

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

MULTIPLATFORMNÍ BRÁNA PRO HLASOVOU KOMUNIKACI V
REÁLNÉM ČASE

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. RADEK STARZYCZNY

BRNO 2015



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

MULTIPLATFORMNÍ BRÁNA PRO HLASOVOU KOMUNIKACI V REÁLNÉM ČASE

MULTIPLATFORM GATEWAY FOR VOICE COMMUNICATION IN REAL-TIME

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

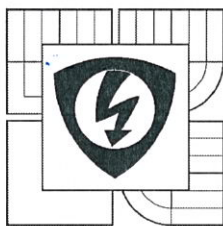
AUTOR PRÁCE
AUTHOR

Bc. RADEK STARZYCZNY

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. BOHUMIL NOVOTNÝ

BRNO 2015



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Diplomová práce

magisterský navazující studijní obor
Telekomunikační a informační technika

Student: Bc. Radek Starzyczny

Ročník: 2

ID: 125642

Akademický rok: 2014/15

NÁZEV TÉMATU:

Multiplatformní brána pro hlasovou komunikaci v reálném čase

POKYNY PRO VYPRACOVÁNÍ:

Důkladně prostudujte problematiku konvergovaných sítí a na základě získaných znalostí navrhnete a následně vytvořte systém využívající VoIP(SIP), GSM, OpenWRT a PBX Asterisk pomocí vhodného HW pro implementaci. Navržený systém bude schopen spojit telefonní hovor mezi analogovým telefonem, HW a SW SIP telefonem a sítí GSM. Spolehlivost a funkčnost systému ověřte analýzou síťové komunikace pomocí dostupných analyzačních prostředků. Brána bude interaktivně reagovat na příchozí komunikaci. Výstupem diplomové práce bude rovněž laboratorní úloha.

DOPORUČENÁ LITERATURA:

[1] ODOM, Wendell, Rus HEALY a Naren MEHTA. Směrování a přepínání sítí: autorizovaný výukový průvodce. Vyd. 1. Brno: Computer Press, 2009, 879 s. ISBN 978-80-251-2520-5.

[2] SIMIONOVICH, Nir. Asterisk gateway interface 1.4 and 1.6 programming: design and develop Asterisk-based VoIP telephony platforms and services using PHP and PHPAGI. 1st ed. Birmingham: Packt Publishing, 2009, v, 200 s. ISBN 978-1-847194-46-6.

Termín zadání: 9. 2. 2015

Termín odevzdání: 26.5.2015

Vedoucí práce: Ing. Bohumil Novotný

Konzultanti diplomové práce:


doc. Ing. Jiří Mišurec, CSc.
předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

ABSTRAKT

Tato diplomová práce je zaměřena na VoIP komunikaci. Popisuje zprovoznění operačního systému OpenWRT, analogového rozhraní TAPI směrovače Gigaset SX762 a GSM brány pro příjem či spojování odchozích hovorů. V práci jsou popsány protokoly podílející se na komunikaci a také základní konfigurace prvků. Nasazení IP telefonie umožňuje snížit náklady na provoz a poskytuje množství přídavných funkcí.

KLÍČOVÁ SLOVA

OpenWrt, PBX Asterisk, GSM, SIP, XMPP, Prosody, Konvergovaná síť, SMS, Hlasová brána, Gigaset SX76x

ABSTRACT

This master's thesis is focused on VoIP communications. It describes deploy of the operating system OpenWRT, analog interface of router Gigaset SX762 and GSM gateway for receiving or place calls. The paper describes the protocols involved in the communication and basic configuration elements. Deploying IP telephony enables to reduce the cost of operation and provides a number of additional functions.

KEYWORDS

OpenWrt, PBX Asterisk, GSM, SIP, XMPP, Prosody, Converged network, SMS, Voice gateway, Gigaset SX76x

STARZYCZNY, Radek *Multiplatformní brána pro hlasovou komunikaci v reálném čase*: diplomová práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2014. 66 s. Vedoucí práce byl Ing. Bohumil Novotný

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Multiplatformní brána pro hlasovou komunikaci v reálném čase“ jsem vypracoval samostatně pod vedením vedoucího diplomové 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 diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

(podpis autora)

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce, kterým byl pan Ing. Bohumil Novotný, za odborné vedení, velice podnětné konzultace, trpělivost a odborné návrhy k práci.

Brno

.....

(podpis autora)

OBSAH

Úvod	10
1 Teoretická část	11
1.1 Telefonní přístroj	11
1.2 Telefonní ústředna	11
1.2.1 Analogové telefonní ústředny	11
1.2.2 Digitální telefonní ústředny	12
1.3 Veřejná telefonní síť	12
1.4 Media brána	12
1.5 Konvergované sítě	12
1.5.1 Typická topologie konvergované sítě	13
1.6 Uvažované technologie a protokoly	14
1.6.1 Protokol TCP	14
1.6.2 Protokol UDP	14
1.6.3 Protokol XMPP	15
1.6.4 Protokol RTP	15
1.6.5 Protokol RTCP	16
1.6.6 Protokol SDP	18
1.6.7 Protokol SIP	18
1.6.8 USB rozhraní	19
1.6.9 Rozhraní TAPI	20
1.6.10 OpenWrt	20
2 Praktická část	22
2.1 Siemens Gigaset SX762	22
2.2 Kompilace OpenWrt	23
2.3 Komunikace s UART rozhraním	24
2.3.1 Aktualizace zavaděče	24
2.3.2 Zavedení OpenWRT do zařízení	26
2.3.3 Úprava HW pro napájení portu	27
2.3.4 USB modem	27
2.3.5 Systémový oddíl na externím zařízení	29
2.4 Uvažovaná topologie	30
2.4.1 Konfigurace LAN a WAN rozhraní směrovače	31
2.5 Pobočková ústředna Asterisk	32
2.5.1 Nastavení SIP	33
2.5.2 Nastavení chan_lantiq	34

2.5.3	Nastavení res_jabber	35
2.5.4	extensions.conf	35
2.5.5	Některé použité funkce a aplikace dialplanu	37
2.5.6	Reakce na příchozí SIP/XMPP zprávu	37
2.6	Analogový telefon	39
2.7	SIP telefon	40
2.8	SW Prosody	40
2.9	Možné směry volání	41
2.10	Zpracování příchozí SMS	43
3	Závěr	44
	Literatura	45
	Seznam symbolů, veličin a zkratk	48
	Seznam příloh	51
A	Makefile pro chan_dongle	52
B	Makefile pro pcapsipdump	54
C	Obrázky zachycující zpracování SMS	55
D	Obsah přiloženého CD	57
E	Laboratorní úloha	58

SEZNAM OBRÁZKŮ

1.1	Oddělené sítě pro hlasovou a datovou komunikaci	13
1.2	Typická konvergovaná síť	14
2.1	Siemens Gigaset SX762	22
2.2	Základní deska routeru Gigaset SX762	22
2.3	Schéma konvertoru RS232 na TTL	24
2.4	Rozvržení významných přípojovacích bodů na základní desce	25
2.5	Zapojení při nahrávání firmware	26
2.6	Úprava HW pro zachování napájení USB	27
2.7	Uvažovaná topologie	30
2.8	Sagem C95	39
2.9	SMC Networks SMCDSP-205	40
C.1	Zaslání SMS z mobilního telefonu	55
C.2	Okno XMPP zprávy v programu Pidgin	56
C.3	Konzole PBX a reakce na příchozí SMS	56

SEZNAM TABULEK

1.1	Základní rozložení hlavičky RTP datagramu	16
1.2	Základní rozložení hlavičky RTCP datagramu	17
1.3	Popis typů SDES	17
1.4	Přehled verzí rozhraní TAPI	20
1.5	Přehled verzí systému OpenWrt	21
2.1	Tabulka s popisem některých integrovaných obvodů	23
2.2	Popis vstupně výstupních bran zařízení	25
2.3	Specifikace použitého modemu	28
2.4	Tabulka popisující rozložení FLASH paměti	29
2.5	Tabulka některých kompatibilních zařízení	34
2.6	Základní nastavení SIP účtů použitých v této práci	34
2.7	Nastavení uživatelů XMPP serveru	41
2.8	Tabulka popisující možnosti volání	42
2.9	Tabulka popisující možnosti přenosu zpráv	42

ÚVOD

Mnoho dnešních firem hledá způsob jak minimalizovat náklady spojené s telefonováním, ať už v rámci firmy nebo mimo ni. Jedna z cest vedoucí ke snížení nákladů je využití moderních technologií pro komunikaci. Tato semestrální práce se věnuje problematice konvergovaných sítí. Pomocí získaných znalostí byl navrhnut systém využívající VoIP (Voice over Internet Protocol), GSM (Groupe Spécial Mobil), OpenWRT a PBX (Private branch exchange) Asterisk. Navržený systém je schopen spojit telefonní hovor mezi analogovým telefonem, HW (Hardware) / SW (Software) SIP (Session Initiation Protocol) telefonem a sítí GSM. Také je možné zaslat XMPP (Extensible Messaging and Presence Protocol) zprávy, například na základě příchozí SMS (Short message service) z GSM sítě.

První část práce se zabývá konvergovanými sítěmi, jejich obecným popisem a typické topologii konvergované sítě menší firmy. V této části je také popis různých protokolů a technologií důležitých pro tuto práci. Následně je popsán postup pro zprovoznění a nastavení uvažovaného řešení. Práce je následně částečně transformována do samostatné laboratorní úlohy, která má za úkol seznámit posluchače s možnostmi moderních komunikačních prostředků.

1 TEORETICKÁ ČÁST

1.1 Telefonní přístroj

Telefon je telekomunikační zařízení přenášející hlas, primárně prostřednictvím elektrických signálů. První telefon pravděpodobně zkonstruoval Ital Antonio Meucci v Havaně na Kubě v roce 1849 [19]. Sloužil mu k tomu, aby mohl mluvit ze suterénu se svou invalidní ženou ve třetím patře domu. Ve frankfurtském Fyzikálním spolku 26. října 1861 Němec Johann Philip Reis z Friedrichsdorfu přenášel hlas pomocí vodičů na vzdálenost 91,4 m. Oba využili, v té době dobře známého, principu přenosu zvukových vibrací pomocí napnutého provazu.

Dne 14. února 1876 prošli dveřmi patentového úřadu Spojených států dva muži, aby zde ohlásili, že učinili vynález, který má změnit svět: profesor fyziologie řeči v Bostonu Alexander Graham Bell a pracovník společnosti Western Union Elisha Gray. Časový sled událostí oné kritické hodiny, v níž oba vynálezci vzali za kliku příslušné kanceláře nebyl nikdy bezpečně prokázán. Nicméně před soudem si jeden z úředníků „vzpomenul“, že první vstoupil Bell, který se tak stal „úředně uznaným“ vynálezcem telefonu [18].

1.2 Telefonní ústředna

Telefonní ústředna je telekomunikační zařízení, které zajišťuje spojování telefonních hovorů a další funkce včetně účtování telekomunikačních služeb. K místním telefonním ústřednám (v poslední době nahrazeným přístupovými koncentrátory) jsou připojeny telefony účastníků. Uzlové nebo také tranzitní telefonní ústředny zajišťují propojení na delší vzdálenosti. Zvláštním druhem jsou mobilní ústředny, které kromě spojování hovorů plní i úlohy spojené s mobility managementem a roamingem ¹.

Během svého vývoje prošly telefonní ústředny mnoha podobami a implementacemi. Největším pokrokem ve vývoji byl přechod z analogové podoby do podoby digitální (nasazení pulsní volby, vícetónové volby), proto také pomocí tohoto kritéria můžeme ústředny rozdělit na dva druhy:

- analogové telefonní ústředny (podrobněji 1. až 3. generace)
- digitální telefonní ústředny (4. a zatím poslední 5. generace)

1.2.1 Analogové telefonní ústředny

- 1. generace - synchronní systémy s krokovými voliči

¹Roaming je poskytování telekomunikačních služeb účastníkovi v jiné síti (zemi), než kde má účastník zaregistrované svoje telekomunikační služby.

- 2. generace - křížové spínače s asynchronními voliči, registry pro záznam čísel
- 3. generace - centrální programové řízení (procesor), elektromechanickým či elektronickým spojováním, analogové systémy

1.2.2 Digitální telefonní ústředny

- 4. generace - rozproštěné programové řízení, digitální spojování (mezi hlavní výrobce patří firmy Ericsson, Alcatel, Siemens)
- 5. generace - systémy na bázi paketového principu. Komutace hovorového signálu pomocí tzv. „soft switch“ - spojovací pole je realizováno softwarem

1.3 Veřejná telefonní síť

Pevná telefonní síť je komutační² telefonní síť, do které jsou telefonní přístroje připojeny pomocí účastnického vedení, hovorově někdy označovaného „pevná linka“. Tato účastnická vedení jsou zakončena v telefonních ústřednách, které umožňují propojení více účastníků v jednom městě. Pomocí meziměstských linek³ lze pak propojovat i účastníky v různých městech.

1.4 Media brána

Media brána⁴ je zařízení umožňující propojení různorodých telekomunikačních sítí jako PSTN, sítě další generace (2G, 3G..) nebo PBX.

Protože media brána propojuje navzájem nekompatibilní sítě, jednou z jejich hlavních funkcí je konverze mezi různými způsoby přenosu a kódování, navíc jsou implementovány funkce jako pulzní nebo tónová volba.

Media brány mohou být ovládány také centrálně pomocí „Media Gateway Controller“, který řídí spojování hovorů a signalizaci. Komunikace mezi těmito prvky je obvykle zajištěna pomocí protokolů jako MGCP, Megaco (H.248) nebo SIP.

VoIP media brány provádějí konverzi hlasu (TDM) na protokoly využívané k přenosu informací, z pravidla se jedná o RTP

1.5 Konvergované sítě

Na konci minulého století se stal protokol IP verze 4 jedním z nejpopulárnějších síťových protokolů vůbec. Díky flexibilitě, jednoduchosti a popularitě tohoto protokolu

²Síť s přepojováním okruhů

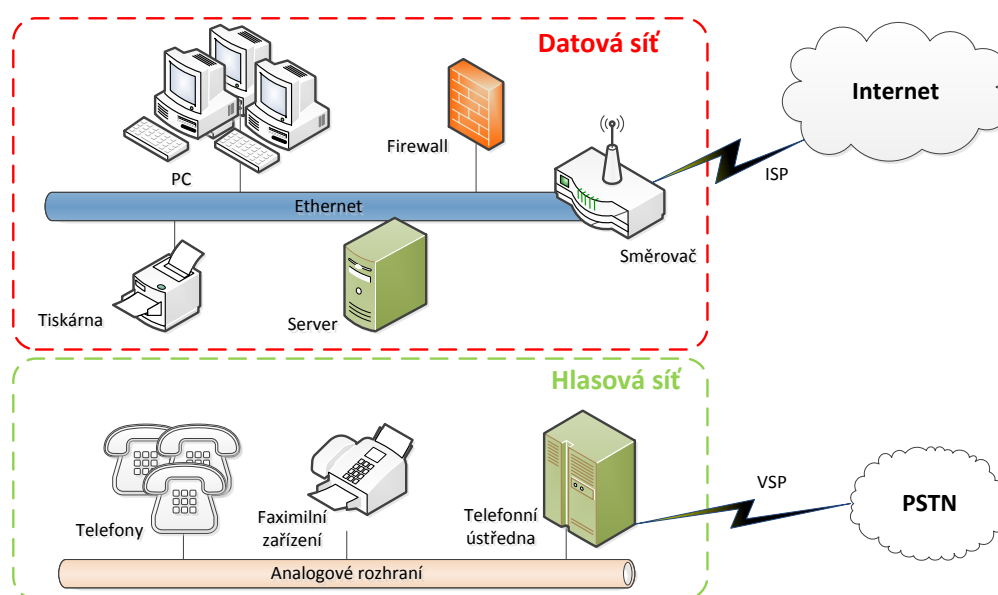
³V literatuře obvykle označováno anglickým názvem trunk

⁴V literatuře obvykle označováno anglickým názvem media gateway

vznikla možnost spojit množství oddělených komunikačních sítí do jedné, založené na protokolu IP. Konvergovanou síť si lze tedy představit jako síť přenášející různý druh dat (soubory, elektronická pošta, hlas nebo video) za pomoci IP datagramu⁵. Protože se sítí přenáší různý obsah, je potřeba tento provoz nějakým způsobem řídit a zajistit správné chování pro konkrétní typ provozu. Obecně se mechanismy pro řízení provozu označují jako kvalita služby QoS (Quality of Service). Důležitá a často opomíjená výhoda konvergentních sítí je schopnost využít stávající infrastruktury počítačové sítě k hlasovým přenosům.

