



ZADÁNÍ BAKALÁ SKÉ PRÁCE

Název:	Nahrávání VoIP hovorů kontaktního centra
Student:	David Miozga
Vedoucí:	Ing. Tomáš Herout
Studijní program:	Informatika
Studijní obor:	Informa ní technologie
Katedra:	Katedra po íta ových systém
Platnost zadání:	Do konce letního semestru 2016/17

Pokyny pro vypracování

Vytvo te aplikaci, která za pomoci Built-in-Bridge (BIB) implementovaného v koncových za ízeních Cisco ve spolupráci s telefonní úst ednou Cisco Unified Communications Manager (UCM) zachytí hovory z kontaktního centra Cisco Unified Contact Center (UCC). Aplikace zároveň zachycené hovory p evede do n kterého ze známých audio souborů a uloží. Do databáze budou uloženy informace o hovoru. Prost ednictvím webové aplikace bude možné záznamy filtrovat, prohledávat i p ehrávat.

1. Zjist te, jaká ešení jsou momentálně k dispozici na trhu a porovnejte jejich klady a zápory. Vyvo te z nich záv ry pro svou aplikaci.
2. Popište, jak nahrávání za pomoci BIB funguje a vyzdvihn te výhody využití BIB oproti zrcadlení sí ového provozu.
3. Umožn te tvorbu nahrávacích pravidel - nap . na základ aktuálního asu, ísla volajícího nebo volaného, atp.
4. Aplikaci otestujte v produk ním nasazení.

Seznam odborné literatury

Dodá vedoucí práce.

L.S.

prof. Ing. Róbert Lórencz, CSc.
vedoucí katedry

prof. Ing. Pavel Tvrdík, CSc.
d kan

V Praze dne 7. ledna 2016

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
KATEDRA POČÍTAČOVÝCH SYSTÉMŮ



Bakalářská práce

Nahrávání VoIP hovorů kontaktního centra

David Miozga

Vedoucí práce: Ing. Tomáš Herout

27. června 2016

Poděkování

Rád bych poděkoval vedoucímu mé bakalářské práce, Ing. Tomáši Heroutovi, za cenné rady a konstruktivní připomínky k řešení dané problematiky. Dále patří poděkování společnosti TETA s. r. o. za poskytnutí potřebného testovacího prostředí a všem blízkým za podporu během tvorby této práce.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 46 odst. 6 tohoto zákona tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou, a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla, a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu), licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 27. června 2016

.....

České vysoké učení technické v Praze
Fakulta informačních technologií

© 2016 David Miozga. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí, je nezbytný souhlas autora.

Odkaz na tuto práci

Miozga, David. *Nahrávání VoIP hovorů kontaktního centra*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2016.

Abstrakt

Tato bakalářská práce se zabývá implementací systému pro nahrávání VoIP hovorů. Samotné nahrání hovorů je provedeno prostřednictvím proprietární funkcionality Built-in-Bridge některých koncových zařízení společnosti Cisco. Součástí práce je také analýza již existujících nahrávacích systémů.

Klíčová slova nahrávání hovorů, kontaktní centrum, VoIP, Cisco, BIB, SIP, RTP

Abstract

This thesis deals with implementation of VoIP calls recording system. The call recording is done by Cisco's proprietary function called Built-in-Bridge which is implemented in some end devices. Existing call recording systems are also analysed.

Keywords call recording, contact center, VoIP, Cisco, BIB, SIP, RTP

Obsah

Úvod	1
1 Cíle a požadavky	3
1.1 Požadavky na systém	3
2 VoIP telefonie	5
2.1 Úvod	5
2.2 Protokol SIP	6
2.3 Protokol SDP	10
3 Analýza	13
3.1 Možnosti řešení	13
3.2 Existující systémy	16
3.3 Závěry pro vlastní systém	18
4 Návrh	19
4.1 Návrh jádra systému	19
4.2 Návrh uživatelského prostředí	21
4.3 Návrh datových struktur	22
4.4 Databáze	23
5 Implementace	25
5.1 Použité technologie	25
5.2 Jádro systému	27
5.3 Uživatelské rozhraní	35
6 Testování	37
Závěr	39
Literatura	41

A	Seznam použitých zkratk	43
B	Komunikační diagramy	45
C	Instalace systému	49
D	Obsah přiloženého CD	51

Seznam obrázků

2.1	Ukázka SIP zprávy. [3]	9
2.2	Ukázka SDP zprávy.[5]	11
3.1	SIP komunikace pro vyjednání nahrávání hovoru.	15
4.1	Diagram entit databáze.	23
5.1	Vývojový diagram kontroly nahrávacích pravidel.	33
B.1	Hlasové proudy v nahrávaném koncovém zařízení s BIB.	46
B.2	Aktivní nahrávání s využitím BIB. [6]	46
B.3	Aktivní nahrávání s využitím hlasové brány. [6]	47
B.4	Pasivní nahrávání. [6]	47

Úvod

První telefonní hovor provedl Alexander Graham Bell již v roce 1876. Od té doby se telefonní technologie postupně vyvinuly tak, až je možné přenést hovor kamkoli v rámci celého světa prostřednictvím počítačových sítí. Tato technologie se nazývá VoIP (Voice over Internet Protocol).

Dnešní doba umožňuje využívat mnoha prostředků, které ušetří čas i peníze. Jedním z těchto prostředků jsou i telefonní kontaktní centra, která mohou obsloužit různé typy požadavků. Operátoři kontaktních center mohou zodpovídat dotazy volajících k určitým produktům či službám, případně sami iniciovat hovory za účelem průzkumů nebo nabídky produktů i služeb.

Telefonie, stejně jako celá síť Internet, je bohužel často zdrojem anonymity a vulgarity. U telefonie k těmto všem nešvarům přibývá fakt, že nejsme schopni vždy jednoduše dokázat, na čem se komunikující strany dohodly, jelikož telefonní hovory nejsou běžně nahrávány, na rozdíl od e-mailové komunikace, kterou lze zpětně dohledat u obou komunikujících stran.

Díky těmto faktům je potřeba implementovat na straně kontaktních center nahrávací systémy. Tyto nahrávací systémy zaznamenají veškerou telefonní komunikaci a v případě potřeby bude možné jednotlivé hovory dohledat, přehrát a z nahrávek vyvodit závěr pro případné spory. Nahrávky hovorů mohou v neposlední řadě sloužit vedoucím kontaktních center k osobnímu ohodnocení samotných operátorů.

Cíle a požadavky

Primárním cílem bakalářské práce je implementace systému pro nahrávání VoIP hovorů v kontaktních centrech. Nahrávání hovorů bude realizováno prostřednictvím technologie BIB, kterou disponují pouze některá koncová zařízení společnosti Cisco. Pro kompletní implementaci je potřeba splnit několik dílčích částí.

Prvním krokem ke splnění cíle je analýza již existujících nahrávacích systémů, které jsou běžně dostupné na trhu. Z jednotlivých vlastností těchto systémů jsou vyvozeny závěry pro implementaci vlastního systému.

Druhým krokem je samotná implementace vlastního systému, který dokáže nahrát požadovaný hovor a uložit ho do souboru v běžném audio formátu.

Posledním krokem je vytvoření webové aplikace, ve které je možné nahrané hovory přehrávat a zároveň spravovat nahrávací pravidla.

1.1 Požadavky na systém

Funkční požadavky

Funkční požadavky definují, co má výsledný systém dělat. [1]

- Nahrávání hovorů cílové telefonní linky.
- Přehrávání nahrávek hovorů prostřednictvím webové aplikace.
- Tvorba nahrávacích pravidel prostřednictvím webové aplikace.

Nefunkční požadavky

Nefunkční požadavky definují vlastnosti a chování výsledného systému jako celku. Může se jednat o stanovené funkce v oblastní výkonnosti, bezpečnosti nebo uživatelské přívětivosti. [1]

- Nahrávání více hovorů současně.

1. CÍLE A POŽADAVKY

- Plná funkcionality na linuxové distribuci Debian 8.4. „Jessie“ s jádrem ve verzi 3.16.0.
- Implementace v jazyce C++.
- Využití existujících knihoven a frameworků.

VoIP telefonie

2.1 Úvod

VoIP je technologií, která umožňuje přenášet hlasová data prostřednictvím počítačových sítí. [2]

Hovor může probíhat na jakékoli počítačové síti využívající IP protokol. VoIP lze využít ve firemních nebo univerzitních sférách, ale i v domácnostech. Díky tomu, že jsou hlasová data přenášena prostřednictvím běžných síťových paketů, je možné s nimi libovolně a relativně jednoduše nakládat. Ať je to již samotné nahrávání hovorů, modifikace hlasových dat nebo směrování hovorů v rámci celého světa.

Telefonie je jedním z druhů komunikace v reálném čase. Na komunikaci v reálném čase je kladen takový požadavek, aby data v rámci komunikace byla doručována co nejrychleji a s co nejmenším zpožděním. [2]

Protože je objem všech dat v počítačových sítích obrovský, je potřeba dát hlasovým datům prioritu, aby byla zajištěna potřebná kvalita komunikace v reálném čase. K tomu slouží technologie QoS (Quality of Service), která umožní na aktivních síťových prvcích upřednostnit hlasová data. Pokud by tato technologie nebyla aplikována, nastávaly by situace, ve kterých by telefonní hovory nedosahovaly požadovaných kvalit, protože by síťové pakety nesoucí hlasová data mohly být aktivními prvky zahazovány nebo pomalu odbavovány. [2]

Koncovým zařízením pro VoIP hovory může být buď stolní telefon komunikující po IP, softwarový klient do počítače, mobilní telefon nebo tablet, telekonferenční jednotka, nebo dokonce starý analogový telefon, protože jsou dostupné analogové převodníky. Koncovým zařízením může být např. i dveřník, IP kamera, atd. Prakticky stejnými technikami je možné přenášet i videohovory.

2.2 Protokol SIP

SIP je velmi rozšířeným protokolem využívaným k signalizaci VoIP hovorů je SIP. Signalizací je myšleno zajištění funkce sestavení nebo ukončení hovoru a vyjednání parametrů potřebných k přenosu hlasových dat. [3]

Alternativou pro SIP může být proprietární protokol SCCP (Skinny Client Control Protocol) společnosti Cisco nebo protokol H.323. Pro účely této práce probíhá veškerá signalizace prostřednictvím protokolu SIP.

Samotná hlasová data nejsou prostřednictvím signalizačních protokolů přenášena. K tomu slouží protokoly, které byly pro přenos médií navrženy. Takovým protokolem může být protokol RTP (Real-time Transport Protocol).

2.2.1 SIP URI

Jednotlivé zdroje SIP komunikace jsou identifikovány prostřednictvím unikátní SIP URI. Tento identifikátor má následující strukturu.

```
sip:uživatel:heslo@hostitel:port;uri-parametry?hlavičky
```

Často nejsou některé položky SIP URI přítomny. SIP URI v následujícím formátu je dostatečná pro úspěšné navázání komunikace mezi zdroji.

```
sip:uživatel@hostitel:port
```

Zdrojům, které jsou SIP URI identifikovány se říká User Agent (UA). Konkrétně UA Client (UAC) pro zdroj, který momentálně zastává roli klienta a UA Server (UAS) pro zdroj, který momentálně zastává roli serveru. Jednotlivé role se totiž mezi UA v rámci dialogu střídají.

