



Technology Scout

Afgeschermdde Live Streaming

Versie 1.2
Datum 3 februari 2010
SURFnet/Kennisnet Innovatieprogramma

Inhoud

Inhoud	2
Samenvatting	3
Opzet proof of concept.....	5
User stories	5
Uitgangspunten	5
Een live stream is conceptueel een media file	5
Gebruik maken van beschikbare streaming faciliteiten buiten MediaMosa	6
Autorisatie via MediaMosa.....	6
MediaMosa schermt streaming infrastructuur af voor EGA.....	7
Beheer van live streams wordt uitgevoerd in EGA.....	7
Vooraf geen controle van binnenkomende live stream	7
PoC infrastructuur	8
Uitbreidingen aan MediaMosa	9
Mountpoints voor binnenkomende streams.....	9
Interface met EGA's.....	10
Interface met streaming omgeving (LiSA)	10
Realisatie POC	11
Aanpassingen MediaMosa	11
Aanpassingen White Label Ega	11
Autorisatie in de WLE.....	13
Conclusies en Aanbevelingen.....	15
Opzet PoC.....	15
Afscherming live streams	15
Live streaming in MediaMosa.....	16

Samenvatting

MediaMosa is open source software waarmee een mediamanagement en distributieplatform kan worden ingericht. Het vormt de basis van tal van succesvolle videodiensten die SURFnet en Stichting Kennisnet aan het onderwijs in Nederland aanbieden.

Op dit moment is MediaMosa ingericht als videoplatform voor on demand video's. Om het mogelijk te maken dat (onderwijs)instellingen met een eigen video bron het materiaal live afgeschermd kunnen distribueren met behulp van de MediaMosa infrastructuur, is een Proof of Concept uitgevoerd, waarin de mogelijkheden hiervoor zijn onderzocht. In de Proof of Concept is een koppeling gemaakt tussen MediaMosa en het SURFnet live streaming platform (het SURFnet CDN), zodat MediaMosa gebruik kon maken van de streaming componenten van dit platform. Zo werd het mogelijk om live streaming functionaliteit aan EGA's aan te bieden.

Het kunnen aanbieden van zowel video on demand als live streaming video via MediaMosa, zorgt voor een integraal en gestructureerd aanbod van verschillende soorten videocontent. Het aanbod van leerobjecten binnen de MediaMosa infrastructuur wordt hiermee groter en gevarieerder. Dit past bij het uitgangspunt van variëteit in leren en biedt daarmee meerwaarde voor de hele onderwijskolom.

Uit de Technology Scout is naar voren gekomen dat het dit moment nog niet mogelijk is om binnen het SURFnet CDN om autorisatiegegevens mee te geven met een live stream. Dit betekent dat wanneer de URL met van een stream wordt doorgestuurd, het voor die persoon ook mogelijk is om de stream te bekijken. Autorisatie is wel mogelijk voor het opvragen en verkrijgen van de URL.

Inleiding

MediaMosa is open source software waarmee een mediamanagement en distributieplatform kan worden ingericht. Het vormt de basis van tal van succesvolle videodiensten die SURFnet en Stichting Kennisnet aan het onderwijs in Nederland aanbieden.

MediaMosa is beschikbaar op www.mediamosa.org. Iedere organisatie die een eigen videoportal wil ontwikkelen, kan MediaMosa gebruiken. Hiermee wordt het voor instellingen of andere organisaties die videodienstverlening willen aanbieden een stuk eenvoudiger om hun videoinfrastructuur in te richten.

MediaMosa is te beschouwen als een software toolkit en biedt tal van (mediamanagement)functionaliteiten die flexibel te gebruiken zijn. Zo kan bijvoorbeeld streaming video worden geïntegreerd met de eigen website, intranet of samenwerkingsomgeving. MediaMosa regelt vervolgens het beheer, de opslag, indexering en het afspelen van videomateriaal.

SURFnet en Kennisnet maken voor hun videodiensten zelf ook gebruik van MediaMosa. Diensten zoals SURFmedia en Teleblik zijn aangesloten op het VP-Core videoplatform, waar MediaMosa software wordt gebruikt.