1.5.1 Typická topologie konvergované sítě

Na obrázku 1.1 je vyobrazena typická topologie firmy před příchodem konvergované sítě. Jak je z obrázku 1.1 patrné, jsou zde dvě oddělené sítě. Každá z nich je náročná na údržbu a provoz. Provozovat takto oddělené sítě je neefektivní.

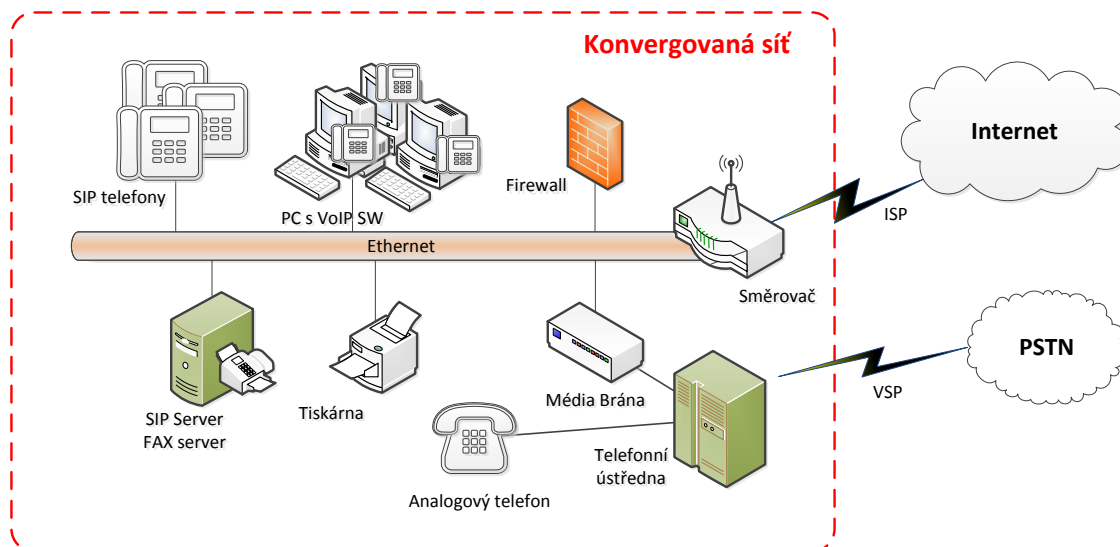


Obr. 1.1: Oddělené sítě pro hlasovou a datovou komunikaci

Na obrázku 1.2 je vidět typické zapojení konvergované sítě, využívající infrastrukturu datové sítě pro hlasové přenosy. Výhodou tohoto řešení je, že se na hlasovém přenosu mohou účastnit navíc počítače s vhodným SW klientem nebo digitální telefony⁶.

⁵IP datagram je v literatuře často označován anglickým názvem packet

⁶Z pravidla se jedná o SIP telefony



Obr. 1.2: Typická konvergovaná síť

1.6 Uvažované technologie a protokoly

1.6.1 Protokol TCP

TCP protokol je protokol transportní vrstvy modelu OSI. Hlavní funkcí TCP je segmentace příchozích dat do velikosti IP datagramu.

- spolehlivá transportní služba
- služba se spojením
- efektivní využití přenosových kanálů
- transparentní přenos
- duplexní spojení
- rozlišení koncových stanic pomocí portů

1.6.2 Protokol UDP

Spolu s TCP je UDP protokol na transportní vrstvě OSI modelu. Oproti TCP se ale jedná o protokol nepotvrzující odeslané datagramy, a to ani v případě výskytu chyb při přenosu. Případné problémy spojené s přenosem dat jsou předávány protokolům na úrovni aplikační vrstvy.

1.6.3 Protokol XMPP

Počátky protokolu XMPP (Extensible Messaging and Presence Protocol) sahají do roku 1998, kdy vývojář Jeremie Miller založil projekt s názvem Jabber. Jedná se o protokol aplikační vrstvy pro posílání zpráv a zjištění stavu. Protokol je založen na výměně XML⁷ elementů. RFC normy popisující XMPP jsou:

- RFC 3920[11] - obecná specifikace protokolu
- RFC 3921[12] - samotný instant messaging a zobrazení stavu.

Servery XMPP protokolu standardně komunikují na TCP portu 5222.

XMPP XEP

XMPP Extension Protocol (XEP)⁸ je zdokumentované rozšíření protokolu XMPP, jedná se o rozšíření nad rámec RFC. Některé zajímavé rozšíření protokolu XMPP jsou:

- XEP-0009 pro vzdálené vykonávání procedur (RPC)
- XEP-0183 Jingle Telepathy Transport Method - IP telefonie přes protokol XMPP využívaný v Google Talk

1.6.4 Protokol RTP

RTP (Real-Time Protocol) je protokol pro přenos multimediálních dat v reálném čase pomocí IP datagramu. Pro transport je využit UDP protokol transportní vrstvy. Nejnovější popis protokolu je v RFC 3550[8]. Protože se k přenosu používá protokol UDP, je výsledný přenos nespolehlivý, respektive bez podpory kvality služeb QoS. Toto je jedním z důvodů, proč je spolu s RTP využit také RTCP (Real-Time Control Protocol) přenášející informace o kvalitě poskytované služby. Každá sestavená relace obsahuje číslo portu pro protokol RTP, RTCP a IP adresy obou zúčastněných.

RTP datagram lze rozdělit na záhlaví o velikosti 12 bajtů (identifikace) a přenášená data s proměnnou velikostí 20 až 160 bajtů (vlastní užitečná data). Velikosti jednotlivých částí jsou ovlivněny druhem hovoru, který je přenášen. Grafické znázornění RTP datagramu je v tabulce 1.1. Při hlasové komunikaci na IP verze 4 je zvětšeno záhlaví o 8 bajtů (UDP) + 20 bajtů (IP) na výslednou velikost 40 bajtů. Kompletní popis je uveden v dokumentu RFC 3550 [8].

- **Version** - verze protokolu, v RFC 3550 je definovaná hodnota 2
- **Padding (P)** - informace o velikosti doplňkových oktetů pro příjemce, které budou ignorovány při dekódování

⁷XML (Extensible Markup Language) je obecný značkovací jazyk standardizovaný konsorciem W3C.

⁸Starší název odvozen od původního projektu Jabber: JEP-Jabber Enhancement Proposal

- **Extension (E)** - informace o nutnosti rozšířit hlavičku RTP datagramu
- **Marker (M)** - využít například pro označení prvního IP datagramu hlasového přenosu
- **CSRC count (CC)** - Definuje počet zařízení (zdrojů), dat v paketu
- **Payload type (PT)** - Určuje typ přenášených dat jako kodek nebo vzorkovací frekvence
- **Sequence number** - Identifikaci paketů, hodnota se zvyšuje o 1 s každým dalším paketem. Počáteční hodnotu nastavuje náhodně vysílač a kvůli bezpečnosti by neměla začínat nulou
- **Timestamp** - Určuje první oktet dat v paketu. Lze využít k nalezení začátku přehrávání multimediálních dat
- **SSRC - Synchronization Source Identifier** - Umožňuje identifikovat účastníky RTP spojení, kdy je každému účastníkovi přiřazeno jedinečné SSRC. Toto číslo pak účastník používá v každém paketu, který odešle. Pro zajištění jedinečného SSRC pro každého účastníka se používá detekce kolize. Při detekci kolize se využije zpráva RTCP BYE (více v popisu zpráv 1.6.5), kde účastník definuje nové SSRC
- **Payload** - Vlastní užitečná data

Tab. 1.1: Základní rozložení hlavičky RTP datagramu

Offset [b]	0-1	2	3	4-7	8	9-15	16-31
0	Version	P	X	CC	M	PT	Sequence Number
32	Timestamp						
64	SSRC – Synchronization source identifier						
96	Payload ...						

1.6.5 Protokol RTCP

Protokol RTCP (Real-Time Control Protocol) je řídicím protokolem pro RTP. Obdobně jako SDP se sám nepodílí na přenosu multimediálních dat, naopak jeho využití je ve zpětném zjištění kvality služby (QoS) poskytované přenosem RTP. Pro svůj přenos využívá UDP protokolu transportní vrstvy. Nejnovější popis protokolu je v RFC 3550 [8], stejně jako RTP. Ověření prostředků nebo šifrování multimediálního toku není součástí implementace, k tomuto účelu může být použit protokol SRTCP (Secure Real-time Transport Control Protocol). Grafické vyobrazení RTCP datagramu je v tabulce 1.2.

Protokolem jsou shromažďovány informace o spojení, jako jsou:

- počet odeslaných paketů (počet odeslaných bajtů)

- počet ztracených paketů
- kolísání zpoždění⁹
- zpětnou vazbu a dobu odezvy

Tab. 1.2: Základní rozložení hlavičky RTCP datagramu

Offset [b]	0-1	2	3-7	8-15	16-31
0	Version	Padding	Item Count	Packet type	Length

Typy RTCP datagramu

Pro protokol RTCP je definováno 5 různých druhů RTCP datagramu.

- **Sender Report (SR)**: Zprávy SR posílají aktivní účastníci (vysílače), které tak umožňují přijímači získat informace o přenosu jako je např. přenosová rychlost nebo kvalita přenosu.
- **Receiver Report (RR)**: Zprávy RR vytváří pasivní účastníci (příjímače) a lze pomocí nich určit statistiky RTP přenosu, což jsou čísla ztracených paketů a informace o změně zpoždění (jitter). Díky těmto souhrnným informacím o přenosu může pak vysílací strana zvýšit nebo snížit kvalitu přenášeného obsahu.
- **Source Description (SDS)**: SDS se používá pro určení informací o uživateli. Nejdůležitější pole je CNAME, které má hodnotu 1 a jednoznačně určuje jméno uživatele v rámci relace. Všechny pole podle doporučení RFC 3550 jsou uvedeny v tabulce 1.3

Tab. 1.3: Popis typů SDS

Hodnota	Název	Význam
0	END	Konec seznamu SDS
1	CNAME	Uživatelské a doménové jméno v kanonickém zápisu
2	NAME	Uživatelské jméno
3	EMAIL	Emailová adresa
4	PHONE	Telefonní číslo (v mezinárodním tvaru)
5	LOC	Geografická poloha uživatele
7	NOTE	Aktuální stav zdroje

- **Goodbye (BYE)** : Účastník použije zprávu BYE v případě, když opouští aktuální relaci nebo když musí změnit svoje SSRC.
- **Application specific (APP)**: Zpráva APP se používá pro testovací účely.

Díky parametru kolísání zpoždění (jitter), který je uveden u paketu SR, lze spočítat průměrné zpoždění RTP proudu:

$$D(P) = (R(P) - R(P - 1)) - (S(P) - S(P - 1)) \quad (1.1)$$

⁹V literatuře obvykle označované anglicky „jitter“

kde:

$S(P)$ – časové razítko RTP paketu P

$R(P)$ – čas, kdy přijde RTP paket P

$D(P)$ – rozdíl časů dvou po sobě jdoucích RTP paketů

$J(P)$ – průměrná hodnota změny zpoždění u příchozích RTP paketů

$$J(P) = \frac{15}{16}J(P-1) + \frac{1}{16}|D(I)| \quad (1.2)$$

1.6.6 Protokol SDP

SDP (Session Description Protocol) je textový protokol¹⁰ sloužící k popisu multimediálních relací. Protokol samotný nepřenáší vlastní užitečná data, ale pouze jejich popis. Nejnovější dokumentace protokolu je v RFC 4566 [9]. Zařízení nabízející multimediální službu pomocí SDP zprávy informuje klienty o vlastnostech přenosu. Tyto informace mohou být následně využity například pro navazování spojení u multimediálních konferencí, uskutečnění telefonního hovoru VoIP nebo třeba pro příjem internetového vysílání (televize, rádia). Pro přenos SDP zpráv se využívají protokoly aplikační vrstvy, jako SIP (Session Initiation Protocol), RTSP (Real-Time Streaming Protocol).

1.6.7 Protokol SIP

SIP (Session Initiation Protocol) protokol je v současné době nepoužívanější protokol pro signalizaci ve VoIP. Vychází z protokolu HTTP, jedná se tedy o textový protokol. Pro přenos vlastních dat se ve VoIP primárně používá RTP, vlastnosti relace pak přenáší SDP. SIP protokol zajišťuje následující činnosti:

- **Vyhledávání uživatele** - vyhledání volaného při inicializaci relace
- **Nastavení relace** - vyjednání parametrů relace
- **Změna relace** - modifikování relace za provozu (přesměrování, ukončení)
- **Definování schopností uživatele** - specifikace koncového terminálu (přenosová rychlost, typ kodeku, atp.)
- **Dostupnost uživatele** - ověření stavu účastníka (obsazeno, přesměrováno, nečinný, atp.)

Základní metody protokolu SIP

Mezi základní metody protokolu patří:

- **REGISTER** - registrace účastníka na serveru
- **INVITE** - zahájení komunikace a vyjednávání o vlastnostech nové relace

¹⁰Využívá znakovou sadu UTF-8

- **ACK** - potvrzení zahájení relace
- **CANCEL** - přerušení relace ještě před jejím navázáním
- **BYE** - ukončení relace
- **OPTIONS** - žádost o možnosti volajícího, bez sestavení volání

Chybová hlášení protokolu SIP

Protokol SIP přejímá z HTTP také rozdělení stavových hlášení. Vedle číselného označení mají jednotlivé chyby také textovou verzi. Kategorie chybových hlášení jsou následující:

- **1xx** - probíhá bez problémů, zatím není relace ukončena
- **2xx** - relace ukončena bez problémů
- **3xx** - přesměrování
- **4xx** - chyba klienta
- **5xx** - chyba serveru
- **6xx** - fatální chyba

1.6.8 USB rozhraní

USB rozhraní nabízí obousměrnou komunikaci mezi periferií a hostovaným systémem. Sběrnice je obecně popisována jako zřetěžená hvězdicová architektura a každé zařízení komunikuje s hostitelským portem, jako kdyby mělo vyhrazené spojení. Propojování lze řetězit až do sedmé úrovně nebo do 127 portů. Unikátní vlastnost USB je, že zařízení může mít hub integrován přímo v sobě. Přes všechna zřetězení se huby chovají pro software transparentně a každé zařízení může být samostatně adresováno. Zařízení poskytující hostiteli své prostředky skrze USB sběrnici lze rozdělit na dvě skupiny:

- **Device** - zařízení neposkytující konkrétní služby, pouze rozšiřuje služby sběrnice, například USB rozbočovač
- **Function** - zařízení poskytující konkrétní služby, například klávesnice

Obvykle je USB modem složené zařízení (typ compound), poskytující rozbočovač (device) a několik zařízení (function). Různé modely modemů se mohou lišit v implementaci, např.:

- Modem K3765 - Jednoduché USB zařízení - není potřeba zařízení přepínat, modem je rozpoznán operačním systémem a vytvoří se sériové zařízení pro komunikaci
- Modem K3770 - Složené USB zařízení - na první pohled se pro hostující operační systém tváří jako CD-ROM, lze pomocí SW¹¹ přepnout do funkce mo-

¹¹pomocí usb-modeswitch

demu (případně zpřístupní dostupné SD karty pomocí emulace), teprve po přepnutí je dostupné sériové zařízení pro komunikaci

- Modem K4305 - MBIM standard - namísto sériového rozhraní je dostupná sada API volání (knihovna libmbim), pomocí kterých je možné zařízení ovládat

1.6.9 Rozhraní TAPI

Telephony Application Programming Interface (TAPI) rozhraní bylo vyvinuto společností Microsoft, aby operační systém firmy Microsoft mohl využívat telefonní služby. Rozhraní zpřístupňuje ovládání telefonních funkcí (volání, záznamník, odmítnutí hovoru. . .) operačnímu systému a jeho aplikacím. TAPI rozhraní je především využíváno pro komunikaci s modemem nebo připojenou pobočkovou ústřednou. Rozdělení verzí je vypsáno v tabulce 1.4.

