INTERNET 2011

The Third International Conference on Evolving Internet

June 19-24, 2011

Luxembourg City, Luxembourg

INTERNET 2011 Editors

Eugen Borcoci, University “Politehnica” Bucharest, Romania

Dirceu Cavendish, Kyushu Institute of Technology, Japan

Mark Yampolskiy, Leibniz-Rechenzentrum (LRZ) - Garching, Germany

INTERNET 2011

Foreword

The Third International Conference on Evolving Internet [INTERNET 2011], held between June 19 and 24, 2011, in Luxembourg, dealt with challenges raised by evolving Internet making use of the progress in different advanced mechanisms and theoretical foundations. The gap analysis aimed at mechanisms and features concerning the Internet itself, as well as special applications for software defined radio networks, wireless networks, sensor networks, or Internet data streaming and mining.

Originally designed in the spirit of interchange between scientists, the Internet reached a status where large-scale technical limitations impose rethinking its fundamentals. This refers to design aspects (flexibility, scalability, etc.), technical aspects (networking, routing, traffic, address limitation, etc), as well as economics (new business models, cost sharing, ownership, etc.). Evolving Internet poses architectural, design, and deployment challenges in terms of performance prediction, monitoring and control, admission control, extendibility, stability, resilience, delay-tolerance, and interworking with the existing infrastructures or with specialized networks.

We take here the opportunity to warmly thank all the members of the INTERNET 2011 Technical Program Committee, as well as the numerous reviewers. The creation of such a broad and high quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and efforts to contribute to INTERNET 2011. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions.

Also, this event could not have been a reality without the support of many individuals, organizations, and sponsors. We are grateful to the members of the INTERNET 2011 organizing committee for their help in handling the logistics and for their work to make this professional meeting a success.

We hope that INTERNET 2011 was a successful international forum for the exchange of ideas and results between academia and industry and for the promotion of progress in the area of the evolving Internet.

We are convinced that the participants found the event useful and communications very open. We also hope the attendees enjoyed the historic charm Luxembourg.

INTERNET 2011 Chairs:

Eugen Borcoci, University “Politehnica” Bucharest, Romania
Christian Callegari, University of Pisa, Italy
Dirceu Cavendish, Kyushu Institute of Technology, Japan
Emmanuel Chaput, ENSEEIHT / IRIT-CNRS, France
Danny De Vleeschauwer, Alcaltel-Lucent Bell Labs - Antwerp, Belgium
Jerome Galtier, Orange Labs, France

Terje Jensen, Telenor Corporate Development - Fornebu / Norwegian University of Science and Technology - Trondheim, Norway

Jaime Lloret Mauri, Polytechnic University of Valencia, Spain

Robert van der Mei, Centrum Wiskunde & Informatica, The Netherlands

Massimo Villari, University of Messina, Italy

Krzysztof Walkowiak, Wroclaw University of Technology, Poland

Sabine Wittevrongel, Ghent University, Belgium

Tingyao Wu, Alcatel-Lucent - Antwerpen, Belgium

Mark Yampolskiy, Leibniz-Rechenzentrum (LRZ) - Garching, Germany

Abdulrahman Yarali, Murray State University, USA

Vladimir Zaborovsky, Technical University - Saint-Petersburg, Russia

INTERNET 2011

Committee

INTERNET Advisory Chairs

Emmanuel Chaput, ENSEEIHT / IRIT-CNRS, France
Jaime Lloret Mauri, Polytechnic University of Valencia, Spain
Eugen Borcoci, University “Politehnica” Bucharest, Romania
Terje Jensen, Telenor Corporate Development - Fornebu / Norwegian University of Science and Technology - Trondheim, Norway
Abdulrahman Yarali, Murray State University, USA

INTERNET Research/Industry Chairs

Tingyao Wu, Alcatel-Lucent - Antwerpen, Belgium
Jerome Galtier, Orange Labs, France
Robert van der Mei, Centrum Wiskunde & Informatica, The Netherlands

INTERNET Special Area Chairs

Routing
Mark Yampolskiy, Leibniz-Rechenzentrum (LRZ) - Garching, Germany

Traffic
Vladimir Zaborovsky, Technical University - Saint-Petersburg, Russia

Performance
Sabine Wittevrongel, Ghent University, Belgium

Security
Christian Callegari, University of Pisa, Italy

Wireless
Dirceu Cavendish, Kyushu Institute of Technology, Japan

P2P
Krzysztof Walkowiak, Wroclaw University of Technology, Poland

Cloud and Internet
Massimo Villari, University of Messina, Italy

Multimedia
Danny De Vleeschauwer, Alcaltel-Lucent Bell Labs - Antwerp, Belgium

INTERNET 2011 Technical Program Committee

Onur Alparslan, Osaka University, Japan
Eleana Asimakopoulou, University of Bedfordshire, UK

Marcelo Emilio Atenas Urzúa, Polytechnic University of Valencia, Spain
Olivier Audouin, Alcatel-Lucent Bell Labs, France
Jacques Bahi, University of Franche-Comte, France
Nik Bessis, University of Bedfordshire, UK
Eugen Borcoci, University "Politehnica" Bucharest, Romania
Christian Callegari, University of Pisa, Italy
Jiannong Cao, Hong Kong Polytechnic University, Hong Kong
Dirceu Cavendish, Kyushu Institute of Technology, Japan
Emmanuel Chaput, ENSEEIHT / IRIT-CNRS, France
Claude Chaudet, ENST, France
Albert M. K. Cheng, University of Houston, USA
Andrzej Chydzinski, Silesian University of Technology, Gliwice, Poland
Danny De Vleeschauwer, Alcaltel-Lucent Bell Labs - Antwerp, Belgium
Matthias Dehmer, Vienna Bio Center, Vienna, Austria
Martin Dobler, FH VORARLBERG - Dornbirn, Austria
Mohamed Dafir El Kettani, ENSIAS - Université Mohammed V-Souissi - Rabat, Morocco
Armando Ferro Vázquez, Universidad del País Vasco - Bilbao, Spain
Jerome Galtier, Orange Labs, France
Miguel Garcia, Polytechnic University of Valencia, Spain
Bezalel Gavish, Southern Methodist University - Dallas, USA
Frans Henskens, University of Newcastle, Australia
Ye Huang, University of Fribourg, Switzerland
Terje Jensen, Telenor Corporate Development - Fornebu / Norwegian University of Science and Technology - Trondheim, Norway
Evangelos Kranakis, Carleton University, Canada
Danny Krizanc, Wesleyan University-Middletown, USA
Sławomir Kukliński, Warsaw University of Technology, Poland
Clement Leung, Hong Kong Baptist University, Hong Kong
Fidel Liberal, University of the Vascon Country, Spain
Jaime Lloret Mauri, Polytechnic University of Valencia, Spain
Henrique João Lopes Domingos, New University of Lisbon & CITI Research Center, FCT/UNL - Lisbon, Portugal
Shawn McKee, University of Michigan, USA
Henning Müller, University Hospitals of Geneva, Switzerland
Josep Rius, Polytechnic High School University of Lleida, Spain
Paul Sant, University of Bedfordshire, UK
Peter Schartner, University of Klagenfurt, Austria
Dimitrios Serpanosm, ISI/R.C. Athena & University of Patras, Greece
Pedro Sousa, University of Minho, Portugal
Parimala Thulasiraman, University of Manitoba - Winnipeg, Canada
Robert van der Mei, Centrum Wiskunde & Informatica ,The Netherland
Massimo Villari, University of Messina, Italy
Krzysztof Walkowiak, Wroclaw University of Technology, Poland
Sabine Wittevrongel Ghent University, Belgium
Tingyao Wu, Alcatel-Lucent - Antwerpen, Belgium
Mark Yampolskiy, Leibniz-Rechenzentrum (LRZ) - Garching, Germany
Vladimir Zaborovsky, Technical University - Saint-Petersburg, Russia

Copyright Information

Table of Contents

A Crosslayer-aware Bandwidth Aggregation & Network Condition Determination System Using Multiple Physical Links

Soma Bandyopadhyay, Shameemraj M Nadaf
Innovation Lab
Tata Consultancy Services
Kolkata, India
soma.bandyopadhyay@tcs.com, sm.nadaf@tcs.com

***Abstract*—This article presents a cross layer aware bandwidth aggregation and network condition determination of a fixed computing system dynamically. The proposed system transmits and receives data simultaneously through multiple physical interfaces, also determines the condition of communication channel/network associated with those physical interfaces while performing the bandwidth aggregation. This proposed cross layer system enhances the download and upload data transmission rates of the applications. Its core functional block resides in between the data link/MAC (medium access control) and network layer and encapsulates the existing multiple physical interfaces. Aggregation of bandwidth and determination of network condition by using the proposed cross layer system is presented with case study and experimental results in detail along with future research scope here.**

Keywords—TCP; UDP; PPP; Ethernet; Bandwidth aggregation; Cross layer architecture; Network Channel

I. INTRODUCTION

A system that enables information sharing across the layers is regarded as a cross layer system. In this article, focus has been given to design and develop a cross layer system to achieve bandwidth aggregation and to determine the network condition of multiple communication links connected with the active physical interfaces of a fixed computing system.

