

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Fakulta elektrotechnická

Katedra telekomunikační techniky



## Sledování místností IP telefonem

Premises surveillance via IP phone

**Vedoucí práce:** Ing. Pavel Bezpalec, Ph.D.

**Bakalant:** Thanh Tuan Nguyen

Praha, leden 2019



## Čestné prohlášení

Prohlašuji, že jsem zadanou bakalářskou práci zpracoval sám pod vedením vedoucího práce a používal jsem literaturu v práci uvedenou. Dále prohlašuji, že nemám námitek proti půjčování nebo zveřejňování mé bakalářské práce nebo její části se souhlasem katedry.

V Praze dne 8. 1. 2019

.....  
Thanh Tuan Nguyen







# ZADÁNÍ BAKALÁŘSKÉ PRÁCE

## I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Nguyen** Jméno: **Thanh Tuan** Osobní číslo: **457115**  
Fakulta/ústav: **Fakulta elektrotechnická**  
Zadávající katedra/ústav: **Katedra telekomunikační techniky**  
Studijní program: **Komunikace, multimédia a elektronika**  
Studijní obor: **Síťové a informační technologie**

## II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

**Sledování místností IP telefonem**

Název bakalářské práce anglicky:

**Premises surveillance via IP phone**

Pokyny pro vypracování:

Analyzujte mechanismy zpracování analogového AV signálu z kamery pomocí embedded systémů. Na základě analýzy zpracujte návrh a realizujte zobrazení videa z dodané analogové kamery na IP telefonním přístroji Cisco.

Seznam doporučené literatury:

Perkins, C.: RTP: Audio and Video for the Internet. Addison-Wesley Professional, June 2003. ISBN: 978-0-67232-249-5.  
Wilson, RW. Embedded Systems and Computer Architecture. Newnes; 1 edition. January, 2002. ISBN: 978-0750650649

Jméno a pracoviště vedoucí(ho) bakalářské práce:

**Ing. Pavel Bezpalec, Ph.D., katedra telekomunikační techniky FEL**

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **26.09.2018** Termín odevzdání bakalářské práce: \_\_\_\_\_

Platnost zadání bakalářské práce: **19.02.2020**

Ing. Pavel Bezpalec, Ph.D.  
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Ing. Pavel Ripka, CSc.  
podpis děkana(ky)

## III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

\_\_\_\_\_  
Datum převzetí zadání

\_\_\_\_\_  
Podpis studenta



## **Anotace**

Tato bakalářská práce se zabývá zobrazováním obrazu z analogové kamery v Cisco IP telefonu řady 7900. Za tímto účelem bylo použito Raspberry Pi, na kterém se odehrála podstatná část práce. Obraz byl získán z analogové kamery od firmy Sony a zprostředkován prostřednictvím zachytávacího zařízení EasyCap do Raspberry Pi. Zpracování analogového obrazu bylo dále řešeno v Raspberry Pi pomocí Video4linux2, Videobuf2 a nástroje ffmpeg. Cisco IP telefon 7971G patří k modelům, které jsou vybaveny barevným displejem. Tento telefon podporuje tzv. Cisco Unified IP Phone Services, tedy speciálně navržené webové služby, které si mohou uživatelé vyžádat a zobrazit na telefonu. Díky této vlastnosti bylo možné napsat pro telefon službu, která umožní uživatelům sledovat např. objekt nebo místnost na svém Cisco IP telefonu. Služba byla vytvořena využitím open-source softwarů jako Apache a Asterisk.

## **Klíčová slova**

Raspberry Pi, Cisco IP telefon, Asterisk, XML, Video4linux2, ffmpeg

## **Abstract**

This bachelor thesis deals with a topic of viewing images from analog camera on Cisco IP phones 7900 series. Raspberry Pi was used to fulfill this purpose. The major part of this thesis was performed on Raspberry Pi. Video was made with Sony analog camera and captured by EasyCap Video Grabber. The images were processed on Raspberry Pi using Video4linux2, Videobuf2 and ffmpeg. Cisco IP phone 7971G belongs among models that are equipped with a color display. This phone supports Cisco Unified IP Phone Services, which are specially designed web services, that can be requested by users to view information on a display. Thanks to this capability, it was possible to write a service for this phone that allowed users, for example, object or room monitoring on their Cisco Ip phone. The service was created utilizing open-source software solutions like Apache and Asterisk.

## **Keywords**

Raspberry Pi, Cisco IP phone, Asterisk, XML, Video4linux2, ffmpeg



## Poděkování

Děkuji vedoucímu práce Ing. Pavlu Bezpalcovi, Ph.D. za velmi užitečnou metodickou pomoc a cenné rady při zpracování této bakalářské práce.

Dále bych chtěl poděkovat celé mé rodině včetně přítelkyně za podporu, trpělivost a povzbuzování při psaní této práce.



# Obsah

<b>1</b>	<b>Úvod</b>	<b>13</b>
<b>2</b>	<b>Teoretický rozbor</b>	<b>15</b>
2.1	Raspberry Pi	16
2.1.1	Raspbian	16
2.2	Analogová kamera	16
2.2.1	Sony EVI-D31	16
2.2.2	Kompozitní video	17
2.2.3	EasyCap USB 2.0 Video grabber	18
2.3	Video4linux2 API	18
2.4	Videobuf2	19
2.5	FFmpeg	19
2.5.1	ffmpeg	20
2.6	Cisco IP telefon 7971G	21
2.6.1	XML objekty	22
2.6.2	TFTP	24
2.6.3	Konfigurační soubor pro Cisco IP telefon	24
2.7	Asterisk	25
2.7.1	Architektura ústředny Asterisk	25
2.7.2	Číslovací plán	25
2.7.3	SCCP protokol	26
2.8	Webová služba	26
2.8.1	PHP	26
2.8.2	Apache	26
2.8.3	HTTP komunikace	27
<b>3</b>	<b>Praktická část</b>	<b>28</b>
3.1	Zprovoznění Raspberry Pi	28
3.1.1	Instalace Raspbianu Stretch	28
3.1.2	Síťová konfigurace	29
3.2	Zprovoznění kamery	29
3.2.1	Ovladač pro Fushicai USBTV007 Audio-Video Grabber	30
3.2.2	Instalace ffmpeg	30
3.3	Instalace a konfigurace ústředny Asterisk	32
3.3.1	Instalace chan_sccp	33
3.3.2	Konfigurační soubory ústředny asterisk	34
3.3.3	Registrace SCCP klienta do ústředny	34
3.4	Konfigurace Cisco IP telefonu	35
3.5	Webová služba	36
3.5.1	Instalace a nastavení Apache2	37
3.5.2	Instalace PHP 7	38
3.5.3	Služba Camera	38
<b>4</b>	<b>Vyhodnocení</b>	<b>41</b>
<b>5</b>	<b>Literatura</b>	<b>42</b>

## Seznam obrázků

Obr. 1	Raspberry Pi 3B+ . . . . .	16
Obr. 2	Sony EVI-D31 . . . . .	17
Obr. 3	RCA pin jack . . . . .	17
Obr. 4	EasyCap USB 2.0 Video grabber . . . . .	18
Obr. 5	Schéma transkódování . . . . .	20
Obr. 6	Schéma simple filtergraph . . . . .	21
Obr. 7	Schéma complex filtergraph . . . . .	21
Obr. 8	Maximální parametry PNG snímku pro jednotlivé Cisco IP telefony řady 7900	22
Obr. 9	Cisco IP telefon 7971G . . . . .	22
Obr. 10	Struktura ústředny Asterisk . . . . .	25
Obr. 11	HTTP komunikace mezi telefonem a serverem . . . . .	27
Obr. 12	Pracovní sestava . . . . .	28
Obr. 13	Zapisovací nástroj Etcher . . . . .	29
Obr. 14	Pixel desktop prostředí . . . . .	29
Obr. 15	Výstup <code>lsusb</code> . . . . .	29
Obr. 16	Výstup <code>lsmod</code> . . . . .	30
Obr. 17	Informace o modulu <code>usbtv</code> . . . . .	30
Obr. 18	Výpis z nástroje <code>ffmpeg</code> . . . . .	31
Obr. 19	Asterisk command line interface (CLI) . . . . .	33
Obr. 20	Stav modulu <code>chan_sccp</code> . . . . .	34
Obr. 21	Průběh registrace telefonu jako SCCP klient . . . . .	35
Obr. 22	Průběh nastavení a registrace telefonu . . . . .	36
Obr. 23	Seznam registrovaných SCCP klientů . . . . .	36
Obr. 24	Průběh komunikace mezi telefonem a servery . . . . .	37
Obr. 25	Práva pro soubor <code>video0</code> . . . . .	39
Obr. 26	Nastavení v <code>sccp.conf</code> . . . . .	40
Obr. 27	Tlačítko pro spuštění služby Camera . . . . .	40
Obr. 28	Služba kamera . . . . .	40



# 1 Úvod

Tato bakalářská práce se zabývá tématem zobrazení obrazu z analogové kamery v Cisco IP telefonu řady 7900. Jeho využití se nabízí jako možnost rozšíření kamerového bezpečnostního systému. Na základě konzultací s panem vedoucím byly stanoveny cíle: analýza zpracování analogového AV signálu z kamery pomocí embedded systémů a realizace zobrazení obrazu z dodané analogové kamery na Cisco IP telefonu.

Za tímto účelem byla dodaná analogová kamera Sony EVI-D31. Analogové kamery nenabízejí tak vysokou kvalitu obrazu jako IP kamery, ale vzhledem k zobrazovací schopnosti displeje telefonu by byla tato přednost IP kamer nevyužitá. Analogové kamery přináší výhodu jak v cenové dostupnosti, tak v možnosti využití starších kamerových systémů.

Zpracování analogového signálu byla realizována na Raspberry Pi pomocí open-source nástrojů. Raspberry Pi 3 Model B+ nabízí dostatečný výkon pro dosažení našich cílů.

V práci byl použit Cisco IP telefon 7971G. Přestože jeho displej má jistá omezení, pro účel zobrazení sekvence snímků z analogové kamery je dostatečný a umožní uživatelům sledovat např. místnost nebo prostor před vstupem do domu.



## 2 Teoretický rozbor

V této části jsou sepsány informace a poznatky o zařízeních, nástrojích a technologiích, které byly získány z různých zdrojů. Na základě těchto dostupných informací byla realizována praktická část. Teoretický rozbor je rozdělen do následujících kapitol.

První kapitola je věnována operačnímu systému Raspbian a Raspberry Pi 3 Model B+, kde je přiblížena jeho hardwarová výbava a vylepšení oproti předchozímu modelu.

Druhá kapitola popisuje analogovou kameru od firmy Sony, kompozitní video, standard PAL a video/audio zachytávací zařízení EasyCap.

O rozhraní jádra pro video v Linuxu - Video4linux2 a o frameworku pro ovladače multimediálních zařízení známém jako Videobuf2 je napsáno ve třetí a čtvrté kapitole.

Hlavní náplní páté kapitoly je popis nástroje ffmpeg a jeho mechanismu video konverze. Dále byl zmíněn způsob ovládání ffmpeg v příkazovém řádku.

Šestá a sedmá kapitola se zaměřují na Cisco IP telefon 7971G, ústřednu Asterisk a rozebírají základní XML objekty podporované telefonem.

Poslední kapitola nás seznámí s webovou službou a open-source nástroji, které byly využity k jejímu vytvoření.

## 2.1 Raspberry Pi

V práci bylo použito Raspberry Pi 3 Model B+, jedná se o vylepšenou verzi Raspberry Pi model 3B. Raspberry Pi model 3B+ pohání čipset Broadcom BCM2837. O výpočetní výkon se starají čtyři jádra Cortex-A53 běžící na taktu 1,4 GHz. [2] Tento model je disponován 1 GB RAM jako předchozí model. Oproti předchozímu modelu byla vylepšená síťová karta, jde o Gigabit Ethernet limitovaný na 300 Mbit/s. Dále se dočkalo vylepšení v podobě Bluetooth 4.2 a Wi-Fi 802.11ac. Novinkou je možnost napájení přes Ethernet. [1] Raspberry Pi 3B+ je tradičně vybavené 4x USB verze 2.0, HDMI rozhraním, čtečkou microSD a 40pinovým GPIO.



