ISCTE IUL
REPOSITÓRIO
INSTITUTO UNIVERSITÁRIO DE LISBOA

Repositório ISCTE-IUL

Deposited in *Repositório ISCTE-IUL*:
2022-05-23

Deposited version:
Accepted Version

Peer-review status of attached file:
Peer-reviewed

Citation for published item:
Sadourny, Y., Conan, V., Serrão, C. & Fonseca, P. (2005). WCAM: secured video surveillance with digital rights management. In Said A., Apostolopoulos J.G. (Ed.), Proceedings of SPIE - The International Society for Optical Engineering. (pp. 27-38). San José: SPIE - The International Society for Optical Engineering.

Further information on publisher's website:
10.1117/12.586963

Publisher's copyright statement:
This is the peer reviewed version of the following article: Sadourny, Y., Conan, V., Serrão, C. & Fonseca, P. (2005). WCAM: secured video surveillance with digital rights management. In Said A., Apostolopoulos J.G. (Ed.), Proceedings of SPIE - The International Society for Optical Engineering. (pp. 27-38). San José: SPIE - The International Society for Optical Engineering., which has been published in final form at https://dx.doi.org/10.1117/12.586963. This article may be used for non-commercial purposes in accordance with the Publisher's Terms and Conditions for self-archiving.

Use policy

Creative Commons CC BY 4.0
The full-text may be used and/or reproduced, and given to third parties in any format or medium, without prior permission or charge, for personal research or study, educational, or not-for-profit purposes provided that:

- a full bibliographic reference is made to the original source
- a link is made to the metadata record in the Repository
- the full-text is not changed in any way

The full-text must not be sold in any format or medium without the formal permission of the copyright holders.

Serviços de Informação e Documentação, Instituto Universitário de Lisboa (ISCTE-IUL)
Av. das Forças Armadas, Edifício II, 1649-026 Lisboa Portugal
Phone: +(351) 217 903 024 | e-mail: administrador.repositorio@iscte-iul.pt
https://repositorio.iscte-iul.pt

WCAM: secured video surveillance with Digital Rights Management

Yulen Sadourny[a], Vania Conan[a], Carlos Serrão[b], Pedro Fonseca[b]
[a]Thales Communications, 160 Bd de Valmy, Colombes
[b]Adetti/ISCTE, Ed. ISCTE, Av. Das Forças Armadas, 1600-082 Lisboa, Portugal

ABSTRACT

The WCAM project aims to provide an integrated system for secure delivery of video surveillance data over a wireless network, while remaining scalable and robust to transmission errors. To achieve these goals, the content is encoded in Motion-JPEG2000 and streamed with a specific RTP protocol encapsulation to prevent the loss of packets containing the most essential data. Protection of the video data is performed at content level using the standardized JPSEC syntax, along with flexible encryption of quality layers or resolution levels. This selective encryption respects the JPEG2000 structure of the stream, not only ensuring end-to-end ciphered delivery, but also enabling dynamic content adaptation within the wireless network (quality of service, adaptation to the user's terminal). A DRM (Digital Rights Management) solution, called OpenSDRM is added to manage all authenticated peers on the WLAN (from end-users to cameras), as well as to manage the rights to access and display conditionally the video data. This whole integrated architecture addresses several security problems such as data encryption, integrity, access control and rights management. Using several protection layers, the level of confidentiality can depend both on content characteristics and user rights, thus also addressing the critical issue of privacy.

Keywords: Video surveillance, JPEG 2000, JPSEC, scalable encryption, digital rights management, content adaptation, privacy

1. INTRODUCTION

On today's World, it is interesting to address all types of content in digital format. This digital world is well suited for current open distribution networks, however it raises interesting challenges that need to be addressed and solved. Particularly, considering the specific WCAM use-case – video-surveillance, there is a number of issues which may affect privacy, integrity and access control. In order to address such requirements, specific technological solutions such as data encryption will be implemented by WCAM. On the other hand, managing the access rights of end-users (or other elements) to the video-surveillance content is an aspect that will be dealt by Digital Rights Management. DRM will enable WCAM to control the access and usage of the video streams that are distributed over the air.

In section 2 of this paper, we present the context of our work, introducing Motion JPEG 2000 streaming and specific issues concerning video surveillance and security. We present the JPSEC framework for securing JPEG 2000 content in section 3, and the OpenSDRM framework used to control users' rights in section 4. Section 5 explains how JPSEC and OpenSDRM work together to achieve a flexible content protection answering the video surveillance requirements in terms of confidentiality and privacy. Finally, section 6 shows the interest of the combined use of selective encryption and RTP streaming, which enables the secure transcoding of Motion JPEG 2000 videos streams.

2. WCAM CONTEXT DESCRIPTION

The objective of the WCAM project is to study, develop and validate a wireless, seamless and secured end-to-end networked audio-visual system. This project started in January 2004 and exploits on the technology convergence between video surveillance and multimedia content distribution over the Internet. WCAM considers the aspects of real-time implementation, security of the delivery and scalability. The video content is encoded in emerging content formats:

Motion JPEG 2000 and MPEG-4 AVC/H.264, and transmitted through Wireless LAN to different types of decoding platforms like PDA's and Set Top Boxes. The content is streamed on the network using the RTP protocol over UDP/IP.

2.1. Streaming Motion JPEG 2000 content

In this paper, we will focus on the Motion JPEG 2000 content case and develop the security features enabled by this content format. Motion JPEG 2000 is a basically a sequence of still JPEG 2000 images: there is no temporal compression involved like in MPEG video formats.