Tab. 1.4: Přehled verzí rozhraní TAPI

verze TAPI	verze Windows	poznámka
1.3	3.1	pouze 16-bit aplikace
1.4	95	32-bit aplikace
2.0	NT 4.0	podpora automatické distribuce hovorů, rozšíření funkcí PBX
2.1	XP	
3.0	2000	Začátek podpory VoIP
3.1	XP	

1.6.10 OpenWrt

Začátky projektu OpenWrt sahají do první poloviny roku 2004. První verze systému OpenWrt byla postavena na zdrojových kódech uvolněných pod GPL [7] licencí pro zařízení WRT54G a tzv. buildroot projektu uclibc [6]. Tato verze byla anglicky často označována jako „stable release“ (stabilní vydání) a takřka ihned si jí oblíbily široké řady uživatelů.

V roce 2005 se k projektu připojili další vývojáři. Po několika měsících uzavřeného vývoje se tým rozhodl uvolnit první experimentální verzi systému. Tato experimentální verze používala značně pozměněný systém k sestavení (build system), založený na buildroot verze 2. OpenWrt používá oficiální zdrojové kódy GNU/Linux jádra, které následně upravují pomocí tzv. patch¹² souborů. Vývojový tým se snaží znovu implementovat většinu proprietárního kódu různých výrobců. Existuje například nástroj pro zápis nového firmware přímo do FLASH s názvem *mtb*, pro

¹²Patch je soupis změn obsahující opravu nebo úpravu software

konfiguraci bezdrátových adaptérů (wlcompat/wificonf) nebo pro konfiguraci přepínačů s podporou VLAN pomocí virtuálního souborového systému */proc* . Historie verzí systému OpenWrt spolu s názvy jednotlivých vydání je vypsána v tabulce 1.5.

Tab. 1.5: Přehled verzí systému OpenWrt

Verze	datum vydání	pojmenování	SVN ID
Barrier Breaker 14.07	2014 – Říjen	Barrier Breaker	r42625
Attitude Adjustment 12.09	2013 – Duben	Attitude Adjustment	r36088
Backfire 10.03.1	2011 – Prosinec	Backfire	r29592
Backfire 10.03	2010 – Duben	Backfire	r20728
Kamikaze 8.09.2	2010 – Leden	Kamikaze	r18801
Kamikaze 8.09.1	2009 – Červen	Kamikaze	r16278
Kamikaze 8.09	2008 – Září	Kamikaze	r14510
Kamikaze 7.09	2008 – Září	Kamikaze	r7831
Kamikaze 7.06	2009 – Červen	Kamikaze	r7204
White Russian 0.9	2007 – Leden	White Russian	r6257

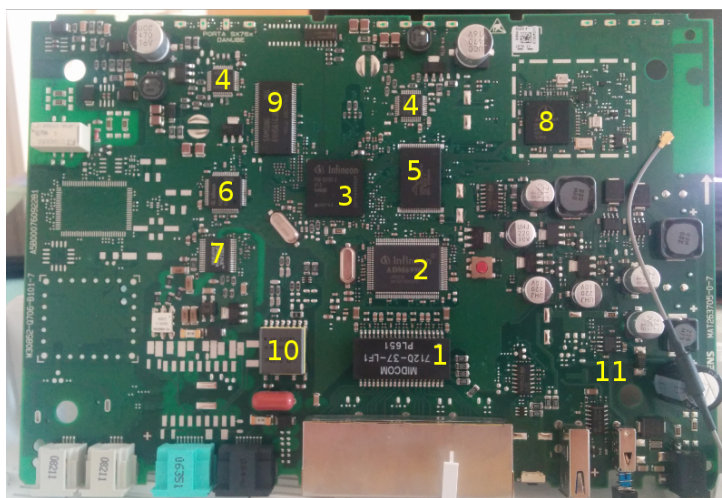
2 PRAKTICKÁ ČÁST

2.1 Siemens Gigaset SX762



Obr. 2.1: Siemens Gigaset SX762

Jako zařízení pro realizaci práce byl vybrán router firmy Siemens Gigaset SX762 (obrázek 2.1). Jedná se o A/VDSL router disponující, krom běžných rozhraní, také dvojicí RJ11 konektorů (pro připojení analogových telefonu, FXS) a USB 2.0 rozhraním (využito pro GSM modem). Vybaven je procesorem Lantiq PSB 50702 [21] o taktovací frekvenci 333 MHz, kde druhé jádro je využito pro VoIP. Operační systém OpenWrt podporuje tento router od verze 12.09.



Obr. 2.2: Základní deska routeru Gigaset SX762

V následující tabulce je popis některých integrovaných obvodů. Číselné označení v tabulce 2.1 odpovídá číselnému označení na obrázku 2.1.

Tab. 2.1: Tabulka s popisem některých integrovaných obvodů

Číslo	Označení obvodu	Popis
	Popis výrobce	
1	000-7120-37	Oddělovací transformátor pro síť LAN
	DATACOM TRANSFORMER FOR LAN; 10/100 BASE-T; AUTOMDIX APPLICATION(S)	
2	ADM6996I	Programovatelný přepínač
	6 port 10/100 Mb/s Single Chip Ethernet Switch Controller	
3	PSB 50702 v1.3	Obvod Danube s integrovaným DSP
	ADSL2/2+ IAD-on-Chip Solution for CPE	
4	Infineon PEF4268F	Rozhraní pro analogovou linku
	SLIC-DC Subscriber Line Interface Circuit with Integrated DC/DC Converter	
5	Spansion S29GL064A10TF	Paměť typu FLASH
	3.0-Volt only Page Mode Flash Memory	
6	Infineon PEB3164F v1.1	Obvod s hlasovým kodekem
	IC VOICE ACCESS CODEC TQFP64	
7	LITELINK CPC5622A	Obvod s telefonním rozhraním
	32-Pin SOIC Phone Line Interface	
8	AR2413A-001	Obvod WiFi rozhraní Atheros
	Atheros AR2413 wireless chipset	
9	K4H561638H-UCB3	Paměť typu DRAM
	256Mb H-die DDR SDRAM Specification	
10	AE 52237 WE-MIDCOM	Transformátor pro odělení A/VDSL sítě
	xDSL Signal transformer Annex A,B,C	
11	MP1409ES MPS918	Obvod dodávající napájení pro USB rozhraní
	DC/DC switching regulator	

2.2 Kompilace OpenWrt

Kompilací je myšlen překlad zdrojového kódu, ve formě čitelné člověkem, do binárního kódu (strojový kód). Na operačních systémech GNU/Linux často používají překladače rodiny gcc (the GNU Compiler Collection). V dobách před vznikem vyšších programovacích jazyků¹ psali programátoři své programy přímo ve strojovém kódu. Strojový kód je však pro větší projekty nevyhovující, protože složitější funkce jsou již velice nepřehledné, a znemožňuje jednoduše přepoužít existující kód. Jak bylo psáno v kapitole 1.6.10 popisující operační systém OpenWrt, pro sestavení systému OpenWrt je použit buildroot [6].

Kroky pro úspěšnou kompilaci systému OpenWrt jsou:

- Aktualizace (stáhnutí) OpenWrt zdrojových kódů

```
user@dev:~# git clone git://git.openwrt.org/14.07/openwrt.git
```

- Aktualizace a instalace balíčků

```
user@dev:~# ./openwrt/scripts/feeds update -a
user@dev:~# ./openwrt/scripts/feeds install -a
```

¹Vyšší programovací jazyk je programovací jazyk s větší mírou abstrakce, např. C, Java, Python

- Volba a úprava výsledných vlastností systému

```
user@dev:~# cd openwrt
user@dev: openwrt# wget -O .config https://downloads.openwrt.org/barrier_breaker
/14.07/lantiq/xway/config.lantiq_xway
user@dev: openwrt# make menuconfig
```

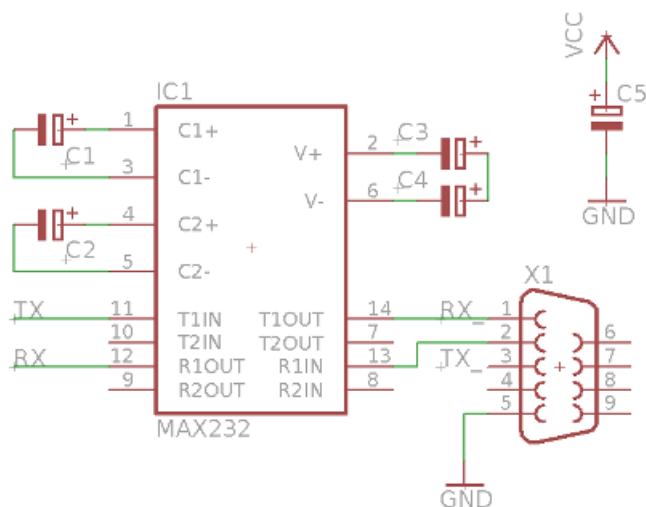
- Vlastní kompilace

```
user@dev: openwrt# make
```

- Instalace vytvořeného firmwaru do zařízení

2.3 Komunikace s UART rozhraním

Většina výrobců umísťuje do svých zařízení rozhraní pro účely testování, nahrávání firmwaru či kontroly při výrobě. Za obsluhu tohoto rozhraní se stará zavaděč. Router řady Gigaset SX76x má takovéto rozhraní také, přesněji se jedná o UART (Universal asynchronous receiver transmitter) s vyvedenými přípojnými body přímo na desce plošných spojů (přesná lokace a popis je popsán v tabulce 2.3.1). Pro komunikaci s rozhraním UART je využit převodník UART <> RS232. Na obrázku 2.3 je vyobrazeno schéma s využitím integrovaného obvodu MAX232².



Obr. 2.3: Schéma konvertoru RS232 na TTL

2.3.1 Aktualizace zavaděče

Firmware výrobce obsahuje dvojici zavaděčů. Úkol prvního z nich (primárního) je ověření kontrolního součtu³ hlavičky druhého zavaděče. Druhý provede stejnou

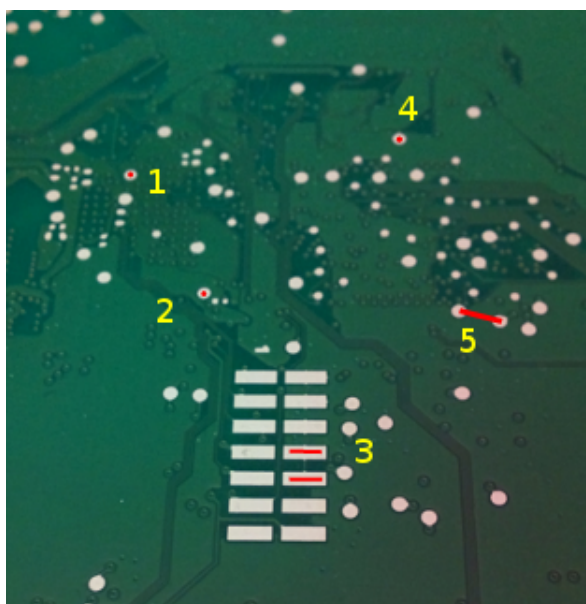
²Jedná se o zapojení doporučené výrobcem.

³Vytvořený pomocí hašovací funkce SHA-2, konkrétně SHA-256.

operaci s firmware. Generování kontrolního součtu je prováděno pomocí privátního klíče⁴, který je nedostupný. Protože je ale kontrolováno pouze počátečních 64 bitů z každého 64 Kb bloku, je možné upravit sekundární zavaděč, jehož velikost je menší než 64 Kb. Pro aktualizaci je potřeba přepnout zařízení do UART módu pomocí propojení vstupně-výstupních bran podle tabulky 2.3.1. Lokace jednotlivých vstupně-výstupních bran je pak vyobrazena na obrázku 2.4.

Tab. 2.2: Popis vstupně výstupních bran zařízení

číslo	popisek	standardní zavedení	UART režim	SPI režim
1	Tx	nezapojeno		
2	Rx	nezapojeno		
3	GND			
4	BOOT-1	nezapojeno	GND	3,3 v
5	BOOT-2	rozpojeno	spojit	spojit



Obr. 2.4: Rozvržení významných přípojovacích bodů na základní desce

Po zapnutí zařízení s volbou UART se zavádění OS zastaví a uboot čeká na primární zavaděč *u-boot.asc*:

```
ROM VER: 1.0.3
CFG 04
Read EEPROMX
X
UART
```

⁴Klíč RSA velikost 1024 bitů

Připravený zavaděč je možné nahrát do zařízení pomocí:

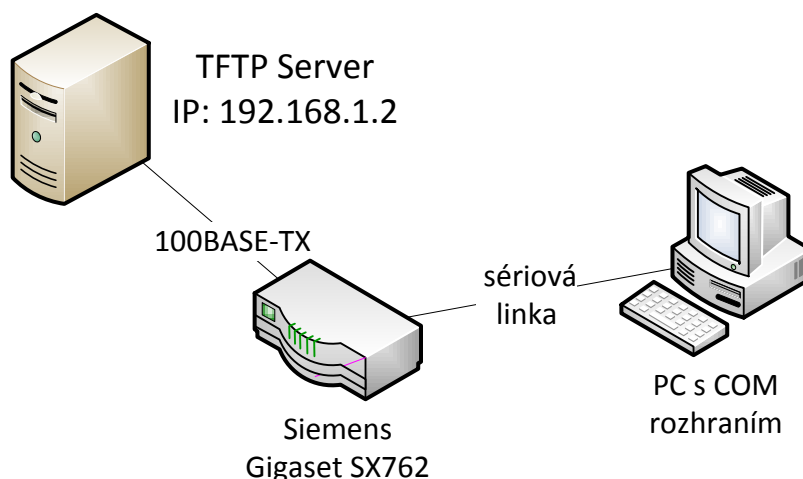
```
cat u-boot.asc > /dev/ttySx
```

kde *ttySx* je zařízení, na kterém je připojen převodník. Po načtení *u-boot.asc* pomocí sériové linky zařízení pokračuje v zavádění nově nahraného zavaděče. Protože je tento zavaděč zatím pouze v RAM je potřeba jej nahrát také do ROM pomocí:

```
run update-uboot-nor
```

po úspěšné aktualizaci je možné zařízení vypnout a nastavit opět standardní zavádění.