Op dit moment is MediaMosa ontworpen voor het opslaan en afspelen van video on demand. Voor het ondersteunen van live streaming zal MediaMosa moeten worden uitgebreid met specifieke functionaliteit. Niet alleen moeten de live streams via MediaMosa beschikbaar zijn, de streams moeten ook door de juiste mensen met de juiste rechten te bekijken zijn.

In deze technology scout is onderzocht hoe live streaming ondersteund kan worden in MediaMosa. Door in de proof of concept (PoC) gebruik te maken van externe streaming componenten van het SURFnet live streaming platform (het SURFnet CDN), zijn de benodigde aanpassingen vooraf aan MediaMosa beperkt. Dit rapport beschrijft een voorstel tot wijzigingen aan MediaMosa en aan een voorbeeldapplicatie die gebruik maakt van de live streaming component.

Voor een definitief ontwerp van live streaming ondersteuning in MediaMosa zullen de ervaringen uit de PoC goed gebruikt kunnen worden. Op basis van de ervaringen en andere meer fundamentele afwegingen kan het ontwerp van MediaMosa worden aangepast voor het ondersteunen van afgeschermd live streaming.

Opzet proof of concept

User stories

Hieronder worden de user stories beschreven die van belang zijn voor het ondersteunen van live streams binnen MediaMosa.

- EGA meldt live stream aan
 - EGA maakt asset aan
 - EGA configureert live stream (bij asset)
 - MediaMosa geeft configuratie informatie door aan CDN
- EGA vraagt status live stream op
 - MediaMosa vraagt status CDN op
- EGA start live stream
 - EGA duwt data naar CDN
 - CDN trekt data van EGA
- CDN verzorgt distributie naar EGA('s)
 - CDN verspreidt streams (via unicast)
 - MediaMosa geeft geautoriseerde stream afnemers door aan CDN
 - CDN geeft kopie stream door aan MediaMosa
- EGA vraagt gebruiksgegevens op bij MediaMosa
 - CDN levert gebruikstatistieken aan MediaMosa

Uitgangspunten

Voor de ondersteuning van live streaming in MediaMosa zijn aanpassingen aan ontwerp en opzet nodig. In de PoC worden bepaalde aanpassingen en uitbreidingen aan MediaMosa geïmplementeerd. Hierbij wordt een aantal uitgangspunten gehanteerd die hieronder beschreven worden.

Deze uitgangspunten zijn niet perse geldig voor de definitieve ontwerpkeuzen. Bij de inrichting van de PoC omgeving en opzet is uitgegaan van beschikbare infrastructuur in het SURFnetlandschap. Op basis van ervaringen in de PoC (en andere overwegingen) kunnen in het vervolgtraject andere richtingen worden ingeslagen.

Een live stream is conceptueel een media file

In MediaMosa worden twee centrale concepten gehanteerd: assets en mediafiles. Een asset bevat (inhoudelijke) metadata die een media fragment beschrijft. Daaraan gekoppeld zijn mediafiles die hun eigen specifieke (technische) metadata bevatten.

Een live stream wordt in MediaMosa conceptueel afgebeeld als mediafile. Een live stream wijkt af van 'normale' files doordat er een URL en tijdsinterval worden toegepast in plaats van een filelocatie.

Gebruik maken van beschikbare streaming faciliteiten buiten MediaMosa

MediaMosa verzorgt de rol van middleware tussen EGA's en achterliggende hard- en software voor het opslaan en afspelen van media bestanden.

De PoC wordt uitgevoerd op het live streaming platform van SURFnet, een Content Distributie Network (CDN). Van dit CDN wordt gebruik gemaakt voor het binnen halen, verwerken en distribueren van live video streams die op dit CDN beschikbaar zijn. Aan MediaMosa zelf worden geen componenten toegevoegd voor dit soort taken.

Autorisatie via MediaMosa

In de architectuur van MediaMosa wordt de autorisatie van eindgebruikers afgehandeld door MediaMosa. EGA's zijn verantwoordelijk voor de authenticatie van (eigen) gebruikers. Dit betekent dat MediaMosa in het geval van live streaming verantwoordelijk is voor:

- toegang tot uitgaande streams (kijken door stream afnemers);
- de toegang voor binnenkomende streams (aanbieden stream door broadcaster).

Het opvragen van een live stream verschilt in concept niet met het opvragen van een 'normale', on demand, video file. De huidige methode voor afscherming binnen MediaMosa zal als basis worden gebruikt voor de afscherming van live streams. Wat wel verschilt is dat de autorisatie gegevens moeten worden overgedragen naar het achterliggende streaming platform.

