

Facultatea de Electronică, Telecomunicații și Tehnologia Informației

Protocoale în timp real

Nae Daniel Mădălin

Master SIVA

Cuprins

Introducere	3
Real Time Protocol	7
Real Time Control Protocol	10
Compresed Real Time Protocol.....	11
HTTP Live Streaming	12
Real Time Messaging Protocol	14
Concluzii	16
Bibliografie:	17

Introducere

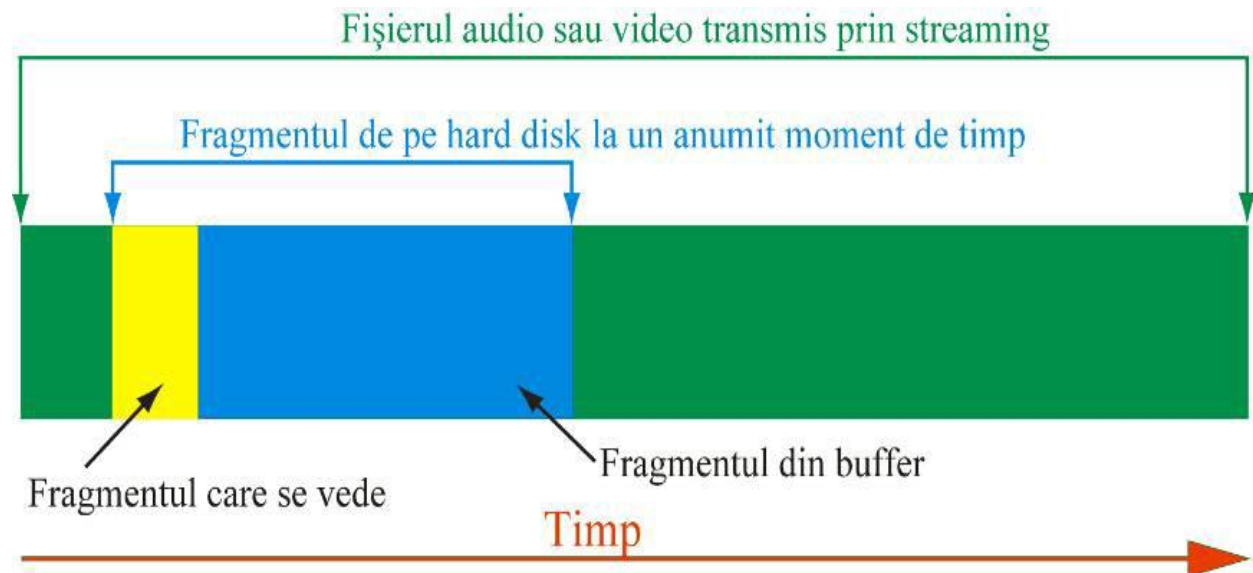
In ziua de azi, tehnologia ne permite sa livram un material video/audio in timp real prin internet, fara a mai fi nevoie sa asteptam pana materialul multimedia sa se descarce in calculator, PDA sau telefon mobil.

Destinatarul trebuie doar sa introduca adresa de internet a fisierului media intr-un program care suporta video streaming precum Windows Media Player sau Real Player.

Video/Audio Streaming - Este procesul prin care un material video/audio in format special pregatit in acest sens se trimite catre clientul care urmeaza sa afiseze materialul video folosind ca suport internetul.

Clientul vede continutul media direct de pe server in timp ce vine fluxul de date, fără a aștepta descărcarea completă

Viteza de transfer disponibila intre server si client este foarte importanta deoarece trebuie sa fie mai mare decat banda ceruta de fluxul de date ca sa poata fi afisate in timp real de catre programul client. Daca viteza de tranfer este insufienta decat decat cea necesara, pe ecran se va afisa materialul cu intreruperi de imagine si/sau sunet.



