



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA PODNIKATELSKÁ

FACULTY OF BUSINESS AND MANAGEMENT

## ÚSTAV INFORMATIKY

INSTITUTE OF INFORMATICS

# NÁVRH A REALIZACE ŘEŠENÍ IPTV VE FIREMNÍM PROSTŘEDÍ

DESIGN AND IMPLEMENTATION OF IPTV SOLUTION IN A CORPORATE ENVIRONMENT

## DIPLOMOVÁ PRÁCE

MASTER'S THESIS

## AUTOR PRÁCE

AUTHOR

**Bc. Michal Psota**

## VEDOUCÍ PRÁCE

SUPERVISOR

**Ing. Petr Dydowicz, Ph.D.**

**BRNO 2024**

# Zadání diplomové práce

Ústav:	Ústav informatiky
Student:	<b>Bc. Michal Psota</b>
Vedoucí práce:	<b>Ing. Petr Dydowicz, Ph.D.</b>
Akademický rok:	2023/24
Studijní program:	Informační management

Garant studijního programu Vám v souladu se zákonem č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů a se Studijním a zkušebním řádem VUT v Brně zadává diplomovou práci s názvem:

## Návrh a realizace řešení IPTV ve firemním prostředí

### Charakteristika problematiky úkolu:

Úvod  
Vymezení problému a cíle práce  
Teoretická východiska práce  
Analýza problému a současné situace  
Vlastní návrh řešení, přínos práce  
Závěr  
Seznam použité literatury

### Cíle, kterých má být dosaženo:

Cílem je navrhnout a realizovat vhodné řešení pro příjem signálu DVB, jeho zpracování a následnou distribuci klientům společnosti pomocí protokolu IP. Součástí cíle je zanalyzovat současný stav včetně zjištění dostupnosti existujících softwarových řešení pro poskytování IPTV služeb a stanovit požadavky na nové řešení. Na základě těchto požadavků bude realizováno odpovídající řešení, které se bude opírat o příslušná teoretická východiska. Výsledkem bude plnohodnotný funkční celek, jehož hlavním úkolem bude poskytování služby IPTV.

### Základní literární prameny:

DONAHUE, Gary A. Kompletní průvodce síťového experta. Brno: Computer Press, 2009. ISBN 978-80-251-2247-1.

JIROVSKÝ, Václav. Vademecum správce sítě. Praha: Grada, 2001. ISBN 80-7169-745-1.

KUROSE, James F. a ROSS, Keith W. Počítačové sítě. Brno: Computer Press, 2014. ISBN 978-80-251-3825-0.

LEGÍŇ, Martin. Televizní technika DVB-T. Praha: BEN - technická literatura, 2006. ISBN 80-7300-204-3.

NEMETH, Evi; SNYDER, Garth a HEIN, Trent R. Linux: kompletní příručka administrátora : 2. aktualizované vydání. Administrace. Brno: Computer Press, 2008. ISBN 978-80-251-2410-9.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2023/24

V Brně dne 4.2.2024

L. S.

---

doc. Ing. Miloš Koch, CSc.  
garant

---

doc. Ing. Vojtěch Bartoš, Ph.D.  
děkan

## **Abstrakt**

Diplomová práce se zaměřuje na návrh a realizaci komplexního funkčního řešení pro příjem signálu DVB-T2, jeho zpracování a následnou distribuci klientům společnosti pomocí IPTV. Součástí realizace je zajištění jednoduché a přehledné správy vytvořeného řešení zahrnující průběžné udržování aktuální programové nabídky.

## **Klíčová slova**

IPTV, DVB-T2, multicast, IGMP, Linux, Python

## **Abstract**

This master's thesis focuses on the design and implementation of a complex functional solution for DVB-T2 signal reception, its processing and subsequent distribution to the company's clients via IPTV. Part of the implementation is to ensure simple and clear management of the created solution including continuous maintenance of the current program offer.

## **Key words**

IPTV, DVB-T2, multicast, IGMP, Linux, Python

### **Bibliografická citace**

PSOTA, Michal. *Návrh a realizace řešení IPTV ve firemním prostředí* [online]. Brno, 2024 [cit. 2024-04-30]. Dostupné z: <https://www.vut.cz/studenti/zav-prace/detail/157348>. Diplomová práce. Vysoké učení technické v Brně, Fakulta podnikatelská, Ústav informatiky. Vedoucí práce Ing. Petr Dydowicz, Ph.D.

### **Čestné prohlášení**

Prohlašuji, že předložená diplomová práce je původní a zpracoval jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem ve své práci neporušil autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

V Brně dne 30. dubna 2024

.....  
podpis autora

### **Poděkování**

Rád bych poděkoval své rodině i přátelům za jejich podporu během mého dosavadního studia.

Také chci poděkovat panu Ing. Petru Dydowiczovi, Ph.D. za vedení této diplomové práce.

# OBSAH

<b>ÚVOD</b> .....	<b>11</b>
<b>VYMEZENÍ PROBLÉMU A CÍLE PRÁCE</b> .....	<b>12</b>
<b>1 TEORETICKÁ VÝCHODISKA</b> .....	<b>13</b>
1.1 Počítačová síť .....	13
1.2 Klasifikace počítačových sítí .....	13
1.2.1 Topologie sítí .....	13
1.2.2 Rozlehlost sítě.....	15
1.3 Síťové modely a architektury .....	16
1.3.1 Referenční ISO/OSI model .....	16
1.3.2 Architektura TCP/IP .....	17
1.3.3 Architektura Ethernet.....	17
1.4 Aktivní prvky počítačové sítě.....	18
1.4.1 Přepínač (switch) .....	18
1.4.2 Směrovač (router) .....	19
1.5 Protokoly TCP, UDP, RTP, SAP, IGMP.....	19
1.5.1 Protokol TCP (Transmission Control Protocol) .....	19
1.5.2 Protokol UDP (User Datagram Protocol) .....	19
1.5.3 Protokol RTP (Real-Time transport Protocol).....	19
1.5.4 Protokol SAP (Session Announcement Protocol) .....	20
1.5.5 Protokol IGMP (Internet Group Management Protocol).....	22
1.6 Přenosové metody .....	23
1.6.1 Unicast .....	23
1.6.2 Multicast .....	23
1.6.3 Broadcast .....	24
1.7 Operační systémy.....	25
1.7.1 Linux.....	25
1.8 Programovací jazyk Python .....	26
1.9 DVB (Digital Video Broadcasting) .....	26
1.9.1 DVB-T (Digital Video Broadcasting Terrestrial).....	26
1.9.2 DVB-T2 (Digital Video Broadcasting Terrestrial 2).....	28
1.10 Vývojový diagram .....	29



1.11 SWOT analýza.....	30
1.11.1 IFE, EFE matice.....	32
<b>2 ANALÝZA PROBLÉMU A SOUČASNÉ SITUACE .....</b>	<b>33</b>
2.1 O společnosti.....	33
2.1.1 Popis.....	33
2.1.2 Historie.....	33
2.1.3 Současnost .....	33
2.2 Komunikační infrastruktura společnosti.....	34
2.3 Hardwarové vybavení .....	34
2.4 Dostupná existující řešení.....	34
2.4.1 TSDuck .....	34
2.4.2 MuMuDVB .....	35
2.4.3 DVBlasť .....	35
2.4.4 Souhrn poznatků .....	36
2.5 Požadavky na řešení.....	36
2.6 SWOT analýza.....	37
2.6.1 Výstup.....	38
<b>3 VLASTNÍ NÁVRH ŘEŠENÍ .....</b>	<b>41</b>
3.1 Funkční předpoklady .....	41
3.2 Architektura a koncept navrhovaného řešení.....	41
3.2.1 Server .....	42
3.2.2 Televizní karta pro příjem DVB-T2 .....	42
3.3 Návrh řešení.....	43
3.3.1 Ověření dostupnosti multiplexů pro oblast Brna a okolí .....	43
3.3.2 Plán adresního prostoru pro IPTV .....	43
3.3.3 Poskytování obsahu ze sítě DVB-T2 do sítě IP .....	44
3.3.4 Oznamování o dostupnosti multimediálního vysílání v síti.....	46
3.3.5 Regionální informační kanál.....	47
3.3.6 Automatické přeladění .....	48
3.3.7 Soubor formátu M3U8.....	52
3.3.8 Záložní řešení.....	52
3.4 Realizace řešení .....	53

3.4.1 Příprava a ladění .....	53
3.4.2 Konfigurace DVBlasť .....	57
3.4.3 Vytvoření služby pro DVBlasť .....	60
3.4.4 Konfigurace SAP .....	61
3.4.5 Poskytování regionálního informačního kanálu .....	63
3.4.6 Automatické přeladění .....	64
3.4.7 Automatizace automatického přeladění .....	78
3.5 Přijímání IPTV .....	78
3.5.1 Aplikace .....	79
3.5.2 Přijímací zařízení .....	83
3.6 Přehled finančních nákladů .....	85
3.7 Přínosy navrženého řešení .....	87
<b>ZÁVĚR .....</b>	<b>88</b>
<b>SEZNAM POUŽITÝCH ZDROJŮ .....</b>	<b>89</b>
<b>SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ .....</b>	<b>92</b>
<b>SEZNAM POUŽITÝCH OBRÁZKŮ .....</b>	<b>94</b>
<b>SEZNAM POUŽITÝCH TABULEK .....</b>	<b>97</b>
<b>SEZNAM POUŽITÝCH DIAGRAMŮ .....</b>	<b>98</b>

# ÚVOD

Tato diplomová práce se zaměřuje na návrh a realizaci řešení IPTV ve firemním prostředí s cílem rozšířit poskytované služby všem klientům společnosti. Klienti tak získají možnost sledování volně šířených TV programů a rozhlasových stanic ze sítě DVB-T2 v IP síti. Klienti mohou pro příjem používat buď osobní počítače, externí set-top-boxy anebo chytré televize, případně jakákoli jiná kompatibilní multimediální centra.

Předložená práce uvádí konkrétní návrh a předpoklady, na základě kterých je realizováno funkční řešení, doprovázené odpovídajícím popisem. V této souvislosti na návrh navazuje realizace, která blíže uvádí použité prostředky a komponenty, jejich konfigurace a doplňující aspekty v širším kontextu. Současně je způsob návrhu i realizace doplněn vhodným odůvodněním použitých technologií, a to včetně dalších alternativ.

Struktura práce je rozdělena do tří základních kapitol. První kapitola obsahuje teoretická východiska, kde jsou vysvětleny nezbytné pojmy, technologie, modely, standardy a veškeré související náležitosti z oblasti počítačových sítí, přenosových protokolů atd.

Druhá kapitola obsahuje popis společnosti, pro kterou je tato práce určena. Dále tato kapitola uvádí přehled o současném stavu možností, jakými lze řešit příjem pozemního televizního vysílání DVB-T2 z hlediska softwarových aplikací. Jednotlivé aplikace jsou vzájemně porovnány, vyhodnoceny a na tomto základě jsou sestaveny nové požadavky, jejichž splnění se očekává od tohoto celkového řešení.

Ve třetí kapitole uvádím vlastní návrh řešení doplněný vývojovými diagramy. Na tento návrh navazuje realizace odpovídající softwarové podpory, která sestává z praktických fragmentů kódu vývoje s vysvětlujícím doprovodným popisem.

Před závěrem práce se nachází kapitola pojednávající o možnostech příjmu IPTV s konkrétními ukázkami existujících aplikací s podporou různých zařízení.

Závěr práce je věnován odhadu finančních nákladů na realizaci a posouzení přínosů, které toto komplexní řešení nabízí.

## VYMEZENÍ PROBLÉMU A CÍLE PRÁCE

Cílem této diplomové práce je návrh a realizace vhodného řešení pro příjem signálu DVB, včetně jeho zpracování, až po distribuci v podobě IPTV s využitím existující počítačové sítě společnosti. Součástí cíle je zajistit dostupnost všech televizních programů, jež jsou volně šířené v rámci terestriálního vysílání DVB-T2 pro klienty společnosti pomocí nové služby IPTV.

Návrh zahrnuje různé softwarové možnosti přijímání terestriálního vysílání, včetně popisu způsobů následného zpracování, až po přenos signálu do přijímačů klientů. Dílčím cílem je navrhnout konkrétní komponenty nezbytné pro následnou realizaci.

V této práci jsou představeny prostředky, s jejichž pomocí chci dosáhnout vytyčených cílů. Následně provedu analýzu prostředků v kontextu současného stavu za účelem zjištění, jaké v současné době existují možnosti přijímání, zpracování a distribuce terestriálního televizního signálu. Na základě zmíněné analýzy provedu vlastní propracovaný návrh řešení, na jehož základě bude možné tento návrh realizovat.

První kapitola obsahuje odborný popis používaných prostředků užitých ve třetí kapitole. Analytická část práce byla sestavena na základě následující metodologie. Práce začíná identifikací a popisem firemního prostředí, pro které je tato práce určena.

Dále byl proveden průzkum současných existujících řešení, na jehož základě bylo možné uskutečnit vzájemné porovnání a zvážení výhod a nevýhod každého z nich. V této souvislosti byly sestaveny požadavky na nové komplexní řešení. Tyto požadavky byly doplněny o zjištěné poznatky vyplynulé ze zadání managementu společnosti. Po vzoru analytické části byla zvolena jednoznačná metodologie i pro poslední kapitolu 3, kterou je vlastní návrh řešení. Celý návrh řešení se opírá o předpoklady, jež musí být splněny. Na tomto základě byla navržena architektura a procesy, které jsou klíčové pro samotnou realizaci řešení.

# 1 TEORETICKÁ VÝCHODISKA

## 1.1 Počítačová síť

Počítačová síť umožňuje propojit více zařízení, která jsou vybavena možností vzájemné komunikace mezi sebou. Tuto síť tvoří aktivní a pasivní prvky. Mezi nejpoužívanější komunikační protokoly patří Ethernet.

## 1.2 Klasifikace počítačových sítí

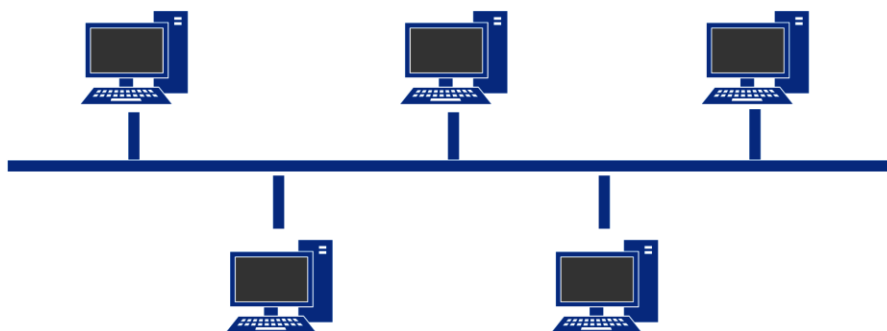
Počítačové sítě je možné klasifikovat dle různých hledisek, kterými jsou například: topologie, rozlehlost, způsob přepojování, uspořádání uzlů atd. V rámci této práce je klíčová klasifikace dle rozlehlosti i topologie.

### 1.2.1 Topologie sítí

„Topologie je způsob, jakým jsou stanice v síti propojeny.“ (1) V základní úrovni lze topologii počítačové sítě rozdělit na fyzickou a logickou. Fyzická topologie popisuje reálnou instalaci a zapojení kabeláže vedoucí k jednotlivým uzlům. Logická topologie vyjadřuje způsob propojení v rámci fyzické topologie (2). To znamená, že se fyzická a logická topologie může lišit. Logickou topologii ovlivňuje konfigurace aktivních síťových prvků, přičemž tuto konfiguraci obvykle provádí správce sítě, příp. externí firma.

#### Sběrníková topologie

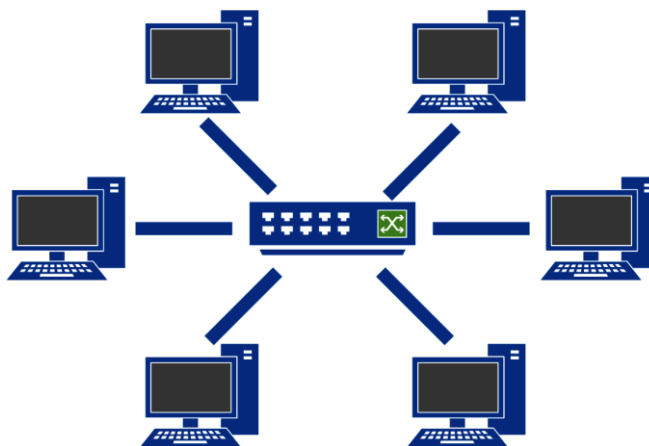
Každá příslušná stanice je připojena k síti pomocí průběžného vedení od stanice ke stanici. Jednotlivé stanice jsou připojeny pomocí T-kusu. Oba konce vedení musí být zakončeny pomocí terminátoru. Tato topologie se v praxi často používá v případech, kde je instalován koaxiální kabel. Nespornou výhodou sběrníkové topologie je významná úspora materiálu při instalaci, protože jsou všechny stanice připojeny k jednomu kabelu. Naopak nevýhodou je, že v případě poruchy se obtížně diagnostikuje závada, protože je-li kabel poškozen na jednom místě, pak není funkční celá síť. Další nevýhodou je vysoký počet spojů. V současné době se využívá sběrníkové topologie minimálně. (5)



**Obrázek č. 1: Schéma sběrnice topologie**  
(Zdroj: Vlastní zpracování)

### Hvězdicová topologie

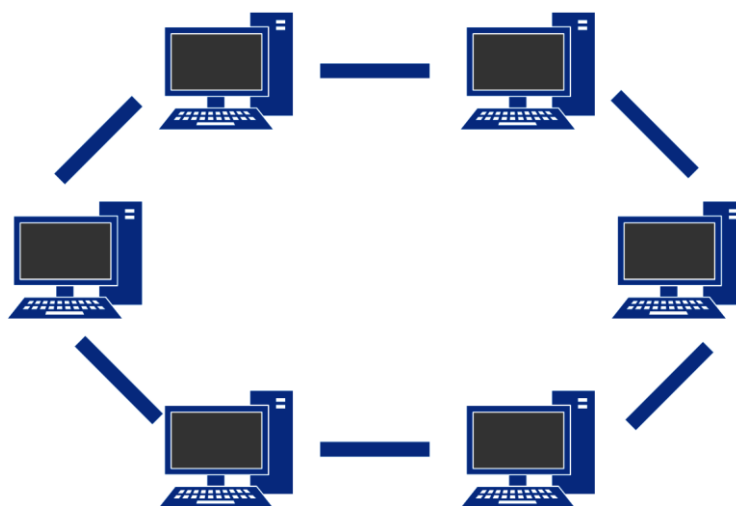
Každá stanice je připojena k centrálnímu síťovému prvku (obvykle přepínači) jedním kabelem. Výhodou je to, že v případě poruchy nedojde k paralýze celé sítě, nýbrž pouze k výpadku spojení příslušné stanice, kde se potíž vyskytla. Další výhodou je, že se snadněji zjišťuje závada než u sběrnice topologie. V současné době se jedná o nejpoužívanější topologii. (1)



**Obrázek č. 2: Schéma hvězdice topologie**  
(Zdroj: Vlastní zpracování)

### Kruhová topologie

V tomto případě jsou uzly zapojeny do kruhu, kde si stanice postupně předávají tzv. token. Tato topologie byla využívána například v síti Token ring. Nevýhody jsou podobné jako u sběrnice topologie, protože přerušení kabelu na libovolném místě má za důsledek poruchu sítě. (5)



**Obrázek č. 3: Schéma kruhové topologie**  
(Zdroj: Vlastní zpracování)

### 1.2.2 Rozlehlost sítě

Vzhledem k široké rozlehlosti sítí lze stěží vymezit hranice mezi pojmy uvedenými níže.

#### **LAN (Local Area Network)**

Místní počítačová síť je z hlediska své velikosti používána nejčastěji v domácnostech nebo menších společnostech. V domácí síti se obvykle vyskytují jednotky, nejvýše nízké desítky zařízení připojených pomocí jednoho síťového prvku. Podniková síť bývá podstatně rozlehlejší, což znamená, že množství připojených zařízení se pohybuje od vyšších desítek až po tisíce různých zařízení. Podnik může mít v areálu více hal nebo budov, které je zapotřebí propojit. I v tomto případě se jedná o síť LAN, z čehož vyplývá, že v této síti jsou propojeny uzly na vzdálenosti stovek metrů či jednotek kilometrů.

#### **WAN (Wide Area Network)**

Volný překlad angl. zkratky WAN vyjadřuje rozlehlou síť. Síť WAN sestává zejména z více vzájemně propojených sítí LAN (1). Toto propojování je realizováno buď pomocí kabelového nebo bezdrátového propojení mezi dvěma vzdálenými body (1,3). Síť WAN svou rozlehlostí zasahuje až do celosvětové sítě Internet, což ale není pro tuto práci relevantní (1).

#### **MAN (Metropolitan Area Network)**

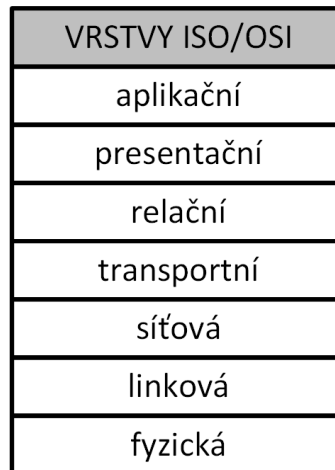
Síť MAN tvoří několik menších sítí LAN a její rozlehlost odpovídá rozsahu města či regionu. Z hlediska velikosti je největší síť WAN, menší síť je MAN a malou sítí je LAN.

(1)

## 1.3 Síťové modely a architektury

### 1.3.1 Referenční ISO/OSI model

Tento model vypracovala Mezinárodní organizace pro normalizaci ISO na počátku 80. let 20. století (1). „Referenční model ISO/OSI je obecným modelem definující síťovou komunikaci“ (2). Tento model popisuje 7 vrstev, které na sebe navzájem navazují. Jednotlivé vrstvy, ze kterých tento model sestává, jsou vyobrazeny na obrázku níže.



**Obrázek č. 4: Vyobrazení vrstev modelu ISO/OSI**  
(Zdroj: Vlastní zpracování)

První tři vrstvy shora (aplikační, presentační, relační), jsou orientovány aplikačně. Ostatní vrstvy se zabývají přenosem. Tyto vrstvy jsou řazené zdola podle důležitosti. Obecný způsob komunikace mezi vrstvami (vertikální spolupráce) je realizována následovně. Podřízená vrstva zpracuje úkol a následně jej předá vrstvě nadřazené. Rozsah a způsob zpracování úkolu není tímto modelem specifikován, takže řešení obou součástí je v kompetenci výrobce síťového zařízení. Model ISO/OSI doporučuje způsob zajištění vzájemné spolupráce mezi dvěma stejnými vrstvami modelu (1).

Nejnižší vyobrazená vrstva, na obrázku č. 4, je fyzická vrstva. Tato vrstva reprezentuje kabelážní systém. Je nezbytné, aby fyzická vrstva disponovala vysokou kvalitou, protože tvoří důležitý předpoklad pro správné fungování celého komunikačního systému. Použití méně kvalitních prostředků, při návrhu síťové infrastruktury, může mít za následek pozdější významnou degradaci přenosových parametrů, například kvůli nízké čistotě mědi ve vodičích. K fyzické vrstvě náleží, kromě kabeláže, některé aktivní síťové prvky, například: přijímač, vysílač a zesilovač jednotlivých portů (2).



### 1.3.2 Architektura TCP/IP

Tato architektura je odvozena od ISO/OSI modelu. Ve srovnání s modelem ISO/OSI, architektura TCP/IP sestává pouze ze čtyř vrstev. Takové uspořádání je v současné době nejrozšířenější a stalo se standardem. Linková a fyzická vrstva ISO/OSI modelu je v architektuře TCP/IP reprezentována jako vrstva síťového rozhraní. Vzhledem k existenci různých přenosových síťových technologií, např. Ethernet, není architekturou TCP/IP definována vrstva síťového rozhraní. Síťovou vrstvu tvoří přenosový protokol IP. Transportní vrstva sestává ze dvou protokolů TCP a UDP, které jsou blíže vysvětleny v kapitolách 1.5.1 a 1.5.2. Na jednotné základy aplikací (e-mailová komunikace, přenosy sdílených souborů, webové stránky, ...) je zaměřena aplikační vrstva. (1) Rozdíl mezi modelem ISO/OSI a architekturou TCP/IP je lépe patrný na obrázku níže.

VRSTVY ISO/OSI	VRSTVY TCP/IP
aplikační	aplikační
presentační	
relační	
transportní	transportní
síťová	síťová
linková	Vrstva síťového rozhraní
fyzická	

Obrázek č. 5: Srovnání vrstev modelu ISO/OSI a architektury TCP/IP  
(Zdroj: Vlastní zpracování dle: 1)

### 1.3.3 Architektura Ethernet

Název Ethernet se vztahuje k množině implementací lokálních sítí (LAN), která zahrnuje čtyři základní kategorie:

- Ethernet a specifikaci IEEE 802.3 pro místní síť LAN, které pracují s rychlostí 10Mbit/s na koaxiálním nebo krouceném kabelu,
- Fast Ethernet, specifikaci lokální síť LAN pracující s rychlostí 100 Mbit/s na krouceném kabelu,
- Giga Bit Ethernet, specifikaci sítě, která pracuje s rychlostí 1000 Mbit/s s využitím optických vláken nebo kroucených kabelů,

- Nově projektovanou optickou pátevní architekturu 10 G Ethernet podle standardu 802.3ae. (3, str. 64)

Ethernet je díky své relativní jednoduchosti, vysoké flexibilitě, a snadné implementaci stále nejpoužívanější architekturou navzdory existenci dokonalejších alternativ. (3)

Z hlediska modelu ISO/OSI Ethernet reprezentuje fyzickou a linkovou vrstvu. Ethernet se vyznačuje využíváním kolizní přístupové metody CSMA/CD. Tato metoda vychází z předpokladu, že se signál šíří nekonečně rychle. Tento předpoklad však z fyzikálního hlediska není reálný, protože se každé vlnění šíří konečnou rychlostí. Z tohoto důvodu byly stanoveny maximální vzdálenosti, při kterých ještě může CSMA/CD fungovat. Maximální rozměr sítě lze vymežit termínem kolizní doména. (1)

## **1.4 Aktivní prvky počítačové sítě**

Aktivní prvek sítě je takový prvek, který aktivním způsobem pracuje s datovým signálem.

### **1.4.1 Přepínač (switch)**

V případě Ethernetu je přepínačem zařízení, které předává na druhé vrstvě rámce podle MAC adresy. Přepínač posílá data pouze na určené porty a na každém portu vytvoří tzv. kolizní doménu. To znamená, že je-li přepínač vybaven například osmi porty, přičemž v každém z nich je připojené aktivní zařízení, pak komunikace mezi libovolnými dvěma zařízeními probíhá přes přepínač přímo mezi dvěma příslušnými fyzickými porty. (4) Kromě přepínačů na druhé vrstvě existují také přepínače, které fungují na třetí vrstvě. Avšak namísto rámců jsou přepínány pakety, z čehož plyne, že disponují možností směřovat síťový provoz podobně jako směrovač.

Na trhu existuje široká nabídka přepínačů, které je možné rozdělit do čtyř nejčastěji se vyskytujících obecných kategorií:

- 1) základní přepínače L2 bez managementu (vhodné pro nenáročné domácnosti),
- 2) pokročilé přepínače na L2 s managementem,
- 3) profesionální přepínače L3 s managementem vhodné i do datacenter,
- 4) průmyslové přepínače (používané v náročných podmínkách s vysokými požadavky na funkčnost a odolnost).

### **1.4.2 Směrovač (router)**

Z hlediska modelu ISO/OSI směrovač pracuje na třetí (síťové) vrstvě. Jeho úkolem je směrovat síťový provoz mezi různými segmenty sítě. Směrovač se řídí podle směrovací tabulky.

## **1.5 Protokoly TCP, UDP, RTP, SAP, IGMP**

### **1.5.1 Protokol TCP (Transmission Control Protocol)**

Komunikační přenosový protokol TCP poskytuje spolehlivý přenos dat v prostředí IP a rovněž zajišťuje řízení přenosu a plně duplexní provoz včetně multiplexace dat. Tento protokol odpovídá, z hlediska referenčního modelu ISO/OSI, transportní vrstvě. TCP doručuje nestrukturovaný tok bytů, který je identifikován pořadovými čísly. Díky tomuto značení mohou aplikace předávat souvislé bloky dat. Protokol TCP člení data do segmentů a předává je vrstvě IP k doručení (3).

Spolehlivost přenosu dat je zajištěna na základě metody doručování paketů mezi koncovými uzly. Tato metoda funguje následovně. V propojených sítích se označují sekvence bytů, kdy se očekává potvrzení od protistrany, zda byly přijaty příslušné přenosové segmenty. Pokud segment není přijat v předem definovaném časovém rozmezí, pak se takový segment znovu vyšle (3). Tento mechanismus zajišťuje minimalizaci potenciálně vznikajících problémů během přenosu.