Obr. 1: Raspberry Pi 3B+

### 2.1.1 Raspbian

Raspbian je operační systém určený pro Raspberry Pi, jedná se o systém postavený na distribuci Debian, přizpůsobený pro Raspberry Pi hardware. Kromě funkce operačního systému nabízí Raspbian přes 35 tisíc balíčků optimalizovaných pro Raspberry Pi a určených k rychlé instalaci na tomto zařízení. Raspbian vznikl jako projekt malého týmu vývojářů a nadšenců Raspberry Pi. [3] Společně s Raspbianem jsou předinstalované další programy pro vzdělávání a programování jako např. Python, Scratch, Sonic Pi, Java, Mathematica a další. [4] Nejjednodušší způsob, jak začít s Raspberry Pi je stáhnout Raspbian obraz, který po zkopírování na microSD kartu a vložení do Raspberry Pi naboootuje zařízení. [5]

## 2.2 Analogová kamera

### 2.2.1 Sony EVI-D31

Obraz je získán z kamery EVI-D31 od firmy Sony. Patří do kategorie tzv. pan, tilt and zoom kamery, tedy dokáže se otáčet, naklánět a zaostřovat. Kamera je vybavena obrazovým snímačem typu 1/3"IT Color CCD, čočkou umožňující rychlé automatické zaostřování a 12x optický zoom. Mezi další funkce patří automatické sledování objektu, detekce pohybu atd. Kamera je ovládána dálkovým ovladačem. Výstupem je obraz o rozlišení 752 x 585 pixelů a je zprostředkován přes rozhraní RCA pin jack nebo S-video. V práci by použit žlutý RCA pin jack, který je typicky určený pro kompozitní video, jedná se o standard PAL. [6]



Obr. 2: Sony EVI-D31

### 2.2.2 Kompozitní video

Kompozitní video je přenosový standard pro SDTV. Pro kompozitní video je charakteristické, že nese dva komponenty - luma signál (informace o jasu) a chroma signál (informace o chrominanci a synchronizaci) v jednom signálu, který je přenášený jedním párem vodičů. Kompozitní barevný video signál se označuje zkratkou CVBS (Color Video, Blank and Sync). Typické konektory určené pro přenos kompozitního signálu jsou BNC konektor, RCA video konektor a SCART. [7] [8]



Obr. 3: RCA pin jack

## PAL

Phase alternating line je jeden ze systémů kódování barevného signálu pro analogové televizní vysílání. PAL se používal ve většině států západní Evropy kromě Francie, kde byl aplikován systém SECAM, který byl používaný i v zemích východního bloku. Systém NTSC, z kterého vycházel PAL byl zavedený v Severní Americe, Střední Americe a Japonsku. [8] Všechny zmíněné systémy musely být zpětně kompatibilní s původním černobílým televizním signálem. Přidaný barvonosný signál nesměl mít vliv na kvalitu černobílého obrazu a musel si vystačit s původní šířkou frekvenčního spektra úplného televizního signálu. Se zavedením barevné televize přibyla nutnost přenést informaci o třech barvách RGB. Tyto tři barevné signály dohromady nesou informaci o výsledné barvě a hodnotu celkového jasu obrazového bodu, tedy sečtením příspěvků R, G a B se získá jas. Tato hodnota odpovídá ve skutečnosti signálu původní černobílé televize. Tento poznatek umožnil přenos jen jasového signálu a chrominančního signálu, který udává

informaci o dvou barvách, jelikož z těchto dvou signálů dokážeme dopočítat informaci o třetí barvě. [9] [10]

PAL využívá kvadrurní amplitudovou modulaci k uložení barevné informace do spektra jasového signálu. Barevný prostor RGB je transformován do prostoru YUV, kde Y reprezentuje jasový signál, který se vysílá tradičně, barevné složky U a V se namodulují na vedlejší nosnou. Automatická korekce zkreslení fáze barvonosné vlny je zajištěna přepínáním fáze složky V o 180° u každé druhé řádky. Synchronizace referenční úrovně amplitudy a fáze zajišťuje referenční signál tzv. color burst, který je přidáván na začátku každé řádky. [10]

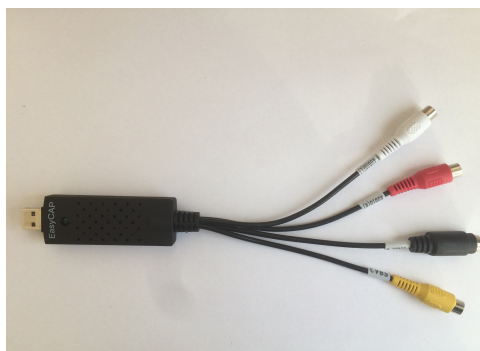
### 2.2.3 EasyCap USB 2.0 Video grabber

EasyCap je čínský výrobce a tímto názvem označuje čtyři hardwarově odlišná zařízení. Jedná se o zařízení, která zachytávají video a audio připojené přes rozhraní S video, CVBS, AUDIO(L) nebo AUDIO (R). Informace na vstupu jsou předány přes rozhraní USB 2.0 do zařízení k zobrazení a zpracování. Toto zařízení dokáže zachytit analogové video s rozlišením až 720 x 576 pixelů u standardu PAL nebo až 720 x 480 pixelů u NTSC.[11]

Nejčastěji se na trhu vyskytují EasyCap produkty s těmito chipsety [12]:

- STK1160 EasyCAP: (Syntek) STK1160 nebo STK1150
- Empia EasyCAP: USB 2861 Device nebo EMP
- Somagic EasyCAP: SM-USB 007 nebo SMI Grabber Device
- UTV007 based EasyCAP: USBTV007

V praktické části byl použit EasyCap s chipsetem USBTV007 (celé označení Fushicai USBTV007 Video Grabber [EasyCAP]). Pro tento chipset byl vytvořen kernel modul s názvem `usbtv` podporující jak video, tak audio a je součástí jádra (Raspbianu) od verze 3.19-rc1. [13]



Obr. 4: EasyCap USB 2.0 Video grabber

## 2.3 Video4linux2 API

Video4linux (někdy označen zkratkou V4L) API je známé jako rozhraní pro programování aplikací pro kamery v linuxu. Jedná se o rozhraní jádra určené pro zachytávání analogového videa. Také umožňuje přístup k audio proudům z AM a FM tunerů. Mezi V4L podporovaná zařízení patří webkamery, analogové TV tunery a zařízení k zachytávání videa. V4L2 (Video4linux verze 2) API bylo poprvé součástí hlavního jádra od verze 2.5.46 vydané v listopadu 2002. [14] [15] Vznikla řada aplikací podporujících jakékoli kamery s V4L ovladačem. Většina ovladačů

byla nejprve vytvořena jako ovladače zařízení v uživatelském prostoru (user space V4L driver). Ovladač pro Raspberry Pi kamera vznikl stejným způsobem. Později Raspberry Pi Foundation vyvíjela s Broadcom oficiální kernel V4L ovladač, který byl dostupný od prosince 2013. [16]

## 2.4 Videobuf2

Jedná se o framework pro ovladače, který je kompatibilní s Video4linux2 API a určený pro multimediální zařízení. Představuje vrstvu mezi ovladačem zařízení fungující v prostoru jádra Linuxu a aplikací fungující v uživatelském prostoru. Tento framework usnadňuje a dbá na správné využití V4L2 API při tvorbě ovladačů. Mimo jiné nabízí nízkoúrovňový, modulární správu paměti pro ovladače. Tato možnost přináší výhody při vytváření ovladačů pro zařízení s neobvyklými požadavky na správu paměti bez nutnosti zasahovat do správy vysokoúrovňové vyrovnávací paměti a API. [17]

Videobuf2 framework poskytuje:

- implementaci tzv. streaming I/O V4L2 ioctlů (Streaming je I/O metoda, kde dochází jen k přenosu adres ve vyrovnávací paměti prostřednictvím ukazatelů mezi aplikací a ovladačem, nikoliv ke kopírování dat.) [18]
- video vyrovnávací paměť, video frontu a funkce správy stavu
- alokaci a správu video vyrovnávací paměti.

## 2.5 FFmpeg

Jedná se o multimediální framework, který podporuje mnoho formátů od nejstarších po nejnovějších. FFmpeg je dostupný na Linuxu, Mac OS, Microsoft Windows, Solaris a dalších. Mezi jeho hlavní schopnosti patří enkódování, dekodování, multiplexování, demultiplexování, stream a filtrování. Důležitou součástí FFmpeg jsou knihovny libavcodec, libavutil, libavformat, libavfilter, libavdevice, libswscale a libswresample. Nabízejí vývojářům mnoho možností využití v aplikacích. Koncoví uživatelé ocení FFmpeg nástroje pro transkódování a přehrávání jako ffmpeg, ffplay a ffprobe. [19]

### FFmpeg knihovny [19]:

- libavutil – jedná se o knihovnu utilit, která obsahuje datové struktury, generátory náhodných čísel a matematické funkce, které zjednodušují programování. Tato knihovna je modulární, malá, nenáročná na paměť a procesor.
- libavcodec – tato knihovna nabízí framework pro kódování/dekodování, obsahuje enkodéry a dekodéry pro audio a video
- libavformat – multiplexuje a demultiplexuje audio a video
- libavdevice – podporuje mnoho vstupních a výstupních zařízení jako např. Video4Linux2, Vfw, DShow, ALSA atd. Provádí zachytávání a vykreslování do mnoha multimediálních vstupních a výstupních zařízení
- Libavfilter – framework poskytující audio/video filtry
- Libswscale – knihovna umožňující změnu rozlišení videa/snímků, převod formátů a barevných prostorů

- Libswresample – vykonává procesy jako audio převzorkování, převod formátu vzorků a rematrixing např. převod ze stera na mono u zvuku

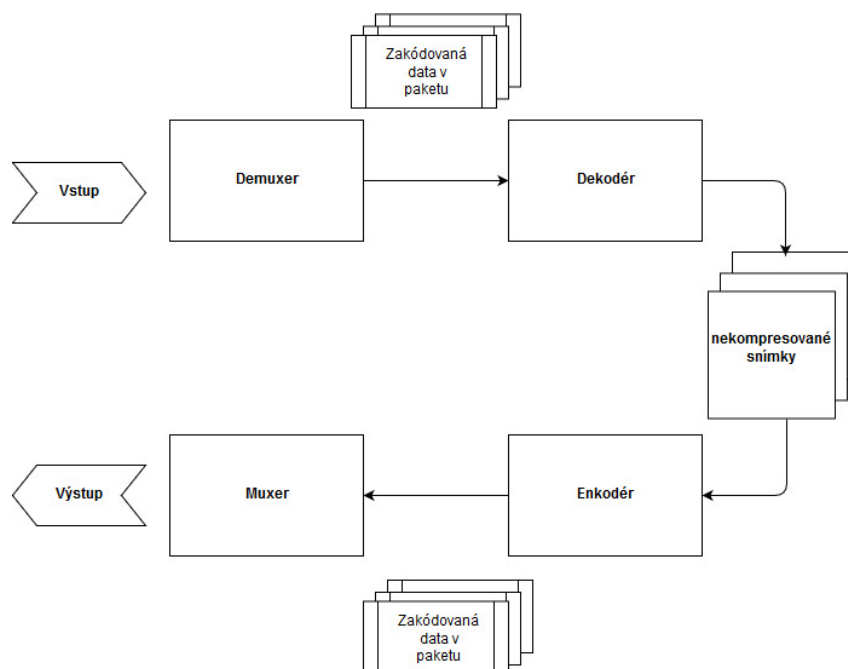
## Nástroje FFmpeg

- ffmpeg – nástroj pro převod formátu multimediálních souborů, běží z příkazového řádku
- ffplay – přehrávač
- ffprobe – multimediální stream analyzer

### 2.5.1 ffmpeg

Ffmpeg je schopný velmi rychle provádět video a audio konverzi. Mezi jeho další přednosti patří schopnost převádět mezi libovolnými vzorkovacími kmitočty, měnit rozlišení obrazu a zachytávat z audio/video live zdrojů. Ffmpeg čte ze vstupu, provede žádoucí změny a zapíše na výstup určený výstupním URL. Vstup pro ffmpeg je definovaný parametrem `-i` a může být různého typu např. normální soubor, roura, stream nebo zařízení pro zachytávání. Ffmpeg umožňuje čtení z několika vstupů a po následném zpracování předat na výstupy, přičemž platí pravidlo, že každé vstupní/výstupní nastavení se aplikuje vždy jen na nejbližší vstup/výstup. [20]

### Schéma transkódování v ffmpeg



Obr. 5: Schéma transkódování

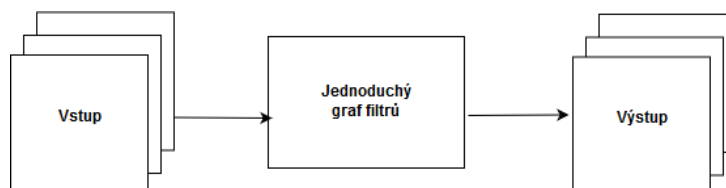
Ffmpeg využívá demuxery z knihovny libavformat ke čtení vstupu a k získání pakety nesoucí zakódovaná data ze vstupu. Pakety poté putují do dekodéru, odkud vycházejí nekompresované snímky, které mohou být pozměněny filtry. Po procesu filtrování jsou pozměněné snímky znovu zakódovány v enkodéru, tím vznikají pakety. Pakety jsou dále zpracovány v muxeru, který je zapisuje do výstupního souboru. [21]



