IMS SECURED CONTENT DELIVERY OVER PEER-TO-PEER NETWORKS

Jens Fiedler, Thomas Magedanz and Alejandro Menendez
Fraunhofer Institute for Open Communications Systems - FOKUS, Kaiserin-Augusta Allee 31, Berlin, Germany

Keywords: IMS, Peer-to-peer, P2P, DRM.

Abstract: Effective content distribution, which is safe against denial of service attacks, is one of the greatest challenges for the content and service providers. Peer-to-peer technologies are known to be unaffected by such attacks, but lack any control by content owners or copyright holders. The work presented in this Paper combines the effective and reliable content availability, known from P2P, with the capabilities of IMS, which is used for access control, charging and service discovery. Commercial use cases are discussed for content consumption and provisioning.

1 INTRODUCTION

In todays internet, a large variety of services is offered by various parties. Most of the observed traffic arises from peer-to-peer (P2P) applications, like KaZaA (KaZaA, 2006), SIPShare (SIPshare, 2006) or BitTorrent (Cohen, 2003), and increases constantly. The acceptance of P2P services come from the two facts, that content is free of charge and no central instance exists, which could control access. Content is shared between users, not offered by central points of service. P2P has proven that it is a reliable, highly scalable technology, something that vendors wish for their own content distribution, if just the content would be secured. Therefore, a way to secure content in a P2P environment has to be found, which takes the benfits from P2P content distribution, but guarantees that only authorized users/customers can access the content in terms of the purchased rights on it.

The proposed approach, which is presented in this paper, combines three elementary technologies from todays computer science and research in network communication. Namely P2P, Digital Rights Management (DRM) and the IP Multimedia Subsystem (IMS) (3GPP, 2006c), specified by the 3GPP (3GPP, 2006a), to realize a safe, but efficient and controllable way to deliver content, manage licenses and unburden file servers.

Section 2 will give an impression of chances arising from the combination of these. In section 3 the state of the art technologies are described, that this work is based on. The proposed architecture is described in detail in section 4. Section 5 discusses two common use cases, while section 6 and 7 summarize the work and give an outlook for future work.

2 MOTIVATION

In P2P technologies, an independent content distribution system is available. Reliability of services and the high availability of contents are the experienced benefits from this technology in the existing applications. BitTorrent is already used by commercial vendors to distribute content to their customers, as it unburdens file servers, from which users would normally download their content. Currently, content distribution in P2P networks lacks any control by content creators, providers or copyright holders.

On the other hand appropriate DRM technologies for protecting content and controlling access by authorized customers to it, like (Microsoft, 2006)(DigiCont, 2006) or (Object-Lab, 2006), exist and are widely discussed (Safenet, 2006)(DRM-Watch, 2006)

Fiedler J., Magedanz T. and Menendez A. (2007).
IMS SECURED CONTENT DELIVERY OVER PEER-TO-PEER NETWORKS.
In *Proceedings of the Second International Conference on Signal Processing and Multimedia Applications*, pages 5-12
DOI: 10.5220/0002137600050012

Table 1: Technology Assignments.

Challenge	Technology
Effective content distribution	Peer-to-peer
File server unburdening	Peer-to-peer
Secure content	DRM
legal usage	DRM
Access control	IMS
Charging	IMS
Service enabling	IMS

Licenses, which are shipped separately from the encrypted content, are used to make it accessible to the customer by the terms of the license agreement, i.e. the purchased rigths on it.

IMS is known as a secure signalling infrastructure for multimedia control. Therefore, IMS is the ideal convergence point to combine P2P content distribution with DRM driven content licensing. Content can be encrypted and only authorized users will be able to decrypt it. Licenses, cryptographic keys and their distribution can be managed by the IMS.

Violation of the copyright holders rigths has always been one of the greatest problems in P2P systems. Approaches for integrating IPR management into P2P have been performed (Rosenblatt, 2003)(Pfeiffer et al., 2006). Looking at the value chain indicates a landscape consisting of the following players.

Content Providers represent the legal owner of content and therefore the party which is interested in a secure and legal distribution. It does usually not care, how the content reaches its customers. Content Providers expect to receive billing information regarding the use of their content.

Content Consumers want to receive content and be enabled to experience it (hear, watch, store, etc. it). They expect it to be available when they demand it and they usually do not care whether the content has to be secured by some means.

