ACTA TECHNICA CORVINIENSIS
BULLETIN OF ENGINEERING

[1.] HENRYK KRAWCZYK, [2.] MARCIN BARYLSKI

ARCHITECTURE OF REQUEST/RESPONSE AND PUBLISH/SUBSCRIBE SYSTEM CAPABLE OF PROCESSING MULTIMEDIA STREAMS

Abstract:
On-the-fly analysis of multimedia flows, containing high quality voice and video data, is still a challenge.The paper depicts an architecture of a system capable of processing real-time multimedia streams with distributed compontents working in both Request/Response and Publish/Subscribe manner, taking advantage of Java Multimedia Framework (JMF). The system is secured with IPSec (Internet Protocol Security) and SSL (Secure Socket Layer) protocols that impact the performance but are fully recompensed with increased system overall security. Seven testing measurement points are defined on the system critical path – the results are going to be applied to multidimensional approach to quality analysis of distributed applications working in public-private network infrastructures.

Keywords:
architecture, multimedia streams, processing, request/response, publish/subscribe

INTRODUCTION

On-the-fly analysis of multimedia flows, containing high quality voice and video data, is still a challenge. Firstly, a system designed to such a target requires wide infrastructure of reliable multimedia stream sources, then network (wired or wireless) capable of secure data high-bandwidth transmission to the system headquarters where stream are analyzed, classified, and quick but correct decision process is performed. Secondly, there is a group of interest that would like to receive the results of the multimedia streams processing but with certain level of reliability, if possible without need of human assistance, without false alarms.

On the other hand application of such analysis and classification is strongly demanded by many business needs. It covers observation of manufacturing processes, security monitoring, detection of fire, flood, or any natural disasters, potential crime scene monitoring, traffic measurements, and so on. Such a system is irreplaceable to support mass entertainments, including sport events, able to detect hooligan behavior, glass break, smoke, fire, shout for help in many languages, and more.

This paper will present the architecture of multimedia streams processing system, with components working in Request/Response (R/R) and Publish/Subscribe (P/S) architectures (Krawczyk H., Barylski M. 2009) The middleware layer will be secured with the use of IPSec and

© copyright FACULTY of ENGINEERING – HUNEDOARA, ROMANIA

HTTPS. Furthermore the required implementation steps are discussed.

SYSTEM CONCEPT

From high-level perspective the proposed system is build from 3 main components: Web Cam Client Network (WCCN), Multimedia Flow Processing Engine (MFPE), and End-Client Network (ECN) (Fig. 1).

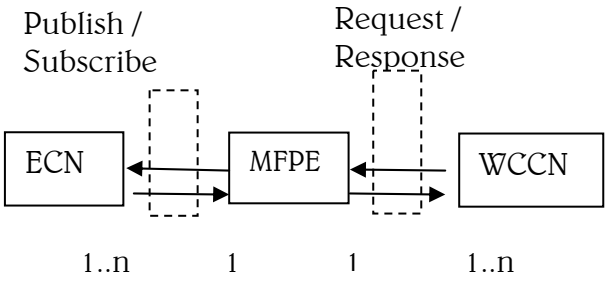

Fig. 1. System overview

WCCN consists of a set of Web Cam Clients (Fig. 3), organized as a group of small Private Networks capable of capturing voice and video data in runtime, reporting the results over Public Network to non-stop listening MFPE services. The Clients cooperate with the MFPE in R/R manner – the multimedia flows are captured by the Web Cam Clients, initially processed (flow standardization to multimedia format recognized by MFPE) and transmitted as a sequence of subsequent requests to MFPE. MPFE acknowledge the successful reception of each flow piece by short response messages. Lack of acknowledgement means the corresponding data must be retransmitted.

ECN is a network of thin end-clients, strongly interested in MFPE processing results, communicated to them over Public Network by Apache Tomcat + JSP. The Clients act in P/S architecture (Krawczyk H., Barylski M. 2009) (Farooq U., Majumdar S., Parsons E. 2007) – they register for certain results of multimedia flows processing (e.g. detection of fire) that are published by MFPE as soon as the multimedia flow classification produces output.