2.2.2 SIP zprávy

SIP zprávy mohou být dvojího typu. A to požadavky od UAC směrem k UAS, nebo odpovědi v opačném směru. Zároveň má každý z těchto typů několik možných metod nebo stavových kódů, kde každý má v rámci signalizace svůj specifický význam. Typická SIP zpráva se skládá z následujících částí:

Uvozovací řádka Identifikuje konkrétní typ zprávy. (požadavek, resp. odpověď s konkrétní metodou, resp. kódem)

Hlavičky Nesou všechny důležité informace pro vyjednání spojení.

Prázdná řádka Musí být vždy přítomna za poslední hlavičkou. Tímto jsou odděleny hlavičky od těla zprávy.

Tělo Je prázdné nebo obsahuje zprávu protokolu SDP (Session Description Protocol), prostřednictvím kterého si komunikující strany domlouvají parametry pro přenos hlasových dat.

2.2.2.1 SIP požadavek

Uvozovací řádka pro požadavek se označuje jako Request-Line a má následující strukturu.

```
metoda cílová_SIP_URI verze_SIP_protokolu<CRLF>
```

Metoda každého z požadavku může být jedna z následujících:

REGISTER Slouží pro registraci klientů k serveru. Tyto zprávy jsou na server zasílány v určité časové periodě, aby server věděl, že klient je stále dostupný. Zároveň díky těmto požadavkům zná server IP adresu klienta a může na něj směřovat příchozí hovory.

INVITE Slouží jako požadavek na sestavení hovoru. V těle může obsahovat SDP zprávu, ve které udává komunikační parametry pro přenos hlasových dat.

ACK Slouží jako potvrzení o přijetí odpovědi s kódem 200-699.

CANCEL Umožňuje klientovi, který vyslal požadavek INVITE, tento požadavek ještě před samotným sestavením hovoru zrušit.

BYE Ukončuje aktivní hovor.

OPTIONS Umožňuje získat ze serveru informace o jeho komunikačních možnostech.

Typický SIP požadavek může vypadat následovně.[3]

```
INVITE sip:bob@biloxi.com SIP/2.0
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bK776asdhds
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142
```

<Tělo SDP zprávy není zobrazeno>

2.2.2.2 SIP odpověď

Uvozovací řádka pro odpověď se označuje jako Status-Line a má následující strukturu.

```
verze_SIP_protokolu kód_stavu fráze_odpovědi<CRLF>
```

Stavové kódy jsou rozděleny do tříd podle všeobecného významu, stejně jako HTTP kódy.

- 1XX** Poskytnutí - požadavek přijat a bude dále zpracován.
- 2XX** Úspěch - požadavek byl úspěšně přijat a potvrzen.
- 3XX** Přesměrování - pro potvrzení požadavku je potřeba vykonat další operace.
- 4XX** Chyba klienta - požadavek není syntakticky správně nebo nemůže být obslužen na tomto serveru.
- 5XX** Chyba serveru - požadavek je syntakticky správně a validní, ale server nemohl požadavek obsloužit.
- 6XX** Obecná chyba - požadavek nemůže být obslužen žádným serverem.

V následujícím přehledu jsou uvedeny příklady některých SIP odpovědí a jejich stavové kódy včetně textové fráze.

- 100** TRYING - žádost byla přijata prvním serverem v pořadí a je potřeba vykonat další nespecifikovanou operaci. Tato zpráva zabrání zasílání opakovaný požadavků INVITE.
- 180** RINGING - UA, který přijal požadavek INVITE se pokouší notifikovat cílového uživatele o příchozím hovoru (telefon vyzvání).
- 200** OK - požadavek byl úspěšně přijat a vykonán. Může být odpovědí např. na požadavek INVITE nebo BYE.
- 486** BUSY HERE - volaný byl úspěšně kontaktován. Hovor ale nemůže obsloužit, jelikož může mít buď jiný aktivní hovor nebo mít zapnutou funkci nerušit.
- 603** DECLINED - volaný byl úspěšně kontaktován, ale hovor nechce obsloužit nebo nemá možnost hovor obsloužit.

Typická SIP odpověď může vypadat následovně.

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP server10.biloxi.com
    ;branch=z9hG4bKnashds8;received=192.0.2.3
Via: SIP/2.0/UDP bigbox3.site3.atlanta.com
    ;branch=z9hG4bK77ef4c2312983.1;received=192.0.2.2
Via: SIP/2.0/UDP pc33.atlanta.com
    ;branch=z9hG4bK776asdhds ;received=192.0.2.1
To: Bob <sip:bob@biloxi.com>;tag=a6c85cf
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:bob@192.0.2.4>
Content-Type: application/sdp
Content-Length: 131

<Tělo SDP zprávy není zobrazeno>
```

Obrázek 2.1: Ukázka SIP zprávy. [3]

2.2.2.3 Hlavičky

Hlavičky obsahují všechny informace potřebné k sestavení, úpravě či terminaci spojení. Vzhledem k velkému množství hlaviček jsou popsány jen některé základní. Syntaxe hlaviček je **název: hodnota**. Je tedy stejná jako u protokolu HTTP.

Via Přesně definuje cestu, kterou požadavek prochází. Každý UA, na který požadavek dorazí, musí přidat na konec seznamu svou IP adresu. Ve chvíli, kdy požadavek dorazí do cíle, je přečten poslední záznam a odpověď je zaslána UA s touto IP adresou. S každým dalším krokem během cesty zpět jsou takto jednotlivé záznamy všemi mezilehlými UA čteny a odebírány do doby, než je odpověď doručena.

To Identifikuje příjemce požadavku. Příjemcem je myšlen konkrétní uživatel.

From Identifikuje uživatele, který požadavek vytvořil.

CSeq Sestává z celého čísla, které je v rámci jedné SIP relace při každém novém požadavku inkrementováno.

Contact Obsahuje SIP URI UA. Pole je povinné v požadavku s metodou INVITE. Určuje totiž, kam mají být odeslány případné budoucí požadavky. To je rozdíl oproti hlavičce Via, která určuje, kam mají být odeslány odpovědi.

Call-ID Obsahuje globálně unikátní identifikátor pro daný hovor. Jelikož musí být v danou chvíli zcela unikátní, generuje se na základě náhodného řetězce a IP adresy nebo doménového názvu UA.

Max-Forwards Určuje, mezi kolika mezilehlými UA může být požadavek směrován, než dorazí k adresátovi. Hodnotou je tedy celé číslo, které je s každým skokem dekrementováno o jedničku. Při dosažení hodnoty 0 je požadavek zamítnut. Lze přirovnat k mechanismu Time To Live (TTL) IP protokolu.

Content-Type Definuje typ obsahu těla zprávy. Pro tělo obsahující SDP zprávu je hodnotou `application/sdp`.

Content-Length Určuje velikost těla zprávy v bajtech. Pokud tělo není obsaženo, ale hlavička Content-Type ano, musí obsahovat hodnotu 0. Pokud je pro přenos signalizace využit transportní protokol TCP, je tato hlavička povinná.

2.3 Protokol SDP

Dva přímo komunikující UA si mezi sebou musí vyjednat parametry pro přenos hlasových dat. Těmito parametry je kodek, cílová adresa a port. K tomu slouží protokol SDP, který je obsažen v těle SIP zprávy. Vyjednání těchto parametrů funguje na principu nabídky a odpovědi na tuto nabídku. Zpravidla iniciátor sezení zahrne do SDP zprávy informace o kodecích, kterými je schopný hlasová data kódovat. UA komunikující přímo s iniciátorem vybere jeden z kodeků a vloží ho do své odpovědi. Touto odpovědí je opět SDP zpráva obsažena v SIP odpovědi. Na základě této komunikace si oba koncové body vyjednaly jeden konkrétní kodek. Jelikož je SDP textově orientovaný protokol, jsou jednotlivé záznamy reprezentovány řetězci, kde každý záznam je na jednom řádku. Každý řádek má formát `znak=hodnota`, kde znak je jeden znak definující typ záznamu.

v version - verze protokolu, momentálně je dostupná pouze verze 0.

o origin - původce sezení, skládá se z uživatelského jména, identifikátoru sezení, verze sezení, typu sítě (např. IN pro Internet) a IP adresy.

s session name - pojmenovává sezení. Pokud nemá sezení konkrétní význam, měla by být obsahem jedna mezera.

t time - určuje, v jakém časovém intervalu je sezení aktivní.

c connection - nese informace o sezení, tedy kdo a jak bude za tuto stranu sezení komunikovat. Skládá se z typu sítě, typu IP adresy (IP4 nebo IP6) a samotné IP adresy.

- m** media description - popisuje typ a možnosti přenášených dat prostřednictvím tohoto sezení. Skládá se z typu médií (v případě telefonních hovorů **audio**), síťového portu, na který budou data zasílána, protokolu, prostřednictvím kterého data budou přenášena (např. RTP/AVP a číselnou reprezentaci kodeků, prostřednictvím kterých mohou být data kódována. Mapování mezi číselnou hodnotou a kodekem lze prohlédnout v tabulce na [4]
- a** attribute - účelem atributů je rozšířit SDP zprávu. Využívají se např. pro přesnější popis jednotlivých kodeků z řádku popisujícího media.