### **1.5.2 Protokol UDP (User Datagram Protocol)**

Protokol UDP rovněž odpovídá, z hlediska referenčního modelu ISO/OSI, transportní vrstvě. Tento protokol se od TCP významně liší, protože UDP nezajišťuje spolehlivost, řízení toků ani nedisponuje funkcí nápravy chyb (3). Díky své jednoduchosti je UDP často využíván pro multimediální přenosy, u kterých není důležité řešit spolehlivost doručení dat.

### **1.5.3 Protokol RTP (Real-Time transport Protocol)**

Tento protokol pracuje na transportní vrstvě ISO/OSI v reálném čase. Nejčastěji se tento protokol používá pro přenos audiovizuálního obsahu. Aplikace obvykle spouští RTP nad UDP, aby využily jeho služeb multiplexování a kontrolních součtů. Oba protokoly přispívají částmi funkcí transportního protokolu. RTP však lze použít i s jinými vhodnými základními síťovými nebo transportními protokoly (např. RTCP) (5).

Při transportu multimédií je hlavní výhodou RTP, oproti prostému UDP, používání časové známky, které umožňují přijímacím zařízením synchronizovat audio a video data a dosáhnout tak plynulého přehrávání. Dále protokol RTP disponuje informacemi o typu přenášených dat (použité kodeky a doplňující vlastnosti přenosu) (6). RTP má řadu dalších výhod, které nejsou pro potřeby této práce směrodatné.

#### 1.5.4 Protokol SAP (Session Announcement Protocol)

Protokol SAP pracuje na UDP a je definován standardem RFC 2974. Protokol funguje na základě oznamování informací o dostupných multicastových adresách a portech v síti, na kterých probíhá vysílání multimediálního obsahu. Činnost oznamování provádí oznamovatel. Ten pravidelně odesílá paket do sítě na konkrétní multicastovou adresu a port, přičemž výchozími hodnotami jsou, dle standardu RFC, IP 224.2.127.254 a port 9875. SAP protokol nedisponuje žádným mechanismem rendez-vous, což znamená, že oznamovatel nemá přehled o aktuálním stavu diváků. Smyslem těchto oznámení je pouze poskytnout aktivním zařízením v síti informaci o tom, kde (resp. na jaké IP) mohou uživatelé těchto zařízení sledovat multimediální obsah. Níže je uveden příklad distribuce několika desítek oznámení protokolem SAP. V rámci příkladu je použit jako zdroj oznamování nástroj SAP server a přijímač oznámení přehrávač VLC.

No.	Time	Source	Destination	Protocol	Length	Info
5679	28.122119606	10.10.3.55	224.2.127.254	SAP/SDP	336	Announcement (v1)
5681	28.193685441	10.10.3.55	224.2.127.254	SAP/SDP	328	Announcement (v1)
5683	28.265214222	10.10.3.55	224.2.127.254	SAP/SDP	328	Announcement (v1)
5689	28.336092402	10.10.3.55	224.2.127.254	SAP/SDP	332	Announcement (v1)
5692	28.406937590	10.10.3.55	224.2.127.254	SAP/SDP	330	Announcement (v1)
5694	28.477654474	10.10.3.55	224.2.127.254	SAP/SDP	332	Announcement (v1)
5976	28.548812063	10.10.3.55	224.2.127.254	SAP/SDP	331	Announcement (v1)
5978	28.619932850	10.10.3.55	224.2.127.254	SAP/SDP	335	Announcement (v1)
6048	28.691232844	10.10.3.55	224.2.127.254	SAP/SDP	334	Announcement (v1)
6051	28.763082360	10.10.3.55	224.2.127.254	SAP/SDP	324	Announcement (v1)
6056	28.834130145	10.10.3.55	224.2.127.254	SAP/SDP	326	Announcement (v1)
6060	28.905923158	10.10.3.55	224.2.127.254	SAP/SDP	332	Announcement (v1)
6063	28.976860725	10.10.3.55	224.2.127.254	SAP/SDP	331	Announcement (v1)
6067	29.047674217	10.10.3.55	224.2.127.254	SAP/SDP	329	Announcement (v1)
6069	29.119143014	10.10.3.55	224.2.127.254	SAP/SDP	329	Announcement (v1)
6072	29.190642678	10.10.3.55	224.2.127.254	SAP/SDP	333	Announcement (v1)

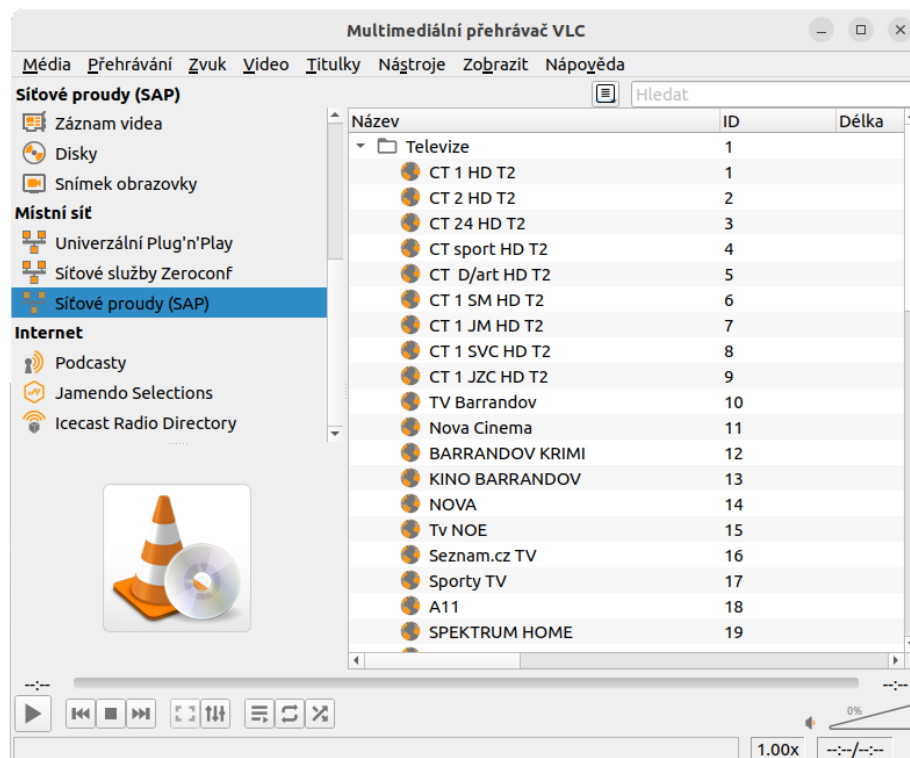
```

▶ Frame 6051: 324 bytes on wire (2592 bits), 324 bytes captured (2592 bits) on interface ens33, id 0
▶ Ethernet II, Src: [redacted], Dst: IPv4mcast_02:7f:fe ([redacted]:7f:fe)
▶ Internet Protocol Version 4, Src: 10.10.3.55, Dst: 224.2.127.254
▶ User Datagram Protocol, Src Port: 40463, Dst Port: 9875
▶ Session Announcement Protocol
▼ Session Description Protocol
  Session Description Protocol Version (v): 0
  ▶ Owner/Creator, Session Id (o): spolecnost 16937 1 IN IP4 localhost
  Session Name (s): NOVA
  URI of Description (u): http://www.videolan.org
  ▶ Connection Information (c): IN IP4 239.0.1.24/255
  ▶ Time Description, active time (t): 0 0
  ▶ Session Attribute (a): tool:miniSAPserver 0.3.6
  ▶ Session Attribute (a): x-plgroup:Televize
  ▶ Session Attribute (a): type:broadcast
  ▶ Session Attribute (a): charset:UTF-8
  ▶ Media Description, name and address (m): video 5004 RTP/AVP 33
  ▶ Media Attribute (a): rtmap:33 MP2T/90000
  ▶ Media Attribute (a): rtcp:5005

```

Obrázek č. 6: Snímek z aplikace Wireshark – protokol SAP  
(Zdroj: Vlastní zpracování)

Obrázek č. 6 obsahuje zachycenou odchozí komunikaci ze zařízení, na kterém je spuštěn již zmíněný nástroj SAP server – viz první zmínka *Session Attribute*. Ten odesílá postupně všechny definované sady informací v jednotlivých datagramech. Z toho plyne, že každý zadaný datový proud v konfiguračním souboru tohoto serveru odesílá v jednom datagramu. V obrázku 6 je uveden příklad s TV Nova – konkrétně *Session Names (s): NOVA*, kdy je dále uvedena multicastová IP adresa 239.0.1.24 a také port 5004, na kterém je možné přijímat toto vysílání včetně uvedeného protokolu RTP v sekci *Media Description*. Druhá hodnota *Session Attribute* obsahuje parametr *x-plgroup* s hodnotou *Televize*. To je instrukce pro přijímač, který na základě definice tohoto parametru by měl toto vysílání zařadit pod stejnojmennou skupinu s názvem Televize. Samotné oznámení nutně neznamena, že je vysílání skutečně dostupné a funkční. Jedná se o to, že vysílání datového proudu zajišťuje jiná služba a oznamování o datových proudech rovněž poskytuje jiná služba, resp. SAP server. Jde tedy o dvě nezávislé služby, které v komplexním řešení vytváří funkční celek. Více tuto problematiku vysvětluje, včetně bližších souvislostí a programových kódů, právě tato práce.



Obrázek č. 7: Snímek z aplikace VLC – protokol SAP  
(Zdroj: Vlastní zpracování)

Z obrázku č. 7 je patrné, že přijímač, v tomto případě přehrávač VLC, správně interpretoval údaje v datagramech a vyobrazil rozříděné datové proudy na TV programy a rádiové programy.

Vzhledem ke způsobu fungování SAP serveru, jak je zřejmé z obrázku č. 6, nejsou datové proudy řazeny žádným kritériem. Tak jak je v cyklu SAP server vysílá, tak takhle jsou od okamžiku přijímání přehrávačem VLC řazeny.

### **1.5.5 Protokol IGMP (Internet Group Management Protocol)**

Protokol IGMP je v překladu internetový protokol se skupinovým adresováním v rámci IPv4. Postupem času prošel tento protokol vývojem a v současné době jsou známé celkem tři verze. První verze IGMPv1 pochází ze srpna roku 1989 a je vymezena v dokumentu RFC 1112. V listopadu 1997 vznikla aktualizace verze jedna, což přineslo IGMPv2. Zmíněná aktualizace s sebou přinesla i nový dokument RFC 2236. Stejně jako v předchozích případech vešla v říjnu 2002 v platnost nová verze 3, tentokrát definovaná dokumentem RFC 3376.

Vzhledem k značnému stáří všech výše uvedených verzí, je dnes zcela běžné, že pokročilejší aktivní síťové prvky jsou vybaveny plnou podporou nejnovější verze IGMPv3.

Hlavním principem protokolu IGMP je správa členství zařízení ve skupinách v rámci vícesměrového neboli multicastového vysílání. Skupina je v tomto kontextu vytvářena dynamicky, v závislosti na použité vícesměrové adrese. Začne-li uživatel pomocí svého síťového zařízení odesílat datový proud na libovolnou vícesměrovou IP adresu, nejbližší přepínač v síti zaregistruje tento typ provozu a vytvoří, na základě žádosti uživatele, jednočlennou skupinu. Vychází se z předpokladu, že daný přepínač podporuje protokol IGMP a nemá nastavena žádná další omezení bránící v používání vícesměrové komunikace. V tento okamžik probíhá komunikace pouze mezi odesílatelem a přepínačem. Pokud odesílatel využije např. protokolu SAP a začne všem aktivním zařízením v téže síti oznamovat dostupnost svého datového proudu, libovolný příjemce může tento proud přijímat na základě žádosti (IGMP Join Group), kterou pošle protokolem IGMP přepínači. Na základě tohoto požadavku přepínač přiřadí do požadované skupiny daného uživatele a následně začne přepínač odesílat datový proud novému členu této skupiny. Pokud uživatel přestane mít o přijímání datového proudu

zájem, odešle se požadavek přepínači IGMP Leave Group. To zajistí, že přepínač tohoto příjemce vyřadí z vícesměrové skupiny. Jedná se o efektivní způsob, jak nezahltit síť zbytečnou komunikací.

Source	Destination	Protocol	Length	Info
10.17.8.6	224.0.0.22	IGMPv3	62	Membership Report / Leave group 239.0.1.1 / Join group 239.0.1.2 for any sources

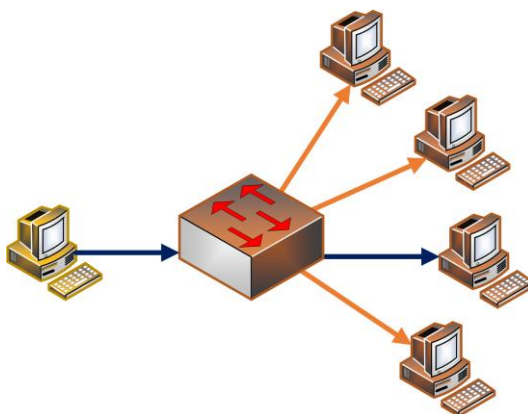
Obrázek č. 8: Snímek z aplikace Wireshark – IGMP Leave, Join group  
(Zdroj: Vlastní zpracování)

## 1.6 Přenosové metody

Přenosové metody v rámci LAN lze rozdělit do tří kategorií: *unicast*, *multicast*, *broadcast*. Jednotlivé kategorie se liší počtem cílových uzlů, ke kterým jsou pakety směrovány (3).

### 1.6.1 Unicast

V případě unicastu jsou pakety směrovány z jednoho konkrétního místa v síti do druhého známého místa v síti. Je zapotřebí, aby vysílající strana jednoznačně definovala cílové místo v síti pomocí IP adresy. Doručení paketů od zdroje k cíli zajišťuje síť (3).



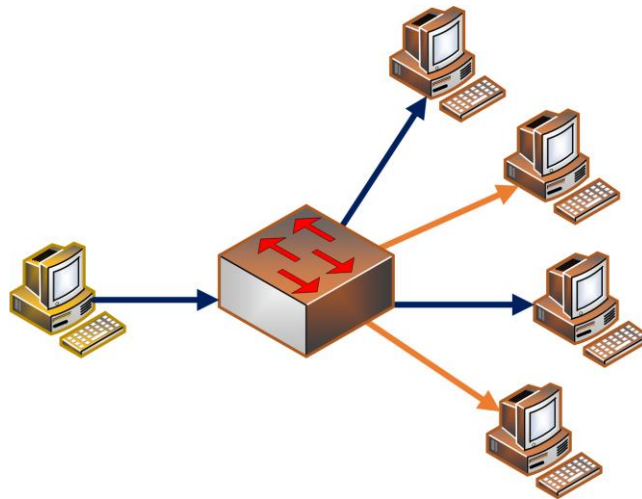
Obrázek č. 9: Schématické znázornění přenosové metody – unicast  
(Zdroj: Vlastní zpracování)

Oranžové šipky reprezentují pouze připojení k přepínači. Tmavě modrými šipkami je označeno jednak připojení, tak i cíl, do kterého směřuje datový proud.

### 1.6.2 Multicast

Tato přenosová metoda má jedno zdrojové místo v síti a určenou skupinu příjemců, kdy počet příjemců ve skupině je proměnlivý. Jak bylo uvedeno u protokolu IGMP, každý uzel sítě může požádat o členství v příslušné skupině anebo požádat o zrušení členství.

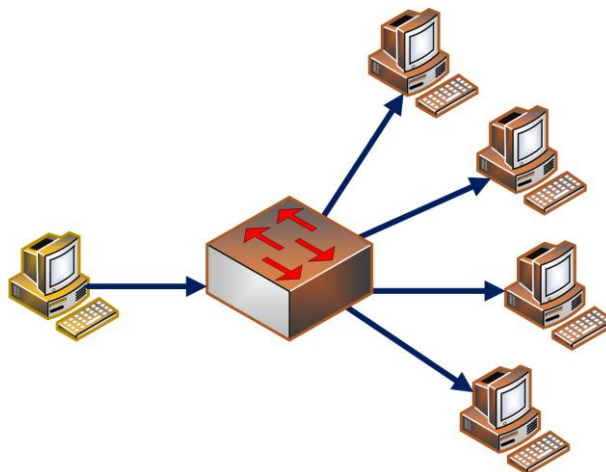
To platí v případě adresace IPv4. Adresace IPv6 disponuje protokolem MLD (Multicast Listener Discovery), který funguje analogicky jako IGMP. Také již bylo uvedeno, že je důležité, aby aktivní síťové prvky podporovaly protokol IGMP. Pokud tomu tak nebude, vícesměrový provoz se bude v síti šířit jako broadcast. Multicast využívá vyhrazené obory IP adres.



**Obrázek č. 10: Schématické znázornění přenosové metody – multicast**  
(Zdroj: Vlastní zpracování)

### 1.6.3 Broadcast

V tomto případě se paket jednoho uzlu rozpošle všem aktivním uzlům sítě. Tuto činnost provádí sama síť. Dle jednoduchého příkladu, vyobrazeného níže, je za rozposlání jednoho paketu všem uzlům odpovědný přepínač.



**Obrázek č. 11: Schématické znázornění přenosové metody – broadcast**  
(Zdroj: Vlastní zpracování)



## 1.7 Operační systémy

Operační systémy jsou komplexní softwarové platformy, které koordinují interakce mezi hardwarem, softwarem a uživatelem. Jednou z jejich základních funkcí je alokace systémových prostředků, mezi které náleží např. procesor, paměť a diskové úložiště. To zajišťuje, že různé aplikace a procesy mohou efektivně spolupracovat bez vzájemného ovlivňování. Dále zprostředkovávají vstupy a výstupy různých zařízení, což umožňuje uživatelům ovládat počítač prostřednictvím grafického uživatelského rozhraní (GUI) nebo příkazového řádku (CLI). Z hlediska služeb operační systémy poskytují flexibilní nástroje, jako jsou správci souborů, síťové služby včetně firewallů a zásad zabezpečení, které zajišťují integritu a ochranu dat. Tyto systémy také spravují práva a přístupy různých uživatelských účtů. Vývoj operačních systémů neustále pokračuje s cílem zlepšit uživatelskou přívětivost, efektivitu a bezpečnostní opatření. (7)

### 1.7.1 Linux

V roce 1991 vznikl Linux jako osobní projekt Linuse Torvaldse, finského vysokoškolského studenta. Pro velký zájem se Linux v celém světě brzy začal využívat a vyvíjet jádro<sup>3</sup>. V roce 1994 byla vydána verze jádra 1.0. V době psaní této práce existuje nejnovější stabilní verze linuxového jádra s označením 6.8.7. (8, 9)

Díky svým unixovým předkům Linux zdědil mnoho funkcí. Historie Unixu se datuje od roku 1969. Tehdy Unix vznikl jako výzkumný projekt v AT&T Bell Labs. O 7 let později byl Unix poskytnut univerzitám zdarma. Stále však Unixový kód obsahoval část licencovaného kódu patřící AT&T. Unix se čím dál častěji používal v komerčním prostředí a tím se rychleji zvyšovala cena licencí. Berkeley Unix si dal za cíl odstranit z tohoto kódu licencovanou část společnosti AT&T a po časově náročném procesu se mu podařilo vytvořit verzi známou jako 4.4BSD-Lite. (8)

Systém Unix nemá shodný kód s Linuxem. Linux je z hlediska funkčnosti někde mezi provedením AT&T a v provedení BSD. Linux obsahuje jen samotné jádro. K tomuto jádru je nezbytné doplnit demony, dílčí aplikace, rozhraní, utility a další software za účelem vytvoření použitelného a úplného operačního systému. V tomto kontextu se takový úplný celek nazývá distribuce. Všechny Linuxové distribuce vychází z téhož jádra, avšak se liší v doplňujícím softwarovém vybavení. (8)

V současné době se mezi nejpoužívanější Linuxové distribuce řadí Debian, Ubuntu, Mint, CentOS, RHEL, Fedora, openSUSE. Existuje i několik dalších desítek distribucí. (10)

## **1.8 Programovací jazyk Python**

Python je výkonný programovací jazyk, který mimo jiné obsahuje objektové orientované programování. Dále je Python vybaven efektivními vysokoúrovňovými datovými strukturami. Python je kompatibilní s většinou platforem. Interpreter tohoto jazyka je k dispozici zdarma ke stažení včetně široké palety knihoven, modulů, nástrojů a dalších programů. (11)

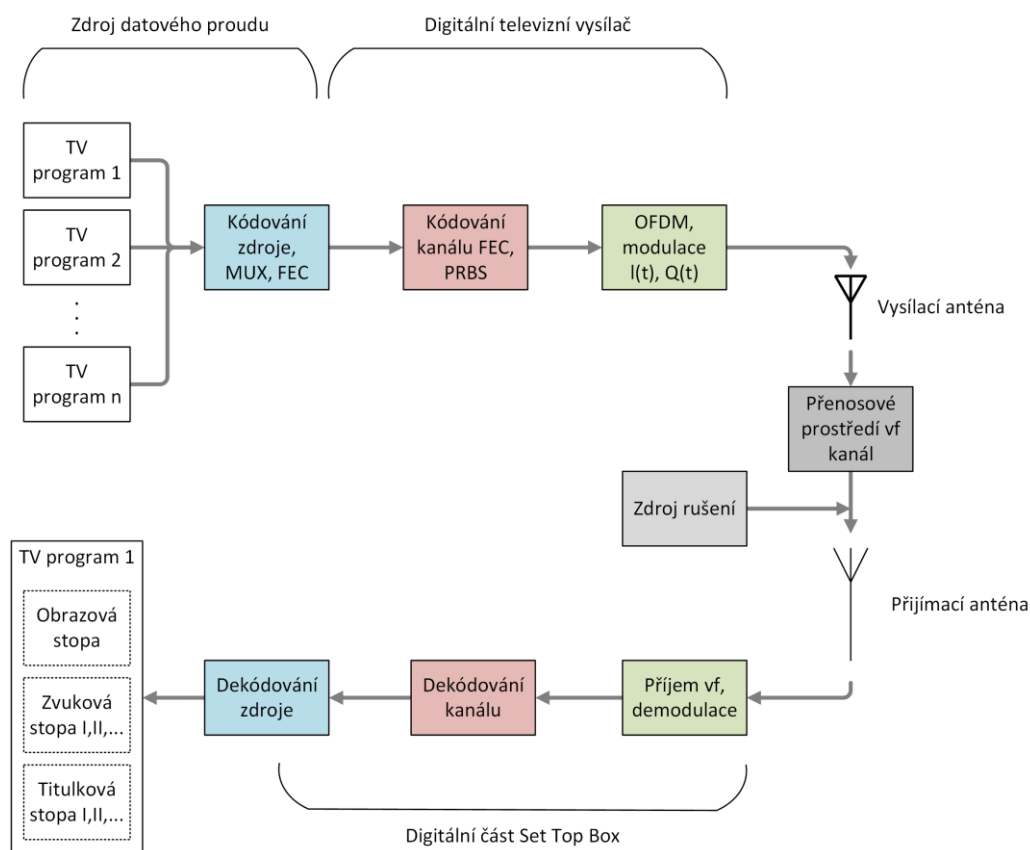
## **1.9 DVB (Digital Video Broadcasting)**

DVB je mezinárodní konsorcium vytvořené televizními společnostmi, výrobci, síťovými operátory, vývojáři softwaru, regulačními organizacemi a dalšími zástupci. Na přípravě DVB se podílelo celkem 260 členů z 35 států celého světa. V roce 1993 došlo k založení tohoto konsorcia, což bylo základním kamenem pro vytvoření norem, dle specifikací DVB v oblasti digitální televize nejen v Evropě, ale i ve světě. Specifikace DVB vydává Evropský telekomunikační institut ETSI ve spolupráci s Evropskou unií pro rozhlasové a televizní vysílání EBU. (12)

Třemi klíčovými normami se staly DVB-S pro satelitní síť, DVB-C pro kabelové síť a DVB-T pro pozemní síť. Kromě nich byla zapotřebí celá řada podpůrných norem pokrývajících oblasti, jako jsou servisní informace DVB-SI, titulky DVB-SUB a rozhraní DVB-ASI. (13)

### **1.9.1 DVB-T (Digital Video Broadcasting Terrestrial)**

Mezinárodní standard DVB-T popisuje digitální televizi první generace šířenou prostřednictvím pozemního vysílání. DVB-T bylo navrženo pro nahrazení starších analogových televizních systémů. Analogové systémy měly hned několik nevýhod. První nevýhodou bylo, že jeden TV program vyžadoval pro svůj přenos velkou šířku pásma. Druhou nevýhodou byla vysoká energetická náročnost vysílačů. Díky DVB-T je možné přenášet více TV a rozhlasových programů současně s nižší energetickou náročností.



**Obrázek č. 12: Schematické znázornění obecného modelu vysílání a příjmu DVB-T**

(Zdroj: Vlastní zpracování dle: 14)

Každý TV program obvykle sestává z minimálně dvou složek, kterými jsou audio a video stopy. Často se k těmto dvěma složkám přidává třetí, kterou jsou informace programové nabídky příslušného TV programu. Několik takových TV programů se slučuje do multiplexu za účasti mechanismu protichybové ochrany FEC (Forward Error Correction). Dále se realizuje přenos specifickými mechanismy a přenosovým prostředím od vysílače až k přijímači DVB-T. Za příznivých podmínek přijímač úspěšně dekóduje anténou přijatý signál a tím poskytne uživateli, resp. divákovi, k dispozici všechny TV programy, jež byly na začátku multiplexovány. Detailní způsob fungování DVB-T není zapotřebí uvádět, protože tvoří relevantní součást práce. (14)

První experimentální vysílání dle standardu DVB-T proběhlo 12. 5. 2000 v Praze, kdy provozovatelem byly České Radiokomunikace a.s. Ostrý provoz jednoho multiplexu nastal v říjnu 2005. Až o rok později se začaly spouštět další multiplexy. (12)

## 1.9.2 DVB-T2 (Digital Video Broadcasting Terrestrial 2)

DVB-T2 je pokročilejší verze DVB-T, která nabízí lepší efektivitu přenosu a podporu vyšších rozlišení obrazu a zvuku. DVB-T2 je postupně přijímán v mnoha zemích jako standard pro digitální pozemní vysílání. (13)

DVB-T2 se poprvé testovalo již v roce 2010 v Praze na Žižkovském vysílači. (15) První oficiální spuštění přechodové sítě DVB-T2 na území ČR bylo zahájeno v odpoledních hodinách 1. března 2017. (16) Přejít na standard DVB-T2 v České republice byl oficiálně dokončen 30. září 2020. Tento den bylo ukončeno vysílání v původním DVB-T formátu z vysílačů Zlín – Tlustá hora a Valašské Klobouky – Ploštiny. Všechny regiony ČR tak přešly na modernější DVB-T2. (17)

Dle internetu bylo zjištěno, že na území České republiky v oblasti Brna a blízkého okolí jsou k dispozici čtyři níže uvedené celoplošné multiplexy.

**Tabulka č. 1: Přehled celoplošných multiplexů, frekvencí a vysílačů DVB-T2 pro Brno a okolí**  
(Zdroj: Vlastní zpracování dle: 18, 19, 20, 21, 22)

Multiplex 21	#	Vysílač	Kanál	Frekvence	Polarizace
ČT1, ČT2, atd...	1	Brno – Kojál	26	514 MHz	H
	2	Brno – Hády	26	514 MHz	H
	3	Brno – Barvičova	26	514 MHz	V
<b>Multiplex 22</b>	1	Brno – Kojál	40	626 MHz	H
Prima, Cool, Love, Zoom, Max, Nova, Cinema,...	2	Brno – Hády	40	626 MHz	H
	3	Brno – Barvičova	40	626 MHz	V
<b>Multiplex 23</b>	1	Brno – Kojál	33	570 MHz	H
Nova, Cinema, Seznam, Barrandov,...	2	Brno – Hády	33	570 MHz	H
	3	Brno – Barvičova	33	570 MHz	V
<b>Multiplex 24</b>	1	Brno – Jihlavská	46	674 MHz	H
Nova, Cinema, Nova2, Action, Gold, Relax, ...	2	Brno – Hády	46	674 MHz	H
	3	Brno – Barvičova	46	674 MHz	V

Kromě celoplošných existují i regionální multiplexy, které se vysílají se specifickou nabídkou a pokrytím. Na území Brna a okolí jsou, dle webu Českého telekomunikačního úřadu, k dispozici níže uvedené regionální multiplexy.