The cross layer system as presented here, enhances the bandwidth of a system significantly by adding up the available bandwidths of the existing active communication interfaces (wired and wireless) without performing any modifications in the physical and data link layer of the existing interfaces. At the same time it determines the channel/network condition associated with each active physical interface. It is mainly an adaptive bandwidth aggregation system. This system runs multiple sessions over multiple interfaces. The proposed system encloses existing network interfaces, i.e., both MAC and physical layers of those interfaces. It does not need any counterpart/negotiation in any node including the final destination or end system of the communication link. It can be used for any transport layer protocol like TCP (transmission control protocol) and UDP (user datagram protocol). Notably, it does not require any service level agreement and a proxy support.

The case study, as presented here, performs a video streaming by using one interface, and a file download by using another interface simultaneously with an improvement in throughput achieved. It also presents the results of channel condition determination obtained by cross layer co-ordination for the multiple physical interfaces.

The remainder of this article is organized as follows. First the related work in cross layer based bandwidth aggregation and network condition determination is presented, followed by an overview of the proposed system. The architecture of the system along with the details of experimental study and analysis are then described. The final section concludes this article with the future scope of the model.

II. RELATED WORK

Data transfer through multiple physical interfaces is broadly studied for improving the total available bandwidth entitled to the applications. Most of the related work in this area requires proxy architecture, service level agreement and counter component at the destination, to realize the bandwidth aggregation either in adaptive or non-adaptive manners. A network layer architecture consisting of an infrastructure proxy [1] or a multilink proxy [2] is applied for simultaneous use of multiple interfaces and aggregation of the throughput of heterogeneous downlink streams. The approach presented in [3] uses dynamic packet reordering mechanism of TCP streams over multiple links, also requires a network proxy. Service level agreement as well as proxy is used in the middle of the network for scheduling the packets through

ISBN: 978-1-61208-141-0

multiple interfaces in [4]. Session-layer striping architecture over multiple links is proposed in [5] based on single virtual layer socket. Furthermore [6], describes a network middleware called Horde, which allows application to control certain aspects of data stripping over multiple interfaces. This middleware architecture comprises three layers in which the higher layer provides an interface to interact with Horde, middle layer handles packet scheduling, bandwidth allocation, and the lower layer deals with network channels.

The framework based on cross layer concept, presented in [7], proposes adaptation across many layers of the protocol stack to support delay-critical applications in adhoc scenario, such as conversational voice or real-time video. The work presented in [8] makes changes to the five layers namely physical, data link, and application, network, and transport layer like TCP to provide seamless delivery of multimedia services in heterogeneous wireless networks. Some modifications have been made to the transport protocols which make deployment of such an approach difficult. A dynamic QoS negotiation scheme has been proposed in [9], to do bandwidth aggregation for video streaming in wireless networks.

Multiple interfaces of the same technology can also be striped for better performance at the link layer. The main idea of bandwidth aggregation on the link layer is to stripe data across a bundle of physical channels, as done in [10]. An adaptive inverse multiplexing technique has been described in [11] where IP Packets are fragmented by the multiplexor and tunneled through multiple links using multiple-link PPP over a link layer transfer protocol. A channel aggregation method in cellular networks is described in [12] where parity codes are applied across channels rather than across packets to improve the resilience. Another interesting approach is followed in [13], where it is proposed that users of WLANs should be able to multihome and split their traffic among all available access points, based on obtained throughput and a charged price. However, a link-layer solution of striping data through heterogeneous networks and to different IP addresses is not feasible because the link layer has no notion of IP. All these approaches described above demand modifications in both ends (i.e. server and client) to achieve bandwidth aggregation, and most of the approaches are tested in simulations, often based on very simple assumptions.

Network condition determination in real-time has also attracted quite a few research works. Most of the works in this area are based on RTT (round-trip-time) measurement, using a single or multiple probe packets. Theory, improvements and some implementations related to Network condition determination based on RTT mechanism have been discussed in [14]. UDP based probe packet mechanism is used to determine the network condition and do rate control in a P2P (peer- to- peer) based video streaming application in [15]. Depending on the determined RTT values the upload bit rates are decided and the available bandwidth is calculated. The bandwidth determined decides whether to accept or reject a new P2P client requesting a connection at a random data rate. This method results in unstable RTT measurements since set of peers may be located anywhere in the world. Pair of probe packets with a fixed delay and packet pair dispersion technique is used to measure the available bandwidth in [16]. Along with the RTT calculation the time delay between the probe packets is measured and analyzed to predict the network bandwidth here. In [17] a variable size probe packet is used to measure the available bandwidths. Along with RTT the packet size is also considered to estimate the network capacity. A testing network including a test box measurement infrastructure is used here.

However, these research efforts are based on single interface and not use multiple interfaces simultaneously and need counterpart in the destination. These approaches also do not make use of any cross layer technique.

In [18], a bandwidth aggregation mechanism without network condition determination based on the application layer has been described. Here, the application data is distributed over multiple sessions communicating over multiple interfaces and assembled to get the complete application traffic. However, this system does not use any cross layer approach. Also importantly it is not adaptive in nature.

III. SYSTEM OVERVIEW

The cross layer aware bandwidth aggregation and network condition determination system as proposed here creates a ‘virtual physical’ interface by using its associated network driver module, which encapsulates all the existing active physical interfaces present in the computing system. It does not perform any modifications in the physical and data link layer of the existing physical interfaces. The virtual physical interface resides as the only default entry in the routing table, and provides a single communication pipe from IP (internet protocol) to the physical interfaces, starting from applications and vice-versa. The packets pertaining to different sessions are distributed to the multiple physical interfaces by this module for transmission, and also upon reception the packets are processed by this module, and pushed to the higher layers. Since there is no counterpart at the receiving side, the packets of a session are distributed through one physical interface only but packets of next session get assigned to the other physical interface. Depending upon the QoS requirement (if any) of an application the packets of that session can be assigned to a physical interface based on its network condition. Thus this proposed module enhances the bandwidth of a system significantly by adding up the available bandwidths of the existing active communication interfaces. It estimates the channel condition of the active physical interfaces by using probe-packet mechanism, and analyzing the transport header statistics.

The proposed system can be configured to operate in following modes:

•Bandwidth Aggregation mode: aggregates the bandwidth of multiple network interfaces.

•Bandwidth Aggregation with Network Determination mode: aggregates the bandwidth of multiple network interfaces and also determines the channel condition of each network interface dynamically at predefined time intervals (configurable).

It has two components - user space and kernel space as depicted in Fig. 1 respectively. User space component exposes APIs (application programming interface) to take the user inputs as well as system captured inputs.