```
v=0
o=RECORDER 1462879840 1462879840 IN IP4 1.1.1.1
s=SpanLess RECORDING
c=IN IP4 1.1.1.1
t=0 0
m=audio 17598 RTP/AVP 0 8 9
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:9 G722/8000
```

Obrázek 2.2: Ukázka SDP zprávy.[5]

Analýza

Analytická část práce se v první fázi věnuje možnostem řešení problematiky nahrávání VoIP hovorů. Ve druhé části jsou představeny a porovnány některé z již existujících systémů, které slouží k nahrávání VoIP hovorů nejen kontaktních center. Z této analýzy jsou vyvozeny závěry pro implementaci vlastního nahrávacího systému.

3.1 Možnosti řešení

Tato sekce představuje jednotlivé způsoby, jakými lze VoIP hovory zaznamenat v infrastruktuře, která sestává z komponent společnosti Cisco. Tím je myšlena telefonní ústředna Unified Communications Manager, kontaktní centrum Unified Contact Center, koncová zařízení společnosti Cisco a v neposlední řadě také aktivní síťové prvky této společnosti.

Na nahrávání hovorů lze nahlížet ze dvou pohledů. Oba tyto pohledy jsou míněny ve směru příchozích hovorů do infrastruktury společnosti, ve které jsou hovory nahrávány. [6]

- Nahrávání hovorů z pohledu volajícího.
- Nahrávání hovorů z pohledu volaného.

Pokud by byl jeden a ten samý příchozí hovor nahráván oběma způsoby, výsledkem by byly nepatrně odlišné nahrávky.

V případě nahrávání hovoru z pohledu volaného, tedy operátora kontaktního centra, by nahrávka obsahovala pouze samotný dialog mezi volajícím a operátorem.

Při nahrávání hovoru z pohledu volajícího, tedy zákazníka kontaktního centra, by nahrávka obsahovala kompletní záznam již od vytočení telefonního čísla volajícím, respektive přijmutí hovoru systémem kontaktního centra. To znamená, pokud by se volající dovolal do kontaktního centra, kde je využíváno

IVR (Interactive Voice Response), bude v nahrávce i ta část hovoru, kde volající prochází jednotlivými položkami IVR. IVR je interaktivní hlasové menu, kde jsou volajícímu pod číselnými kódy nabízeny jednotlivé služby kontaktního centra.

Dalším využitím, kdy je potřeba nahrávat hovory z pohledu volajícího, je v situacích, kdy nejsou hovory směrovány na některé z koncových zařízení, které se nachází uvnitř sítě společnosti a je nastaveno pro nahrávání hovorů. Těmito koncovými zařízeními mohou být např. mobilní telefony, které se nachází ve VTS (Veřejná Telefonní Síť). V tuto chvíli se totiž koncové zařízení volaného nenachází v prostředí společnosti a není tak možné toto zařízení ovládat, případně ho nechat zasílat jakákoli data do sítě společnosti.

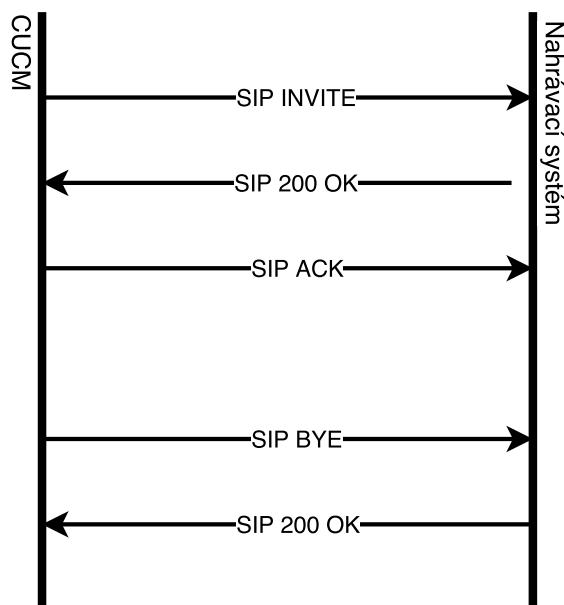
Samotné nahrávací metody lze rozdělit na dvě základní kategorie a podkategorie.

3.1.1 Aktivní nahrávání

Nahrávání hovorů je vyjednáno prostřednictvím SIP komunikace mezi telefonní ústřednou a nahrávacím systémem. [7] Každý hovor se skládá ze dvou proudů hlasu. Jeden nese hlas volajícího, druhý volaného. Pro každý z proudů je sestaveno jedno SIP spojení mezi nahrávacím systémem a telefonní ústřednou, proto nahrávací systém obdrží pro nahrání jednoho hovoru dva SIP INVITE požadavky.

3.1.1.1 IP Phone-Based

Založeno na nahrávání koncového VoIP zařízení. Telefonní ústředna v případě hovoru na tomto koncovém zařízení vyše prostřednictvím dedikovaného přenosového spoje (anglicky SIP trunk) SIP INVITE požadavky na nahrávací systém a sestaví se mezi nimi spojení, ve kterém jsou dohodnuty všechny parametry přenosu médií. Na straně nahrávaného koncového zařízení jsou oba z proudů hlasu volajícího i volaného duplikovány a odesílány přímo na nahrávací server. Duplikaci hlasových proudů zajišťuje proprietární funkcionalita BIB (Built-In-Bridge) od společnosti Cisco. Tato metoda je využita v implementaci vlastního nahrávacího systému.



Obrázek 3.1: SIP komunikace pro vyjednání nahrávání hovoru.

Komunikační diagram je znázorněn na obrázku B.2

3.1.1.2 Network-Based

Založeno na nahrávání hlasové brány. Duplikace proudů hlasu nahrávaného hovoru je prováděna na straně hlasové brány. Ta většinou bývá hraničním prvkem VoIP infrastruktury. Proto tato metoda najde využití především při nahrávání hovorů, které běžně procházejí touto branou. Jedná se tedy o hovory z VTS nebo jiného VoIP segmentu. Pro nahrávání interních hovorů by musely všechny tyto hovory procházet hlasovou branou, což je ve většině případů zbytečné a mohlo by to mít dopad na její výkon.

Komunikační diagram je znázorněn na obrázku B.3

3.1.2 Pasivní nahrávání

Veškerý síťový provoz přichází do jednoho monitorovaného rozhraní na aktivním síťovém prvku (přepínači) je duplikován a mimo svůj cíl také směrován do jednoho konkrétního rozhraní na tomto přepínači, které je směrované na nahrávací systém. Nahrávací systém tedy musí projít veškerá data, která přijme na svém síťovém rozhraní a vybrat si z nich pouze ta data, která jsou relevantní pro nahrávání VoIP hovorů.

Tato metoda je primárně určena k monitorování síťového provozu. Proto se tato funkcionality přepínačů nazývá SPAN (Switched Port Analyzer). Nahrávání hovorů prostřednictvím této metody se také někdy nazývá SPAN-Based

(založeno na SPAN). Metody aktivního nahrávání jsou nazývány **SPAN-less** (bez využití SPAN). [7]

Duplikována nejsou pouze data VoIP hovorů, ale kompletní provoz, který prochází monitorovaným portem. Tato skutečnost může mít dopad na výkon daného přepínače, případně i část sítě.

Hlavní výhodou tohoto řešení je možnost nasazení v jakémkoli prostředí, ve kterém je využito IP telefonních ústředen a koncových zařízení jiných společností, než je společnost Cisco. [7]

Komunikační diagram je znázorněn na obrázku B.4

3.2 Existující systémy

Komerčně dostupných nahrávacích systémů existuje na trhu mnoho. Proto bude představen pouze zlomek z nich, ale i z takto malého vzorku nahrávacích systémů lze vyvodit důležité závěry a vlastnosti pro implementaci vlastního nahrávacího systému.

3.2.1 ZOOM CallRec

Vyvinuto českou společností ZOOM International s.r.o., která má s nahrávacím hovorů již dvacetiletou zkušenost. Díky tomu si vybojovala dominantní pozici na trhu nahrávacích systémů a její produkty jsou implementovány i za hranicemi České republiky. [8]

Klady

- Umožňuje všechny metody nahrávání.
- Propracovaný přehrávač nahrávek.
- Tvorba nahrávacích pravidel.
- Systém v podobě virtuální appliance.
- Umožňuje integraci s kontaktním centrem od společnosti Cisco. To znamená, že jsou v nahrávacím systému dostupné další informace o hovorech, jako je např. jméno operátora, který hovor obsloužil.
- Webové uživatelské prostředí.
- ZOOM nezůstává jen u nahrávání hovorů. Nabízí komplexní řešení pro kontaktní centra, díky kterému lze hovory monitorovat v reálném čase, nahrávat obrazovky operátorů, analyzovat jednotlivé hovory, atd. [9]

Zápory

- API, prostřednictvím kterého lze získávat prakticky pouze samotné nahrávky hovorů.

Informace byly čerpány z [7].

3.2.2 MiaRec

MiaRec je společnost z Kalifornie zabývající se nahráváním VoIP hovorů již přes deset let. Za tu dobu si také našla zákazníky po celém světě.

Klady

- Umožňuje integraci s kontaktním centrem od společnosti Cisco. To znamená, že jsou v nahrávacím systému dostupné další informace o hovorech, jako je např. jméno operátora, který hovor obsloužil.
- Má k dispozici REST API, prostřednictvím kterého lze získávat ze systému informace o hovorech i samotné nahrávky.
- SOAP API, prostřednictvím kterého lze nastavovat některé vlastnosti systému.
- Přímý přístup do databáze v případě, že REST API neumožňuje potřebnou funkcionalitu.
- Webové uživatelské prostředí.

Zápory

- Pouze aktivní nahrávání prostřednictvím BIB.

Informace byly čerpány z [10].

3.2.3 Cisco MediaSense

MediaSense není sám o sobě zcela soběstačný systém jako předchozí dva systémy. Obstará samotné nahrávání hovorů, ale pro procházení nahrávek je k dispozici pouze jednoduché webové prostředí, které ovšem nenabízí zdaleka tolik funkcionalit jako předchozí dva systémy. Sama společnost Cisco tento systém prezentuje spíše jako framework pro integraci se systémy třetích stran. Lze tedy využít pro implementaci vlastního nahrávacího systému. [11]

Klady

- Vytvořeno společností Cisco.
- Umožňuje všechny metody nahrávání.
- Systém v podobě virtuální appliance.
- Propracované API, prostřednictvím kterého lze získat o hovorech veškeré informace a zároveň dostávat notifikace o změnách v systému, např. započetí nahrávání hovoru, atd.

Zápory

- Pouze základní uživatelské rozhraní umožňující přehrát zaznamenané nahrávky.
- Cena může být vyšší než u předchozích systémů, jelikož se musí platit licence společnosti Cisco a ještě samotný systém integrující MediaSense.

Informace byly čerpány z [11]

3.3 Závěry pro vlastní systém

Vzhledem k zadání práce a vlastnostem jednotlivých možností řešení nahrávání VoIP hovorů bude ve vlastní implementaci nahrávacího systému využito aktivního nahrávání s využitím BIB. Pasivní nahrávání je celkově nevyhovující a aktivní nahrávání s využitím hlasové brány je pro primární účely kontaktních center zbytečné. Pro pracovníky kontaktních center na vedoucích pozicích (tzv. supervizoři) je důležitá samotná komunikace mezi volajícím a operátorem kontaktního centra.

Vlastnosti nahrávacího systému byly vyvozeny z vlastností, které mají nebo naopak nemají již existující nahrávací systémy. Následující seznam obsahuje výčet vlastností, které jsou pro nahrávací systém vhodné a budou v rámci bakalářské práce a v budoucnu implementovány.

- Aktivní nahrávání hovorů s využitím BIB.
- REST API pro přístup k informacím o hovorech.
- Přihlašování do webového aplikace LDAP/Active Directory účty.
- Moderní a jednoduché uživatelské prostředí.
- Přehrávač nahrávek integrovaný do webové aplikace.

Návrh

4.1 Návrh jádra systému

Před tím, než bylo možné přistoupit k samotné implementaci, bylo potřeba navrhnout strukturu jádra systému, které obstarává komunikaci s telefonní ústřednou, koncovými zařízeními a dokáže přijaté proudy hlasu volajícího a volaného spojit v jednu audio stopu, která je následně uložena do audio souboru ve známém formátu. Tento výsledný audio soubor je možné přehrát libovolným přehrávačem audio souborů, který podporuje daný formát audio souboru.

Pro návrh struktury jádra musel být znám způsob komunikace mezi jednotlivými zúčastněnými uzly, kterými jsou telefonní ústředna, nahrávací systém a koncová zařízení. Podrobný popis komunikace je dostupný v oficiální dokumentaci společnosti Cisco týkající se technologie BIB na [12], ze které bylo pro potřeby této práce čerpáno.