Principiul video/audio streaming

Streaming-ul în timp real este instantaneu și indiferent de lungimea fișierului video începe să ruleze în mai puțin de un minut. Utilizatorul poate „derula” înainte și înapoi, dar de fiecare dată când se va deplasa prin fișier, este nevoie de un nou proces de „buffer”.

Streaming-ul în timp real presupune servere specifice dedicate streaming-ului, iar în funcție de formatul/protocolul folosit se poate opta pentru un anumit server și/sau anumite protocoale.

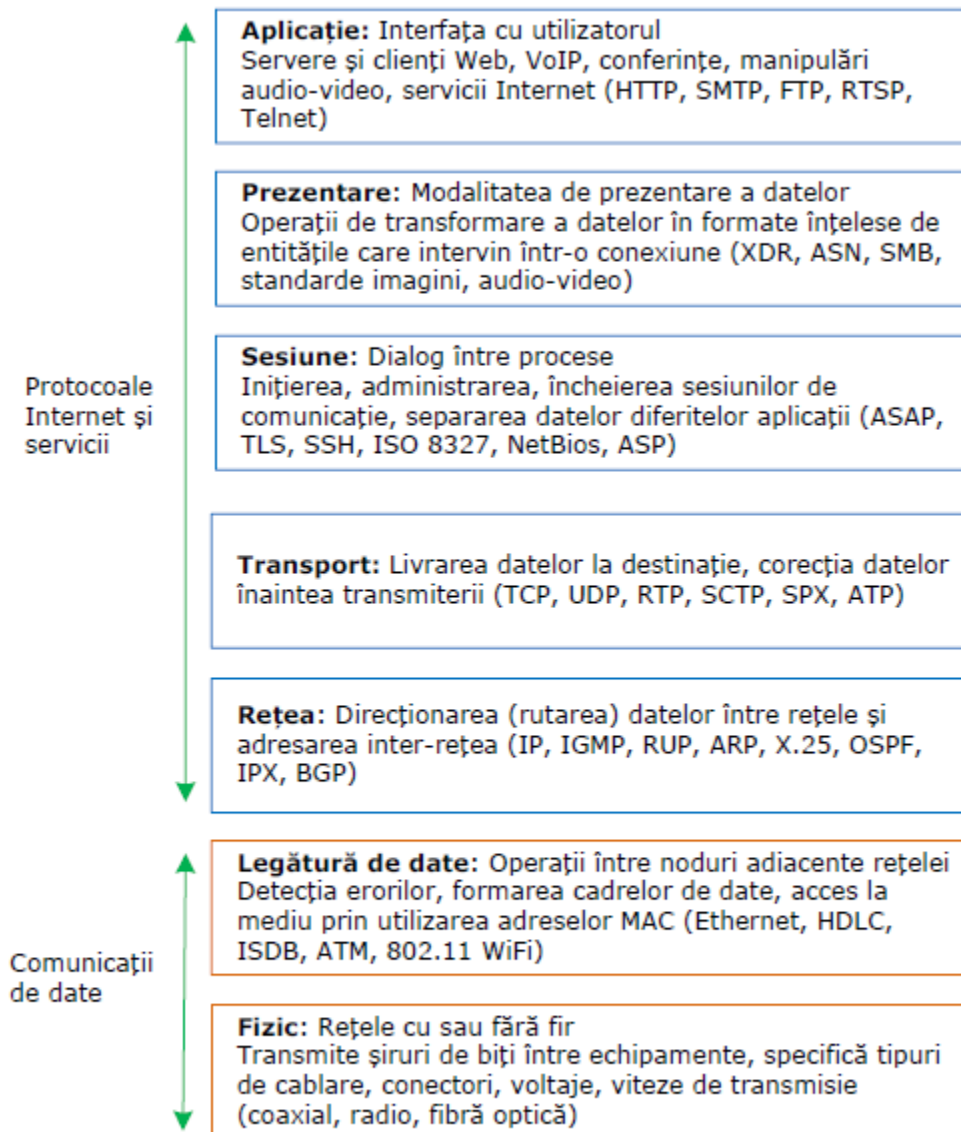
Un aspect importantă este **MBR-ul (Multiple BitRate)** adica crearea mai multor versiuni a aceleiași transmisii, între versiuni făcându-se o trecere dinamică în funcție de viteza/banda utilizatorului, astfel încât recepția să fie în timp real cu unele scăderi în calitate la viteze mai mici.