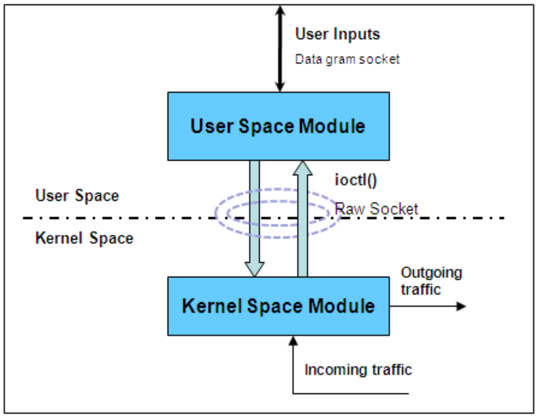


Figure 1. Building blocks of the system

User inputs:

1. Domain name of the server to which probe packets are sent for network condition determination

2. Time interval for performing network condition determination

3. QoS requirement of an application (optional)

System captured inputs:

- Information about the network interfaces (i.e. interface identifier, IP addresses, IP address of Gateways if any)

User inputs are provided by the user through command line. System captured inputs are obtained by doing an ioctl () function call with a datagram socket.

User Space and kernel space have a close bonding and both these components exchange control information by using ioctl () function call from user space component to kernel space component with a raw socket. The various operations performed by the user space components are as follows:

- GETINFO: to collect information available with kernel space module about the network interfaces for aggregation.
- SETINFO: to pass the information about the available network interfaces to the kernel space module.
- NDMINFO (Network Diagnostics): to pass information related to network condition determination, QoS requirement of an application to the kernel space module and also to collect the information about the channel condition of the network interfaces from the kernel space module.

IV. ARCHITECTURE OF PROPOSED SYSTEM

The architecture of the proposed cross layer aware bandwidth aggregation and network condition determination system is portrayed in the current section. The overview of the system architecture along with its functional components is depicted in Fig. 2. The proposed system acts as a gateway of all the data paths from upper layers to the physical interfaces and vice versa. It distributes the data flow among the active physical interfaces for transmission based on the network condition determination feedback. QoS requirement, incase available, is also going to be used for data flow distribution. It receives the data from the interfaces and passes it to the respective applications.

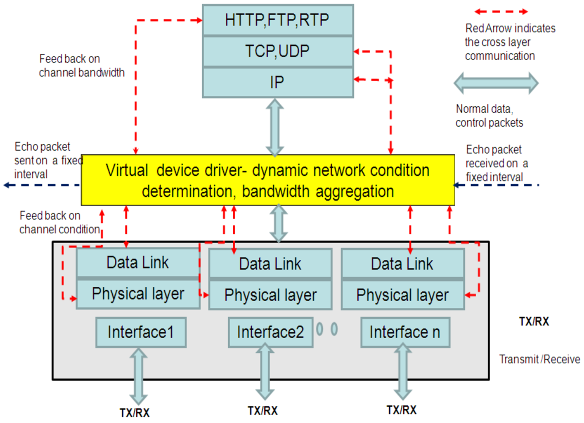

HTTP,FTP,RTP
TCP,UDP
IP
Red Arrow indicates the cross layer communication
Normal data, control packets
Feed back on channel bandwidth
Echo packet sent on a fixed interval
Virtual device driver- dynamic network condition determination, bandwidth aggregation
Echo packet received on a fixed interval
Feed back on channel condition
Data Link
Physical layer
Data Link
Physical layer
Data Link
Physical layer
Interface1
Interface2
Interface n
TX/RX
Transmit /Receive
TX/RX
TX/RX
TX/RX

Figure 2. Functional blocks of the system

The proposed system creates a virtual interface and assigns an IP address, net mask to it and adds this interface as the default entry in the routing table. All the applications data coming from upper layer uses this IP address as the source address. The proposed system replaces its own IP address with the corresponding active physical interfaces IP addresses, while distributing the packets to those interfaces and performs the necessary checksum calculations for IP, and transport protocols headers as well. After receiving the packets it replaces the actual IP addresses of the interfaces with its own IP address and performs the necessary checksum calculations for IP and transport protocols headers. It distributes the data packets from application based on some predefined identifiers for example port number (HTTP packet or FTP packet etc.) and QoS requirement as specified by the application through the user space module. It sends the distributed data packets directly to the transmit queue of its slave interfaces i.e., the active physical interfaces. It uses packet filtering (net filter) mechanism and associates a hook function for processing the received packets. The hook function is used to filter the packets just after their reception by the active interfaces. The associated hook function of the packet-filter performs the necessary modifications in the data packets, and assembles the data packet before sending to the application.

The proposed system uses a predefined ICMP echo packet for measuring the network channel condition. The ICMP echo packet is sent to the public IP address (for example-www.google.com) defined by user (defined via user input 1) simultaneously through the existing multiple active interfaces. The destination with the public IP address sends back the echo-reply to active physical interfaces. It determines the time difference between the sent ICMP echo-request and received ICMP echo-reply packets i.e. the round trip time (RTT) for the active interface and estimates the network condition. The time difference with a higher value signifies a poor network condition. The echo packet is sent at a fixed time interval (configurable, via user input 2), for this the proposed system maintains a timer. Based on the network condition and QoS requirement (via user input 3), along with the above described address mapping from virtual to physical interface and vice-versa, the data packets distribution among the physical interfaces is performed by this system. There is also a provision to judge the transport header (TCP) statistics and to take the average RTTs obtained from both the transport header and ICMP probe packet for assessment of network condition; this is a future scope of the current work.

V. EXPERIMENTAL ANALYSIS

Here the proposed system is implemented using a Linux based PC, two CDMA 1XRTT based interfaces are used for internet access and aggregation of bandwidth, and the Wireshark network analyzer tool running on the PC is used for analyzing the network traffic. In this Experiment, the identifier used for data flow distribution to multiple physical interfaces is the port number (HTTP packet or FTP packet). Here, the numbers of applications running are equivalent to the numbers of physical interfaces. The experiment consists of following two phases.

1. Phase 1- tasks are carried out with a single CDMA 1XRTT interface in absence of the proposed system.
2. Phase 2 - tasks are carried out with two CDMA 1XRTT interfaces in presence of the proposed system.

Following two tasks are carried out simultaneously in each phase of the experiment:

1. Video streaming from YouTube link.
2. File download from ftp link.

In case of the first phase both video streaming and file download take place via a single interface whereas in case of second phase the video streaming takes place through one interface and the file download takes place via the other interface. Throughputs achieved in both the phases are mentioned in Table 1. The average throughput obtained by single interface in phase 1 is 5.7602 kbps (kilobytes per second), this throughput is shared amongst the two tasks namely video streaming and file download.

In case of phase 2, average throughput obtained by interface 1 is 6.1565 kbps (kilobytes per second) and average throughput obtained by interface 2 is 5.8978 kbps (kilobytes per second). As mentioned earlier the two tasks video streaming and the file download take place via interface 1 and interface 2 respectively, hence the throughput received by the tasks is almost twice the throughput received in case of phase1.Thus a significant improvement is seen in the throughput achieved by use of proposed aggregation module. Throughput graphs shown in Fig. 3 & Fig. 4 depict the variation of throughput with elapsed time in case of Phase 1 and Phase 2 of the experiment respectively, whereas the Fig. 5 shows the effective aggregated throughput achieved by the proposed system.