4.1.1 Struktura jádra systému

Po analýze komunikace mezi nahrávacím systémem a telefonní ústřednou bylo možné začít se samotným návrhem jednotlivých komponent jádra systému.

V jádru systému je potřeba vykonávat rozličné typy úloh. Každá z těchto úloh je vykonávána ve svém vlastním výpočetním prostředku. Bylo nutné rozhodnout, zda těmito výpočetními prostředky budou jednotlivá vlákna procesu nebo několik samostatných procesů.

Vlákna Systém je spuštěn pouze v jednom jediném procesu, ze kterého jsou spouštěna postupně jednotlivá vlákna. Každé z vláken má na starost konkrétní část řešení problému nahrávání hovoru.

Procesy Systém je prvotně spuštěn v jednom procesu. Pro řešení jednotlivých problémů nahrávání hovoru jsou ovšem z tohoto procesu spouštěny další samostatné procesy.

Po zvážení všech kladů a záporů bylo rozhodnuto o využití vícevláknového přístupu. Mezi stěžejní výhody spadá jednodušší sdílení dat mezi vlákny, nižší paměťová náročnost a nižší režie na přepínání kontextu. [13]

Hlavní vlákno

První vlákno, které se po spuštění aplikace spustí, slouží pro deklaraci a inicializaci datových struktur. Tyto datové struktury jsou následně využívány ostatními pracovními vlákny, která jsou spouštěna z tohoto hlavního vlákna.

Vlákna pro síťovou komunikaci

Jelikož ústředna iniciuje SIP komunikaci s nahrávacím systémem na TCP portu, musí být nejprve dostupný tento port ze strany systému, aby mohl obsloužit všechny požadavky telefonní ústředny.

Iniciální požadavky typu SIP INVITE a odpovědi na ně jsou přenášeny prostřednictvím transportního protokolu TCP (Transmission Control Protocol), protože tyto zprávy jsou pro nahrání hovoru klíčové. Pokud na nahrávací systém žádný požadavek SIP INVITE nedorazí, případně telefonní ústředně nepříjde potvrzení SIP 200 OK na tento požadavek, nedostane koncové zařízení od telefonní ústředny potřebné informace o nahrávacím systému a tím pádem nemůže začít vysílat proudy hlasu do nahrávacího systému. Těmito potřebnými informacemi jsou IP adresa nahrávacího systému, síťový port a seznam použitelných kodeků. Obsaženy jsou v těle SIP 200 OK zprávy, které tvoří SDP zpráva.

Veškerá další SIP komunikace týkající se daného hovoru probíhá na UDP (User Datagram Protocol) portu 5060.

Mezi zprávy, které přichází na UDP port patří SIP ACK, jež jsou v tomto případě odpovědí od telefonní ústředny na SIP 200 OK. V SDP zprávě v těle SIP ACK zprávy jsou obsaženy informace o kodeku, který si koncové zařízení vybralo pro data proudu hlasu, která bude vysílat do telefonní ústředny.

Další, po síťové stránce nepotvrzovanou zprávou, je SIP BYE určující konec nahrávání daného proudu hlasu. V systému tedy může dojít k uzavření UDP portu, na němž byla přijímána hlasová data. Se SIP BYE zprávou souvisí odpověď na tuto zprávu, kterou je opět SIP 200 OK. Tato odpověď již nenese žádné dodatečné informace. Telefonní ústředně dává pouze na vědomí, že nahrávací systém už nahrávání daného proudu hlasu ukončil a žádá další data pro tento proud nebude schopen přijmout a zaznamenat.

Vlákno pro kontrolu aktivních hovorů

Všechny informace o aktuálně nahrávaných hovorech jsou dočasně ukládány do pomocného seznamu, který obsahuje záznamy pro všechny nahrávané hovory. Jedna z ukládaných informací o hovoru jsou příznaky, které určují, zda byl již ukončen příjem proudů hlasu.

Seznam aktivních hovorů je periodicky po pěti sekundách procházen a je zjišťováno, zda se mezi nimi nenachází již ukončený hovor. Ukončený hovor má nastaveny příznaky identifikující, že příjem obou proudů hlasu byl ukončen. V případě, že se v seznamu objeví ukončený hovor, je hovor odebrán ze seznamu aktivních hovorů a je zařazen do fronty pro následné zpracování.

Vlákno pro zpracování ukončených hovorů

Z fronty jsou hovory postupně odebírány a zpracovávány. Tím rozumíme slučování jednotlivých proudů hlasu volajícího a volaného do jedné stereofonní audio stopy. Hlas každé z komunikujících stran je ve svém vlastním kanálu. Tím je zajištěno, že bude možné vyslechnout pouze jednu z komunikujících stran bez rušení druhou stranou. Pokud by totiž oba hlasy byly pouze v jednom kanálu a volající i volaný by hovořili ve stejnou dobu, mohlo by při přehrávání nahrávek docházet k problémům s porozuměním jednotlivým stranám.

Vlákno pro práci s databází

Informace o všech nahraných hovorech jsou perzistentně ukládány do databáze. Komunikace s databází probíhá ve vlastním vlákne.

4.2 Návrh uživatelského prostředí

Aby systém mohl využívat supervizor kontaktního centra, je potřeba, aby bylo možné hovory zobrazit a přehrát co nejjednodušeji. Zároveň je potřeba uživatelsky přívětivě definovat nahrávací pravidla.

Pro interakci s nahrávacím systémem na uživatelské úrovni nejlépe poslouží webová aplikace. Je možné ji zobrazit jak na stolních počítačích a noteboocích, tak na mobilních zařízeních, jimiž jsou mobilní telefony a tablety. Díky tomu má supervizor možnost pracovat se systémem odkudkoli.

4.2.1 Architektura

Webové prostředí nahrávacího systému je založeno na návrhovém vzoru MVC (Model-View-Controller). Tento návrhový vzor rozděluje aplikaci na tři samostatné vrstvy, úrovně. Tyto vrstvy mají každá svůj vlastní význam a funkci. Zároveň MVC určuje, které vrstvy spolu smějí komunikovat a které ne.

Model Vrstva Model pracuje s objekty, jež reprezentují samotná data aplikace. Těmi mohou být např. objekty reprezentující databázové entity.

View Vrstva View pracuje s objekty, jež reprezentují uživatelské prostředí aplikace. Data pro uživatelské rozhraní, která vidí samotný uživatel aplikace, jsou získávána prostřednictvím Controlleru z Modelu.

Controller Vrstva Controller je prostředník mezi vrstvou Model a View. Směrem od Modelu dodává data pro View, která uvidí uživatel a směrem od View k Modelu zprostředkovává data a interakce, které uživatel zadal a provedl prostřednictvím uživatelského rozhraní. V Controlleru zároveň může probíhat nad samotnými daty další výpočetní nebo modifikační logika.

4.3 Návrh datových struktur

S daty, která společně vytváří jeden logický celek, se nejlépe pracuje, když jsou jako jeden logický celek i ukládána a reprezentována. Využití zapouzdření určitých dat do jednoho jediného logického celku v rámci jádra nahrávacího systému je nastíněno v této sekci.

4.3.1 Entita reprezentující SIP zprávu

Pro vytváření a parsování SIP zpráv je využita open-source knihovna oSIP pro programovací jazyk C. Tato knihovna umožňuje kompletní převod z textové podoby SIP zprávy do struktury reprezentující danou SIP zprávu. Pro přístup k jednotlivým položkám má knihovna k dispozici funkce. Bohužel k některým položkám není možné se dostat přímo jednou funkcí, ale je potřeba využít určitou sekvenci různých funkcí, kde výstup jedné funkce je vstupem pro další.

Z toho důvodu je vytvořena entita, která obaluje knihovní strukturu reprezentující SIP zprávu. Do vlastní entity je možné dopsat metody, které sjednotí složitější konstrukce. Jelikož je se SIP zprávami zacházeno v jádru systému prakticky při každé činnosti, byl by kód kvůli složitým konstrukcím zbytečně nepřehledný a zdlouhavý. S využitím zapouzdření je každá operace otázkou jednoho řádku, což samotné knihovní funkce neumožňují.

4.3.2 Entita reprezentující SDP zprávu

Stejně jako pro manipulaci se SIP zprávami je pro manipulaci se SDP zprávami využito open-source knihovny oSIP. OSIP knihovna implementuje i manipulaci s SDP zprávami, jelikož jsou nedílnou součástí SIP zpráv, byť tvoří jen volitelné tělo SIP zprávy.

SDP zprávy je potřeba také jednoduše vytvářet, případně číst, proto je implementována další entita, jež SDP zprávu reprezentuje a především obaluje knihovní SDP zprávu. Důvody jsou prakticky totožné jako u entity reprezentující SIP zprávu.

4.3.3 Struktura reprezentující RTP paket

Proudy hlasu volajících nepřicházejí po síti jako čistě hlasová data, která by bylo možné okamžitě ukládat nebo přehrávat. Každý příchozí datový paket

totiž obsahuje nejen hlasová data, ale také data potřebná ke správnému fungování RTP protokolu. Z toho důvodu je potřeba každý přijatý datový paket rozdělit na několik částí. Struktura RTP paketu je definována v RFC na [14].

Z tohoto důvodu je vytvořena struktura, ve které jsou uloženy všechny části RTP paketu, aby se s nimi dalo následně jednoduše pracovat, např. získat samotná hlasová data.

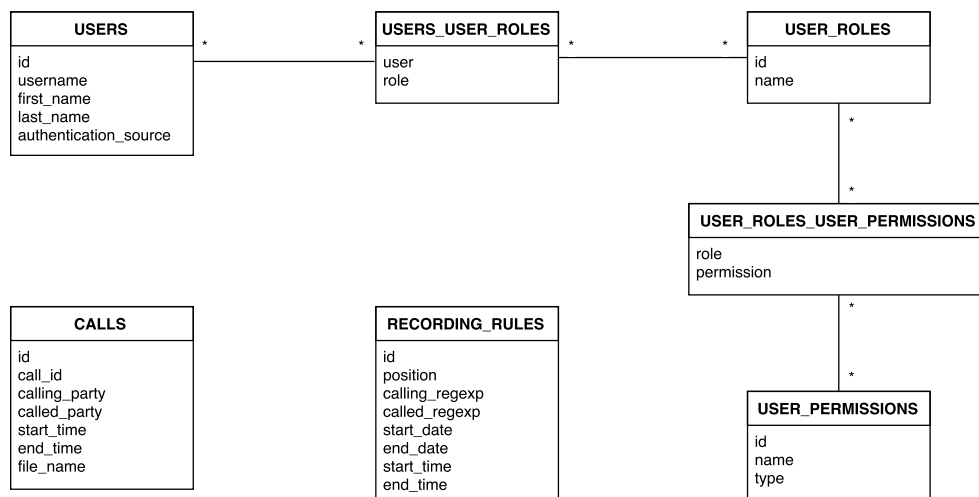
4.3.4 Struktura reprezentující hovor

Společným jmenovatelem pro všechny komponenty nahrávacího systému je nahrávaný hovor. Vzhledem k tomu, že hovor je sdílen napříč všemi pracovními vlákny, je žádoucí, aby byl každý konkrétní hovor reprezentován jako jeden celek. Všechny důležité informace o hovoru jsou tedy okamžitě, v případě potřeby, k dispozici.

4.4 Databáze

Data o nahraných hovorech a nahrávací pravidla musejí být perzistentně ukládána. K tomu je využita relační databáze Postgres.

Databáze je složena z několika tabulek, kde část z nich využívá jak jádro nahrávacího systému, tak uživatelské prostředí a část z nich slouží pouze pro webovou aplikaci.



Obrázek 4.1: Diagram entit databáze.

calls Entita udžující informace o hovorech.

recording_rules Entita udržující nahrávací pravidla.

4. NÁVRH

users Entita udržující uživatele webové aplikace.

users__user__roles Entita vztahů, která vznikla dekompozicí tabulek users a user_roles. Tato entita udržuje vztahy mezi uživateli a jejich rolemi, které určují jejich možnosti v rámci webové aplikace.

user__roles Entita obsahující uživatelské role.

user__permissions Entita obsahující oprávnění pro jednotlivé uživatelské role.

user__roles__user__permission Entita vztahů, která vznikla dekompozicí entit user_roles a user_permissions. Tato entita udržuje vztahy mezi rolemi a jejich oprávněními. Každá uživatelská role (administrátor a supervizor) se skládá z různých oprávnění. Oprávněním může být třeba možnost zobrazit přehled hovorů, přehrát nahrávky, vytvářet nahrávací pravidla, atd.

Implementace

5.1 Použité technologie

V této kapitole budou představeny základní nástroje, díky kterým bylo možné celý nahrávací systém implementovat. Těmito nástroji jsou programovací a skriptovací jazyky, podpůrné knihovny a frameworky třetích stran, s jejichž využitím bylo možné provést implementaci mnoha částí systému jednodušeji a efektivněji.

5.1.1 Programovací a skriptovací jazyky

C++

