Kajan, M.; aj.: Architektura systému pro zákonné odposlechy. Technical Report FIT-TR-2011-008, Faculty of Information Technology, Brno University of Technology, Brno, Czech Republic, 2011.

[66] Resnick, P.: *Internet Message Format*. IETF, 2008, RFC 5322 (Draft Standard).

[67] Rigney, C.; Rubens, A. C.; Simpson, W. A.; aj.: *Remote Authentication Dial In User Service (RADIUS)*. IETF, 2000, RFC 2865 (Draft Standard).

[68] Rosenberg, J.; Schulzrinne, H.; Camarillo, G.; aj.: *SIP: Session Initiation Protocol*. IETF, 2002, RFC3261 (Proposed Standard).

[69] Saint-Andre, P.: *Extensible Messaging and Presence Protocol (XMPP): Core*. IETF, 2011, RFC 6120 (Proposed Standard).

[70] Saint-Andre, P.: *Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence*. IETF, 2011, RFC 6121 (Proposed Standard).

[71] Sbírka zákonů ČR: *Zákon č. 259/2010 Sb. o elektronických komunikacích ve znění pozdějších předpisů*. 2014, hlava V, díl 1.

[72] Sharma, S.; Hussain, A.; Saran, H.: Experience with Heterogenous Clock-skew Based Device Fingerprinting. In *Workshop on Learning from Authoritative Security Experiment Results*, ACM, 2012, ISBN 978-1-4503-1195-3, s. 9–18.

[73] Simpson, W. A.: *The Point-to-Point Protocol (PPP)*. IETF, 1994, RFC 1661 (Internet Standard).

[74] Simpson, W. A.: *PPP Challenge Handshake Authentication Protocol (CHAP)*. IETF, 1996, RFC 1994 (Proposed Standard).

[75] Utimaco Safeware AG: Lawful Interception in the Digital Age: Vital Elements of an Effective Solution. 2010, White Paper.

[76] Zander, S.; Murdoch, S. J.: An Improved Clock-skew Measurement Technique for Revealing Hidden Services. In *Proceedings of the 17th Conference on Security Symposium*, Berkeley, CA, USA: USENIX Association, 2008, s. 211–225.

Příloha A

Seznam zkratek

AF *Administration Function*: Administrační funkce systému pro zákonné odposlechy, spravuje probíhající odposlechy, str. 10.

BRAS *Broadband Remote Access Server*: RAS pro vysokorychlostní připojení ke vzdálené síti, str. 53.

CC *Content of Communication*: Obsah odposlouchávané komunikace, str. 9.

CC-IIF *Content of Communication – Internal Interception Function*: Odposlouchávací funkce systému pro zákonné odposlechy, odposlouchává provoz zasílaný skrze počítačovou síť, str. 10.

CCCI *Content of Communication Control Interface*: Konfigurační rozhraní CC-IIF v systému pro zákonné odposlechy, str. 10.

CCTF *Content of Communication Trigger Function*: Trigerovací funkce systému pro zákonné odposlechy, nastavuje CC-IIF pro zahájení a ukončení odposlechu, str. 10.

CCTI *Content of Cummunication Trigger Interface*: Trigerovací rozhraní systému pro zákonné odposlechy, spojuje IRI-IIF a CCTF, str. 10.

CID *Communication Identifier*: Identifikátor konkrétní komunikace detekované v rámci zákonného odposlechu, str. 22.

CIN *Communications Identity Number*: Číslo komunikace, které je součástí CID, str. 22.

DCC *Delivery Country Code*: Identifikace země, ve které se nachází MF, součást CID, str. 22.

DHCP *Dynamic Host Configuration Protocol*: Protokol pro přidělování IPv4 adres a dalších informací koncovým stanicím, str. 48.

DHCPv6 *Dynamic Host Configuration Protocol for IPv6*: Prokol pro dynamickou konfiguraci v prostředí IPv6, str. 57.

DSL *Digital Subscriber Line*: Technologie umožňující přenos dat skrze telefonní rozvody. V ČR se nejčastěji setkáváme s připojením k Internetu pomocí ADSL, nebo VDSL., str. 47.

DUID *DHCPv6 Unique Identifier*: Identifikátor počítače používaný v rámci DHCPv6, str. 57.

ETSI *European Telecommunications Standards Institute*: Evropský ústav pro telekomunikační normy, str. 5.