Streaming-ul „live” are la bază transmisia imaginilor video in acelasi timp cu realizarea evenimentelor, fără a fi necesară stocarea lor pe server. Cu alte cuvinte „materialul” poate fi accesat o singură dată (atunci când evenimentul are loc.

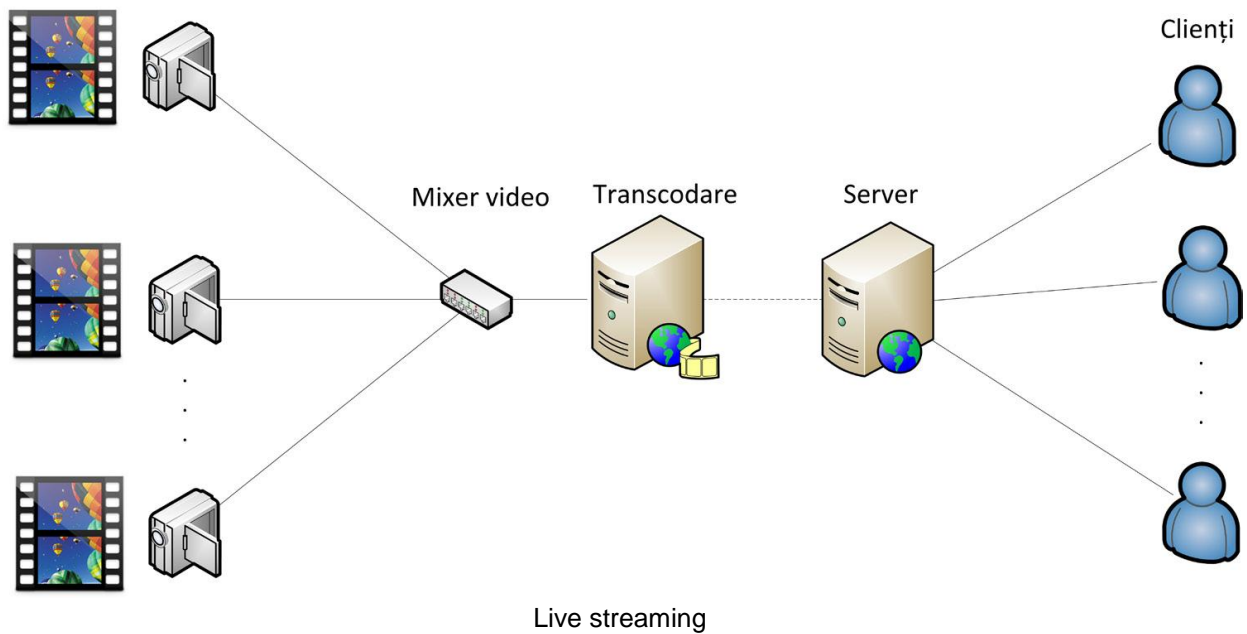
Nu există cale de întoarcere atunci când stream-ul este „live”, nu există posibilitatea de a vizualiza materialul din nou.

Protocoalele ne ajuta sa realizam reguli și convenții care determina modul în care datele media sunt transmise de la server la client într-o rețea de internet.

Protocolul determină formatul sau structura mesajului. Exemplan de protocoale :HTTP, UDP, TCP, RTP, RTSP, SCTP, RSVP.