MFPE is a supercomputer capable of fast-enough and accurate classification of multimedia flow received from WCNN, constantly updating the final results matrix available to ECN, stored in data repository (JDBC + MySQL) with Web Services infrastructure. MFPE must be equipped with multimedia processing queue able to store the received data without loss. Unquestionably MFPE needs to be a powerful, multithread and multicore processing unit. MFPE is based on Java Media Framework (JMF) that enables the playback and transmission of Real Transfer Protocol (RTP) streams through the APIs defined in the javax.media.rtp, javax.media.rtp.event, and javax.media.rtp.rtcp packages (Fig. 2).

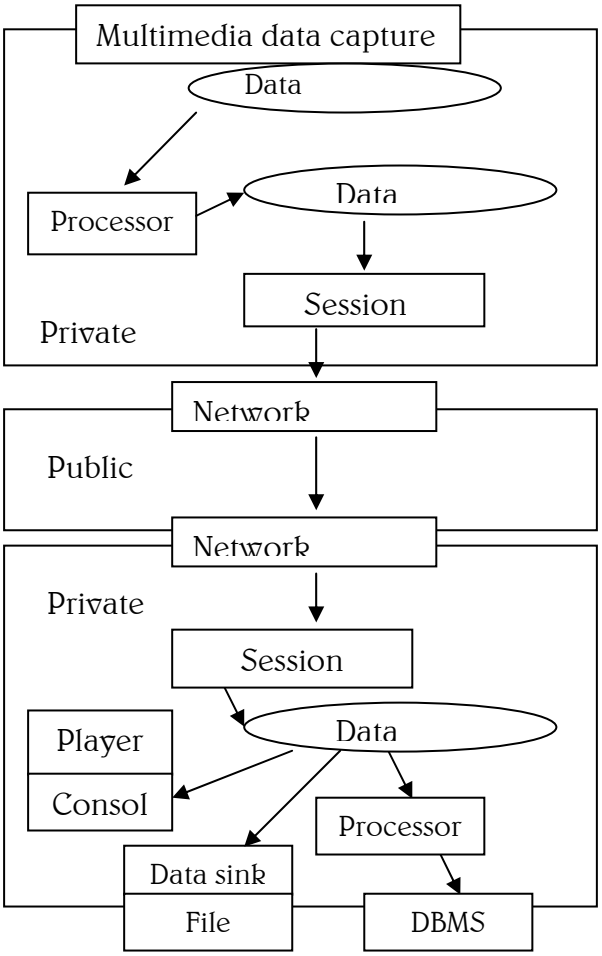

Fig. 2. RTP multimedia streams transmission and reception

The JMF RTP APIs are designed to work seamlessly with the capture, presentation, and processing capabilities of JMF. Players and processors are used to present and manipulate RTP media streams just like any other media content. Media streams that have been captured from a local capture device using a capture DataSource or that have been stored to a file using a DataSink can be transmitted. Similarly, JMF can be extended to support additional RTP formats and payloads through the standard plug-in mechanism. SessionManager is used to coordinate an RTP session. The session manager keeps track of the session participants and the

streams that are being transmitted. It maintains the state of the session as viewed from the local participant. In effect, a session manager is a local representation of a distributed entity, the RTP session. It also handles the RTCP control channel, and supports RTCP for both senders and receivers.
The SessionManager interface defines methods that enable an application to initialize and start participating in a session, remove individual streams created by the application, and close the entire session. Several RTP-specific events are defined in javax.media.rtp.event. These events are used to report on the state of the RTP session and streams. The streams within an RTP session are represented by RTPStream objects. There are two types of RTPStreams: ReceiveStream and SendStream. Each RTP stream has a buffer data source associated with it. For ReceiveStreams, this DataSource is always a PushBufferDataSource. The session manager automatically constructs new receive streams as it detects additional streams arriving from remote participants. New send streams are constructed by calling createSendStream on the session manager. To implement a custom packetizer or depacketizer, JMF Codec interface is implemented.