2.3.2 Zavedení OpenWRT do zařízení



Obr. 2.5: Zapojení při nahrávání firmware

K zavedení alternativního firmware je využit TFTP server, který má v kořenové složce uloženy soubory vzniklé kompilací:

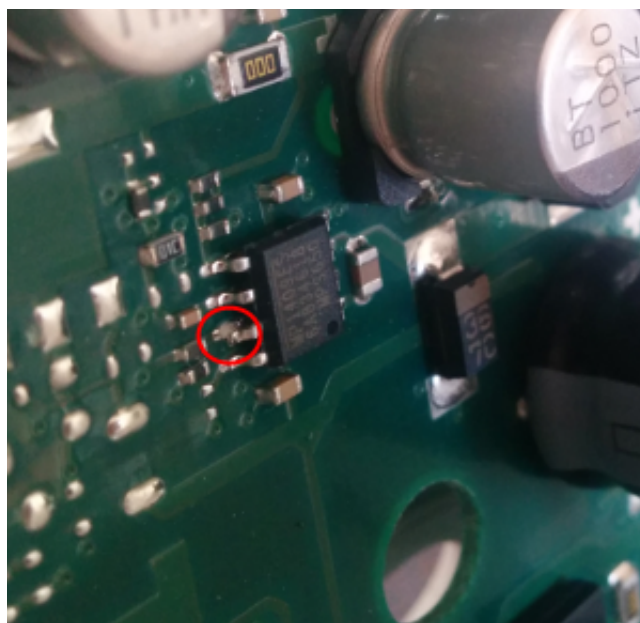
- *openwrt-lantiq-gigaset-sx76x-nor-u-boot.img* (u-boot.bin)
- *openwrt-lantiq-xway-GIGASX76X-squashfs.image* (fw.img)

Poté, co je úspěšně aktualizován zavaděč, je možné nahrát do zařízení zkompilovaný obraz (*openwrt-lantiq-xway-GIGASX76X-squashfs.image*). Po zapnutí zařízení je nutné zastavit zavádění OS pomocí klávesy *space*, tím se dostaneme do interaktivní konzole zavaděče, umožňující zavedení firmware:

```
u-boot# tftp sx762.img
u-boot# protect off $kernel_addr + $filesize
u-boot# erase $kernel_addr + $filesize
u-boot# cp.b $fileaddr $kernel_addr $filesize
```

2.3.3 Úprava HW pro napájení portu

USB sběrnice je napájena pomocí integrovaného obvodu 11 (seznam obvodů je v tabulce 2.1). Firmware výrobce nastavuje na vstupně-výstupní bráně 7 tohoto integrovaného obvodu napájení o hodnotě 3,3 V přibližně po dobu 2 vteřin. Poté je tato vstupně-výstupní brána uzemněna. Operační systém OpenWrt verze 12.07 ale stejné chování nevykazuje a integrovaný obvod zůstává neaktivní po celou dobu. Toto chování lze obejít odpojením vstupně-výstupní brány 7, čímž se dosáhne jeho neustálého napájení USB zařízení.



Obr. 2.6: Úprava HW pro zachování napájení USB

Novější verze OpenWrt tímto neduhem již netrpí díky následující opravě ⁵:

```
68      writel(readl(DANUBE_PMU_PWDCR) & ~0x8000, DANUBE_PMU_PWDCR);  
69 +    writel(readl(IFXMIPS_GPIO_P1_OUT) | (1<<13), IFXMIPS_GPIO_P1_OUT);  
70      #if 1    //defined (DWC_HOST_ONLY)
```

2.3.4 USB modem

Modem byl vybrán tak, aby byl kompatibilní s chan dongle⁶ a jeho provoz nevyžadoval SW usb_modeswitch. Další kompatibilní USB modemy s chan_dongle jsou uvedeny v tabulce 2.5 níže⁷. Pro správnou funkčnost USB Modemu (jednoduché USB zařízení) je potřeba do systému OpenWrt nainstalovat následující balíčky:

⁵Patch https://dev.wlan-si.net/attachment/ticket/1073/sx763_aa_usbpower.patch

⁶Rozšíření PBX Asterisk

⁷Aktualizovaný seznam je možné navštívit na <http://wiki.e1550.mobi/doku.php?id=requirements>

```

root@VoIPgw:~# opkg update
root@VoIPgw:~# opkg install comgt kmod-usb-serial kmod-usb-serial-option kmod-usb-serial-wwan

```

Po úspěšné instalaci a připojení modemu do USB portu by se mělo ve výpisu `dmesg`⁸ zobrazit rozpoznání zařízení a jeho inicializace jako GSM zařízení:

```

USB Serial support registered for generic
usbserial_generic 1-1:1.0: generic converter detected
USB Serial support registered for generic
usbserial_generic 1-1:1.0: generic converter detected
usb 1-1: generic converter now attached to ttyUSB0
usbserial_generic 1-1:1.1: generic converter detected
usb 1-1: generic converter now attached to ttyUSB1
...
usbcore: registered new interface driver usbserial_generic
usbserial: USB Serial Driver core
USB Serial support registered for GSM modem (1-port)
usbcore: registered new interface driver option
option: v0.7.2:USB Driver for GSM modems

```

Pokud je modem složené zařízení, je potřeba použít program `usb_modeswitch` pro přepnutí. Tento postup ale popsán nebude, protože vybraný modem takto „přepínat“ není potřeba. Specifikace parametrů použitého modemu je v tabulce 2.3.

Tab. 2.3: Specifikace použitého modemu

Výrobce	Huawei
Model	K3765
Rozhraní	USB 2.0 (A plug)
Chipset	Qualcomm MSM6290
GSM frekvence	850, 900, 1800, 1900
UMTS frekvence	900, 2100
HSDPA	7,2 MBps
HSUPA	5,76 MBps
EDGE	236,8 kBps
GPRS	57,6 kBps
Rozhraní pro externí antenu	Ano-nepřístupná
Typ konektoru	CRC9
Podpora hlasových přenosů	Ano
NAND-flashmemory	Ano
Slot pro microSD kartu	Ano (až 8 GB)

⁸Rídící zprávy systému od posledního restartu

2.3.5 Systémový oddíl na externím zařízení

Rozdělení FLASH⁹ paměti zařízení, definované výrobcem, je popsáno v tabulce 2.4. Jak je z popisu patrné, pro samostatný operační systém je vyhrazen oddíl „mtd3“ o velikosti 6208 KB. Tato velikost postačuje pro instalaci základního operačního systému (cca 3-4 MB) a několika programů či ovladačů. Pro účely této práce bylo nutné nainstalovat základní operační systém a také balíčky s ovladači pro jednotlivé subsystémy (např. TAPI rozhraní, PBX Asterisk, Prosody server), díky čemuž hrozí, že na zařízení již nebude volné místo. Toto omezení se dá vyřešit vhodnou konfigurací, připojením externího zařízení a vytvořením systémového oddílu¹⁰ na tomto zařízení.

Tab. 2.4: Tabulka popisující rozložení FLASH paměti

oddíl	začátek	velikost	popis
mtd0	0x00000000	56 kB	první zavaděč
mtd1	0x0000e000	8 kB	nvrám
mtd2	0x00010000	7104 kB	sekundární zavaděč
mtd3	0x000f0000	6208 kB	systémový oddíl (squashfs)
mtd4	0x00700000	128 kB	oops
mtd5	0x00720000	896 kB	PSA (konfigurace)

Balíčky nutné pro přesun systémového oddílu na externí zařízení se souborovým systémem EXT4¹¹ jsou:

- **block-mount** - balíček skriptů pro připojení a kontrolu blokových zařízení
- **kmod-usb-storage** - modul jádra pro podporu protokolu USB Mass Storage devices
- **kmod-fs-ext4** - jaderný modul pro podporu EXT4 souborového systému

Konfigurace systému pro systémový oddíl na externím zařízení:

```
root@VoIPgw:~# cat /etc/config/fstab
config 'swap'
    option device      'sdb1'
    option label       'swap'

config mount
    option target      /overlay
    option device      /dev/sdb2
    option fstype      ext4
    option options     rw, sync
    option enabled     1
```

⁹Flash paměť je elektricky programovatelná paměť vnitřně organizovaná do bloků, který lze každý programovat samostatně.

¹⁰přípojný bod /

¹¹Žurnálovací souborový systém EXT4 je vhodný pro USB Flash disk

```

option enabled_fsck 0

config mount
option target      /overlay-boot
option device      /dev/mtdblock3
option fstype      jffs2
option options     rw, sync
option enabled     1
option enabled_fsck 0

```

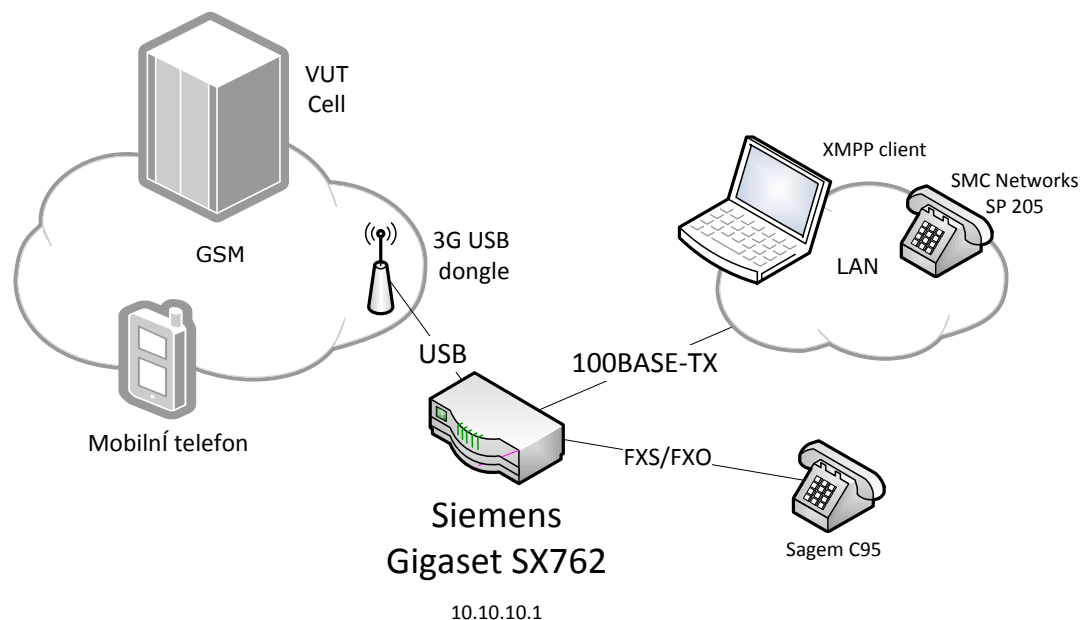
Systémový daemon pro obsluhu připojitelných zařízení se povolí pomocí příkazu:

```
root@VoIPgw:~# /etc/init.d/fstab enable
```

```
root@VoIPgw:~# df -h
```

Filesystem	Size	Used	Available	Use%	Mounted on
rootfs	3.6G	28.2M	3.4G	1%	/
/dev/root	2.5M	2.5M	0	100%	/rom
tmpfs	13.8M	96.0K	13.7M	1%	/tmp
/dev/sdb2	3.6G	28.2M	3.4G	1%	/overlay
overlayfs:/overlay	3.6G	28.2M	3.4G	1%	/
tmpfs	512.0K	0	512.0K	0%	/dev

2.4 Uvažovaná topologie



Obr. 2.7: Uvažovaná topologie

2.4.1 Konfigurace LAN a WAN rozhraní směrovače

Protože se jedná o A/VDSL router (a předpokládá se využívání na A/VDSL linkách), WAN rozhraní je v základním systému definováno na ADSL portu. Uvažované řešení ale WAN port definuje na jednom z ethernet rozhraní (na obrázku 2.1 označen jako „LAN/WAN“). Úprava systému:

```
root@VoIPgw:~# cat /etc/config/network
config interface 'lan'
    option type 'bridge'
    option proto 'static'
    option ipaddr '10.10.10.1'
    option netmask '255.255.255.0'
    option ifname 'eth0.1'

config switch
    option name 'eth0'
    option reset '1'
    option enable_vlan '1'

config switch_vlan
    option device 'eth0'
    option vlan '1'
    option ports '4 3 2 5t'

config switch_vlan
    option device 'eth0'
    option vlan '2'
    option ports '1 5t'

config interface 'wan'
    option ifname 'eth0.2'
    option proto 'dhcp'
```

Aby bylo možné se k zařízení přihlásit pomocí SSH klienta, je potřeba upravit firewall¹² pravidla pro propuštění SSH provozu na nově definované WAN rozhraní:

```
/etc/config/firewall
config rule
    option name 'Povolit SSH, WAN' ; název, popis
    option src wan ; zdrojové rozhraní
    option proto tcp ; protokol
    option dest_port 22 ; cílový port
    option target ACCEPT ; akce – povolit
```

Obdobně je potřeba upravit také pravidla pro WAN rozhraní a SIP provoz, pokud bude požadavek na hlasovou komunikaci přes WAN rozhraní:

```
/etc/config/firewall
config rule
    option name 'Povolit SIP, UDP' ; název, popis
    option src wan ; zdrojové rozhraní
    option proto udp ; protokol
    option dest_port 5060 ; cílový port
    option target ACCEPT ; akce – povolit
```

¹²iptables


```

config rule
  option name      'Povolit SIP, TCP' ; název , popis
  option src       wan                 ; zdrojové rozhraní
  option proto     tcp                 ; protokol
  option dest_port 5060                ; cílový port
  option target    ACCEPT              ; akce – povolit

```

Zobrazení aktuální konfigurace SW přepínače (ADM6996M):

```

root@VoIPgw:~# swconfig dev switch0 show

switch0: eth0(ADM6996M), ports: 6 (cpu @ 5), vlans: 16
  —switch
    Attribute 1 (int): enable_vlan (Enable VLANs)
    Attribute 2 (none): apply (Activate changes in the hardware)
    Attribute 3 (none): reset (Reset the switch)
  —vlan
    Attribute 1 (int): vid (VLAN ID)
    Attribute 2 (ports): ports (VLAN port mapping)
  —port
    Attribute 1 (int): pvid (Primary VLAN ID)

```

2.5 Pobočková ústředna Asterisk

Dostupný software pro implementaci hlasové komunikace podporovaný systémem OpenWRT pro zařízení využívající integrovaného obvodu Danube je:

- owsip - jednoduchá implementace SIP daemonu s podporou FXS portů
- FreeSWITCH - software pro hlasovou komunikaci s licencí MPL¹³
- Asterisk - software pro hlasovou komunikaci s otevřeným zdrojovým kódem.

Pro samotnou práci byla vybrána PBX Asterisk, protože umožňuje použít rozšíření chan_dongle pro komunikaci s USB modemem.

Seznam balíčků nutných pro běh PBX Asterisk verze 1.8 a jednotlivých balíčků pro uvažovaný systém:

```

root@VoIPgw:~# opkg update
root@VoIPgw:~# opkg install asterisk18 asterisk18-chan-gtalk asterisk18-chan-lantiq
                        asterisk18-chan-local asterisk18-func-channel asterisk18-func-global
                        libopenssl libncurses libpopt zlib libiconv-full libiksemel

```

chan_lantiq

chan_lantiq je rozšíření pro PBX Asterisk umožňující přístup k TAPI rozhraní na zařízení s architekturou Lantiq. Ve výpisu dmesg je možné nalézt následující varování: „WARNING: FW version 12.1.0 too old. Minimum required FW version is

¹³Program pod MPL licencí lze kombinovat s nesvobodným softwarem, pouze převzatá část musí nadále splňovat podmínky licence MPL. (Při začlenění cizí tvorby uveřejněné pod GPL je nutno uvolnit pod GPL celý program.)