## Filtrování

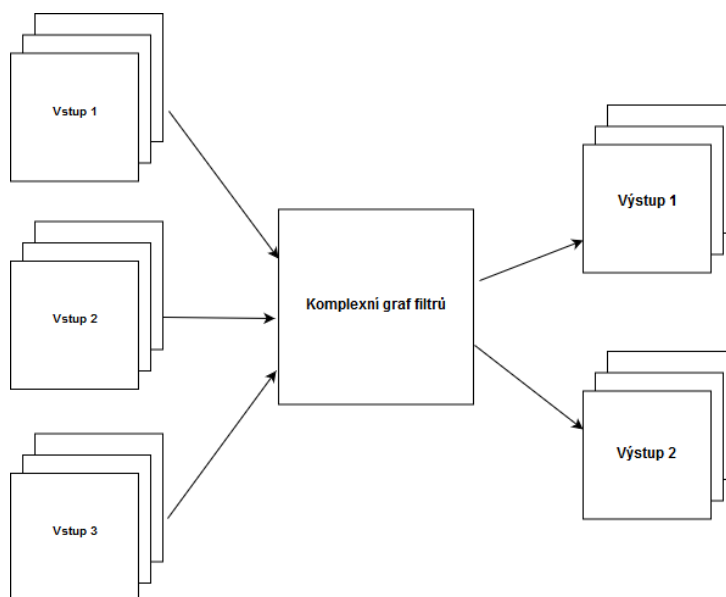
Ffmpeg provádí filtrování pomocí knihovny libavfilter. Jedná se o proces, při kterém často dochází ke změně obsahu snímků např. při použití filtru `scale`, který mění rozlišení snímků. Filtry jako `fps` nebo `setpts`, které mění snímkovou frekvenci či časovou známku ovlivňují vlastnosti, nikoliv obsah snímků. Mezi encodérem a dekodérem mohou být na snímky aplikovány několik filtrů za sebou, takový scénář se nazývá graf filtrů. Rozlišují se dva typy grafy filtrů jednoduchý a komplexní. [22]

Pro jednoduchý graf filtrů platí, že má právě jeden vstup a výstup stejného typu. Tyto filtry se nadefinují pomocí `-filter:v` pro video a `-filter:a` pro audio. [23]



Obr. 6: Schéma simple filtergraph

Komplexní graf filtrů je charakteristický tím, že je k dispozici několik vstupů a výstupů nebo jsou vstupní a výstupní proudy rozdílného typu. [24]



Obr. 7: Schéma complex filtergraph

## 2.6 Cisco IP telefon 7971G

Cisco IP telefony řady 7900 patří ke starším telefonům, které již nejsou podporované výrobcem.[25] Řada 7900 má prohlížeč, který používá k připojování na webové služby. Modely 7921G, 7925G, 7926G, 7925G-EX, 7945G, 7965G, 7970G, 7971G a 7975G jsou vybaveny barevným displejem. Podporují XML objekt pro zobrazení snímku formátu PNG s maximálním rozlišením a barevnou hloubkou v závislosti na modelu telefonu viz Obr 8. [26] Bohužel žádný z nich nedisponuje podporou videa. Jediná zmínka ohledně těchto telefonů a podpoře videa byla v souvislosti se službou Cisco Unified Video Advantage, která společně s Cisco VT Camera III rozšiřuje funkce

telefonů o možnost video hovoru. S touto službou je telefon schopný uskutečnit nebo přijímat video hovor, nicméně přijímaný obraz je zobrazen na počítači, ke kterému je připojen telefon. [27]

Model	Rozlišení	Barevná hloubka
Cisco Unified Wireless IP Phones 7921G, 7925G, 7926G, 7925G-EX	176 x 140 px	16 bitů
Cisco Unified IP Phones 7945G, 7965G	298 x 156 px	16 bitů
Cisco Unified IP Phone 7970G, 7971G	298 x 168 px	12 bitů
Cisco Unified IP Phone 7975G	298 x 168 px	16 bitů

Obr. 8: Maximální parametry PNG snímku pro jednotlivé Cisco IP telefony řady 7900

K demonstraci webové služby byl vybrán Cisco IP telefon model 7971G. Nalevo od displeje se nacházejí programovatelná tlačítka, která mohou být nastavena jako tlačítka pro volbu linky (line button), pro rychlé vytáčení nebo pro vyžádání webové služby. Mimo tato tlačítka je telefon opatřen tradičním číselníkem, navigačními tlačítky, tlačítkem pro hlasovou schránku *Messages*, tlačítkem pro vstup do adresářů *Directories*, tlačítkem *Settings* a tlačítkem *Services*. Telefon je schopný zobrazit XML soubory se speciálními tagy. Stisknutím tlačítka *Services* se zobrazí hlavní nabídka služeb. Zadní strana telefonu je osazena porty pro napájení, pro sluchátko či náhlavní sluchátka a pro Cisco rozšiřující modul. Dále je vybaven integrovaným přepínačem skládající se z 10/100/1000 Mbit/s portu pro přístup do sítě a z přístupového portu pro PC. [28]



Obr. 9: Cisco IP telefon 7971G

### 2.6.1 XML objekty

Pro Cisco IP telefony jsou definovány XML objekty se speciálními tagy. Mezi základní objekty patří *CiscoIPPhoneMenu*, *CiscoIPPhoneInput*, *CiscoIPPhoneDirectory*, *CiscoIPPhoneImageFile* atd. *CiscoIPPhoneImageFile* je v práci využita k zobrazení jednotlivých snímků z kamery. [26]

#### **CiscoIPPhoneMenu**

Tento XML objekt byl použit na hlavní menu služby. Párový tag `<Title>` zobrazuje název menu, párový tag `<Prompt>` zobrazuje nápovědu. Jednotlivé položky menu vymezuje párový tag `<MenuItem>`, uvnitř kterého najdeme párový tag `<Name>` pro název dané služby. Párový tag `<URL>` je určen pro URL adresu, na kterou provede telefon HTTP požadavek při volbě dané služby. Uživatel si vybírá služby/položky navigačními tlačítky a danou položku zvolí stiskem softwarového tlačítka *Select*. [26]

```

<CiscoIPPhoneMenu>
<Title>Název</Title>
<Prompt>Nápověda</Prompt>
<MenuItem>
<Name>Název položky/služby</Name>
<URL>URL dané služby</URL>
</MenuItem>
</CiscoIPPhoneMenu>

```

### CiscoIPPhoneInput

Telefon zobrazí CiscoIPPhoneInput jako formulář, kam zadá uživatel informace, které se pošlou na URL adresu. Jednotlivé pole jsou vymezeny párovým tagem <InputItem>. [26]

```

<CiscoIPPhoneInput>
<Title>Název</Title>
<Prompt>Nápověda</Prompt>
<URL>URL adresa, kam se posílají vstupy</URL>
<InputItem>
<DisplayName>Název pole, kam se zadá vstup</DisplayName>
<QueryStringParam>Proměnná nesoucí vstup</QueryStringParam>
<DefaultValue>Výchozí hodnota pole</DefaultValue>
<InputFlags>Určuje povolené typy vstupu</InputFlags>
</InputItem>
</CiscoIPPhoneInput>

```

Typy párového tagu <InputFlags> jsou:

- A pro prostý text (velká a malá písmena) včetně čísel a speciálních znaků
- T pro telefonní číslo včetně znaků #, \*
- N jen pro čísla
- U jen pro velká písmena
- L jen pro malá písmena
- P pro heslo, znaky jsou převedeny na \*\*\*

### CiscoIPPhoneDirectory

Kontakty jsou zobrazeny pomocí objektu CiscoIPPhoneDirectory, kde každý kontakt je určen párovým tagem <DirectoryEntry>. [26]

```

<CiscoIPPhoneDirectory>
<Title>Název</Title>
<Prompt>Nápověda</Prompt>
<DirectoryEntry>
<Name>Jméno kontaktu</Name>
<Telephone>Telefonní číslo</Telephone>
</DirectoryEntry>

```

```
</CiscoIPPhoneDirectory>
```

## CiscoIPPhoneImageFile

CiscoIPPhoneImageFile XML objekt umožňuje zobrazení snímků ve formátu PNG o rozlišení 298 x 168 pixelů nebo 320 x 168 pixelů v režimu tzv. Wide Mode. Telefon vykresluje snímky s barevnou hloubkou 12 bitů. Tento objekt začíná tagem `<CiscoIPPhoneImage>` nebo tagem `<CiscoIPPhoneImage WindowMode='Wide'>` pro Wide Mode. Párové tagy `<LocationX>` `</LocationX>` a `<LocationY>` `</LocationY>` definují polohu levého horního rohu snímku, tedy nastavením na hodnoty (0,0) se umístí snímek na levý horní roh displeje. Párový tag `<URL>` `</URL>` definuje cestu k PNG souboru. [26]

```
<CiscoIPPhoneImageFile>
<Title>Název</Title>
<Prompt>Nápověda</Prompt>
<LocationX>Horizontální pozice snímku</LocationX>
<LocationY>Vertikální pozice snímku</LocationY>
<URL>Cesta k PNG souboru</URL>
</CiscoIPPhoneImageFile>
```

### 2.6.2 TFTP

TFTP je označení pro Trivial File Transfer Protokol. Jedná se o technologii pro přenos souborů po síti mezi zařízeními a je zjednodušenou verzí protokolu FTP (File Transfer Protocol). TFTP byl vyvíjen v sedmdesátých letech pro počítače s malou kapacitou paměti, u kterých chybí plná podpora FTP. Tento protokol našel uplatnění v domácích či komerčních routerech a jiných síťových zařízeních, kde ho administrátoři využívají k aktualizaci firmwaru routeru nebo k distribuci softwaru po síti. [29]

Cisco telefon žádá o konfigurační soubor s názvem `SEP<MAC>.cnf.xml` poté, co dostane IP adresu od DHCP serveru (IP adresa může být nastavena staticky). V tomto projektu byl TFTP využit k doručení konfiguračního souboru.

### 2.6.3 Konfigurační soubor pro Cisco IP telefon

V kapitole o TFTP bylo zmíněno, jakým způsobem si žádá Cisco IP telefon o konfigurační soubor `SEP<MAC>.cnf.xml`. Konfigurační soubor obsahuje informace definující nastavení telefonu. Nejdůležitější párový tag `<processNodeName>` udává IP adresu ústředny, ke které se má telefon zaregistrovat. Informaci o umístění webové služby najde telefon mezi tagy `<servicesURL>` a `</servicesURL>`, kam se zadá URL adresa webového serveru včetně cesty k hlavní nabídce služeb např. `http://10.0.0.1 /menu.xml`. Uživatel se dostane do hlavní nabídky služeb po stisknutí tlačítka *Services*. [30]

Další důležité párové tagy jsou:

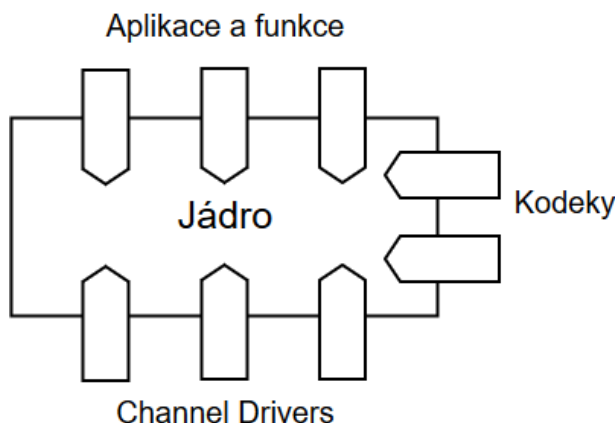
<code>&lt;device&gt;</code> <code>&lt;/device&gt;</code>	tag vymezující začátek a konec
<code>&lt;versionStamp&gt;</code> <code>&lt;/versionStamp&gt;</code>	udává typ telefonu a formát času
<code>&lt;timeZone&gt;</code> <code>&lt;/timeZone&gt;</code>	definice časového pásma
<code>&lt;ethernetPhonePort&gt;</code> <code>&lt;/ethernetPhonePort&gt;</code>	definice čísla portu
<code>&lt;loadInformation&gt;</code> <code>&lt;/loadInformation&gt;</code>	definice firmwaru, který se načte do telefonu
<code>&lt;userLocale&gt;</code> <code>&lt;/userLocale&gt;</code>	místní nastavení telefonu např. jazyk