**Tabulka č. 2: Přehled regionálních multiplexů, frekvencí a vysílačů DVB-T2 pro Brno a okolí**  
(Zdroj: Vlastní zpracování dle: 23)

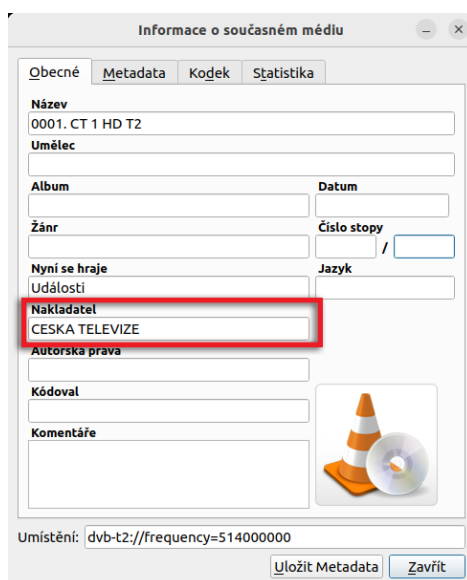
Označení multiplexu	Vysílač	Kanál	Frekvence	Polarizace
Regionální síť 7	Brno – Jihlavská	37	602 MHz	H
Regionální síť 15	Adamov	32	562 MHz	H

Každý multiplex spravuje jiný poskytovatel. Za využití přehrávače VLC byl proveden průzkum těchto poskytovatelů vždy u zástupce jednoho TV programu z každého multiplexu.

**Tabulka č. 3: Přehled poskytovatelů jednotlivých multiplexů sítě DVB-T2 v oblasti Brna**  
(Zdroj: Vlastní zpracování)

Označení multiplexu	Frekvence (MHz)	Název provozovatele
MUX21	514	Česká televize
MUX22	626	České radiokomunikace
MUX23	570	CDG (Czech Digital Group)
MUX24	674	DB (Digital Broadcasting)

Příklad zdroje informací o názvech poskytovatelů z tabulky č. 3 je objasněn níže.



**Obrázek č. 13: Snímek z aplikace VLC – poskytovatel MUX21**  
(Zdroj: Vlastní zpracování)

## 1.10 Vývojový diagram

Vývojový diagram slouží pro grafické znázornění algoritmů. Díky tomuto znázornění může širší okolí uživatelů lépe porozumět popisované problematice. Na obrázku č. 14 jsou uvedené některé obrazce používané ve vývojových diagramech.



**Obrázek č. 14: Některé grafické obrazce vývojového diagramu**  
(Zdroj: Vlastní zpracování dle: 24)

Jednotlivé obrazce se spojují tzv. spojnicemi. Spojnice je pravoúhlá a orientovaná čára, což znamená, že má směr a zakresluje se jako šipka. Při konstrukci vývojového diagramu se dodržuje pravidlo shora dolů, a proto není zapotřebí spojnice doplňovat o šipku. V případě porušení zmíněného pravidla, je nutné šipku zakreslit. Odkaz na stránce se využívá jako spojka mezi částmi diagramu. Toho se například využívá v situacích, kdy je zapotřebí, aby diagram pokračoval například na další straně. Kosočtverec se využívá při rozhodování. Obvykle symbolizuje podmínku jako v programovacím jazyce. Do tohoto obrazce vstupuje jedna spojnice, ale vystupují alespoň dvě; v tomto místě dochází k vyhodnocení podmínky, zda platí či nikoli a dále dochází k větvení. Je vhodné odpovídajícím způsobem popsat jednotlivé výstupní spojnice, aby bylo zřejmé, která větev vyjadřuje danou situaci. Diagram musí být ucelený a všechny spojnice z jednotlivých obrazců musí mít nějaký cíl. (24)

### 1.11 SWOT analýza

SWOT analýza je univerzální metoda pro identifikaci a hodnocení faktorů, které externě i interně ovlivňují podnikání společnosti. Název SWOT je akronym z anglických slov pro silné stránky (Strengths), slabé stránky (Weaknesses), příležitosti (Opportunities) a hrozby (Threats). Cílem SWOT analýzy je nejen identifikace problémů, ale také navržení možných řešení a určení strategických směrů pro další rozvoj firmy. (25)



Obrázek č. 15: Vyobrazení struktury SWOT analýzy  
(Zdroj: Vlastní zpracování dle: 25)

**Silné stránky** jsou vlastnosti, díky kterým má firma konkurenční výhodu a které jí pomáhají dosáhnout jejich vytyčených cílů. Tyto vlastnosti mohou zahrnovat vysoce kvalifikovaný personál, inovativní technologie, silné zákaznické vztahy, nebo významný tržní podíl. Důležité otázky pro identifikaci silných stránek mohou být například: Co firma dělá dobře? Jaké unikátní zdroje má k dispozici? Jak je firma vnímána svými zákazníky?

**Slabé stránky** představují oblasti, kde firma zaostává nebo čelí výzvám, které mohou omezovat její úspěch nebo růst. Mohou to být například nesprávně fungující procesy, zastaralé technologie, finanční omezení, nebo slabá pozice na trhu. Při analýze slabých stránek je užitečné se ptát: Co potřebujeme zlepšit? Jaké zdroje nám chybí? Kde jsme méně konkurenceschopní?

**Příležitosti** se týkají externích faktorů nebo trendů, které společnost může využít ke svému prospěchu. Tyto příležitosti mohou zahrnovat technologický pokrok, regulační změny, nové tržní segmenty, nebo pozitivní ekonomické podmínky. Relevantní otázky pro identifikaci příležitostí zahrnují: Jaké nové technologie můžeme využít? Jaké nové potřeby zákazníků můžeme uspokojit? Jaké trhy jsou nyní dostupné pro expanzi?

**Hrozby** jsou externí výzvy, které by mohly negativně ovlivnit výkonnost nebo pozici firmy. Tyto hrozby mohou zahrnovat rostoucí konkurenci, negativní ekonomické trendy, změny v zákonech, nebo technologické změny, které mají vliv na současné nabídky firmy. Při hodnocení hrozeb je dobré si uvědomit: Jaké ekonomické nebo politické změny nás mohou ovlivnit? Co dělá konkurence lépe? Jaké technologie by mohly ohrozit naše produkty nebo služby? (25)

### **1.11.1 IFE, EFE matice**

IFE (Internal Factor Evaluation) a EFE (External Factor Evaluation) matice jsou nástroje, které se používají v rámci SWOT analýzy pro podrobnější kvantitativní hodnocení vnitřních a vnějších faktorů, které ovlivňují organizaci. Tyto matice poskytují strukturovaný způsob, jak analyzovat a porovnávat různé faktory na základě jejich důležitosti a vlivu na výkonnost organizace. (26)

#### **IFE Matice (Internal Factor Evaluation)**

IFE matice je zaměřena na vnitřní faktory, což jsou silné a slabé stránky společnosti. V této matici jsou jednotlivé faktory ohodnoceny podle toho, jaký mají vliv na schopnost organizace dosáhnout svých strategických cílů. Každý faktor je hodnocen váhou, která určuje jeho relativní důležitost, a skóre, které vyjadřuje, jak dobře organizace s faktorem nakládá. Výsledné skóre pomáhá určit, jaké vnitřní faktory jsou klíčové pro úspěch organizace a které vyžadují zlepšení. (26)

#### **EFE Matice (External Factor Evaluation)**

EFE matice se zaměřuje na vnější faktory, tedy na příležitosti a hrozby. Podobně jako IFE, i v EFE matici jsou faktory hodnoceny podle váhy a vlivu. Váha udává, jaký je relativní význam daného faktoru pro organizaci, zatímco skóre vlivu ukazuje, jak efektivně organizace využívá příležitosti nebo čelí hrozbám. EFE matice tedy poskytuje přehled o vnějším prostředí a pomáhá identifikovat klíčové externí faktory, které mohou ovlivnit strategii a výkonnost organizace. (26)

V obou maticích jsou uvedeny subjektivně stanovené hodnoty váhy a vlivu. Rozsah vlivu je vymezen od jedné do čtyř, kdy číslo 1 vyjadřuje nejnižší důležitost, zatímco číslo 4 reprezentuje nejvyšší důležitost. Váha se stanovuje pro každou matici (IFE, EFE) souhrnně tak, aby celkový součet vah činil 1. Vážené skóre se vypočítává jako součin váhy a vlivu.



## **2 ANALÝZA PROBLÉMU A SOUČASNÉ SITUACE**

### **2.1 O společnosti**

Tato kapitola pojednává o společnosti, pro kterou je tato práce realizována. Společnost je popsána nejprve obecně. Následuje stručná podkapitola věnována historii vzniku, kdy tuto kapitolu uzavírá vzhled do současných obchodních záměrů společnosti.

#### **2.1.1 Popis**

Společnost ABC, s.r.o. je anonymizovaný název existující společnosti. Hlavní náplní firmy jsou elektromontážní práce, výroba rozváděčů nízkého napětí a dodávky elektrických a elektronických zařízení. Dále společnost vyvíjí činnost v oblasti strojírenské výroby, kde se specializuje na výrobu prvků pro technologická zařízení mísení krmných směsí. Podstatnou pracovní náplní firmy je i projekční činnost.

Díky využívání moderních technologií a speciální techniky, jakož i znalosti řešení daného problému a schopnosti rychle reagovat, se úspěšně společnost prosazuje na domácí půdě. Firma dále poskytuje veřejnou službu internetového připojení ve vybraných lokalitách. Společně s internetovým připojením nabízí tato firma také VoIP telefonii. K poskytovaným službám firma svým klientům nabízí zákaznickou podporu.

#### **2.1.2 Historie**

Společnost byla založena v roce 1991 jako sdružení tří fyzických osob. V roce 1993 se firma transformovala na společnost s ručením omezeným a v dnešní době již zaměstnává 24 zaměstnanců.

#### **2.1.3 Současnost**

V současné době oslovil společnost developer se záměrem vybudovat novou městskou výstavbu. Společnost ABC, s.r.o. se má stát poskytovatelem internetového a televizního připojení ve všech nově připravovaných bytech a domech. Jedná se o přibližně 350 nových přípojek, přičemž v budoucnu je naplánovaná další etapa výstavby nových bytů a domů. Celkový počet přípojek může být v budoucnu orientačně až 500. Záleží na finančních možnostech developera. Z tohoto důvodu je pro společnost nezbytné zkvalitnit své služby, aby mohl být každý zákazník plně spokojen. Zejména se jedná o zavedení nové služby IPTV.

## **2.2 Komunikační infrastruktura společnosti**

Společnost ABC, s.r.o. provozuje kabelovou i bezdrátovou komunikační infrastrukturu ve 12 obcích. Stávající infrastruktura je stále udržovaná a nadále rozvíjená, díky čemuž disponuje optimálními přenosovými parametry. Na základě takto vysoké úrovně infrastruktury je společnost schopna nabízet svým zákazníkům stabilní a spolehlivé internetové připojení. Zákazníky společnosti jsou jak fyzické, tak i právnické osoby. Kabelová infrastruktura je pouze optická a převážně využívá technologii GPON.

## **2.3 Hardwarové vybavení**

V současné době firma provozuje na jednom fyzickém serveru virtualizační technologii Hyper-V. Pomocí této technologie jsou virtualizovány dva operační systémy Windows Server 2022, na kterých běží nezbytný software nutný pro každodenní firemní činnosti.

Na střeše zmíněného domu se nachází dvě směrové antény, přičemž obě získávají DVB-T2 signál čtyř multiplexů z brněnského vysílače umístěného na ulici Barvičova. Za oběma anténami se nachází dva specializované filtry pro odstranění elektromagnetického šumu pocházejícího z mobilních sítí LTE a 5G. Následně je signál mírně zesílen. Dále je vedení signálu připraveno na rozbočení pro přijímací zařízení, kterými v tomto případě budou tunery DVB-T2.

Z této podkapitoly plyne, že firma již podnikla první kroky nezbytné pro spuštění nové služby, díky které zákazníci získají možnost sledovat televizní vysílání pomocí internetového připojení. Jedná se o tzv. IPTV.

## **2.4 Dostupná existující řešení**

V této kapitole jsou popsána nalezená existující řešení pro poskytování IPTV.

### **2.4.1 TSDuck**

TSDuck je open-source nástrojová sada pro práci s MPEG transportními streamy, používaná v digitálním televizním vysílání pro testování, monitorování, integraci a ladění. Umožňuje získávání transportních streamů z DVB, ATSC, ISDB a IP multicastu. Dále je tento software schopen analyzovat stream z hlediska jeho vlastností, což znamená audia i videa, monitorovat jej a reportovat jeho stav. TSDuck je vybaven funkcí transformovat nebo vkládat obsah za běhu a také modifikovat, odstraňovat, přejmenovávat TV programy. Podporuje práci jak s živými, tak i s off-line transportními

streamy, tzn. např. příjem z DVB-T nebo DVB-C, atd... Tento software je vyvíjen v C++ s modulární architekturou. TSDuck je snadno rozšiřitelný prostřednictvím pluginů a je možné ho ovládat pomocí CLI. TSDuck je k dispozici pro platformy Windows, Fedora, Ubuntu, RedHat, Debian a Raspbian (Raspberry Pi). (27)

### **2.4.2 MuMuDVB**

MuMuDVB (Multi Multicast DVB) je program pro streamování televize přes IP síť. Tento software dokáže přerozdělit stream z DVB zdroje (digitální satelitní, pozemní a kabelové televize nebo ATSC) do sítě, v multicastu nebo v HTTP unicastu. Jeho hlavní funkcí je zpracovat celý transpondér a umístit každý kanál do různé multicastové skupiny s jednoduchou konfigurací a vylepšenou kompatibilitou se set-top boxy. MuMuDVB má nízkou spotřebu paměti a CPU a je schopen běžet na vestavěných platformách. Servery s více než 20 DVB kartami jsou úspěšně používány s MuMuDVB s nepřetržitou dobou provozu více než dva roky. Tento software funguje pod Linuxem a využívá Linux DVB API. Jedná se o svobodný software, který je vydán pod licenčními podmínkami GPLv2. Tento software disponuje možností vzdáleného monitoringu pomocí webového rozhraní. (28)

### **2.4.3 DVBlast**

DVBlast je minimalistický a výkonný software pro demultiplexování a streamování MPEG-2 a MPEG-4 transportních streamů (TS). Je vyvíjen organizací VideoLAN. Primárním cílem DVBlastu je efektivně zpracovávat a streamovat digitální televizní signály s nízkou spotřebou systémových prostředků, což z něj činí ideální řešení pro hardware s omezenými kapacitami. DVBlast podporuje příjem různých druhů DVB standardů, včetně DVB-S, DVB-S2, DVB-T, DVB-T2 a DVB-C. (29)

Software je schopen pracovat s více DVB kartami současně, což umožňuje simultánní zpracování několika streamů. Uživatelé mohou snadno konfigurovat DVBlast pro streamování jednoho nebo více kanálů pomocí unicastu nebo multicastu do IP sítě, což usnadňuje distribuci obsahu v lokálních sítích nebo přes internet. Díky jeho jednoduchosti a efektivitě je DVBlast oblíbený mezi profesionály v oblasti broadcastingu a mezi technicky zdatnými uživateli, kteří potřebují stabilní a spolehlivé řešení pro digitální televizní vysílání. (29)

DVBlast je distribuován jako svobodný software pod licencí GNU General Public License (GPL), což znamená, že uživatelé mohou software volně používat, modifikovat a šířit. Jeho konfigurace probíhá prostřednictvím textových konfiguračních souborů, což nabízí pokročilé uživatelské nastavení, ale zároveň vyžaduje určitou úroveň technických znalostí. Vývojáři DVBlastu se zaměřují na efektivitu a výkon, což činí z DVBlastu spolehlivou volbu pro profesionální aplikace v oblasti digitálního vysílání. (29)

#### **2.4.4 Souhrn poznatků**

Z nabízených softwarů nejlépe vyhovuje DVBlast, protože TSDuck má pro účely navrhovaného použití zbytečně komplikované rozhraní a má značné požadavky na hardware. Problém s neintuitivním rozhraním se vyskytuje i u MuMuDVB, ale tento software pracuje efektivněji a díky tomu nespotřebovává nadbytek HW prostředků. DVBlast disponuje vysokým výkonem a nízkou náročností na procesor a operační paměti. Tato významná výhoda s sebou nese nedostatky v podobě nízké flexibility a omezených funkcionalit. Nespornou výhodou DVBlastu je jednoduchá konfigurace a použití. Z tohoto důvodu je pro potřeby této práce nástroj DVBlast zcela vyhovující.

#### **2.5 Požadavky na řešení**

V podkapitole 2.2 bylo zjištěno, že současná infrastruktura společnosti je vhodná pro provozování IPTV. Dále byla uvedena existující softwarová řešení, která se v praxi používají pro IPTV. Porovnáním výhod a nevýhod jednotlivých SW řešení bylo vybráno pouze jedno odpovídající požadavkům pro navrhovanou aplikaci. V následujícím odstavci jsou uvedeny požadavky na nové řešení.

Management společnosti požaduje zajištění dostupnosti základní programové nabídky, jinak dostupné v běžné síti v rámci celoplošného terestriálního vysílání DVB-T2. Smyslem je zjednodušení příjmu televizních programů klientům, kteří si tak nemusí instalovat vlastní domovní televizní rozvody. Druhým požadavkem je nabídnout možnost sledování TV také v osobním počítači v domácnosti. Třetí požadavek se týká přidání regionálního informačního kanálu nabízejícího divákům přehled nadcházejících událostí v různých obcích. Případně tento prostor mohou využívat i obecní samosprávy pro sdělování zpráv z úřední desky. Předposledním požadavkem je zajištění přístupu přijímače k programové nabídce. Díky tomu získají diváci možnost nahlédnout například

na název právě vysílaného pořadu. Posledním požadavkem je zajištění flexibilního ovládání zvoleného řešení zahrnující průběžné udržování aktuální programové nabídky.

## **2.6 SWOT analýza**

SWOT analýza se využívá pro fúzi interních a externích faktorů společnosti. Interní faktory společně tvoří silné a slabé stránky. Externí faktory zahrnují příležitosti a hrozby tvořené externím prostředím společnosti.

### **Silné stránky**

Společnost se může opřít o vysokou úroveň kvalifikace svých zaměstnanců, což je zásadní pro poskytování kvalitních služeb a inovací. Firma disponuje kvalitní komunikační infrastrukturou. Díky ní je možné poskytovat internetové připojení spolehlivě. Dále popisovaná společnost klade důraz na dobrou zákaznickou podporu. Dlouholetá přítomnost a zkušenosti na trhu umožňují společnosti dobře rozumět potřebám svých zákazníků a hledat další potenciál pro svůj růst.

### **Slabé stránky**

Společnost zatím neexpandovala do oblasti televizních služeb, což může být vnímáno jako nevýhoda ve srovnání s konkurenty, kteří již nabízejí televizní služby. Společnosti se v současné době daří udržet si stávající klientelu, navzdory absenci IPTV. Personální kapacity jsou na dostatečné úrovni, aby společnost dostála svým závazkům. Nicméně při využívání potenciálu společnosti by mohl současný počet zaměstnanců omezovat budoucí růst. Finanční omezení může komplikovat nebo zpozdit expanzi do nových lokalit, což je kritické pro udržení konkurenceschopnosti.

### **Příležitosti**

Aktuální obchodní příležitost s developerskou firmou představuje zásadní možnost rozšíření působnosti společnosti do nových lokalit. Jedná se o klíčový faktor pro růst a expanzi společnosti, který má silný potenciál pozitivně ovlivnit její budoucí vývoj. Tato aktuální příležitost současně vytváří prostor pro budoucí spolupráci této společnosti s týmem developerem na dalších projektech. Rozšíření nabídky o IPTV služby může společnosti ABC, s.r.o. pomoci oslovit nové segmenty zákazníků a posílit její tržní pozici. Rozvoj nových služeb může přispět k získání nové klientely, obzvláště v případě nové městské výstavby s 350 přípojkami.

## **Hrozby**

Potenciální legislativní změny mohou představovat riziko pro stávající procesy a budoucí expanzi společnosti ABC, s.r.o. V současné době nejsou plánovány žádné majoritní legislativní změny. Momentálně je finanční situace developera dobrá, a proto se nepředpokládá jeho bankrot nebo nedostatek finančních prostředků na realizaci zmíněného projektu nové městské výstavby. Ostatní poskytovatelé internetového připojení v nové lokalitě zatím neprojevily zájem o zřízení své infrastruktury, a tak je společnost ABC, s.r.o. jediná, která má dohodu s developerem. Dalším konkurentem je jeden celorepublikový operátor, který má v této lokalitě dostupnou bezdrátovou síť LTE. Tato služba je vhodná pouze pro některé domácnosti, protože při větším zatížení může docházet k výraznému poklesu rychlosti a prodloužení doby odezvy všech uživatelů. Hrozba konkurence je momentálně nízká. Poslední hrozbou je nezískání nových klientů z plánovaného projektu. Pro klienty by mohla být nabídka společnosti ABC, s.r.o. nezajímavá či nevýhodná a z tohoto důvodu by se mohli rozhodnout pro konkurenci. Dalším důvodem může být upřednostnění možnosti připojení pomocí bezdrátové technologie LTE mobilního operátora. Oba důvody jsou málo pravděpodobné.

### **2.6.1 Výstup**

Výstup celé SWOT analýzy je uveden pomocí dvou matic IFE (Internal Factor Evaluation) a EFE (External Factor Evaluation).

**Tabulka č. 4: IFE matice (interní faktory SWOT analýzy)**  
(Zdroj: Vlastní zpracování)

Oblasti	Popis	Váha	Vliv (1-4)	Vážené skóre
S	Kvalifikovaní zaměstnanci	0,18	4	0,72
	Kvalitní komunikační infrastruktura	0,16	3	0,48
	Velmi dobrá zákaznická podpora	0,15	2	0,3
	Dobrá orientace na regionálním trhu díky dlouholetým zkušenostem	0,13	3	0,39
	<b>Součet</b>	<b>0,62</b>		<b>1,89</b>
W	Společnost dosud neposkytuje žádné televizní služby	0,22	4	0,88
	Ztráta současné klientely	0,07	1	0,07
	Omezený počet zaměstnanců	0,04	2	0,08
	Hrozící nedostatek finančních prostředků na výstavbu nové infrastruktury v nové lokalitě	0,05	1	0,05
	<b>Součet</b>	<b>0,38</b>		<b>1,08</b>
<b>Celkový součet</b>		<b>1</b>		<b>2,97</b>

Z analýzy IFE matice pro danou společnost vyplývá, že firma má silné stránky v oblasti kvalifikace zaměstnanců, kvality komunikační infrastruktury, zákaznické podpory a regionální orientaci díky dlouholetým zkušenostem. Tyto faktory mají celkové vážené skóre 1,89, což naznačuje, že jsou zásadní pro úspěch prosperující společnosti.

Slabé stránky jako je absence televizních služeb, případná ztráta současné klientely, omezený počet zaměstnanců a málo pravděpodobný nedostatek finančních prostředků pro nové projekty indikují, že by popisovaná společnost rozhodně měla co nejdříve vytvořit a poskytovat službu IPTV. Celkové vážené skóre pro slabé stránky je 1,08, což je relativně nízké skóre ve srovnání se silnými stránkami. Z toho plyne, že tyto slabé stránky mohou významně ovlivnit schopnost firmy ABC, s.r.o. dosáhnout strategických cílů, pokud nebudou řešeny.

Celkový součet váženého skóre pro IFE matici činí 2,97, což značí, že firma má pozitivní hodnocení svých vnitřních faktorů, ale je důležité provést kroky ke zlepšení identifikovaných slabých oblastí. Společnost by měla zvážit strategie pro diverzifikaci svých služeb, zlepšení finančního zdraví a udržení lidských zdrojů s jejich případným budoucím posílením. Společnost tak bude lépe čelit výzvám a bude moci více využívat svých silných stránek.

**Tabulka č. 5: EFE matice (externí faktory SWOT analýzy)**  
(Zdroj: Vlastní zpracování)

Oblasti	Popis	Váha	Vliv (1-4)	Vážené skóre
<b>O</b>	Aktuální obchodní příležitost s developerskou firmou → rozšíření pole působnosti společnosti ABC, s.r.o. o další lokalitu	0,23	4	0,92
	Budoucí spolupráce s developerem	0,18	2	0,36
	Připravované rozšíření služeb o IPTV	0,11	3	0,33
	Přilákání nových klientů	0,15	3	0,45
	<b>Součet</b>	<b>0,67</b>		<b>2,06</b>
<b>T</b>	Změna legislativy	0,06	1	0,06
	Nedostatek finančních prostředků pro aktuální projekt developera	0,08	2	0,16
	Konkurence	0,09	2	0,18
	Společnost nezíská nové klienty z připravovaného projektu	0,1	1	0,1
	<b>Součet</b>	<b>0,33</b>		<b>0,5</b>
<b>Celkový součet</b>		<b>1</b>		<b>2,56</b>

Z EFE matice vyplývá, že nejvýznamnější příležitostí je aktuální obchodní příležitost s developerskou firmou, která by mohla výrazně rozšířit působnost společnosti ABC, s.r.o. do nových lokalit, což je reflektováno vysokým váženým skórem 0,92. Další příležitosti zahrnují plánované rozšíření služeb o IPTV a přilákání nových zákazníků, což směřuje na využití potenciálu pro růst a diverzifikaci služeb.

Na straně hrozeb je společnost vystavena riziku změny legislativy, potenciálního bankrotu plánovaného developerského projektu a konkurenci, které mohou ovlivnit její stabilitu a růst. Nicméně, celkové skóre pro hrozby je nízké 0,5, což naznačuje, že tyto hrozby nejsou v současnosti považovány za kritické nebo nejsou dostatečně pravděpodobné, aby výrazněji ohrozily firmu.

Konečné vážené skóre 2,56 indikuje, že společnost poměrně efektivně reaguje na externí faktory, které ovlivňují její činnosti.



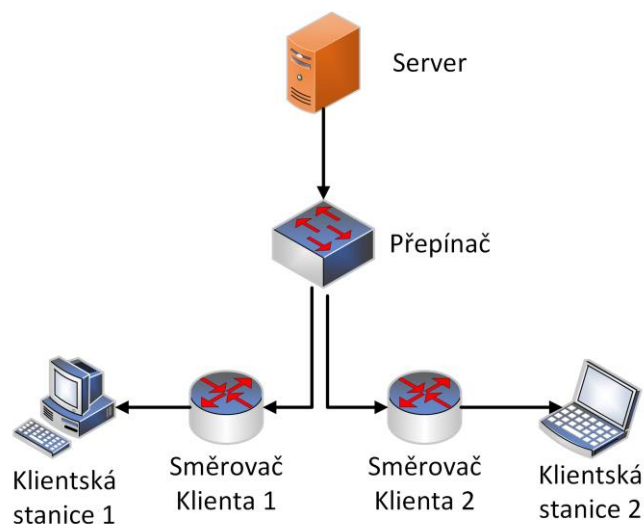
## 3 VLASTNÍ NÁVRH ŘEŠENÍ

### 3.1 Funkční předpoklady

Pro realizaci tohoto návrhu musí být validní následující předpoklady:

- Funkční server s nainstalovaným OS Ubuntu Server.
- Server s vysokorychlostní konektivitou, alespoň 1 Gbps.
- Přepínač v síti podporuje protokol IGMP.
- Směrovače klientů podporují funkci IGMP proxy.
- Místo provozu serveru je vybaveno vhodnou televizní anténou kompatibilní se standardem DVB-T2 s dostatečným ziskem a kvalitním signálem.

### 3.2 Architektura a koncept navrhovaného řešení



**Obrázek č. 16: Ilustrační schéma architektury navrhovaného řešení**  
(Zdroj: Vlastní zpracování)

Navrhované řešení předpokládá využití klasické architektury klient-server. Server má za úkol distribuovat klientům pozemní televizní vysílání DVB-T2 prostřednictvím IP sítě včetně programové nabídky, tzv. EPG. Vyobrazené schéma výše je pouze ilustrační ve zjednodušené formě logické hvězdicové topologie. Skutečné fyzické uspořádání se v praxi významně liší především z pohledu rozlehlosti počítačové sítě. To znamená, že v obrázku znázorněný přepínač je ve skutečnosti sítí několika propojených přepínačů.

### 3.2.1 Server

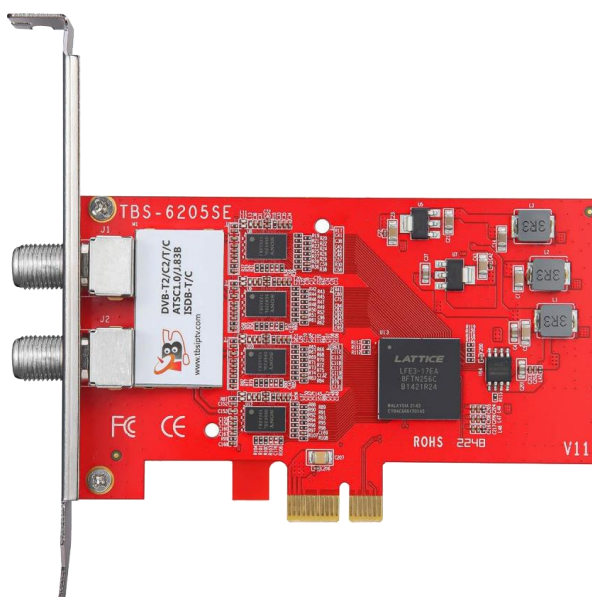
Nejdůležitější součástí tohoto řešení je server Dell PowerEdge R250, na kterém bude spuštěn software DVBlas, jehož cílem bude poskytování vysílání DVB-T2 klientům. K tomuto účelu musí být server vybaven odpovídajícím hardwarem, zejména tunerem pro příjem pozemního televizního vysílání.