Table 1. Improvement in Throughput

Phase of Experiment	**IP address of Interfaces**	**Average throughput achieved in kilobytes per second**
Phase 1 : one CDMA 1xRTT interface	Single interface : 14.96.13.4	Single interface: 5.7602
Phase 2 : two CDMA 1xRTT interfaces	Interface 1 : 14.96.25.157 Interface 2 : 14.96.17.240	Interface 1 : 6.1565 Interface 2 : 5.8978 Combined : 12.0543

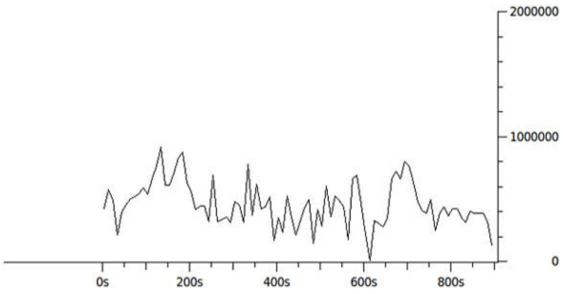


Figure 3. Throughput in Phase 1 (x-axis = time elapsed in seconds, y-axis = bits per second)

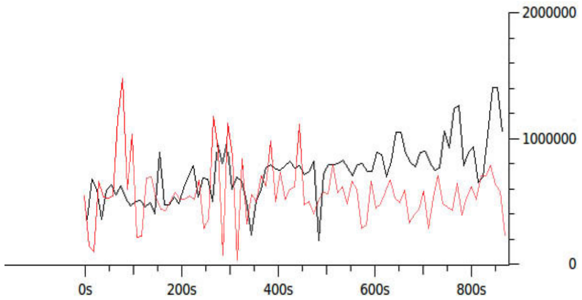


Figure 4. Throughput in Phase 2 (x-axis = time elapsed in seconds, y-axis = bits per second, black – interface 1 throughput, red – interface 2 throughput)

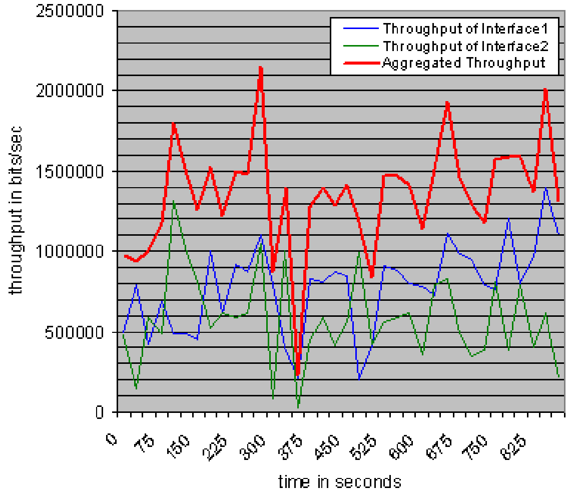


Figure 5. Aggregated throughputs in Phase 2

Determination of network condition of active physical interfaces is done by the aggregation module using the ICMP Echo packets here. Variation of RTT values with elapsed time at intervals of 20 seconds for the interfaces in phase 2 (with the proposed system) is shown in Fig. 6. It is observed from Fig. 6 that as the throughput increases values of RTTs obtained is low and the RTTs values obtained is high with reduction in throughput. However the variation in RTT values is not consistent due to the dynamic network condition of wireless interface and queuing of received packets by net filter hook mechanism. The ICMP Probing is done to a public IP address (e.g. www.google.com) here and also there is no proxy involved in between the source and destination. Hence the delay encountered at the hops in between the source and destination is not considered for RTT measurement. It should be noted that since the proposed system is based on a cross layer solution different use cases can be implemented by making negotiations at the multiple layers of the protocol stack as required in general.

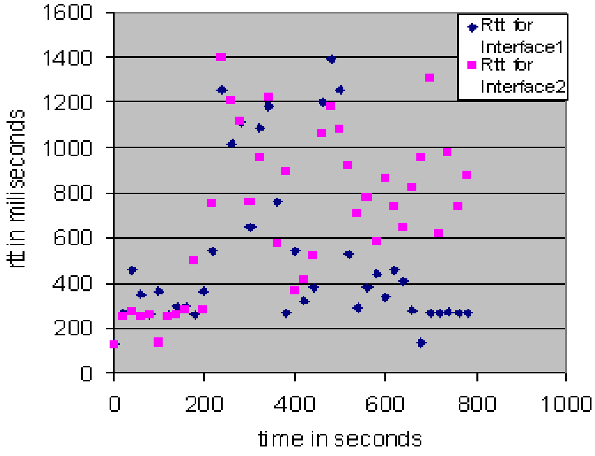


Figure 6. RTT measurement in Phase 2

VI. CONCLUSION AND FUTURE SCOPE

In this paper a cross layer aware bandwidth aggregation and network condition determination system with multiple physical interfaces is proposed, which performs the bandwidth aggregation and also determines the channel condition of the existing interfaces. Bandwidth aggregation is performed based on the determined channel condition of the existing interfaces. The experimental results with multiple interfaces, as presented, have shown a significant improvement in throughput achieved. The network condition determination is performed by sending ICMP echo packet to a fixed public destination periodically. The proposed system coordinates with application layer to get the QoS requirement of the applications through its exposed APIs. This does not need any counterpart in any node including the final destination or end system of the communication link, therefore, easy to manage, deploy and is cost effective. It can be used for any transport layer protocol like TCP (transmission control protocol) and UDP (user datagram protocol). Importantly, it does not require any proxy support and service level agreement.

There is a scope for future research work to decouple the network condition determination module from the bandwidth aggregation module. Optimal utilization of network bandwidth i.e., to use unused bandwidth of any interface, at a particular time interval based on the feedback of network condition determination function, and managing multiple number of applications, when the number of applications are greater than the number of active physical interfaces. Network condition determination by taking the average of RTTs obtained from both ICMP echo packet and transport (TCP) header statistics, and also transferring the session from one interface to other with out breaking the connection when the network condition degrades is a part of our future research activity. This proposed system can be further customized to achieve enhancement of QoS of different multimedia applications like video conferencing, remote healthcare system etc.

REFERENCES

[1] K. Chebrolu and R. R. Rao, “Bandwidth aggregation for real-time applications in heterogeneous wireless networks”, IEEE Transactions on Mobile Computing, vol. 5, no. 4, pp. 388–403, 2006.

[2] K. Evensen, D. Kaspar, P. Engelstad, A.F. Hansen, C. Griwodz, and P. Halvorsen, “A network-layer proxy for bandwidth aggregation and reduction of IP packet reordering”,

http://simula.no/research/nd/publications/Simula.ND.369/simula_pdf_file, 28.04.2011.

[3] K. Chebrolu, B. Raman, and R. R. Rao, “A network layer approach to enable TCP over multiple interfaces”, IEEE Transactions on Wireless Networks, vol. 11, no. 5, pp. 637–650, 2005.

[4] J.C. Fernandez, T. Taleb, M. Guizani, and N. Kato, “Bandwidth Aggregation-Aware Dynamic QoS Negotiation for Real-Time

Video Streaming in Next-Generation Wireless Networks”, IEEE Transactions on Multimedia, vol. 11, no. 6, pp. 1082-1093, 2009.

[5] A. Habib, N. Christin, and J. Chuang, “Taking advantage of multihoming with session layer striping”, http://www.andrew.cmu.edu/user/nicolasc/publications/gistripping.pdf, 28.04.2011.

[6] Horde: Flexible Application Driven Network Striping, http://publications.csail.mit.edu/abstracts/abstracts05/horde/horde.html, 28.04.2011.

[7] Cross Layer Design of Ad-hoc Wireless Networks for Real-Time Media,

http://www.stanford.edu/~zhuxq/adhoc_project/adhoc_project.html, 28.04.2011.

[8] T. Taleb, K. Kashibuchi, A. Leonardi, S. Palazzo, K. Hashimoto, N. Kato, and Y. Nemoto, “A cross-layer approach for an efficient delivery of TCP/RTP-based multimedia applications in heterogeneous wireless networks”, IEEE Transactions on Vehicular Technology, vol. 57, no. 6, pp. 3801-3814, 2008.

[9] T. Taleb, T. Nakamura, and K. Hashimoto, “Bandwidth aggregation-aware dynamic Qos negotiation for real-time video streaming in next-generation wireless networks”, IEEE Transactions on Multimedia, vol. 11, no. 6, pp. 1082-1093, 2009.

[10] H. Adiseshu, G. Parulkar, and G. Varghese, “A reliable and scalable striping protocol”, ACM SIGCOMM, vol. 26, no. 4, pp. 131-141, 1996.

[11] A.C. Snoeren, “Adaptive inverse multiplexing for wide-area wireless networks”, GLOBECOM, vol. 3, pp. 1665-1672, 1999.

[12] J. Chesterfield, R. Chakravorty, I. Pratt, S. Banerjee, and P. Rodriguez, “Exploiting diversity to enhance multimedia streaming over cellular links”, INFOCOM, vol. 3, pp. 2020-2031, 2005.