Wanneer een gebruiker een live stream opvraagt in een EGA, zal dit verzoek door de EGA aan MediaMosa worden doorgegeven. MediaMosa controleert of de desbetreffende gebruiker voldoende rechten heeft om de stream te mogen zien. Als dat het geval is, roept MediaMosa het achterliggende CDN aan.

Deze aanroep is in de huidige implementatie van de CDN managementinterface nog niet beschikbaar. Om afscherming, gebaseerd op MediaMosa, in de PoC mogelijk te maken, zal de CDN interface moeten worden uitgebreid met de mogelijkheid de autorisatie over te dragen.

In de PoC zal er geen voorziening worden getroffen voor het daadwerkelijk afschermen van de binnenkomende streams. Een EGA gebruiker mag een live stream aanmaken zonder dat hem daarvoor een specifiek recht is toegekend.

De keuze om MediaMosa verantwoordelijk te maken voor de toegang van zowel leverancier als afnemer van streams betekent dat onderliggende systemen (zoals in dit geval het CDN) MediaMosa hierin moeten vertrouwen.

MediaMosa schermt streaming infrastructuur af voor EGA

EGA's die live streams aanbieden of afnemen worden daarbij door MediaMosa gefaciliteerd. Bij het afnemen van streams (bekijken door eindgebruikers van de EGA) zal MediaMosa de achterliggende infrastructuur afschermen. Voor een EGA zal er geen verschil zijn tussen het opvragen van een 'normale' asset en een asset met een live stream.

Bij het aanbieden van een live stream vanuit de EGA zal deze afscherming minder zijn omdat het CDN informatie nodig heeft over de live stream voor de verwerking hiervan. Het aanmaken van een live stream zal wel via de normale REST interface tussen EGA en MediaMosa worden ondersteund.

Beheer van live streams wordt uitgevoerd in EGA

EGA's worden door MediaMosa gefaciliteerd bij het aanbieden van media functies aan hun eindgebruikers. De EGA's zullen in het geval van streaming video gebruikers in staat stellen de streams te bekijken, maar ook deze aan te bieden. Dit betekent dat MediaMosa deze functies aan moet bieden via de interface voor EGA's.

Het opvragen van live media streams via de REST interface zal weinig verschil kennen met het 'normale' opvragen van on demand video fragmenten. In de afhandeling is er wel verschil. MediaMosa zal (naast de normale controles) ook testen of een aanvraag op een correct moment (binnen het ingestelde tijdsinterval) binnenkomt. Daarnaast dient een geldige aanvraag doorgezet te worden naar de achterliggende omgeving (in dit geval CDN).

Het opzetten van een live stream vindt ook plaats in de EGA. De informatie wordt via de REST interface aan MediaMosa doorgegeven, die op zijn beurt deze aan de achterliggende streaming omgeving doorgeeft.

De informatie over de live stream wordt in MediaMosa vastgelegd en bevat zowel informatie uit de EGA (gebruiker, tijdschema, metadata, etc.) als uit de streaming omgeving (URL waarop stream beschikbaar komt, mountpoint, etc.).

Vooraf geen controle van binnenkomende live stream

In de PoC wordt na het aanmaken van een streaming mountpoint geen controle uitgevoerd door MediaMosa op formaat en kwaliteit van de stream. De achterliggende streaming omgeving krijgt rechtstreeks contact met de aanbiedende partij bij het doorgeven van de stream. Wanneer de stream niet in het verwachte formaat wordt geleverd (of een ander mankement wordt geconstateerd) zal de CDN streamer de inkomende stream niet accepteren en wordt de keten verbroken. MediaMosa heeft op deze beslissing geen invloed.