## 2.7 Asterisk

Asterisk je pobočková telefonní ústředna, anglicky označená PBX - Private Branch Exchange. Tato ústředna je řešena čistě softwarově. Umožňuje telefonům vybaveným různými hardwarovými technologiemi volat si navzájem a připojovat se na služby. Asterisk je někdy brán jako jedno z nejvíce flexibilních a rozšiřitelných řešení v oblasti integrovaného telekomunikačního softwaru. Je modulární a nabízí mnoho možností konfigurace, jelikož je napsán v jazyku C. Díky své nenáročnosti na hardware je častým řešením pro malá embedded zařízení. [31]

### 2.7.1 Architektura ústředny Asterisk

Základem Asterisku je jádro, které interaguje s moduly. Poskytují všechny ostatní funkce, které jádro neposkytuje a mohou být různého typu např. Channel Drivers, aplikace v číslovacím plánu, funkce v číslovacím plánu, kodeky atd. Moduly jsou umístěny v adresáři `/usr/lib/asterisk/modules` jako soubory s příponou `.so`. Většina je nezávisle konfiguratelná a má vlastní konfigurační soubory s příponou `.conf`, které najdeme v adresáři `/etc/asterisk`. Každý protokol tedy potřebuje modul a konfigurační soubor. [32] [33]



Obr. 10: Struktura ústředny Asterisk

### 2.7.2 Číslovací plán

Číslovací plán je důležitou součástí ústředny Asterisk. Veškeré příchozí a odchozí hovory jsou směrovány pomocí číslovacího plánu. Nastavení číslovacího plánu se provádí v konfiguračním souboru `/etc/asterisk/extensions.conf`. Tento soubor je členěn do sekcí, kde se nacházejí statické definice nebo kontexty. [34]

#### Statické definice

Sekce `globals` a `general` obsahují statické definice. V sekci `globals` jsou definovány globální proměnné, s kterými můžeme nakládat v rozsahu celého číslovacího plánu. Sekce `general` nastavuje pravidla pro samotný soubor `extensions.conf`. Např. `writelock` nabízí možnost zakázat přepisování souboru, `clearglobalvars` maže globální proměnné při reloadu atd. [34]

#### Kontext

Kontext je označení pro spustitelný číslovací plán. Lze ho pojmenovat a obsahuje proměnné. Zde se volá na aplikace, které vykonají požadované operace na základě předem definovaných

podmínek. Speciálním typem kontextu jsou makra, která můžeme volat z celého číselového plánu. Makra definujeme pomocí předpony `macro-`, tedy `macro-company` znamená makro s názvem `company`. [34]

### 2.7.3 SCCP protokol

Skinny Client Control Protocol je Cisco proprietární protokol. SCCP používá proprietární zprávy, kterými komunikují IP zařízení s Cisco Unified Communications Manager. Jedná se o ústřednu, která umožňuje navázání a řízení video a hlasové komunikace. Mezi charakteristické rysy tohoto protokolu patří jednoduchost a nenáročnost na výpočetní výkon. Cisco IP telefon se stane SCCP klientem po registraci a zahájení komunikace s Cisco Unified Communications Manager. Během procesu registrace získá telefon informace o linkách a konfiguraci od ústředny. [35]

#### Modul `chan_sccp`

Tento modul patří do skupiny Channel Drivers, prostřednictvím kterých provádí a přijímá Asterisk hovory. Channel drivers fungují jako výchozí brána mezi jádrem Asterisku a vnějšími zařízeními. Jak již název napovídá, `chan_sccp` modul je Skinny Control Client Protocol channel driver pro Asterisk, určený k SCCP připojení mezi naším Cisco IP telefonem a ústřednou Asterisk. [36] [37]

## 2.8 Webová služba

Cisco Unified IP Phone Services umožňují uživatelům spustit na telefonu webové služby vytvořené přímo Ciscem, integrované do ústředny Cisco Unified Communications Manager (Call-Manager). Mezi nejčastější služby patří počasí, akcie, kontakty, denní rozvrh a další. Prostřednictvím Cisco Unified IP Phone Services dává Cisco vývojářům možnost vytvářet a přizpůsobovat služby, které budou uživatelé využívat a ovládat pomocí dostupných tlačítek. Díky tomu mohou být uživatelům doručeny informace přes webové služby. [38]

Cílem této práce bylo napsat pro Cisco IP telefon 7971G službu, která bude zobrazovat sekvence snímků pořízených kamerou využitím otevřených nástrojů. Linux, Apache, MySQL a PHP je skupina open-source softwarů označovaná jako LAMP. Webová služba pro zobrazování snímků z kamery je napsána v jazyce PHP verze 7 s využitím XML šablony a její běh zajišťuje webový server Apache.

### 2.8.1 PHP

PHP neboli Hypertext Preprocessor je skriptovací jazyk pracující na straně serveru. Vzniklo v roce 1966 a je nejčastěji využívané k ukládání a změně dat webových stránek. PHP skript je spuštěn na serveru a výstupem je HTML (případně XML) soubor, který je doručen prohlížeči. Tento skriptovací jazyk je nejčastěji používán k generování dynamického obsahu stránek, modifikaci dat v databázi nebo na serveru. Uživatelé ocení rozsáhlou dokumentaci na oficiálních stránkách `php.net`. PHP je spustitelné na operačních systémech Windows, Linux, Unix, Mac OS X a kompatibilní se servery Apache, IIS atd. [39]

### 2.8.2 Apache

Apache je open-source HTTP server pro moderní operační systémy zahrnující UNIX a Windows. Vzniklo se snahou poskytovat HTTP služby v souladu se současnými HTTP standardy.

Apache je považován za zabezpečený a efektivní webový server, který je možný rozšířit moduly třetích stran. [40]

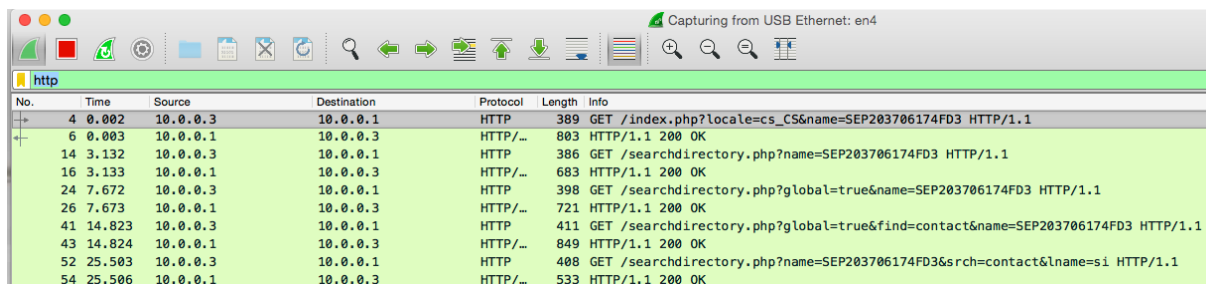
### 2.8.3 HTTP komunikace

Cisco IP telefon komunikuje s webovým serverem protokolem HTTP. Tento protokol je založen na principu požadavek/odpověď. Součástí firmwaru telefonu je jak HTTP klient, tak HTTP server. Telefon používá klienta při posílání HTTP požadavku a HTTP server při přijetí HTTP požadavku. [41]

#### HTTP client request

Cisco IP telefon provede HTTP požadavek metodou HTTP GET, u které se předávají všechna formulářová data jako součástí URL za otazníkem. Konkrétně se jedná o parametr `locale` a `name`, který má hodnotu MAC adresy daného telefonu. [42]

Metoda GET slouží k vyžádání XML objektu. Telefon očekává XML objekty s MIME typem `text/xml` od webového serveru. Dostane-li XML objekt s jiným MIME typem např. `application/xml`, objeví se chybová hláška `XML Parse Error`[4]. [43] Server používá MIME typ k informování telefonu o typu a formátu doručeného souboru, aby došlo k jeho úspěšnému zpracování. Na Obr. 11 je zachycena komunikace mezi telefonem s IP adresou 10.0.0.3 a webovým serverem s IP adresou 10.0.0.1.

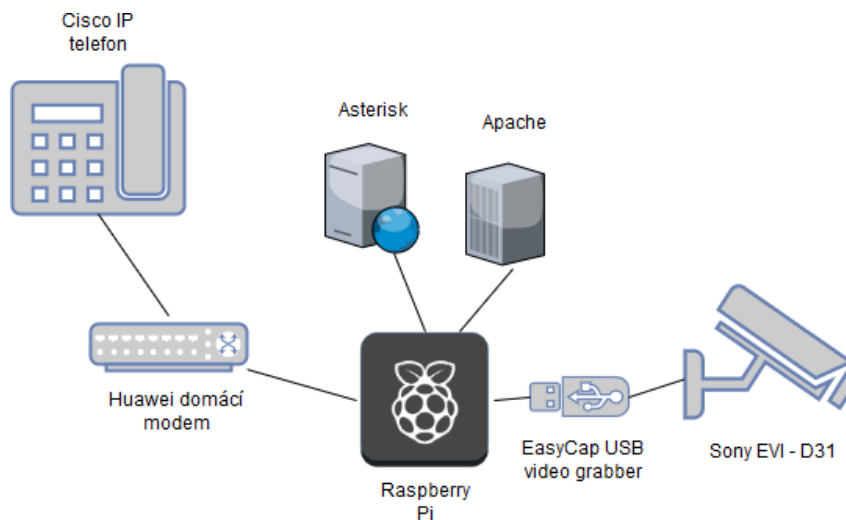


The screenshot shows a Wireshark capture of network traffic on the USB Ethernet interface (en4). The filter is set to 'http'. The table below represents the data shown in the packet list pane.

No.	Time	Source	Destination	Protocol	Length	Info
4	0.002	10.0.0.3	10.0.0.1	HTTP	389	GET /index.php?locale=cs_CS&name=SEP203706174FD3 HTTP/1.1
6	0.003	10.0.0.1	10.0.0.3	HTTP	803	HTTP/1.1 200 OK
14	3.132	10.0.0.3	10.0.0.1	HTTP	386	GET /searchdirectory.php?name=SEP203706174FD3 HTTP/1.1
16	3.133	10.0.0.1	10.0.0.3	HTTP	683	HTTP/1.1 200 OK
24	7.672	10.0.0.3	10.0.0.1	HTTP	398	GET /searchdirectory.php?global=true&name=SEP203706174FD3 HTTP/1.1
26	7.673	10.0.0.1	10.0.0.3	HTTP	721	HTTP/1.1 200 OK
41	14.823	10.0.0.3	10.0.0.1	HTTP	411	GET /searchdirectory.php?global=true&find=contact&name=SEP203706174FD3 HTTP/1.1
43	14.824	10.0.0.1	10.0.0.3	HTTP	849	HTTP/1.1 200 OK
52	25.503	10.0.0.3	10.0.0.1	HTTP	408	GET /searchdirectory.php?name=SEP203706174FD3&srch=contact&lname=si HTTP/1.1
54	25.506	10.0.0.1	10.0.0.3	HTTP	533	HTTP/1.1 200 OK

Obr. 11: HTTP komunikace mezi telefonem a serverem

### 3 Praktická část



Obr. 12: Pracovní sestava

Pro praktickou část bylo postaveno pracoviště skládající se z Cisco IP telefonu 7971G, Raspberry Pi 3B+, Huawei EchoLife HG520i modemu, EasyCap USB 2.0 Video Grabberu a kamery Sony EVI-D31. Domácí modem značky Huawei se zabudovaným switchem sloužil jako přepínač, přes který probíhala komunikace mezi telefonem a Raspberry Pi. Tato zařízení byla propojena UTP kabely k přepínači, který přiděloval zařízením IP adresy pomocí DHCP serveru. Na Raspberry Pi byly spuštěny zpracování obrazu z kamery, webová služba pro telefon a ústředna Asterisk. Výstupní video z kamery bylo přenášeno přes RCA pin jack do zachytávacího zařízení, které bylo připojené přes rozhraní USB 2.0 k Raspberry Pi.

#### 3.1 Zprovoznění Raspberry Pi

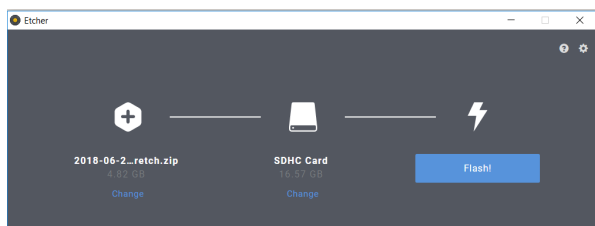
Raspberry Pi hraje důležitou roli v této práci, stará se o veškerý výpočetní výkon spojený s úpravou snímků z kamery, webovou službou a provozováním ústředny Asterisk. MicroSD karta, Micro USB napájecí kabel, monitor, HDMI kabel, USB klávesnice a myš jsou příslušenství potřebná k prvnímu spuštění.