[13] S. Shakkottai, E. Altman, and A. Kumar, “The case for non-cooperative multihoming of users to access points in IEEE 802.11 WLANs”, INFOCOM, pp. 1-12, 2006.

[14] J. Curtis and T. McGregor, “Review of Bandwidth Estimation Techniques”,

http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.29.779, 28.04.2011.

[15] M. Saubhasik and U. Schmidt, “Bandwidth Estimation and Rate Control in Bit Vampire”,

http://www.uweschmidt.org/files/network_project.pdf, 28.04.2011.

[16] J. Navratil and R. Les. Cottrell, “ABwE: A Practical Approach to Available Bandwidth Estimation”, http://www.nlanr.net/PAM2003/PAM2003papers/3781.pdf, 28.04.2011.

[17] T.G. Sultanov and A.M. Sukhov, “Simulation technique for available bandwidth estimation”,

http://arxiv.org/PS_cache/arxiv/pdf/1007/1007.3341v1.pdf, 28.04.2011.

[18] Soma Bandyopadhyay, Arpan Pal, and Shameemraj Nadaf, “A Novel Bandwidth Aggregation System Using Multiple Physical Links“, ICCAIE, pp. 34-37, 2010.

Trustworthy Distribution and Retrieval of Information over HTTP and the Internet

Isaí Michel Lombera, Yung-Ting Chuang, Peter Michael Melliar-Smith, Louise E. Moser
Department of Electrical and Computer Engineering
University of California, Santa Barbara
Santa Barbara, CA 93106 USA
imichel@ece.ucsb.edu,ytchuang@ece.ucsb.edu,pmms@ece.ucsb.edu,moser@ece.ucsb.edu

***Abstract*— This paper describes a novel information distribution and retrieval system, named iTrust, that operates over HTTP and the Internet and that provides trustworthy access to information. iTrust is a completely distributed system, with no centralized mechanisms and no centralized control, that avoids subversion or censorship of information. Individuals submit information they wish to share to nodes on the Internet that distribute metadata to random participating nodes. Likewise, users submit requests containing metadata for information they wish to retrieve to random participating nodes. The paper presents an overview of the iTrust strategy, implementation, user interface, and performance. iTrust can effectively enable citizens to distribute and retrieve information over the Internet, even in the presence of subverted or non-operational nodes.**

***Keywords*-information distribution and retrieval; distributed search; trustworthy information access; citizen-centric service**

I. Introduction

Our trust in the accessibility of information over the Internet and the Web (hereafter referred to as the Internet) depends on benign and unbiased administration of centralized search engines and centralized search indexes. Unfortunately, the experience of history, and even of today, shows that we cannot depend on such administrators to remain benign and unbiased in the future.

To ensure that the distribution and retrieval of information over the Internet is not subverted or censored, alternative search mechanisms must be provided. The availability of multiple search engines is important, but that protection is weakened by the small number of search engines available today. An alternative to centralized search, an effective completely distributed search, without centralized mechanisms and without centralized control, is an important assurance to users of the Internet that a small number of administrators cannot prevent them from distributing their ideas and information, and from retrieving the ideas and information of others.

The thesis of this research is that the distribution and retrieval of information without centralized search engines and without centralized search indexes is a critical enabling technology for users to have trust in the Internet for access to information. It is important to ensure that such a trustworthy information distribution and retrieval system is available when it is needed, even though a user might normally use a conventional centralized search engine.

In this paper, we describe iTrust, a novel information distribution and retrieval system that operates over the HyperText Transfer Protocol (HTTP) and the Internet and that provides trustworthy access to information. Section II of the paper presents an overview of the iTrust strategy, Section III describes our implementation of iTrust based on HTTP, and Section IV describes the user interface. Performance results are presented in Section V. Related work, and the conclusion and future work, are presented in Sections VI and VII, respectively.

II. Overview of iTrust

The iTrust information distribution and retrieval system involves no centralized mechanisms and no centralized control. We refer to the nodes that participate in an iTrust network as the participating nodes or the membership. An iTrust network might correspond to participants with specific interests, or it might correspond to a social network. Multiple iTrust networks within the Internet may exist at any point in time, and a node may participate in several different iTrust networks at the same time.

In an iTrust network, some nodes (the source nodes) produce information, and make that information available to other participating nodes. The source nodes produce metadata that describes their information, and distribute that metadata to a subset of participating nodes that are chosen at random, as shown in Figure 1. The metadata are distinct from the information that they describe, and include a list of keywords and the URL of the source of the information.

Other nodes (the requesting nodes) request and retrieve information. Such nodes generate requests (also referred to as queries) that refer to the metadata, and distribute the requests to a subset of the participating nodes that are chosen at random, as shown in Figure 2.

The participating nodes compare the metadata in the requests (queries) they receive with the metadata that they hold. If such a node finds a match, which we call an encounter, the matching node returns the URL of the associated information to the requesting node. The requesting node then

Copyright (c) IARIA, 2011. ISBN: 978-1-61208-141-0

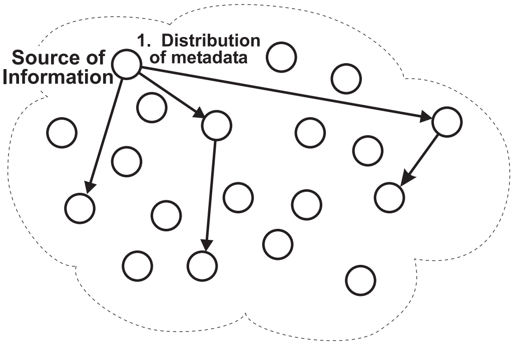


Figure 1. A node (a source node) distributes metadata, describing its information, to randomly selected nodes in the network.

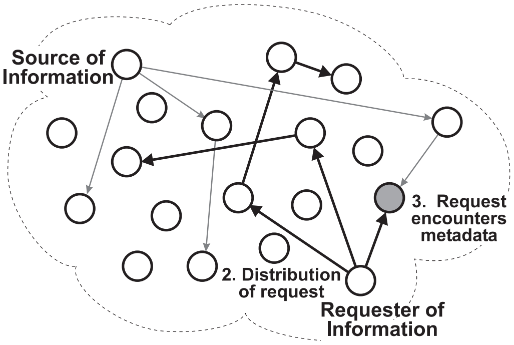


Figure 2. A node (a requesting node) distributes its request to randomly selected nodes in the network. One of the nodes has both the metadata and the request and, thus, an encounter occurs.

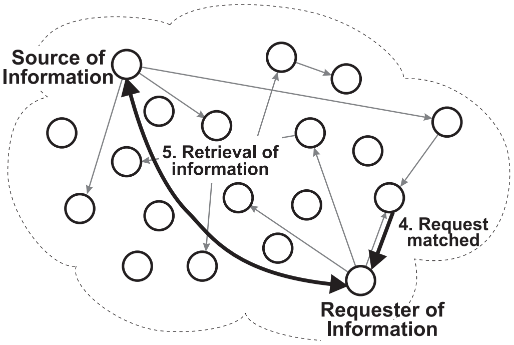


Figure 3. A node matches the metadata and the request and reports the match to the requester, which then retrieves the information from the source node.

uses the URL to retrieve the information from the source node, as shown in Figure 3.

Distribution of metadata and requests to relatively few nodes suffices to achieve a high probability that a match occurs. Moreover, the strategy is robust. Even if some of the randomly chosen nodes are subverted or non-operational, the probability of a match is high, as shown in Section V. Moreover, it is not easy for a small group of nodes to subvert the iTrust mechanisms to control which information is delivered and which is suppressed.

III. HTTP Implementation

The current implementation of iTrust uses HTTP to distribute metadata and requests. Each node is implemented using PHP (PHP: Hypertext Preprocessor) on an Apache Web server, thereby allowing any user with a Web browser on any platform to interact with the node. Node information is stored locally in an SQLite database. Multiple nodes can be installed on a single Web server by creating multiple virtual Web sites; multiple nodes on a single Web server have separate SQLite databases.