2.1.1. JPEG 2000 standard

JPEG 2000 is the most recent international standard developed by the Joint Photographic Expert Group, JPEG [1] [2]. It defines an image compression system that allows great flexibility not only for the compression of images but also for accessing data in the codestream. A key feature of JPEG 2000 is the flexible bit stream representation of the images that allows to access different representations of images using its scalability features (resolution, quality, position and image component).

A JPEG 2000 compressed image uses markers and marker segments to delimit and signal the compressed information, organized in headers (Main and Tile Parts) and packets. This modular organization allows a flexible bit stream organization for progressive data representation: for instance quality progressive and resolution progressive data progression. A JPEG 2000 codestream always starts with the Main Header followed by one or several Tile Part Headers, each of them followed by compressed data packets, and ends by an End of Codestream marker (EOC). Therefore, JPEG 2000 allows the scalable decoding of compressed images at a desired bit-rate, or for a given image resolution, image region, color component. Consequently, it also allows scalable protection, as we show further below.

2.1.2. Motion JPEG 2000 over RTP

RTP (Real-time transport protocol) provides end-to-end network transport functions for applications transmitting real-time data, such as video data, over multicast or unicast network services. The data transport is augmented by a control protocol (RTCP) to allow monitoring of the data delivery in a manner scalable to large multicast networks, and to provide minimal control and identification functionality [4].

However, the basic RTP does not guarantee quality-of-service for real-time services. On the Internet, a few percents packet loss rate is common and it can be even worse on wireless networks. That is why an efficient packetization of the JPEG 2000 video streams into RTP packets is required to minimize decoding problems due to missing code-blocks. Additionally, if the main header is lost during the transmission, the image cannot be decoded: an error correction mechanism is needed to avoid such a loss.

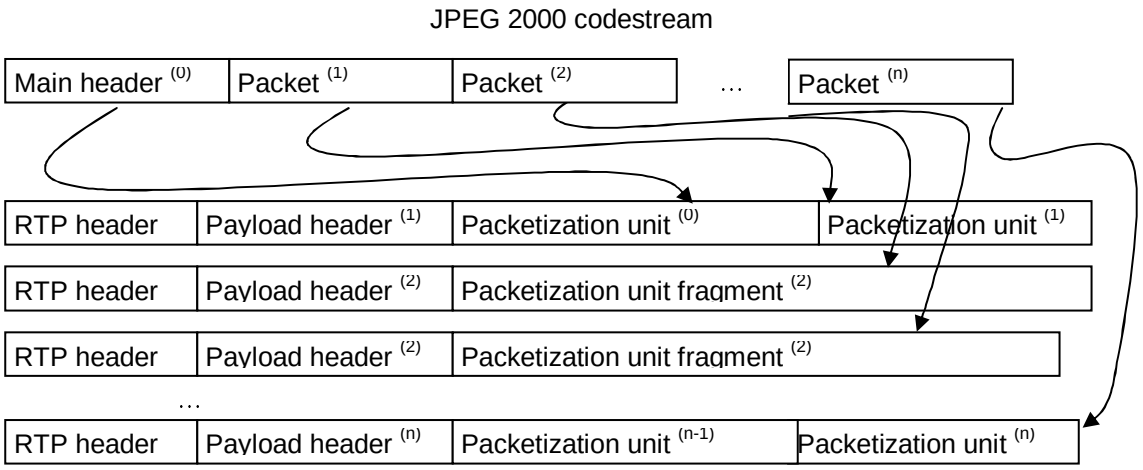

Figure 1 – JPEG 2000 RTP Packetization

To address these issues, a specific payload format for RTP packets is currently under standardization by the IETF for Motion JPEG 2000 streaming [5]. In this draft, the codestreams must be packetized in packetization units: a packetization unit being defined as a JPEG 2000 main header, a tile-part header, or a packet as defined in the standard. First, the server divides the JPEG 2000 codestream into packetization units by parsing the codestream or by getting information from the encoder, and packs the packetization units into RTP packets. The sender puts an arbitrary number of packetization units into an RTP packet, and preserves the codestream order. If a packetization unit with headers is larger than the MTU size, it can be fragmented (Figure 1).

Among the fields introduced in the IETF draft [5], three are especially useful:

- The main header identification field is used for JPEG 2000 main header recovery. The same value is used as long as the coding parameters described in the main header remain unchanged. It is increased by 1 every time a new main header is transmitted.
- The priority field indicates the importance of the JPEG 2000 packet included in the payload. Typically, a higher priority is set in the packets containing JPEG 2000 packets containing the lower sub-bands. This is especially useful for scalable bit streams, to transcode the JPEG 2000 content and to ensure quality of service. We also use this field to transcode protected codestreams, as explained below in section 6.1.
- The fragment offset field is used to rebuild the stream at the receiver's side.

2.2. Specific video surveillance security requirements

Video surveillance introduces many security constraints. Some basic requirements are essential: confidentiality of the content and perfect integrity of the stored/streamed video data. The content may be used as a proof in court, so tampering is strictly forbidden and has to be detected. The DRM system has to provide means to prove that integrity.
The content has to be streamed in a secure way to prevent man in the middle attacks:

- The use of strong cryptographic functions to protect the stream is necessary.
- The content must be encrypted from end to end, i.e. from the camera to the storage device or the end-user player. This protection should be independent from the transmission, which means that it should be applied at content level rather than at network level.
- The content may contain sensitive information, therefore only authenticated and authorized clients can access the sensitive parts.

The content has to be protected in a scalable way: users with different rights can access content with different confidentiality levels. In other words, the content protection scheme should support multiple layers of protection, corresponding to multiple access levels in the DRM system.