Service Providers give the requested services (s.a.) to both, content providers and consumers. They have to take care of securing the content, make sure it is available, enable the legal consumer to use the content and forward charging information.

Taking all this together, it is clear, that the answer to those challenges is a combination of the already mentioned technologies. Table 1 shows the assignments for each technology.

3 RELATED WORK

3.1 BitTorrent

BitTorrent (BT) is a P2P file transfer protocol. BT was designed with the aim to distribute huge files in a cheap and fast manner, means to unburden the uploader. The experienced performance of if (Pouwelse et al., 2005) and its open source nature make it interesting for developers. The first node which has the initial copy of the file is called *seeder*. Itself and the other peers, which are interested in the same file are called *swarm*. The file which is going to be distributed is split into pieces and the seeder uploads different pieces to different peers in the swarm. Thus the first copy is split over the swarm, but none of the peers (except the seeder) have a full copy of it. In the following steps, the peers in the swarm can interchange missing pieces to complete their copy of the file. This happens without any further interaction with the seeder, but the seeder can help with the upload by acting as normal peer in the swarm. New peers entering the swarm can download pieces potentially from all the peers simultanously, therefore not burdening one single data source for the full file, but multiple for small pieces. This mechanism is depicted in figure 1.

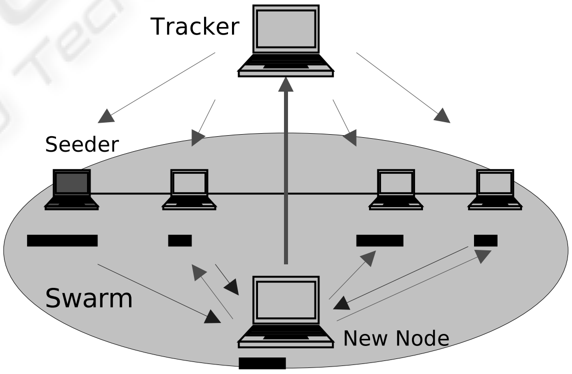

Figure 1: BitTorrent piece distribution.

Before a peer can enter the swarm, it must locate the other peers. For this the tracker and the torrent file exist.

The **torrent file** identifies the shared file, name, size, and hash values for each piece, which allow a downloader to verify the consistency of each piece that it downloaded. It also holds the URL of a tracker, which must be contacted to enter the swarm.

The **tracker** is the relevant network resource, which knows about the members of a swarm. A new node enters the swarm by contacting the obtained tracker, which adds the new node to the list of peers. This is then given to the node, which can now contact other nodes in the swarm for pieces.

The torrent file and the tracker make BT a semi-decentralized P2P system, as those are necessary for content location. This fact makes BT a very interesting technology for combining centralized architectures with fully decentralized architectures.

3.2 OpenIPMP

OpenIPMP is an open-source project developed by Mutable (Object-Lab, 2006). The first version consisted of user-authenticating DRM technology for the MPEG-4 (MPEG, 2006) codec that used MPEG-4 IPMP (Intellectual Property Management and Protection), a way of binding rights metadata to content and supported the ODRL (ODRL, 2006) and MPEG REL rights expression languages (RELs). The new version 2.0 (released in 2006) consists of core plugins for encrypting and protecting media content offering compatibility with the following specifications.

- ISMA (ISMA, 2006) with AES (Advanced Encryption Standard) (Daemen and Rijmen, 1999)
- OMA (OMA, 2006) DRM with AES
- OpenIPMP with AES and Blowfish (Schneier, 1994)

In order to communicate protection details to the DRM server, their SDK implements two messaging systems, OpenIPMP messaging system (used by ISMA and OpenIPMP DRM) and OMA messaging system as defined by OMA DRM. MPEG-4 and MPEG-2 format protection standards are supported as well. Working with MPEG, ISMA and other organizations, OpenIPMP provides a practical vision of the state of the art with respect to open standards based DRM technology by developing its reference implementation.