Modelul OSI. Protocoale utilizate



Serverele care asigura flux media protejeaza datele private (fișierul nu este descărcat) și permite utilizatorilor să „interacționeze” cu fișierul (doar vizionare). De asemenea în cazul streaming-ului se asigură un control bun asupra accesului utilizatorilor la fișiere.

Real Time Protocol

RTP (de la engl. *Real-time Transport Protocol*) este un protocol care transmite in timp real informații de tip media printr-o rețea de internet . Acesta este specific pentru nivelul Transport din stiva OSI.

Scopul si problemele pe care incerca sa le rezolve acest protocol:

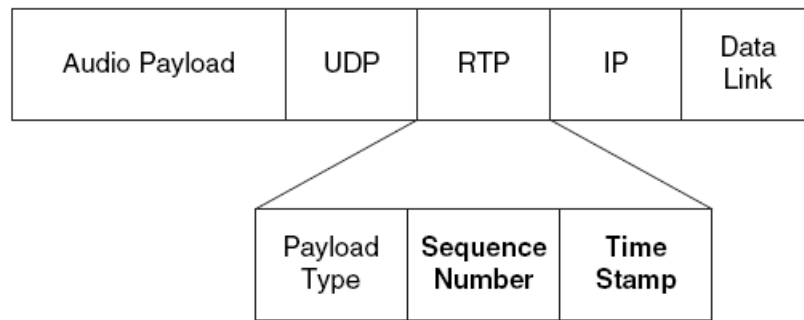
- Transporta pachete de date in timp real;
- Audio conferinte multicast
- Audio and Video Conference
- Suporta Layered Encodings (aplicatia multimedia poate modifica rata de transmiterea a pachetilor in functie de conexiune)

Caracteristici RTP:

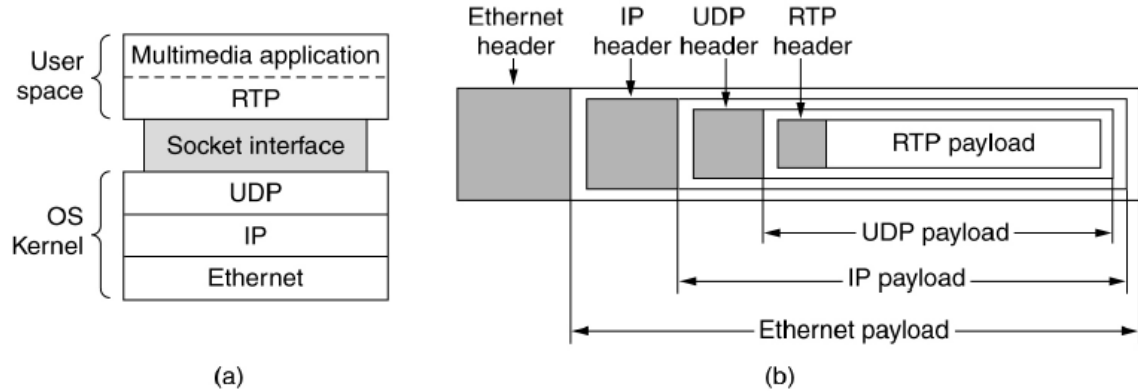
- precizează tipul câmpului de date
- numerotează pachetele
- înregistrează reperul de timp
- monitorizează transmiterea datelor
- permite transmiterea concomitentă pe adrese diferite, dacă multicastul este posibil la nivel de rețea.
- aplicațiile de obicei folosesc RTP implementat peste UDP, pentru ca să se poată folosi de posibilitatea sa de multiplexare și controlul checksum
- nu garantează transmiterea la timp a pachetelor..

