

Zpracování televizního vysílání na Linuxu

Ondřej Caletka



RIPE NCC
RIPE NETWORK COORDINATION CENTRE

cesnet
■■■■■■

6. března 2021



Uvedené dílo podléhá licenci Creative Commons Uvedte autora 3.0 Česko.

Obsah

- 1 Historie a teorie
- 2 Příjem na Linuxu
- 3 Použití v domácnosti
- 4 Použití ve větším prostředí
- 5 Zpracování nahrávek

Historie a teorie

Historie TV vysílání u nás

- 1953 zahájení černobílého SD vysílání v pásmu VHF
- 1970 II. program v pásmu UHF (první **set-top-boxy**)
- 1973 barevné vysílání SECAM (**výroba v PAL**)
- 1988 teletext (**7 Mbps**)
- 1992 barevné vysílání PAL (úprava sovětských TVP)
 - 90s podpora pro **stereofonní** nebo **dvojazyčné** vysílání
- 1997 **digitální vysílání** CzechLink (DVB-S)
- 2005 přepínání **poměru stran obrazu** mezi 4:3 a 16:9
- 2008 start DVB-T, první **HD vysílání**
- 2012 vypnutí analogového vysílání
- 2013 vysílání **zvukového popisu**
- 2017 start **DVB-T2** s HEVC
- 2020 vypnutí DVB-T

DVB-T2: making TV great again

- výrazný nárůst **datové kapacity**: 32 Mb/s místo 20 Mb/s
- velké **jednofrekvenční sítě** – výrazná úspora pásma
- odstřížení analogové televizní historie:
 - pouze **čtvercové pixely**
 - poměr stran **pouze 16:9**
 - žádné *prokládání*: 50 úplných snímků každou sekundu
 - **dvojazyčné vysílání** v samostatných stopách
- skončily i pěkné věci:
 - kódy VPS/PDC pro automatické spouštění domácích rekordérů
 - teletextové skryté podtitulky

Kodeky obrazu podle ITU-T a ISO/IEC

- H.261, MPEG-1
- H.262, MPEG-2
- H.263, MPEG-4 Part 2 (SP, ASP, DivX, XviD, ...)
- H.264 = MPEG-4 Part 10 (AVC)
- H.265 = MPEG-H Part 2 (HEVC)
- H.266 = MPEG-I Part 3 (VVC)

Základní principy kodeků MPEG

- norma definuje formát *bitstreamu* a dekodér
- chování kodéru není definováno; obvykle se zdokonaluje v průběhu životního cyklu formátu
- kodér pro nejnovější standard obvykle nevyužívá všechny vlastnosti *bitstreamu*
- kvalitu ovlivňuje především typ kodéru a jeho nastavení
- při vysoké kvalitě je dosaženo **transparence**, kdy nelze pozorovat změnu proti originálu
- hardwarové kodéry jsou obvykle rychlejší a **méně kvalitní**
- softwarové kodéry jsou **nejvíce nastavitelné**, umožňují nejvyšší kvalitu

Jak vypadá vysílání

- multiplex = datový tok o dané kapacitě
- uvnitř je MPEG Transport Stream = proud 188 oktetů dlouhých paketů
- každý paket má 32bitové záhlaví, začínající 0x47 (G)
- 13bitový *identifikátor paketu* (PID) označuje různé multiplexované toky (obraz, zvuk, metadata)
- uvnitř je buď Packetised Elementary Stream nebo PSI
- Program-Specific Information jsou metadata
 - Program Association Table (PID=0, tabulka *kanálů*)
 - Program Map Table (mapa obrazu, zvuku, titulků pro daný *kanál*)
 - Network Information Table (informace o dalších vysílačích)
 - Event Information Table (elektronický průvodce programem)
 - Time and Date Table (přesný čas a časové pásmo)