Historie jazyka C++ sahá až do roku 1979, kdy v té době devětadvacetiletý Bjarne Stroustrup pracoval na své dizertační práci, pro kterou chtěl využít jazyk Simula 67, který byl považován za první objektově orientovaný model programování.

Stroustrup bohužel zjistil, že Simula 67 nedosahuje takových výkonů, jakých by potřeboval a že je pro praktické použití nevhodný. Krátce na to tedy začal pracovat na programovacím jazyku, kterému říkal "C with classes", tedy C s třídami. Jak název vypovídá, snažil se do existujícího jazyka C implementovat možnost objektově orientovaného programování. Jazyk C si vybral z důvodu, že byl a stále je velmi využívaný díky své přenositelnosti, aniž by musel být obětován výkon nebo možnost pracovat na nejnižší možné úrovni.

V roce 1983 byl jazyk přejmenován z „C with classes“ na C++. Jelikož je ++ v jazyce C operátor pro inkrementaci proměnné, dává nám tušit, jak Stroustrup na svůj programovací jazyk nahlížel. [15]

Kvůli vlastnostem, jako je vysoký výkon a možnost pracovat s daty na úrovni jednotlivých bitů, je jazyk C++ vhodný pro implementaci jádra systému.

Bash

Bash je konkrétním typem shellu pro GNU operační systémy. Jeho název je akronymem pro Bourne Again SHell, jelikož vychází z Bourne shellu. Shell je procesor, který interpretuje příkazový řádek.

Shell nejen, že interpretuje příkazový řádek, ale je zároveň programovacím, respektive skriptovacím jazykem. [16]

Shell může být spouštěn buď v interaktivním nebo neinteraktivním módu. V případě interaktivního módu jsou příkazy a vstup zadávány přímo do příkazové řádky. V neinteraktivním módu jsou příkazy spouštěny ze souboru. [16]

V této práci je bash využit k realizaci skriptu, který zpracovává soubory obsahující záznamy hlasu volajícího a volaného, a následně z nich vytváří jeden audio soubor, který je již považován za finální nahrávku hovoru.

Java

Java je objektově orientovaný jazyk, který byl vyvíjen začátkem devadesátých let minulého století společností Sun Microsystems. Vedoucím vývojového týmu byl James Gosling.

Na rozdíl od tradičních programovacích jazyků, které jsou kompilovány do nativního kódu nebo přímo interpretovány ze zdrojového kódu v běhovém prostředí, je Java určena ke kompilaci do byte kódu, který je následně spuštěn ve virtuálním prostředí JVM (Java Virtual Machine). [17]

Java svou syntaxí vychází z jazyka C/C++.

V této práci je v programovacím jazyku Java napsáno jádro webové aplikace. Z pohledu návrhového vzoru MVC se jedná konkrétně o vrstvy Model a Controller.

5.1.2 Použité knihovny

Knihovna oSIP

Knihovna oSIP je stabilní, přenositelný a flexibilní nástroj, který implementuje protokol SIP podle standardu RFC 3261.

Celá knihovna je napsána v jazyce C a pro svůj běh nevyžaduje žádné jiné závislosti nebo knihovny s výjimkou standardní C knihovny. Je thread-safe, což znamená, že využití ve vícevláknovém programu je bezpečné a nemůže docházet k žádným chybám. Toto je vzhledem k vícevláknové architektuře jádra systému velice důležité. [18]

Další výhodou oSIP knihovny je fakt, že má malý objem kódu a tím pádem i velikost celé knihovny je malá, pohybuje se totiž okolo 400 kB.

Knihovna je rozdělena na několik modulů. V této práci je využito modulů SIP message parser a SDP message parser. Těchto modulů je využito k parsování příchozích zpráv a sestavování odpovědí na příchozí zprávy.

Knihovna pqxx

Knihovna pqxx je oficiální C++ klientské API pro databázi Postgres, jež je open-source relační databáze. Knihovna nám umožňuje využívat databázových transakcí. V případě potřeby je ovšem přepnout do režimu autocommit, kde každý provedený příkaz je okamžitě potvrzený. [19]

Knihovny pqxx je využito při ukládání informací o jednotlivých hovorech do databáze Postgres a čtení nahrávacích pravidel v jádru systému.

5.1.3 Verzovací systém

Git je distribuovaný verzovací systém. Původně vznikl pro vývojáře linuxového jádra v roce 2005. Git byl vyvinut Linusem Torvaldsem po té, co se rozpadl vztah mezi komunitou vyvíjející jádro linuxu a společností, která dodávala proprietární distribuovaný verzovací systém BitKeeper pro tuto komunitu. Díky tomu, že se Linus Torvalds poučil z vlastností a chyb BitKeeperu, snažil se těmto chybám v implementaci gitu vyhnout. [20]

Verzovacího systému bylo využito při implementaci všech částí nahrávacího systému, od jádra až po webové uživatelské rozhraní. Umožňuje totiž jednoduše sledovat provedené změny nebo se v případě potřeby vrátit k určité starší verzi.

5.2 Jádro systému

Samotné jádro systému, tedy ta část nahrávacího systému, která zajišťuje komunikaci s telefonní ústřednou, zpracování přijatých dat a vytvoření výsledného souboru s nahrávkou hovoru je rozčleněna do několika logických celků. Tyto logické celky jsou obsluhovány pomocí jednotlivých pracovních vláken tak, jak je navrženo v sekci 4.1.

5.2.1 Struktura Call

Základním prvkem celého nahrávacího systému je hovor. Je tedy potřeba si pro lepší představu v dalších částech práce ukázat, jak je samotný hovor v jádru systému reprezentován a implementován. Hovor je reprezentován strukturou Call. Členské proměnné struktury jsou následující:

callId Je unikátním identifikátorem hovoru datového typu unsigned int. Tento identifikátor se nachází v hlavičce From požadavku INVITE, konkrétněji je parametrem SIP URI v této hlavičce a jeho název je x-refci.

nStreamFinished Příznak typu boolean určující, zda byla přijata zpráva SIP BYE pro proud hlasu volajícího.

fStreamFinished Význam této členské proměnné je stejný jako u členské proměnné `nStreamFinished` s rozdílem, že se jedná o proud hlasu volaného.

callingParty Identifikátor volajícího typu string. String z důvodu, že identifikátor nemusí obsahovat pouze telefonní číslo, ale může obsahovat SIP URI složenou např. ze jména a příjmení volajícího.

calledParty Význam této členské proměnné je stejný jako u členské proměnné `callingParty` s rozdílem, že se jedná o identifikátor volaného.

startTime Časová značka určující, kdy hovor začal. Tato značka je nastavena hned při vzniku nové instance struktury.

endTime Časová značka určující, kdy hovor skončil. Tato značka je nastavena v případě, že je právě nastavován druhý z příznaků `nStreamFinished` nebo `fStreamFinished`.

mediaType Číselný identifikátor označující druh kodeku, ve kterém jsou kódovány proudy hlasu.

5.2.2 Čtyři úrovně síťové komunikace

Síťová komunikace mezi nahrávacím systémem, telefonní ústřednou a nahrávaným koncovým zařízením probíhá z hlediska nahrávacího systému v principu na 4 různých úrovních. Tuto skutečnost již z části nastínila sekce 4.1.1.

Naslouchání na SIP TCP portu

Nahrávací systém a telefonní ústředna využívají pro veškerou SIP komunikaci dedikovaný přenosový spoj. V konfiguraci tohoto přenosového spoje na straně ústředny lze nastavit mimo jiné i cílovou IP adresu protistrany a zároveň cílový TCP port. Běžně se jedná o port 5060.

Na straně nahrávacího systému je možné tento port změnit nastavením direktivy `SIP_TCP_PORT` v konfiguračním souboru. Ve výchozí konfiguraci je nastavena na hodnotu 5060.

Vzhledem k tomu, že požadavky SIP INVITE iniciující nahrávání proudu hlasu přichází na tento cílový port, je nezbytné, aby prvním síťovým portem, který bude na straně nahrávacího systému otevřen, byl právě TCP port 5060.

Jakmile je síťový port otevřen, naslouchá systém v nekonečné smyčce, zda není iniciováno nové spojení od telefonní ústředny. Nekonečné naslouchání je potřebné z důvodu, že systém může obsluhovat více telefonních ústředen a TCP je spojově orientovaný protokol. To znamená, že každý nový klient vyžadující komunikaci, v tomto případě telefonní ústředna, si musí s nahrávacím systémem otevřít vlastní TCP spojení.

Příjem dat na SIP TCP portu

Po tom, co je sestaveno spojení mezi telefonní ústřednou a nahrávacím systémem, může ústředna vysílat SIP INVITE požadavky pro nahrání jednotlivých proudů hlasu právě probíhajícího hovoru na nahrávaném koncovém zařízení.

Příjem dat již probíhá mimo nekonečnou smyčku, ve které systém naslouchá požadavkům nových klientů. Přesněji je vytvořeno nové vlákno, kterému je předán klientský TCP soket (file descriptor, díky němuž je možné přistupovat k datům, která přichází na síťový port, případně data odesílat). Z tohoto soketu je již možné číst jednotlivá data, v tomto případě SIP INVITE požadavky.

Data jsou ze soketu vyzvedávána ve formě proudu dat. To znamená, že není možné vyzvedávat příchozí data po jednotlivých síťových paketech tak, jak dorazí, ale většinou jsou spojena data z více paketů a závisí na implementaci systému, jak velkou část příchozích dat ze soketu vyzvedne.

V závislosti na tom, jak velký objem dat dorazí od telefonní ústředny a jak velký objem dat je vyzvednut ze soketu, mohou nastat tři odlišné situace.

1. Ze soketu je vyzvednuto přesně tolik bytů dat, kolik bytů má SIP INVITE požadavek. To znamená, že je k dispozici celý SIP INVITE požadavek, se kterým lze dále pracovat. Tato situace je velmi málo pravděpodobná, jelikož objem dat, který je ze soketu vyzvedáván, je konstantní, zatímco délka požadavku je variabilní v závislosti na mnoha faktorech. Velikost vyzvedávacího bufferu je nastavena na 2000 B.
2. Ze soketu je vyzvednuto méně bytů dat, než je velikost příchozího SIP INVITE požadavku. To znamená, že se musí ze soketu načíst data další, ve kterých je většinou zbytek zprávy obsažen. S vyzvednutím zbytku dat může souviset situace třetí.
3. Ze soketu je vyzvednuto více bytů dat, než je samotná velikost příchozího SIP INVITE požadavku. V tomto případě jsou čtena vyzvednutá data do té doby, než je načten celý SIP INVITE požadavek. Jelikož by vyzvednutá data, která ovšem nebyla zpracována, chyběla, je nutné zkontrolovat tento zbytek dat, zda také není fragmentem příchozího požadavku, případně celou zprávou. Prakticky se dají na zbylá data aplikovat všechny tři zde popsané situace.

Ve skutečnosti je načtení příchozího požadavku rozděleno do dvou částí. A to načtení hlaviček a těla požadavku. Nejprve proběhne načtení hlaviček. Z hlavičky Content-Length je zjištěna velikost těla požadavku, jímž může být SDP zpráva. Pokud je hodnotou 0, znamená to, že požadavek neobsahuje žádné tělo a tím pádem byl již celý načten. Pokud byla zjištěna nenulová kladná hodnota, značí tato velikost SDP zprávy v bytech. Je tedy potřeba načíst buď ze soketu nebo již načtených, ale nezpracovaných dat přesně tolik

bytů, kolik bylo zjištěno z hlavičky Content-Length. Po přečtení těchto dat může být načtení požadavku prohlášeno za kompletní.