Dále bude server a jeho dílčí běžící služby monitorovány pro zajištění maximální spokojenosti klientů. Monitorovací řešení není předmětem této práce.

Vzhledem k tomu, že prostředí pozemního vysílání není stálé a občas se přidávají a ruší televizní a rozhlasové programy, je zapotřebí zajistit, aby server poskytoval vždy aktuální programovou nabídku.

### 3.2.2 Televizní karta pro příjem DVB-T2

Dle předpokladů, jež byly vymezeny v kapitole 3.1, je již k dispozici anténa, kterou lze připojit k TV kartě podporující standard DVB-T2. Zmíněná karta je osazena v serveru.



Obrázek č. 17: Vyobrazení čtyřtunerové PCIe karty pro příjem DVB-T2  
(Zdroj: 30)

Zvolená karta disponuje čtyřmi tunery, z čehož vyplývá, že je možné přijímat paralelně až čtyři multiplexy. To je zcela dostatečné pro pokrytí potřeb klientů, protože se v současné době více celoplošných multiplexů v pozemním vysílání nevysílá.

Každý multiplex obsahuje několik televizních programů. Je-li přijímán jeden multiplex, pak je možné současně sledovat všechny programy v rámci přijímaného multiplexu.

Zmíněné čtyři celoplošné multiplexy se vysílají v různých oblastech v rámci ČR na odlišných nosných frekvencích. Tato práce je zaměřena pouze na oblast Brna a okolí.



**Obrázek č. 18: Fotografie zařízení Evolveo Sigma T2**  
(Zdroj: 31)

Zařízení Evolveo Sigma T2 bude použito v rámci procesu Automatického přeladování. Budou-li se aktivně využívat všechny čtyři tunery v PCIe kartě, pak již nezbude žádná volná kapacita, díky které by mohlo probíhat automatické vyhledávání současného stavu programové nabídky jak nových, tak i zrušených TV programů. Dodatečný, tj. pátý, tuner bude použit pouze pro skenování pásma DVB-T2 a tím i pro přípravu podmínek pro ostatní čtyři tunery.

### **3.3 Návrh řešení**

#### **3.3.1 Ověření dostupnosti multiplexů pro oblast Brna a okolí**

V teoretických východiscích byl proveden předběžný orientační průzkum vycházející z internetových informací. Z výsledků průzkumu vyplývá, že pro oblast Brna a okolí jsou k dispozici čtyři celoplošné multiplexy a jeden regionální multiplex nesoucí označení Regionální síť 7 na frekvenci 602 MHz.

V následujícím kroku jsou ověřeny informace z kapitoly 1.9 pomocí programového nástroje *w\_scan2*. Tento nástroj provede skenování celého pásma vyhrazeného pro DVB-T2 a automaticky vygeneruje soubor obsahující všechny potřebné údaje pro provoz IPTV. Spuštění a rozbor výstupu je uvedeno v kapitole 3.4.1 realizace řešení.

#### **3.3.2 Plán adresního prostoru pro IPTV**

Tento plán je sestaven z multicastového adresního prostoru 239.0.0.0/10 v rámci IPv4. Z tohoto velkého rozsahu byl vyčleněn menší rozsah o velikosti 256 adres začínající 239.0.1.0/24 (použitelných adres je 254).

**Tabulka č. 6: Plán adresního prostoru pro IPTV**

(Zdroj: Vlastní zpracování)

IPv4 adresa	Televizní programy
239.0.1.1	Česká televize 1 (ČT1)
239.0.1.2	Česká televize 2 (ČT2)
239.0.1.X	atd...

Z tabulky č. 6 je patrné, že všechny IP adresy pro příslušné TV programy budou postupně inkrementovány (po jedné) na místě označeného velkým písmenem X v takovém pořadí, jaké vygeneruje nástroj *w\_scan2*.

Pro zachování dostatečného prostoru mezi IP adresami, bude pro regionální informační kanál využita adresa 239.0.10.1.

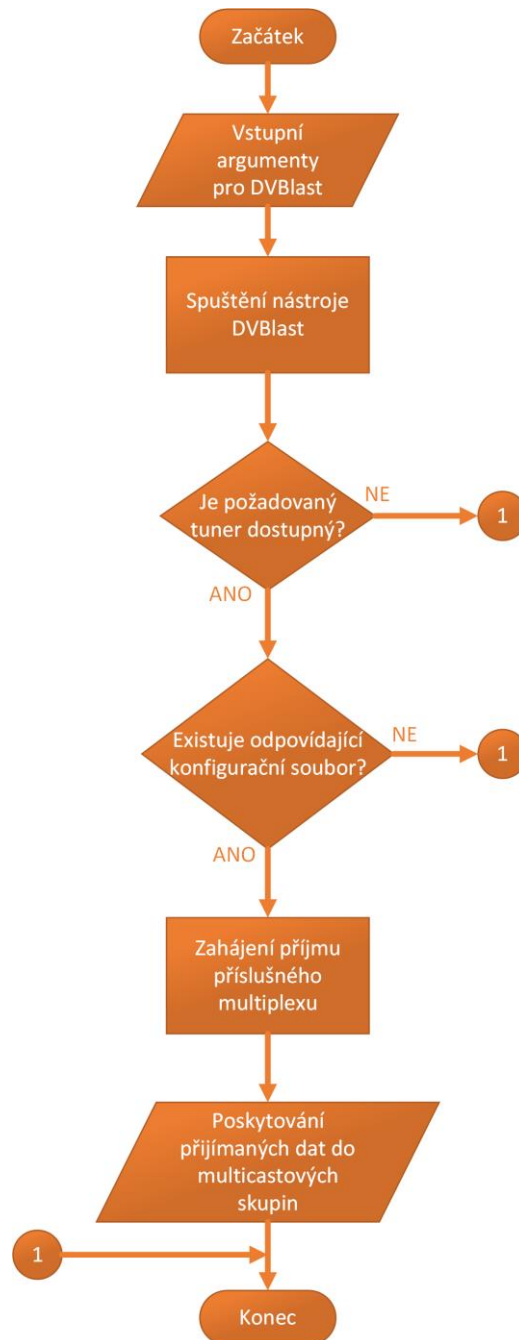
### 3.3.3 Poskytování obsahu ze sítě DVB-T2 do sítě IP

Ze souhrnu poznatků ve druhé kapitole jednoznačně vyplynulo, že nejvhodnějším softwarovým řešením pro poskytování obsahu ze sítě DVB-T2 do IP sítě je nástroj DVBlast. Tomuto nástroji je zapotřebí na vstupu specifikovat konkrétní frekvenci vybraných multiplexů a několik dalších parametrů, které jsou blíže upřesněny v kapitole realizace. Na výstupu se vymezi obor multicastových IP adres, na které bude server přepínači neustále odesílat datový proud. Tento proud bude z hlediska šířky pásma velmi objemný, ale ne trvale. To znamená, že je používán proměnlivý datový tok, známý pod zkratkou VBR (= Variable Bitrate). Hodnota datového toku může být u jednoho TV programu v HD rozlišení od přibližně 4 do 12 Mbps. Datový tok u polovičního HD rozlišení může být v rozsahu 2–6 Mbps. Má-li každý multiplex více programů a existují-li 4 multiplexy, pak lze spočítat, že se celkem vysílá přibližně 60 TV programů. Po provedení jednoduchého součtu uvažujícího pesimistickou variantu, tj. maxima dat. toků, zjistíme, že celková orientační hodnota může dosahovat až 414 Mbps. Vzhledem k tomu, že se vysílá 9 HD programů a zbytek programů v polovičním rozlišení, výpočet byl proveden následovně:  $9 \cdot 12 + 51 \cdot 6 = 414 \text{ Mbps}$ . Navíc ve výpočtu nejsou započítány rozhlasové programy, které se rovněž vysílají v síti DVB-T2. Nicméně rádia tvoří, v porovnání s TV programy, prakticky zanedbatelný podíl. Z tohoto důvodu se v kapitole 3.1 předpokládá dostupnost vysokorychlostního připojení. Díky využití multicastu není datovým provozem z DVBlastu zahlcena celá síť, nýbrž pouze jedna linka ze serveru do

přepínače. Odtud dál se multicast šíří pouze na vyžádání klientů, kteří se mohou pomocí IGMP protokolu přihlásit k odběru vybraného TV programu.

K šíření pozemního vysílání DVBlasť využívá protokol RTP. V nastavení popisovaného nástroje je možné vynutit použití obyčejného protokolu UDP, což by mělo smysl pouze v případě použití starších přijímačů nebo za účelem testování.

Tento proces je přehledně zpracován prostřednictvím vývojového diagramu č. 1.



**Diagram č. 1: Vývojový diagram procesu Poskytování obsahu ze sítě DVB-T2 do sítě IP**  
(Zdroj: Vlastní zpracování)

### 3.3.4 Oznamování o dostupnosti multimediálního vysílání v síti

K tomuto účelu bude využit protokol SAP. Tento protokol podporuje nástroj SAP server, který bude oznamovat klientům seznam dostupných televizních programů. Toto oznamování je nezbytné realizovat v souladu s připraveným IP adresním prostorem multicastových adres. SAP server pravidelně vysílá na výchozí adrese 224.2.127.254. Pokud klient projeví o poskytnutí seznamu zájem, SAP server zajistí doručení kompletního seznamu všech dostupných multimediálních proudů zahrnující názvy a IP adresy TV programů. Jedná se o tzv. playlist.

Konkrétní způsob využití tohoto způsobu oznamování pro klienty vyobrazuje vývojový diagram č. 2.

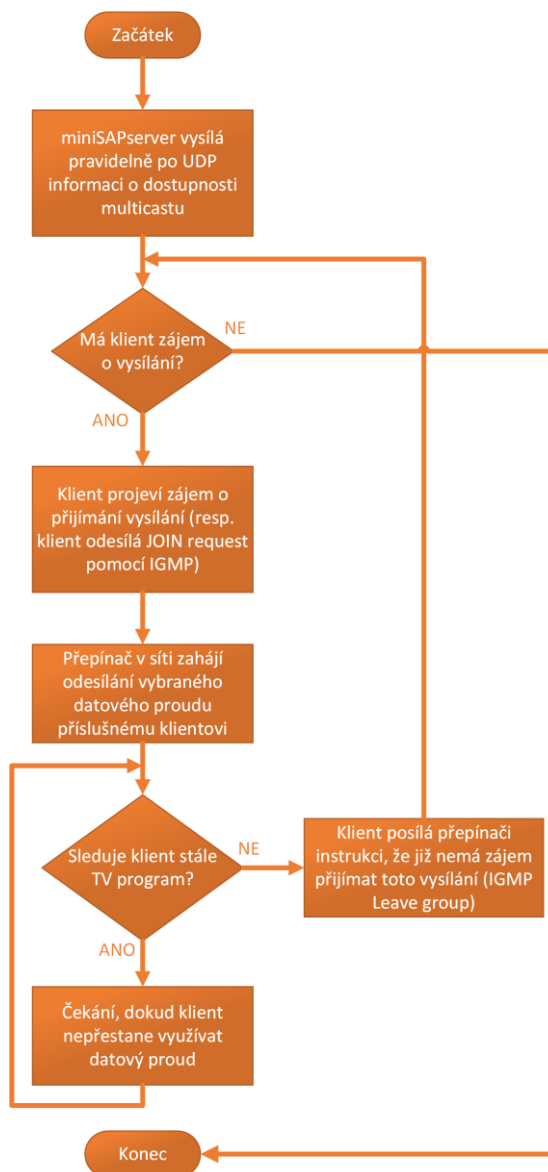


Diagram č. 2: Vývojový diagram procesu oznamování dostupnosti multimediálního vysílání v síti (Zdroj: Vlastní zpracování)

Celý proces zahajuje klient. Klient spustí nástroj disponující možností přijímat oznamování SAP protokolem. Po doručení playlistu ze SAP serveru si může klient zvolit sledování libovolného programu. Jakmile klient dokončí výběr, požádá o konkrétní TV program. Tato žádost obsahuje požadavek o připojení k vybrané multicastové skupině (tzv. IGMP Join Group). Žádost se zasílá přepínači, v případě IPv4, pomocí protokolu IGMP. Poté, co přepínač obdrží žádost, přiřadí příslušného klienta do jím zvolené skupiny, a začne mu posílat datový proud v podobě TV programu. Díky tomu se klientovi spustí přehrávání audio i video stopy, případně i dalších stop (dle TV programu) včetně programové nabídky, tzv. EPG.

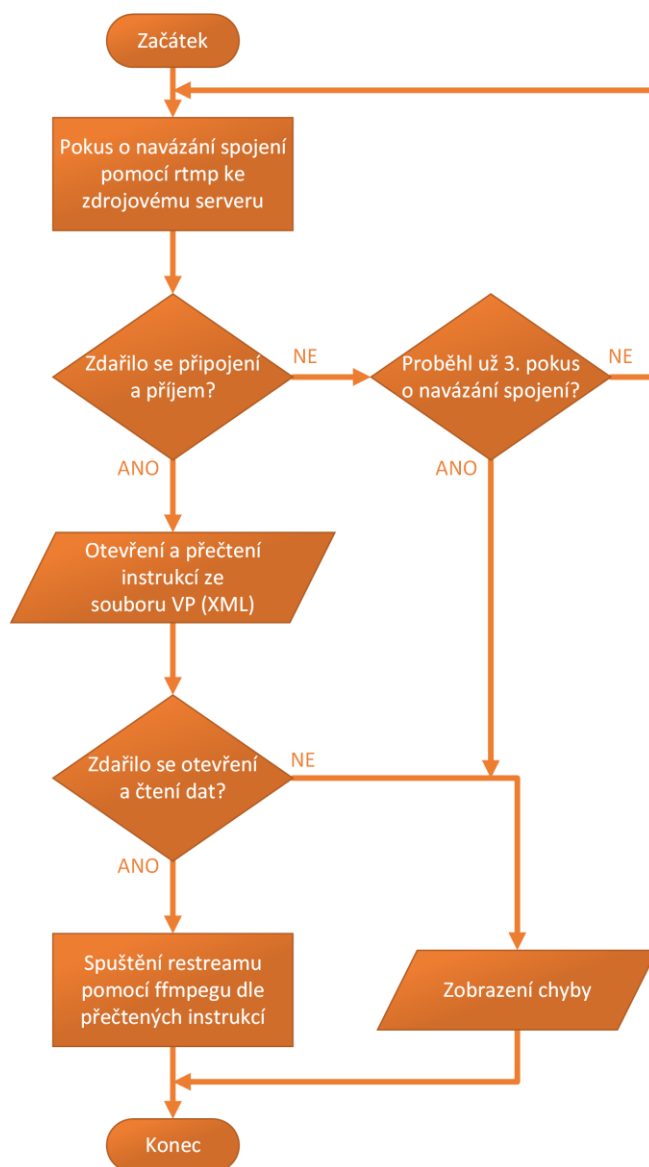
Když klient laicky řečeno „přepne TV program“, z technického hlediska požádá přepínač o ukončení přijímání jednoho programu (IGMP Leave Group) a současně zašle požadavek o zahájení přijímání jiného TV programu (IGMP Join Group). Na základě různých přehrávačů a přijímacích zařízení se mohou způsoby těchto žádostí během „přepínání“ mírně lišit. Podrobně je logika IGMP vysvětlena v kapitole 1.5.5.

### **3.3.5 Regionální informační kanál**

Jedním z požadavků investora bylo zřízení regionálního informačního kanálu dostupného rovněž pomocí služby IPTV. V rámci realizace je nezbytné připravit řešení tak, aby bylo možné poskytovat tento kanál všem klientům společnosti stejně jako TV programy ze sítě DVB-T2.

Pro tento účel bude využit nástroj FFmpeg, jehož úkolem bude přijímat vysílání pomocí protokolu RTMP a následně toto vysílání poskytovat do IP sítě společnosti v rámci vícesměrového vysílání protokolem RTP. Je vhodné realizovat tento požadavek v obecnější rovině, aby v budoucnu existovala možnost analogicky poskytovat vysílání případně dalšího infokanálu.

Je nezbytné vytvořit návrh samotné struktury souboru, který bude obsahovat dodatečné vlastnosti volitelných přenosů. Dále je zapotřebí vytvořit možnosti nastavení, zda je žádoucí zahrnout volitelné přenosy pro SAP server a do souboru M3U8. Tento proces je vyobrazen prostřednictvím vývojového diagramu č. 3.



**Diagram č. 3: Vývojový diagram procesu Odbavování regionálního televizního vysílání**  
(Zdroj: Vlastní zpracování)

### 3.3.6 Automatické přeladění

Vzhledem k tomu, že se programová nabídka občas mění, tzn. že se spouští a ruší různé TV programy, je zapotřebí zajistit, aby existovala možnost automatického přeladění. Bylo by možné takovou činnost realizovat i ručně, ale zvyšuje to režie správce a také chybovost lidského faktoru. U orientačního odhadu celkové šířky pásma DVB-T2 bylo zjištěno, že v současné době se provozuje asi 60 TV programů. To je 60 IP adres, které musí být uvedeny v konfiguračních souborech jak DVBlastu, tak i SAP serveru. Teoreticky lze vysílat všechny TV programy na jedné IP adrese a rozlišovat je pomocí identifikace portů. To by mohlo vytvořit případný problém u příjemce, protože by nemusel správně



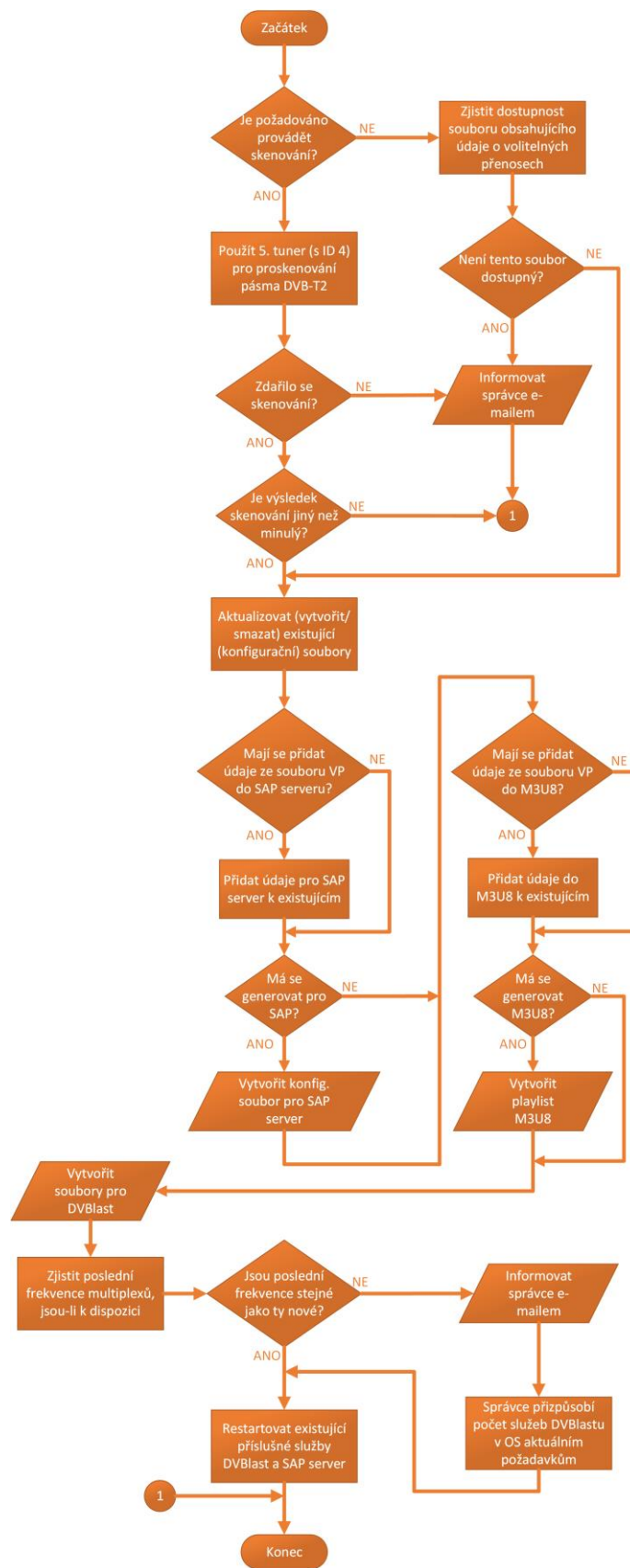
rozpoznat, jaký je použitý přenosový protokol. Další nevýhodou rozlišování při použití portů je horší orientace v konfiguraci firewallu, protože je nepsaným pravidlem vyhrazovat vždy vybrané porty konkrétním službám. Je tedy nežádoucí, aby jedné službě, resp. IPTV, bylo vyhrazeno například 70 portů. Kdyby totiž došlo ke spuštění nového TV programu, musel by správce sítě provést změnu konfigurace ručně. Využití více adres a jednoho portu je tedy vhodné řešení, protože například na firewallu tak stačí nastavit naplánovaný rozsah IP adres, přičemž jejich využití lze určovat až při využití DVBlasu. Navíc automatické řešení se může spouštět plánovaně mimo pracovní dobu zaměstnance, který by jinak během dne mohl omezit divácký komfort.

Vzhledem k tomu, že skenování může trvat nízké jednotky minut, využije se k tomuto účelu pátý USB tuner. Jakmile se dokončí skenování a řídicí skript automatického přeladění dokončí změny souvisejících konfiguračních souborů, pak dojde pouze k restartování těch služeb DVBlasu, kterých se daná změna týká a SAP serveru. Celková doba výpadku by neměla být delší než přibližně 3 sekundy. Dílčím cílem je tedy vytvoření řídicího skriptu, jež automaticky připraví všechny náležitosti související s přeladěním TV programů.

Proces Automatického přeladění začíná vyhodnocením, zda se má provádět skenování. Pokud ano, zahájí se skenování pásma DVB-T2 nástrojem *w\_scan2*. Není-li požadováno provádět skenování, má se zato, že se provede pouze aktualizace všech existujících konfiguračních souborů v souladu s nastavením včetně zahrnutí dat obsažených v souboru s volitelnými přenosy (VP). Není-li soubor VP dostupný, je opět upozorněn správce. Pokud bylo skenování požadováno a nezdařilo se, pak je automaticky upozorněn správce prostřednictvím e-mailu a proces končí. V opačném případě se vyhodnocuje, jestli výsledek stejný jako ten minulý. Je nelogické a zbytečné provádět změny, pokud v DVB-T2 žádné nenastaly. Z tohoto důvodu se budou porovnávat výsledky starého a nového skenování.

Dále se vyhodnocuje, jestli se mají k existujícím datům z *xspf* souboru připojovat volitelná data ze souboru VP. Následně se z nastavení zjišťuje, zda se má vytvořit konfigurační soubor pro SAP server. Pokud není žádána ani jedna ze dvou zmíněných úloh, jednoduše se úlohy přeskočí a pokračuje se dál k souboru formátu M3U8. U tohoto souboru jsou podmínky jeho vytvoření totožné jako v případě souboru pro SAP server. Jako poslední se vytváří soubory (spouštěcí a konfigurační) pro nástroj DVBlas. Pokud

jsou k dispozici staré soubory, funkce se pokusí získat z jejich názvů hodnoty frekvencí. Jedná se o jednoduchý způsob, jak udržovat přehled starých hodnot frekvencí bez nutnosti správy dodatečného seznamu, který by tyto hodnoty obsahoval. Správa takového seznamu by zbytečně zvýšila režie spjaté s vývojem, protože by bylo nutné ověřovat dostupnost seznamu, syntaxi, strukturu a další náležitosti. Na základě zjištěných rozdílů se provede restartování příslušných služeb, aby mohla vejít v platnost nová aktualizace a současně se minimálně omezil divácký komfort. Dojde-li k výjimečné situaci, že by se spustil nový multiplex anebo se změnila frekvence existujícího, pak na tuto situaci musí reagovat správce manuálně a adekvátně upravit systémové služby. Vzhledem k tomu, že zmíněná výjimečná situace může nastat maximálně jednou za rok, nedává smysl vyvíjet řešení pro automatické úpravy. Dalším důvodem je nutnost administrátorského oprávnění pro správu systémových služeb. Bylo by možné přidělit běžnému uživateli OS oprávnění pro správu dané systémové služby, ale tím se vytváří potenciální bezpečnostní zranitelnost. Z těchto uvedených důvodů je navržené řešení ideální. Celý proces je vyobrazen pomocí diagramu č. 4.



**Diagram č. 4: Vývojový diagram procesu automatického přeladění**  
(Zdroj: Vlastní zpracování)

### **3.3.7 Soubor formátu M3U8**

Tato podkapitola uvádí důvody, proč je používán soubor M3U8, a kde je možné jej využít. Soubor formátu M3U (MPEG Version 3.0 URL) je jednoduchý textový soubor, který poskytuje mediálnímu přehrávači seznam různých odkazů na zdroje pro příjem multimediálního obsahu. Tyto zdroje mohou být místní i vzdálené. Soubor M3U obvykle obsahuje cesty k souborům nebo internetovým streamům a může zahrnovat informace o pořadí přehrávání a další související metadata.

Soubor M3U8 je stejný jako M3U s jedním rozdílem, kterým je použití kódování UTF-8. Použití tohoto kódování je v ČR důležité, protože čeština obsahuje diakritiku, což zahrnuje znaky mimo základní sadu ASCII.

Některé přijímače nepodporují protokol SAP a vyžadují import playlistu, nejčastěji právě ve formátu M3U8. Z tohoto důvodu bude společnost ABC, s.r.o. mít k dispozici webový server, pouze pro své klienty, na kterém bude možné si takto vygenerovaný playlist stáhnout a následně použít. Tato práce neobsahuje postup instalace ani konfigurace zmíněného webového serveru, protože to není součástí cílů.

### **3.3.8 Záložní řešení**

Aby bylo možné pro klienty společnosti zajistit odpovídající kvalitu služby IPTV, bude celé navržené řešení duplikováno, tj. tentýž hardware se zakoupí dvakrát a každý bude vybaven tímž softwarem. Okamžik výpadku jednoho serveru, poskytujícího tuto službu, detekuje monitorovací služba, která automaticky zareaguje na tento stav a uvede do provozu záložní server. Oba servery budou zcela totožné v aktivním provozu. Rozdíl však bude v tom, že na druhém (záložním) serveru nebudou spuštěny služby nezbytné pro IPTV. Tyto služby se spustí až v okamžiku, kdy dojde k výpadku prvního (hlavního) serveru prostřednictvím flexibilního monitorovacího softwaru.

Služby potřebné pro poskytování IPTV nemohou být kvůli totožné konfiguraci spuštěny více než jednou. Absolutní soulad konfigurací by narušoval konzistenci již spuštěné IPTV, protože by byly podvojně využity cílové multicastové IP adresy. Je-li z hlediska architektury TCP/IP použit protokol UDP, není zaručeno doručení datagramů. Došlo by k situaci, že by dva servery posílaly téměř úplně identické datové proudy do jednoho prepínače na stejný cíl. To by mělo za příčinu smíchání obou provozů do sebe, kdy klientovi (divákovi) by byl doručen nejprve jeden snímek z hlavního serveru a o setinu

sekundy později by přišel jiný snímek ze záložního serveru. V konečném důsledku klient nebude mít plynulý obraz ani zvuk, kvůli vzájemnému narušování datových provozů z obou serverů.

Jako alternativní řešení se nabízí změnit konfiguraci záložního serveru, aby nebyla totožná s hlavním serverem. Úskalí tohoto řešení spočívá zejména v nastavení zařízení klienta. Klient si připojí svůj přijímač IPTV k síti, obdrží ze SAP serveru aktuální programovou nabídku na příslušných IP adresách a eventuálně si ji uloží. Následně si TV programy seřadí, protože SAP server nedodržuje zachování zadaného pořadí programů a tímto způsobem klient používá IPTV. V momentě změny konfigurace záložního serveru, by si klient musel znovu přeladit svůj přijímač a znovu řadit programy. Jakmile by se podařilo odstranit potíže na hlavním serveru, opět by klient nemohl sledovat televizi, protože by ji měl nastavenou na záložní server.