3.3 IMS

The IMS is defined by the 3GPP industry forum for 3G mobile phone systems in UMTS networks. IMS first appeared in the release 5 when SIP (Rosenberg et al., 2002) was added as the signalling protocol for establishing multimedia sessions. The IMS makes use of protocols defined by the Internet Engineering Task Force (IETF). 3GPP collaborates with the IETF adapting Internet protocols to IMS and with the OMA standardizing service enablers on top of IMS. IMS was originally conceived to bring Internet services to mobile users adding important features not usually present on the Internet: Quality of Service (QoS), charging, security and integration of different services. The IMS takes care of QoS provision when users establish a session. It also helps operators to apply different and flexible charging schemes to the user when they establish multimedia sessions. The IMS defines standard interfaces to be used by services developers and enables seamless integration of services, personalizing and enriching the user's experience. It is out of the scope of this work to cover all aspects of the signalling and media plane of the IMS, for that reason this overview will be centred on the most relevant aspects of IMS and those which will be useful for understanding the development of the final application.

The IMS defines functions, linked by standardized interfaces. Functions can be implemented as a single node but could also be grouped into one node or split into more. Figure 2 depicts the architecture of the IMS and relations (reference points) between selected IMS components. IMS supports different types of devices and access technologies, therefore the user terminal can also be a mobile device or personal computer using i.e. an ADSL connection.

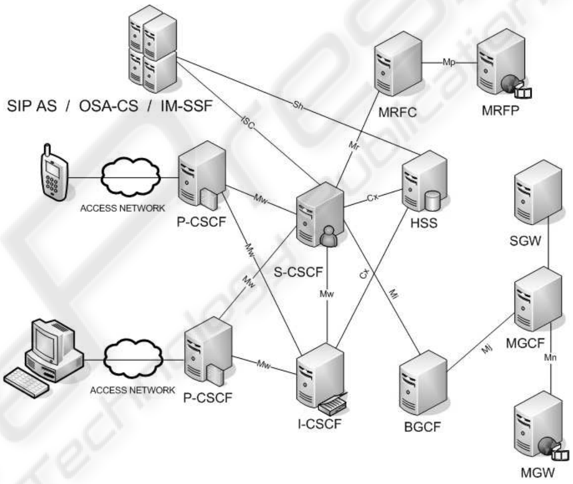

Figure 2: IMS architecture.

The **P-CSCF** acts as the point of connection between the IMS terminal and the IMS network. All the requests destined for the IMS terminal or originated at the IMS terminal are going to traverse the P-CSCF. During the process of registration one of the P-CSCFs of the IMS network is allocated to the IMS terminal and does not change for the duration of the registration. Being the outbound/inbound proxy server, the P-CSCF has to meet some security requirements. For that reason, it establishes two IPsec security associations with the IMS terminal. Furthermore, the P-CSCF asserts the identity of the user to the rest of the nodes, so that the other nodes do not need to authenticate user's identity again. The P-CSCF performs also compression and decompression of SIP messages, when the messages between the terminal and the P-CSCF are sent over a narrowband channel. In addition, the P-CSCF verifies the correctness of SIP requests sent by the IMS terminal. The Policy Decision Function (PDF) may form part of the P-CSCF. The PDF (not depicted) manages QoS over the media plane and authorizes media plane resources as well.

I-CSCF: In order to find the next SIP hop for a certain message, the SIP server obtains the address of an I-CSCF of the destination domain. Its address is available in the DNS records of the domain. The I-CSCF works as proxy server routing the SIP request to the appropriate destination (normally an S-CSCF). To find out the address of the next hop (e.g. the S-CSCF allocated to the user) the I-CSCF retrieves user location information from the HSS (and SLF if necessary) using Diameter over the Cx interface.

The **S-CSCF** is the central element of the signaling plane. The S-CSCF is a SIP server that acts as registrar and performs session control as well. It maintains a binding between the user location (e.g., the IP address of the terminal in use) and the user's SIP address of record (also known as a Public User Identity). Therefore, all the SIP signalling, that the IMS terminal sends or receives, traverses the allocated S-CSCF. Because all the signalling traffic traverses the allocated S-CSCF, it is capable of performing various control session tasks. It inspects every SIP message and determines whether the SIP signalling should visit one or more ASs, which might provide a service to the user. It keeps users from performing unauthorized operations, enforcing the policy of the network operator. It provides routing services, e.g., the user dials a number instead of a SIP URI and the number needs to be translated into a SIP URI. The S-CSCF needs to obtain user-related information from the HSS, consequently, it implements a Diameter interface to it. If a user wants to access the IMS, the S-CSCF downloads authentication vectors from the HSS to authenticate this user. Moreover, the S-CSCF also downloads the user profile from the HSS, which includes the service profile. The service profile lets the S-CSCF know when a SIP message should be routed through one or more Application Servers. Finally, when a S-CSCF is allocated to a certain user (for the duration of the registration) the HSS is informed by that S-CSCF.