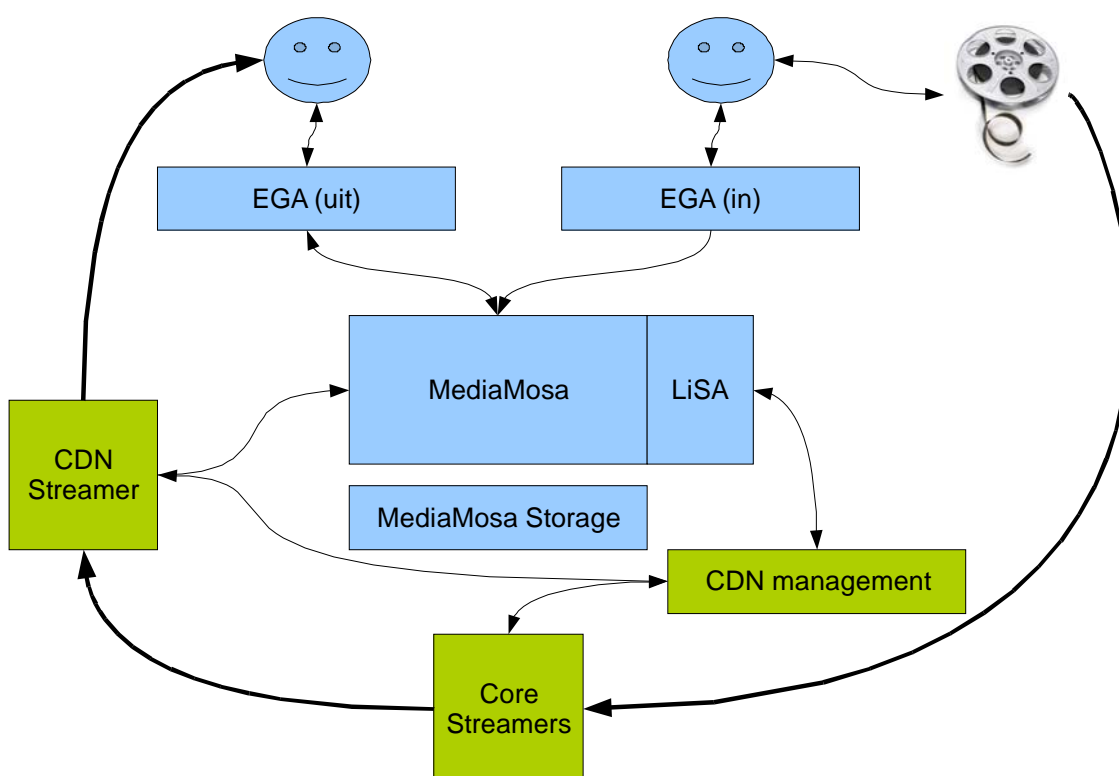
Deze opzet schendt het principe dat MediaMosa materiaal vooraf analyseert en controleert. In het geval van on demand video wordt materiaal geüpload naar MediaMosa, waar het vervolgens geanalyseerd (en getranscodeerd in bepaalde gevallen) wordt. De door de EGA/eindgebruiker opgegeven technische metadata wordt daarbij overruled door de uitkomsten van de analyse.

Hierdoor kan MediaMosa een zeker kwaliteit- en beschikbaarheidsniveau garanderen bij het opvragen en afspelen van on demand video files. In de opzet zoals in de PoC toegepast, heeft MediaMosa nauwelijks invloed op kwaliteit en beschikbaarheid van live streams.

Een alternatief voor de PoC opzet zou een validatie vooraf van de live stream zijn. MediaMosa zou test assets met bijbehorende test media files aanbieden aan EGA's. Na het aanmaken van een mountpoint en voorafgaand aan het streamen van de daadwerkelijke content, zou de EGA een stream kunnen starten op basis van de test media files. Pas na deze validatie zou een EGA een live stream mogen aanbieden aan eindgebruikers.

PoC infrastructuur

De uitbreiding van MediaMosa met live streaming wordt getest met behulp van de live streaming omgeving van SURFnet¹. Deze opzet is specifiek voor de PoC en deze omgeving. De koppelingen en informatiestromen tussen EGA's, MediaMosa en de SURFnet omgeving worden hieronder beschreven.



Figuur 1 Opzet live streaming in MediaMosa

¹ Op basis van het product VDO-X van <http://www.streamzilla.nl/>

Figuur 1 (Opzet live streaming in MediaMosa) geeft een hoog niveau overzicht van de opzet van live streaming in MediaMosa. In blauw zijn de MediaMosa componenten weergegeven, in groen de SURFnet CDN onderdelen.