##### 3.1.1 Instalace Raspbianu Stretch

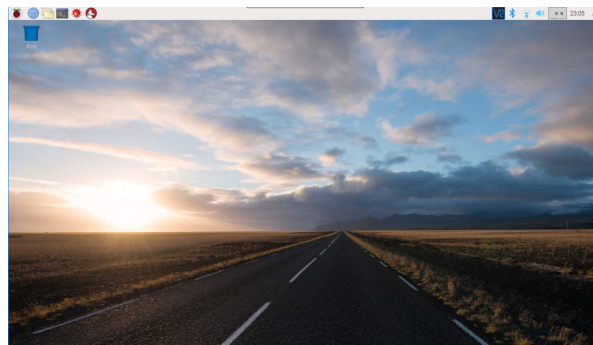
Raspberry Pi Foundation nabízí možnost zakoupit si předinstalovanou NOOBS microSD kartu. Zkratka NOOBS znamená New Out Of Box Software, jedná se o jednoduchý instalační program operačního systému Raspbian. Další možnost je stažení Raspbian obrazu z oficiálních stránek.

Pro Raspberry Pi 3B+ byl zvolen Raspbian Stretch, jehož základem je kernel 4.14. Tato verze byla vydána v listopadu 2018. Ke zprovoznění Raspberry Pi byl potřeba obraz Raspbianu Stretch, který se nainstaloval na MicroSD kartu pomocí zapisovacího nástroje Etcher viz Obr 13. Bylo nutné předtím naformátovat kartu na systém FAT. [44] Po dokončení instalace obrazu byla karta vložena do Raspberry Pi. Pokud byl Raspbian úspěšně nainstalován, dojde k naboťování zařízení do tzv. Raspbian PIXEL desktop prostředí viz Obr 14.





Obr. 13: Zapisovací nástroj Etcher



Obr. 14: Pixel desktop prostředí

### 3.1.2 Síťová konfigurace

Síťová nastavení se provádějí v souboru `/etc/network/interfaces` jako v Debianu. Editací tohoto souboru nastavíme pro Raspberry Pi buď statickou IP adresu nebo nadefinujeme, aby IP adresu získalo od DHCP serveru.

Nastavení rozhraní `eth0` v `/etc/network/interfaces`, aby využívalo službu DHCP serveru:

```
auto eth0
allow-hotplug eth0
iface eth0 inet dhcp
```

Změny v souboru `interfaces` se aplikují po restartu `eth0`:

```
sudo ifdown eth0
sudo ifup eth0
```

## 3.2 Zprovoznění kamery

Výstupní kompozitní video z kamery Sony EVI-D31 bylo zprostředkováno přes žlutý RCA pin jack do zařízení EasyCap USB 2.0 Video grabber. Zachytávací zařízení bylo připojené do Raspberry Pi přes rozhraní USB 2.0, dostupnost zařízení byla ověřena příkazem `lsusb`, výstup je zachycen na Obr. 15 .

```
pi@raspberrypi:~ $ lsusb
Bus 001 Device 007: ID 1b71:3002 Fushicai USBTV007 Video Grabber [EasyCAP]
Bus 001 Device 006: ID 0424:7800 Standard Microsystems Corp.
Bus 001 Device 003: ID 0424:2514 Standard Microsystems Corp. USB 2.0 Hub
Bus 001 Device 002: ID 0424:2514 Standard Microsystems Corp. USB 2.0 Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
pi@raspberrypi:~ $
```

Obr. 15: Výstup `lsusb`

Po připojení zachytávacího zařízení do Raspberry Pi se načetly moduly `usbtv`, `videobuf2_vmalloc`, `videobuf2_memops`, `videobuf2_v4l2`, `videobuf2_core`, `videodev` a `media`. Seznam všech načtených kernel modulů byl uchován v souboru `/proc/modules`. Informace o stavech načtených modulů se zobrazily příkazem `lsmod` viz Obr. 16.

```

pi@raspberrypi:~/Videos $ lsmod
Module                Size  Used by
usbtv                 20480  0
videobuf2_vmalloc    16384  1 usbtv
videobuf2_memops     16384  1 videobuf2_vmalloc
videobuf2_v4l2       24576  1 usbtv
videobuf2_core       45056  2 usbtv,videobuf2_v4l2
videodev             184320  3 usbtv,videobuf2_core,videobuf2_v4l2
media                 32768  1 videodev
joydev               20480  0
evdev                24576  0
rfcomm               49152  4
bnep                 20480  2
fuse                 110592  3
hci_uart             36864  1
btbcm                16384  1 hci_uart
serdev               20480  1 hci_uart
bluetooth            368640  29 hci_uart,bnep,btbcm,rfcomm
ecdh_generic         28672  1 bluetooth
brcmfmac             307200  0
brcmutil             16384  1 brcmfmac
cfg80211             573440  1 brcmfmac
rfkill               28672  6 bluetooth,cfg80211
snd_bcm2835          32768  1
snd_pcm              95304  2 usbtv,snd_bcm2835
snd_timer            32768  1 snd_pcm
snd                  69632  6 usbtv,snd_timer,snd_bcm2835,snd_pcm
uio_pdrv_genirq      16384  0
uio                  20480  1 uio_pdrv_genirq
fixed                16384  0
i2c_dev              16384  0
ip_tables            24576  0
x_tables             32768  1 ip_tables
ipv6                 425984  39
pi@raspberrypi:~/Videos $

```

Obr. 16: Výstup lsmod

### 3.2.1 Ovladač pro Fushicai USBTV007 Audio-Video Grabber

Modul `usbtv` je ovladač pro zachytávací zařízení Fushicai USBTV007 Audio-Video Grabber. Jedná se ointree modul, tedy byl přijatý a zahrnutý do jádra. Tento modul využívá služby modulů `videobuf2-v4l2`, `videodev`, `snd-pcm`, `videobuf2-vmalloc`, `videobuf2-core` a `snd`. Podrobnější informace o `usbtv` jsou zobrazeny na Obr. 17.

```

pi@raspberrypi:~ $ modinfo usbtv
filename:       /lib/modules/4.14.74-v7+/kernel/drivers/media/usb/usbtv/usbtv.ko
license:       Dual BSD/GPL
description:   Fushicai USBTV007 Audio-Video Grabber Driver
author:       Lubomir Rintel, Federico Simoncelli
srcversion:    0473A6663EF68AB19A84E6B
alias:         usb:v1F71p330d*dc*dsc*dp*ic*isc*ip*in*
alias:         usb:v1B71p3002d*dc*dsc*dp*ic*isc*ip*in*
depends:        videobuf2-v4l2,videodev,snd-pcm,videobuf2-vmalloc,videobuf2-core,snd
intree:       Y
name:         usbtv
vermagic:     4.14.74-v7+ SMP mod_unload modversions ARMv7 p2v8
pi@raspberrypi:~ $

```

Obr. 17: Informace o modulu `usbtv`

Modul `media` je určený k registraci tzv. device node pro ovladače medií. Device node jsou soubory, které systém používá pro přístup k hardwarovým zařízením. Modul `videodev` slouží jako registr zařízení pro video4linux ovladače a využívá k tomu `media` modul. Jedná se ointree moduly. Zachytávací zařízení se zaregistrovalo jako `/dev/video0`.

### 3.2.2 Instalace `ffmpeg`

Nástroj `ffmpeg` má schopnost rychlé video a audio konverze. V práci byl použit ke zpracování obrazu ze zachytávacího zařízení.

Knihovny `libav` a nástroj `ffmpeg` se nainstalovaly příkazy:

```
sudo apt-get install libav-tools
```

```
sudo apt-get install ffmpeg
```

## Práce s ffmpeg v příkazovém řádku

Video konverzi či jiné funkce nástroje ffmpeg se spouští přes příkazový řádek. Parametrem `-i` se určuje vstup, pro který platí nastavení vypsané těsně před ním. Výstupní URL je definován jako zbytek znaků v příkazu, který nemůže být interpretován jako nastavení. [20]

Definice vstupu, výstupu a jejich nastavení:

```
ffmpeg [globální nastavení] {[nastavení vstupního souboru] -i URL ke vstupnímu souboru nebo zařízení} {[nastavení pro výstupní soubor] URL výstupu}
```

## Přízpusobení obrazu pro Cisco IP telefon 7971G

Pomocí `CiscoIPPhoneImageFile` XML objektu lze vytvořit službu, která zobrazí PNG snímky s rozlišením 298 x 168 pixelů a barevnou hloubkou 12 bitů. V práci byl nástroj ffmpeg využit ke změně rozlišení snímku získaného ze zachytávacího zařízení. Upravený snímek byl poté uložen do adresáře `/var/www/html/photo` ve formátu PNG pod názvem `security.png`. Telefon si poradí se snímky s vyšší barevnou hloubkou tím, že provede redukci barevné hloubky na 12 bitů.

Příkaz pro pořízení snímku a úpravu rozlišení:

```
ffmpeg -f video4linux2 -i /dev/video0 -frames:v 1 -vf scale=298:168 -y /var/www/html/photo/security.png
```

- `f` – vnucení formátu vstupního/výstupního souboru. V případě nedefinování dojde k chybě “Unable to find a suitable output format for video4linux2”
- `i` – definování vstupu, konkrétně `/dev/video0`
- `frames:v` – nastavení počtu snímků na výstupu
- `vf` – parametr vytvářející filtr, konkrétně filtr `scale`
- `y` – možnost přepsat výstupní soubor bez potvrzení

Jako vstup bylo zvoleno `video4linux2` vstupní zařízení, které se zaregistrovalo pod device node `/dev/video0`. Tento příkaz udělá jeden snímek, na který se aplikuje filter `scale`, který změní rozlišení daného snímku na 298 x 168 pixelů a výstup uloží na webový server. Výpis z ffmpeg je zachycen na Obr. 18

```
Input #0, video4linux2,v4l2, from '/dev/video0':
  Duration: N/A, start: 27917.905059, bitrate: 165722 kb/s
  Stream #0:0: Video: rawvideo (YUY2 / 0x32595559), yuyv422, 720x480, 165722 kb/s, 29.97 fps, 29.97 tbr, 1000k tbn, 1000k tbc
Output #0, image2, to '/var/www/html/photo/security.png':
  Metadata:
    encoder      : Lavf57.56.101
  Stream #0:0: Video: png, rgb24, 298x168, q=2-31, 200 kb/s, 29.97 fps, 29.97 tbn, 29.97 tbc
  Metadata:
    encoder      : Lavc57.64.101 png
```

Obr. 18: Výpis z nástroje ffmpeg

### 3.3 Instalace a konfigurace ústředny Asterisk

Pro tuto práci byl vybrán Asterisk 13. Nezbytnou nutností je program *make*. Asterisk se nainstaloval kompilací ze zdrojových kódů, které jsou dostupné na oficiálních stránkách v sekci downloads.

Před samotnou kompilací bylo potřeba nainstalovat potřebné balíčky:

```
apt-get install build-essential libsqlite3-dev libxml2-dev libncurses5-dev
libncursesw5-dev libmysqlclient-dev libiksemel-dev libssl-dev libnewt-dev
libusb-dev libeditline-dev libedit-dev curl libcurl4-gnutls-dev libjansson-dev
```

Zdrojové kódy Asterisku se získají:

```
wget http://downloads.asterisk.org/pub/telephony/asterisk/asterisk-13-current.tar.gz
```

Archiv se rozbalí a zároveň proběhne dekomprese souborů pomocí příkazu:

```
tar -xvzf asterisk-13-current.tar.gz
```

Instalace ústředny se provede příkazy:

```
./configure
make menuconfig
make
make install
make samples
make config
```

Příkaz `./configure` zjistí informace o systému, hardwaru a knihovnách. Ústředna se nainstalovala příkazy `make`, `make install`, `make samples` a `make config`. Příkaz `make samples` vytvořil základní configurační soubory a příkaz `make config` vytvořil init skripty. [45]

První spuštění ústředny Asterisk v konzolovém módu s možností detailního výstupu pro případnou diagnostiku chyb:

```
sudo asterisk -vvvgc
```

- `c` : Spuštění Asterisku v konzolovém módu, tedy s možností ho ovládat pomocí příkazového řádku.
- `g` : Možnost výpisu paměti při havarování programu.
- `v` : Udává stupeň podrobnosti výpisu.