Organizarea unei audio conferinte folosind RTP:

- fiecare membru trebuie sa aibă o adresă
- două porturi, unul pentru semnalele audio altul pentru schimbul de pachete RTP (care pot fi codate)
- fiecare segment este inglobat in pachetul RTP, care la randul sau este inglobat in UDP.
- In antetul RTP se afla informatii de codare (PCM, ADPCM sau LPC)



Antet RTP



(a) Poziționarea RTP în stiva de protocoale (b) încapsularea pachetului

Structura pachetului

+ Biți	0-1	2	3	4-7	8	9-15	16-31
0	Ver.	S	Ex	CC	M	SU	Numărul secvenței
32	Timp la transmitere						
64	identificatorul SSRC						
96	... identificatorii CSRC ...						
96+(CC×32)	Antet adițional (opțional), indică lungimea "AHL"						
96+(CC×32) + (Ex × (AHL + 16))	Data						

Câmpurile au următoarea semnificație:

- **Ver.** (versiune - 2 biți) - indică versiunea protocolului RTP utilizată. Versiunea actuală este 2.
- **S** (spațiere - un bit) - indica dacă există informație suplimentară la finalul pachetului RTP.
- **Ex** (extensie - un bit) - indică dacă sunt utilizate extensii ale protocolului în pachet.
- **CC** (patru biți) conține numărul identificatorului CSRC care îi urmează antetului fix.
- **M** (marcaj - un bit) este folosit la nivelul aplicație și este definit în cadrul profilelor. Dacă este setat, semnifică că datele curente au o semnificație specială pentru nivelul aplicație.
- **SU** (sarcina utilă - 7 biți) indică formatul payloadului și determină interpretarea de către aplicație.
- **SSRC** indică sursa de sincronizare.

Real Time Control Protocol

RTCP este succesorul protocolului RTP care ofera o buna raportare a statisticilor si informatii de control pentru un flux RTP.

Scopul si problemele pe care incerca sa le rezolve acest protocol:

- Transporta pachete de date in timp real;
- Audio conferinte multicast
- Audio and Video Conference
- Suporta Layered Encodings (aplicatia multimedia poate modifica rata de transmirea a pachetlor in functie de conexiune)
- QoS feedback (imbunatatire adusa la RTP)
- Controlul sesiunii (pachetul RTCP BYE – care anunta parasirea sesiunii a unui participant)
- Identificarea paticipantilor la sesiune – nume, e-mail si telefon (imbunatatire adusa la RTP)

Alte informatii pe care le ofera RTCP:

- Numarul de pachete trimise
- Pierderile de pachete
- Intarzierile de pachete Round-trip delay time
- Jitter (variata) intarzierilor
- Nume codec

Desi aceste informatii sunt utile, ele nu sunt critice (asa cum sunt cele transportate de protocolul RTP). De aceea, la configurarea de QoS se tine cont de asta.

Tipuri de mesaje RTCP:

- **Sender report (SR)** – Acest raport este trimis periodic de care senderii activi intr-o conferinta pentru verificare transmisiei
- **Receiver report (RR)** – verifica prezenta participantilor pasivi care nu trimis pachete RTP. Acest raport informeaza sendorul despre QoS.
- **Source description (SDES)** – trimite CNAME catre participantii: nume, e-mail, telefon etc.
- **End of participation (BYE)** - trimite un mesaj de inchiere al conexiunii
- **Application-specific message (APP)**

Compressed Real Time Protocol

cRTP, (RFC 2509) a fost creat pentru a micșora dimensiunea antetelor IP, UDP și RTP. Totuși, acesta a fost proiectat să funcționeze în conexiuni peer to peer sigure și rapide.

Scopul și problemele pe care încerca să le rezolve acest protocol:

- Identic cu RTCP dar cu dimensiunea antetelor micșorată
- Rata de livrare a pachetelor este îmbunătățită în rețele care au lățimea de bandă mică.

În mediu real, unde ar putea exista întârzieri mari, pierderi de pachete și pachete ajunse neordonat, cRTP nu funcționează cum trebuie pentru aplicații de tip VoIP. Pentru a soluționa această problemă s-a conceput un nou standard: Enhanced cRTP.