In het MediaMosa deel is een nieuwe component toegevoegd, de Live Streaming API (LiSA), die een abstractielaag gaat vormen tussen MediaMosa en live streaming componenten. In de PoC zal deze abstractielaag minimaal worden geïmplementeerd en zich beperken tot de (SOAP) calls naar de CDN management server.

Live streaming start bij het aanmelden van de live stream in MediaMosa door de EGA (in). MediaMosa biedt hiervoor nieuwe functies via de gebruikelijke EGA REST interface. MediaMosa zet op basis van deze informatie een live stream mountpoint op binnen het CDN. De informatie hierover wordt opgeslagen in de MediaMosa database, als metadata bij de media file.

MediaMosa geeft aan de EGA de benodigde informatie om rechtstreeks de upload stream aan de CDN Core streamers te kunnen aanbieden. Afhankelijk van de opzet zal dit een pull of een push methode zijn

De verwerking van de stream en het doorsturen daarvan via de CDN streamer(s) naar afnemers wordt door CDN componenten uitgevoerd. MediaMosa regelt dit proces via de CDN-managementkoppeling in en schermt af voor EGA's.

Uitbreidingen aan MediaMosa

Mountpoints voor binnenkomende streams

Van binnenkomende streams moeten gegevens worden genoteerd, afhankelijk van toegepaste technologie (Windows media, Flash/H.264), methode (push of pull) en dergelijke. MediaMosa zal deze data moeten registreren, waarvoor vier alternatieven zijn:

- in metadata van mediafile
- in mountpoint tabel
- in het upload ticket
- in nieuwe structuur

In de PoC zal deze informatie worden opgeslagen in de technische metadata van media files, omdat dit zonder grote aanpassingen eenvoudig te realiseren is. Na afloop van de PoC zal hier een definitieve (conceptuele) keuze in worden gemaakt.

Interface met EGA's

De interface tussen EGA en MediaMosa zal de mediafile/create operatie worden aangepast om live streams te kunnen aanmaken. Na een aanroep van deze operatie door een EGA zal MediaMosa via de nieuwe LiSA interface een live stream aanmaken en configureren in de achterliggende SURFnet live streaming omgeving.

Interface met streaming omgeving (LiSA)

Dit is een nieuwe component in MediaMosa die een communicatielaag vormt naar achterliggende streaming infrastructuur. Deze uitbreiding is nodig om de specifieke implementatie van functionaliteit van streaming componenten af te schermen voor MediaMosa, zodat in andere toepassingen andere hard- en software kan worden toegepast.

Via deze interface zal MediaMosa de live streaming infrastructuur aansturen. De volgende operaties zijn daarbij nodig:

- Aanmaken live stream: Het aanmaken van de live stream, waarbij informatie over de koppelmethode wordt uitgewisseld
- Opvragen status live stream: Opvragen over de status (in de streaming omgeving) van de live stream
- Afnemer toevoegen: Doorgeven aan de streaming omgeving van één of meer afnemers
- Opvragen gebruikstatistieken: MediaMosa vraagt gegevens over het gebruik (aantallen kijkers, etc.) op aan de streaming omgeving.
- Opvragen stream kopie: Een EGA kan aangeven dat een live stream na het uitzenden beschikbaar moet worden binnen MediaMosa. In dat geval haalt (na afloop van het doorgeven van de live stream) MediaMosa dit bestand op.

Voor deze laatste operatie is het de vraag hoe deze geïmplementeerd moet worden. De externe streamer moet in staat zijn een live stream als één bestand aan MediaMosa aan te bieden of MediaMosa moet worden uitgebreid met functionaliteit om een live stream te ontvangen en live te encoderen.

Realisatie POC

De voorgestelde wijzigingen in het vorige hoofdstuk hebben geleid tot aanpassingen in de MediaMosa core en in een aangepaste White Label Ega (WLE). De demo applicatie is toegepast op <http://livestreaming.mediamosa.surfnet.nl>.

VDO-X van Streamzilla kan in de SURFnet versie o.a.² de volgende formaten aan:

- Windows Media Streaming
- Flash Streaming over Wowza Media servers, voor rtmp, rtsp en http

Aanpassingen MediaMosa

Er zijn aanpassingen in MediaMosa gedaan voor opslaggegevens van de stream en voor de connectie. De Mediafile tabel in de database heeft een extra "stream_id" naast "uri" gekregen wat het interne stream id bij VDO-X identificeert.