Concerning privacy aspects, the use of video surveillance may require the anonymity of the monitored "actors". It should be possible to encrypt some parts of the images, such as faces, on another level of protection than the rest of the content. Authorized users could view a clear image except for these specific parts.

3. OVERVIEW OF JPSEC

JPSEC is part of an on-going effort of the JPEG standardization group to provide ways to develop interoperable applications dealing with secure JPEG-2000 images.
JPSEC specifies the following two normative components:

- a normative codestream syntax containing information for interpreting secure image data
- a normative process for registering JPSEC tools at a central registration authority

The codestream syntax is presented in more detail in section 3.1. The role of the registration authority is mainly to provide unique identification to proprietary JPSEC tools. It also allows JPSEC tool providers to offer more information on their tools allowing implementers and end users to get a more direct access to their technologies.
JPSEC also provides additional non normative examples and information aiming to help developers of secure JPEG 2000 image applications. It gives examples of JPSEC tools in typical use cases and some guidelines on how to implement the security services and to provide the corresponding metadata.

3.1. JPSEC codestream syntax

From an implementation point of view, the critical step is to generate a JPSEC codestream that is compliant with the JPSEC specifications: this means applying some security tools to the actual JPEG 2000 image content, for example applying some signature scheme or encryption to the image data, and adding to it a specific JPSEC header containing the JPSEC related meta information. This information is used on the processing side (e.g. by the client viewer in a client/server application) to learn about the type of security functions that have been applied to the image, and to know which tools to apply to process it securely.

More precisely the main components that are signaled in the JPSEC header are the following:

- A few global parameters valid for the entire codestream (number of tools used, backward compliance information, …)
- The zone of influence that the protection is applied to.
- The list of tool specifications that have to be applied in sequence in order to process the image in a secure way

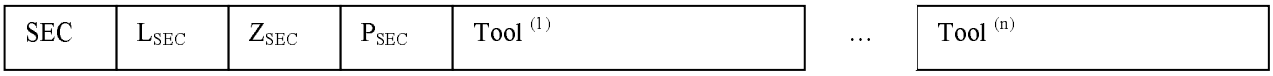

Figure 2 – Basic example structure for JPSEC header: SEC is the reserved value for signaling the start of the header, L_{SEC} is the total length of the header, Z_{SEC} is the zone of influence, P_{SEC} is the list of global parameters; then follows the list of tool specifications.

The zone of influence defines the coverage area of the JPSEC tool. This area can be defined using several domains and semantics:

- in the image related domain, based on the structure defined by JPEG-2000 (e.g. resolutions, quality layers, image area).
- In the non-image related domain (e.g. bitstream segments, packet indices).

There are two categories of tools: proprietary tools, for which the critical information is their identification numbers, followed with proprietary lists of parameters. The registration authority keeps track of all the referenced JPSEC tools ensuring unique identification. The second category corresponds to a finite set of tools whose description is standardized by the JPSEC specifications. They cover decryption (block ciphers, stream ciphers, asymmetric ciphers), authentication (hash-based MAC, cipher-based MAC, digital signature), and integrity.

3.2. JPSEC status

JPSEC is still work in progress; at the time of publication of this paper, the JPSEC working group has produced the Final Committee Draft of its specifications [3]. Following the ISO process, this is now going to ballot and feedback from the National Bodies will be examined by the group at the next JPEG meeting next spring. The International Standard is planned for mid 2006.

In the context of this paper, we wish to illustrate several features that are supported by the JPSEC specifications:

- how to provide selective encryption of JPEG 2000 images (see section 5);
- how to process continuous streams *à la* motion JPEG (see section 6);
- how to integrate the image related security (provided by JPSEC) into a comprehensive DRM framework.

4. DRM FRAMEWORK

DRM is generating much interest, and in the forthcoming time we will assist to many changes on what concerns this hot issue. Companies, such as Microsoft, Apple or Real, private organization consortiums, such as DMP or the Coral Consortium Group, or ISO initiatives, such as MPEG are actively working on DRM-related technologies. Especially, MPEG is considering a technology called IPMP – Intellectual Propriety Management and Protection, which allows to go a step further in terms of interoperability at all levels (MPEG-2/4 IPMPX and MPEG-21 IPMP) 12] [15].

The following part of this paper presents a DRM solution that is based on some of the concepts being developed by ISO MPEG groups, particularly on what concerns the IPMP approach.

4.1. OpenSDRM Presentation

The OpenSDRM rights management platform is composed of a set of distributed components that exchange standardized messages over open networks (such as the Internet) [13] [14] [16]. The OpenSDRM conceptual architecture (presented in Figure 3) defines a scenario capable of handling a multiplicity of different business models for content distribution.

This conceptual architecture comprises three different types of components: the user (not necessarily the end-users) roles; a set of external entities to the DRM process itself; and the internal DRM entities which provide the DRM functionality.