Dacă în realitate dimensiunea acestor antete este de 40 bytes, prin utilizarea cRTP ele sunt reduse la **2-4 bytes** (4 bytes în cazul în care compresia se face cu UDP checksum-(suma de control al erorilor) și 2 bytes dacă nu se face UDP checksum).

Acest protocol micșorează antetele "redundante" de la nivelul rețea și transport, prin păstrarea acestor informații în memoria cache a routerelor.

În fiecare pachet trimis numai e nevoie trimiterea adresei IP sau a numărului de port la nivelul de transport, rămânând același pe tot parcursul sesiunii. Informațiile menționate se trimit numai în primul pachet, iar restul de pachete se trimit cu compresie de header cRTP.

HTTP Live Streaming

HTTP Live Streaming cunoscut si ca HLS, este un protocol media streaming implementat de Apple Inc pentru softul QuickTie si iOS.

Scopul si problemele pe care incerca sa le rezolve acest protocol:

- Streaming, apeluri video/audio optimizat pentru mobile
- Protocol de nivel Aplicatie din stica TCP/IP
- Live Radio sau TV
- Cripoteaza pachete oferind securitate sesiunii (imbunatire adusa la RTP)
- Poate trece de firewall sau server proxy (imbunatire adusa la RTP)
- Se poate adapta in functie de conxiunea de internet

Acesta sparge intreg streamingul in pachete mici HTTP care vor fi download-ate. In timp ce fluxul este pornit, clientul poate sa aleaga rata de citire datelor in functie de conexiune. La inceputul sesiunii de streaming, clientul descarca un playlist M3U (m3u8) care conține metadatele pentru diferite sub-fluxuri care sunt disponibile.

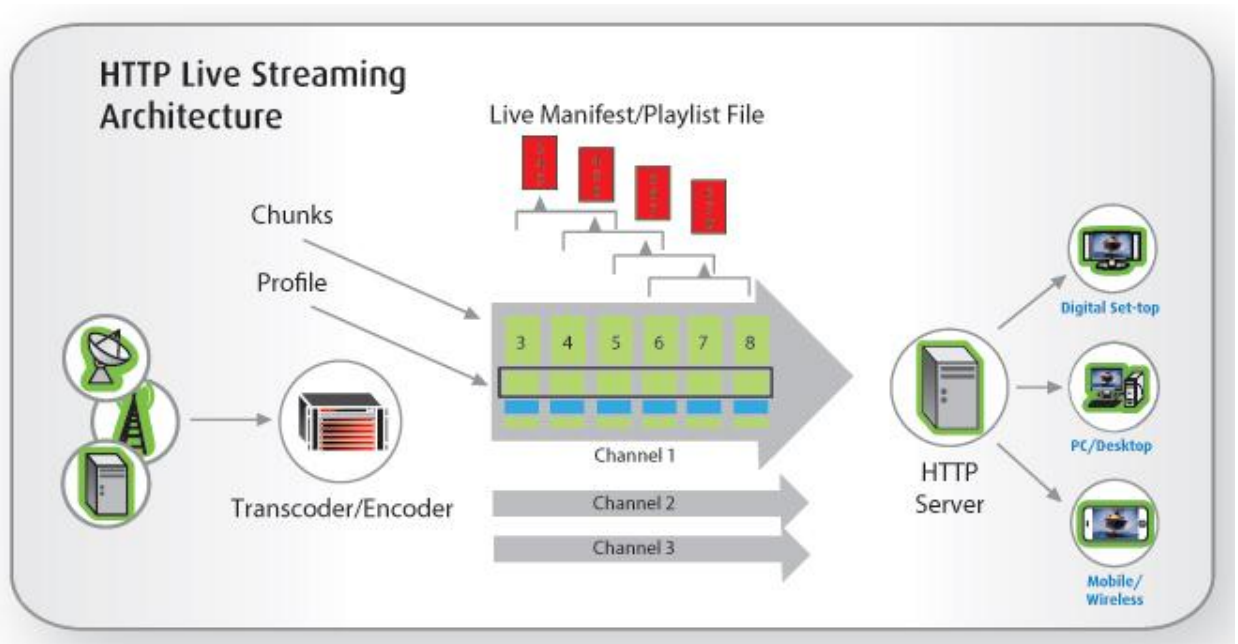
Tinand cont ca necesita numai tranzactii HTTP, acest protocol poate trece de orice firewall sau server proxy care permite trecerea traficului HTTP standard. Acest lucru reprezinta un avataj fata de RTP care este bazat pe UDP.