Per client application is er de mogelijkheid gekomen om de connectie settings met VDO-X op te geven, hiertoe is aan de client table toegevoegd: vdox_url, vdox_user, vdox_password. Client applications kunnen dezelfde login account van VDO-X delen. Om vervolgens geen onduidelijkheid te krijgen in het stream id, wordt deze geprefixed met het applicatie id.

De Playproxy (module van MediaMosa die de afspeelobjecten verzorgt), is aangepast om de juiste livestream objecten te kunnen serveren. Voor zowel de windows streaming als de flash streaming was een ander objectcode noodzakelijk. Momenteel is de objectcode in de playproxy module opgenomen, maar dit dient (net als de overige objectcodes) configureerbaar gemaakt te worden in de MediaMosa beheermodules. Voor het wmv object volstond een aangepast html object, voor het flv object was ook nog een nieuwere jwplayer³ noodzakelijk. MediaMosa werkt nog met de oudere, maar nog opensource, versie van de jwplayer. Het object van de live streaming servers werkt alleen met nieuwere versies van deze player.

Aanpassingen White Label Ega

De Upload van de WLE is aangepast voor deze PoC; De aanpassingen zoals ze in de WLE zijn uitgevoerd zijn basaal en dienen slechts ter illustratie van het concept.

² <http://www.streamzilla.nl/content-delivery-services/enterprise-content-delivery/tech-specs-1>

³ <http://www.longtailvideo.com/players/jw-flv-player/>

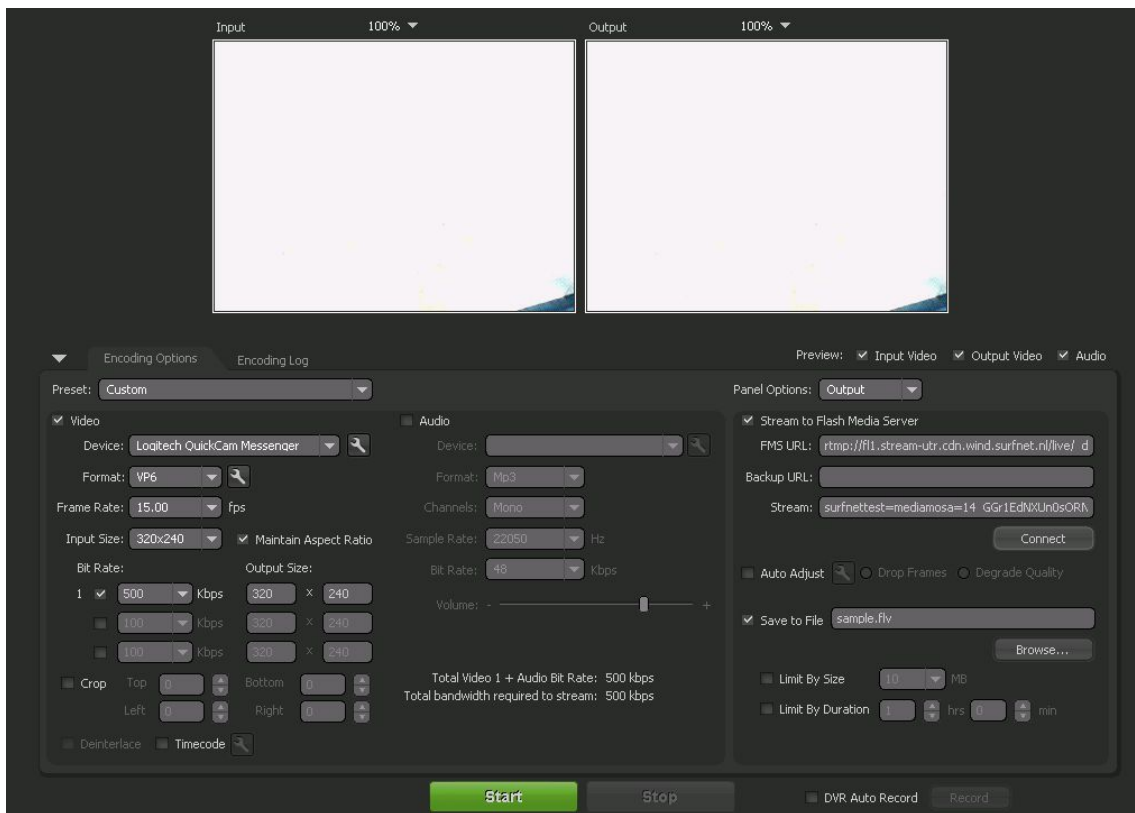
Een meer definitieve uitwerking zal meer informatie aan eindgebruikers moeten leveren omtrent de opzet van de encoder en de configuratie ervan. Het zou te adviseren zijn om een 'wizard' model hiervoor te gebruiken.