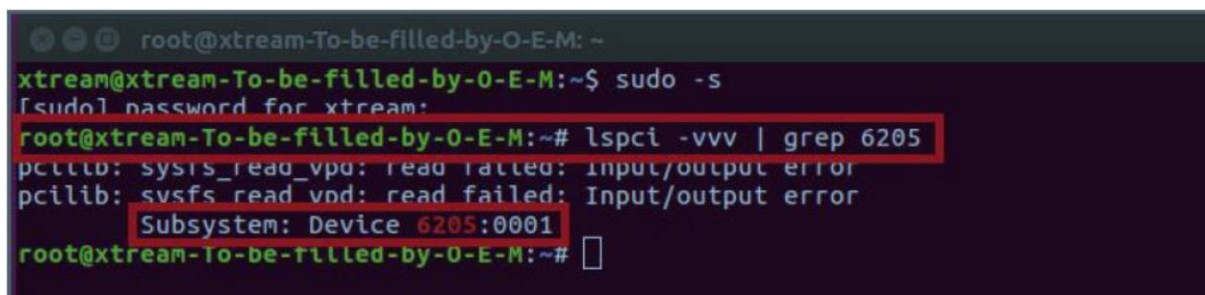
Z tohoto důvodu je značně jednodušší používat totéž nastavení na obou serverech a spouštět záložní IPTV pouze v případě výpadku hlavního serveru.

### 3.4 Realizace řešení

#### 3.4.1 Příprava a ladění

Nejprve se provede ověření správnosti připojení tunerů k serveru a jejich správnému rozpoznání operačním systémem. K tomuto účelu se využije příkaz *lspci* a *lsusb*.

Výstup prvního příkazu je vyobrazen na obrázku níže.



```
root@xtream-To-be-filled-by-O-E-M: ~
xtream@xtream-To-be-filled-by-O-E-M:~$ sudo -s
[sudo] password for xtream:
root@xtream-To-be-filled-by-O-E-M:~# lspci -vvv | grep 6205
pci11b: sysfs_read_vpd: read failed: input/output error
pci11b: sysfs_read_vpd: read failed: Input/output error
Subsystem: Device 6205:0001
root@xtream-To-be-filled-by-O-E-M:~#
```

Obrázek č. 19: Ověření dostupnosti čtyřtunerové PCIe DVB-T2 karty  
(Zdroj: 32)

Příkaz *lspci* se používá k zobrazení podrobných informací o všech zařízeních připojených do sběrnice PCI a PCIe. Argument *vvv* vynucuje poskytnutí podrobného výpisu informací. Výrobce této PCIe karty má identifikátor 6205, který je v obrázku č. 19 zvýrazněn červeným rámečkem. Čtyřčíslí 0001, nacházející se bezprostředně za ID karty,

znamená ID zařízení. Tyto identifikátory pomáhají jednoznačně určit přesný model a výrobce zařízení. Tímto úkonem byla ověřena přítomnost PCIe karty v serveru.

```
krokodyl@krokokomp:~$ lsusb
Bus 004 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 003 Device 004: ID 0e0f:0002 VMware, Inc. Virtual USB Hub
Bus 003 Device 003: ID 0e0f:0002 VMware, Inc. Virtual USB Hub
Bus 003 Device 006: ID 0572:c68a Conexant Systems (Rockwell), Inc. EyeTV Stick
Bus 003 Device 002: ID 0e0f:0003 VMware, Inc. Virtual Mouse
Bus 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 001 Device 002: ID 0e0f:0002 VMware, Inc. Virtual USB Hub
Bus 001 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
```

Obrázek č. 20: Ověření dostupnosti DVB-T2 zařízení Evolveo Sigma T2  
(Zdroj: Vlastní zpracování)

Analogicky jako v předchozím případě je použit příkaz pro vypsání dostupných zařízení připojených pomocí sběrnice USB. Bíle zvýrazněný text *Conexant Systems (Rockwell), Inc. EyeTV Stick* indikuje, že je k dispozici i poslední zařízení značky Evolveo Sigma T2.

```
krokodyl@krokokomp:~$ ls /dev/dvb
adapter0 adapter1 adapter2 adapter3 adapter4
krokodyl@krokokomp:~$
```

Obrázek č. 21: Ověření dostupnosti všech DVB-T2 tunerů (PCIe i USB)  
(Zdroj: Vlastní zpracování)

Příkaz `ls /dev/dvb` vyobrazuje všechna dostupná zařízení, které operační systém úspěšně rozpoznal jako zařízení podporující standard DVB. V tomto případě je jisté, že jsou připojena zařízení podporující konkrétně standard DVB-T2.

Pomocí série příkazů bylo výše potvrzeno, že DVB-T2 tunery jsou k dispozici a připraveny k použití. Nyní lze přistoupit k použití nástroje `w_scan2`, který automaticky provede skenování celého spektra DVB-T2. Následně se vygeneruje strukturovaný soubor s příponou `xspf` obsahující seznam TV programů, který je kompatibilní např. s přehrávačem VLC.

```
w_scan2 -a 4 -f t2 -c CZ -E 0 -L > dvbt2_channels.xspf
```

Příkaz `w_scan2` je doplněn níže uvedenými argumenty a hodnotami s vysvětlením.

- `a 4` → Argument `a` je zkráceným označením angl. slova *adapter*. Pouhé číslo 4 je zkráceným zápisem jinak daným následovně: `/dev/dvb/adapter4`.
- `f t2` → Argument `f` vyjadřuje typ tuneru. Některá zařízení mohou být současně vybavena více různými druhy tunerů (např. současně DVB-T/T2/C). Argument je volitelný, kdy v případě nepoužití `w_scan2` postupně analyzuje všechny dostupné

tunery. V tomto případě je vhodné upřesnit použití konkrétního typu tuneru, protože by se zbytečně prodlužovala doba skenování sítí DVB včetně těch již nepoužívaných. Hodnota  $t2$  specifikuje použití tuneru pro terestriální vysílání, tj. DVB-T2.

- $c$  CZ → Tento argument značí identifikátor země. Hodnota tohoto ID je vždy dvoupísmenná.
- $E$  0 → Argument s hodnotou nula je použit jako filtr pouze bezplatných TV programů. Není důvod zahrnovat i šifrované TV programy, protože investor takový požadavek nezadal.
- $L$  → Tento argument bez hodnoty instruuje  $w\_scan2$ , aby výsledný soubor byl ve formátu *xspf*.
- Ostrá uzavírací závorka a následný název souboru na konci příkazu nastavuje cílový název výstupního souboru v relativně nastaveném adresáři, přičemž lze doplnit i absolutní cestu.

Výstup příkazu výše je uveden v obrázku č. 22, ale pouze jeho část z důvodu přílišné rozsáhlosti výstupu. Výstup také zahrnuje podrobné informace o jednotlivě prohledávaných frekvencích včetně kompletního výpisu TV programů a rozhlasových programů. Tento samotný výpis není tak důležitý jako vygenerovaný strukturovaný soubor *xspf* dle příkazu v rámečku výše.

```

krokodyl@krokokomp:~$ w_scan2 -a 4 -f t2 -c CZ -E 0 -L > dvbt2_channels.xspf
w_scan2-1.0.14 (compiled for DVB API 5.11)
https://github.com/stefantalpalaru/w_scan2

using settings for CZECH REPUBLIC
DVB aerial
DVB-T Europe
scan type TERRESTRIAL, delivery system SYS_DVB_T2, channellist 4
output format vlc xspf playlist
output charset 'UTF-8'
Info: using DVB adapter auto detection.
      /dev/dvb/adapter4/frontend0 -> TERRESTRIAL "Silicon Labs Si2168": very good :-)

Using TERRESTRIAL frontend (adapter /dev/dvb/adapter4/frontend0)
-_-_- Getting frontend capabilities-_-_-
Using DVB API 5.12
frontend 'Silicon Labs Si2168' supports
DVB-T2
INVERSION_AUTO
QAM_AUTO
TRANSMISSION_MODE_AUTO
GUARD_INTERVAL_AUTO
HIERARCHY_AUTO
FEC_AUTO
BANDWIDTH_AUTO not supported, trying 6/7/8 MHz.
FREQ (48.00MHz ... 870.00MHz)
-_-_- Scanning DVB-T2...
Scanning 7MHz frequencies...
177500: plp0 (time: 00:00.435)
184500: plp0 (time: 00:02.495)
191500: plp0 (time: 00:04.543)
198500: plp0 (time: 00:06.579)
205500: plp0 (time: 00:08.591)
212500: plp0 (time: 00:10.655)
219500: plp0 (time: 00:12.695)
226500: plp0 (time: 00:14.719)
Scanning 8MHz frequencies...
474000: plp0 (time: 00:16.763)
482000: plp0 (time: 00:18.819)
490000: plp0 (time: 00:20.899)
498000: plp0 (time: 00:23.050)
506000: plp0 (time: 00:25.062)
514000: plp0 (time: 00:27.134)          signal ok:      QAM_AUTO f = 514000 kHz I999B8C999D999T999G999Y999P0 (0:0:0)
QAM_AUTO f = 514000 kHz I999B8C999D999T999G999Y999P0 (0:0:0) : updating transport_stream_id -> (0:0:8465)
QAM_AUTO f = 514000 kHz I999B8C999D999T999G999Y999P0 (0:0:8465) : updating network_id -> (0:12608:8465)
522000: plp0 (time: 00:29.158)
530000: plp0 (time: 00:31.226)
538000: plp0 (time: 00:33.274)

```

**Obrázek č. 22: Opis části výsledku skenování DVB-T2 nástrojem *w\_scan2***  
(Zdroj: Vlastní zpracování)

Z výstupu nástroje *w\_scan2* je patrné, že byl nalezen signál na frekvenci 514 MHz. To přesně odpovídá frekvenci, jež byla teoreticky zjištěna a uvedena v tabulce č. 1. Z plnohodnotného výstupu *w\_scan2* vyplývá, že se úspěšně podařilo najít i ostatní tři multiplexy, tj. 570, 626 a 674 MHz. Nepodařilo se naladit Regionální síť 7 na frekvenci 602 MHz. K tomuto neúspěchu pravděpodobně došlo kvůli slabému signálu či špatně orientované anténě. Vzhledem k požadavkům společnosti není potřebné řešit příjem tohoto multiplexu. V druhé části výstup obsahuje podrobně vypsané TV programy – viz ukázka na obrázku č. 23.

```

tune to: QAM_AUTO f = 514000 kHz I999B8C999D999T999G999Y999P0 (0:12608:8465) (time: 02:05.539)
service = CT 1 HD T2 (CESKA TELEVIIZE)
service = CT 2 HD T2 (CESKA TELEVIIZE)
service = CT 24 HD T2 (CESKA TELEVIIZE)
service = CT sport HD T2 (CESKA TELEVIIZE)
service = CT :D/art HD T2 (CESKA TELEVIIZE)
service = CT 1 SM HD T2 (CESKA TELEVIIZE)
service = CT 1 JM HD T2 (CESKA TELEVIIZE)
service = CT 1 SVC HD T2 (CESKA TELEVIIZE)
service = CT 1 JZC HD T2 (CESKA TELEVIIZE)
service = CRo RADIOZURNAL T2 (CESKY ROZHLAS)
service = CRo DVOJKA T2 (CESKY ROZHLAS)
service = CRo VLTAVA T2 (CESKY ROZHLAS)
service = CRo RADIO WAVE T2 (CESKY ROZHLAS)
service = CRo D-DUR T2 (CESKY ROZHLAS)
service = CRo RADIO JUNIOR T2 (CESKY ROZHLAS)
service = CRo PLUS T2 (CESKY ROZHLAS)
service = CRo JAZZ T2 (CESKY ROZHLAS)
service = CRo RZ SPORT T2 (CESKY ROZHLAS)
service = CRo POHODA T2 (CESKY ROZHLAS)

```

**Obrázek č. 23: Výpis části výsledku skenování DVB-T2 nástrojem *w\_scan2* – TV programy**  
(Zdroj: Vlastní zpracování)



V obrázku č. 23 je uveden anglický pojem „service“. V českém doslovném překladu se jedná o službu. Zjednodušeně lze službu považovat za obyčejný televizní program (např. ČT1). Tato zmínka o názvosloví je zcela na místě, protože v následující podkapitole se používají zkratky spjaté s angl. označením service. Analogicky *w\_scan2* vypsal TV programy všech čtyř multiplexů a současně vygeneroval požadovaný formát *xspf*.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <playlist xmlns="http://xspf.org/ns/0/" xmlns:vlc="http://www.videolan.org/vlc/playlist/ns/0/" version="1">
3   <title>DVB Playlist</title>
4   <creator>w_scan2-1.0.14</creator>
5   <info>https://github.com/stefantalpalaru/w_scan2</info>
6   <trackList>
7     <track>
8       <title>0001. CT 1 HD T2</title>
9       <location>dvb-t2://frequency=514000000</location>
10      <extension application="http://www.videolan.org/vlc/playlist/0">
11        <vlc:option>dvb-bandwidth=8</vlc:option>
12        <vlc:option>dvb-ts-id=8465</vlc:option>
13        <vlc:id>2</vlc:id>
14        <vlc:option>program=268</vlc:option>
15      </extension>
16    </track>
17    <track>
18      <title>0002. CT 2 HD T2</title>
19      <location>dvb-t2://frequency=514000000</location>
20      <extension application="http://www.videolan.org/vlc/playlist/0">
21        <vlc:option>dvb-bandwidth=8</vlc:option>
22        <vlc:option>dvb-ts-id=8465</vlc:option>
23        <vlc:id>3</vlc:id>
24        <vlc:option>program=270</vlc:option>
25      </extension>
26    </track>
```

Obrázek č. 24: Ukázka souboru *xspf* vygenerovaného nástrojem *w\_scan2*  
(Zdroj: Vlastní zpracování)

Popisovaný nástroj *w\_scan2* disponuje i více druhy výstupů, kterými jsou strukturované soubory. V tomto případě byl zvolen formát *xspf*, protože jednak vychází z formátu XML a také je ihned přehratelný v přehrávači VLC. Výhoda použitého formátu XML je ještě zmíněna v rámci procesu Automatického přeladění.

### 3.4.2 Konfigurace DVBlast

Jak již bylo řečeno v návrhu řešení, DVBlast je nezbytný pro převod signálu z DVB-T2 do IP sítě.

Konfigurace DVBlastu sestává z činností:

- 1) nastavení hodnot argumentům v příkazové řádce,
- 2) tvorba konfiguračních souborů obsahujících cílové multicastové IPv4 adresy a identifikátory TV programů (SID).

Pro lepší porozumění jednotlivým parametrům byla využita utilita *man*, volaná stejnojmenným příkazem, což vyjadřuje manuál. Spustí-li se příkaz *man dvblast*, dojde k vyobrazení textové stránky v příkazové řádce, která obsahuje detailní informace o nástroji DVblast. Touto utilitou disponuje většina novějších Linuxových OS.

```

DVBLAST(1) User Commands
DVBLAST(1)

NAME
  DVblast - Simple and powerful dvb streaming application

SYNOPSIS
  dvblast [-q] [-c <config_file>] [-r <remote_socket>] [-t <ttl>] [-o <SSRC_IP>] [-i <RT_priority>] [-a <adap-
  ter>] [-n <frontend_number>] [-v <ca_number>] [-S <diseqc>] [-k <uncommitted_port>] [-f
  <frequency>] [-D <src_host>[:<src_port>][[:<src_mcast>][[:<port>]][/<copts>]] [-A <ASI_adapter>] [-s <symbol_r-
  ate>] [-v <0|13|18>] [-p] [-b <bandwidth>] [-I <inversion>] [-F <fec_inner>] [-m <modula-
  tion>] [-R <rolloff>] [-P <pilot>] [-K <fec_lp>] [-G <guard_interval>] [-H <hierarchy>] [-X <tra-
  nsmission>] [-O <lock_timeout>] [-u] [-w] [-U] [-L <latency>] [-E <retention>] [-d
  <dest_ip>[:<port>][[/<copts>]] [-z] [-C] [-e] [-M <network_name>] [-N <network_ID>] [-T] [-j <system_charset>]
  [-W] [-Y] [-l] [-g <logger_ident>] [-Z <mrtg_file>] [-V] [-h] [-0 <pid_mapping>] [-1
  <mis_id>] [-2 <size>] [-5 <DVBS|DVBS2|DVBC_ANNEX_A|DVBT|ATSC>] [-J <DVB_charset>] [-B <provider_name>] [-Q <
  quit_timeout>] [-x <text|xml>]

DESCRIPTION
  DVblast is a simple and powerful streaming application based on the linux-dvb API. It opens a DVB device, tu-
  nes it, places PID filters, configures a CAM module, and demultiplexes the packets to sev-
  eral RTP outputs.

  DVblast is designed to be the core of a custom IRD or CID, based on a PC with Linux-supported DVB cards.

  DVblast does not do any kind of processing on the elementary streams, such as transcoding or remultiplexing.
  it does not stream from plain files, only DVB devices. If you were looking for these fea-
  tures, switch to VLC.

OPTIONS
  -a, --adapter <adapter>
      Address of the adapter

  -A, --asi-adapter
      Read packets from an ASI adapter (0-n). If support for Deltacast ASI cards is compiled in, prefix the
      adapter number with "deltacast:" to select a Deltacast input. Deltacast inputs are num-
      bered as (100 * card number) + channel number.

  -b, --bandwidth <bandwidth>
Manual page dvblast(1) line 1 (press h for help or q to quit)

```

**Obrázek č. 25: Ukázka manuálu k nástroji DVblast v OS Ubuntu 22**  
(Zdroj: Vlastní zpracování)

```

dvblast -a 0 -f 514000000 -m qam_auto --delsys DVBT2 --dvb-plp-id -
1 -z -b 8 -e -c /dvblast/dvbt2_config.cfg

```

Příklad v rámečku výše ilustruje spuštění nástroje DVblast s následujícími argumenty:

- a 0 → Identifikátor zařízení v OS. Stejně jako u nástroje *w\_scan2*, jde i zde o zkrácený zápis cesty */dev/dvb/adapter0*, což znamená, že je používán adaptér s ID 0.
- f 514000000 → Přijímaná vybraná frekvence jednoho multiplexu v Hz.
- m QAM\_auto → Modulace signálu – je nastavena na automatické zjištění.
- delsys DVBT2 → Volba přijímaného standardu (DVB-S, DVB-S2 a další možnosti).

- 1 → Nastavení ID datového toku pro transpondéry podporující více datových toků.
- z → Tento argument instruuje DVBLast, aby propustil veškeré vysílané proudy ze sítě DVB (audio, video, titulky) do IP.
- b 8 → Bandwidth (šířka pásma).
- e → Aktivace propouštění dat EPG ze sítě DVB do IP sítě.
- c /dvblast/dvbt2\_config.cfg → Absolutní cesta ke konfiguračnímu souboru.

Obsah konfiguračního souboru je vyobrazen na obrázku č. 26.

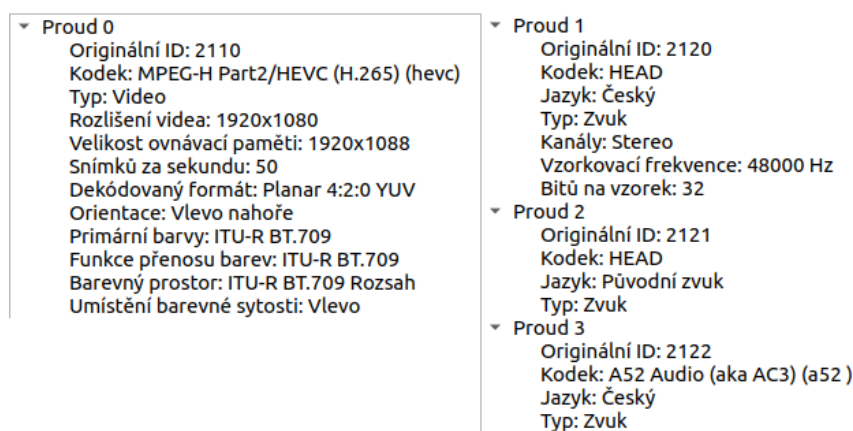
```

1 239.0.1.1:5004 1 268 # CT 1 HD T2
2 239.0.1.2:5004 1 270 # CT 2 HD T2
3 239.0.1.3:5004 1 272 # CT 24 HD T2
4 239.0.1.4:5004 1 274 # CT sport HD T2
5 239.0.1.5:5004 1 276 # CT D/art HD T2
6 239.0.1.6:5004 1 280 # CT 1 SM HD T2

```

**Obrázek č. 26: Ukázka konfiguračního souboru pro nástroj DVBLast**  
(Zdroj: Vlastní zpracování)

Konfigurační soubor byl vytvořen přesně podle připraveného plánu. První sloupec obsahuje multicastové adresy doplněné číslem portu protokolu RTP a třetí sloupec vyjadřuje tzv. SID, neboli Service ID. Dokumentace DVBLastu uvádí možnost doplnění identifikátorů dílčích proudů, kterými jsou např. video a audio stopa anebo titulky, do čtvrtého sloupce. Není-li tento sloupec vyplněn, pak se klientům přeposílají všechny dostupné proudy. Navíc byly doplněny komentáře, začínající křížkem #, ke každému řádku v ukázce na obrázku č. 26 v podobě názvu příslušného TV programu. Je tak učiněno proto, aby se zaměstnanec, v případě nutného manuálního zásahu, snadněji orientoval a mohl účinně provést potřebné změny v konfiguračním souboru.



**Obrázek č. 27: Snímek z aplikace VLC – vyobrazení některých proudů TV programu ČT1**  
(Zdroj: Vlastní zpracování)

Na obrázku č. 27 jsou vypsány některé proudy, které jsou dostupné v rámci TV programu České televize. Bude-li potřeba omezit přijímané proudy, lze najít jejich identifikátory (2110, 2120, 2121, ...), jejichž označení aplikace VLC nazývá jako *Originální ID*. Tato ID lze vpsat do zmiňovaného čtvrtého sloupce konfiguračního souboru DVBlastu.

### 3.4.3 Vytvoření služby pro DVBlast

Pro jednodušší řízení a monitoring provozované IPTV se vytvoří služby. Řešení systému antén pro příjem DVB-T2 může být různorodé. To znamená, že může být použito více směrových antén pro zajištění dostatečně silného a kvalitního signálu jednotlivých multiplexů pro následné zpracování. Proto je namíste navrhované řešení diverzifikovat. Pro každý multiplex se vytvoří jedna služba. Díky tomu je možné, např. v případě údržby části anténních rozvodů, zastavit pouze ty služby, o kterých technik ví, že budou dočasně nedostupné. Jde tedy o flexibilní řešení. Bude-li potřeba hromadné spuštění či zastavení všech služeb současně, stačí sestavit jednoduchý skript. Tento skript může vypadat, pro případ spuštění služeb, dle obrázku č. 28.

```
1 systemctl start dvblast_514.service
2 systemctl start dvblast_570.service
3 systemctl start dvblast_626.service
4 systemctl start dvblast_674.service
```

Obrázek č. 28: Ukázka skriptu pro hromadné spuštění služeb DVBlastu  
(Zdroj: Vlastní zpracování)

Obdobně lze sestavit skript, jež zastaví všechny služby současně. Stačí zaměnit slova *start* za *stop*. Níže je uvedena ukázka na obrázku č. 29 konfigurace jedné vybrané služby.

```
[Unit]
Description=DVBlast Service for MUX24 on freq 674MHz
After=network.target

[Service]
Type=simple
ExecStart=/iptv/dvblast/run_674000000.sh
Restart=no

[Install]
WantedBy=multi-user.target
```

Obrázek č. 29: Ukázka konfigurace služby pro DVBlast – MUX24  
(Zdroj: Vlastní zpracování)

První sekce *[Unit]* konfiguračního souboru služby má tyto parametry:

- *Description* – Popis služby zobrazující se např. při zjišťování aktuálního stavu služby.

- After – Je důležitý při spuštění OS, protože nepřímo operačnímu systému oznamuje, kdy se má služba spustit. V tomto případě má dojít ke spuštění služby až po té, kdy je připraveno síťové připojení (*network.target*)

Druhá sekce [*Service*] se již týká přímo vlastností služby:

- Type – Typ služby.
- ExecStart – Absolutní cesta ke skriptu, který má být řízen službou.
- Restart – Toto je nezbytný parametr vyjadřující, za jakých okolností má dojít k restartování služby. Lze nastavit mnoho různých událostí, na které může parametr reagovat. V tomto případě je hodnota nastavena na *no*, protože je naplánován aktivní monitoring externí aplikací. Pokud dojde k samovolnému ukončení služby, pak se chyba očekává v konfiguraci DVBlastu, a tudíž je zbytečné se pokoušet službu opakovaně spouštět.

Poslední sekce [*Install*] již obsahuje jen jeden parametr.

- WantedBy – Tento parametr určuje, jací uživatelé nebo skupiny mohou tuto službu spravovat. Vzhledem k tomu, že server, popisovaný v této práci, počítá pouze s jedním správcem, lze použít hodnotu *multi-user.target*.

### 3.4.4 Konfigurace SAP

SAP server je z hlediska ovládání podobný DVBlastu. I v tomto případě je zapotřebí vytvořit konfigurační soubor a spouštěcí příkaz.

```

1  [general]
2  interface=ens33
3  sap_delay=5
4
5  [program]
6  type=rtp
7  name=CT 1 HD T2
8  address=239.0.1.1
9  port=5004
10 user=spolecnost
11 playlist_group=Televize
12
13 [program]
14 type=rtp
15 name=CT 2 HD T2
16 address=239.0.1.2
17 port=5004
18 user=spolecnost
19 playlist_group=Televize

```

Obrázek č. 30: Ukázka konfiguračního souboru SAP serveru  
(Zdroj: Vlastní zpracování)

Celý konfigurační soubor, uvedený na obrázku č. 30, se skládá ze dvou částí. První část je obecná a týká se nastavení celého SAP serveru. Druhá část je již konkrétní a obsahuje všechny informace o datových proudech nebo, jinak řečeno, informace o TV programech.

První část [*general*] zahrnuje:

- *interface=ens33* – Označení síťového rozhraní, které má být použito pro oznamování SAP protokolem.
- *sap\_delay=5* – Četnost oznamování v sekundách do sítě. Pět vteřin by měl být dostatečně rozumný kompromis mezi nadbytečným zatěžováním sítě a také dobou čekání uživatele, než obdrží oznámení.

Dle dokumentace lze zadat i více parametrů včetně například IP adresy, na kterou se mají oznámení zasílat. Oznamování, v tomto kontextu, prakticky vyjadřuje, že SAP server pravidelně odesílá po UDP datagramy obsahující informace z druhé části konfiguračního souboru vyobrazeného na obrázku č. 30.

Druhá část sestává z několika menších, stejně označených, součástí [*program*] a zahrnuje:

- *type=rtp* – Určuje použitý typ datového proudu.
- *name=ČT 1 HD T2* – Název datového proudu, resp. TV programu (je kompatibilní s kódováním UTF-8, tzn. lze používat diakritiku).
- *address=239.0.1.1* – Adresa IP z oboru vyhrazeného pro šíření multicastu.
- *port=5004* – Byl použit výchozí port pro protokol RTP.
- *user=spolecnost* – Navzdory zavádějícímu názvu parametru *user*, dokumentace uvádí, že tento parametr je volitelný a slouží k zadání názvu organizace. Během praktického testování SAP serveru bylo vyzkoušeno zadání textového řetězce s diakritikou, avšak s negativním výsledkem. SAP server se sice spustil, ale přehrávač VLC nezachytil žádné oznámení ze SAP serveru. Z tohoto experimentu lze vyvodit, že podpora UTF-8 není u všech parametrů jednotná.
- *playlist\_group=Televize* – Tento parametr je použit pro třídění datových proudů do skupin. V tomto případě byly vytvořeny dvě skupiny *Televize* a *Radia*. Kritéria, podle kterých se rozlišuje, zda se jedná o TV program nebo rádio, blíže popisuje následující podkapitola.

Takto vytvořený konfigurační soubor se následně využije při spouštění samotného SAP serveru.

```
sapserver -c /cesta/k/konfig/souboru/sap_config.cfg
```

Z příkazu v rámečku výše je na první pohled patrné, že stačí použít argument *c* a jako hodnotu postačí uvést cestu ke zmíněnému konfiguračnímu souboru.

### 3.4.5 Poskytování regionálního informačního kanálu

V návrhu v podkapitole 3.3.5 bylo uvedeno, že popisovaná společnost požaduje možnost poskytování regionálního informačního kanálu ve své IPTV. Způsob poskytování tohoto kanálu je řešen přesně dle navrženého postupu. Vychází se z předpokladu, že tento kanál provozuje jiná organizace a společnosti ABC, s.r.o. poskytuje již výsledný datový proud pomocí protokolu RTMP. Pro zjednodušení distribuce tohoto kanálu bude klientům společnosti pomocí aplikace FFmpeg přeposílán tento proud vícesměrovým vysíláním v IP síti za využití RTP. Tento způsob přeposílání datového proudu je efektivní zejména při distribuci. Server společnosti ABC, s.r.o. bude neustále tento proud přijímat, přičemž následně bude klientům poskytován na základě žádosti dle protokolu IGMP. Rámeček níže obsahuje příkaz zajišťující přeposílání tohoto datového proudu.