V případě nahrávání hovorů požadavek SIP INVITE neobsahuje SDP zprávu. Systém samozřejmě s přítomností SDP zprávy počítá. Potenciální data ale nejsou přidávána do těla požadavku. Jsou pouze odebírána z bufferu, případně soketu tak, aby bylo možné obsloužit další příchozí požadavky.

Odpověď na požadavek SIP INVITE může být dvojího typu v závislosti na záznamech definujících nahrávací pravidla. Pokud má být hovor nahrán, odpovídá nahrávací systém zprávou SIP 200 OK. Pokud hovor nahráván být nemá, je odpovědí zpráva SIP 603 DECLINE.

200 OK Na základě informací obsažených v hlavičkách požadavku je vytvořena odpověď SIP 200 OK, která navíc obsahuje SDP zprávu v níž jsou informace pro samotné nahrávané koncové zařízení. Těmito informacemi jsou mimo jiné IP adresa a cílový port, na které bude koncové zařízení zasílat proud hlasu. Další z informací je seznam kodeků, ve kterých smí zařízení vysílat proud hlasu tak, aby byl nahrávací systém data v tomto kodeku schopen zpracovat.

603 DECLINE Na základě informací obsažených v hlavičkách požadavku je vytvořena odpověď SIP 603 DECLINE. Tato odpověď neobsahuje žádné tělo. Telefonní ústředně dává informaci o tom, že nahrávání proudu hlasu nebude uskutečněno.

Naslouchání a příjem dat na SIP UDP portu

Stejně jako je potřeba otevřít na straně nahrávacího systému TCP port 5060, je potřeba otevřít i UDP port 5060. Tento port na straně serveru lze změnit nastavením direktivy SIP_UDP_PORT v konfiguračním souboru. Výchozí hodnotou je port 5060. Na tento port totiž přichází zprávy SIP ACK a SIP BYE. Případy, kdy nahrávací systém obdrží tyto zprávy jsou zobrazeny na obrázku 3.1.

Protokol UDP není na rozdíl od protokolu TCP spojově orientovaný, proto není potřeba dělit obsluhu příchozích dat na dvě části, jako tomu je u TCP.

Systém v nekonečné smyčce naslouchá požadavkům od telefonních ústředěn. Jakmile dorazí data, je systém schopen tato data ze soketu načíst do bufferu. Rozdíl oproti TCP je v tom, že data ze soketu jsou vyzvedávána po jednotlivých paketech, respektive s využitím dostatečně velkého bufferu mohou být. Pokud použijeme příliš malý buffer, bude načtena pouze část zprávy a dostáváme se do situace, která byla popsána u zpracování dat přijatých na TCP portu.

Pokud byla přijata zpráva SIP ACK, která obsahuje ve svém těle SDP zprávu, je potřeba tuto zprávu přečíst, jelikož nese informace o kodeku, který koncové zařízení používá ke kódování proudu hlasu. Bez této informace by nebylo možné přijatá data převést do potřebného kodeku a formátu.

Pro SIP BYE zprávu je vytvořena pouze SIP 200 OK odpověď obsahující základní hlavičky, která slouží jako potvrzovací. Pokud by systém tuto odpověď nevyslal, bude telefonní ústředna SIP BYE zprávy pro tentýž hovor posílat po určitých periodách znova v domnění, že nahrávací systém zprávu neobdržel.

Naslouchání a příjem hlasových dat na UDP portu

V jádru operačního systému je pro každý z proudů hlasu rezervován jeden náhodně vybraný volný UDP port. Na tomto portu jsou přijímána hlasová data odeslaná koncovým zařízením.

Proud hlasu je přenášen prostřednictvím RTP paketů. Každý z příchozích RTP paketů je reprezentován strukturou `RtpPacket`. Instance této struktury je inicializována daty načtenými ze soketu. Protože mají data strukturu RTP paketu, musejí být rozparsována přesně podle definice v RFC 3550 [14].

Přijatá data jsou reprezentována jako pole bytů a díky tomu je možné data číst po jednotlivých bytech.

Jeden byte RTP paketu může obsahovat více polí RTP hlavičky. Z toho důvodu je nutné jednotlivá pole extrahovat za pomoci bitových operací, kterými jsou bitový posun, součin i součet. Např. první byte RTP hlavičky nese čtyři různé informace.

Ze všech informací dostupných z hlaviček je možné spočítat počet bytů, které v RTP paketu zabírají hlasová data. Ta už je možné z bufferu jednoduše zkopírovat do členské proměnné.

Mimo samotná hlasová data je důležitým polem RTP hlavičky nesoucí informaci o typu přenášených dat. Toto pole může sloužit jako záložní v případě, že nahrávací systém neobdrží od telefonní ústředny zprávu SIP ACK, v jejímž těle je obsažena SDP zpráva obsahující totožnou informaci.

Protokol UDP nemá na rozdíl od TCP implementován žádný potvrzovací mechanismus o přijetí zpráv. Mimo přijetí všech dat ve správném pořadí mohou tedy nastat dvě situace. Data mohou být přijata buď ve špatném pořadí, nebo dokonce vůbec.

Pořadové číslo paketu je obsaženo v poli „sequence number“ RTP hlavičky a pro každý další paket je inkrementováno o jedničku. [14]

Pokud dorazí paket mimo pořadí, musí být odložen do pomocné datové struktury, která tyto pakety dočasně uloží a zpřístupní je v době, kdy budou potřeba. Zároveň je v této situaci spuštěn odpočet trvající 1000 ms. Pokud do konce odpočtu není přijat paket s požadovaným pořadovým číslem, je prohlášen za ztracený a místo něj je použit paket s nejnižším pořadovým číslem, který je dostupný v pomocné struktuře.

Paket, který byl již prohlášen za ztracený, může být ovšem přijat i s delším zpožděním, než je deklarovaných 1000 ms. V takovém případě je paket zahozen.

Vzhledem k vzorkovací frekvenci kodeků, která je 64 kbps, je možné během jedné sekundy přijmout 8 KB hlasových dat. Režie na hlavičky RTP paketu je zanedbatelná. Proto by měla být pomocná struktura schopna uchovat minimálně 8 KB hlasových dat, aby bylo možné udržet pakety při výpadku jednoho paketu za sekundu.

Pomocná struktura je realizována za pomoci asociativního kontejneru `mapa`. Klíčem je pořadové číslo paketu a mapovanou hodnotou je objekt typu `RtpPacket` reprezentující paket s daným pořadovým číslem.

Extrahovaná hlasová data jsou ukládána do souboru po 8 KB blocích, nikoli po blocích obsažených v jednom paketu. Důvodem je vysoká režie na zápis do souboru a vysoký počet vstupně-výstupních operací.

Každý soubor, který je takto vytvořen a je již kompletní, obsahuje právě jeden proud hlasu. Je pojmenován unikátním identifikátorem hovoru a zároveň je do názvu přidána zkratka pro označení, zda se jedná o proud hlasu volajícího či volaného. Tím je zajištěno, že hlas volajícího i volaného bude u každého nahraného hovoru vycházet vždy ze stejného audio kanálu (levého a pravého).

5.2.3 Kontrola nahrávacích pravidel

Součástí zadání práce je umožnit prostřednictvím webové aplikace tvorbu nahrávacích pravidel. Část webové aplikace, ve které je možné definovat nahrávací pravidla, je zpřístupněna pouze uživatelům, kteří mají v aplikaci roli správce. Uživatel je z pohledu systému pouze konzument obsahu. Tímto obsahem jsou nahrávky hovorů, a proto má uživatel možnost nahrávky procházet, filtrovat a především přehrávat.

Identifikátorem volajícího a volaného může být buď telefonní číslo nebo SIP URI. Vzor identifikátoru musí být možné popsat i na obecnější úrovni, než jen výčtem jednotlivých telefonních čísel. Pro řešení tohoto problému poslouží regulární výrazy. Regulární výrazy nám umožní popsat i obecnější vzory, jako je např. rozsah telefonních čísel, SIP URI pouze s určitou doménou, atd.

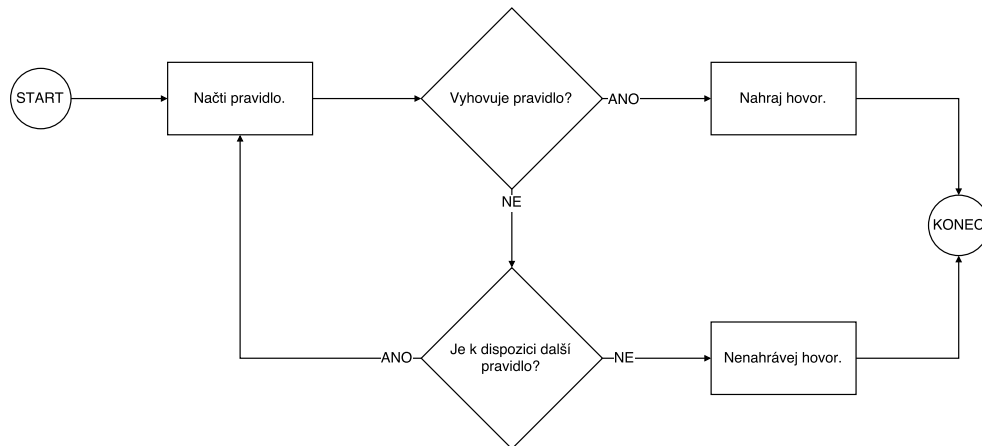
Nahrávací pravidla vytvářejí seznam a jsou vyhodnocována na základě první shody.

Každé nahrávací pravidlo se skládá z následujících částí.

- Regulární výraz, který popisuje identifikátor volajícího.
- Regulární výraz, který popisuje identifikátor volaného.
- Časový interval, který určuje, v jakém časovém úseku dané pravidlo platí.
- Pořadové číslo, které určuje pořadí pravidla v seznamu.

Nahrávací pravidla jsou vyhodnocována ihned, jakmile je dostupný celý požadavek SIP INVITE. Tento požadavek obsahuje v hlavičce `From` parametry

x-nearendaddr a x-farendaddr, jejichž hodnotou je identifikátor volajícího a volaného.



Obrázek 5.1: Vývojový diagram kontroly nahrávacích pravidel.

5.2.4 Kontrola aktuálně nahrávaných hovorů

Všechny aktuálně nahrávané hovory jsou uloženy v asociativním kontejneru typu `mapa`. Klíčem je unikátní identifikátor hovoru a mapovanou hodnotou je objekt typu `Call` reprezentující hovor s daným identifikátorem.

Mapa nahrávaných hovorů je v pravidelných intervalech pěti sekund lineárně procházena a u každého z hovorů je kontrolováno, zda nemá nastaveny oba příznaky `nStreamFinished` a `fStreamFinished`. Pokud ano, znamená to, že příjem proudů hlasu volajícího a volaného byl ukončen. Takové hovory jsou přesunuty z mapy do kontejneru typu `fronta`, odkud jsou již na principu odebírány a jsou vytvářeny výsledné soubory s nahrávkami.

5.2.5 Zpracování nahraných hovorů

Když fronta obsahuje hovory, u kterých již bylo ukončeno nahrávání, jsou tyto hovory zpracovávány.

Nejprve jsou do databáze uloženy všechny základní informace o hovoru. Těmito informacemi je identifikátor volajícího i volaného a časové razítko určující začátek a konec hovoru. Pro přístup k databázi je využita knihovna `pqxx`.

Následně je spuštěn externí skript, který vytvoří výslednou nahrávku.

5.2.6 Vytvoření výsledného souboru s nahrávkou.

Každý záznam hovoru je bezprostředně po ukončení nahrávání reprezentován dvěma soubory. Jeden soubor obsahuje záznam proudu hlasu volajícího a druhý záznam proudu hlasu volaného. Data v těchto souborech jsou kódována kodekem G.711 nebo G.722. Jsou to surová audio data, která nejsou obohacena o žádná metadata.

Oba soubory je potřeba sloučit, zkomprimovat a převést do některého ze známých audio formátů. Tyto operace jsou prováděny v bash skriptu, kterému je v argumentech předán identifikátor hovoru a typ kodeku, ve kterém jsou zdrojová data. Tento skript je volán z jádra nahrávacího systému. Veškeré konverze a vytváření výstupního souboru jsou prováděny prostřednictvím programu `avconv`. Programu `avconv` jsou dvojnásobným použitím přepínače `-i` zadány vstupní soubory pro zpracování. Ke každému z nich je ještě určen kodek, ve kterém jsou vstupní data kódována. Přepínačem `-filter_complex` je nastaven filtr, který říká, co se s danými vstupy má provést. V tomto případě je nastaven filtr `join`, který zajistí, že každý ze vstupního proudu hlasu je ve výsledném souboru pouze v jednom ze dvou audio kanálů. Argumentem programu je název výstupního souboru i s příponou `.mp3`. Díky přítomné příponě program `avconv` automaticky určí, v jakém formátu soubor uložit.

5.2.7 Logování

Logování je velmi důležitou součástí systému. Logování je zprostředkováno pomocí nástroje Syslog a pro jeho použití je potřeba do zdrojového kódu vložit hlavičkový soubor `syslog.h`. Zprávu lze zalogovat zavoláním funkce s následujícím předpisem.