The **Home Subscriber Server (HSS)** is an evolution of the Home Location Register (HLR) present in GSM networks. All the user-related subscription data required to establish multimedia sessions is stored in this central repository. The most significant items of information include location information, security information (authentication and authorization information), user profile information (e.g. the services the user is subscribed to), and the S-CSCF allocated to the user.

The **Application Server (AS)** hosts and executes services interfacing the S-CSCF using SIP and optionally the HSS. The AS can operate in SIP proxy mode, SIP User Agent (UA) mode, or SIP Back to Back UA (B2BUA) mode. SIP AS (Application Server): this is the native Application Server that hosts and executes IP Multimedia Services based on IP. This type of server will be used to develop the final application as it is expected that new IMS services will be developed in this way. The OSA-SCS (Open Service Access- Service Capability Server) provides an interface to the OSA framework application server. On one side the AS is interfacing the S-CSCF and on the other there is an interface between the OSA AS and the OSA Application Programming Interface. The IP Multimedia Switching Function (IM-SSF) enables to reuse CAMEL (Customized Applications for Mobile network Enhanced Logic) services developed originally for GSM in IMS.

4 ARCHITECTURE

This section presents the proposed hybrid architecture for the content delivery service within IMS. The content is distributed using the BitTorrent protocol, which has proven to be robust and very scalable. The files shared within the swarms are encrypted, which means that they can be shared among the peers without almost any risk of unauthorized use. The DRM solution chosen for the architecture is OpenIPMP from Mutable, which is an open source DRM solution that includes a DRM server. Figure 3 depicts the proposed architecture for the content delivery service. The big arrow between the content access plane and the content distribution plane represents the interaction between both planes. Entities belonging to the content distribution plane (IMS/P2P clients) do interact with components from the content access plane (e.g. Trackers or DRM Application Servers). The IMS signaling plane provides the typical features for user and service management, which are user authentication, charging, routing sip messages and storing the user profile. The content distribution plane is composed by peers forming torrent swarms. They communicate with each other using the bitTorrent peer-wire protocol. Downloading files is performed within this plane without using any central media server. Finally, the content access plane provides the necessary elements

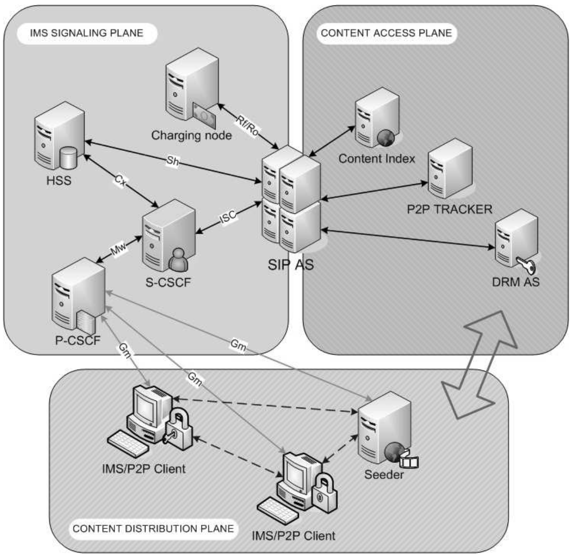

Figure 3: Proposed architecture.

to get access to the content, which are the content index for finding the content, the tracker for finding the peers to download the content from and the DRM AS to obtain the licenses for the content.

4.1 Components

The **IMS Core functions** (HSS and the CSCFs) do not need to provide any extra functionality to make the service work properly. However, the HSS should register a PSI (Public Service Identity), which identifies globally the service, for the SIP AS and update the Initial Filter Criteria for each user's profile if that user is subscribed to the content delivery service. This allows the S-CSCF to route the service requests to the corresponding AS by applying the initial filter criteria, obtained from the HSS.