Pokud Asterisk běží v pozadí, lze se na něj připojit pomocí příkazu:

```
sudo asterisk -r
```

Uživatelské rozhraní prostřednictvím příkazového řádku neboli Asterisk Command Line Interface (CLI) vypadá viz Obr. 19.

```
pi@raspberrypi:~$ sudo asterisk -rvvvvv
Asterisk 13.23.1, Copyright (C) 1999 - 2014, Digium, Inc. and others.
Created by Mark Spencer <markster@digium.com>
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for details.
This is free software, with components licensed under the GNU General Public
License version 2 and other licenses; you are welcome to redistribute it under
certain conditions. Type 'core show license' for details.
=====
Connected to Asterisk 13.23.1 currently running on raspberrypi (pid = 25462)
raspberrypi*CLI> █
```

Obr. 19: Asterisk command line interface (CLI)

Základní příkazy potřebné pro práci v Asterisk CLI [46]:

<code>core stop now</code>	Vypnout Asterisk okamžitě.
<code>core stop gracefully</code>	Nepřijímat nové hovory a vypnout Asterisk po odbavení všech právě probíhajících hovorů.
<code>core stop when convenient</code>	Vypnout Asterisk po odbavení všech právě probíhajících hovorů.
<code>core restart now/gracefully/when..</code>	Restart Asterisku podle volby.
<code>module show</code>	Zobrazit všechny dostupné moduly.
<code>module load/unload/reload chan_sccp</code>	Načíst/znovu načíst/uvolnit modul <code>chan_sccp</code> .
<code>sccp show devices</code>	Zobrazit všechny SCCP klienty.
<code>dialplan reload</code>	Aktualizace číslovacího plánu.

### 3.3.1 Instalace `chan_sccp`

Pro správu spojení mezi ústřednou Asterisk a telefonem byl použit modul `chan_sccp`. Tento modul je nutný stáhnout a zkompileovat dodatečně. Poté je potřeba zakázat načítání výchozího modulu `chan_skinny`.

Pro stažení zdrojových kódů modulu `chan_sccp` je potřeba program `git`. Stažení souborů se provede příkazem:

```
git clone https://github.com/chan-sccp/chan-sccp chan-sccp_master
```

Modul `chan_sccp` se nainstaluje a načte pomocí:

```
./configure
make
make install
make reload
```

Zakáz načítání výchozího modulu `chan_skinny` a povolení načtení `chan_sccp` se provede přidáním nastavení do souboru `/etc/asterisk/modules.conf` [47]:

```
noload => chan\_skinny.so
load => chan\_sccp\_b
```

Stav modulu `chan_sccp` se zjistí v Asterisk CLI příkazem `module show like chan_sccp.so`. Na Obr. 20 můžeme vidět, že modul je ve stavu `running`.

```

pi@raspberrypi:~$ sudo asterisk -rvvvvv
Asterisk 13.23.1, Copyright (C) 1999 - 2014, Digium, Inc. and others.
Created by Mark Spencer <markster@digium.com>
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for details.
This is free software, with components licensed under the GNU General Public
License version 2 and other licenses; you are welcome to redistribute it under
certain conditions. Type 'core show license' for details.
=====
Connected to Asterisk 13.23.1 currently running on raspberrypi (pid = 1082)
raspberrypi*CLI> module show like chan_sccp.so
Module                Description                Use Count  Status  Support Level
chan_sccp.so          Skinny Client Control Protocol (SCCP). R 0      Running  unknown
1 modules loaded
raspberrypi*CLI>

```

Obr. 20: Stav modulu chan\_sccp

### 3.3.2 Konfigurační soubory ústředny asterisk

Konfigurační soubory ústředny Asterisk se nacházejí v adresáři `/etc/asterisk`. V tomto adresáři najdeme např. soubory `sccp.conf`, `skinny.conf` a `sip.conf`, které obsahují nastavení pro SCCP/Skinny/SIP připojení a seznam SCCP/Skinny/SIP klientů včetně jejich nastaveních a linek. Soubor `extentions.conf` neboli číslovací plán Asterisku je statický konfigurační soubor používaný modulem `pbx_config`. V tomto souboru se nastavují všechny příchozí a odchozí hovory. Soubor `modules.conf` je určený k nastavení načítání modulů.

### 3.3.3 Registrace SCCP klienta do ústředny

SCCP klienti musejí být registrováni v souboru `/etc/asterisk/sccp.conf`, aby mohli volat nebo být voláni. V sekci `[general]` se nachází všeobecná nastavení, která platí pro všechna SCCP připojení, pokud není v definici účtu uvedeno jinak. Telefon s MAC adresou `SEP001A6D876CFD` byl zaregistrován přidáním řádku `[SEP001A6D876CFD]`, pod kterým se definovala jednotlivá nastavení. Pod nastaveními se přiřadily telefonní linky např. `[98011]` a následovala nastavení dané linky. Celý obsah souboru `sccp.conf` je k dispozici zde [48].

Základní nastavení pro SCCP klient [50]:

```

[SEP001A6D876CFD]
description = Phone Number One ; popis zařízení
devicetype = 7971                ; typ zařízení
park = off                       ; přidržet a pokračovat na jiném zařízení
button = line, 98011,default    ; přidělit a používat linku 98011 jako výchozí
cfwdall = off                   ; přesměrování všech hovorů
type = device
keepalive = 60                  ; keep alive zprávy každých 60s
transfer = on                   ; možnost přepojení hovorů
cfwdall = off                   ; přesměrování všech hovorů
cfwdbusy = off                  ; přesměrování v případě, že je linka busy
cfwdnoanswer = off             ; přesměrování pokud nikdo nezvedne
deny=0.0.0.0/0.0.0.0           ; odmítnout všechny IP adresy vyjma IP v permit
permit=10.0.0.0/255.255.255.0 ; povolený rozsah IP adres
mwilamp = on                    ; možnost signalizačních LED

```

Po provedení změn bylo potřeba znovu načíst chan\_sccp modul. V Asterisk CLI se provedl příkaz:

```
module reload chan_sccp
```

V Asterisk CLI byl viděn průběh registrace telefonu viz Obr. 21, pokud byl nastaven vyšší stupeň podrobnosti výpisu.

```
Connected to Asterisk 13.23.1 currently running on raspberrypi (pid = 1082)
-- -- SCCP: Accepted Client Connection from 10.0.0.2:51591
-- -- SCCP: Connected on server via 10.0.0.1:55211
== == Alarm Type: LastOutOfServiceInformation
== == Reason Enum: 10
== == Last Event Sent: 0:KeepAlive
== == Last Event Received: 256:KeepAliveAck
-- -- SEP001A6D876CFD: Device Not NATTED. Device IP '10.0.0.2' falls in localnet scope
> > SEP001A6D876CFD: Asked for our protocol capability (22).
> > SEP001A6D876CFD: Phone protocol capability : 22
> > SEP001A6D876CFD: Our protocol capability : 22
-- -- SEP001A6D876CFD: Joint protocol capability : 22
-- -- SEP001A6D876CFD: Ask the phone to send keepalive message every 56 seconds
-- -- SEP001A6D876CFD: using softkeyset: default (0xb29080)!
-- -- SEP001A6D876CFD: Sending ConfigStatMessage, lines 2, speeddials 2
-- -- SEP001A6D876CFD: Device has 11 Capabilities
-- -- SEP001A6D876CFD: Got Forward Status Request. Line: 1
-- -- SEP001A6D876CFD: Got Forward Status Request. Line: 2
-- -- SEP001A6D876CFD: Device registered; performing post registration tasks...
-- -- SEP001A6D876CFD: (sccp_devstate_deviceRegisterListener) device registered
```

Obr. 21: Průběh registrace telefonu jako SCCP klient

### 3.4 Konfigurace Cisco IP telefonu

Pro přenos konfiguračního souboru SEP<MAC>.cnf.xml do Cisco IP telefonu byl nainstalován TFTP server na Raspberry Pi. Konfigurační soubor SEP001A6D876CFD.cnf.xml určený pro telefon s danou MAC adresou byl umístěn v adresáři /home/pi/tftpboot. Celý obsah souboru je dispozici zde [49].

#### Síťové nastavení v telefonu

Stisknutím tlačítka *Settings* se zobrazil panel nastavení. Síťové nastavení telefonu se provedlo v sekci *Network Configuration* -> *IPv4 Configuration*. Možnost editování jednotlivých nastavení se odemkla stisknutím sekvence znaku **\*\*#**.

Nastavení telefonu pro získání síťového nastavení od DHCP serveru:

```
Settings -> Network Configuration -> IPv4 Configuration -> DHCP -> Yes ->Save
```

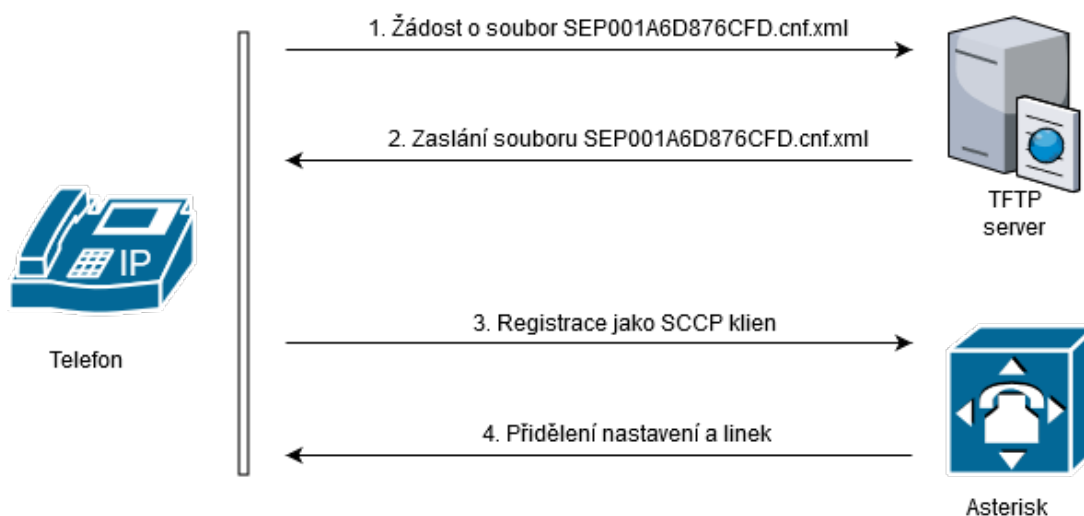
Nastavení IP adresy TFTP serveru v telefonu:

```
Settings -> Network Configuration -> IPv4 Configuration -> Alternate TFTP ->Yes
```

```
Settings -> Network Configuration -> IPv4 Configuration -> TFTP server 1 -> Edit
->Validate
```

Po změně nastavení bylo potřeba restartovat telefon. Po spuštění dostal telefon IP adresu, masku sítě a výchozí bránu od DHCP. Následně se spojil s TFTP serverem za účelem získání

konfiguračního souboru SEP001A6D876CFD.cnf.xml. Po úspěšném načtení konfiguračního souboru se zaregistroval k ústředně jako SCCP klient, od které získal nastavení telefonu a linek. Komunikace mezi telefonem, TFTP a ústřednou Asterisk je znázorněna na Obr. 22.



Obr. 22: Průběh nastavení a registrace telefonu

V Asterisk CLI lze vypsat všechny SCCP klienty příkazem `sccp show devices` viz Obr. 23

```

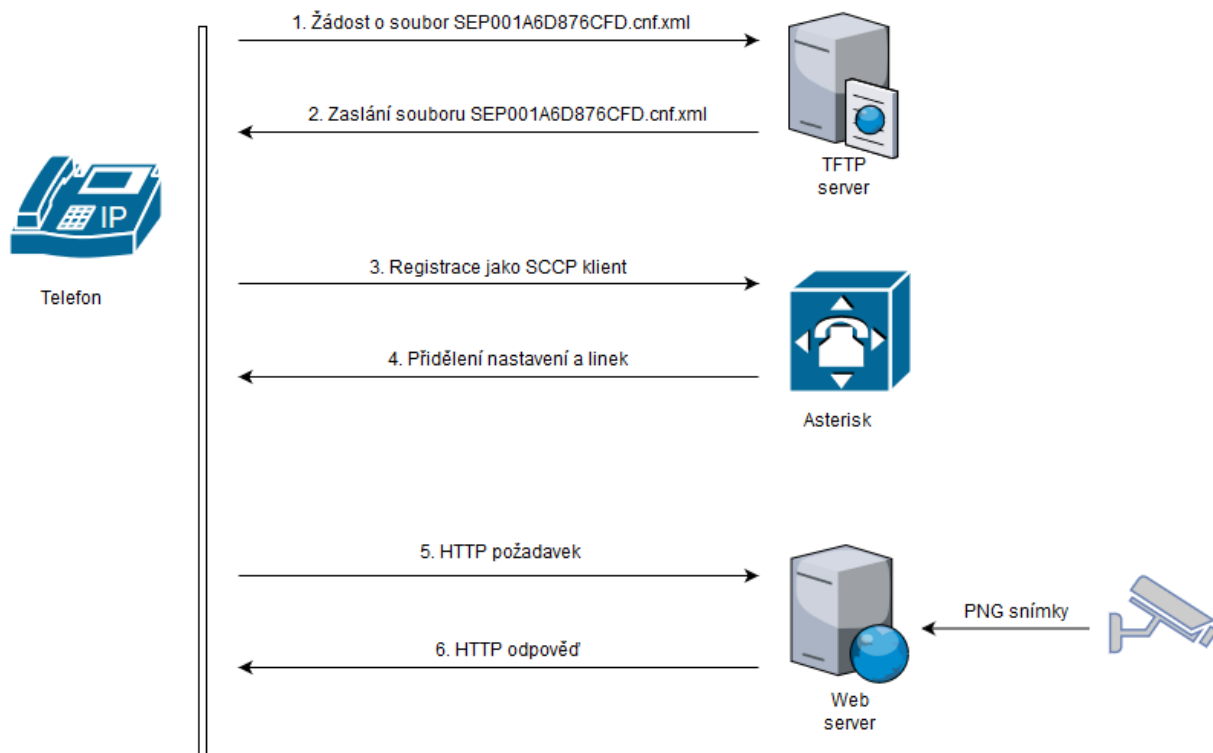
raspberrypi*CLI> sccp show devices
+--- Devices -----
|-----+
| Descr          | Address          | Mac          | RegState | Token | RegTime          | Act | Lines | N
|-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Phone Number One | 10.0.0.2:51591 | SEP001A6D876CFD | OK       | No To | Tue Dec 4 13:43:19 2018 | No 2 | (
| Auto)0ff |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
raspberrypi*CLI> █
  
```

Obr. 23: Seznam registrovaných SCCP klientů

### 3.5 Webová služba

Cílem bylo napsat službu pro telefon, která bude zobrazovat snímky z kamery na displeji telefonu. Za tímto účelem se použily PHP 7 a Apache. Registrace a následná komunikace mezi telefonem a webovým serverem je znázorněno na Obr. 24.