FLAP *Frame Layer Protocol*: Protokol, který zapouzdřuje všechny zprávy zasílané protokolem OSCAR, str. 71.

HI *Handover Interface*: Vnější rozhraní systému pro zákonné odposlechy, obsahuje konfigurační část (HI1), část pro přenos metadat o odposlechu (HI2) a část pro odposlech komunikace podezřelého (HI3), str. 9.

HI1ID *Handover Interface 1 Identifier*: Identifikátor specifikující předmět odposlechu, např. jméno osoby, IP adresa, apod., str. 21.

IAP *Intercept Access Point*: Místo pro připojení odposlouchávacího zařízení do počítačové sítě., str. 12.

INI *Internal Network Interface*: Vnitřní rozhraní systému pro zákonné odposlechy, obsahuje konfigurační část (INI1), část pro přenos metadat o odposlechu (INI2) a část pro odposlech komunikace podezřelého (INI3), str. 10.

IP *Internet Protocol*: Dominantní protokol používaný pro komunikaci mezi počítači připojenými k Internetu. V současné době se nejčastěji používá verze 4 (IPv4), postupně se přechází na novější verzi 6 (IPv6), str. 6.

IPCP *IP Control Protocol*: Kontrolní protokol pro přenos IP skrze spoj PPP, str. 53.

IPv6CP *IPv6 Control Protocol*: Kontrolní protokol pro přenos IPv6 skrze spoj PPP, str. 54.

IRC *Internet Relay Chat*: Protokol pro komunikaci v reálném čase, používaný především na skupinovou komunikaci, str. 68.

IRI *Intercept Related Information*: Metadata související s odposlechem, str. 9.

IRI-IIF *Intercept Related Information – Internal Interception Function*: Funkce dynamické identity systému pro zákonné odposlechy, spravuje identifikátory detekované v síti a generuje metadata k odposlechům (zprávy IRI), str. 10.

LEA *Law Enforcement Agency*: v českém prostředí se většinou jedná o orgány činné v trestním řízení, str. 8.

LEAID *Law Enforcement Agency Identifier*: Řetězec jednoznačně identifikující v rámci SLIS orgán, na jehož žádost byl odposlech aktivován, str. 25.

LI *Lawful Interception*: Zákonný odposlech, str. 5.

LIID *Lawful Interception Identifier*: Jednoznačný identifikátor odposlech dodávaný LEA, str. 18.

LIS *Lawful Interception System*: Systém pro zákonné odposlechy, str. 5.

MAC *Media Access Control*: Součást sítí postavených nad standardy IEEE 802, např. Ethernet, Wi-Fi a další, str. 6.

MDA *Mail Delivery Agent*: Proces ukládající e-mail do e-mailové schránky na cílovém serveru, str. 77.

MF *Mediation Function*: Mediační funkce systému pro zákonné odposlechy, shromažďuje zachycené zprávy IRI a CC a dále je předává odposlouchávajícímu orgánu, str. 10.

MSA *Mail Submission Agent*: Proces přijímající e-maily na serveru, str. 77.

MTA *Mail Transfer Agent*: Proces přeposílající e-mail na jiný server, str. 77.

MUA *Mail User Agent*: Program na správu e-mailů, str. 77.

NAT *Network Address Translation*: Překlad adres nejčastěji protokolu IPv4, str. 6.

ND *Neighbor Discovery*: Mechanismus používaný v rámci IPv6 mimo jiné pro ověřování unikátnosti nakonfigurované adresy, str. 61.

NID *Network Identifier*: síťový identifikátor identifikující osobu, zařízení, spojení apod., str. 19.

$\mathbf{NID_{CC}}$ NID pomocí kterého je sondami hledán zájmový obsah, str. 21.

$\mathbf{NID_{In}}$ NID určený k odposlouchávání při specifikaci odposlechu pro AF, str. 21.

NWO/AP *Network Operator/Access Provider*: Provozovatelé počítačových síti, str. 5.

OSCAR *Open System for Communication in Realtime*: Protokol pro komunikaci v reálném čase používaný v sítích AIM a ICQ, str. 71.

PCAP Formát pro uložení zachycených síťových dat (paketů), str. 23.

PPP *Point-to-Point Protocol*: Protokol pro propojení dvou uzlů v síti samostatným okruhem, str. 53.