The **SIP AS** is the central element of the architecture. It coordinates the process of acquiring content and of publishing some new content as well. It receives SIP requests forwarded from the S-CSCF. These requests include information about the digital asset that is going to be acquired or published. The SIP AS confirms the identity of the sender before it performs any action by checking the `P-asserted-identity` header and informs the corresponding charging node. If the transaction was successful the SIP AS is ready to go on processing the request. In the case of acquiring some content, the SIP AS will retrieve a reference pointing to the torrent file for that content. The request includes a Digital Object Identifier (DOI) that is used by the AS to query a database and obtain the torrent file reference. The SIP AS will send back a response including this reference so that the client may start downloading the file. When a commercial transaction has been successful, the SIP AS is also responsible for authorizing rights over the content. Using web services the SIP AS connects to the DRM AS and authorizes certain rights (e.g. being able to play the content for 24 hours) over the content for a certain user. In the case of publishing new content, the SIP AS stores all necessary information obtained from the publisher in the corresponding content access nodes. The SIP AS chooses a tracker to track the file, chooses a location for the torrent file and updates the content index with information about the new released content identified by its DOI.

The SIP AS has been implemented in Java as a SIP Servlet (Servlet, 2005).

The **content index** function represents a web server that stores information about each digital asset. Its web pages contain sip URIs that trigger IMS clients to send requests to the SIP AS. These URIs contain all the necessary information to perform a purchase. The content index entries are updated by the SIP AS when new content is released.

The **tracker** offers peer discovery for each peer connected to it. Apart from peer discovery, the tracker collects statistics about the state of the swarm that are available for the SIP AS (e.g. the AS could use the statistics to assign a tracker for some new content that is less loaded). Some BitTorrent clients provide trackerless alternatives for the peer lookup service such as DHTs (Distributed Hash Tables), which would be considered, but having a tracker present allows more control over the swarm and makes collectingstatistics easier too.

The **DRM AS** chosen for this architecture is the OpenIPMP DRM server from Mutable. This open source DRM server allows content registration assigning a DOI for each new digital content item and stores as well the content key used to encrypt each digital asset. As explained before, the SIP AS is capable of authorizing a user to enjoy some content by using the DRM server web services. In addition, when the user tries to reproduce an encrypted file using the DRM-enabled player for the first time, the player connects to the DRM server for license acquisition. Again this transaction is performed using the DRM server web services. If reproduction rights were previously authorized, the player receives the license and stores it in a secure keystore for further use.

IMS/P2P Client: The client side of the architecture is composed of an IMS client, a BitTorrent client, the DRM-enabled encoder and player which interact among each other. The IMS client component manages the communication with the IMS (signalling).

The BitTorrent client manages the up- and download of encrypted content files in pieces. The DRM encoder and player communicates with the DRM AS to obtain a license and en- and decrypts content, depending on its role as content provider or consumer.
Seeder: The seeder refers to a peer that has a complete copy of the content file and offers it for upload. The number and performance of seeders available within a swarm is fundamental. When a new content item is being published, the content owner will act as initial seeder. The content owner's client will start uploading the file using BitTorrent to the connecting peers.

5 USE CASES

In this section, two commercial use cases are discussed, which show the architectures features for content consumption and customer sided content provisioning. Signalling follows the normal IMS Call Setup procedure (3GPP, 2006b).

5.1 Pay per Download

The first use case is the well known scenario, where a user wants to enjoy some content and he is charged for it. He discovers the content (e.g. a movie) over some external mechanism (e.g. a web site). Also the SIP event notification framework could let the consumer subscribe to content lists (e.g. a TV series, or genre lists, etc.). The content is identified to him by a SIP URI. His client initiates a SIP `MESSAGE` transaction which reaches the SIP AS, related to the content, after passing the assigned S-CSCF, which has checked the users profile for this content. The SIP AS informs the DRM AS to provide a license for the requesting user. The user's client will then download the torrent file from the content index, whose URL it received in the response from the `MESSAGE` request to the SIP AS, and query the tracker for the IP addresses of the peers in the swarm. From now on, download of the encrypted content can start. The prepared license is then obtained from the DRM AS and can be used to decrypt the content according to the terms od the shipped license. Figure 4 illustraets the message flow for this scenario.