```
void syslog (int priority, const char* format, ...);
```

priority Určuje úroveň, s jakou má být zpráva zalogována.

format Definuje formát zprávy. Formátovací řetězce jsou stejné jako u funkce `printf(3)`.

zbylé parametry slouží jako hodnoty pro formátovací řetězec.

Ve výchozí konfiguraci jsou zprávy logovány do souboru `/var/log/syslog` a jsou označeny názvem programu.

5.2.8 Konfigurace systému

Systém je možné konfigurovat prostřednictvím konfiguračního souboru. Relevantní řádky mají formát *direktiva = hodnota*. Řádky obsahující bílé znaky

nebo prázdné řádky jsou ignorovány, stejně tak jako řádky začínající znakem „#“, které jsou chápány jako komentáře. Všechny dostupné konfigurační direktivy jsou definovány a popsány ve výchozím konfiguračním souboru.

Mezi možné konfigurační direktivy patří např. adresa databázového serveru včetně údajů o uživateli, číslech jednotlivých síťových portů, atd.

5.3 Uživatelské rozhraní

Uživatelské rozhraní je implementováno jako webová aplikace.

Aplikace je určena pro dva typy uživatelů. Jedním je supervizor kontaktního centra a druhým administrátor nahrávacího systému. Supervizor má možnost v aplikaci zobrazit přehled nahraných hovorů a pro každý hovor přehrát nahrávku. Administrátor vzhledem k citlivosti dat nemá možnost přístupu k přehledu hovorů ani samotným nahrávkám, ale je mu zpřístupněno prostředí, ve kterém definuje nahrávací pravidla.

5.3.1 Backend uživatelského rozhraní

Backend webové aplikace je kompletně napsaný v jazyce Java s využitím frameworku Hibernate.

5.3.1.1 Práce s daty

Přístup k datům, která jsou uložena v databázi, je zajištěn prostřednictvím frameworku Hibernate. Tento framework zajišťuje objektově-relační mapování a persistentní ukládání dat. Objektově-relační mapování umožňuje mapovat záznamy v tabulkách relační databáze na objekty reprezentující tyto záznamy v aplikaci.

Webová aplikace využívá přístupu prostřednictvím Hibernate k oběma tabulkám databáze, kterými jsou `calls` a `recording_rules`.

Z tabulky `calls` jsou data pouze čtena pro zobrazení přehledu nahraných hovorů.

Nad tabulkou `recording_rules` jsou již prováděny všechny příkazy pro manipulaci s daty. Těmi jsou příkazy pro čtení, vkládání, úpravu a mazání dat.

Veškeré operace nad daty jsou zpřístupněny prostřednictvím REST rozhraní. REST rozhraní umožňuje tvorbu dalších aplikací, které budou moci přistupovat k datům nahrávacího systému. Takovou aplikací může být např. mobilní aplikace, případně jiné aplikace třetích stran, do kterých budou data z nahrávacího systému zakomponována. Přenášená data jsou ve formátu JSON.

5.3.2 Frontend uživatelského rozhraní

Frontend webové aplikace je vytvořen za pomoci značkovacího jazyka HTML a skriptovacího jazyka JavaScript. Pro implementaci byl využit javascriptový

framework Sencha. Veškeré prvky webové aplikace jsou již ve formě hotových komponent, které lze použít. Přehled komponent je dostupný na [21].

S využitím frameworku Sencha je získává webová aplikace data prostřednictvím REST rozhraní.

5.3.2.1 Přehled hovorů

Nejdůležitějším prvkem webové aplikace je stránka, ve které je přehled všech nahraných hovorů. Přehled je reprezentován komponentou Grid, která představuje tabulku. Má tedy definovány jednotlivé sloupce téměř jako databázová tabulka.

Dalším prvkem na stránce s přehledem hovorů je formulář, který umožňuje vyhledávání nahrávek. Nahrávky lze vyhledávat na základě čísla volajícího nebo volaného a začátku nebo konce hovoru. Formulář je realizován prostřednictvím komponenty Form.

5.3.2.2 Nahrávací pravidla

Stránka zobrazující seznam nahrávacích pravidel obsahuje také tabulku s jednotlivými nahrávacími pravidly. Opět je realizována komponentou Grid z frameworku Sencha.

Každý řádek tabulky představuje jedno nahrávací pravidlo a skládá se ze stejných atributů jako databázová tabulka. Pravidla může administrátor prostřednictvím této stránky přidávat, upravovat i mazat.

5.3.3 Bezpečnost

Osobní údaje a nahrávky hovorů jsou velmi citlivá data. Proto nemohou být přístupná prostřednictvím webové aplikace bez jakéhokoli ověření přístupu. Ověření do aplikace probíhá prostřednictvím uživatelských údajů, které mohou být uloženy buď v lokální databázi nahrávacího systému nebo v adresářovém serveru využívají protokolu LDAP. Tím může být i adresářový server ActiveDirectory od společnosti Microsoft, který je často využíván pro správu uživatelských účtů a přístupů ve společnosti.

Testování

Testování systému proběhlo v produkčním kontaktním centru, ve kterém je během jednoho dne obslouženo přibližně 500 hovorů.

S pomocí výpisu všech hovorů z telefonní ústředny byl překontrolován počet nahraných oproti reálně uskutečněným hovorům. Dále byla překontrolována délka nahrávek a u náhodně vybraných hovorů, zda soubory s nahrávkou opravdu obsahují záznam hovoru.

Nejprve fungoval systém jeden pracovní týden jako nahrávací server pro operátory kontaktního centra. Během celého týdne nedošlo v systému k žádné závažné chybě, která by měla za následek zhavarování celého systému. U malého počtu hovorů ovšem nebylo nahrání úspěšné. Důvodem byla špatná práce s přijatými daty signalizace. Nebyly rozpoznány některé SIP INVITE požadavky. Tato chyba byla opravena a systém byl v kontaktním centru nasazen na další pracovní týden. Počet nahraných a reálně obslužených hovorů z tohoto týdne už byl stejný. Zároveň u náhodně vybraných hovorů byly všechny nahrávky kompletní.

Proveden byl i zátěžový test, ve kterém systém také obstál. Systém musel v jednu chvíli nahrávat 50 souběžných hovorů. Pro tento test byl napsán skript, program, který prostřednictvím JTAPI inicioval v jednu chvíli hovory z koncových zařízení. Hovory byly směrovány na různé bílé linky, kde je systém IVR a na straně telefonů, ze kterých byly hovory iniciovány, byla z rádia pouštěna hudba, aby proud hlasu neobsahoval jen ticho.

Závěr

Cílem této bakalářské práce bylo vytvořit systém, který umožní prostřednictvím funkcionality BIB nahrávat VoIP hovory v prostředí, kde je pro VoIP využíváno technologií společnosti Cisco. Součástí systému mělo být také uživatelské prostředí, prostřednictvím kterého by bylo možné jednotlivé nahrávky procházet a přehrávat. Dalším prvkem uživatelského prostředí mělo být rozhraní pro vytváření nahrávacích pravidel.