ANALYSIS OF R/R COMPONENTS

In the R/R architecture one program is asking the other for any new information that has arrived since the last time it asked by sending a request message and expecting a corresponding response message (Krawczyk H., Barylski M. 2009). The main advantage of R/R is simplicity what directly causes that a chance of potential defect is lower. On the other hand it causes high mean communication channel utilization - every message exchange between client and server must be initialized by a request message. It causes that the performance of the R/R model is not optimal from application point of view.
WCCN is a set of Web Cam Clients. Each Web Cam Client is built from the camera HW and firmware, able to continuously capture, pack to appropriate encoding format and transmit the results to its destination end-point (Web Cam Server = MFPE) as soon as possible. A piece of WCCN may consist of one or more Web Cams. The transmission from WCCN to MFPE happens over Public Network (Ethernet + IPv4 + TCP), insecure, available to eavesdroppers, tractable to forgery, data manipulation, open to hackers and DoS attacks. To secure the communication IPSec ESP (Kent S. 2005) (Barylski M. 2007) + IKEv2 mechanism is incorporated.
The edge of WCCN is equipped with IPSec Gateway (GW) (Frankel S., Kent S., Lewkowski R., Orebaugh A. 2005) with appropriate Security Policies Database (SPD), maintaining the Security Associations (SAs) within Security Association Database (SAD) (Kent S., Seo K. 2005), handling SA lifetime, SA expiration events, SA SN (Sequence Number) overflow, SA renegotiation via two-phased IKEv2. On the opposite communication side (MFPE) stands the IPSec GW able to capture and decrypt the received flows from WCCN. SPD and SAD on both end-point must be synchronized by IKEv2. For the security reasons, to protect the data, a pair of HMAC-SHA1 + AES-CBC256 algorithms is used, with setkey infrastructure incorporated.

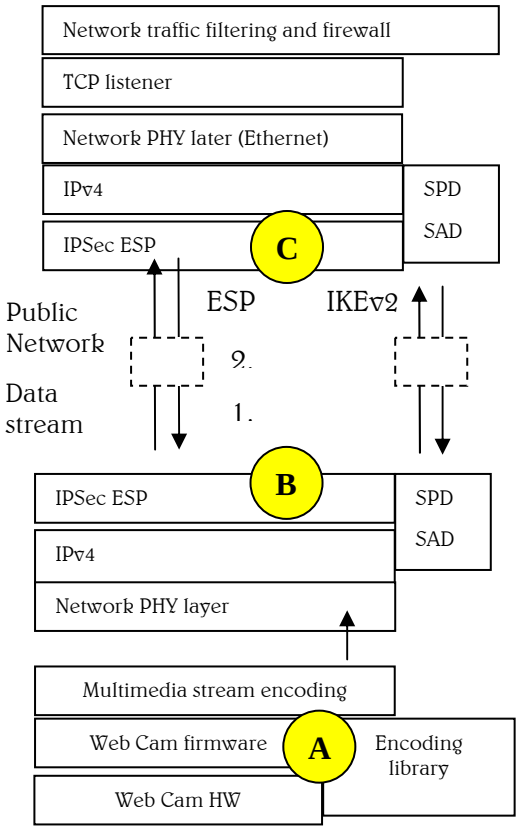

Fig. 3. System R/R components with measurement points

For R/R the overhead of IPSec and IKEv2 is as follows: need of SA renegotiation via two-phased IKEv2 if previous SA has expired (SA lifetime expiry, SA kilobytes expiry or SA sequence number overflow), additional bytes of ESP

header and recommended HMAC-SHA1 authentication at the end of the datagram. In comparison to the IP traffic over Ethernet without IPSec it is obvious that less application data is sent over the wire for the same Ethernet frame length (Kim O., Montgomery D. 2003). However if both client and server incorporate IPSec with AES-CBC-256 encryption and HMAC-SHA1 authentication algorithms with long keys known to these parties only the communication security increases significantly.

ANALYSIS OF P/S COMPONENTS

The P/S approach expands and optimizes the communication channel utilization in comparison to R/R (Krawczyk H., Barylski M. 2009). The Client (Subscriber) registering an interest in certain data with a server program and Server (Publisher), asynchronously sends new information to the subscriber each time it is ready.