A. Membership

The membership list for a node is stored locally in an SQLite database table that contains node records whose fields are the node identifier and the node address. The node address may be either an IP address or a URL. A node is not verified when it is added to the membership as there is no guarantee that a node is always available. For example, a cell phone or a laptop with a WiFi connection may enter and leave its base station signal range multiple times a day. In practice, the only restriction on node addresses is that the Web site document root has Web server write permissions for saving uploaded resources.

B. Resources

Resources are files or groups of files uploaded to nodes in the membership. The list of resources on a node is stored in an SQLite database table that contains resource records whose fields are a resource handle, file path, and expiration date. The resource handle is a shortened random name (typically with 32-64 characters), which can be referenced across participating nodes. The file path is the name of the file on the local node disk. Thus, a participating node retrieving data may simply request a resource by the handle to the source node, instead of using the entire file path. The expiration date specifies when a given resource and associated keywords should be deleted. It allows time-sensitive information to be removed automatically from queries past the expiration date when the information is no longer relevant. For example, yesterday's weather forecast is not included in today's weather queries.

Resource files can be placed on a node using the Common Gateway Interface (CGI) by means of a file dialog, or through the cURL package provided by PHP. In the case of cURL, the file(s) is fetched directly by the node and written to the local disk.

When a resource is entered into the SQLite database, metadata for the resource file is generated using the Apache Tika/Lucene packages. These packages classify metadata based on content such as text strings, and file attributes such as data type, file size, *etc*. Also, the user may supply additional metadata keywords. The metadata keywords are stored in a SQLite database table, and the associations

Copyright (c) IARIA, 2011. ISBN: 978-1-61208-141-0

between a resource and a keyword are stored in a separate table, thus normalizing the database.

C. Metadata distribution

Periodically, metadata keywords and other information about one (or more) resources on a node are collated and compiled into an XML file (also referred to as the metadata list) which describes the resource. The periodicity depends on the node and the platform; however, in practice, the user can update the metadata list at any time by clicking a button or running a cron job. The resource description in the metadata list includes the resource handle, file path, and expiration date for the resource. The metadata list includes all associated keywords for the resource.

After creation of the metadata list, random nodes in the membership are contacted and informed of a metadata update by means of an HTTP POST statement. The number of random nodes that are selected for metadata distribution is a tunable parameter in the iTrust node configuration file. The contact message includes the source node IP address and metadata list URL, which are stored on the receiving node. Each contacted node decides if and when to retrieve the metadata list. The retrieval period is receiving node dependent, so as not to trigger an instant download of the metadata list file by multiple nodes.

If a retrieval occurs, the receiving node retrieves the metadata list file from the source node and processes the XML. The metadata list is stored on the receiving node. If there are multiple resources represented by sets of metadata in the metadata list, then processing continues on the next set of metadata, until the end of the metadata list is reached. XML processing is performed using the SimpleXML PHP extension.

D. Query relaying

Search queries (requests) are the main interaction between the user and an iTrust node and, as such, require the most processing. A search query originates at a single node, but the query message may be relayed among multiple nodes in the iTrust network. A query message may take any available network path with the sole restriction that a node never relays the same query twice.

The query field is a simple HTML form text box on the current node; the command to begin the query is detection of pressing a submit button or enter key. The query text itself is URL encoded to facilitate later operations; no custom processing on the query text (*e.g.*, duplication detection, grammar checking, *etc.*) is performed.

Two additional variables are created before a node sends the query: the node IP address and the query identifier. The node IP address is read from the iTrust configuration file; it ensures that queried nodes know which node originally sent the query. The query identifier ensures that no query is relayed twice and helps manage multiple queries sent from a single node.

Once the query is ready to be sent, multiple random nodes are selected from the node database table and the query is packaged into an HTTP POST statement. The frequency of node selection is another customizable iTrust configuration variable and need not be the same at different nodes. Node selection for querying is not necessarily the same as node selection for metadata list distribution; both variables may be set by the administrator of the node.

A node sends the HTTP POST statement to random nodes, and each node (both sender and receivers) saves a copy of the query before processing. If any text in the query matches the metadata keywords, an encounter occurs. The queried node then sends an HTTP POST response back to the originating node. The originating node is obtained from the sender node's IP address given in the query package. The response includes the query identifier (again obtained from the query package), the encountered node's IP address (hereafter referred to as the source node), and the resource handle of the matching resource. Additionally, the querying node saves the response in an SQLite database table for later processing.

A queried node, regardless of whether or not it has an encounter, may relay the query as if it were the originating query node. The only difference is that it does not recreate the two additional variables (source IP address and query identifier); those variables are relayed without modification. However, before relaying the query, the node checks the query identifier to ensure that the node has not already seen the query. If the node has already saved the query identifier, it does not relay the message.

Note that the current node in this context may be either the node sending the query or the node receiving the query. In case the receiving node has not yet relayed the query, it relays the query to nodes randomly selected from the membership and records the query identifier. In case the receiving node has already relayed the query, the receiving node ignores the query. Because of the decentralized random nature of iTrust, the original node that sent the query might have the same query relayed back to itself. In this case too, the current node ignores the query.

After waiting for responses a certain amount of time, the querying node displays all of the source nodes with appropriate resource handles. All encounters are thus recorded on the querying node, and the user is informed of which nodes have resources matching the query.

E. Retrieving resources

All query results are recorded in the requesting node's SQLite database table, *i.e.*, the source node IP address and resource handle. When the user selects a query result, the source node is sent an HTTP GET statement with the resource handle, and the source node returns the resource file directly to the user. Alternatively, the source address and resource handle can be encoded directly into a URL; the user then accesses the file using an HTML anchor tag.

IV. User Interface

The iTrust user interface is a Web-based interface where the user can both administer and query the nodes through Web pages. Query results from multiple nodes are presented in a single Web page following a query. Node administration and user queries are separated into distinct Web pages to keep tasks distinct and easily manageable.

A. *Node administration*

The user may change the membership, add source nodes, distribute metadata, and perform other administration tasks through the administration interface shown in Figure 4.

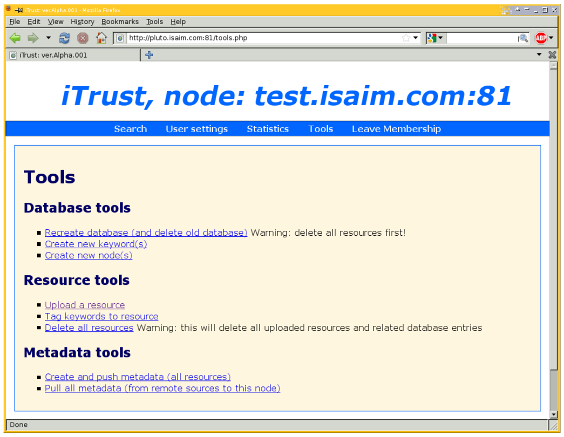
A node is added to the membership by entering the node IP address or URL on a comma delimited list inside an HTML form text box. Double listing is not permitted; duplicates are removed from the list. However, multiple nodes are permitted as long as the Web site document root is distinct (*e.g.*, both www.example.com/foo and www.example.com/bar are allowed).

Figure 5 shows the resource insertion Web page. A resource is added to a node by means of an HTML form file control; this control permits the user to upload a file from his/her local machine. Alternately, a Web site URL can be specified, and the node then fetches the contents at that URL. The uploaded contents are post-processed, using the Apache Tika/Lucene package, to generate descriptive metadata (*i.e.*, keywords) automatically. The user can customize several parameters for metadata creation, including indexing by file raw content (literal text strings) or file meta content (file size, type, *etc.*). In addition to automatic metadata creation for an uploaded resource, the user may add new keywords or remove existing keywords. Finally, the user may assign an expiration date to the resource.

Administration tasks also include file administration functions to allow the user to setup, restore, or reset iTrust nodes easily. Clearing the membership, deleting all resources and metadata associations, and resetting a node to its initial setup state can all be done with a single button click. The task of pushing all metadata changes to random nodes is also accomplished with a single button click.