```
ffmpeg -re -reconnect_streamed 1 -reconnect_max_retries 3 -  
reconnect_delay_max 20 -i "rtmp://zdrojova.domena/stream" -c copy -  
f rtp_mpegts "rtp://239.0.10.1:5004"
```

Tento základní příkaz tvoří následující argumenty.

- *re* – Použití tohoto argumentu zajistí, že FFmpeg bude běžet v reálném čase.
- *reconnect\_streamed 1* – Pokud je tento argument nastaven na hodnotu 1 neboli True, pak se FFmpeg bude pokoušet znovu připojit živý proud.
- *reconnect\_max\_retries 3* – Tento argument nastavuje počet pokusů o znovupřipojení zdrojového proudu. Podle navrženého procesu dle diagramu č. 3 má být počet pokusů roven třem.
- *reconnect\_delay\_max 20* – Nastavení maximální prodlevy v sekundách, po jejímž uplynutí se mají pokusy o obnovení připojení ukončit.
- *i* – Odkaz na zdroj datového proudu příslušné organizace.
- *c* – Souhrnně nastavuje kodek pro audio i video stopu. V tomto případě hodnota *copy* instruuje FFmpeg, že má být použit stejný kodek jako zdrojový.

- f – Formát výstupního proudu.
- "rtp://239.0.10.1:5004" – Cílový protokol, IPv4 adresa a port. Zvolená IP adresa odpovídá návrhu.

### 3.4.6 Automatické přeladění

Automatické přeladění je realizováno na základě připraveného procesu, který je popsán v kapitole 3.3.6. Skript je vyvíjen pomocí programovacího jazyka Python. Stěžejní částí této podkapitoly je vytvoření funkčního skriptu, na jehož vstupu je výstup nástroje *w\_scan2* a výstupem je sada konfiguračních souborů pro DVblast, SAP server, a navíc je generován playlist ve formátu M3U8.

Nastavení různých parametrů a souvisejících vlastností spojených s chováním skriptu je centralizováno do jedné globální proměnné s názvem *settings*.

```

8 settings = {
9     "scan": {
10         "id_tuner": "4",
11         "band": "t2",
12         "country": "CZ",
13         "free_ch": "0",
14         "dir": "/iptv/xspf/",
15         "filename": "dvbt2_channels.xspf",
16         "filename_temp": "dvbt2_channels_temp.xspf"
17     },
18     "DVblast": {
19         "conf_files_dir": "/iptv/dvblast/",
20         "cmd_files_dir": "/iptv/dvblast/",
21         "id_tuner": [0,1,2,3]
22         "id_tuner_default": 0,
23         "ipv4_conf": {
24             "base_IP": "239.0.1.",
25             "start_IP": 1,
26             "inkrement": 1,
27             "port": "5004"
28         }
29     },
30     "SAP_server": {
31         "create": True,
32         "dir": "/iptv/sap_server/",
33         "filename_cfg": "sap_config.cfg",
34         "interface": "ens33",
35         "sap_delay": "5",
36         "user": "spolecnost",
37         "type": "rtp",
38         "tridit_tv_radia": True,
39         "tridit_tv_radia_meze": {
40             "od": 10000,
41             "do": 22000
42         }
43     },
44     "playlist_m3u8": {
45         "create": True,
46         "protocol": "rtp",
47         "dir": "/iptv/http_server/",
48         "filename": "dvb_playlist.m3u8"
49     },
50     "my_streams": {
51         "add_SAP": True,
52         "add_m3u8": True,
53         "source": "/iptv/my_streams/my_streams.xml"
54     }
55 }

```

Obrázek č. 31: Automatické přeladění – skript – proměnná *settings*  
(Zdroj: Vlastní zpracování)



Globální proměnná *settings* sestává ze čtyř hlavních součástí, kterými jsou:

- *scan*,
- *DVBlast*,
- *SAP\_server*,
- *playlist\_m3u8*.

Globální proměnná byla zvolena proto, aby ji bylo možné volat odkudkoli, což právě v případě této strukturované proměnné, obsahující hlavní parametry týkající se nastavení celého skriptu, je potřebné.

První součást obsahuje nastavení určené pro nástroj *w\_scan2*. Je specifikován identifikátor tuneru *id\_tuneru*, což v tomto případě znamená, jaký tuner se má využít pro skenování sítě DVB-T2. Dále lze nastavit pásmo *band*, což reprezentuje možnost volby přijímaného standardu, zda je v rámci zařízení s *id\_tuneru* vyžadováno použití DVB-T/T2/C, atp... V neposlední řadě lze hodnotou parametru *free\_ch* určit, zda se mají vyhledávat pouze nešifrované TV programy – hodnota 0 vyjadřuje pouze zdarma dostupné vysílání, zatímco hodnota 1 zahrnuje veškeré dostupné programy včetně šifrovaných. Nedílný doplněk nastavení tvoří možnost definování výchozí složky *dir* a názvů souborů, které má *w\_scan2* vytvořit. Jak je z obrázku č. 31 zřejmé, je zapotřebí zadat jak *filename*, tak i *filename\_temp*, přičemž důvod tohoto způsobu zadání je uveden dále.

Druhá součást obsahuje následující nastavení pro *DVBlast*.

- *conf\_files\_dir* → Adresář, do kterého se mají vytvářet konfigurační soubory.
- *cmd\_files\_dir* → Adresář sloužící pro spustitelné soubory, obvykle s příponou *sh*, uváděný v jazyce BASH.
- *id\_tuner* → Zde je použit datový typ pole, do kterého je nezbytné postupně zadat identifikátory těch tunerů, které má používat *DVBlast*. Zjistit dostupnost tunerů musí provést zaměstnanec ručně dle postupu uvedeného v návrhové části této práce.
- *id\_tuner\_default* → Tento parametr je významný ve chvíli, kdy se použijí všechny tunery uvedené v předchozím poli a současně bude zapotřebí přiřadit tuner multiplexu, pro který žádný volný tuner nezbyl.

- *ipv4\_conf* → Tento parametr obsahuje podmnožinu bližšího nastavení konfigurace adresace IP verze 4.
  - *base\_IP* → Obsahuje základní multicastovou IP adresu bez zadaného posledního oktetu. Tento poslední oktet je vyhrazen pro počítač, které je ozřejmáno dále.
  - *start\_IP* → Číslo uváděné v posledním oktetu, od kterého se má začít inkrementovat.
  - *inkrement* → Číslo, po kterém se má přičítat hodnota v posledním oktetu adresy.
  - *port* → Číslo portu pro RTP proud.

V některých případech se nachází číselná hodnota v uvozovkách jako textový řetězec. Nejedná se o chybu, nýbrž o záměr. Číselná hodnota, v odpovídajícím datovém typu, je použita jen tehdy, kdy je potřebné s hodnotou pracovat jako s číslem a provádět s ní např. aritmetické operace. V této práci je ve většině situací potřebné pracovat spíše s text. řetězci nežli s čísly, a proto je zvolené řešení v pořádku.

V pořadí třetí součást *SAP\_server* sestává z níže uvedených parametrů.

- *create* → Má booleovskou hodnotu určující, zda se má generovat soubor pro SAP server nebo nikoli.
- *dir* → Absolutní cesta do adresáře vyhrazeného pro SAP server.
- *filename\_cfg* → Název konfiguračního textového souboru včetně přípony. Přípona *cfg* není nutná. Může být i varianta *conf* anebo žádná. Přípona je uvedena pouze pro uživatelskou přehlednost.
- *interface* → Nastavení síťového rozhraní, které se má využívat pro oznamování SAP serverem. Lze zjistit např. pomocí utility *ifconfig*.
- *sap\_delay* → Hodnota udávající četnost odeslaných oznámení do sítě v sekundách.
- *user* → Nepovinný parametr používaný k označení společnosti provozující IPTV.
- *type* → Volba typu protokolu RTP/UDP
- *tridit\_tv\_radia* → Tento parametr může být buď True nebo False. Určuje, jestli se mají třídit rádia a televize podle rozsahu ID programu (SID) do skupin.

- *tridit\_tv\_radia\_meze* → Zde je možné vymežit hranice SID, podle kterých bude probíhat třídění TV programů a rádií do skupin.

Předposlední dodatečnou součástí je funkce tvoření playlistů ve formátu M3U8. Pro vygenerování tohoto souboru je zapotřebí definovat níže uvedené parametry.

- *create* → Parametr určující, zda uživatel požaduje vygenerování souboru tohoto formátu. Datový typ hodnoty je Boolean.
- *protocol* → Určuje, jaký protokol má být použit.
- *dir* → Absolutní výchozí cesta, kam se má tento soubor uložit.
- *filename* → Plnohodnotný název souboru včetně přípony (zde již přípona má význam).

Poslední dodatek *my\_streams* obsahuje nastavení týkající se volitelných přenosů (VP). Pokud je zapotřebí přidat k automaticky vygenerovaným TV programům vlastní zdroje datových proudů, lze tak učinit právě za pomoci nastavení níže uvedených hodnot.

- *add\_SAP* → Pomocí boolovské hodnoty definuje, zda se mají připojit data ze souboru VP do konfiguračního souboru SAP serveru.
- *add\_m3u8* → Tento parametr funguje analogicky jako *add\_SAP*, ale v kontextu pro soubor M3U8.
- *source* → Hodnotou parametru je lokální cesta k souboru VP

Příklad struktury souboru VP je zřejmý z obrázku č. 32.

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <mystreams>
3    <stream>
4      <title>Live stream Alpha3</title>
5      <protocol>rtp</protocol>
6      <address>239.0.20.1</address>
7      <port>5004</port>
8      <company></company>
9      <group>skupina1</group>
10     <note>Vlastní poznámka autora záznamu 1</note>
11   </stream>
12   <stream>
13     <title>Live stream Beta4</title>
14     <protocol>rtp</protocol>
15     <address>239.0.20.2</address>
16     <port>5004</port>
17     <company>spolecnost ABC</company>
18     <group>skupina2</group>
19     <note>Vlastní poznámka autora záznamu 2</note>
20   </stream>
21 </mystreams>

```

Obrázek č. 32: Vyobrazení struktury souboru VP (vlastních přenosů)  
(Zdroj: Vlastní zpracování)

Jak je z obrázku č. 32 patrné, byla použita struktura typu XML, kdy kořenová úroveň *my\_streams* obsahuje několik informací o volitelně vytvořených přenosech definovaných značkou *stream*. Hodnoty uvedené v rámci každé značky *stream* jsou použity v souladu s nastavením, v proměnné *settings*, což je lépe vysvětleno dále.

## Funkce a procedury

```
64 def w_scan2(id_tuner, band, country, free_ch, filename):
65     return "w_scan2 -a {0} -f {1} -c {2} -E {3} -L > {4}".format(id_tuner, band, country, free_ch, filename)
```

Obrázek č. 33: Automatické přeladění – skript – funkce *w\_scan2*

(Zdroj: Vlastní zpracování)

Pro jednoduché dynamické sestavování spustitelných příkazů je použita funkce *format*. Prakticky je v textovém řetězci uveden regulární příkaz, jehož hodnoty argumentů jsou zastoupeny číslem ve složené závorce. Funkce *format* zajistí odpovídající dosazení hodnot přesně v zadaném pořadí. Takže proměnná *id\_tuner* se dosadí na místo označené *{0}*, proměnná *band* se dosadí na místo *{1}*, atd. Jednotlivé proměnné jsou stejnojmenně definovány v nadřazené funkci uvozené *def w\_scan2*. Díky tomuto pokročilému řešení skládání textových řetězců není zapotřebí komplikovaně slučovat řetězce za použití operátorů *+*. Následuje sada menších podpůrných funkcí, které jsou popsány individuálně.

```
90 def add_freq(freq):
91     if freq not in frequencies:
92         frequencies.append(freq)
93
94 def SplitS(str):
95     return str.split('=')[1].strip()
96
97 def SplitS2(str):
98     return str.split(' ')[1].strip()
99
100 def sap_server(interface, sap_delay, user, type, name, IPAddress, port, playlist_group):
101     return {
102         "zahlatvi": ""
103         [general]
104         interface={0}
105         sap_delay={1}
106         "" .format(interface, sap_delay),
107         "telo": ""
108         [program]
109         type={0}
110         name={1}
111         address={2}
112         port={3}
113         user={4}
114         {5}
115         "" .format(type, name, IPAddress, port, user, playlist_group)
116     }
117
118 def dvblast_cmd(tuner, frequency, conf_path): #v základní hodnotě - např. 674000000
119     return "dvblast -a {0} -f {1} -m qam_auto --delsys DVBT2 --dvb-plp-id -1 -z -b 8 -e
-c {2}".format(tuner, frequency, conf_path)
```

Obrázek č. 34: Automatické přeladění – skript – podpůrné funkce I

(Zdroj: Vlastní zpracování)

Na obrázku č. 34 je definována procedura *add\_freq*, která zajišťuje vkládání unikátních hodnot frekvencí multiplexů DVB-T2 do globálního pole *frequencies*. Bude-li tato funkce zavolána víckrát s identickou vstupní hodnotou, bude tato hodnota uložena pouze jednou. Následující dvě funkce *SplitS* a *SplitS2* pouze vrací pole obsahující textové řetězce, které byly rozděleny na základě znaku jako oddělovače. První funkce rozděluje textové řetězce podle oddělovacího znaku rovná se (=), zatímco druhá funkce používá jako oddělovač znaky tečku s mezerou.

V pořadí čtvrtá funkce, dle obrázku č. 34, s názvem *sap\_server* je řešena analogicky jako funkce *w\_scan2*. V této funkci je uvedena základní struktura konfiguračního souboru pro SAP server, kdy hodnoty parametrů jsou dosazovány za pomoci dílčí funkce *format*.

Poslední funkce *dvblast\_cmd* je opět vyřešena stejným způsobem jako funkce *w\_scan2*. Tato funkce vrací příkaz pro správné spuštění nástroje DVblast.

```

67 def parseXML(file_path):
68     doc = xml.dom.minidom.parse(file_path)
69     track = []
70     track_info = {}
71     track_subinfo = {}
72     for ele in doc.getElementsByTagName("trackList")[0].getElementsByTagName("track"):
73         title = ele.getElementsByTagName("title")[0].firstChild.data
74         location = ele.getElementsByTagName("location")[0].firstChild.data
75         for ext in ele.getElementsByTagName("extension"):
76             bw = ext.getElementsByTagName("vlc:option")[0].firstChild.data
77             ts_id = ext.getElementsByTagName("vlc:option")[1].firstChild.data
78             program = ext.getElementsByTagName("vlc:option")[2].firstChild.data
79             vlc_id = ext.getElementsByTagName("vlc:id")[0].firstChild.data
80             frequency = int(SplitS(location))
81             track_subinfo = {"bandwidth": int(SplitS(bw)), "ts_id": int(SplitS(ts_id)), "vlc_id": int(vlc_id), "program": int(SplitS(program)) }
82             track_info = {"title": title, "location": location, "frequency": frequency, "extension": track_subinfo}
83             add_freq(frequency)
84             track.append(track_info)
85             track_info = {}
86             track_subinfo = {}
87     return track

```

Obrázek č. 35: Automatické přeladění – skript – funkce *parseXML*

(Zdroj: Vlastní zpracování)

Jednou ze stěžejních funkcí, nezbytných pro celý tento skript, je funkce *parseXML* (viz obrázek č. 35) s jedním vstupním parametrem, a to cestou k souboru *xspf*. Za použití knihovny Minimal DOM se do proměnné *doc* uložil strukturovaný objekt, ze kterého je možné získávat data za pomoci ukazatelů na názvy tagů v XML souboru *xspf*. Díky této knihovně lze snadno vybrat potřebné údaje a uložit je do strukturované dočasné proměnné *tracks*. Na obrázku č. 36 je vyobrazena první část této proměnné. Je zřejmé, že hierarchicky nejvýše se nachází standardní pole indexované čísly od nuly, přičemž každý prvek tohoto pole obsahuje datový typ *dictionary*, zkráceně jen *dict*, s podrobnými informacemi o každém TV programu. V souvislosti s touto přípravou dat jsou extrahovány některé parametry a následně převedeny na odpovídající datový typ. Příkladem je parametr *frequency* nebo *program* (SID). Při procházení zdrojového souboru *xspf* je volána zmíněná procedura *add\_freq*, protože se jedná o jediný vhodný

okamžik, kdy je zapotřebí sestavit pole všech unikátních dostupných multiplexů v síti DVB-T2.

```
tracks: [{ 'title': '0001. CT 1 HD T2', 'location': 'dvb-t2://frequen
> special variables
> function variables
00: { 'title': '0001. CT 1 HD T2', 'location': 'dvb-t2://frequency=5
> special variables
> function variables
'title': '0001. CT 1 HD T2'
'location': 'dvb-t2://frequency=514000000'
'frequency': 514000000
'extension': { 'bandwidth': 8, 'ts_id': 8465, 'vlc_id': 2, 'program
> special variables
> function variables
'bandwidth': 8
'ts_id': 8465
'vlc_id': 2
'program': 268
len(): 4
len(): 4
01: { 'title': '0002. CT 2 HD T2', 'location': 'dvb-t2://frequency=5
02: { 'title': '0003. CT 24 HD T2', 'location': 'dvb-t2://frequency=
03: { 'title': '0004. CT sport HD T2', 'location': 'dvb-t2://frequen
04: { 'title': '0005. CT D/art HD T2', 'location': 'dvb-t2://frequen
```

Obrázek č. 36: Automatické přeladění – skript – struktura dočasné proměnné *tracks* (Zdroj: Vlastní zpracování)

Pro čtení souboru volitelných přenosů formátu XML je použita funkce uvedena na obrázku č. 37.

```
def parseXML_my_streams(file_path):
    doc = xml.dom.minidom.parse(file_path)
    streams = {
        "m3u8": "",
        "sap": ""
    }
    m3u8 = ""
    sap = ""
    for ele in doc.getElementsByTagName("mystreams")[0].getElementsByTagName("stream"):
        title = ele.getElementsByTagName("title")[0].firstChild.data
        protocol = ele.getElementsByTagName("protocol")[0].firstChild.data
        address = ele.getElementsByTagName("address")[0].firstChild.data
        port = ele.getElementsByTagName("port")[0].firstChild.data
        if ele.getElementsByTagName("company")[0].firstChild:
            company = ele.getElementsByTagName("company")[0].firstChild.data
        else:
            company = settings["SAP_server"]["user"]

        if ele.getElementsByTagName("group")[0].firstChild:
            group = "playlist_group=" + ele.getElementsByTagName("group")[0].firstChild.data
        else:
            group = ""
        m3u8 = m3u8 + "#EXTINF:-1," + title + "\n" + protocol + "://" + address + ":" + port + "\n"
        sap = sap + sap_server("","company,protocol,title,address,port,group){"telo"}
    streams["m3u8"] = m3u8
    streams["sap"] = sap
    return streams
```

Obrázek č. 37: Automatické přeladění – skript – funkce *parseXML\_my\_streams* (Zdroj: Vlastní zpracování)

Tato funkce na výstupu vrací datový typ *Dict* obsahující doplňující textové řetězce pro soubor M3U8 a SAP server. Využití těchto řetězců se rozhoduje až v proceduře *prepare\_files*.

Následují čtyři podpůrné funkce, jež mají za úkol spravovat soubory viz obrázek č. 38.

```
120 def modify_file(file_path, content):
121     f = open(file_path, "w+")
122     f.write(content)
123     f.close()
124
125 def delete_file(file_path):
126     os.remove(file_path)
127
128 def delete_files2(folder):
129     i=0
130     for filename in os.listdir(folder):
131         file_path = os.path.join(folder, filename)
132         try:
133             if os.path.isfile(file_path) or os.path.islink(file_path):
134                 os.unlink(file_path)
135             elif os.path.isdir(file_path):
136                 shutil.rmtree(file_path)
137         except Exception as e:
138             return [False, "Failed to delete"]
139         i=i+1
140     return [True, "Úspěšně vymazáno", i]
141
142 def delete_files_general():
143     ret2 = True
144     if (settings["DVBlast"]["conf_files_dir"] == settings["DVBlast"]["cmd_files_dir"]):
145         ret = delete_files2(settings["DVBlast"]["conf_files_dir"])
146     else:
147         ret2 = delete_files2(settings["DVBlast"]["cmd_files_dir"])
148         ret = delete_files2(settings["DVBlast"]["conf_files_dir"])
149     if ret[0] == False:
150         return [False, ret[1], settings["DVBlast"]["conf_files_dir"]]
151     elif isinstance(ret2, bool) == False:
152         return [False, ret2[1], settings["DVBlast"]["conf_files_dir"]]
153     else:
154         return [True, ret[1], ret[2]]
155
156 def prepare_fn():
157     for frequency in frequencies:
158         file_names["config"][str(frequency)] = file_names["DVBlast"]["config_file_name"] + str
159         (frequency) + file_names["DVBlast"]["config_file_ext"]
160         file_names["run"][str(frequency)] = file_names["DVBlast"]["cmd_file_name"] + str
161         (frequency) + file_names["DVBlast"]["cmd_file_ext"]
```

Obrázek č. 38: Automatické přeladění – skript – podpůrné funkce II (správa souborů)  
(Zdroj: Vlastní zpracování)

První funkce *modify\_file* funguje dvojím způsobem. Buď vytvoří nový soubor a vloží do něho požadovaný text, pokud neexistuje, nebo přepíše existující soubor novým textem. Pokud by bylo žádoucí přidávat další obsah do souboru, bylo by zapotřebí změnit druhý argument v rámci dílčí funkce *open* z *w+* na hodnotu *a*.

Druhá funkce *delete\_file* pouze vymaže požadovaný soubor prostřednictvím knihovny *os*.

Třetí funkce *delete\_files2* má za úlohu provést vymazání všech souborů v zadané složce. Pokud se vymazání zdaří, do návratové hodnoty se připojí celkový počet vymazaných souborů.

Předposlední funkce *delete\_files\_general* je nadřazená funkce výše zmíněným funkcím. Jejím úkolem je zjistit, jak je nakonfigurován DVBlast z hlediska proměnné *settings*. Přesněji se porovnává, zda je stejně definovaná cesta ke spouštěcím skriptům i ke

konfiguračním souborům. Pokud jsou obě cesty stejné, pak dojde k promazání pouze daného adresáře. V opačném případě je vymazán obsah obou adresářů. Tato funkce poskytuje návratové hodnoty v poli a informuje o výsledném stavu, zda se podařilo soubory vymazat či nikoli.

Poslední funkce *prepare\_fn* zkráceně vyjadřuje prepare file names. Jak je z tohoto názvu na první pohled patrné, jedná se o přípravu názvů souborů pro nástroj DVBlasť. Tyto názvy se sestavují na základě uživatelem definovaných preferencí v proměnné *settings*.

Na obrázku č. 39 jsou uvedeny další dvě podpůrné funkce, jež jsou pro správnou funkcionalitu nezbytné.

```
169 def extract_freq(input_string):
170     start_index = input_string.find('_') + 1
171     end_index = input_string.find('.')
172
173     if start_index == 0 or end_index == -1:
174         return [-1]
175
176     extracted_part = input_string[start_index:end_index]
177     try:
178         cislo = int(extracted_part)
179         return [1, cislo]
180     except ValueError:
181         return [0]
182
183 def compare_arrays(old, actual):
184     set1 = set(old)
185     set2 = set(actual)
186     unique_in_arr1 = set1 - set2
187     unique_in_arr2 = set2 - set1
188
189     if unique_in_arr2:
190         response = [False, "V poli starých frekvencí chybí níže uvedené hodnoty, které byly nově nalezeny", unique_in_arr2]
191     elif unique_in_arr1:
192         response = [False, "V poli aktuálně nalezených frekvencí se nenachází níže uvedené staré frekvence", unique_in_arr1]
193     else:
194         response = [True, "Obě pole jsou stejná."]
195
196     return response
```

**Obrázek č. 39: Automatické přeladění – skript – podpůrné funkce III**

(Zdroj: Vlastní zpracování)

První funkce *extract\_freq* se používá pro extrakci čísla frekvence z názvů spustitelných skriptů nástroje DVBlasť. Je zde uplatněn zpětný mechanismus, aby bylo možné při budoucím spuštění skriptu srovnat staré a nové počty multiplexů. Tento mechanismus je jednodušší varianta k udržování starých hodnot frekvencí. Jinak by bylo nutné vždy mít v textovém souboru uložené zmíněné frekvence multiplexů. Celá funkce pracuje s jednou vstupní hodnotou, kterou je cesta k adresáři. Následně funkce v tomto textovém řetězci vyhledá dva klíčové znaky, kterými jsou podtržítka a tečka. Výsledný textový řetězec, mezi těmito dvěma znaky, se funkce pokusí převést z textu na celé číslo. Pokud se extrakce, resp. přečtení čísla, úspěšně zdařila, je vrácena hodnota *True*. Ve všech ostatních případech je vráceno *False*.



Druhá podpůrná funkce `compare_arrays` má za úkol vzájemně srovnat čísla ve dvou vstupních polích. Principem je, že se nejprve převedou vstupní pole na množiny a provede se operace odčítání. Tato operace se provede dvakrát, přičemž se nejprve odečte jedna množina od druhé a poté se odčítání opakuje v opačném pořadí. Cílem je identifikovat čísla, která jsou odlišná, ať už z hlediska celkového počtu, tak i z hlediska své velikosti.

```
198 def calculate_sha256(file_path):
199     sha256_hash = hashlib.sha256()
200     with open(file_path, "rb") as file:
201         for byte_block in iter(lambda: file.read(4096), b''):
202             sha256_hash.update(byte_block)
203     return sha256_hash.hexdigest()
```

Obrázek č. 40: Automatické přeladění – skript – podpůrná funkce `calculate_sha256`  
(Zdroj: Vlastní zpracování)

Zcela poslední podpůrnou funkcí, uvedenou na obrázku č. 40, je výpočet kontrolního součtu souboru SHA256. Úkolem této funkce je vrátit odpovídající otisk souboru v podobě tzv. hashe.

Pro zachování konvence jsou všechny návratové hodnoty, v případě jejich vyšších počtů, řešeny pomocí datového typu pole. Logika spočívá v tom, že nultý prvek pole má boolovskou hodnotu určující úspěšnost výsledku funkce. To znamená, že pokud je hodnota zmíněného prvku `True`, pak lze očekávat, že budou definovány i další položky pole obsahující upřesňující výsledky příslušné funkce.

Díky všem výše uvedeným podpůrným funkcím a procedurám je možné představit hlavní funkci `prepare_files`, která zajišťuje majoritní část tohoto skriptu. Zmíněná funkce je vyobrazena na obrázku č. 41 po menších částech s doprovodným popisem.

```
209 def prepare_files(source_path):
210     tracks = parseXML(source_path)
211     prepare_fn()
212     old_freq = []
213     browsed = browse_files(settings["DVblast"]["cmd_files_dir"])
214     for item in browsed:
215         extracted_freq = extract_freq(item)
216         if extracted_freq[0] == 1:
217             if extracted_freq[1] not in old_freq:
218                 old_freq.append(extracted_freq[1])
219         else:
220             print("Chyba při procházení adresáře. Nepodařilo se rozpoznat hodnotu frekvence...")
221             sys.exit()
222     compare = compare_arrays(old_freq, frequencies)
223     print(compare[1])
224     if compare[0] == False:
225         send_email()
226     print(f"položky: {' ', '.join(map(str, compare[2]))}")
227
228     dfg = delete_files_general()
229     if dfg[0] == False:
230         print("Chyba při mazání souborů. Důvod: " + dfg[1] + " | přesněji: " + dfg[2])
231         print("KONEC")
232         send_email()
233         sys.exit()
234     else:
235         print("Počet vymazaných souborů: " + str(len(dfg[2])))
```

Obrázek č. 41: Automatické přeladění – skript – funkce `prepare_files I`  
(Zdroj: Vlastní zpracování)

Funkce začíná spuštěním dílčí funkce *parseXML* za účelem poskytnutí seznamu TV programů a rozhlasových stanic se souvisejícími informacemi a také vytvoření aktuálního seznamu dostupných frekvencí multiplexů. V dalším kroku se sestaví všechny názvy souborů pomocí procedury *prepare\_fn* a uloží se do globální proměnné *file\_names*. Následuje prohledání adresáře funkcí *browse\_files*, ve kterém by se měly podle nastavení nacházet spouštěcí soubory DVBlastu. Z názvů těchto souborů se extrahují čísla frekvencí, které se následně prostřednictvím funkce *compare\_arrays* srovnají s nově zjištěnými frekvencemi multiplexů. Pokud se zjistí, že staré frekvence neodpovídají novým, pak se upozorní správce e-mailem, protože této nové nebo změněné frekvenci musí přizpůsobit nastavení systémové služby operačního systému.