12.4.0“ - verze 12.1.0 je poslední funkční firmware TAPI rozhraní, novější již nemá funkce, které jsou využívány.

kmod-ltq-vmmc

kmod-ltq-vmmc je jaderný modul zpřístupňující hlasový subsystém.

chan_dongle

chan_dongle je rozšíření pro Asterisk umožňující připojení a ovládání Huawei UMTS USB modemů. Zpřístupňuje následující funkce:

- Hlasové hovory, vytáčení pomocí názvu modemu, skupiny, názvu poskytovatele, IMEI nebo IMSI.
- Podržení hovoru
- Hlasovou konferenci
- Zaslání a příjem SMS (znaková sada latinka, podpora víceřádkového textu)
- Zaslání a příjem USSD (znaková sada latinka, podpora víceřádkového textu)
- Zaslání a příjem DTMF (mimo A,B,C,D které nejsou podporované výrobcem Huawei)

OpenWrt verze 12.09 (attitude adjustment) standardně neobsahuje chan_dongle, proto je potřeba jej zkompilevat spolu s ostatními balíčky standardním procesem kompilace. K tomu využijeme *Makefile*, který řídí proces kompilace balíčku. Vytvořený *Makefile* je dostupný v příloze A

2.5.1 Nastavení SIP

Konfigurace všech extension¹⁴ využívající protokol SIP je v souboru *sip.conf*.

```
[general]
context=default          ; výchozí kontext
bindport=5060            ; port pro komunikaci
videosupport=yes         ; umožnit video hovory

[202]                    ; klapka 202
type=friend              ;
secret=heslo2            ; heslo
username=202             ; uživatelské jméno
callerid="Uživatel_202" <202>
host=dynamic             ; nevázat na IP adresu
qualify=yes              ; testování dostupnosti
disallow=all             ; zakáže všechny kodeky
allow=alaw                ; povolí alaw
```

Při každé změně v souboru *sip.conf* je potřeba nahrát novou konfiguraci pomocí:

¹⁴Pojem extension lze vysvětlit jako přípojku k zařízení (telefon / SW klient) a také jako pátevní spoj Trunk

Tab. 2.5: Tabulka některých kompatibilních zařízení

Model	Hlasová komunikace	SMS	USSD
E153	Ano	Ano	Ano
E1550	Ano	Ano	Ano
E1552	Ano	Ano	Ano
E156	Ano	Ano	Ano
E160	Ano	Ano	Ano
E1612	Ne	Ano	Ano
E169	Ano	Ano	Ano
E1692	Ano	Ano	Ano
E171	Ano		Ano
E175X	Ano	Ano	Ano
E180	Ano	Ano	Ano
E220	Ne	Ano	Ano
E270	Ne	Ano	Ano
K3520	Ano	Ano	Ano
K3715	Ano	Ano	Ano
K3765	Ano	Ano	Ano

```
VoIPgw*CLI> sip reload
```

Tab. 2.6: Základní nastavení SIP účtů použitých v této práci

Klapka	Uživatel	Heslo
101	101	tajne_heslo
202	202	heslo2

2.5.2 Nastavení chan_lantiq

Nastavení TAPI rozhraní je prováděno v souboru *lantiq.conf*

```
[interfaces]
channels = 2          ; Počet FXS portu

[general]
echocancel=es        ; potlačení ozvěny pomocí metody Echo suppression
```

2.5.3 Nastavení res_jabber

Nastavení rozšíření protokolu XMPP pro PBX Asterisk se provádí v konfiguračním souboru *jabber.conf*. Pro nastavení hlasových hovorů slouží *gtalk.conf*.

```
root@VoIPgw:~# cat /etc/asterisk/jabber.conf
[general]
debug=no ; Povolení ladících informací

[asterisk]
type=client ; Typ Client nebo Component
serverhost=voipgw ; Název serveru
username=asterisk@voipgw ; JID
secret=asterisk ; Heslo
priority=1 ; Priorita
port=5222 ; Port serveru 5222
usetls=yes ; Použití tls
usesasl=yes ; Použití sasl
buddy=radek@voipgw ; Výchozí uživatel
distribute_events=yes ; Zasílat události
status=available ; Status: chat, available, away, xaway, dnd
..
statusmessage=Jsem tu abych sloužil ; Popis statusu
;timeout=5 ; Timeout

root@VoIPgw:~# cat /etc/asterisk/gtalk.conf
[general]
context=asterisk_call ; Výchozí kontext
allowguest=yes ; Povolit hovory od uživatelů mimo seznam účtu

[guest] ; nastavení neznámých příchozích volání
disallow=all ; zakázat všechny kodeky
allow=ulaw ; povolit pouze ulaw kodek
context=asterisk_call ; Výchozí kontext

[asterisk_call]
username=asterisk@voipgw ; uživatelský účet přes který se volá a přijímá
disallow=all ; zakázat všechny kodeky
allow=ulaw ; povolit pouze ulaw kodek
context=asterisk_call ; Výchozí kontext
connection=asterisk ; název componenty v jabber.conf pro spojení
```

Při každé změně v souboru *jabber.conf* nebo *gtalk.conf* je potřeba nahrát novou konfiguraci pomocí:

```
VoIPgw*CLI> jabber reload
```

2.5.4 extensions.conf

V tomto konfiguračním souboru je definován číslovací plán ústředny. Číslovací plány jsou součástí kontextů. Kontext s označením *[default]* je v této práci považován jako hlavní. Jak je z konfigurace patrné, číslovací plán obsahuje více kontextů a pravidel pro čísla:

- **[default]** - hlavní kontext s vloženými kontexty pro SMS a USSD
 - extend 101 - obsluha volaného čísla 101 - SIP účet
 - extend 202 - obsluha volaného čísla 202 - XMPP účet
 - extend _XXXXXXXX - obsluha volaného čísla s délkou 9 číslic
- **[dongle-incoming-sms]** - kontext pro obsluhu příchozích SMS zpráv
- **[dongle-incoming-ussd]** - kontext pro obsluhu příchozích USSD zpráv
- **[asterisk_call]** - kontext pro obsluhu příchozích XMPP hovorů

```
[general]
usercontext=default

[globals]

[default]
include => dongle-incoming-sms
include => dongle-incoming-ussd

exten => 101,1,JabberSend(asterisk,dialplan@voipgw,hovor ${CALLERID(all)} pro ${
  EXTEN} ${STRFTIME(${EPOCH},,%Y-%m-%d %H:%M:%S)})
exten => 101,2,Dial(SIP/101)
exten => 101,n,Hangup()

exten => 202,1,JabberSend(asterisk,dialplan@voipgw,hovor ${CALLERID(all)} pro ${
  EXTEN} ${STRFTIME(${EPOCH},,%Y-%m-%d %H:%M:%S)})
exten => 202,n,Dial(gtalk/asterisk/dialplan@voipgw)
exten => 202,n,Hangup()

; volani do GSM site pomoci 9mistneho cisla
exten => _XXXXXXXX,1,JabberSend(asterisk,dialplan@voipgw,Odchozi volani do GSM
  site pro ${EXTEN})
exten => _XXXXXXXX,n,Dial(Dongle/dongle0/+420${EXTEN})
exten => _XXXXXXXX,n,Hangup()

[dongle-incoming-sms]
; prichozi SMS posli pomoci jabber konfigurace asterisk, na JID radek@starky.eu
exten => sms,1,JabberSend(asterisk,dialplan@voipgw,Prichozi SMS, prijato ${STRFTIME
  (${EPOCH},,%Y-%m-%d %H:%M:%S)} - zarizeni - ${DONGLENAME} - od - ${CALLERID(num
  )}: ${BASE64_DECODE(${SMS_BASE
  64})}).)
exten => sms,n,Hangup()

[dongle-incoming-ussd]
; prichozi USSD posli pomoci jabber konfigurace asterisk, na JID radek@starky.eu
exten => ussd,1,JabberSend(asterisk,dialplan@voipgw, '${STRFTIME(${EPOCH},,%Y-%m-%d
  %H:%M:%S)} - ${DONGLENAME}: ${BASE64_DECODE(${USSD_BASE64})}' )
exten => ussd,n,Hangup()

[asterisk_call]
exten => s,1,JabberSend(asterisk,dialplan@voipgw,hovor ${CALLERID(all)} pro ${EXTEN
  } ${STRFTIME(${EPOCH},,%Y-%m-%d %H:%M:%S)})
exten => s,n,Dial(SIP/202)[default]
```

Při každé změně v souboru *extensions.conf* je potřeba nahrát novou konfiguraci číslovacího plánu pomocí:

```
VoIPgw*CLI> dialplan reload
```

2.5.5 Některé použité funkce a aplikace dialplanu

- **include** - umožňuje rozdělit dialplan na menší kousky (kontexty). Nejčastěji se tato funkce používá pro zabezpečení hranic mezi různými třídami volajících.
- **Dial** - funkce se pokusí o připojení k jinému zařízení nebo koncovému bodu a překlenout volání
 - Dongle - koncové zařízení, GSM rozhraní
 - SIP - koncový bod pro SIP účastníky
 - Gtalk - koncový bod pro XMPP účastníky
 - TAPI - koncové zařízení pro analogové rozhraní
- **JabberSend** - umožňuje zasílání zpráv pomocí XMPP protokolu
- **Hangup** - aplikace pro ukončení hovoru na daném kanálu
- **Set** - umožňuje přiřazení hodnot k proměnným
- **Goto** - nepodmíněný skok na jinou část dialplanu (priorita, rozšíření, nebo kontext)
- **Base64_Decode** - provádí převod z řetězce kódovaného pomocí base64¹⁵ (návratová hodnota je prostý text)

2.5.6 Reakce na příchozí SIP/XMPP zprávu

Podpora zpracování příchozích SIP a XMPP zpráv, mimo probíhající hovor, byla přidána do PBX Asterisk ve verzi 10. Následně, ve verzi 11, byli provedeny úpravy týkající jádra aplikace a také některých rozšíření a funkcí. Mezi hlavní novinky verze 11 patří přidání podpory protokolu WebRTC¹⁶, vznik nového rozšíření *Motif* podporujícího XMPP Jingle¹⁷, přepsání rozšíření *res_jabber* a jeho přejmenování na *res_xmpp*[23].

Pro umožnění zpracování zpráv je potřeba tuto funkcionalitu povolit v následujících konfiguračních souborech:

- sip.conf
- xmpp.conf

¹⁵Base64 je kódování, které převádí binární data na posloupnosti tisknutelných znaků. Umožňuje přenos binárních dat kanály, které dovolují pouze přenos textů.

¹⁶WebRTC je webový standard, jehož zakladatelem je společnost Google. Tento standard umožňuje real-time komunikaci ve webovém prohlížeči bez nutnosti instalace přídatných modulů jako Flash nebo Java.

¹⁷včetně Google Talk

Následně je potřeba specifikovat akce v číslovacím plánu, které budou tuto událost obsluhovat.

Úprava souboru sip.conf

V sekci *[general]* je nutné umožnit zpracovávání SIP zpráv (parametr *accept_outofcall_message*) a definovat kontext, do kterého se bude zpráva směřovat (parametr *outofcall_message_context*). Jako užitečné se jeví nastavení parametru *auth_message_requests* na hodnotu „no“, který zamezí autentizaci SIP metody MESSAGE - implementace RFC 3428[20] na koncových zařízeních nemusí být úplná.

```
[general]
accept_outofcall_message=yes      ; zpracování zpráv povoleno
outofcall_message_context=sipim   ; kontext do kterého bude směřována zpráva
auth_message_requests = no       ; vypnutí požadavku na autentizaci
```

Úprava souboru xmpp.conf

Obdobně jako u *sip.conf* je nutné povolit zpracování XMPP zpráv (parametr *sendtodialplan*) a specifikovat kontext, do kterého se bude zpráva směřovat (parametr *context*).

```
sendtodialplan=yes ; Vyvolat událost v číslovacím plánu. Standardně vypnuto
context=xmppim     ; kontext do kterého bude směřována zpráva
```

Úprava souboru extensions.conf

Číslovací plán pro kontext *[xmppim]*, který se vyvolá při příchozí XMPP zprávě, může vypadat například následovně:

```
[xmppim]
; default kontext pro příjem XMPP zpráv

exten => s,1,NoOp(XMPP Message)
same => n,Set(XMPPRESOURCE=asterisk)
; uložit zprávu, číslo
same => n,Set(MSG=${JABBER_RECEIVE(asterisk,${MESSAGE(from):5}, 15)})
same => n,Set(NumberToSend=${FILTER(0-9, ${MSG})})
same => n,Set(Text=${MSG:10})

same => n,GotoIf(${REGEX("0-9" ${NumberToSend})} != 1)?KO_1
same => n,JabberSend(${XMPPRESOURCE},${MESSAGE(from):5}, Ok, budem psat na číslo: ${
    FILTER(0-9,${NumberToSend})}. Text zprávy bude "${Text}" )

same => n,DongleSendSMS(dongle0,+420${FILTER(0-9,${NumberToSend})},${Text})

same => n,Goto(end)
same => n(KO_1),JabberSend(${XMPPRESOURCE},${MESSAGE(from):5}, Ahoj, prikaz je ve
    formatu XXXYYYXXYY,Text zprávy a Enter. Například "777000110,□Text□SMS")

same => n(end),NoOp(XMPP Message end)
```

2.6 Analogový telefon

Pro účely testování analogového rozhraní směrovače byl zapůjčen analogový telefon značky Sagem, model C95. Tento telefonní přístroj nevyžaduje dodatečné napájení (je napájen z FXS rozhraní). Telefon je vyobrazen na obrázku 2.8. Mezi přednosti tohoto přístroje patří:

- malé rozměry
- zobrazení čísla volajícího (funkce CLIP)
- seznam volaných (20 míst) a přijatých hovorů (16 míst)
- indikace zmeškaných hovorů
- vizuální indikace zvonění pomocí LED diody



Obr. 2.8: Sagem C95

2.7 SIP telefon

Pro účely testování hlasových přenosů byl zapůjčen SIP telefon značky SMC Networks, model SMCDSP-205. Konfigurace tohoto přístroje může probíhat pomocí směrových šipek a menu přímo na telefonu, nebo pomocí webového rozhraní, které naslouchá na portu 9999 a poskytuje pokročilé možnosti [22]. Parametry použité pro nastavení účtu jsou v tabulce 2.6.



Obr. 2.9: SMC Networks SMCDSP-205

2.8 SW Prosody

Jedná se o rychlý XMPP server psaný pomocí jazyku lua s nízkými požadavky na zdroje HW. Instalace a spuštění serveru prosody v OpenWRT:

```
root@VoIPgw:~# opkg update
root@VoIPgw:~# opkg install liblua lua luasocket luafilesystem libidn libexpat
luaexpat luasec prosody
root@OpenWrt:~# /etc/init.d/prosody start
Started
```

Balíček pro prosody je v OpenWrt verze 12.09 nedodělaný. Po úspěšné instalaci je potřeba nastavit oprávnění pro konfigurační soubory a adresáře serveru. Pokud se při spuštění objeví chyba „Unable to store, perhaps you don't have permission?“ je potřeba nastavit navíc oprávnění také pro podsložky s konfigurací:

```
root@VoIPgw:/etc/prosody# chmod 755 /etc/prosody/prosody.cfg.lua
root@VoIPgw:/etc/prosody# chown prosody:prosody $(find -type d)
```

následně by se měl prosody server úspěšně spustit:

```
root@OpenWrt:~# /etc/init.d/prosody start
Started
```

přidání uživatele pro server prosody pomocí *prosodyctl*

```
root@VoIPgw:~# prosodyctl adduser
Usage: /usr/bin/prosodyctl adduser JID
Create the specified user account in Prosody
root@VoIPgw:~# prosodyctl adduser asterisk@voipgw
Enter new password: 1234
Retype new password: 1234
```

V semestrální práci je využito dvou XMPP klientů (účtů). První účet (*asterisk@VoIPgw*) je využit pro propojení PBX Asterisk a XMPP serveru Prosody. Druhý účet je určen pro příjemce informací. Základní parametry nutné pro připojení jsou uvedeny v tabulce 2.7.

Tab. 2.7: Nastavení uživatelů XMPP serveru

uživatel	heslo	doména	poznámka
asterisk	asterisk	VoIPgw	účet pro Asterisk, ze kterého se posílají informace
dialplan	dialplan	VoIPgw	účet na který se přeposílají informace

2.9 Možné směry volání

Rozličné druhy koncových zařízení mohou vyžadovat různé způsoby kódování nebo řízení. Řešením nekompatibility jednotlivých koncových zařízení je právě ústředna Asterisk, která pomocí vhodných rozšíření obstarává vzájemnou komunikaci. V tabulce 2.8 je uvedena matice koncových zařízení a jejich možnost spojit se s protistranou pomocí navrhovaného řešení¹⁸:

Uvažované řešení umožňuje také přenášet krátké zprávy. Jednotlivé protokoly obvykle obsahují funkce umožňující přenos zpráv¹⁹. Následující tabulka 2.9 obsahuje koncové body a jejich možnosti příjmu/odeslání zpráv:

¹⁸Prázdné místo znamená, že zařízení se nepodílí na komunikaci.