B. *User queries*

The user may perform queries, view the query results, and obtain resources through the user interface.


iTrust: ver.Alpha.001 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
iTrust: ver.Alpha.001
Search
User settings
Statistics
Tools
Leave Membership
Done

http://pluto.isaim.com:81/tools.php

iTrust, node: test.isaim.com:81

Tools

Database tools

- Recreate database (and delete old database) Warning: delete all resources first!
- Create new keyword(s)
- Create new node(s)

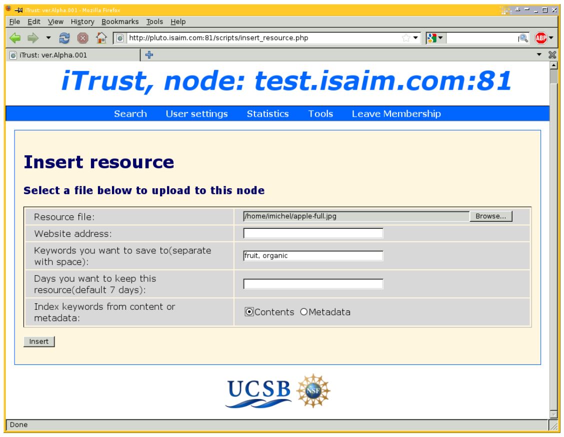
Resource tools

- Upload a resource
- Tag keywords to resource
- Delete all resources Warning: this will delete all uploaded resources and related database entries

Metadata tools

- Create and push metadata (all resources)
- Pull all metadata (from remote sources to this node)

Figure 4. The administration interface.



http://pluto.isaim.com:81/scripts/insert_resource.php

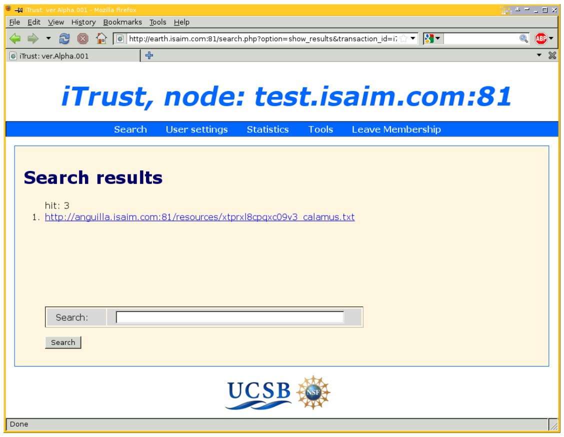
iTrust, node: test.isaim.com:81

Insert resource

Select a file below to upload to this node

Resource file:	/home/imichel/apple-full.jpg
Website address:	
Keywords you want to save to(separate with space):	fruit, organic
Days you want to keep this resource(default 7 days):	
Index keywords from content or metadata:	● Contents ○ Metadata

Figure 5. The insert resource Web page.



http://earth.isaim.com:81/search.php?option=show_results&transaction_id=i

iTrust, node: test.isaim.com:81

Search results

hit: 3

1. http://anguilla.isaim.com:81/resources/xtprxl8cpqxc09v3_calamus.txt

Figure 6. The query results Web page.

Querying is done through a single HTML form text box, whereupon the query is registered on the node and distributed throughout the iTrust network. The user is shown a status/wait Web page while the query is relayed among nodes; a result Web page is shown after a wait page timeout. The default timeout is 3 seconds and, thus, a query incurs a 3 second latency between initialization of the query and display of the query results. However, the wait page timeout is also configurable by the node administrator.

Figure 6 shows the query results displayed on a new Web page (the wait page automatically redirects to the new page) in a simple HTML list. Each encounter is shown as a list item with the source address and resource handle encoded into a single URL.

The user may click the URL to retrieve the resource file; the format of the file is the originally uploaded format (there is no MIME-type modification). If the Web browser recognizes the file type, it handles the data accordingly; otherwise, it calls the operating system to open the data file.

C. User settings

For querying, the three primary user settings (which the user sets on the user settings page) are the number of nodes to which the metadata are distributed, the number of nodes to which the requests are distributed, and the search duration.

The number of nodes to which the metadata are distributed and the number of nodes to which the requests are distributed must, of course, be less than the number of participating nodes in the membership.

The search duration refers to the lifetime that a search query exists. The user may specify how many days a query will be stored in the database. When a user initiates a query, the system adds its creation time to the database. Later, when the user initiates a new query, the system checks and deletes expired queries from the database.

These user settings apply to the entire duration of a search session. The search session starts when a user accesses the search Web page and ends when the user exits the browser window or tab. The PHP session functions are used to automate this process.

V. PERFORMANCE EVALUATION

If a node receives a request and it holds metadata that matches the request, we say that the node has a match. In the performance evaluation, we consider the probability of a match, using both analysis and simulation based on our HTTP implementation. We assume that all of the participating nodes have the same membership set. In addition, we assume that the Internet is reliable and that all of the participating nodes have enough memory to store the source files and the metadata.

A. Analysis

In an iTrust network with a membership of n nodes, we distribute the metadata to m nodes and the requests to r nodes. The probability p that a node has a match then is:

$$p = 1 - \left(\frac{n-m}{n}\right)\left(\frac{n-m-1}{n-1}\right)\dots\left(\frac{n-m-r+1}{n-r+1}\right) \qquad (1)$$

Formula 1 holds for $n \geq m + r$. If $m + r > n$, then $p = 1$.

As above, we distribute the metadata to m nodes and the requests to r nodes in an iTrust network with a membership of n nodes. But now we introduce another variable x, which represents the proportion of the n nodes that are operational. In an iTrust network with a membership of n nodes, where x nodes are operational, the probability p that a node has a match is:

$$p = 1 - \left(\frac{n-mx}{n}\right)\left(\frac{n-mx-1}{n-1}\right)\dots\left(\frac{n-mx-r+1}{n-r+1}\right) \qquad (2)$$

Formula 2 holds for $n \geq mx + r$. If $mx + r > n$, then $p = 1$.

Figures 7, 8, and 9 show the probability p of a match obtained from Formulas 1 and 2 with $n = 72$ nodes where $x = 100\%$, 80%, and 60% of the participating nodes are operational, respectively, as a function of $m = r$ (the number of nodes to which the metadata and requests are distributed). As we see from the graphs, the probability p of a match increases and asymptotically approaches 1, as $m = r$ increases.

B. Simulation

Using our HTTP implementation described in Section III, we performed simulation experiments to validate the analytical formulas given above. In our simulation, we used libCURL (which is a free client-side URL transfer library for transferring data using various protocols) to collect the match probabilities.

Before we run our simulation program, we delete all resources and data from the SQLite databases. Next, the program adds all the nodes to the membership. Once all the nodes are added to the membership, we supply the number of nodes for distribution of metadata and requests, and the proportion of operational nodes, to the simulation program. Next, we call the source nodes to upload files and the program then creates the corresponding metadata. Then, the program randomly selects the nodes for metdata distribution and distributes the metadata to those nodes. Next, the program randomly selects the nodes for the requests and distributes the requests. If one or more nodes returns a response, there is a match and the simulation program returns 1; otherwise, there is no match and the simulation program returns 0.

Figures 7, 8, and 9 show the simulation results with 72 nodes where 100%, 80%, and 60% of the participating nodes are operational, respectively. As we see from these graphs, the simulation results are very close to the analytical results calculated from Formulas 1 and 2 where 100%, 80%, and 60% of the participating nodes are operational.

The lesson we learned from this performance evaluation is that iTrust retains significant utility even in the case where a substantial proportion of the nodes are non-operational.

VI. RELATED WORK

Today, centralized search engines are used commercially for Internet search, where metadata for the information are held in a centralized search index [2]. Requests are submitted to the central site, where they are matched against

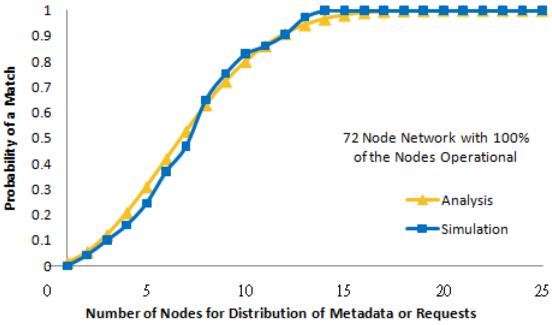


Figure 7. Match probability vs. number of nodes for distribution of metadata and requests in a network with 72 nodes where 100% of the nodes are operational.