Příklad: regionální odpojování ČT 1

- v PAT existují 3 samostatné varianty ČT 1
- každý má svou PMT tabulku, většinu času odkazují na stejné proudy obrazu a zvuku
- při regionálním vysílání se modifikují tabulky PMT tak, že začnou ukazovat na nové proudy regionálních mutací
- přijímače musí průběžně vyhodnocovat změny PMT

```
CT 1 HD T2      (0x010c) 1f: A 0x0848 (cze) 0x0849 (qaa) 0x084b (cze) TT 0x0852 AC3 0x084a SUB 0x0866
CT 1 JM HD T2   (0x011a) 1f: A 0x0848 (cze) 0x0849 (qaa) 0x084b (cze) TT 0x0852 AC3 0x084a SUB 0x0866
CT 1 SM HD T2   (0x0118) 1f: A 0x0848 (cze) 0x0849 (qaa) 0x084b (cze) TT 0x0852 AC3 0x084a SUB 0x0866
CT 2 HD T2      (0x010e) 1f: A 0x08ac (cze) 0x08ad (qaa) 0x08af (cze) TT 0x0852 AC3 0x08ae SUB 0x08ca
CT 24 HD T2     (0x0110) 1f: A 0x0910 (cze) 0x0913 (cze) TT 0x0852 SUB 0x092e
CT 3 HD T2      (0x0116) 1f: A 0x0a3c (cze) 0x0a3f (cze) TT 0x0852 SUB 0x0a5a
CT sport HD T2  (0x0112) 1f: A 0x0974 (cze) 0x0977 (cze) TT 0x0852 AC3 0x0976 SUB 0x0992
```

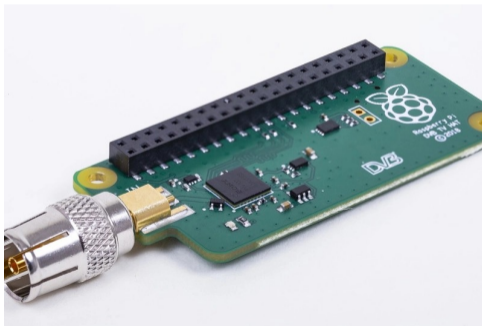
```
CT 1 HD T2      (0x010c) 1f: A 0x0848 (cze) 0x0849 (qaa) 0x084b (cze) TT 0x0852 AC3 0x084a SUB 0x0866
CT 1 JM HD T2   (0x011a) 1f: A 0x0b04 (cze) 0x0b05 (qaa) 0x0b07 (cze) TT 0x0852 AC3 0x0b06 SUB 0x0b22
CT 1 SM HD T2   (0x0118) 1f: A 0x0aa0 (cze) 0x0aa1 (qaa) 0x0aa3 (cze) TT 0x0852 AC3 0x0aa2 SUB 0x0abe
CT 2 HD T2      (0x010e) 1f: A 0x08ac (cze) 0x08ad (qaa) 0x08af (cze) TT 0x0852 AC3 0x08ae SUB 0x08ca
CT 24 HD T2     (0x0110) 1f: A 0x0910 (cze) 0x0913 (cze) TT 0x0852 SUB 0x092e
CT 3 HD T2      (0x0116) 1f: A 0x0a3c (cze) 0x0a3f (cze) TT 0x0852
CT sport HD T2  (0x0112) 1f: A 0x0974 (cze) 0x0977 (cze) TT 0x0852 AC3 0x0976 SUB 0x0992
```

Příjem na Linuxu

- subsystém V4L-DVB spravovaný na `linuxtv.org`
- podpora především pro **USB přijímače**
- samostatné ovladače pro tuner a *frontend* (demodulátor)
- komponenty velmi často vyžadují **nahrání firmware** (je třeba číst logy)
- najít průnik aktuálně podporovaného a aktuálně dostupného tuneru *není vždy jednoduché*

Raspberry Pi TV uHAT

- plně podporovaný linuxový přijímač
- citlivý a selektivní
- garantovaná výroba do ledna 2023