¹⁹V literatuře často označováno jako IM - Instant messaging

Tab. 2.8: Tabulka popisující možnosti volání

Směr volání	SW VoIP (SIP) klient	SW XMPP klient	GSM Síť	HW Tel (TAPI, analog)	HW Tel (SIP, digital)
SW VoIP (SIP) klient	úspěšně spojený hovor	úspěšně spojený hovor	úspěšně spojený hovor	úspěšně spojený hovor	úspěšně spojený hovor
SW XMPP klient	úspěšně spojený hovor	netestováno, bez známých překážek	netestováno, bez známých překážek	netestováno, bez známých překážek	netestováno, bez známých překážek
GSM Síť	úspěšně spojený hovor	netestováno, bez známých překážek		úspěšně spojený hovor	úspěšně spojený hovor
HW Tel (TAPI, analog)	úspěšně spojený hovor	netestováno, bez známých překážek	úspěšně spojený hovor	netestováno, bez známých překážek	úspěšně spojený hovor
HW Tel (SIP, digital)	úspěšně spojený hovor	netestováno, bez známých překážek	úspěšně spojený hovor	úspěšně spojený hovor	úspěšně spojený hovor

Tab. 2.9: Tabulka popisující možnosti přenosu zpráv

Směr zprávy	SIP IM	GSM SMS	GSM USSD	XMPP IM
SIP IM	otestováno	možné s Asterisk 11	v práci neřešeno	možné s Asterisk 11
GSM SMS	možné s Asterisk 11		v práci neřešeno	otestováno
GSM USSD	v práci neřešeno	v práci neřešeno		neověřeno
XMPP IM	možné s Asterisk 11	otestováno	v práci neřešeno	otestováno

2.10 Zpracování příchozí SMS

Pro ověření funkčnosti zpracování SMS hlasovou bránou byla zaslána SMS zpráva z mobilního telefonu (obrázek C.1) na GSM číslo SIM karty modemu. Příchozí SMS je nejprve zpracována modemem, následně je posílána zpráva pomocí rozšíření `chan_dongle` do PBX Asterisk (obrázek C.3), kde se v závislosti na konfiguraci (`dialplan.conf`) provede příslušná akce. V této konfiguraci je nastaveno přeposílání SMS zpráv na XMPP účet `dialplan@voipgw` C.2, kde se zasílají také ostatní informace, například při zpracovávání hovorů.

3 ZÁVĚR

Uvedené zařízení Siemens Gigaset SX762 umožňuje provoz operačního systému OpenWrt. Na tomto operačním systému byla zprovozněna pobočková ústředna Asterisk a XMPP server Prosody. V práci byly popsány protokoly podílející se na komunikaci, jakým způsobem se kompiluje operační systém OpenWrt a vytváří *Makefile* soubor, řídící kompilaci vlastních balíčků. Detailně byly popsány kroky umožňující náhradu firmware výrobce. Díky popsané konfiguraci jednotlivých částí je možné spojit hovory z GSM sítě, SW/HW SIP klienta nebo XMPP účtu. Uvedená konfigurace umožňuje také příjem SMS či USSD zpráv a její následné transformování na XMPP zprávu. Byli prozkoumány a popsány možnosti zpracování XMPP zpráv poskytující PBX Asterisk verze 11. Byla vytvořena laboratorní úloha pro předmět Vysokorychlostní komunikační systémy (VKS) pro seznámení posluchačů s možnostmi komunikace pomocí nestandardních nástrojů.

LITERATURA

- [1] KAPOUN, V. *Digitální ústředny*. 1. vyd. Brno: FEI VUT, 1998, 54 s. ISBN 80-720-4089-8.
- [2] BAZALA, D. *Telekomunikace*. 1. vyd. Praha: BEN - technická literatura, 2006, 222 s. ISBN 80-730-0201-9.
- [3] PETERKA, J. *Transportní vrstva*. EArchiv.cz [online]. 1993 [cit. 1. 12. 2014]. Dostupné z URL: <<http://www.earchiv.cz/a92/a224c110.php3>>.
- [4] PETERKA, J. *Protokol TCP*. EArchiv.cz [online]. 1993 [cit. 1. 12. 2014]. Dostupné z URL: <<http://www.earchiv.cz/a93/a305c110.php3>>.
- [5] PETERKA, J. *Protokol UDP*. EArchiv.cz [online]. 1993 [cit. 1. 12. 2014]. Dostupné z URL: <<http://www.earchiv.cz/a93/a303c110.php3>>.
- [6] The Buildroot developers *Buildroot about* [online]. 2014, poslední aktualizace 31. 10 2014 [cit. 1. 12. 2014]. Dostupné z URL: <<http://buildroot.uclibc.org/about.html>>.
- [7] LOREMA: *About OpenWRT*. OpenWRT Wiki [online]. Poslední aktualizace 13. 9. 2013 [cit. 1. 12. 2014]. Dostupné z URL: <<http://wiki.openwrt.org/about/start>>.
- [8] SCHULZRINNE, H.; CASNER, S.; FREDERICK, Ron Frederick. *RTP: A Transport Protocol for Real-Time Applications*. Internet Engineering Task Force [online]. 2003 [cit. 1. 12. 2014]. Dostupné z URL: <<http://www.ietf.org/rfc/rfc3550.txt>>.
- [9] HANDLEY, M. ; JACOBSON, V. ; PERKINS, C. *SDP: Session Description Protocol*. The Internet Engineering Task Force [online]. 2006 [cit. 1. 12. 2014]. Dostupné z URL: <<http://tools.ietf.org/html/rfc4566>>.
- [10] SCHULZRINNE, H. ; RAO, A. ; LANPHIER, R. *Real Time Streaming Protocol (RTSP)*. Internet Engineering Task Force [online]. 1998 [cit. 1. 12. 2014]. Dostupné z URL: <<http://www.ietf.org/rfc/rfc2326.txt>>.
- [11] SAINT-ANDRE, P. *Extensible Messaging and Presence Protocol (XMPP): Core*. Jabber Software Foundation [online]. 2004 [cit. 1. 12. 2014]. Dostupné z URL: <<http://www.ietf.org/rfc/rfc3920.txt>>.
- [12] SAINT-ANDRE, P. *Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence*. Jabber Software Foundation [online]. 2004

- [cit. 1. 12. 2014]. Dostupné z URL: <<http://www.ietf.org/rfc/rfc3921.txt>>.
- [13] ROSENBERG, J. ; SCHULZRINNE, H. ; CAMARILLO, G. ; JOHNSTON, A. ; PETERSON, J. ; SPARKS, R. ; HANDLEY, M. ; SCHOOLER, E. *SIP: Session Initiation Protocol*. Internet Engineering Task Force [online]. 2002 [cit. 1. 12. 2014]. Dostupné z URL: <<http://tools.ietf.org/html/rfc3261>>.
- [14] HANDLEY, M. ; PERKINS, C. ; WHELAN, E. *Session Announcement Protocol*. The Internet Engineering Task Force [online]. 2000 [cit. 1. 12. 2014]. Dostupné z URL: <<http://tools.ietf.org/html/rfc2974>>.
- [15] *Wireshark Documentation*. Wireshark [online]. 2011 [cit. 1. 12. 2014]. Dostupné z URL: <<http://www.wireshark.org/docs/>>.
- [16] *Table of Ports for Internet Services*. TCP/IP Ports [online]. 2007 [cit. 1. 12. 2014]. Dostupné z URL: <<http://www.chebucto.ns.ca/rakerman/port-table.html>>.
- [17] Microsoft Corporation: *TAPI Versioning*. MSDN Microsoft [online]. [cit. 1. 12. 2014]. Dostupné z URL: <[http://msdn.microsoft.com/en-us/library/ms737251\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/ms737251(v=vs.85).aspx)>.
- [18] CANOV, M. *Antonio Santi Giuseppe Meucci* [online]. Poslední aktualizace 19. 03. 2005 [cit. 1. 12. 2014]. Dostupné z URL: <<http://canov.jergym.cz/objevite/objev4/meu.htm>>.
- [19] *Telefon*. Magazín Quido [online]. [cit. 1. 12. 2014]. Dostupné z URL: <<http://www.quido.cz/objevy/telefon.htm>>.
- [20] SCHULZRINNE, H. ; CAMPBELL B., Ed. ; ROSENBERG, J.; dynamicsoft ; Columbia University ; HUITEMA, C. ; GURLE, D. ; Microsoft Corporation: *Session Initiation Protocol (SIP) Extension for Instant Messaging*. Internet Engineering Task Force [online]. 2002 [cit. 1. 12. 2014]. Dostupné z URL: <<http://tools.ietf.org/html/rfc3428>>.
- [21] Gigaset Communication: *Gigaset SX762 WLAN dsl*. [online]. Poslední aktualizace 2008 [cit. 1. 12. 2014]. Dostupné z URL: <http://www.gigaset.com/fileadmin/legacy-assets/SX762_EN_A31008-M709-R121-4x-7619_23-10-2008.pdf>.

- [22] SMC Networks: *SMCDSP-200/SMCDSP-205 Series - VoIP Phone Administration Guide*. [online]. Poslední aktualizace 1/10/2007. [cit. 1. 4. 2015]. Dostupné z URL: <http://www.edge-core.com/temp/edm/old_downloads/mn/MN_SMCDSP-205.pdf>.
- [23] JORDAN, M. *New in 11*. [online]. Poslední aktualizace 2013 [cit. 1. 4. 2015]. Dostupné z URL: <<https://wiki.asterisk.org/wiki/display/AST/New+in+11>>.

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

3G	Třetí generace mobilní sítě
ADSL	Asymmetric Digital Subscriber Line
CLIP	Calling line identification presentation
DC	Direct current
DPS	Deska plošných spojů
DTMF	Dual-tone multi-frequency
EXT4	Fourth Extended Filesystem
FXS	Foreign eXchange Subscriber
gcc	the GNU Compiler Collection
GNU	GNU's Not Unix
GPL	General Public License
GSM	Groupe Spécial Mobil
HTTP	Hypertext Transfer Protocol
HW	Hardware
IMEI	International Mobile Equipment Identity
IMSI	International Mobile Subscriber Identity
IP	Internet Protocol
LAN	Local Area Network
MBIM	Mobile Interface Broadband Model
MPL	Media Gateway Control Protocol
MGCP	Media Gateway Control Protocol
PBX	Private branch exchange
PSTN	Public Switched Telephone Network
QoS	Quality of Service

RFC	Request For Comments
SW	Software
RTCP	Real-Time Control Protocol
RTP	Real-Time Protocol
RTSP	Real-Time Streaming Protocol
SDP	Session Description Protocol
SIP	Session Initiation Protocol
SLIC	Subscriber Line Interface Circuit
SMS	Short message service
SRTCP	Secure Real-time Transport Control Protocol
SSH	Secure Shell
SSRC	Synchronization Source Identifier
SW	Software
TAPI	Telephony Application Programming Interface
TCP	Transmission Control Protocol
TDM	Time Division Multiplex
TFTP	Trivial File Transfer Protocol
UART	Universal asynchronous receiver transmitter
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunication System
USB	Universal Serial Bus
USSD	Unstructured Supplementary Service Data
VoIP	Voice over Internet Protocol
WAN	Wide Area Network
XEP	XMPP Extension Protocol

XML	Extensible Markup Language
XMPP	Extensible Messaging and Presence Protocol

SEZNAM PŘÍLOH

A	Makefile pro chan_dongle	52
B	Makefile pro pcapspidump	54
C	Obrázky zachycující zpracování SMS	55
D	Obsah přiloženého CD	57
E	Laboratorní úloha	58

A MAKEFILE PRO CHAN_DONGLE

```
#
# Copyright (C) 2011 OpenWrt.org
#
# This is free software, licensed under the GNU General Public License v2.
# See /LICENSE for more information.
#

include $(TOPDIR)/rules.mk

PKG_NAME:=asterisk18-chan-dongle
PKG_REV:=10
PKG_VERSION:=1.1.r$(PKG_REV)
PKG_RELEASE:=18

PKG_SOURCE:=chan_dongle-$(PKG_VERSION).tgz
PKG_SOURCE_URL=http://asterisk-chan-dongle.googlecode.com/files/
PKG_MD5SUM:=4ced3ab7094b62be2a0e843a2d4b4111

PKG_SOURCE_SUBDIR:=chan_dongle-$(PKG_VERSION)
PKG_BUILD_DIR=$(BUILD_DIR)/$(PKG_SOURCE_SUBDIR)

WITH_ASTERISK:=asterisk-1.8.10.1

include $(INCLUDE_DIR)/package.mk

define Package/asterisk18-chan-dongle
    SUBMENU:=Telephony
    SECTION:=net
    CATEGORY:=Network
    URL:=https://code.google.com/p/asterisk-chan-dongle/
    MAINTAINER:=Jiri Slachta <slachta@cesnet.cz>
    DEPENDS:= +asterisk18 +libiconv-full
    TITLE:=Huawei UMTS 3G dongle support
endef

define Package/asterisk18-chan-dongle/description
    Asterisk channel driver for Huawei UMTS 3G dongle
endef

MAKE_ARGS:= \
    CC="$(TARGET_CC)" \
    LD="$(TARGET_CC)" \
    CFLAGS="$(TARGET_CFLAGS) -DLOW_MEMORY -D_XOPEN_SOURCE=600 $(TARGET_CPPFLAGS) \
    -I$(STAGING_DIR)/usr/lib/libiconv-full/include -I$(BUILD_DIR)/$(WITH_ASTERISK)/include -DHAVE_CONFIG_H -I. -fPIC" \
    LDFLAGS="$(TARGET_LDFLAGS) -L$(STAGING_DIR)/usr/lib/libiconv-full/lib -liconv" \
    DESTDIR="$(PKG_INSTALL_DIR)/usr/lib/asterisk/modules"

define Build/Configure
    $(call Build/Configure/Default, \
        --with-asterisk=$(BUILD_DIR)/$(WITH_ASTERISK)/include \
        $(MAKE_ARGS) \
    )
endef
```

```

define Build/Compile
    mkdir -p $(PKG_INSTALL_DIR)/usr/lib/asterisk/modules
    $(MAKE) -C "$(PKG_BUILD_DIR)" $(MAKE_ARGS) all install
endef

define Package/asterisk18-chan-dongle/conffiles
/etc/asterisk/dongle.conf
endef

define Package/asterisk18-chan-dongle/install
$(INSTALL_DIR) $(1)/etc/asterisk
$(INSTALL_DATA) $(PKG_BUILD_DIR)/etc/dongle.conf $(1)/etc/asterisk/
$(INSTALL_DIR) $(1)/usr/lib/asterisk/modules
$(INSTALL_BIN) $(PKG_INSTALL_DIR)/usr/lib/asterisk/modules/chan_dongle.so $(1)/usr/lib/asterisk/modules/
endef

$(eval $(call BuildPackage, asterisk18-chan-dongle))