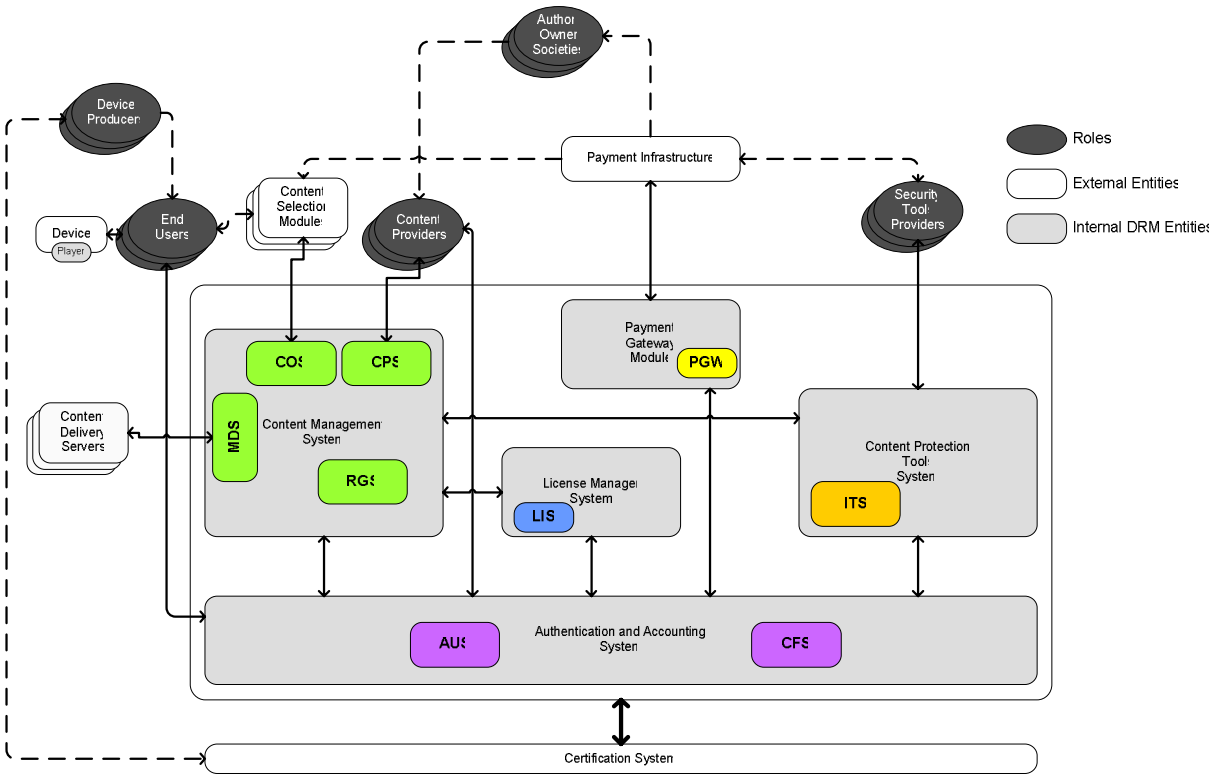

Figure 3 – OpenSDRM conceptual architecture

4.2. OpenSDRM adapted to WCAM context

In this section we introduce the components of the OpenSDRM platform that will be implemented in WCAM. For the WCAM scenarios every one of these components will be in use, not withstanding the fact that implementations of every context dependent components (like the CPS, for instance) will be substantially different from one another.

4.2.1. CPS – Content Preparation Server

This component is responsible for the content preparation. By preparation we mean all the required steps to take any raw content and to produce an output in a specified format that is consistent and coherent with the requirements of the platform being served by OpenSDRM (H264 and Motion JPEG-2000, in WCAM case). An example of a CPS content production operation would be receiving a raw video format from a given video capture device, adding metadata and protecting it, thus preparing the content to be injected in OpenSDRM. Again, the CPS is intrinsically very dependent on the underlying platform, hence the development status. Its development will congregate all the necessary steps to

enforce the content preparation, and as such it is mandatory to have an active participation of every party that has a specific role (as small as it may be) in the content preparation phase.
The CPS has to be developed in conformity with the type of content to be used (using the appropriate encoding tools) and also using specific protection tools developed to be used with that specific type of content.

This component plays an important role on the registration of the content and associated metadata, as well as the active protection of the same content. The following picture demonstrates how the CPS interacts with other components of the system.

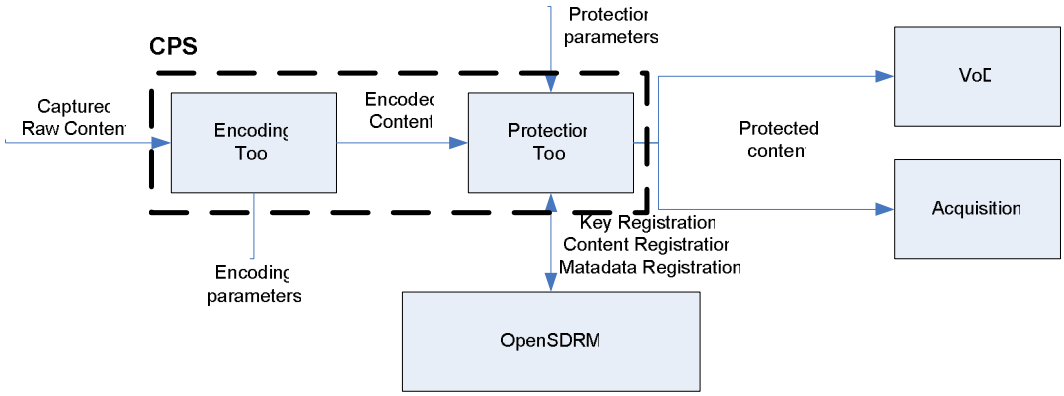

Figure 4 – CPS detailed structure

4.2.2. COS – Commerce Server

This component is the server component responsible for introducing the final user to the content available and registered on the DRM platform. It is the means of navigating, consulting relevant metadata, and in general accessing this available content. Again, and as is the case of the CPS, this component is also very dependent on the nature of the requirements of the underlying platform served by OpenSDRM. Due to this, the range of possible implementations for the COS varies greatly, ranging the web browser application offering generic metadata consultation and content access to a full-fledged application offering content price information and negotiation rules, usage conditions and license composition for accessing the protected content.