Základní nástroje dvb-tools

`dvbv5-scan` naladí kartu a vypíše dostupné kanály

`dvbv5-zap` naladí kartu na příslušný multiplex a ukazuje kvalitu signálu

`dvb-fe-tool` čte data z frontendu (sílu a kvalitu signálu)

Příklad inicializačního souboru pro dvbv5-scan

```
[CHANNEL]
    DELIVERY_SYSTEM = DVBT2
    FREQUENCY = 514000000
    BANDWIDTH_HZ = 8000000
```

Kanál 21 = 474 MHz a dále v rastru 8 MHz

- *Wireshark* pro MPEG-TS a DVB karty
- dekóduje zejména PSI z naložené karty
- dokáže analyzovat i záznamy v MPEG-TS souborech

```
dvbsnoop V1.4.50 -- http://dvbsnoop.sourceforge.net/
```

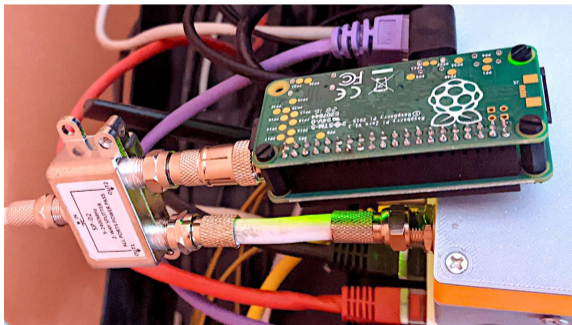
```
-----  
SECT-Packet: 00000001   PID: 2260 (0x08d4), Length: 105 (0x0069)  
Time received: Fri 2021-03-05 19:37:40.515  
-----
```

```
0000: 74 f0 66 00 10 cb 00 00   f0 00 f0 59 00 00 01 90   t.f.....Y....  
0010: 00 01 01 f0 50 02 29 00   03 01 24 68 74 74 70 3a   ....P.)...$http:  
0020: 2f 2f 68 62 62 74 76 2e   63 65 73 6b 61 74 65 6c   //hbbtv.ceskatel  
0030: 65 76 69 7a 65 2e 63 7a   2f 73 74 61 72 74 2f 00   evize.cz/start/.  
0040: 00 09 05 00 00 01 01 01   ff 01 01 01 0c 63 7a 65   .....cze  
0050: 08 43 54 2d 73 74 61 72   74 15 0a 69 6e 64 65 78   .CT-start..index  
0060: 2e 68 74 6d 6c 0c 51 f1   98   .html.Q..
```

Použití v domácnosti

- MythTV, vdr, nebo **TVheadend**
- server-klient architektura
- inteligentní přeladování přijimačů
- klient součástí Kodi, včetně distribucí jako LibreELEC
- možnost nahrávat podle televizního programu
- možnost nastavit pravidla nahrávání podle metadat

- otevřený protokol pro připojení DVB přijímačů pomocí TCP/IP
- hardwarové krabičky s nulovou konfigurací
- objevování pomocí UPnP/multicastu
- podporováno v TVheadend



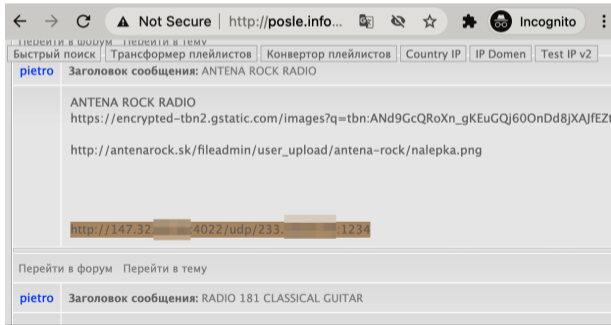
Použití ve větším prostředí

Multicastová televize v síti

- pro situace s mnohem větším množstvím klientů, než přijímaných multiplexů
- vyžaduje podporu v síťovém hardwaru (zejména **Wi-Fi**)
- lze ale používat *host-only multicast* jako formu meziprocesové komunikace
- 7 paketů MPEG-TS se balí do UDP nebo RTP/UDP datagramů
- není vhodné streamovat **celý multiplex**, vhodnější je *rozdělit* multiplex po jednotlivých kanálech
- při dělení je třeba přegenerovat společné PSI (např. PAT)
- nejlépe funguje `dvb-lst` z dílny VideoLAN