PPP LCP *Link Control Protocol*: Protokol používaný při navazování a ukončování spojení protokolem PPP, str. 53.

PPPoE *Point-to-Point Protocol over Ethernet*: Varianta protokolu PPP přenášená v síti postavené na protokolu Ethernet. Používá se například u přípojek DSL, str. 53.

RADIUS *Remote Authentication Dial In User Service*: Protokol používaný pro autentizaci, autorizaci a účtování, str. 50.

RAS *Remote Access Service*: Server provozující služby důležité pro vzdálený přístup k síti (realizovaný typicky pomocí protokolu PPP), str. 50.

RID *Reason Identifier*: Zkrácená verze SID platná pro konkrétní sondu CC-IIF, str. 26.

SDN *Software Defined Networking*: Softwarem řízené sítě, str. 47.

Sec6Net Projekt *Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace* financovaný grantem Ministerstva vnitra ČR VG20102015022, str. 5.

SID *System Identifier*: Pomocný identifikátor množiny odposlechů vztažených k rozsahu odposlouchávaných $\mathrm{NID}_{\mathrm{CC}}$, str. 26.

SIMS *Sec6Net Identity Management System*: Částečný výsledek umožňující obecné využití funkce pro dynamickou identifikaci, str. 114.

SLAAC *Stateless Address Autoconfiguration*: Bezstavová autokonfigurace adres IPv6 na koncových zařízeních, str. 59.

SLIS *Sec6Net Lawful Interception System*: Systém pro zákonné odposlechy vytvořený v rámci popisovaného projektu *Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace*, str. 6.

SMTP *Simple Mail Transfer Protocol*: Protokol pro odesílání e-mailových zpráv a výměnu e-mailových zpráv mezi servery, str. 76.

SNAC *Simple Network Atomic Communication*: Protokol zapouzdřující přenášená data zasílaná protokolem OSCAR, str. 71.

SvP *Service Provider*: Poskytovatel služeb, str. 5.

XMPP *Extensible Messaging and Presence Protocol*: protokol pro komunikaci v reálném čase, str. 64.

YMSG *Yahoo! Messenger Protocol*: Protokol pro komunikaci v reálném čase v síti Yahoo!, str. 75.

Příloha B

Závislosti SLIS, software třetích stran a licence

SLIS je jedním z výsledků vytvořených v rámci projektu *Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace* – software pro Softwarový nástroj pro dohledání pachatele počítačové kriminality, jako takový je šířen v souladu se zákony ČR o vysokoškolském výzkumu v souladu s licenční politikou projektu *Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace*.

Jednotlivé moduly SLIS jsou řešeny jako samostatné funkční programy a jsou licencován převážně pod licencí GPLv3[1]. Moduly využívající knihovnu scapy jsou licencovány pod licencí GPLv2[2]. Moduly využívající knihovnu *ndtrack* jsou licencovány pod licencí Open Source licencí VUT v souladu s rozhodnutím rektora VUT č. 23/2010 (text licence je umístěn v příloze 1). Záplata pro SDN kontroler Pox je šířena pod licencí Apache version 2.0[3].

Modulární systém pro detekci identity je obecně použitelný a jeho části jsou šířeny pod výše zmíněnými licencemi jako samostatný software *Správa identity z projektu Sec6Net*.

SLIS nebo jeho části využívají následující software:

- Python – `http://www.python.org/`
- bash – `http://tiswww.case.edu/php/chet/bash/bashtop.html`
- Cython – `http://www.cython.org/`
- Scapy – `http://www.secdev.org/projects/scapy/`
- IPy – `http://pypi.python.org/pypi/IPy`
- libpcap – `http://www.tcpdump.org/`
- ndwatch – `http://www.fit.vutbr.cz/~lampa/ipv6/`
- libxml2 – `http://www.xmlsoft.org/`
- python-pcapy – `http://corelabs.coresecurity.com/index.php?module=Wiki&action=view&type=tool&name=Pcapy`

[1] https://www.gnu.org/licenses/gpl-3.0.txt
[2] https://www.gnu.org/licenses/gpl.txt
[3] https://www.apache.org/licenses/LICENSE-2.0.txt

- libdb – http://www.oracle.com/technology/software/products/berkeley-db/index.html
- libmysql++ – http://tangentsoft.net/mysql++/
- php5 – http://www.php.net/
- smarty – http://www.smarty.net/
- graphviz – http://www.graphviz.org/