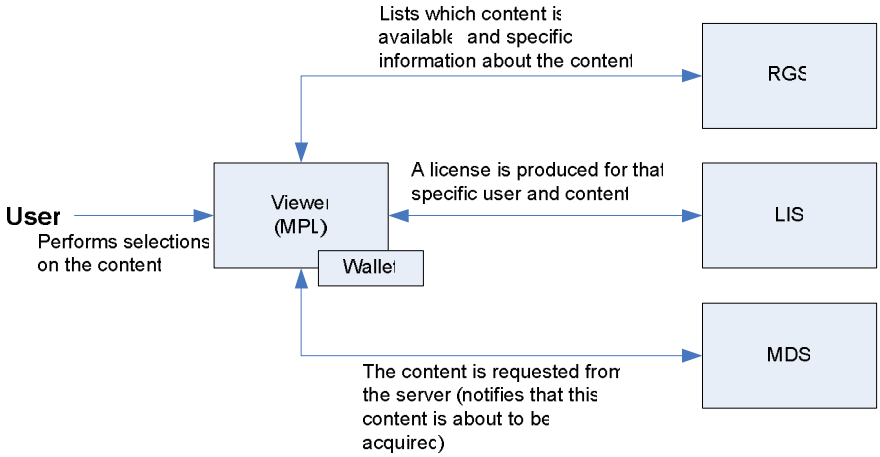

Figure 5 – The player interaction while requesting content

The COS is an entity that is not supplied by the OpenSDRM platform itself, but it is an entity that must exist to allow the user to select and establish content conditions.

4.2.3. MDS – Media Delivery Server

The Media Delivery Server is responsible for keeping track of all registered content on the platform. MDS stores the location (URI) of all registered content. MDS thus remains independent of content and delivery types (JPEG 2000 streams and RTP).

4.2.4. RGS – Registration Server

The sole role of this component is to assign unique identifiers to the content inside the DRM platform, as well as to register and keep metadata information for that specific content. Ultimately it is responsible for keeping a bullet-proof consistency in terms of content identification, assuring no ambiguities in protected content management. If required specific usage scenarios can devise their internal identification and metadata information protocols. OpenSDRM relies on the general assumption that architectures should be as close as possible to standards. Therefore, and by default, regarding the content identification matter, it follows the MPEG-21 directives about Digital Item Identification (DII), by using a reduced version of the MPEG-21 DII Digital Object Identifiers (DOI).

4.2.5. AUS – Authentication Server

The AUS is the OpenSDRM component responsible for authenticating and validating the access of all the other components in the platform, both internal and external. It functions as a SSO (Single Sign-On) point for the whole system, registering and managing components and users in it. This component implements the Application Layer level security using cryptographic XML credentials to authenticate both components and users in order to validate all the transactions exchanged between them (XML Encryption and XML Signatures). The AUS plays a significant and critical role in the OpenSDRM platform. It is completely independent of the specific requirements of any given implementation, and as such its development status is complete.

4.2.6. LIS – License Server

The LIS is the server component responsible for storing and maintaining the rules associating a given content, an end user, and his corresponding access rights. The LIS is also responsible for the establishment of license templates associated with particular types of content or business models. This component will accept connections from authenticated client Media Players for the downloading of licenses, which will be applied to the protected content through an appropriate IPMP tool. The licenses are XML formatted using Open Digital Rights Language (ODRL/OMA profile), and in the future they will migrate to the Rights Expression Language (REL), currently being developed by MPEG-21.

4.2.7. ITS – IPMP Tools Server

The ITS is the server component responsible for registering new IPMP tools on the system and for receiving authenticated client Media Player requests for the downloading of a specific IPMP tool. These IPMP tools available on the system can be used by the final user, on the Media Player, in order to appropriately render a given content; or by the CPS, in order to enable the production of content.

5. CONTENT PROTECTION TECHNIQUES

This section describes methods for encrypting JPEG 2000 images, and consequently Motion JPEG 2000 streams. This scalable encryption approach relies on JPSEC and is integrated in a DRM framework to provide confidentiality as well as privacy. Other examples of content protection using JPSEC can be found in [6] [8] [9] [17].

5.1. Scalable encryption

Scalable encryption of data implies the possibility to select the parts of data that are encrypted. This means that the chunk of data to be encrypted is not seen as a single block, but as a structured piece of data. For example, a video stream can be looked at as a single file, which can be encrypted as a whole. But one can also wish to keep the frame structure of the video, and, at the same time, deliver it in encrypted form. Scalable encryption permits it.

More generally scalable encryption preserves a given degree of content structure; it supports transcoding and other content processing functionality without the need to access the cryptographic key and to perform decryption and re-encryption. For network and processing efficiency reasons, the scalable encryption schemes should aim not to interfere with the coding and decoding processes, have very limited adverse impact on the compression efficiency and no adverse impact on error resilience.

In this paper we apply scalable encryption to JPEG-2000 image streams. In this context, encryption is applied at packet level: each individual packet may be encrypted using one or multiple secret keys, relying on symmetric key block cipher algorithms (in our case AES). The images are protected after they pass by all the normal JPEG-2000 production pipeline, resulting in a JPEG-2000 structured codestream in which some packets are encrypted.

Relevant to this approach is the selective encryption approaches: only the higher energy coefficients in the compressed domain are encrypted. This has been applied originally on I-frames of MPEG streams [10], and then to JPEG, MPEG and MP3 formats. Encryption of the DCT coefficients before Huffman coding may reduce dramatically compression efficiency [11]. In our scheme, encryption is carried out on the coded data, so the only increase in size is due to some security signaling and possible padding (only for some cipher modes – this is avoided if stream ciphers are used).