Překlenování nefunkčního multicastu

- nástroje jako UDPxy nebo můj rtp2httpd
- překládá MPEG-TS do nekonečného HTTP souboru
- není dobrý nápad *provozovat na veřejném Internetu*



Ukládání a timeshift

- nástroj `multicat` opět z dílny VideoLAN
- ukládá přijatá data do souboru `.ts`
- zároveň ukládá časové značky přijatých paketů do `.aux`
- automaticky dělí na soubory (po hodinách)
- dokáže odvysílat znovu kteroukoli část záznamu

Problém s diskovým bufferem v Linuxu

- zápisy na disk probíhají do diskového bufferu v RAM
- ten se vyprázdní zápisem na disk:
 - **na popředí** po zaplnění `vm.dirty_ratio = 20` procent paměti nebo uplynutí `vm.dirty_expire_centisecs = 3000`
 - **na pozadí** po zaplnění `vm.dirty_background_ratio = 10` procent paměti
- jenže dnešní počítače mají *hodně* paměti
- vyprázdnění bufferu na popředí zablokuje běžící aplikaci
- UDP zprávy přijaté během vyprazdňování bufferu na popředí jsou **nenávratně ztraceny**
- **řešení**: snížit `vm.dirty_background_bytes`

Zpracování nahrávek

- lze načíst do nelineárního video editoru (případně po konverzi do jiného kontejneru)
- výpadky v příjmu **mohou způsobit desynchronizaci obrazu a zvuku**
- editace je zdlouhavá a vyžaduje rekompresi
- přitom nahrubo lze stříhat jen *binárními nůžkami*

Střih s přesností na GOP

- stříhání bez rekompresce
- lze stříhat pouze na hranicích Group-of-Pictures (cca. dvakrát za sekundu)
- jak najít správná místa ke střihu?
- FFmpeg má výstupní modul, který vytváří samostatně stojící MPEG-TS soubory

Remultiplexování do samostatných MPEG-TS souborů

```
$ ffmpeg -i rawrecord.ts -map 0:v -map 0:a \  
-map 0:s -map -0:i:2130 -c copy \  
-f ssegment -segment_time 0.5 frag-%05d.ts
```

Střih s přesností na GOP

- vlastní *střih* spočívá v odmazání nepotřebných fragmentů
- zbylé fragmenty jednoduše binárně spojíme
- výsledný záznam můžeme překomprimovat pro úsporu místa

Spojení finálního záznamu

Bez rekompresce:

```
$ cat frag-*.ts | ffmpeg -i - -map 0 -c copy TheMovie.mkv
```

S rekompresí a konverzí na 25 fps:

```
$ cat frag-*.ts | ffmpeg -i - -map 0 -c copy -r 25 \  
-crf 28 -c:v libx265 TheMovie.mkv
```

- přehrát *receiver-mix* audio popis
 - přehrávání původního zvuku a popisu zároveň
 - iVysílání používá *broadcast-mix*
- zpracovat bitmapové DVB titulky
 - textové titulky má iVysílání, ale ne pro živé vysílání
- softwarový prohlížeč HbbTV
 - k prohlížení internetové služby nelze použít počítač :)

Děkuji za pozornost

Ondřej Caletka

Ondrej@Caletka.cz

<https://Ondrej.Caletka.cz>



RIPE NCC
RIPE NETWORK COORDINATION CENTRE

cesnet
■■■■■■