The screenshot shows a web form titled "Stream". At the top, there is a section labeled "Your Stream URL:" with an empty text input field. Below this, a small note reads: "You can upload an external stream URL. Leave it empty, if it is a Wowza stream". Underneath is a section labeled "Type of the stream:" with a dropdown menu. The dropdown is currently open, showing three options: "External" (highlighted in orange), "Soap: Windows Media", and "Soap: Wowza". To the right of the dropdown, there is a small text label: "normal external stream or a soap live stream". At the bottom of the form is a blue "Submit" button.

Figuur 2 aanmaken Live stream in een eindgebruiker applicatie

De 'External' is de bestaande functionaliteit van de WLE dat resulteert in de opslag van een externe URI. De 'Soap: Windows Media' en 'Soap: Wowza' zijn de twee uitgewerkte livestreams van VDO-X. VDO-X biedt voor beide streaming oplossingen een push en een pull oplossing. De encoders zijn in één van de twee versies uitgewerkt.

- Windows Media Streaming;
 - Dit is voor eindgebruikers de meest transparante mogelijkheid. Bij stream URL kan een URI worden opgegeven naar de eigen stream, waarna 'submit' de URI vanuit de EGA naar MediaMosa wordt verstuurd. MediaMosa 'push'-ed vervolgens de stream naar VDO-X. Na enige tijd (VDO-X neemt de tijd om de stream te verwerken) is de stream beschikbaar. Zie voor onze teststream: <http://livestreaming.mediamosa.surfnet.nl/asset/IQzm7f9vqGkg6GcUWftXOFQi> (beperkt beschikbaar).
 - In de PoC is voor Windows Media Streaming de Pull methode vanuit VDO-X uitgewerkt, niet de Push vanuit de encoder.
- Flash Streaming over Wowza Media servers, voor rtmp, rtsp en http;
 - Flash streaming wordt ondersteund met "Adobe Flash Media Live Encoder". Er wordt rtmp, rtsp en push aangeboden. Tijdens de PoC gaf rtmp en rtsp een fout in VDO-X, maar push leverde een werkende applicatie op. De eindgebruiker moet in dit geval zelf de stream bij VDO-X aanmelden; Na submit (alleen 'Soap:Wowza' aanklikken, zonder url) levert MediaMosa een stream_id terug. Deze stream_id moet de eindgebruiker zelf bij de VDO-X server aanmelden in de "Adobe Flash Media Live Encoder" (zie figuur 3).
 - In de PoC is voor Flash streaming de Push methode vanuit de encoder uitgewerkt, niet de Pull vanuit VDO-X.



Figuur 3 aanmaken Live stream in Flash Media Live Encoder

Na enige tijd (minuten) is de stream afspeelbaar via de WLE.

Een test met ffserver⁴, wat ook geschikt is om Flash te streamen leverde geen resultaat op (Melding 'Error' in VDO-X), en wordt helaas ook niet officieel ondersteund. Het verdient aanbeveling om bij de leverancier meer aandacht te vragen voor debugging faciliteiten en Open Source Streaming oplossingen.

Autorisatie in de WLE

Ten behoeve van de afschermingstest is in de WLE de standaard 'domain' autorisatie aangezet op mediafiles. Hiermee kan een eigenaar van een mediafile op basis van de domeinnaam mediafiles afschermen.

⁴ <http://ffmpeg.org/ffserver-doc.html>

Available mediafiles | [add stream](#) | [delete all stills](#)

Orig.  Stream [Watch](#) |  | 

Your host is: `madcaphq.xs4all.nl`

Figuur 4 Mediafile details in de WLE

Na kiezen van de 'i' levert het technische metadata scherm een tekstveld op waarmee de stream op domainnaam kan worden afgeschermd:

ACL

Figuur 5 toevoegen van acl regel

Invoeren van 'surfnnet.nl' schermt de stream af voor iedereen behalve clients uit het domein surfnnet.nl. In de WLE is het wel zo dat de eigenaar (de persoon is ingelogd) de video altijd mag zien, dus een test hiermee moet met een andere browser (zonder in te loggen) of in dezelfde browser door uit te loggen uitgevoerd worden.