Dalším cílem bylo analyzovat již existující nahrávací systémy a vyvodit z nich závěr pro implementaci vlastního nahrávacího systému.

Vlastní implementace nahrávacího systému splňuje všechny požadavky zadání práce. Výsledný systém jako celek umožňuje nahrávání VoIP hovorů požadovanou metodou a zároveň jednotlivé nahrávky zpřístupňuje prostřednictvím webové aplikace. Webová aplikace také umožňuje tvorbu nahrávacích pravidel.

Aplikace může být v budoucnu rozšířena o další nahrávací metody. Tím je myšlena např. metoda, při které jsou proudy hlasů hovoru zasílány z hlasové brány do nahrávacího systému. Zároveň je možné systém navázat na informační systémy, které jsou využívány v prostředí, ve kterém je nahrávací systém nasazen. Díky tomu by přehled hovorů mohl místo samotných identifikátorů komunikujících zobrazovat jejich jména.

Literatura

- [1] Eurotel Praha, s r. o.: Šablona pro specifikaci požadavků [online]. 2003, [Cit. 2016-04-01]. Dostupné z: http://www.volere.co.uk/pdffiles/template_cz.pdf
- [2] Cisco inc.: What is VoIP [online]. [Cit. 2016-04-01]. Dostupné z: http://www.cisco.com/c/en/us/products/unified-communications/networking_solutions_products_genericcontent0900aecd804f00ce.html
- [3] IETF: Session Initiation Protocol [online]. 2002, [Cit. 2016-04-03]. Dostupné z: <https://www.ietf.org/rfc/rfc3261.txt>
- [4] IETF: Real-Time Transport Protocol (RTP) Parameters [online]. 2016, [Cit. 2016-04-26]. Dostupné z: <https://www.ietf.org/assignments/rtp-parameters/rtp-parameters.xml>
- [5] IETF: Session Description Protocol [online]. 1998, [Cit. 2016-05-01]. Dostupné z: <https://www.ietf.org/rfc/rfc2327.txt>
- [6] Cisco inc.: Unified CM Call Monitoring and Recording [online]. 2016, [Cit. 2016-05-01]. Dostupné z: http://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/srnd/collab10/collab10/recordng.html#33479
- [7] ZOOM International s.r.o.: Cisco Call Recording Architecture [online]. 2016, [Cit. 2016-05-04]. Dostupné z: <https://docs.zoomint.com/display/QMS5/Cisco+Call+Recording+Architecture#CiscoCallRecordingArchitecture-CiscoRecordingMethods>
- [8] ZOOM International s.r.o.: ZOOM Customers [online]. 2016, [Cit. 2016-05-04]. Dostupné z: <http://www.zoomint.com/customers>

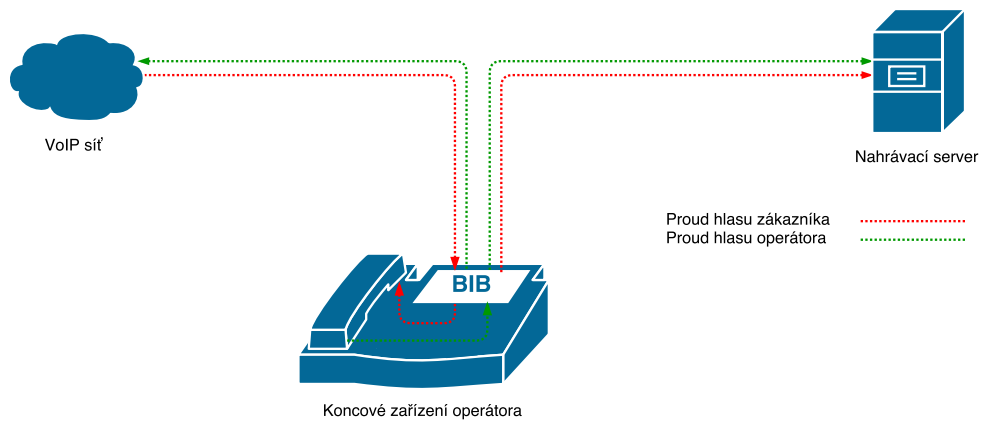
- [9] ZOOM International s.r.o.: ZOOM Solutions [online]. 2016, [Cit. 2016-05-29]. Dostupné z: <http://www.zoomint.cz/produkty/prehled-reseni>
- [10] MiaRec Inc.: Developer guide [online]. 2016, [Cit. 2016-05-29]. Dostupné z: <https://www.miarec.com/doc/developer-guide/>
- [11] Cisco Inc.: Cisco MediaSense [online]. 2016, [Cit. 2016-05-30]. Dostupné z: http://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/srnd/collab10/collab10/recordng.html#41499
- [12] Cisco Inc.: Features and Services Guide for Cisco Unified Communications Manager [online]. 2016, [Cit. 2016-06-02]. Dostupné z: http://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/admin/10_0_1/ccmfeat/CUCM_BK_F3AC1C0F_00_cucm-features-services-guide-100/CUCM_BK_F3AC1C0F_00_cucm-features-services-guide-100_chapter_0101010.html#CUCM_RF_R8A92E56_00
- [13] Tomáš, M.: Vlákna vs. procesy [online]. 2009, [Cit. 2016-06-02]. Dostupné z: <http://tojaj.com/vlakna-vs-procesy/>
- [14] IETF: RTP: A Transport Protocol for Real-Time Applications [online]. 2003, [Cit. 2016-06-03]. Dostupné z: <https://tools.ietf.org/html/rfc3550>
- [15] Albatross: History of C++ [online]. 2016, [Cit. 2016-06-03]. Dostupné z: <http://www.cplusplus.com/info/history/>
- [16] Bash Reference Manual [online]. 2016, [Cit. 2016-06-04]. Dostupné z: <https://www.gnu.org/software/bash/manual/bashref.html>
- [17] History of Java programming language [online]. 2016, [Cit. 2016-06-04]. Dostupné z: <http://www.freejavaguide.com/history.html>
- [18] Aymeric, M.: GNU oSIP library [online]. 2012, [Cit. 2016-06-05]. Dostupné z: <https://www.gnu.org/software/osip/>
- [19] jtv: libpqxx [online]. 2016, [Cit. 2016-06-05]. Dostupné z: <http://pqxx.org/development/libpqxx/>
- [20] Chacon Scott, S. B.: A Short History of Git [online]. 2016, [Cit. 2016-06-05]. Dostupné z: <https://git-scm.com/book/en/v2/Getting-Started-A-Short-History-of-Git>
- [21] Sencha Inc.: Ext JS Kitchen Sink [online]. 2016, [Cit. 2016-06-05]. Dostupné z: <http://examples.sencha.com/extjs/6.0.2/examples/kitchensink/#all>

Seznam použitých zkratek

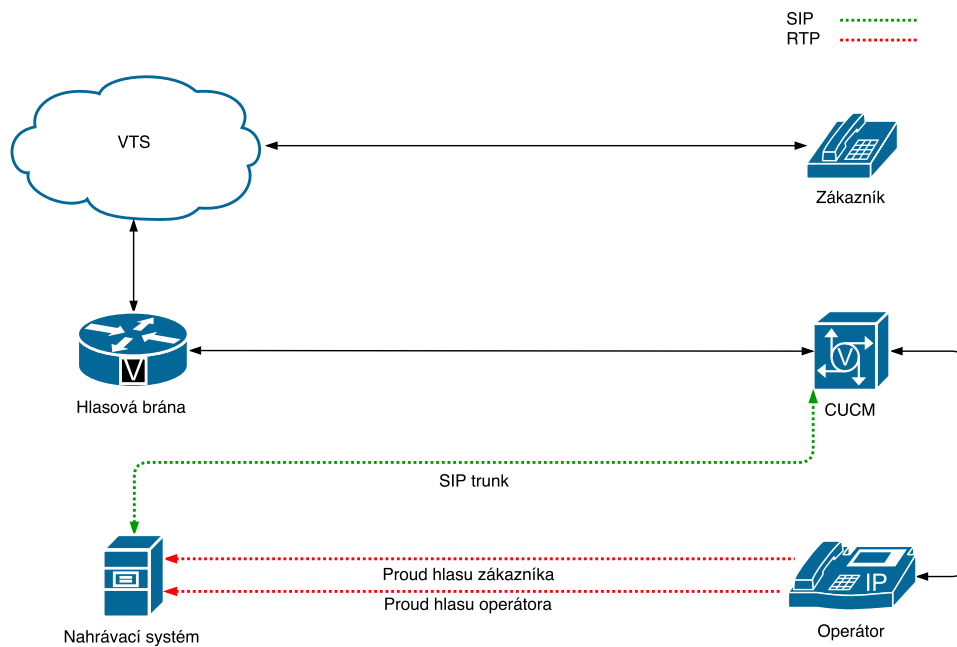
API	Application Programming Interface
BIB	Built-in-Bridge
JTAPI	Java Telephony Application Programming Interface
IP	Internet Protocol
IVR	Interactive Voice Response
QoS	Quality of Service
REST	Representational State Transfer
RTP	Real-time Transport Protocol
SCCP	Skinny Client Control Protocol
SDP	Session Description Protocol
SIP	Session Initiation Protocol
TCP	Transmission Control Protocol
UA	User Agent
UAC	User Agent Client
UAS	User Agent Server
UDP	User Datagram Protocol
URI	Uniform Resource Identifier
VoIP	Voice over Internet Protocol
VTS	Veřejná Telefonní Síť

Komunikační diagramy

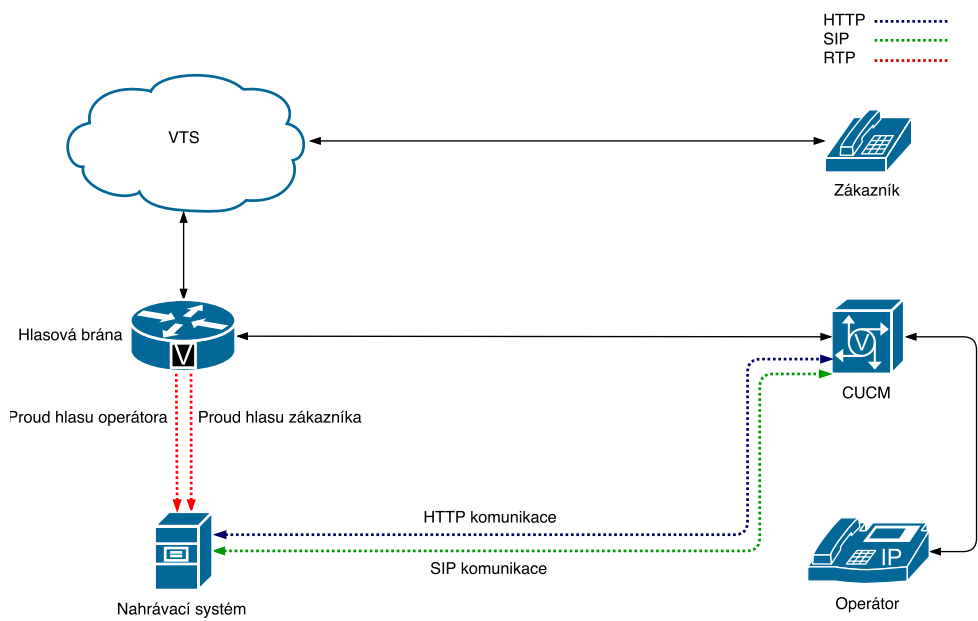
B. KOMUNIKAČNÍ DIAGRAMY



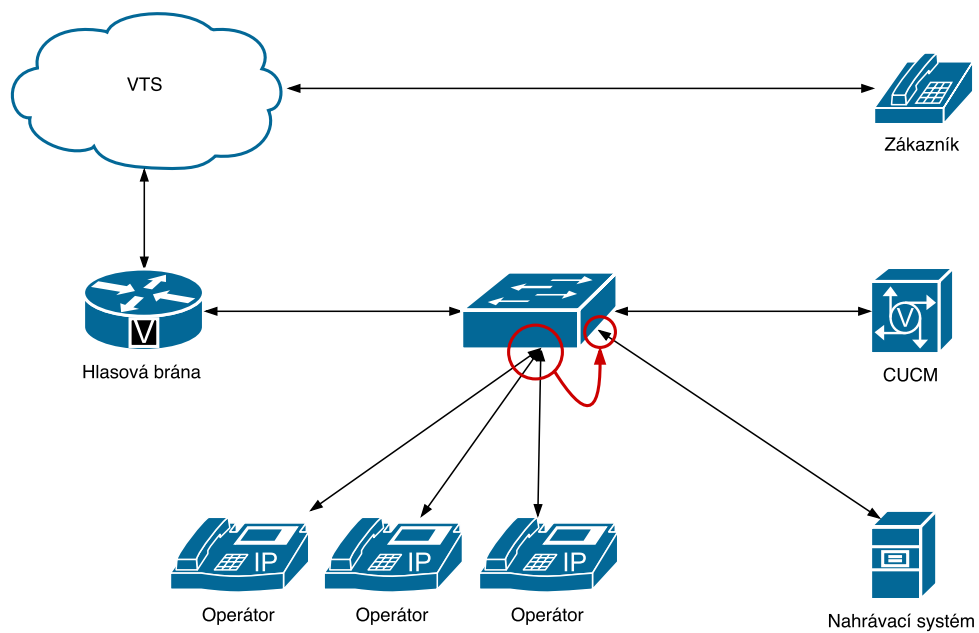
Obrázek B.1: Hlasové proudy v nahrávaném koncovém zařízení s BIB.



Obrázek B.2: Aktivní nahrávání s využitím BIB. [6]



Obrázek B.3: Aktivní nahrávání s využitím hlasové brány. [6]



Obrázek B.4: Pasivní nahrávání. [6]

Instalace systému

Pro instalaci a fungování systému je potřeba kompilátor jazyka C++. Kompilace v rámci vývoje systému byla prováděna kompilátorem g++ 4.9.2. Dále je potřeba program `make` a `avconv`. Dalšími knihovnami potřebnými ke kompilaci programu jsou knihovny `oSIP2 4.1.0`, `pqxx 4.0.1` a `pthread`. Pro spuštění instalace systému stačí spustit programem `make` soubor `Makefile`. Podrobný postup instalace jádra a webové aplikace systému je v souboru `ReadMe.txt`. `Dát.le` je obsažen `create skript` pro vytvoření databáze.

Obsah přiloženého CD

readme.txt.....	stručný popis obsahu CD
src	
├ BP_Miozga_David_2016.tex..	zdrojová forma práce ve formátu $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$
├ img.....	obrázky
└ *	ostatní soubory šablony
text.....	zadání a text práce
├ BP_Miozga_David_2016.pdf	text práce ve formátu PDF
└ ZZP_Miozga_David_2016.pdf	zadání práce ve formátu PDF