```

B MAKEFILE PRO PCAPSIPDUMP

```
#
# Copyright (C) 2009–2010 OpenWrt.org
#
# This is free software, licensed under the GNU General Public License v2.
# See /LICENSE for more information.
#

include $(TOPDIR)/rules.mk

PKG_NAME:=pcapsipdump
PKG_VERSION:=0.2
PKG_RELEASE:=3

PKG_SOURCE:=$(PKG_NAME)-$(PKG_VERSION).tar.gz
PKG_SOURCE_URL:=http://downloads.sourceforge.net/project/pcapsipdump/pcapsipdump
    /0.2/pcapsipdump-0.2.tar.gz?r=&ts=1412022226&use_mirror=freefr
PKG_MD5SUM:=2c04c54c3478f7cb94978d54538006d2

include $(INCLUDE_DIR)/uclibc++.mk
include $(INCLUDE_DIR)/package.mk

define Package/pcapsipdump
    SECTION:=net
    CATEGORY:=Network
    SUBMENU:=Telephony
    DEPENDS:=+libpcap $(CXX_DEPENDS)
    TITLE:=SIP sessions dumping tool
    URL:=http://sourceforge.net/projects/pcapsipdump/
endef

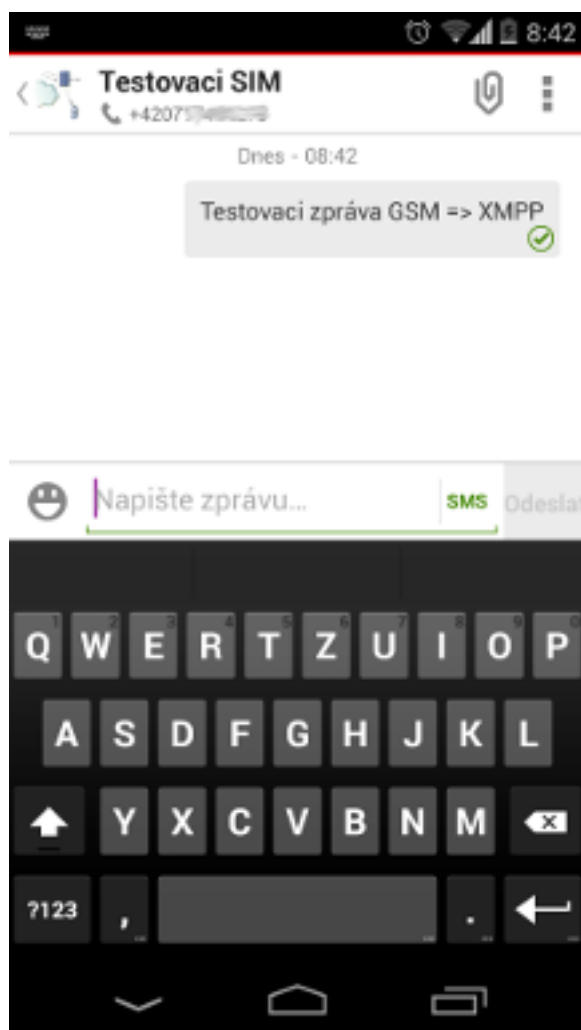
define Package/pcapsipdump/description
    pcapsipdump is a tool for dumping SIP sessions (+RTP traffic, if available)
    to disk in a
    fashion similar to "tcpdump -w" (format is exactly the same), but one file
    per sip session
    (even if there is thousands of concurrent SIP sessions).
endef

TARGET_CC=$(TARGET_CXX)

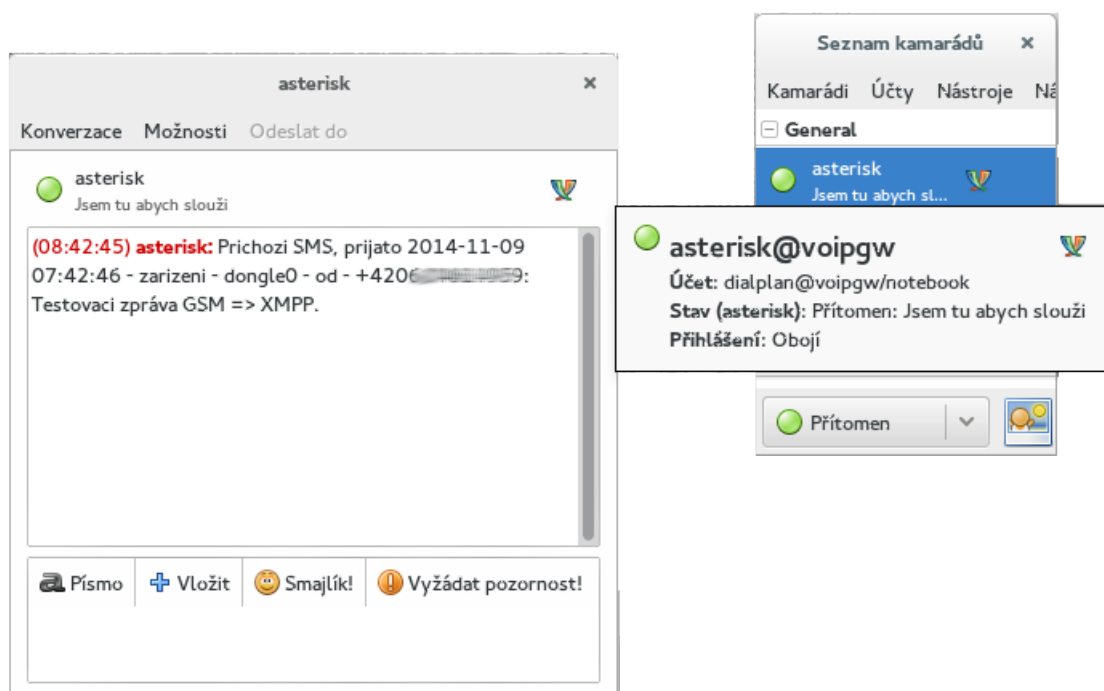
define Build/Compile
    $(TARGET_CONFIGURE_OPTS) \
        $(MAKE) -C $(PKG_BUILD_DIR) \
        CPPFLAGS="$(TARGET_CXXFLAGS) -fno-rtti" \
        LIBS="-lpcap"
endef

define Package/pcapsipdump/install
```

C OBRÁZKY ZACHYCUJÍCÍ ZPRACOVÁNÍ SMS



Obr. C.1: Zaslání SMS z mobilního telefonu



Obr. C.2: Okno XMPP zprávy v programu Pidgin

```
=====
Connected to Asterisk 1.8.30.0 currently running on VoIPgw (pid = 1073)
Verbosity is at least 21
-- Remote UNIX connection
[dongle0] Got SMS from +4206...: 'Testovací zpráva GSM => XMPP'
-- Executing [sms@default:1] JabberSend("Local/sms@default-00000002;1",
"asterisk,dialplan@voipgw,Prichodzi SMS, prijato 2014-11-09 07:42:46 - zarizeni - dongle0 -
od - +4206...: Testovací zpráva GSM => XMPP.") in new stack
-- Executing [sms@default:2] Hangup("Local/sms@default-00000002;1", "") in new stack
== Spawn extension (default, sms, 2) exited non-zero on 'Local/sms@default-00000002;1'
VoIPgw*CLI> 
```

Obr. C.3: Konzole PBX a reakce na příchozí SMS

D OBSAH PŘÍLOŽENÉHO CD

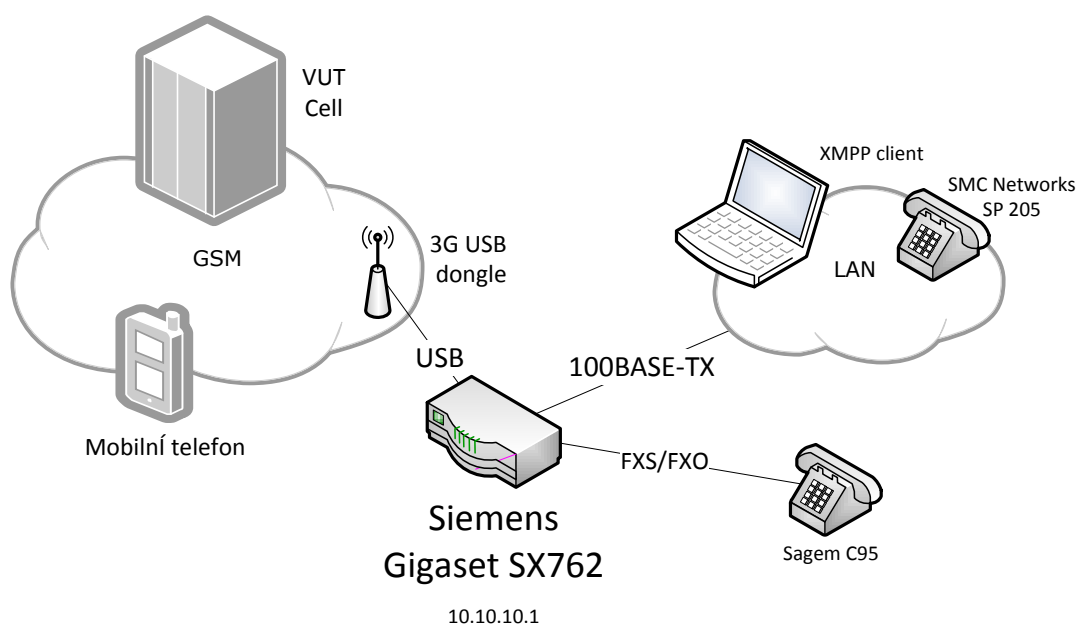
```
CD
├── Diplomová práce
│   ├── chan_dongle
│   │   └── Makefile
│   └── OpenWRT FW
│       ├── openwrt-lantiq-xway-GIGASX76X-squashfs.image
│       ├── openwrt-lantiq-xway-GIGASX76X-uImage
│       └── telephony
│           ├── asterisk18-app-alarmreceiver_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-app-exec_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-app-chanisavail_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-app-chanspy_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-app-mixmonitor_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-app-playtones_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-app-readexten_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-app-read_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-app-senddtmf_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-app-setcallerid_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-app-sms_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-app-stack_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-app-system_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-app-talkdetect_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-app-verbose_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-app-waituntil_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-app-while_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-cdr_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-codec-alaw_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-codec-a-mu_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-codec-g722_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-codec-g726_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-format-g726_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-format-g729_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-format-sln16_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-format-sln_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-func-base64_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-func-blacklist_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-func-devstate_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-func-extstate_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-func-global_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-func-channel_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-func-shell_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-func-uri_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-func-vmcount_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-chan-dongle_1.1.r10-18_lantiq.ipk
│           ├── asterisk18-chan-gtalk_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-chan-lantiq_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-chan-local_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-res-agi_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-res-musiconhold_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-res-srtp_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-res-timing-pthread_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-res-timing-timerfd_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18-voicemail_1.8.30.0-2_lantiq.ipk
│           ├── asterisk18_1.8.30.0-2_lantiq.ipk
│           ├── libiksemel_1.4-1_lantiq.ipk
│           └── libsrtp_1.4.4-1_lantiq.ipk
├── uboot-lantiq-gigasx76x_nor
│   ├── openwrt-lantiq-gigasx76x_nor-u-boot.img
│   └── uboot-lantiq-gigasx76x_ram
│       ├── openwrt-lantiq-gigasx76x_ram-u-boot.asc
│       └── openwrt-lantiq-gigasx76x_ram-u-boot.img
├── patch
│   └── 0153-lantiq-VPE-softdog.patch
├── pcapspidump
│   └── Makefile
└── Laboratorní úloha
    ├── LAB_zadání.pdf
    └── MN_SMCDSP-205.pdf
```

9 directories, 58 files

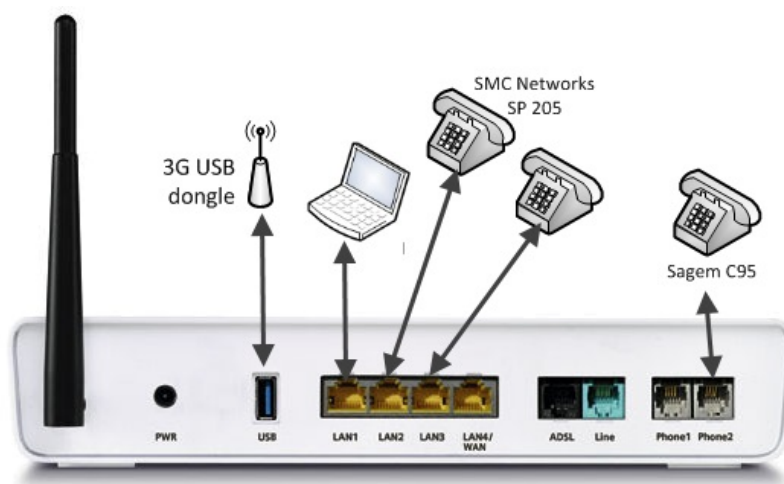
E LABORATORNÍ ÚLOHA

Teoretický úvod

Mnoho dnešních firem hledá způsob jak minimalizovat náklady spojené s telefonováním, ať už v rámci firmy nebo mimo ni. Jedna z cest vedoucí ke snížení nákladů je využití moderních technologií pro komunikaci. V této laboratorní úloze bude pro spojování hovorů využito PBX (Private branch exchange) Asterisk jako ústřednu, GSM (Globální Systém pro Mobilní komunikaci) modemu jako brány mezi GSM sítí a PBX a směrovače Gigaset SX76x. Pomocí mobilního telefonu je možné volat na SIP telefon a zaslanou zprávu zpracovat pomocí XMPP (Extensible Messaging and Presence Protocol) serveru Prosody a zaslat tuto zprávu na XMPP klienta Pidgin.



Obrázek 1: Základní topologie



Obrázek 2: Zapojení portů směrovače

Zadání

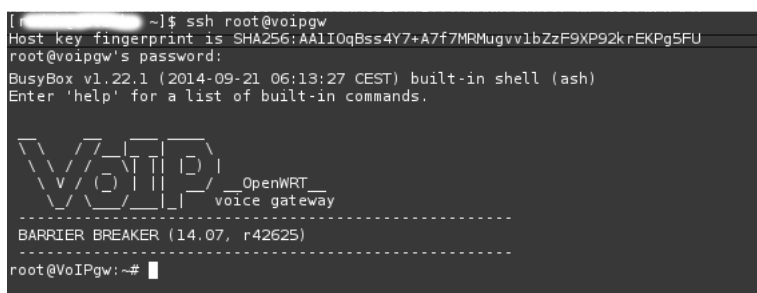
1. Seznamte se s dostupným HW na pracovišti
2. Zapojte dle schématu pracoviště
3. Proveďte konfiguraci číslovacího plánu pro SIP účty (klapky 100 a 200)
4. Proveďte konfiguraci SIP telefonu
5. Ověřte konfiguraci pomocí volání z čísla 100 na 200 a obousměrně
6. Proveďte konfiguraci rozšíření XMPP
7. Nainstalujte XMPP klienta Pidgin a nakonfigurujte účet
8. Proveďte konfiguraci číslovacího plánu pro zasílání informací o volání pomocí XMPP
9. Proveďte volání z GSM sítě na HW SIP telefon
10. Proveďte konfiguraci číslovacího plánu pro zasílání příchozích SMS zpráv na XMPP účet
11. Ověřte konfiguraci přeposílání SMS zpráv

Vybavení pracoviště

- USB 3G modem Huawei K3765
- Nokia 8800e¹
- 2x SIM²
- Směrovač Siemens Gigaset SX762
- 2x SIP telefon SMC Networks SP-205
- PC a SW Pidgin, Putty

Pracovní postup

Seznamte se s pracovištěm a prostudujte si návody dostupné na pracovišti. Směrovač Gigaset SX762 má předinstalovaný operační systém OpenWRT a všechny potřebný SW pro splnění úlohy. Na směrovači je zprovozněn také DHCP (Dynamic Host Configuration Protocol) server³. Veškeré připojené zařízení tedy budou získávat IP adresu automaticky z rozsahu IP adres rezervovaných pro DHCP. V laboratorní úloze bude konfigurace probíhat vzdáleně pomocí CLI. Libovolným způsobem se připojte pomocí na SSH protokolu na směrovač⁴ - rozhraní na kterém naslouchá SSH server má přidělenou IP adresu 10.10.10.1, uživatel „root“, heslo „toor“. Zkontrolujte, že SW ústředna Asterisk běží pomocí libovolného příkazu⁵. Pokud služba neběží, je možné ji spustit pomocí příkazu `/etc/init.d/asterisk start`. Následně se připojte do příkazového režimu ústředny pomocí `asterisk -rvv` (počet písmen *v* znamená úroveň vypisovaných informací - log level). Restart služby lze provést pomocí `/etc/init.d/asterisk reload`. Při restartu PBX je možné narazit na již obsazený port USB předchozí instancí PBX, v tom případě je nutné celé zařízení restartovat. Přibližně do 2 minut bude zařízení opět spuštěné spolu s SW PBX.