Vzhledem ke komplikovanému procesu spjatém se spuštěním a ukončováním provozu multiplexů je nepravděpodobné, že by k těmto událostem v praxi docházelo často. To znamená, že stane-li se jakákoli změna frekvence multiplexu, rozhodně to nebude častěji než jednou za rok a z tohoto důvodu není zapotřebí tuto činnost automatizovat.

V další fázi funkce *prepare\_files* pokračuje pokusem o vyprázdnění adresářů souvisejících s DVBlastem. Pokud se úspěšně nepodaří soubory vymazat, je ukončen celý skript, protože není možné pokračovat. I v tomto případě je o této závažné skutečnosti informován správce.

```

237 i = settings["DVBlast"]["ipv4_conf"]["start_IP"]
238 out = ""
239 out2 = ""
240 sap = ""
241 sap_i = 0
242 m3u8 = "#EXTM3U\n"
243 tuner_id = 0
244 tuner_id_max = 0
245 playlist_group = ""
246 for freq in frequencies:
247     for track in tracks:
248         if track["frequency"] != freq: continue
249         ipv4 = settings["DVBlast"]["ipv4_conf"]["base_IP"] + str(i)
250         port = str(settings["DVBlast"]["ipv4_conf"]["port"])
251         out = out + ipv4 + ":" + port + " 1 " + str(track["extension"]["program"]) + " \t# " + SplitS2(track
["title"]) + "\n"
252         #---SAP---
253         if sap_i == 0:
254             sap = sap_server(settings["SAP_server"]["interface"], settings["SAP_server"]["sap_delay"], settings
["SAP_server"]["user"], settings["SAP_server"]["type"], SplitS2(track["title"]), ipv4, port, "")
255             sap = sap["zhlavi"] + "\n"
256             if settings["SAP_server"]["tridit_tv_radia"] == True:
257                 if (track["extension"]["program"] > settings["SAP_server"]["tridit_tv_radia_meze"]["od"]) and (track
["extension"]["program"] < settings["SAP_server"]["tridit_tv_radia_meze"]["do"]): #pak jde o rádio
258                     playlist_group = "playlist_group=Radia"
259                 else: #jde o TV
260                     playlist_group = "playlist_group=Televize"
261             sap = sap + sap_server(settings["SAP_server"]["interface"], settings["SAP_server"]["sap_delay"],
settings["SAP_server"]["user"], settings["SAP_server"]["type"], SplitS2(track["title"]), ipv4, port,
playlist_group)["telo"]
262             sap_i = sap_i + 1
263         #---SAP---

```

Obrázek č. 42: Automatické přeladění – skript – funkce *prepare\_files II*  
(Zdroj: Vlastní zpracování)

Druhá část funkce *prepare\_files*, která je uvedena na obrázku č. 42, na začátku vyobrazuje definici pomocných proměnných. V tomto případě nebyl využit žádný strukturovaný datový typ, protože se jedná pouze o dočasné lokální, nikoli globální proměnné. Následně jsou spuštěny dva klíčové cykly *for* s konečným počtem opakování. První cyklus prochází všechny aktuálně nalezené frekvence a vnořený cyklus postupně prochází všechny TV programy a rozhlasové stanice. Díky těmto dvěma cyklům se už jen v přímé závislosti na nastavení provádí příprava všech potřebných komponent zahrnujících konfigurační soubory DVBlastu, SAP serveru i seznamu TV programů ve formátu M3U8. Pro úplnost popisované funkce je na obrázku č. 43 uvedena její poslední, tj. třetí, část.

```

264     #--m3u8--
265     if settings["playlist_m3u8"]["create"] == True:
266         m3u8 = m3u8 + "#EXTINF:-1," + SplitS2(track["title"]) + "\n" + settings["playlist_m3u8"]
           ["protocol"] + "://" + ipv4 + ":" + port + "\n"
267     #--m3u8--
268     i = i + settings["DVBlast"]["ipv4_conf"]["inkrement"]
269     file_name_conf = settings["DVBlast"]["conf_files_dir"] + file_names["config"][str(freq)]
270     modify_file(file_name_conf, out)
271     out = ""
272     #second: create executable .sh files
273     if tuner_id >= len(settings["DVBlast"]["id_tuner"]):
274         tuner_id_max = settings["DVBlast"]["id_tuner_default"]
275     else:
276         tuner_id_max = settings["DVBlast"]["id_tuner"][tuner_id]
277     out2 = dvblast_cmd(tuner_id_max, freq, file_name_conf)
278     modify_file(settings["DVBlast"]["cmd_files_dir"] + file_names["run"][str(freq)], out2)
279     tuner_id = tuner_id + 1
280
281     if settings["SAP_server"]["create"]:
282         modify_file(settings["SAP_server"]["dir"] + settings["SAP_server"]["filename_cfg"], sap)
283     if settings["playlist_m3u8"]["create"]:
284         modify_file(settings["playlist_m3u8"]["dir"] + settings["playlist_m3u8"]["filename"], m3u8)

```

Obrázek č. 43: Automatické přeladění – skript – funkce *prepare\_files* III  
(Zdroj: Vlastní zpracování)

Na obrázku č. 43 jsou na konci cyklů vidět použitá počítadla, která byla využívána pro inkrementování jak identifikátorů DVB tunerů, tak i počítadlo IPv4 adres. Dále je patrné, že se soubory pro DVBlast generují vždy bez možnosti nastavení. Volitelné generování výstupních souborů má jen SAP server a formát M3U8 (parametr *create* v *settings*).

Obrázek č. 44 uvádí poslední snímek první části závěru celého skriptu, který již není definován funkcí ani procedurou, protože se neočekává další rozsáhlé rozšiřování nebo jiné marginální editace skriptu.

```

328  checksum = {}
329  xspf_path_temp = settings["scan"]["dir"] + settings["scan"]["filename_temp"]
330  xspf_path = settings["scan"]["dir"] + settings["scan"]["filename"]
331  xspf_path_select = ""
332
333  if len(sys.argv) == 2:
334      if sys.argv[1] == "manual_update":
335          if os.path.exists(xspf_path):
336              prepare_files(xspf_path)
337              print("Manuální aktualizace provedena.")
338          else:
339              print("Zdrojový soubor " + xspf_path + " neexistuje. Nelze pokračovat.")
340              print("KONEC")
341              sys.exit()
342
343  if os.path.exists(xspf_path):
344      xspf_path_select = xspf_path_temp
345  else:
346      xspf_path_select = xspf_path

```

**Obrázek č. 44: Automatické přeladění závěr skriptu I**

(Zdroj: Vlastní zpracování)

Nejprve byly, v rámci skriptu, definovány funkce a procedury, takže nyní lze přistoupit k popsání hlavního kódu. Tento kód začíná definicí proměnné *checksum*, dat. typu *Dict*, určenou pro uložení vypočítané hodnoty kontrolního součtu SHA256 výstupního souboru nástroje *w\_scan2*. Dále jsou připraveny obě cesty k souborům *xspf*, přičemž jeden soubor z nich je pouze dočasný. Je-li skript spuštěn poprvé, použití dočasného souboru se neuvažuje.

Návrh obsahuje požadavek na manuální spuštění skriptu, které uvede v platnost aktualizaci souboru volitelných přenosů. K tomuto účelu je využito proměnné *sys.argv*, která obsahuje všechny vstupní argumenty, uplatněné při spouštění, v datové struktuře typu pole. Pro spuštění manuální aktualizace je striktní podmínkou, že argument smí být pouze jeden v této podobě *manual\_update*. Jakýkoli dodatečný nebo chybný argument bude ignorován. Následně dojde k běžnému spuštění skriptu, který jinak žádný vstupní argument nevyžaduje. Jakmile se ověří náležitosti spjaté se vstupním argumentem, dojde ke kontrole dostupnosti dříve vytvořeného souboru *xspf*. Selže-li tato kontrola, pak je nutné nejprve spustit skript výchozím způsobem, aby byly vytvořeny všechny související komponenty. Až po prvním spuštění skriptu je možné používat argument *manual\_update*.

```

348 w_scan2_cmd = w_scan2(settings["scan"]["id_tuner"], settings["scan"]["band"], settings["scan"]["country"],
349 settings["scan"]["free_ch"], xspf_path_select)
350 os.system(w_scan2_cmd)
351
352 if (os.path.exists(xspf_path_select)):
353     if (os.stat(xspf_path_select).st_size == 0):
354         print("w_scan2 selhal. Je připojený zadaný adaptér: /dev/dvb/adapter" + settings["scan"]
355             ["id_tuner"] + "?")
356         send_email()
357         print("KONEC")
358         sys.exit()
359
360 if (xspf_path_select == xspf_path_temp):
361     chksum[xspf_path] = calculate_sha256(xspf_path)
362     chksum[xspf_path_temp] = calculate_sha256(xspf_path_temp)
363     if chksum[xspf_path] == chksum[xspf_path_temp]:
364         delete_file(xspf_path_temp)
365         print("Nově vygenerovaný soubor: " + xspf_path_temp + " není potřeba, protože je stejný jako
366             ten starý. Mažu.")
367     else:
368         prepare_files(xspf_path_select)
369         delete_file(xspf_path)
370         os.rename(xspf_path_temp, xspf_path)
371         print("Starý soubor xspf: " + xspf_path + " byl smazán a nový temp soubor: " + xspf_path_temp
372             + " byl přejmenován zpět na starý.")
373 else:
374     prepare_files(xspf_path_select)
375     print("Dokončeno")
376 else:
377     print("CHYBA: Nástroj w_scan2 nevygeneroval soubor: " + xspf_path_select)
378     print("KONEC")

```

**Obrázek č. 45: Automatické přeladění závěr skriptu II**

(Zdroj: Vlastní zpracování)

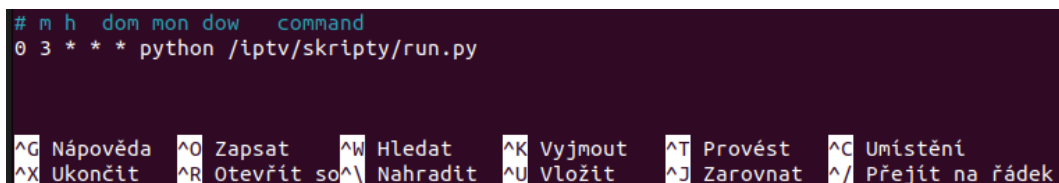
V navazujícím kroku je pomocí funkce `w_scan2` a s využitím hodnot v nastavení, sestaven spouštěcí příkaz pro stejnojmenný nástroj, viz obr. č. 45. Tento nástroj je dále spuštěn bez zachytávání jeho výstupu. Vychází se z předpokladu, že v případě chyby `w_scan2` vytvoří nový, leč prázdný soubor. Na základě tohoto předpokladu se vždy zjišťuje, zda soubor nemá nulovou kapacitu. Pokud je nulová kapacita zjištěna, je upozorněn správce a celý skript je ukončen, protože se jedná o závažnou chybu, kvůli které by nemohlo pokračovat následné zpracování. Je-li vše v pořádku, pak se, v případě existence základního `xspf` souboru, vypočítá kontrolní součet obou `xspf` souborů (základního i dočasného). Vyjde-li rovnost těchto součtů, pak lze jednoznačně prohlásit, že jsou oba soubory absolutně shodné. Z toho vyplývá, že nový výsledek skenování dopadl stejně jako ten předchozí, a proto lze nový dočasný soubor odstranit. Jsou-li oba výsledky shodné, pak není zapotřebí znovu vytvářet již existující konfigurační ani jiné soubory a skript může být ukončen. Nejsou-li kontrolní součty shodné, pak dojde ke spuštění procedury `prepare_files`, která zajistí vše potřebné. Následně se odstraní starý `xspf` soubor a nahradí se novým, resp. dočasným souborem. Tím končí proces Automatického přeladění.

## Nedostatky při realizaci

Toto softwarové řešení předpokládá používání zkušeným odborně vzdělaným správcem, kterým je zaměstnanec společnosti ABC, s.r.o. Za tohoto předpokladu bylo řešení realizováno, a tudíž nebylo nutno při realizaci zohledňovat některé kontrolní mechanismy ve zdrojovém kódu. V případě obsluhy nekvalifikovaným správcem by mohlo dojít k vyvolání neošetřené výjimky.

### 3.4.7 Automatizace automatického přeladění

Automatické přeladování bylo vyřešeno v předchozí podkapitole, a proto stačí provést jen nastavení nástroje Cron v OS Ubuntu. Zjednodušeně lze Cron přirovnat k plánovači úloh (známého z OS Windows). Je možné přesně vymezit časové okamžiky, kdy má docházet ke spuštění libovolných procesů. Konkrétní nastavení Cronu, spuštěného příkazem *crontab -e* je uvedeno níže.



```
# m h dom mon dow  command
0 3 * * * python /iptv/skripty/run.py
```

Obrázek č. 46: Vyobrazení části Cron tabulky v textovém editoru Nano  
(Zdroj: Vlastní zpracování)

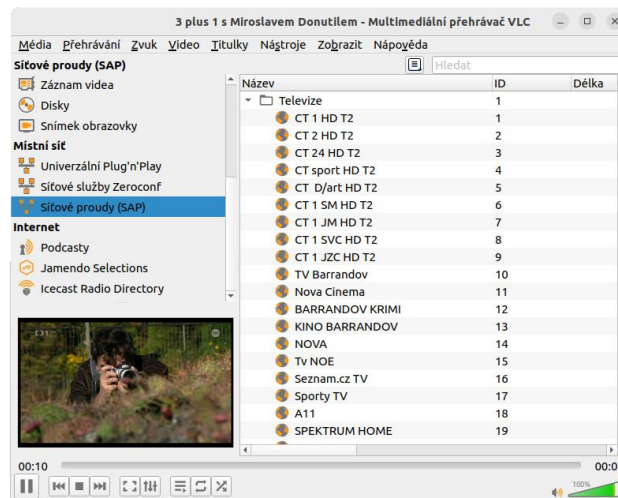
V tomto odstavci je postupně rozložen příkaz v Cron tabulce uvedený bílým fontem v obrázku č. 46. První nula znamená, že spuštění nastane každou celou minutu. Tuto skutečnost upravuje následující hodnota 3. To znamená, že skript pro automatické přeladění bude pravidelně spuštěn každý den ve tři hodiny ráno. Je-li skenování vyhrazen jeden tuner, není omezen divácký komfort, a přesto je průběžně aktualizována kompletní programová nabídka.

## 3.5 Přijímání IPTV

Dosud byl popsán způsob získání, zpracování a přenos signálu ze sítě DVB-T2 do IP sítě. Pro úplnost této práce je doplněna ještě tato závěrečná podkapitola, jež pojednává o možnostech příjmu IPTV. Příjem byl testován pomocí aplikací VideoLAN VLC a Kodi. Níže jsou obě varianty blíže specifikovány s doprovodnými obrázky, ze kterých je lépe patrné praktické využití.

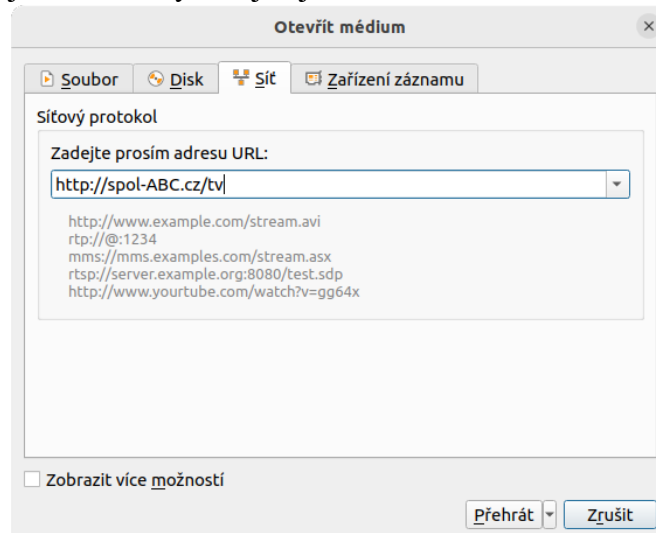
### 3.5.1 Aplikace

Jak již bylo uvedeno v teoretických východiscích, uživatelé počítačů mohou pro příjem výše popsané IPTV použít přehrávač VLC.



Obrázek č. 47: Snímek aplikace VLC – příjem vícesměrového vysílání TV programu ČT1 pomocí protokolu SAP  
(Zdroj: Vlastní zpracování)

Na obrázku č. 47 je vyobrazena aplikace VLC, která získala informace o datových proudech pomocí protokolu SAP. Z uvedeného je patrné, že přehrávání je zcela funkční, a to včetně doprovodných EPG dat. Výhodou této aplikace je také to, že takzvaně následuje odkazy, jsou-li zadány tak, jak je uvedeno na obrázku č. 48.



Obrázek č. 48: Snímek aplikace VLC – zadání URL  
(Zdroj: Vlastní zpracování)

Společnost ABC, s.r.o. může mít spuštěný webový server s aktivovaným přesměrováním nebo aliasem. Odkaz pak může být velmi krátký, kdy webový server poskytne aplikaci VLC plnohodnotnou URL adresu ke zdrojovému souboru M3U8 a zahájí se přehrávání.

Tato funkce je velmi praktická, protože si uživatel nemusí pamatovat dlouhé URL adresy. Funkcí následování odkazů však disponuje velmi málo přehrávačů. Z tohoto důvodu si bude muset každý klient ručně stáhnout playlist v souboru M3U8 pomocí webového prohlížeče a následně jej importovat do příslušného přehrávače či jiného multimediálního zařízení.

Další možností příjmu je populární multimediální aplikace Kodi. Kodi je open-source multimediální centrum, které umožňuje přehrávání široké škály obsahu, včetně videí, hudby a obrázků přímo na televizoru, počítači nebo jiném zařízení. Konkrétně Kodi podporuje tyto platformy:

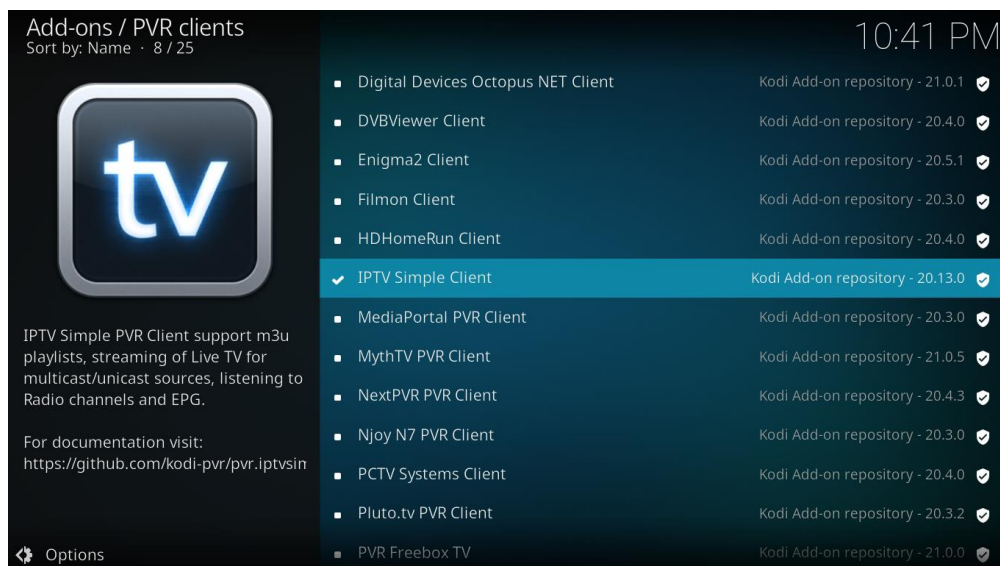
- Microsoft Windows,
- MacOS, iOS, tvOS,
- Linux,
- Android,
- webOS,
- Raspberry Pi. (33)

Kodi je vybaveno přehledným uživatelským rozhráním, které je optimalizováno pro pohodlné ovládání. Uživatelé mohou pomocí doplňků, které jsou k dispozici přes oficiální Kodi knihovnu, rozšiřovat jeho funkcionality. (33)

Kodi je licencováno pod GNU General Public License (GPL), verze 2. Tato licence umožňuje uživatelům svobodně modifikovat a distribuovat software, avšak jakékoliv modifikace nebo rozšíření musí být také zpřístupněny pod stejnou licenci. (33)

Ve zmíněné knihovně doplňků lze najít např. doplněk *IPTV Simple Client*, takže není nutné nic stahovat ani instalovat ze zdrojů třetích stran. Tento doplněk nabízí možnost importování playlistu ve formátu M3U8, což je důvod, proč byl zahrnut do procesu Automatického přeladění. Obrázek č. 49 vyobrazuje konkrétní doplněk ve zmíněné knihovně.





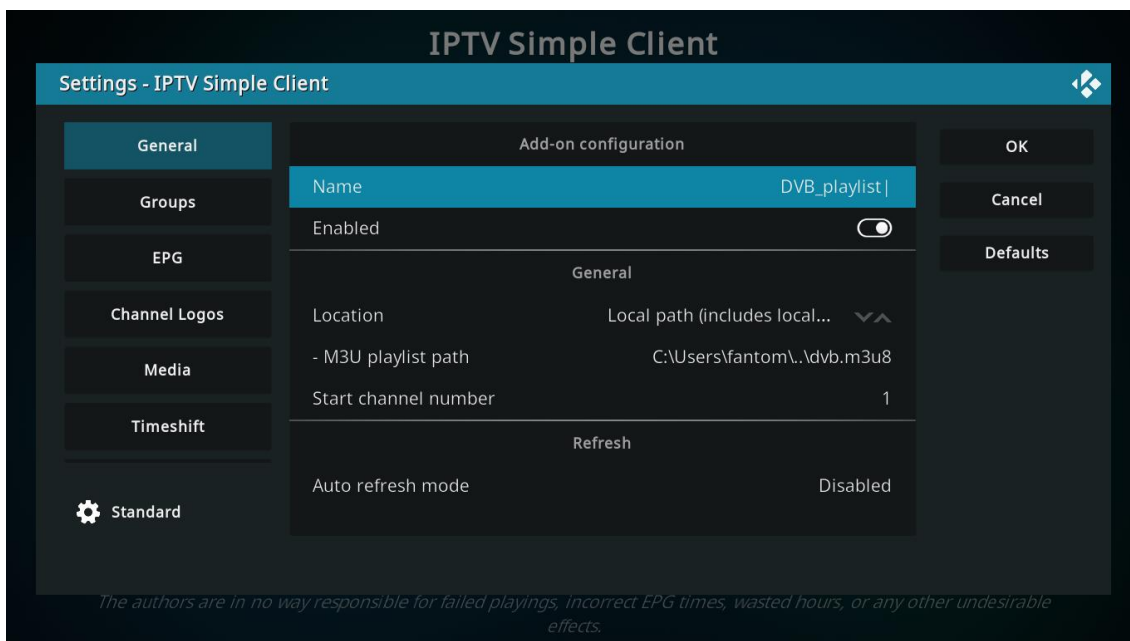
**Obrázek č. 49: Snímek z aplikace Kodi – knihovna**  
(Zdroj: Vlastní zpracování)

Po stažení a nainstalování je doplněk vybaven níže vyobrazeným spravovatelným rozhraním, ve kterém lze zkontrolovat verzi doplňku, odstranit jej, deaktivovat a provádět další operace.



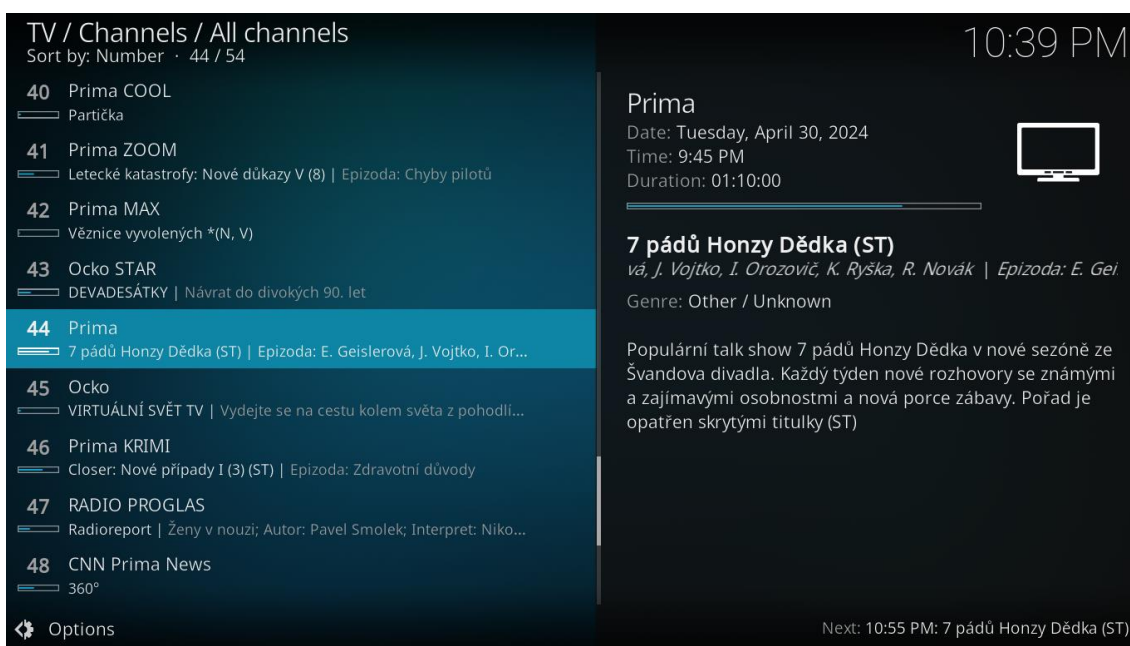
**Obrázek č. 50: Snímek z aplikace Kodi – podrobnosti doplňku IPTV Simple Client**  
(Zdroj: Vlastní zpracování)

Stažený playlist z webové stránky společnosti stačí snadno importovat v nastavení tohoto pluginu – viz obrázek č. 51.



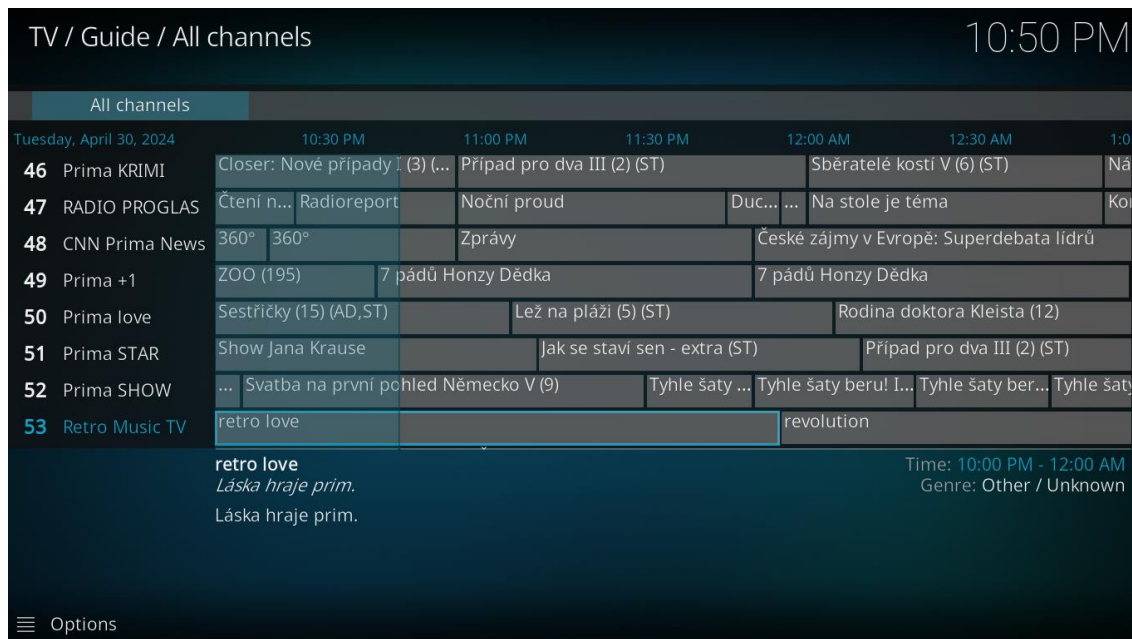
**Obrázek č. 51: Snímek z aplikace Kodi – nastavení doplňku IPTV Simple Client**  
(Zdroj: Vlastní zpracování)

Jakmile je nastavení dokončeno, lze přejít k sledování televize. Následující obrázek č. 52 uvádí část programové nabídky včetně EPG dat.



**Obrázek č. 52: Snímek z aplikace Kodi – programová nabídka**  
(Zdroj: Vlastní zpracování)

Kodi umožňuje vyobrazit podrobné informace EPG dat vybraných TV programů, což je zřejmé na dalším obrázku č. 53.



**Obrázek č. 53: Snímek z aplikace Kodi – podrobné informace z EPG některých TV programů**  
(Zdroj: Vlastní zpracování)

Výše je vysvětleno, jak je možné přijímat navrženou IPTV pomocí aplikace Kodi, která je díky své flexibilitě vhodná pro toto použití.