Obr. 24: Průběh komunikace mezi telefonem a servery

### 3.5.1 Instalace a nastavení Apache2

Apache2 se nainstaloval příkazem:

```
apt-get install apache2
```

Apache2 musí doručovat XML stránky s MIME typem `text/xml`. [41] Cisco telefon je schopný zpracovat XML objekty typu `text/xml`. Bylo nutné přidat nastavení v souboru `/etc/apache2/mods-enabled/mime.conf`, aby webový server posílal XML objekty s daným MIME typem. [51]

Nastavení v `/etc/apache2/mods-enabled/mime.conf`:

```
AddType text/html .shtml
AddType text/xml .xml
AddOutputFilter INCLUDES .shtml
```

Nastavení v souboru `/etc/apache2/ports.conf` přidáním:

```
<IfModule mime_module>
AddType text/xml .xml
</IfModule>
```

Restart Apache příkazem:

```
/etc/init.d/apache2 restart
```

### 3.5.2 Instalace PHP 7

Instalace PHP 7 a modulu zajišťující komunikaci s webovým serverem Apache2 se provedla příkazem:

```
apt-get install libapache2-mod-php7 php7
```

Po dokončení instalace `libapache2-mod-php7` a `php7` byl nutný restart Apache.

### 3.5.3 Služba Camera

Služba pro zobrazování snímků z kamery byla nazvána Camera. Uživatel si může službu Camera spustit z hlavní nabídky služeb *Services* nebo stisknutím specializovaného tlačítka, které bylo nastaveno pro vyžádání této služby.

#### Hlavní nabídka služeb `menu.xml`

Hlavní nabídka `menu.xml` byla vytvořena pomocí XML objektu `CiscoIPPhoneMenu`. Soubor `menu.xml` byl umístěn v adresáři webového serveru `/var/www/html`. Jestliže uživatel zvolí službu Camera, vykoná telefon HTTP GET na URL `http://10.0.0.1/photo.php`.

Struktura `menu.xml`:

```
<CiscoIPPhoneMenu>
  <Title>Menu</Title>
  <Prompt>Make your selection</Prompt>
  <MenuItem>
    <Name>Camera</Name>
    <URL>http://10.0.0.1/photo.php</URL>
  </MenuItem>
</CiscoIPPhoneMenu>
```

#### Soubor `photo.php`

Soubor `headers.php` byl vložen do `photo.php` konstrukcí `require_once`. Soubor `headers.php` vygeneruje HTTP hlavičku, kterou dokáže telefon zpracovat. Funkce `shell_exec`, jak již název napovídá, vykoná příkaz přes shell. Soubor `photo.php` používá tuto funkci k uskutečnění příkazu pro vytvoření a přizpůsobení snímku pomocí nástroje `ffmpeg`. Výsledný snímek formátu PNG je uložen v adresáři webového serveru. Příkaz `echo` odešle na výstup `CiscoIPPhoneImageFile` XML objekt ve formě řetězce. Tento objekt zná cestu k PNG snímku a je doručen telefonu, který ho zpracuje na základě podporované HTTP hlavičky. Daný snímek je poté zobrazen na displeji telefonu viz Obr 28. Celý proces se opakuje každou sekundu a za tuto dobu se aktualizuje snímek.

Obsah souboru `photo.php`:

```
<?php
require_once "lib/headers.php";

shell_exec("/usr/bin/ffmpeg -f video4linux2 -i /dev/video0 -frames:v 1
-vf scale=298:168 -y /var/www/html/photo/security.png");
```

```

echo "
<CiscoIPPhoneImageFile>
<Title>Camera service</Title>
<Prompt>Cam 1</Prompt>
<LocationX>0</LocationX>
<LocationY>0</LocationY>
<URL>http://10.0.0.1/photo/security.png</URL>
</CiscoIPPhoneImageFile>
";
?>

```

Soubor `/dev/video0`, jehož vlastník `root` a skupina `video` mají právo čtení a zápisu viz Obr. 25. Aby vše fungovalo, bylo nutné přidat uživatele `www-data`, pod kterým běží webový server Apache, do skupiny `video`.

```

pi@raspberrypi:~$ ls -l /dev/video0
crw-rw----+ 1 root video 81, 0 pro 16 17:17 /dev/video0
pi@raspberrypi:~$ █

```

Obr. 25: Práva pro soubor `video0`

Přidání uživatele `www-data` do skupiny `video0`:

```
sudo usermod -a -G video www-data
```

### Soubor `headers.php`

Funkce `header()` odešle HTTP hlavičku telefonu. Tato funkce zajistí, aby součástí hlavičky byl řetězec `"Content-type: text/xml"`, který nastaví MIME typ na `text/xml`. Řetězec `"refresh: 1"` aktualizuje obsah stránky každou sekundu. Tento interval je pro Cisco IP telefony omezen na 1 s. Funkce `header()` musí být použita před tím, než se vypíše jakýkoli výstup. [52]

Obsah souboru `headers.php`:

```

<?php
header("Content-type: text/xml");
header("refresh: 1;");
?>

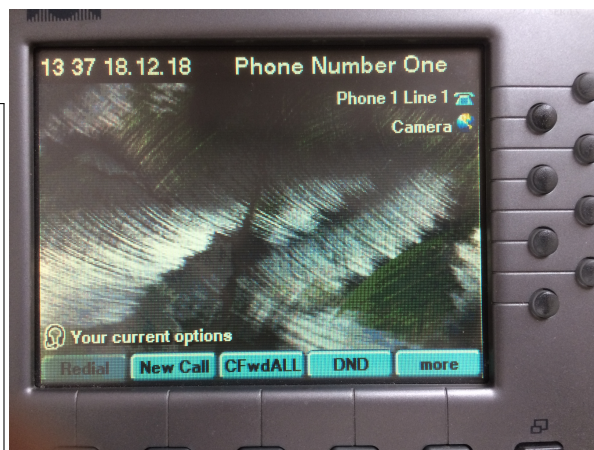
```

### Nastavení specializovaného tlačítka

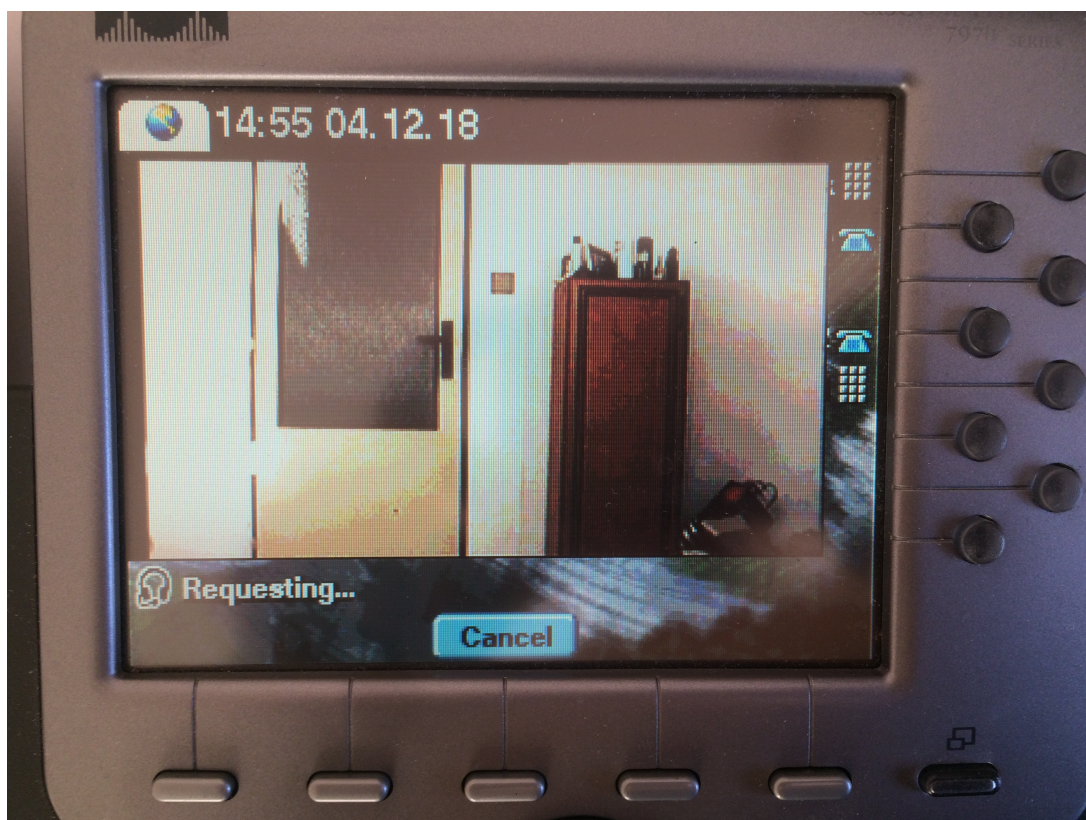
Telefon je vybaven programovatelnými tlačítky, které se nacházejí nalevo od displeje. Druhé programovatelné tlačítko v pořadí bylo nastaveno pro vyžádání služby Camera. Přidalo se nastavení `button = service, Camera, http://10.0.0.1/photo.php` do souboru `sccp.conf` viz Obr. 26. Tímto se nakonfigurovalo specializované tlačítko viz Obr. 27, jehož stisknutím se spustila služba Camera viz Obr. 28. [53]

```
[SEP001A6D876CFD]
description = Phone Number One
addon = 7914
devicetype = 7971
park = off
button = line, 98011, default
button = service, Camera, http://10.0.0.1/photo.php
cfwdall = off
type = device
keepalive = 60
;tzoffset = +2
transfer = on
park = on
cfwdall = off
cfwdbusy = off
cfwdnoanswer = off
directed_pickup = on
```

Obr. 26: Nastavení v sccp.conf



Obr. 27: Tlačítko pro spuštění služby Camera



Obr. 28: Služba kamera

## 4 Vyhodnocení

Teoretická část této bakalářské práce byla věnována popisu zařízení, technologií a charakteristiky použitých nástrojů. Tato část se zabývala termíny jako kompozitní video, PAL, Video4linux2 či Videobuf2 a rovněž Cisco IP telefonem 7971G, Sony kamerou, nástrojem ffmpeg či ústřednou Asterisk. Tímto byl splněn cíl: analýza zpracování analogového AV signálu z kamery pomocí embedded systémů. Dále se soustředila na XML objekty podporované telefonem, na open-source nástroje ze skupiny LAMP a na schopnost telefonu vykonat HTTP spojení s webovým serverem.

Na základě získaných poznatků v teoretické části bylo realizováno zobrazení obrazu z dodané analogové kamery na Cisco IP telefonu. K dosažení cílu byly využity nástroje a prostředky popsané v teoretické části. V praktické části byla vytvořena služba, která byla nazvána Camera. Tato služba zobrazuje snímky z kamery na displeji telefonu a uživatel si ji může spustit z hlavní nabídky služeb *Services* nebo přes specializované tlačítko. Její implementace se nabízí jako možnost rozšíření stávající infrastruktury využívající analogové kamery k zabezpečení prostor nebo jako doplněk ke klasickému zvonku u vstupních dveří. Výstup služby je zachycen na Obr. 28 výše.