Conclusies en Aanbevelingen

MediaMosa biedt op dit moment nauwelijks ondersteuning voor live streams. Er wordt uitgegaan van on demand opvragen en doorgeven van video materiaal. Er zijn een aantal aanpassingen aan ontwerp en opzet nodig om live streaming mogelijk te maken.

In de PoC is MediaMosa uitgebreid met functies voor het aanmaken, configureren en distribueren van live streams via het SURFnet CDN. De nadruk ligt daarbij op het implementeren van een koppeling tussen MediaMosa en CDN en het aanbieden van streaming video functionaliteit aan EGA's.

Opzet PoC

In de PoC is gekozen voor een opzet waarbij gebruik wordt gemaakt van het CDN als al beschikbare streaming infrastructuur. Het eigenlijke streamen zelf, het verwerken van de binnenkomende stream en het doorgeven daarvan aan eindgebruikers, wordt door deze infrastructuur afgehandeld.

MediaMosa is via een nieuwe interface component, de Live Streaming API (LiSA), gekoppeld aan het SURFnet live streaming platform. Via de LiSA communiceert MediaMosa gegevens over mountpoints, de koppeling tussen stream aanbieder en het streaming platform en de toegang tot de streams. Andersom ontvangt MediaMosa de locatie (URI) van streams en gegevens over gebruik terug van het platform via deze interface.

In de PoC implementatie van LiSA zijn alleen de minimaal benodigde functies voor koppeling met het huidige streaming platform uitgewerkt. Dit betekent ook dat het SOAP protocol is toegepast (in plaats van REST koppeling zoals toegepast op andere plaatsen binnen MediaMosa) omdat er (nog) geen REST interface beschikbaar is.

Afscherming live streams

Het opvragen (bekijken) van live streams wijkt conceptueel niet af van het opvragen van een 'normale' on demand video fragment. Het huidige MediaMosa mechanisme voor het opvragen en afschermen is als basis genomen voor de PoC met live streams. Er zijn aanpassingen gedaan in het serveren van de html objectcode van de videoplayer. Het afspelen en afschermen van live streams wijkt wel af doordat hier gebruik wordt gemaakt van streamers binnen een ander systeem, het SURFnet live streaming platform. De huidige API van dit platform kent (nog) geen operaties voor het doorgeven van de autorisatie gegevens.

Om de MediaMosa afscherming toe te kunnen passen zal deze functionaliteit eerst moeten worden toegevoegd aan het CDN. De sterkte van de afscherming haalt nu zijn kracht uit de afscherming van MediaMosa en de correcte werking van het afschermingstoken van VDO-X.

Live streaming in MediaMosa

MediaMosa heeft al een beperkte ondersteuning voor het afspelen van live streams. Bij mediafiles kan in plaats van de locatie van de file op het filesysteem een URI worden opgegeven. Via deze URI kan verwezen worden naar een video stream, waarbij ook een geldigheidsinterval (in de tijd) kan worden aangegeven.

Voor het verwerken van een binnenkomende live stream bestaat geen ondersteuning in MediaMosa. In de PoC zijn de accountgegevens van het CDN middels de beheerinterface te configureren per EGA, waarbij meerdere EGA's hetzelfde account kunnen delen. In de uiteindelijke uitwerking van het LiSA component dienen voor de EGA REST-calls beschikbaar te komen.

De PoC biedt geen antwoord op vragen over in hoeverre de integratie van live streams in MediaMosa consistent moet zijn met die van on-demand streams. In de opzet van de PoC verloopt de flow van de stream in een 'u-bocht' rond MediaMosa. Gezien het feit dat MediaMosa zich concentreert op media management en niet op het streamen zelf, is dit geen bezwaar.

Conceptueel wordt een live stream een speciaal geval van een media file bij een asset. Het MediaMosa ontwerp blijft daarmee in stand. Wel wordt een belangrijk uitgangspunt van MediaMosa, de controle van media vooraf, geschonden door deze opzet.