### 3.5.2 Přijímací zařízení

V dnešní době jsou na trhu televize s různými operačními systémy. Například značka LG používá webOS, značka Samsung využívá TizenOS. Jak již bylo uvedeno výše, Kodi má podporu pro webOS, takže uživatelé značky LG by neměli mít problém IPTV společnosti ABC, s.r.o. sledovat. Potíže by neměly mít ani uživatelé televizí s operačním systémem Android. Dále je uvedeno jedno z levnějších multimediálních zařízení podporující IPTV, které mohou uživatelé používat jako přijímač. Cena tohoto zařízení je 1900 Kč včetně DPH.



**Obrázek č. 54: Vzhled zařízení STRONG Android box SRT LEAP-S3+**  
(Zdroj: 34)

Poslední zajímavou alternativou příjmu IPTV je Raspberry Pi (zkráceně jen RPI). Toto zařízení je malý jednodeskový počítač, který byl vyvinut v Británii společností Raspberry Pi Foundation. RPI bylo původně navrženo pro vzdělávací účely, ale jeho vysoká přizpůsobitelnost a nízké náklady z něj učinily oblíbenou volbu pro různé hobby projekty nebo multimediální centra. Pro toto zařízení existuje nenáročný OS LibreELEC, jež je vybaven aplikací Kodi. Nespornou výhodou RPI je mimo jiné podpora funkce HDMI-CEC (Consumer Electronics Control). Tato funkce zajišťuje aktivní komunikaci mezi RPI a televizí, avšak za předpokladu, že příslušná televize rovněž HDMI-CEC podporuje. Díky této funkci je tak možné používat pro ovládání obou zařízení (RPI a TV) jeden originální dálkový ovladač televize. Jedná se tak o velmi pohodlné řešení pro diváky. Tato výhoda je částečně vyvážena menší nevýhodou, protože je zapotřebí stáhnout daný OS a vložit jej na microSD kartu pro RPI. Toto zařízení není vhodné pro většinu klientů, protože vyžaduje po uživateli odpovídající znalosti. RPI se v současné době prodává v několika různých variantách a velikostech. Vzhledem k tomu, že v DVB-T2 je používán kodek H265, který může být na takové jednodeskové počítače příliš náročný, je vhodné zvolit alespoň RPI 4B. Běžný spotřebitel může při nákupu zvolit některou z připravených sad prodejců, která již obsahuje všechny součásti, jež jsou pro provoz RPI nezbytné. Pro srovnání s Android zařízením značky Strong, je níže vyobrazena jedna zmíněná sada s RPI, vhodná pro jednoduché složení a uvedení do provozu.



**Obrázek č. 55: Vyobrazení součástí oficiální sady s Raspberry Pi 4B – 2GB**  
(Zdroj: 35)

Na obrázku č. 55 je vyobrazena tzv. startovací sada, která již obsahuje krabičku, napájecí zdroj, SD kartu a HDMI kabel. Cena této sady v květnu 2024 činí 2 151 Kč včetně DPH u dodavatele RPIshop. Toto RPI má dostatečný výkon pro dekodování kodeku H265 při plném HD rozlišení.

### **3.6 Přehled finančních nákladů**

Všechny použité softwarové prostředky a nástroje jsou open-source, a tudíž bezplatné. Náklady je zapotřebí zohlednit především z hlediska času k návrhu, vývoji a testování kódu, využívaného k automatickému přeladění programů. Dále se musí počítat s rizikem, že se vyskytne nepředpokládaná chyba ve zmíněném kódu a že bude potřeba provést dodatečné úpravy. Pro realizaci tohoto řešení byl využit hardware, zahrnující server s rozšiřující PCIe televizní kartou. Tabulka č. 7 uvádí přehled jednotlivých nákladů, jež je nutné vynaložit, aby celé řešení mohlo být funkční jak po softwarové, tak po hardwarové stránce.

**Tabulka č. 7: Přehled nákladů na softwarové řešení**  
(Zdroj: Vlastní zpracování)

Náklad	Pracnost (MD)	Cena (Kč)
Návrh řešení	6	42 000,-
Realizace řešení	8	56 000,-
Testování	4	28 000,-
Oprava chyb	3	21 000,-
Vytvoření dokumentace	2	14 000,-
<b>Celkem</b>	<b>23</b>	<b>161 000,-</b>

Pokud se určí částka 7000 Kč za 1 MD, pak celkové náklady na realizaci softwarového řešení činí 161 000 Kč, což odpovídá 23 MD. Toto řešení uvažovalo použití konkrétního hardwaru. Všechny použité součásti jsou uvedeny v tabulce č. 8.

**Tabulka č. 8: Přehled nákladů na hardwarové vybavení**  
(Zdroj: Vlastní zpracování)

Položka	Množství (ks)	Cena za kus (Kč)	Cena celkem (Kč)
HW – server 1U (36) <i>Dell PowerEdge R250</i>	2	21 983,-	43 966,-
HW – PCIe TV karta (30) <i>TBS6205 DVB-T2/T/C Quad TV Tuner</i>	2	5 777,-	11 554,-
HW – USB TV karta <i>Evolveo Sigma T2</i>	2	850,-	1 700,-
<b>Celkem</b>	<b>6</b>	<b>28 610,-</b>	<b>57 220,-</b>

Všechny uvedené ceny jsou bez DPH. Celkové náklady na toto řešení, včetně hardwaru a softwaru, činí 218 220 Kč.

### 3.7 Přínosy navrženého řešení

Předložené softwarové řešení v diplomové práci umožňuje společnosti ABC, s.r.o. poskytovat novou službu IPTV spolehlivě a kvalitně zejména díky záložnímu řešení. Dále je vyřešena i otázka automatizace programové nabídky pomocí skriptu, což výrazně snižuje režie zaměstnance společnosti, který by jinak musel ručně provádět změny v konfiguračních souborech. Zmíněné režie zaměstnance jsou blíže vysvětleny v následujícím odstavci.

Nástroj *w\_scan2* vygeneruje vždy tentýž požadovaný strukturovaný soubor obsahující aktuální programovou nabídku. Tato nabídka v celkovém součtu obvykle sestává z přibližně 60 TV programů a 10 rozhlasových stanic. Každý program a stanice má svůj identifikátor (SID). Tento identifikátor je nezbytné uvádět v konfiguračních souborech DVBlastu. Ze strany poskytovatelů multiplexů v síti DVB-T2 občas dochází k nepatrným úpravám programové nabídky. To například může znamenat, že se spustí jeden či dva nové TV programy. Pokud by nebyl vytvořen skript pro automatické přeladění, zaměstnanec by pak musel ručně prozkoumat celý výsledek skenování softwaru *w\_scan2* a vyhledat nové a zrušené TV programy. S tím souvisí nutnost manuálních úprav všech dílčích konfiguračních souborů včetně SAP serveru a souboru M3U8. Konfigurační soubory DVBlastu obsahují kromě SID i cílové vícesměrové IP adresy. Pokud by zaměstnanec dlouhodobě všechny související operace prováděl ručně, mohlo by docházet i k nepřehlednosti v těchto IP adresách.

Existuje vysoké riziko, že zaměstnanec nebude schopen sám monitorovat všechny změny v síti DVB-T2 a zajistit klientům společnosti aktuální programovou nabídku.

Další nespornou výhodou je to, že není zapotřebí platit žádný pravidelný licenční poplatek, protože byly použity nástroje a utility pod svobodnými licencemi.

Pro klienty firmy je nabízená IPTV pohodlným řešením, protože není potřeba do každého domu instalovat televizní rozvody, antény ani jiné komponenty. Další výhodou je to, že společnost ABC, s.r.o. nabízí IPTV všem svým klientům v ceně internetového připojení. Posledním přínosem je možnost, že klienti mohou sledovat televizi takřka na libovolném zařízení. Při příjmu IPTV záleží hlavně na příslušném směrovači klienta a jeho funkcích. Vzhledem k tomu, že používaný protokol IGMP byl již definován dokumentem RFC v roce 2003, existuje vysoká pravděpodobnost, že i starší síťové prvky klientů budou protokoly navržené v této práci podporovat.

## ZÁVĚR

Cílem této diplomové práce bylo navrhnout a realizovat komplexní řešení pro poskytování služby IPTV ve firemním prostředí. Toto nové řešení spojuje prvky existujících řešení s požadavky plynoucími z analýzy současného stavu.

První část práce pojednává o teoretických východiscích, na jejichž základě byl realizován návrh a tvorba řešení. Nejprve jsem v této kapitole rozvedl pojmy jako je počítačová síť a její klasifikace včetně síťových modelů a architektur a pokračoval jsem uvedením některých aktivních síťových prvků. Následně jsem se podrobněji zabýval vybranými síťovými přenosovými protokoly, jež vytváří klíčovou součást mé práce. Také jsem představil existující přenosové metody a operační systémy, zejména OS Linux, protože ten je využíván v tomto funkčním řešení. Teoretická východiska dokumentují oblast DVB a zvláště jsou blíže uvedeny pozemní standardy DVB-T a DVB-T2. Kapitulu uzavírá objasnění vývojových diagramů a SWOT analýzy.

V následující kapitole byla popsána anonymizovaná společnost ABC, s.r.o. Dále jsem se věnoval popisu a vzájemnému srovnání výhod a nevýhod současných existujících softwarů, které mohou zpracovávat signál z různých typů sítí DVB. Výstupem toho bylo formulování nových požadavků, které využívá nově vytvořené softwarové řešení.

Poslední kapitola obsahuje algoritmy několika procesů vycházející z požadavků společnosti ABC, s.r.o. Majoritní část návrhu je podložena procesy, které jsou náležitě identifikovány, popsány a také graficky znázorněny pomocí vývojových diagramů. Praktická realizace softwarového řešení podle návrhu obsahuje programové fragmenty kódu s popisem jejich funkčnosti. Tyto fragmenty kódu jsem sestavil ve skriptovacím jazyce Python a slouží pro automatické generování potřebných souborů pro provoz IPTV. Dále jsou představeny možné varianty příjmu IPTV včetně ukázek vybraných zařízení.

V závěru kapitoly uvádím finanční náročnost a přínosy tohoto nového řešení.

V předložené diplomové práci uvádím komplexní řešení problematiky ve shodě se zadáním diplomní práce a které také vyhovuje požadavkům popisovaného firemního prostředí. Z uvedeného tak vyplývá, že bylo dosaženo vytyčených cílů.



## SEZNAM POUŽITÝCH ZDROJŮ

- (1) HORÁK, Jaroslav a Milan KERŠLÁGER. *Počítačové sítě pro začínající správce*. 5., aktualiz. a rozš. vyd. Brno: Computer Press, 2013, 303 s. ISBN 978-80-251-3176-3.
- (2) JORDÁN, Vilém a Viktor ONDRÁK. *Infrastruktura komunikačních systémů I: univerzální kabelážní systémy*. Druhé, rozšířené vydání. Brno: CERM, akademické nakladatelství, 2015. ISBN 978-80-214-5115-5.
- (3) JIROVSKÝ, Václav. *Vademecum správce sítě*. Praha: Grada, 2001. ISBN 80-716-9745-1.
- (4) DONAHUE, Gary A. *Kompletní průvodce síťového experta*. Brno: Computer Press, 2009. ISBN 978-80-251-2247-1.
- (5) *RFC 3550 - RTP: A Transport Protocol for Real-Time Applications* [online]. [cit. 2023-12-03]. Dostupné z: <https://datatracker.ietf.org/doc/html/rfc3550>
- (6) KUROSE, James F. a ROSS, Keith W. *Počítačové sítě*. Brno: Computer Press, 2014. ISBN 978-80-251-3825-0.
- (7) SILBERSCHATZ, Abraham; B. GALVIN, Peter a GAGNE, Greg. *Operating System Concepts*. 10th Edition. 2018. ISBN 978-1-119-32091-3
- (8) NEMETH, Evi; SNYDER, Garth a HEIN, Trent R. *Linux: kompletní příručka administrátora : 2. aktualizované vydání*. Administrace. Brno: Computer Press, 2008. ISBN 978-80-251-2410-9.
- (9) *The Linux Kernel Archives* [online]. [cit. 2023-12-06]. Dostupné z: <https://www.kernel.org/>
- (10) *Tecmint* [online]. [cit. 2023-12-06]. Dostupné z: <https://www.tecmint.com/linux-distributions/>
- (11) *Python documentation* [online]. [cit. 2024-01-15]. Dostupné z: <https://docs.python.org/3/tutorial/index.html>
- (12) LEGÍŇ, Martin. *Televizní technika DVB-T*. Praha: BEN - technická literatura, 2006. ISBN 80-7300-204-3.
- (13) KRATOCHVÍL, Tomáš. *Analysis of transmission distortions in digital television DVB-T/H: Analýza přenosových zkreslení digitální televize DVB-T/H: short version of habilitation thesis*. Brno: VUTIUM, 2009. ISBN 978-80-214-3859-0.
- (14) BEDNÁŘ, Jiří a GREGORA, Pavel. *Přijem DVB-T: Analýza přenosových zkreslení digitální televize DVB-T/H: short version of habilitation thesis*. Praha: BEN - technická literatura, 2007. ISBN 978-80-7300-221-3.
- (15) *České Radiokomunikace hostily Svět televizní budoucnosti v České republice* [online]. [cit. 2024-01-20]. Dostupné z: <https://www.cra.cz/ceske-radiokomunikace-hostily-svet-televizni-budoucnosti-v-ceske-republice>

- (16) *Spuštěno vysílání v první části přechodové sítě DVB-T2* [online]. [cit. 2024-01-21]. Dostupné z: <https://www.cra.cz/spusteno-vysilani-v-prvni-casti-prechodove-site-cra-dvb-t2>
- (17) *Přechod na nový vysílací standard DVB-T2 se blíží ke konci, přeladování čeká poslední oblast – Zlínsko* [online]. [cit. 2024-01-21]. Dostupné z: <https://www.cra.cz/prechod-na-novy-vysilaci-standard-dvb-t2-se-blizi-ke-konci-preladovani-ceka-posledni-oblast-zlinsko>
- (18) *Multiplex 21 - CZ - Vysílače* [online]. [cit. 2024-01-22]. Dostupné z: <https://www.parabola.cz/vysilace/cz/multiplex-21/>
- (19) *Multiplex 22 - CZ - Vysílače* [online]. [cit. 2024-01-22]. Dostupné z: <https://www.parabola.cz/vysilace/cz/multiplex-22/>
- (20) *Multiplex 23 - CZ - Vysílače* [online]. [cit. 2024-01-22]. Dostupné z: <https://www.parabola.cz/vysilace/cz/multiplex-23/>
- (21) *Multiplex 24 - CZ - Vysílače* [online]. [cit. 2024-01-22]. Dostupné z: <https://www.parabola.cz/vysilace/cz/multiplex-24/>
- (22) *Převod kanálu a frekvence - Nové digitální vysílání* Informační web o digitálním vysílání DVB-T a DVB-T2 [online]. [cit. 2024-01-23]. Dostupné z: <https://dvb-t2.cz/prevod-kanalu-a-frekvence/>
- (23) *Pokrytí | Zemské digitální televizní a rozhlasové vysílání v ČR* [online]. [cit. 2024-01-23]. Dostupné z: <https://digi.ctu.cz/dtv/>
- (24) TRETEROVÁ, Eliška. *Návrh a vývoj algoritmů: modul-vývojové diagramy a příkazy jazyka Borland Pascal*. Ostrava: Ostravská univerzita, 2003. Systém celoživotního vzdělávání Moravskoslezská. ISBN 80-7042-854-6.
- (25) *Co je SWOT analýza a jak ji vypracovat* [online]. [cit. 2024-01-25]. Dostupné z: <https://www.generaliceskaprofi.cz/ze-zivota/co-je-swot-analyza-a-jak-ji-vypracovat>
- (26) DAVID, Fred R. *Strategic Management: Concepts and Cases*. 13th Edition. Pearson College Div, 2010. ISBN 978-0-13-612098-8.
- (27) *TSDuck* [online]. [cit. 2024-02-05]. Dostupné z: <https://tsduck.io/>
- (28) *MuMuDVB* [online]. [cit. 2024-02-05]. Dostupné z: <https://mumudvb.net/>
- (29) *DVBlast - Open Source fast DVB/ASI network streamer! - VideoLAN* [online]. [cit. 2024-02-05]. Dostupné z: <https://www.videolan.org/projects/dvblast.html>
- (30) *olmi.tv - TBS6205se DVB-T2/C2/T/C(J.83A/B/C)/ISDB-T/C /ATSC1.0 Quad TV Tuner Card* [online]. [cit. 2024-02-07]. Dostupné z: <https://www.olmi.tv/d/tbs6205se-dvb-t2-c2-t-cj83a-b-c-isdb-t-c-atsc10-quad-tv-tuner-card-1001636/>

- (31) *EVOLVEO.cz / oficiální eshop značky* [online]. [cit. 2024-02-07]. Dostupné z: [https://cdn.myshoptet.com/usr/eshop.evolveo.cz/user/shop/big/3618-5\\_image1.jpg?5f48ec46](https://cdn.myshoptet.com/usr/eshop.evolveo.cz/user/shop/big/3618-5_image1.jpg?5f48ec46)
- (32) *TBS6205 uživatelská příručka* [online]. [cit. 2024-02-07]. Dostupné z: [https://www.olmi.tv/files/eshop/download/tbs6205\\_uzivatelska\\_prirucka.pdf](https://www.olmi.tv/files/eshop/download/tbs6205_uzivatelska_prirucka.pdf)
- (33) *Kodi* [online]. [cit. 2024-02-22]. Dostupné z: <https://kodi.tv/>
- (34) *STRONG android box SRT LEAP-S3+* [online]. [cit. 2024-02-22]. Dostupné z: [https://www.tsbohemia.cz/strong-android-box-srt-leap-s3-4k-uhd-h-265-hevc-netflix-o2-tv-hbo-max-hdmi-usb-lan-wi-fi-android-tv-11\\_d512752.html](https://www.tsbohemia.cz/strong-android-box-srt-leap-s3-4k-uhd-h-265-hevc-netflix-o2-tv-hbo-max-hdmi-usb-lan-wi-fi-android-tv-11_d512752.html)
- (35) *Oficiální sada s Raspberry Pi 4B - 2GB* [online]. [cit. 2024-02-22]. Dostupné z: <https://rpishop.cz/280601/raspberry-pi-4-oficialni-star-t-box-2gb/#tab-description>
- (36) *CZC* [online]. [cit. 2024-03-14]. Dostupné z: <https://www.czc.cz/dell-poweredge-r250-e-2314-16gb-1x2tb-sata-7-2k-idrac-9-exp-h355-1u-3y-basic-on-site/353763/produkt>

## SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ

5G	Pátá generace technologie bezdrátových systémů pro mobilní sítě
API	Application Programming Interface (rozhraní pro programování aplikací)
ASCII	American Standard Code for Information Interchange
ATSC	Advanced Television Systems Committee (mezinárodní soubor norem pro vysílání a digitální televizní přenos prostřednictvím pozemních, kabelových a satelitních sítí)
BASH	Bourne-Again Shell (příkazový interpret)
CEC	HDMI-CEC Consumer Electronics Control
CLI	Command Line Interface (Rozhraní příkazového řádku)
CPU	Central Processing Unit (Centrální procesorová jednotka)
DVB	Digital Video Broadcasting
DVB-C	Digital Video Broadcasting – Cable
DVB-T	Digital Video Broadcasting – Terrestrial (první generace)
DVB-T2	Digital Video Broadcasting – Terrestrial (druhá generace)
EBU	European Broadcasting Union (Evropská unie pro rozhlasové a televizní vysílání)
EFE	External Factor Evaluation (matice pro SWOT analýzu)
EPG	Electronic Program Guide (Elektronický programový průvodce)
ETSI	European Telecommunications Standards Institute (Evropský telekomunikační institut)
Gbps	Gigabit per second (Gigabit za sekundu)
GNU	GNU's Not Unix (Projekt GNU je prvkem svobodného softwaru a iniciativy open source)
GPL	General Public License (licenční smlouva pro svobodný software)
GPON	Gigabit Passive Optical Network (Gigabitová pasivní optická síť)
GUI	Graphical User Interface (Grafické uživatelské rozhraní)
H265	High Efficiency Video Coding (HEVC nebo také H265 - videokodek)
HD	High definition (vysoké rozlišení – obvykle 1920x1080)
HDMI	High-Definition Multimedia Interface (rozhraní pro přenos zejména obrazu)
IFE	Internal Factor Evaluation (matice pro SWOT analýzu)
IGMP	Internet Group Management Protocol
IP	Internet Protocol

IPTV	Internet Protocol Television (internetová televize)
IPv4	Internet Protocol version 4 (Internet Protokol verze 4)
LTE	3GPP Long Term Evolution (technologie pro vysokorychlostní přenos dat v mobilních sítích)
M3U	MPEG Version 3.0 URL
M3U8	MPEG Version 3.0 URL s podporou kódování UTF-8
Mbps	Megabit per second (Megabit za sekundu)
MLD	Multicast Listener Discovery
MPEG	Moving Picture Experts Group (videokodek)
OS	Operační systém
PCI	Peripheral Component Interconnect (počítačová sběrnice pro připojení periférií k základní desce)
PCIe	Peripheral Component Interconnect Express
RPI	Raspberry Pi (malý jednodeskový počítač)
RTP	Real-Time transport Protocol (přenosový mediální protokol)
SAP	Session Announcement Protocol (protokol oznamující dostupná vysílání v IP sítích)
SD karta	Secure Digital (kompaktní médium s flash pamětí)
SHA256	Secure Hash Algorithm 256-bit (kryptografická hashovací funkce)
SID	Service ID (ve volném významu: TV program)
SW	Software
TS	Transport Stream (formát digitálního kontejneru pro přenos multimédií)
URL	Uniform Resource Locator (zjednodušeně odkaz na zdroj v počítačových sítích)
UTF-8	kódování znaků
VBR	Variable Bitrate (proměnlivý datový tok)
VP	Soubor s volitelnými přenosy, který je využíván v rámci procesu Automatického přeladění
XML	Extensible Markup Language (Rozšiřitelný značkovací jazyk)

## SEZNAM POUŽITÝCH OBRÁZKŮ

Obrázek č. 1: Schéma sběrnice topologie .....	14
Obrázek č. 2: Schéma hvězdicové topologie .....	14
Obrázek č. 3: Schéma kruhové topologie .....	15
Obrázek č. 4: Vyobrazení vrstev modelu ISO/OSI.....	16
Obrázek č. 5: Srovnání vrstev modelu ISO/OSI a architektury TCP/IP.....	17
Obrázek č. 6: Snímek z aplikace Wireshark – protokol SAP .....	20
Obrázek č. 7: Snímek z aplikace VLC – protokol SAP .....	21
Obrázek č. 8: Snímek z aplikace Wireshark – IGMP Leave, Join group .....	23
Obrázek č. 9: Schématické znázornění přenosové metody – unicast .....	23
Obrázek č. 10: Schématické znázornění přenosové metody – multicast .....	24
Obrázek č. 11: Schématické znázornění přenosové metody – broadcast .....	24
Obrázek č. 12: Schematické znázornění obecného modelu vysílání a příjmu DVB-T ..	27
Obrázek č. 13: Snímek z aplikace VLC – poskytovatel MUX21 .....	29
Obrázek č. 14: Některé grafické obrazce vývojového diagramu.....	30
Obrázek č. 15: Vyobrazení struktury SWOT analýzy .....	31
Obrázek č. 16: Ilustrační schéma architektury navrhovaného řešení .....	41
Obrázek č. 17: Vyobrazení čtyřtunerové PCIe karty pro příjem DVB-T2.....	42
Obrázek č. 18: Fotografie zařízení Evolveo Sigma T2.....	43
Obrázek č. 19: Ověření dostupnosti čtyřtunerové PCIe DVB-T2 karty .....	53
Obrázek č. 20: Ověření dostupnosti DVB-T2 zařízení Evolveo Sigma T2.....	54
Obrázek č. 21: Ověření dostupnosti všech DVB-T2 tunerů (PCIe i USB).....	54
Obrázek č. 22: Opis části výsledku skenování DVB-T2 nástrojem <i>w_scan2</i> .....	56
Obrázek č. 23: Výpis části výsledku skenování DVB-T2 nástrojem <i>w_scan2</i> – TV programy .....	56
Obrázek č. 24: Ukázka souboru xspf vygenerovaného nástrojem <i>w_scan2</i> .....	57
Obrázek č. 25: Ukázka manuálu k nástroji DVblast v OS Ubuntu 22.....	58
Obrázek č. 26: Ukázka konfiguračního souboru pro nástroj DVblast .....	59

Obrázek č. 27: Snímek z aplikace VLC – vyobrazení některých proudů TV programu ČT1 .....	59
Obrázek č. 28: Ukázka skriptu pro hromadné spuštění služeb DVBlasu .....	60
Obrázek č. 29: Ukázka konfigurace služby pro DVBlas – MUX24.....	60
Obrázek č. 30: Ukázka konfiguračního souboru SAP serveru .....	61
Obrázek č. 31: Automatické přeladění – skript – proměnná <i>settings</i> .....	64
Obrázek č. 32: Vyobrazení struktury souboru VP (vlastních přenosů) .....	67
Obrázek č. 33: Automatické přeladění – skript – funkce <i>w_scan2</i> .....	68
Obrázek č. 34: Automatické přeladění – skript – podpůrné funkce I.....	68
Obrázek č. 35: Automatické přeladění – skript – funkce <i>parseXML</i> .....	69
Obrázek č. 36: Automatické přeladění – skript – struktura dočasné proměnné <i>tracks</i> ..	70
Obrázek č. 37: Automatické přeladění – skript – funkce <i>parseXML_my_streams</i> .....	70
Obrázek č. 38: Automatické přeladění – skript – podpůrné funkce II (správa souborů)	71
Obrázek č. 39: Automatické přeladění – skript – podpůrné funkce III .....	72
Obrázek č. 40: Automatické přeladění – skript – podpůrná funkce <i>calculate_sha256</i> ..	73
Obrázek č. 41: Automatické přeladění – skript – funkce <i>prepare_files</i> I.....	73
Obrázek č. 42: Automatické přeladění – skript – funkce <i>prepare_files</i> II .....	74
Obrázek č. 43: Automatické přeladění – skript – funkce <i>prepare_files</i> III .....	75
Obrázek č. 44: Automatické přeladění závěr skriptu I .....	76
Obrázek č. 45: Automatické přeladění závěr skriptu II .....	77
Obrázek č. 46: Vyobrazení části Cron tabulky v textovém editoru Nano .....	78
Obrázek č. 47: Snímek aplikace VLC – příjem vícesměrového vysílání TV programu ČT1 pomocí protokolu SAP.....	79
Obrázek č. 48: Snímek aplikace VLC – zadání URL .....	79
Obrázek č. 49: Snímek z aplikace Kodi – knihovna.....	81
Obrázek č. 50: Snímek z aplikace Kodi – podrobnosti doplňku IPTV Simple Client ...	81
Obrázek č. 51: Snímek z aplikace Kodi – nastavení doplňku IPTV Simple Client .....	82
Obrázek č. 52: Snímek z aplikace Kodi – programová nabídka.....	82

Obrázek č. 53: Snímek z aplikace Kodi – podrobné informace z EPG některých TV programů.....	83
Obrázek č. 54: Vzhled zařízení STRONG Android box SRT LEAP-S3+ .....	84
Obrázek č. 55: Vyobrazení součástí oficiální sady s Raspberry Pi 4B – 2GB .....	85



## SEZNAM POUŽITÝCH TABULEK

Tabulka č. 1: Přehled celoplošných multiplexů, frekvencí a vysílačů DVB-T2 pro Brno a okolí .....	28
Tabulka č. 2: Přehled regionálních multiplexů, frekvencí a vysílačů DVB-T2 pro Brno a okolí .....	29
Tabulka č. 3: Přehled poskytovatelů jednotlivých multiplexů sítě DVB-T2 v oblasti Brna .....	29
Tabulka č. 4: IFE matice (interní faktory SWOT analýzy) .....	39
Tabulka č. 5: EFE matice (externí faktory SWOT analýzy).....	40
Tabulka č. 6: Plán adresního prostoru pro IPTV .....	44
Tabulka č. 7: Přehled nákladů na softwarové řešení .....	86
Tabulka č. 8: Přehled nákladů na hardwarové vybavení .....	86

## SEZNAM POUŽITÝCH DIAGRAMŮ

Diagram č. 1: Vývojový diagram procesu Poskytování obsahu ze sítě DVB-T2 do sítě IP .....	45
Diagram č. 2: Vývojový diagram procesu oznamování dostupnosti multimedialního vysílání v síti.....	46
Diagram č. 3: Vývojový diagram procesu Odbavování regionálního televizního vysílání .....	48
Diagram č. 4: Vývojový diagram procesu automatického přeladění.....	51