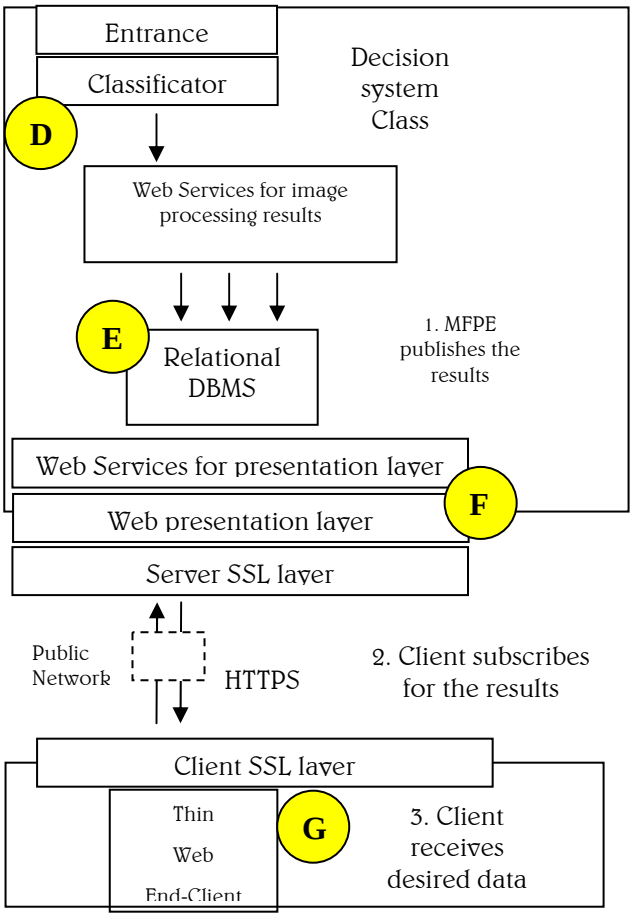

Fig. 4. System P/S components: deployment diagram with 4 measurement points

The main advantage of the P/S cooperation model is significant reduction of bandwidth requirements; Subscriber is no longer constantly asking for new data. Publisher only sends data changes for a specific point to those clients that have registered for exceptions on that point. The data is not delayed by polling cycles. IP multicast can decrease network traffic by sending the data from the publisher to the subscriber with a single message on the wire.

HTTPS used in P/S approach incorporates benefits of SSL (Secure Socket Layer) which allows managing accessibility by certificates and PKI (Public Key Infrastructure). It slows down the browsing of WWW resources - additional datagrams must be sent - but makes it harder to steal HTTP session variables or fool user authorization mechanisms over network.

LOCALIZATION OF MEASUREMENT POINTS

Implementation of the system is essential part of research related to verifying the testing model, based on security and performance testing, for improving quality of distributed applications working in public-private network infrastructures (Barylski M., Krawczyk H. 2009). Model fundamentals are the security and performance metrics gathered at the system critical points.

There are seven critical points (Tab. 1) identified in the discussed system architecture that should monitored, situated on a process flow from the multimedia stream source to system end-client presentation. Point A is the Web Cam Client who is open to intensive HW resources consumption. Then there is IPSec GW of WCCN (Point B), responsible for securing the data, with the IPSec ESP and IKEv2 processing tasks (Barylski M. 2008). On the opposite side of ESP tunnel there is point C: the destination IPSec GW that handles a bunch of IPSec connections – efficient SPD and SAD processing is required there. Point D is the multimedia stream processing layer, the backbone of the system.

The results of analysis are placed in the distributed data repository with appropriate Web Services layer exposing the available methods – point E. Then results are published, being available to the subscribers – point F – the latency from the time of happening to the time of results published. Point G, the last one, is the

thin-end-client, able to assess the system performance from the end-user point of view.