Our approach to scalable encryption allows a maximum of flexibility in the implementation of scenarios with various levels of security requirements, as detailed in the next section.

5.2. JPEG 2000 content protection integrated with DRM

To protect the video surveillance data, we use the scalable encryption approach described above, along with the OpenSDRM framework.
One of the major enhancements provided by the DRM solutions is the possibility to define under which conditions the content can be used, and also the possibility to enforce such conditions. These conditions may vary according to the type of content, the user, the device or many others. To achieve the construction of such conditions a Rights Expression Language is used, and in the case of WCAM the rights are expressed using ODRL.
The WCAM ODRL licenses allow the specification of the user and content identification, decryption key(s), number of usage and validity period. The player will remove the protections corresponding to the keys provided in the license.

JPEG 2000 video streams can be protected in various ways, thanks to our method. These are the two main approaches for ensuring confidentiality:

- The codestreams are partially encrypted with one key: some parts of the codestream are encrypted while others are not; in this case the same key is used to encrypt the parts of the codestream. In this situation, the user could be allowed to access freely a low resolution version of the image (whose corresponding packets are not encrypted), while the higher image resolutions are encrypted because they have some kind of value. The user is only allowed to access these higher resolutions levels with the appropriate license;
- The codestream is partially encrypted with different keys: some parts of the codestream are protected, and these parts are protected with different keys. This means that different users might have different license levels to access different image parts. This situation allows a more flexible solution, since different business models can be deployed and used in this case.

As part of the proposed security solution, each of the protected codestreams will need some piece of information indicating which protection was applied and how the protection tools can be obtained. This signaling information is necessary for the user in order to be able to properly decrypt or process the JPSEC-protected data.

The figure below (Figure 6) shows a protected JPEG 2000 frame illustrating the second approach. The image contains three decomposition levels. Its two higher levels are encrypted with Key1, hence its blurry appearance. Additionally, the face zone has been encrypted with Key2, for all decomposition levels: that part of the image is completely encrypted and one cannot guess the hidden face.

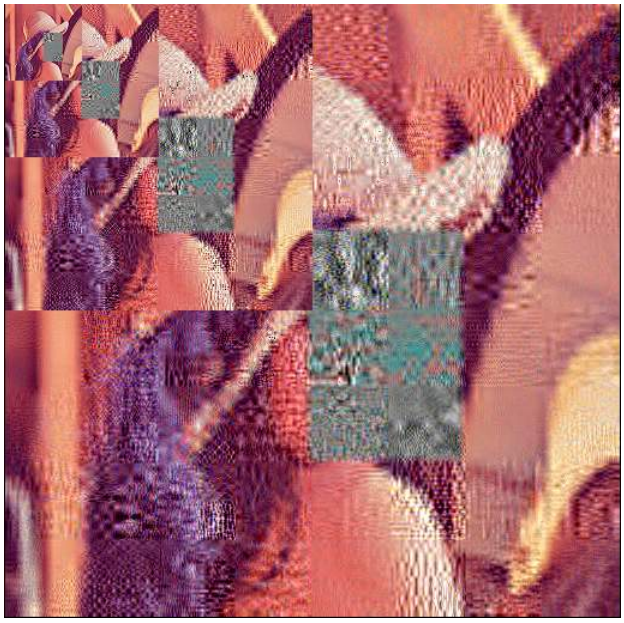

Figure 6a: Encrypted image (res. 1 up to 3)

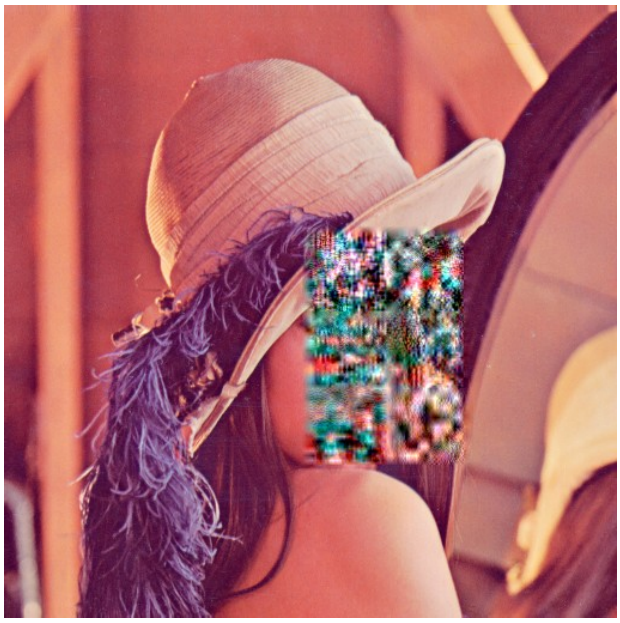

Figure 6b: Partially decrypted image

Figure 6 – Example of scalable encryption

Figure 6.a shows the encrypted image at all four available resolutions. The lowest resolution is in clear, except for the face. However, a user won't be able to access the detailed image without Key1.
Figure 6.b shows the partially decrypted image. In that case, the player has been granted a license with Key1, meaning that the user can access the highly detailed image, with the exception of the face. In WCAM, we use this feature to add protection levels for very sensitive data. For privacy reasons, a face can often be blurred or encrypted in video surveillance applications.