V navazující práci bych chtěl pracovat na přidání další kamery a využití schopnosti kamery Sony EVI-D31 otáčet, naklánět a zaostřovat díky rozhraní RS-232C, kterým je vybavena.

## Literatura

- [1] Václavík L.: Raspberry představilo svůj nejlepší minipočítač. Pi 3 Model B+ má rychlejší CPU i síť [online].2018, dostupné z: <https://www.cnews.cz/raspberry-pi-3-model-b-plus-predstaveni> [cit. 2018-12-6].
- [2] Raspberry Pi 3 Model B+ [online].2018, dostupné z: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/> [cit. 2018-12-6].
- [3] Welcome to Raspbian [online].2018, dostupné z: <https://www.raspbian.org/FrontPage> [cit. 2018-12-8].
- [4] Raspbian [online].2018, dostupné z: <https://www.raspberrypi.org/downloads/raspbian/> [cit. 2018-12-8].
- [5] Raspbian Images [online].2018, dostupné z: <https://www.raspbian.org/RaspbianImages> [cit. 2018-12-8].
- [6] Joseph J. Janus Sony EVI-D30/D31 Pan/Tilt/Zoom Color Video Cameras [online].2017, dostupné z: <http://j3soft.net/webcam/evi-d30.htm> [cit. 2018-12-9].
- [7] Silva R. Composite Video - The Basics [online].2018, dostupné z: <https://www.lifewire.com/composite-video-the-basics-1846869> [cit. 2018-12-9].
- [8] Video Basics [online].2002, dostupné z: <https://www.maximintegrated.com/en/app-notes/index.mvp/id/734> [cit. 2018-12-9].
- [9] Pazderák J., Košťál E. Historie televize [online].2007, dostupné z: [http://radio.feld.cvut.cz/media/resources/Historie\\_Televize.pdf](http://radio.feld.cvut.cz/media/resources/Historie_Televize.pdf) [cit. 2018-12-9].
- [10] Analogové zdrojové kódování [online].2017, dostupné z: <https://moodle.fel.cvut.cz/mod/resource/view.php?id=48677> [cit. 2018-12-9].
- [11] Easycap [online].2016, dostupné z: <https://linuxtv.org/wiki/index.php/Easycap> [cit. 2018-12-9].
- [12] Easycap Identification [online].2016, dostupné z: <https://linuxtv.org/wiki/index.php/Easycap#Identification> [cit. 2018-12-9].
- [13] USBTV007 EasyCAP [online].2016, dostupné z: [https://linuxtv.org/wiki/index.php/Easycap#USBTV007\\_EasyCAP](https://linuxtv.org/wiki/index.php/Easycap#USBTV007_EasyCAP) [cit. 2018-12-9].
- [14] Krátký R. Jaderné noviny - Video4Linux2 API: úvod [online].2006, dostupné z: <http://www.abclinuxu.cz/clanky/jaderne-noviny/jaderne-noviny-video4linux2-api-uvod> [cit. 2018-12-11].
- [15] What is V4L or DVB? [online].2015, dostupné z: [https://www.linuxtv.org/wiki/index.php/What\\_is\\_V4L\\_or\\_DVB%3F](https://www.linuxtv.org/wiki/index.php/What_is_V4L_or_DVB%3F) [cit. 2018-12-11].
- [16] Tranter J. The Raspberry Pi Camera Module [online].2014, dostupné z: <https://www.ics.com/blog/raspberry-pi-camera-module> [cit. 2018-12-11].
- [17] Szyprowski M. Videobuf2 framework [online].2010, dostupné z: <https://lwn.net/Articles/416649/> [cit. 2018-12-11].

- [18] Streaming I/O (Memory Mapping) [online].2018, dostupé z: <https://linuxtv.org/downloads/v4l-dvb-apis/uapi/v4l/mmap.html> [cit. 2018-12-11].
- [19] About FFmpeg [online].2018, dostupé z: <https://www.ffmpeg.org/about.html> [cit. 2018-12-12].
- [20] ffmpeg Documentation: Description [online].2018, dostupé z: <https://www.ffmpeg.org/ffmpeg.html#Description> [cit. 2018-12-12].
- [21] ffmpeg Documentation: Detailed description [online].2018, dostupé z: <https://www.ffmpeg.org/ffmpeg.html#Detailed-description> [cit. 2018-12-12].
- [22] ffmpeg Documentation: Filtering [online].2018, dostupé z: <https://www.ffmpeg.org/ffmpeg.html#Filtering> [cit. 2018-12-12].
- [23] ffmpeg Documentation: Simple filtergraphs [online].2018, dostupé z: <https://www.ffmpeg.org/ffmpeg.html#Simple-filtergraphs> [cit. 2018-12-12].
- [24] ffmpeg Documentation: Complex filtergraphs [online].2018, dostupé z: <https://www.ffmpeg.org/ffmpeg.html#Complex-filtergraphs> [cit. 2018-12-12].
- [25] Cisco Unified IP Phone 7900 Series [online].2018, dostupé z: <https://www.cisco.com/c/en/us/products/collaboration-endpoints/unified-ip-phone-7900-series/index.html> [cit. 2018-12-13].
- [26] CiscoIPPhone XML Objects [online].2018, dostupé z: [https://www.cisco.com/c/en/us/td/docs/voice\\_ip\\_comm/cuipph/all\\_models/xsi/7\\_0/english/programming/guide/70xsi/xsi70obj.html#wp1033622](https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cuipph/all_models/xsi/7_0/english/programming/guide/70xsi/xsi70obj.html#wp1033622) [cit. 2018-12-13].
- [27] Cisco Unified Video Advantage [online].2018, dostupé z: <https://www.cisco.com/c/en/us/products/video/unified-video-advantage/index.html> [cit. 2018-12-13].
- [28] Cisco Unified IP Phone Guide 7970G and 7971G for Cisco Unified Communications Manager Express Version 7.1 [online].2009, dostupé z: [https://www.cisco.com/c/dam/en/us/td/docs/voice\\_ip\\_comm/cucme/ip\\_phones/7970\\_71/4-4/english/user/guide/ug7071.pdf](https://www.cisco.com/c/dam/en/us/td/docs/voice_ip_comm/cucme/ip_phones/7970_71/4-4/english/user/guide/ug7071.pdf) [cit. 2018-12-13].
- [29] Mitchell B.: Trivial File Transfer Protocol [online].2018, dostupé z: <https://www.lifewire.com/definition-of-tftp-817576> [cit. 2018-12-12].
- [30] Asterisk phone cisco 79x1 xml configuration files for SIP [online].2018, dostupé z: <https://www.voip-info.org/asterisk-phone-cisco-79x1-xml-configuration-files-for-sip/> [cit. 2018-12-12].
- [31] Červenka: Asterisk PBX [online].2010, dostupé z: <https://sip.cesnet.cz/cs/swahw/asterisk> [cit. 2018-12-6].
- [32] Newton R.: Asterisk Architecture, The Big Picture [online].2014, dostupé z: <https://wiki.asterisk.org/wiki/display/AST/Asterisk+Architecture%2C+The+Big+Picture> [cit. 2018-12-11].
- [33] Newton R.: Types of Asterisk Modules [online].2014, dostupé z: <https://wiki.asterisk.org/wiki/display/AST/Types+of+Asterisk+Modules> [cit. 2018-12-11].

- [34] Binder T.: SPRÁVA A KONFIGURACE VOIP ÚSTŘEDNÝ ASTERISK [online].2008, dostupé z: [https://www.vutbr.cz/www\\_base/zav\\_prace\\_soubor\\_verejne.php?file\\_id=8082](https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=8082) [cit. 2018-12-6].
- [35] Understanding IP Telephony Protocols [online].2018, dostupé z: [https://www.cisco.com/c/en/us/td/docs/voice\\_ip\\_comm/cucm/admin/8\\_5\\_1/ccmsys/accm-851-cm/a08procl.pdf](https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/admin/8_5_1/ccmsys/accm-851-cm/a08procl.pdf) [cit. 2018-12-6].
- [36] Channel Driver Modules [online].2018, dostupé z: <https://wiki.asterisk.org/wiki/display/AST/Channel+Driver+Modules> [cit. 2018-12-6].
- [37] What is chan\_sccp? [online].2018, dostupé z: <http://chan-sccp.org> [cit. 2018-12-6].
- [38] Custom Client Services Overview [online].2018, dostupé z: [https://www.cisco.com/c/en/us/td/docs/voice\\_ip\\_comm/cuipph/all\\_models/xsi/9-1-1/CUIP\\_BK\\_P82B3B16\\_00\\_phones-services-application-development-notes/CUIP\\_BK\\_P82B3B16\\_00\\_phones-services-application-development-notes\\_chapter\\_01.html](https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cuipph/all_models/xsi/9-1-1/CUIP_BK_P82B3B16_00_phones-services-application-development-notes/CUIP_BK_P82B3B16_00_phones-services-application-development-notes_chapter_01.html) [cit. 2018-12-6].
- [39] PHP 5 Introduction [online].2018, dostupé z: [https://www.w3schools.com/php/php\\_intro.asp](https://www.w3schools.com/php/php_intro.asp) [cit. 2018-12-12].
- [40] The Number One HTTP Server On The Internet [online].2018, dostupé z: <https://httpd.apache.org/> [cit. 2018-12-12].
- [41] HTTP Requests and Header Settings [online].2018, dostupé z: [https://www.cisco.com/c/en/us/td/docs/voice\\_ip\\_comm/cuipph/all\\_models/xsi/9-1-1/CUIP\\_BK\\_P82B3B16\\_00\\_phones-services-application-development-notes/CUIP\\_BK\\_P82B3B16\\_00\\_phones-services-application-development-notes\\_chapter\\_0110.html](https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cuipph/all_models/xsi/9-1-1/CUIP_BK_P82B3B16_00_phones-services-application-development-notes/CUIP_BK_P82B3B16_00_phones-services-application-development-notes_chapter_0110.html) [cit. 2018-12-12].
- [42] HTTP Methods [online].2018, dostupé z: [https://www.w3schools.com/tags/ref\\_httpmethods.asp](https://www.w3schools.com/tags/ref_httpmethods.asp) [cit. 2018-12-12].
- [43] Cisco IP Phone Displays "XML Parse Error"when Accessing the Cisco Corporate Directory [online].2009, dostupé z: <https://www.cisco.com/c/en/us/support/docs/voice-unified-communications/unified-ip-phone-7900-series/100895-xml-parse-error.html> [cit. 2018-12-12].
- [44] Installing operating system images [online].2018, dostupé z: <https://www.raspberrypi.org/documentation/installation/installing-images/README.md> [cit. 2018-12-13].
- [45] Základní instalace Asterisk na Debian 6 (Squeeze) [online].2017, dostupé z: <http://wiki.4smart.cz/doku.php/asterisk-zakladni-instalace-asterisk-na-debian-6-squeeze> [cit. 2018-12-6].
- [46] Asterisk CLI [online].2018, dostupé z: <https://www.voip-info.org/asterisk-cli> [cit. 2018-12-13].
- [47] Newton R.: Configuring the Asterisk Module Loader [online].2014, dostupé z: <https://wiki.asterisk.org/wiki/display/AST/Configuring+the+Asterisk+Module+Loader> [cit. 2018-12-10].
- [48] <https://github.com/nguyet38/bakalarka/blob/master/sccp.conf>



- [49] <https://github.com/nguyet38/bakalarka/blob/master/SEP001A6D876CFD.cnf.xml>
- [50] Groot D. Setup chan-sccp-b [online].2017, dostupé z: <https://raw.githubusercontent.com/chan-sccp/chan-sccp/develop/conf/sccp.conf.annotated> [cit. 2018-12-10].
- [51] Cisco Unified IP Phone Services Application Development Notes, Release 8.5(1) [online].2018, dostupé z: [https://www.cisco.com/c/en/us/td/docs/voice\\_ip\\_comm/cuipph/all\\_models/xsi/9-1-1/CUIP\\_BK\\_P82B3B16\\_00\\_phones-services-application-development-notes/CUIP\\_BK\\_P82B3B16\\_00\\_phones-services-application-development-notes\\_chapter\\_0110.html#CUIP\\_RF\\_M138E4FB\\_00](https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cuipph/all_models/xsi/9-1-1/CUIP_BK_P82B3B16_00_phones-services-application-development-notes/CUIP_BK_P82B3B16_00_phones-services-application-development-notes_chapter_0110.html#CUIP_RF_M138E4FB_00) [cit. 2018-12-12].
- [52] Network Functions: header [online].2018, dostupé z: <http://php.net/manual/en/function.header.php> [cit. 2018-12-15].
- [53] Button Configuration [online].2016, dostupé z: <https://github.com/chan-sccp/chan-sccp/wiki/Button-Configuration> [cit. 2018-12-16].