Tab. 1. Testing points with adequate metrics

	Performance metrics	Security metrics
A	▪ Average CPU time usage [%] ▪ Operational memory average usage [MB]	▪ # of system malfunctions ▪ Area covered by the observation [m2]
B	▪ # of frames dropped due to missing SA ▪ Data throughput [Mb/s] ▪ IPSec throughput [Mb/s]	▪ # of SA rekeyed per second [SA/s] Ciphering strength
C	▪ Peak # of open connections ▪ IPSec throughput [Mb/s] ▪ Average CPU time usage [%] ▪ Operational memory average usage [MB]	▪ # of SA rekeyed per second [SA/s] ▪ IPSec IKE latency [ms] ▪ IPSec SA latency [ms] ▪ # of replayed IPSec packets/s [packet/s]
D	▪ Correctness of stream classification [%]	▪ # of system malfunctions
E	▪ Average DB query response time [ms]	▪ # of DB deadlocks per second [deadlock/s]
F	▪ System latency [ms]	▪ # of SQL injections ▪ # of exceptions ▪ time of system availability [%]
G	▪ HTTPS response latency [ms]	▪ # of exceptions

SUMMARY AND FUTURE WORK

The paper presents the architecture of a system capable of processing real-time multimedia streams, with decomposition to R/R and P/S components. Implementation details, based on Java framework, are discussed. Multimedia streaming is supported by JMF, able to transmit RTP traffic through public network. Communication over public network is secured with IPSec ESP + IKEv2 between IPSec Gateways (R/R components), and HTTPS (P/S components). Seven important measurement points are defined on the main application flow path.

The results of experiments will be used against multidimensional approach to quality analysis based on security and performance testing, presented in (Barylski M., Krawczyk H. 2009). Inputs to the testing model are the numerical values of the metrics.

Presented solution is a part of work sponsored by project Mayday Euro 2012: Supercomputer Platform for Context Analysis of Multimedia Data Streams to Identify Specialized Objects or Dangerous Events (POIG.02.03.03-00-008/08).

REFERENCES

[1] KRAWCZYK H., BARYLSKI M.: Performance and Security Testing of Distributed Applications Working in Public/Private IPSec and HTTPS Environments. In Proceedings of the Work In Progress Session of 17th Euromicro Conference on Parallel, Distributed and Network-based Processing, PDP2009, Germany, Weimar, 2009.

[2] BARYLSKI M., KRAWCZYK H.: Multidimensional Approach to Quality Analysis of IPSec and HTTPS Application. The Third IEEE International Conference on Secure Software Integration and Reliability Improvement SSIRI 2009. Shanghai, China, July 8-10, 2009.

[3] BARYLSKI M.: Throughput Analysis as an Effective Method of Discovering Serious Defects of Network Devices. 8th International Scientific Conference Technology Systems Operation'07. Presov, Slovakia, 2007.

[4] BARYLSKI M.: Introduction to implementation and validation of complex information systems with IPSec technology as an example. Scientific Journal of Faculty of Electronics, Telecommunications and Informatics, Gdansk University of Technology, vol.13, Gdansk, May 2007, pp. 283-290.

[5] BARYLSKI M.: On IPSec Performance Testing of IPv4/IPv6 IPSec Gateway. In Proceedings of 1st IEEE International Conference on Information Technology, Gdańsk University of Technology, Gdańsk, 2008, pp.175-178.

[6] KENT S.: RFC 4303: IP Encapsulating Security Payload (ESP). Network Working Group, December 2005.

[7] KENT S., SEO K.: RFC 4301: Security Architecture for the Internet Protocol. Network Working Group, December 2005.

[8] FAROOQ U., MAJUMDAR S., PARSONS E.: High Performance Publish / Subscribe Middleware for Mobile Wireless Networks. Mobile Information Systems, Vol.3, Issue 2, 2007, pp. 107-132.

[9] FRANKEL S., KENT S., LEWKOWSKI R., OREBAUGH A., RITCHEY R., SHARMA S.: Guide to IPSec VPNs – Recommendations of the NIST. Computer Security Division, Information Technology Laboratory, Special Publication 800-77, Gaithersburg, December 2005.

[10] KIM O., MONTGOMERY D.: Behavioral and Performance Characteristics of IPSec/IKE in Large Performance Scale VPNs. Advances Network Technologies Division, National Institute of Standards and Technology, MO, 2003.

AUTHORS & AFFILIATION

[1.] HENRYK KRAWCZYK,
[2.] MARCIN BARYLSKI

[1.] FACULTY OF ELECTRONICS, TELECOMMUNICATIONS AND INFORMATICS, GDAŃSK UNIVERSITY OF TECHNOLOGY, POLAND

[2.] PLATFORM EXTENSION SERVICES
INTEL TECHNOLOGY POLAND