5.3. DRM Messages protection

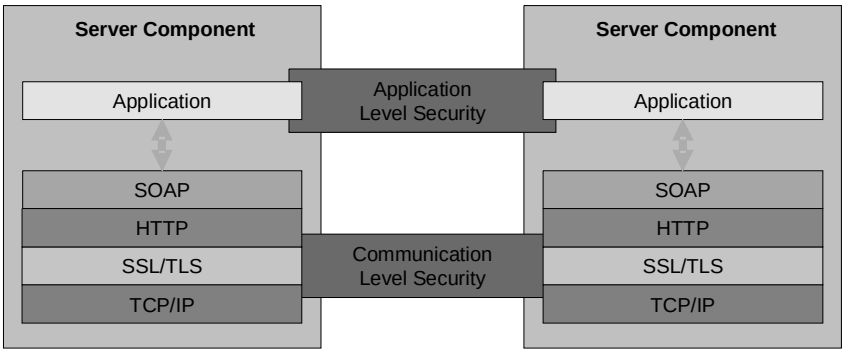

Figure 7 – OpenSDRM protocol stack

Additionally to video content encryption, it is necessary to protect the DRM framework itself, since the decryption keys and licenses are exchanged through that channel. Due to the distributed architecture nature of the OpenSDRM platform, internal components interact with each other through SOAP formatted messages. The SOAP protocol runs over HTTP, which in turn runs over SSL/TLS. The latter is responsible for providing the first front of security for this DRM platform, assuring that every channel used to exchange every message is completely secured. Moreover, an additional

level of security (at the Application Layer level) assures the security and authentication of the SOAP messages themselves (and therefore of the intervening components).

The OpenSDRM architecture relies on the SSL/TLS protocol to ensure secure transport. Each of the servers on which the software components are installed has an X.509 certificate issued by a Certification Authority (CAU). OpenSDRM thus establishes an underlying secure and authenticated transport channel that will allow the messages to flow from component to component securely.

6. SECURE CONTENT ADAPTATION

WCAM addresses video surveillance scenarios where the cameras are networked through wireless links: this poses a number of challenges to traditional security approaches. The first challenge is error resilience and stream ciphers are used to limit their impact. The second challenge is varying bandwidth conditions. The transcoding of still JPEG 2000 images has already been introduced in [7], for instance. In this section, we propose a new approach to dealing with this problem on relying on a combination of on-going standardization work: JPSEC for content security and JPEG 2000 over RTP for video data streaming. By mapping the JPEC 2000 codestream structure, preserved by JPSEC scalable encryption, onto RTP packets, we allow efficient dynamic secure content adaptation of streamed JPEG 2000 videos.

6.1. Transcoding mechanisms

The transcoding of a JPEG 2000 image is nothing more than a truncation of selected data in the codestream. Consequently, a transcoding node in the network needs to understand the structure of the codestream in order to cut the right parts. There are several ways to do it.

The most straightforward way to understand the structure of a codestream is to parse and index it: this is the first step of a decoding process. While this method is quite simple to implement, it has two major drawbacks. First, the transcoding node needs to have access to the whole codestream and cannot begin the adaptation as soon as it receives the first packets. Secondly, this parsing process needs some computational power which may not be available if the network is heavily loaded.

Another transcoding mechanism is described in [8]. It works well in coordination with JPSEC, and does not require heavy calculations. The only problem with this method is that it is not as flexible as the JPEG 2000 codestream structure would allow. Indeed, the JPSEC protection step fixes the transcoding possibilities once and for all, through the definition of "zones of influence" which correspond to the available adaptation granularity.

The mechanism we propose takes advantage of the RTP packetization and of the payload header described above in section 2.1.2. The RTP packets have been created on the server according to packetization units, which match the structure of the codestream. The parsing and indexing process has already been performed: the transcoding application just needs to link each packet to a given resolution level, quality layer and component.
To achieve that, we use the priority field of the RTP payload header: actually the IETF draft does not define any mandatory structure for that field. The syntax used in WCAM to carry that information is the following:

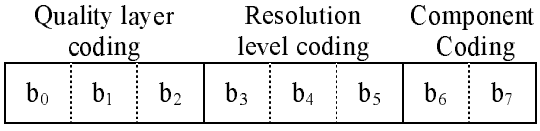

Figure 8 – RTP payload header *Priority* field

This method allows us to transcode a video stream in terms of up to eight quality layers, eight resolution levels and three components (given the fact that the *zero* priority is reserved for JPEG 2000 codestream headers). This is far enough for the majority of Motion JPEG 2000 video streams, especially in the WCAM use case of a wireless network dedicated to video surveillance.

6.2. Performance results

A transcoding proxy was developed for demonstration and benchmarking purposes. The goal of the implementation is to analyze the overall scalability of the transcoding process itself. The proxy was implemented in Java, and although not optimized, it is useful to assess whether a Motion JPEG 2000 video transcoding application is able to process a high bit rate stream in real-time.

Figure 9 shows the delay introduced by the transcoding application for the first three images. The video stream used for this test is a 25 frames per second video, encoded at 5Mbps. As shown in the graphs below, the packets are sent every 40ms. The delay between the first RTP packet of an image and the last one comes from the network bandwidth limitation at around 10Mbps. The second image is sent a few milliseconds late, due to the lack of real-time precision of the operating system, but this is not an issue at such a frame rate.
The results show a 300ms delay introduced by the transcoding application, which is all right for a video surveillance application; however it would be the acceptable limit for a video conferencing application, due to interactivity requirements.