HLS prezinta un mecanism standard de criptare bazat pe AES si HTTPS (trimiterea cheilor). Ceea ce il face foarte sigur si greu de decriptat.

Mai tarziu acest protocol a primit comenzi de derulare inapoi, inainte, pauza si chiar posibilitatea de a adauga subtitrare.

Dezavantaje: la inceput a fost compatibil numai cu produsele Apple, dar cand a fost facut public (Mai 2009) si alti dezvoltatori il folosesc.

Arhitectura este similara cu *Smooth Streaming in Microsoft's Silverlight*



Cum functioneaza:

1. Fluxul de intrare este codificat. Acesta poate preveni de la un satelit sau chiar de la alt server. Continutul video si audio este codat intr-un MPEG-2 container de transport, cu H.264 video si AAC audio (codecuri suportate).
2. Se creaza un profil de output. Normal un sigur flux de intrare se va codifica in cateva fluxuri output cu rata de redare diferita (in functie de performanta device-ului sau latimea de banda a clientului).
3. Fluxul este segmentat. Pentru fiecare profil fluxul este segmentat si facut disponibil pentru device-ul de tip client.

Optional fiecare segment poate fi criptat pentru securitate.

Real Time Messaging Protocol

Real Time Messaging Protocol (RTMP) a fost initial dezvoltat de Macromedia pentru fluxuri video, audio si date dintre playerul Flash si server. Acum este proprietatea Adobe.

Scopul si problemele pe care incerca sa le rezolve acest protocol:

- Protocol de nivel Aplicatie din staca TCP/IP
- Proiectat pentru multiplexare și împachetare a fluxurile de de transport multimedia (cum ar fi audio, video și conținut interactiv)
- Poate transporta in paralel diferite tipuri de fluxuri audio, video sau data. (imbunatire adusa la RTP si HLS)
- Eticheteaza pachetele cu un ordin de prioritate (imbunatateste calitatea fluxului de date live)
- Se poate adapta in functie de conxiuna de internet

Tipuri de RTMP:

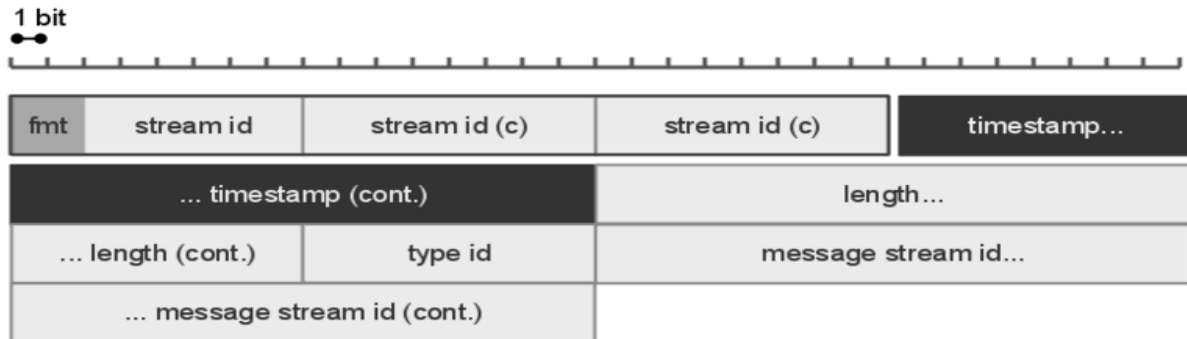
1. Protocolul „simplu” care functioneaza prin TCP port 1935 „default”.
2. RTMPS -> RTMP cu conexiune TLS/SSL.
3. RTMPE -> RTMP criptat cu un mecanism propriu de la Adobe.
4. RTMPT->acesta este incapsulat in requestul HTTP si poate sa treaca de firewall-uri.
Foloseste requesturi de tip text prin portul TCP 80 si 443.