Obrázek 3: Ukázka připojení k SSH serveru směrovače

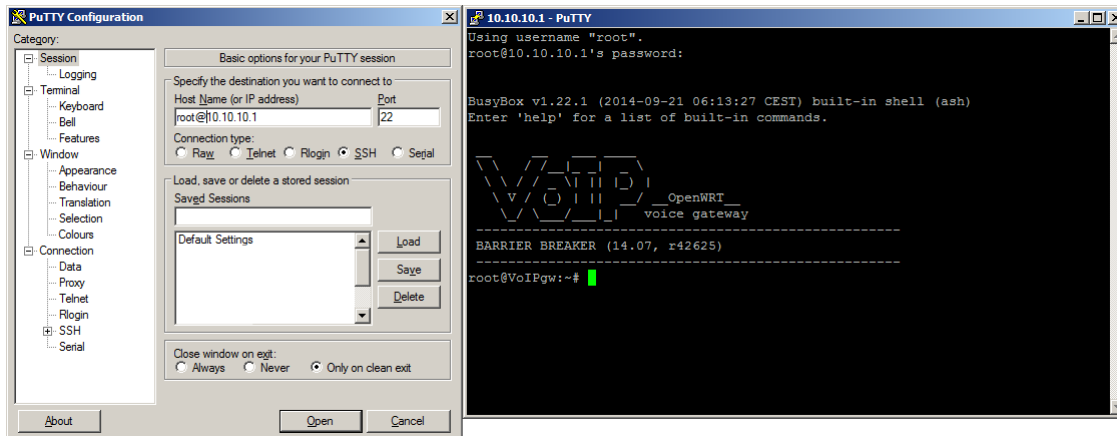
¹Může se lišit v závislosti na dostupném HW

²Subscriber Identity Module je účastnická identifikační karta, sloužící pro identifikaci účastníka v mobilní síti

³rozsah přidělovaných IP adres 10.10.10.100150

⁴K tomuto lze například využít program Putty

⁵například pomocí `ps a | grep -i [A]sterisk`



Obrázek 4: Ukázka připojení k SSH serveru směrovače pomocí programu Putty

Konfigurace SIP účtů

Konfigurace všech extension⁶ využívající protokol SIP je v konfiguračním souboru **sip.conf**. Tento konfigurační soubor je možné dělit do logických oddílů. Vlastnosti, které jsou platné pro všechny SIP účty, jsou uloženy v sekci [general]. Pro jednotlivé klapky lze změnit vlastnosti protokolu v sekci týkající se dané klapky, např. [100]. Nastavení platí pro všechny směry volání. Při změně v konfiguraci je potřeba načít změny konfigurace pomocí konzole PBX a příkazu *sip reload*.

Nejdůležitější parametry SIP konfigurace jsou následující:

- **disallow** = zakázání kodeků, výchozí se doporučuje all,
- **allow** = povolení kodeků [all, gsm, alaw, μ law, g723, g726, g729, h263, h264, atp.],
- **bindaddr** = adresa, na které Asterisk naslouchá (výchozí 0.0.0.0 – všechna rozhraní),
- **bindport** = port, na kterém Asterisk naslouchá (obvykle 5060),
- **context** = řetězec (příchozí volání budou směrovány do tohoto kontextu, tělo kontextu je definováno v *extensions.conf*),
- **nat** = [yes, no], povolení nebo zakázání NAT,
- **qualify** = [yes, no, xxx], testování dostupnosti klienta, standardně 60 sekund, pokud je zadána hodnota, pak je namísto 60 sekund testováno v intervalech xxx.

Specifické parametry pro jednotlivé účty mohou být:

- **type** = typy SIP entit, možnosti jsou friend, user, peer
- **auth** = řetězec, autentizace pomocí hash funkce,
- **username** = řetězec, slouží pro autentizaci,
- **secret** = heslo v otevřené textové formě,
- **callerid** = řetězec (identifikace volajícího),
- **port** = SIP port klienta,
- **host** = dynamic - umožňuje registraci klienta z libovolné IP adresy, pokud klient zná přihlašovací údaje,
- **context** = závisí od typu klienta, v případě *friend* je využit číslicový plán daného kontextu pro oba směry komunikace.

Příklad konfigurace SIP účtu pro klapku 100:

```
# cat /etc/asterisk/sip.conf

[general]
context=default      ; výchozí kontext

[100]
type=friend          ; klapka 100
secret=klapka1       ; heslo
username=100         ; uživatelské jméno
callerid="David" <100>
```

⁶Pojem extension lze vysvětlit jako přípojku k zařízení (telefon / SW klient) a také jako páteřní spoj Trunk

```
host=dynamic      ;
qualify=yes       ; testování dostupnosti
disallow=all      ; zakáže všechny kodeky
allow=alaw        ; povolí alaw
```

Nastavte pomocí výše uvedených parametrů klapky 100 a 200 dle specifikací v tabulce 1:

Tabulka 1: Nastavení SIP účtů

Klapka	Heslo	Testování dostupnosti	Zobrazené jméno
100	klapka1	ano	David
200	klapka2	40	Goliash

Ověření dostupnosti SIP klientů

Zkontrolujte stav účtů pomocí `sip show peers` v příkazovém řádku PBX Asterisk. Příkazem `sip show peer <klapka>` si můžete zobrazit podrobné informace o konkrétním účtu. Srovnajte Váš výpis klientů s níže uvedeným, nad rozdílly diskutujte.

```
VoIPgw*CLI> sip show peers
Name/username Host Dyn Forcerport ACL Port Status
101/101 10.10.10.121 D N 5060 OK (5 ms)
102/102 10.10.10.131 D N 5060 UNREACHABLE
2 sip peers [Monitored: 1 online, 1 offline Unmonitored: 0 online, 0 offline]
VoIPgw*CLI>
```

```
VoIPgw*CLI> sip show peer 101
* Name : 101
Dynamic : Yes
Callerid : "User_101" <101>
Addr->IP : 10.10.10.121:5060
Codecs : 0x80000008000e (gsm|ulaw|alaw|h263|testlaw)
Status : OK (5 ms)
Reg. Contact : sip:101@10.10.10.121:5060
```

Nechte si zkontrolovat výstup vyučujícím a vysvětlete jednotlivé položky.

Konfigurace číslovacího plánu

V konfiguračním souboru **extensions.conf** je definován číslovací plán ústředny. Číslovací plány jsou často děleny do menších logicky oddělených částí. Kontext s označením [default] je v této práci považován jako hlavní. Číslovací plán může obsahovat více kontextů a pravidel. SIP klient patřící například do kontextu [ABCD] bude v případě volání klapky prohledávat právě kontext, do kterého patří. Najde-li příslušnou klapku, inicializuje hovor, nenajde-li klapku, spojení ukončí. Tímto způsobem se dají logicky oddělit například různé skupiny určité organizace.

Základní konfigurace pro klapku 100:

```
[default] ; hlavní kontext
exten => 100,1,Dial(SIP/100) ; obsluha klapky 100
exten => 100,n,Hangup()
```

Některé funkce používané v číslovacím plánu:

- **include** - umožňuje rozdělit číslovací plán na menší kousky (kontexty). Nejčastěji se tato funkce používá pro zabezpečení hranic mezi různými třídami volajících.
- **Dial** - funkce se pokusí o připojení k jinému zařízení nebo koncovému bodu a překlenout volání SIP - koncový bod pro SIP účastníky
TAPI - koncové zařízení pro analogové rozhraní
- **Hangup** - aplikace pro ukončení hovoru na daném kanálu
- **Set** - umožňuje přiřazení hodnot k proměnným
- **Goto** - nepodmíněný skok na jinou část dialplanu (priorita, rozšíření, nebo kontext)

Editujte *extensions.conf* podle vzoru tak, aby bylo možné realizovat hovor mezi stanicemi 100 a 200 obousměrně. Po změně konfiguračního souboru je nutné změny načíst příkazem *dialplan reload*.

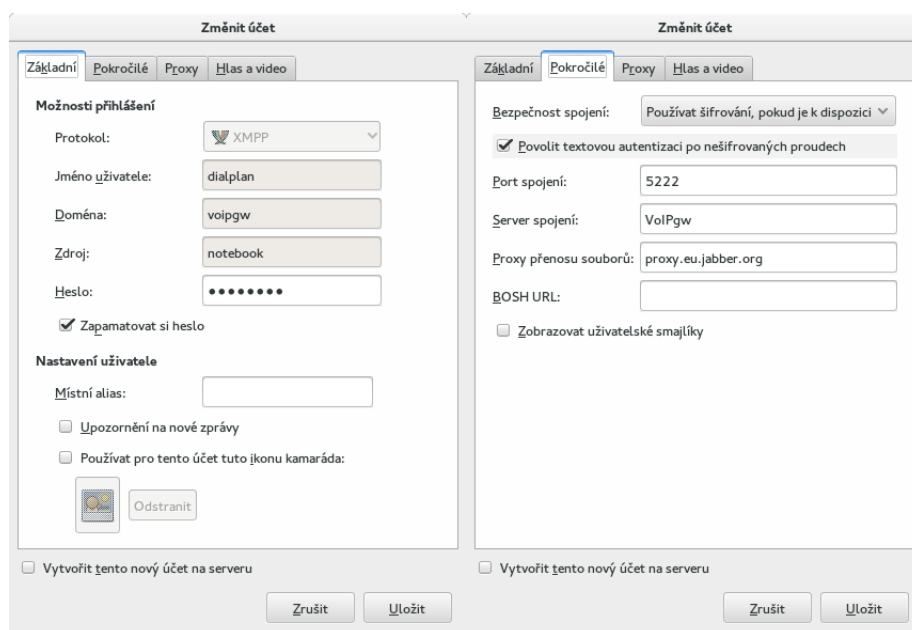
Konfigurace SIP telefonu

Nastavte HW SIP telefony podle tabulky 1 aby se dokázali připojit k SW PBX Asterisk. Pro získání oprávnění ke konfiguraci telefonu je potřeba vložit heslo **9876** v *Menu - Sys. Authority*. Nastavení účtu je poté možné provést v *Menu - SIP Settings*. Bližší informace o způsobu práce s přístrojem⁷ je možné nalézt v manuálu k zařízení dostupném na pracovišti.

Ověřte správnost konfigurace pomocí volání mezi stanicemi 100 a 200. Funkčnost prezentujte vyučujícímu.

Nastavení XMPP klienta Pidgin

Zařízení má nainstalovaný XMPP server Prosody. Parametry dostupných účtů jsou popsány v tabulce 2. Příjem zpráv bude realizováno pomocí IM klienta Pidgin a účtu „dialplan“. Nastavení účtu je vyobrazeno na obrázku 4.



Obrázek 5: Ukázka nastavení Sw Pidgin

Tabulka 2: Nastavení uživatelů XMPP serveru

uživatel	heslo	doména	poznámka
asterisk	asterisk	VoIPgw	účet pro Asterisk, ze kterého se posílají informace
dialplan	dialplan	VoIPgw	účet na který se posílají informace

Proveďte nastavení účtu IM klienta Pidgin a připojte se na XMPP server směrovače. **Funkčnost si nechejte zkontrolovat vyučujícím.**

XMPP rozšíření

K XMPP serveru směrovače bude přistupovat Asterisk pomocí jednoho z rozšíření a připraveného účtu „asterisk“. Konfigurace rozšíření protokolu XMPP pro PBX Asterisk se provádí v konfiguračním souboru *jabber.conf*. Nastavení hlasových hovorů pro protokol XMPP je dostupné v konfigurační soubor *gtalk.conf*. Hlasové funkce XMPP rozšíření nebudou v laboratorní úloze využity, uvedená konfigurace slouží pro inspiraci. Při každé změně konfigurace je potřeba nahrát změny pomocí *jabber reload* v příkazovém řádku PBX.

⁷Nebo konfigurace pomocí internetového prohlížeče


```
# cat /
[general]
debug=no ; Povolení ladících informací

[asterisk]
type=client ; Typ účtu – Klient
serverhost=voipgw ; Název serveru
username=asterisk@voipgw ; JID
secret=asterisk ; Heslo
priority=1 ; Priorita
port=5222 ; Port serveru 5222
usetls=yes ; Použití tls
usesasl=yes ; Použití sasl
distribute_events=yes ; Zásílat události
status=available ; Status: chat, available, away, xaway, dnd..
statusmessage=Jsem tu abych sloužil ; Popis statusu
```

```
# cat /
[general]
context=asterisk_call ; Výchozí kontext
allowguest=yes ; Povolit hovory od uživatelů mimo seznam účtu

[guest] ; nastavení neznámých příchozích volání
disallow=all ; zakázat všechny kodeky
allow=ulaw ; povolit pouze ulaw kodek
context=asterisk_call ; Výchozí kontext

[asterisk_call]
username=asterisk@voipgw ; uživatelský účet přes který se volá a přijímá
disallow=all ; zakázat všechny kodeky
allow=ulaw ; povolit pouze ulaw kodek
context=asterisk_call ; Výchozí kontext
connection=asterisk ; název componenty v jabber.conf pro spojení
```

Zprovoznění volání do sítě GSM pomocí SIP účtu

Na pracovišti jsou 2 SIM karty, které jsou využity pro přístup k experimentální GSM síti z laboratoře Komunikací prostředky mobilních sítí (MKPM). Čísla MSISDN jednotlivých karet jsou popsány v tabulce 3.

Tabulka 3: MSISDN

MSISDN	Zařízení	IMSI
2040	telefon	
2050	USB modem	230100000000050

V laboratorní úloze lze realizovat hovor také do GSM sítě, díky USB 3G modemu Huawei. V číslovacím plánu je tedy nutné uvést specifikovat požadované chování pro hovor splňující vlastnosti sítě.

Příklad volání do GSM sítě za pomoci 4 místního čísla

```
; volani do GSM site pomoci 4místneho čísla
exten => _XXXX,1,Dial(Dongle/dongle0/${EXTEN})
exten => _XXXX,n,Hangup()
```

Některé další funkce použité v číslovacím plánu:

- **Dial** - funkce se pokusí o připojení k jinému zařízení nebo koncovému bodu a překlenout volání
- **Dongle** - koncové zařízení - GSM rozhraní

Konfigurace číslovacího plánu s využitím XMPP - SMS

V následujícím příkladu konfigurace *extension.conf* je pomocí funkce *JabberSend* zasílaná zpráva pro uživatele dialplan@voipgw pokaždé, když je pomocí USB 3G modemu přijata SMS zpráva (klapka „SMS“). Funkce *JabberSend* má následující parametry:

Odesílatel - účet specifikovaný v *jabber.conf*

Příjemce - účet příjemce zpráv, parametr *buddy* v *jabber.conf*

Zpráva - samotná zpráva

V následujícím bloku kódu je příklad použití funkce v číslovacím plánu:

```
[dongle-incoming-sms]
exten => sms,1,JabberSend(asterisk,dialplan@voipgw,Zpráva... )
exten => sms,n,Hangup()
```

V praxi se pak tělo zprávy dá rozšířit o vhodné proměnné a funkce, například:

- \$EPOCH - počet sekund od půlnoci 1.1. 1970 GMT
- \$STRFTIME - formátování data/času dle specifikovaného nastavení
- \$CALLERID(num) -
- \$BASE64__DECODE - převod z řetězce kódovaného pomocí base64⁸ (návrátová hodnota je prostý text)
- \$SMS__BASE64 - tělo SMS zprávy, text může obsahovat nový řádek

výsledná konfigurace *extension.conf* doplněný o vhodné proměnné a funkce pak může vypadat například následovně:

```
[dongle-incoming-sms]
exten => sms,1,JabberSend(asterisk,dialplan@voipgw,Prichodi SMS, prijato ${STRFTIME(${EPOCH},,%Y-%m-%d %H:%M:%S)} - zarizeni - ${DONGLENAME} - od - ${CALLERID(num)}: ${BASE64__DECODE(${SMS__BASE64})}.)
exten => sms,n,Hangup()
```

⁸Base64 je kódování, které převádí binární data na posloupnosti tisknutelných znaků. Umožňuje přenos binárních dat kanály, které dovolují pouze přenos textů.

Úklid pracoviště

Po dokončení všech úloh a demonstraci vyučujícímu obnovte původní konfigurační soubory PBX Asterisk zadáním příkazu *cleanup* v CLI. Následně ukončete všechny příkazové řádky a všechna vzdálená spojení, vypněte telefon. Smažte vytvořené účty ve všech telefonech.

Doplňující otázky

- Vysvětlíte zkratky XMPP, SIP, PBX, GSM
- Který konfigurační soubor je nutné editovat pro změny nastavení SIP klientů?
- K čemu slouží konfigurační soubor *extension.conf*?
- Jakým způsobem se v číslovacím plánu směřuje hovor do GSM sítě pokud známe jen počet číslic? Jak by vypadala konfigurace pro GSM sítě dostupné v ČR?
- Jaký kodek byl použit pro hlasový přenos mezi HW SIP telefony?