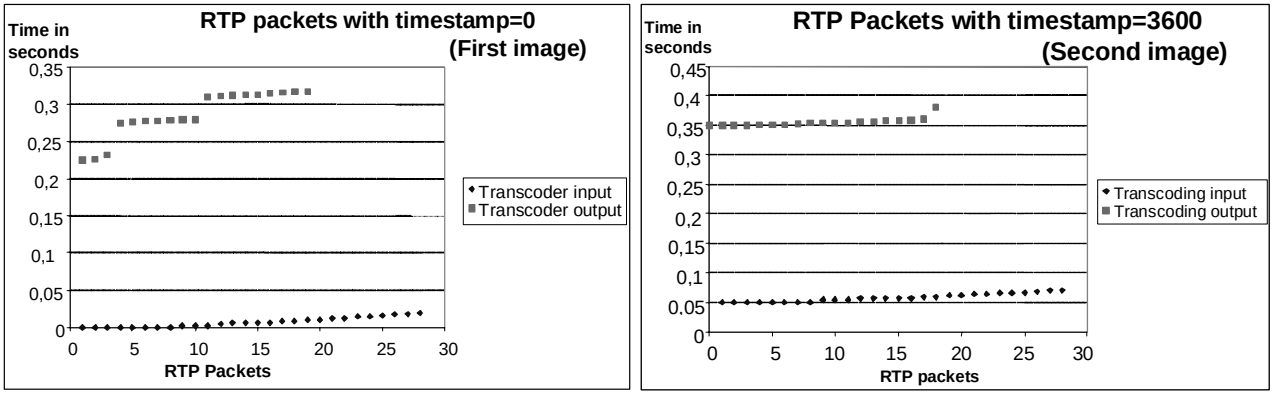

Figure 9 – Delays due to video transcoding

Our results also show that the measured delay is independent of video bit rate or hardware configuration. However, the CPU usage can vary a lot. But any computer should be able to transcode a video, even for high bitrate streams. Typically, a 30Mbps stream processed by a 2GHz processor (or equivalent), takes up to 40% of CPU usage to perform a full transcoding, both in terms of resolution levels and quality layers.

7. CONCLUSION

In this paper we have proposed a secure and scalable framework for the management of video surveillance data over a wireless network. The approach relies as much as possible on standards (Motion-JPEG2000 for encoding, JPSEC for content security, RTP packetization for transport and MPEG for Digital Rights Management). We have investigated how to combine these technologies to address the security requirements of such applications and demonstrated that it is possible to support fine grained content security in an efficient scalable framework.

In our future work we plan to investigate further how to support more interactivity in the framework. This includes updating dynamically security policies and taking more efficiently into account the network and security context to raise alarms.

ACKNOWLEDGEMENTS

This work is partially funded by the European Commission under IST project WCAM.

REFERENCES

1. ISO/IEC 15444-1/ IUT-T T.800, *JPEG2000 Image Coding System - Part 1: Core Coding System*, 2000.
2. D. Taubman and M. Marcellin, *JPEG 2000: Image Compression Fundamentals, Standards and Practice*, Kluwer Academic Publishers, 2002.
3. *JPSEC Final Committee Draft 1.0*, ISO/IEC JTC1/SC29 WG1 N3480, November 2004.
4. H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, *RTP: A Transport Protocol for Real-Time Applications*, IETF RFC3350, July 2003.
5. S. Futemma, E. Itakura, A. Leung, *RTP Payload Format for JPEG 2000 Video Streams*, *Internet Engineering Task Force (IETF)*, draft-ietf-avt-rtp-jpeg2000-06.txt, October 2004.
6. Y. Sadourny and V. Conan, *A proposal for supporting selective encryption in JPSEC*, In IEEE Transactions on Consumer Electronics, vol. 49, no. 4, pp. 846-849, November 2003.
7. S. Wee and J. Apostopoulos, *Secure scalable streaming and secure transcoding with JPEG 2000*, In IEEE Proc. Int. Conf. On Image Processing (ICIP), September 2003.
8. F. Dufaux, S. Wee, J. Apostopoulos and T. Ebrahimi, *JPSEC for Secure Imaging in JPEG 2000*, SPIE Proc. Applications of Digital Image Processing XXXVII, August 2004.
9. Y. Wu, D. Ma, and R. Deng, *Progressive protection of JPEG 2000 codestreams*, In IEEE Proc. Int. Conf. On Image Processing (ICIP), Singapore, October 2004.
10. T. Maples and G. Spanos, *Performance study of a selective encryption scheme for the security of networked, real-time video*, In Proceedings of the 4th International Conference on Computer Communications and Networks, Las Vegas, Nevada, September 1995.
11. L. Tang, *Methods for encrypting and decrypting MPEG video data efficiently*, Proceedings of the 4th ACM International Conference on Multimedia, pp. 219-229, Boston, November 1996.
12. Lacy J., Rump N., Kudumakis P., MPEG-4 Intellectual Property Management & Protection (IPMP) - Overview & Applications Document, ISO/IEC JTC1/SC29/WG11/N2614, 1998
13. Siegert G., Serrão C., An Open-Source Approach to Content Protection and Digital Rights Management in Media Distribution Systems, Proceeding ICT Conference 2003, Copenhagen, 2003
14. Serrão C., Siegert G., Open Secure Infrastructure to control User Access to multimedia content, WOSIS2004, Porto, Portugal, 2004
15. Serrão C., Marques J., Enabling Digital Content Protection on Super-Distribution Models, Virtual Goods 2004, Ilmenau, Germany, 2004
16. Siegert G., Serrão C., An Open-Source Approach to Content Protection and Digital Rights Management in Media Distribution Systems , 8th Anual CTI Conference, Denmark, 2004
17. Serrão C., Serra A., Fonseca P., Dias M., A Method for Protecting and Controlling Access to JPEG2000 Images, SPIE 2003 Anual Meeting, San Diego, California, United States of America, August 2003