RTMP este un protocol bazat pe TCP care mentine o conexiune stabile si permite o comunicare cu latent scazuta. Pentru a livra fluxul lin si cat mai multa informative, este divizat in segmente cu marime dinamica. Aceasta marime este negociata de server si in functie de latimea de banda. Marimea default este 64 byte pentru audio si 128 pentru video sau alte tipuri.

RTMP definește mai multe canale virtuale, pe care pachetele pot fi trimise și primite, și care funcționează în mod independent unul de celălalt. De exemplu, există un canal de soluționare a cererilor și răspunsurilor RPC, un canal de flux de date video, un canal de flux de date audio, un canal de mesaje de out-of-band de control (de negociere dimensiunea fragment, etc), și așa mai departe . In timpul unei sesiuni de tipice RTMP, mai multe canale pot fi active simultan. Când datele RTMP sunt codificate, este generat un antet de pachete. Antetul pachetului specifică, printre altele, id-ul de canal pe care urmează să fie trimis, un timestamp (dacă este necesar), și dimensiunea sarcină utilă pachetului. Acest antetul este apoi urmat de conținutul efectiv al pachetului, care este fragmentat în funcție de dimensiunea agreata înainte de a fi transmise prin conexiunea.

La un nivel mai înalt, RTMP încapsulează fișiere MP3 sau AAC audio și fluxuri video multimedia FLV1, și pot efectua apeluri de proceduri la distanță (CRP-urile). Orice serviciu RPC necesar este realizat asincron, folosind un singur model client / server de cerere / răspuns, astfel încât nu este nevoie de comunicare în timp real.

Structura pachetului RTMP:



Concluzii

Acest document prezinta pe scurt notiunea de streaming in timp real si cateva protocoale care stau la baza acestuia.

Primul protocol semnificativ si important de mentionat care trimite pachete de date in timp real a fost RTP-ul. Acesta a fost imbunatatit prin controlul sesiuni si QoS feedback (RTCP) si prin compresia antetelor (cRTP).

Alte doua protocoale care sunt folosite des in ziua de azi sunt HLS de la Apple si RTMP de la Adobe, care aduc imbunatatiri semnificative in ceea ce priveste modul de trimiterea al pachetelor si securitatea lor in retea.

Trebuie mentionat faptul ca ambele companii au publicat numai prima versiune a acestor protocoale si nu este cunoscuta in totalitate modalitatea de functionare.

Bibliografie:

- [1] Lucian Ioan, Graziela Niculescu, „*Calitatea servirii în rețele cu comutație de pachete*”, Ed. Matrix Rom, București, 2008
- [2] Eugen Borcoci, *Note de Curs „Arhitecturi si protocoale de comunicatii”*, Universitatea Politehnica Bucuresti, Facultatea de Electronica, Telecomunicatii si Tehnologia Informatiei
- [3] <http://ro.wikipedia.org/wiki/RTP>
- [4] <http://en.wikipedia.org/wiki/RTCP>
- [5] <http://reeves.csc.ncsu.edu/Classes/csc573/rtp-rtcp.pdf>
- [6] <http://www.cs.columbia.edu/~hgs/teaching/ais/slides/2003/RTSP.pdf>
- [7] Note de curs streaming <http://mihai.cm.upt.ro/Discipline/STV/Test3.pdf>