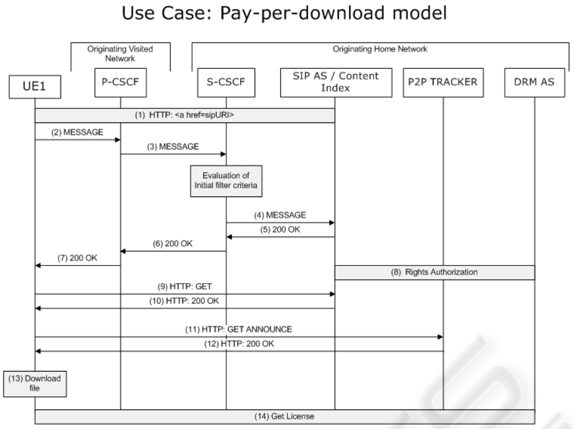

Figure 4: Signaling: Pay per Download.

5.2 Content Injection

Users can also be put in the role of content providers. Therefor a method exists which allows users to share their content and appear as professional content providers. For this the content file is first encrypted and a DOI is generated, which gets registered along with the encryption key to the DRM AS. The encrypted content is stored only at the side of the user, and is not uploaded to any IMS component. The hash values for the torrent file are generated and a `PUBLISH` transaction to the SIP AS is initiated in order to upload the torrent file. The SIP AS prepares a tracker with the torrent file. After the transation is complete, the client, joins the swarm and can start seeding, as soon as downloaders appear in it.

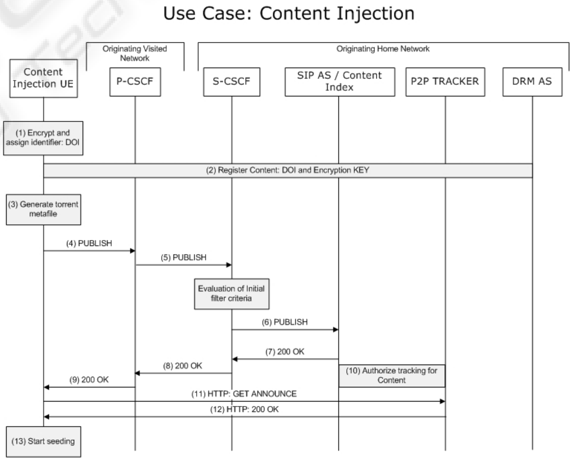
Use Case: Content Injection
Originating Visited Network
Originating Home Network
Content Injection UE
P-CSCF
S-CSCF
SIP AS / Content Index
P2P TRACKER
DRM AS
(1) Encrypt and assign identifier: DOI
(2) Register Content: DOI and Encryption KEY
(3) Generate torrent metafile
(4) PUBLISH
(5) PUBLISH
Evaluation of Initial filter criteria
(6) PUBLISH
(7) 200 OK
(8) 200 OK
(9) 200 OK
(10) Authorize tracking for Content
(11) HTTP: GET ANNOUNCE
(12) HTTP: 200 OK
(13) Start seeding

Figure 5: Signaling: Content Injection.

6 SUMMARY

In this paper, an architecture has been presented, which integrates well known service control technologies (IMS) with an accepted Peer-to-peer content distribution method, which is secured by DRM licensing.

The architecture uses the well known IMS, BitTorrent and OpenIPMP technologies to achieve a secure way to utilize the power of P2P content distribution. IMS controls access, charging and security for content as it controls the DRM Application server as well as the trackers for the torrents. Content can simply be removed from the network by disabling the related tracker.
Two typical use cases have been shown, depicting the various ways that this approach can satisfy the needs of content creators, providers, customers and service providers. People can act as end user content creator, provider or customer. The service providers can serve as mediator between the interests of content creators and consumers by taking care of the value chain. The term of ”service” can be interpreted not only for consuming content, but also for the possibility to provide content. Therefore, content creators and consumers are both service consumers.

7 FUTURE WORK

This paper presented a secure method to distribute static content from content providers to authorized users over a peer-to-peer network. The proposed architecture could be applied to home servers, i.e. service nodes in users homes, extending the existing termination nodes (DSL/Cable modems/routers, etc). In such a scenario, the P2P technology would be hidden from the user and integrated in the service provider’s network. Content may be provided by customers, which act as seeders for their files. In order to unburden a users client from uploading, it is considerable to provide (e.g. as a premium service) additional seeder machines, which are run by its service provider. Such a user could upload its content to the service providers infrastructure, where it is then distributed in the same manner, but without involving the content providers client anymore. Initial upload can be traditional FTP or also BitTorrent with only the users client as seeder and some service provider machines as only peers in the swarm. This could be achieved by muting the tracker, which will be enabled, as soon as the client has finished the upload.

An obvious problem is the ability of customers to circumvent the DRM system by re-publishing downloaded content. Publishing requests should only be accepted from trusted accounts or supporting technologies should be used to filter content that infringes copyright such as watermarking or digital fingerprints.

Ongoing work will focus on the evaluation and testing of the proposed architecture for security and scalability. The outcome of this is expected to reveal the usability for large scale scenarios with a large variety of different content and a varying number of customers. Also an estimation about the number and expected average load of the initial media servers is a desired outcome of this. Evaluation will take place in the FOKUS - OpenIMS Playground (Knuettel et al., 2005) running the open source IMS core (Vingarzan et al., 2006)

Another topic is to extend the architecture to streaming media, like video, for digital TV broadcasting, using a Peer-to-peer network as stream multiplicator for end users. For streaming applications, Quality of Service (QoS) (3GPP, 2005) is going to be much more important than for static content. This introduces new challenges to Peer-to-peer technologies, as they do not include any QoS model yet. Streaming media also introduces new trick functions (e.g. fast-forward, timeshift, etc.) which will have impact on the media distribution.

REFERENCES

3GPP (2005). Quality of service (qos) concept and architecture. TS 23.107.

3GPP (2006a). The 3rd generation partnership project (3gpp). http://www.3gpp.org/.

3GPP (2006b). Ip multimedia (im) session handling; im call model. TS.23.218, Stage 2; v7.3.1.

3GPP (2006c). Ip multimedia subsystem (ims). TS 23.228.

Cohen, B. (2003). Incentives build robustness in bittorrent.

Daemen, J. and Rijmen, V. (1999). Aes proposal: Rijndael. http://www.nist.gov/.

DigiCont (2006). Secure digital container ag. http://www.digicont.ch/c_index.html.

DRM-Watch (2006). Analysis of digital rights management technology. http://www.drmwatch.com/.

ISMA (2006). Internet streaming media alliance home page. http://www.isma.tv/.

KaZaA (2006). Home page. http://www.kazaa.com/.

Knuettel, K., Witaszek, D., and Magedanz, T. (2005). The ims playground @fokus - an open testbed for generation network multimedia services.

Microsoft (2006). Windows media drm. http://www.microsoft.com/windows/windowsmedia/forpros/drm/default.mspx.

MPEG (2006). Moving picture experts group (mpeg). http://www.chiariglione.org/mpeg/.

Object-Lab (2006). Open ipmp. http://objectlab.com/clients/openipmp/index.htm.

ODRL (2006). The open digital rights language initiative. http://odrl.net.

OMA (2006). The open mobile alliance. http://www.openmobilealliance.org/.

Pfeiffer, T., Savage, P., Brazil, J., and Downes, B. (2006). Vidshare: A management platform for peer-to-peer multimedia asset distribution across heterogeneous access networks with intellectual property management.

Pouwelse, J., Garbacki, P., Epema, D., and Sips, H. (2005). he bittorrent p2p file sharing system: Measurements and analysis.

Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and Schooler, E. (2002). Sip: Session initiation protocol. RFC 3261.

Rosenblatt, B. (2003). Integrating drm with peer-to-peer networks. enabling the future of online content business models. http://www.giantstepsmts.com.

Safenet (2006). Recommendations for drm usage. white paper, http://mktg.safenet-inc.com/mk/get/DRM_Usage_WP.

Schneier, B. (1994). Description of a new variable-length key, 64-bit block cipher (blowfish).

Servlet (2005). Sip servlet 1.0. http://www.jcp.org/en/jsr/detail?id=116.

SIPshare (2006). Sipshare: Sip beyond voice and video. http://www.research.earthlink.net/p2p.

Vingarzan, D., Weik, P., and Magedanz, T. (2006). Introducing the fokus open source ims core system for multi access multimedia service environments.