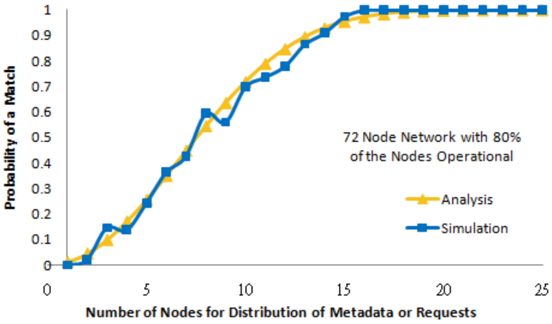


Figure 8. Match probability vs. number of nodes for distribution of metadata and requests in a network with 72 nodes where 80% of the nodes are operational.

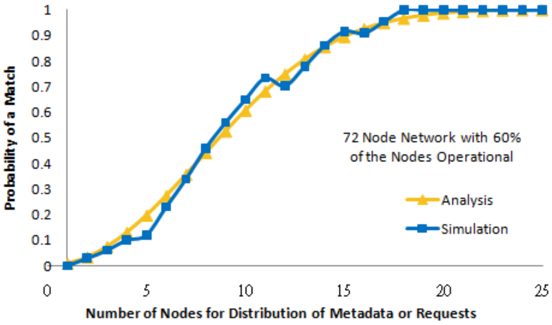


Figure 9. Match probability vs. number of nodes for distribution of metadata and requests in a network with 72 nodes where 60% of the nodes are operational.

the metadata keywords. Centralized search engines are efficient and scalable, but are vulnerable to manipulation by administrators. Similarly, the centralized publish/subscribe approach uses a centralized search index [5], where all of the information and all of the queries are published. The centralized publish/subscribe approach has the same issues of trust as the centralized search engine approach.

Mischke and Stiller [11] provide a taxonomy of distributed search mechanisms in peer-to-peer networks. Risson and Moors [12] provide another useful survey of distributed search mechanisms in such networks. The distributed publish/subscribe approach is categorized as either structured or unstructured.

The structured approach [1], [8], [10], [14] requires the nodes to be organized in an overlay structure, based on Distributed Hash Tables (DHTs), trees, rings, *etc.* The structured approach is more efficient than the unstructured approach, but it involves administrative control and additional overhead for constructing and maintaining the overlay network. Moreover, churn or malicious disruptions can break the structure.

The unstructured approach [4], [6], [7], [9], [15], [16], [17], [19] is typically based on gossiping, uses randomization, and requires the subscriber nodes and the publisher nodes to find each other by exchanging messages over existing links. The iTrust system uses the unstructured approach.

Gnutella [7] is the great grandfather of unstructured distributed search systems; it uses flooding of requests to find information. GIA [3] is an unstructured Gnutella-like peer-to-peer system that combines biased random walks with one-hop data replication to make search more scalable. Likewise, Sarshar *et al.* [13] combine random walk data replication with a two-phase query scheme in a Gnutella-like network for scalability. Yang and Garcia-Molina [18] use supernodes to improve efficiency, but reintroduce some of the trust risks of centralized strategies in doing so.

Freenet [4] is a more sophisticated and efficient system than Gnutella, because it learns from previous requests. In Freenet, nodes that successfully respond to requests receive more metadata and more requests. Thus, it is easy for a group of untrustworthy nodes to conspire together to gather most of the searches into their group, making Freenet vulnerable to subversion.

Ferreira *et al.* [6] use a random-walk strategy in an unstructured network to replicate both queries and data. BubbleStorm [15] is a probabilistic system for unstructured peer-to-peer search that replicates both queries and data, and combines random walks with flooding. Pub-2-Sub [16] is a publish/subscribe service for unstructured peer-to-peer networks of cooperative nodes, that uses directed routing (instead of gossiping) to distribute subscription and publication messages to the nodes. None of the above unstructured systems is particularly concerned with trust, as iTrust is.

Two other systems that, like iTrust, are concerned with trust are Quasar and OneSwarm. Quasar [17] is a probabilistic publish/subscribe system for social networks with many social groups. The authors note that “an unwarranted amount of trust is placed on these centralized systems to not reveal or take advantage of sensitive information.” iTrust does not use a structured overlay, and has a different trust objective than Quasar. OneSwarm [9] is a peer-to-peer data sharing system that allows data to be shared either publicly or anonymously, using a combination of trusted and untrusted peers. OneSwarm is part of an effort to provide an alternative to cloud computing that does not depend on centralized trust. Its initial goal is to protect the privacy of the users; iTrust does not aim to conceal the users like OneSwarm does.

VII. Conclusion and Future Work

We have described iTrust, a novel information distribution and retrieval system with no centralized mechanisms and no centralized control. iTrust involves distribution of metadata and requests, matching of requests and metadata, and retrieval of information corresponding to the metadata. We have shown that, with iTrust, the probability of matching a query is high even if some of the participating nodes are subverted or non-operational. The iTrust system is particularly valuable for individuals or citizens who wish to share information, without having to worry about subversion or censorship of information.

In the future, we plan to evaluate the ease of installation and use of iTrust with various user populations and also to evaluate its reliability and efficiency in PlanetLab. We also plan to investigate the scalability of the iTrust system to thousands of nodes and, then, to extrapolate those results to millions of nodes. In addition, we plan to investigate a range of possible attacks on iTrust and countermeasures to such attacks. Our objective for iTrust is a network in which individual nodes can detect a potential attack, and can adapt to an attack to maintain trustworthy information distribution and retrieval even when under attack.

Acknowledgment

This research was supported in part by U.S. National Science Foundation grant number NSF CNS 10-16193.

References

[1] S. Bianchi, P. Felber and M. Gradinariu, "Content-based publish/subscribe using distributed r-trees," *Proceedings of Euro-Par*, Rennes, France, August 2007, pp. 537–548.

[2] S. Brin and L. Page, "The anatomy of a large-scale hypertextual Web search engine," *Proceedings of the 7th International Conference on the World Wide Web*, Brisbane, Australia, April 1998, pp. 107–117.

[3] Y. Chawathe, S. Ratnasamy, L. Breslau, N. Lanham and S. Shenker, "Making Gnutella-like P2P systems scalable," *Proceedings of the ACM SIGCOMM Applications Technologies, Architectures and Protocols for Computer Communications Conference*, Karlsruhe, Germany, August 2003, pp. 407–418.

[4] I. Clarke, O. Sandberg, B. Wiley and T. Hong, "Freenet: A distributed anonymous information storage and retrieval system," *Proceedings of the Workshop on Design Issues in Anonymity and Unobservability*, Lecture Notes in Computer Science, Berkeley, CA, July 2000, pp. 46–66.

[5] P. T. Eugster, P. A. Felber, R. Guerraoui and A. M. Kermarrec, "The many faces of publish/subscribe," *ACM Computing Surveys* 35:2, June 2003, pp. 114–131.

[6] R. A. Ferreira, M. K. Ramanathan, A. Awan, A. Grama and S. Jagannathan, "Search with probabilistic guarantees in unstructured peer-to-peer networks," *Proceedings of the Fifth IEEE International Conference on Peer-to-Peer Computing*, Konstanz, Germany, August 2005, pp. 165–172.

[7] Gnutella, http://gnutella.wego.com/

[8] A. Gupta, O. D. Sahin, D. Agrawal and A. El Abbadi, "Meghdoot: Content-based publish/subscribe over P2P networks," *Proceedings of the 5th ACM/IFIP/USENIX International Middleware Conference*, Toronto, Canada, 2004, pp. 254–273.

[9] T. Isdal, M. Piatek, A. Krishnamurthy and T. Anderson, "Privacy preserving P2P data sharing with OneSwarm," Technical Report UW-CSE, Department of Computer Science, University of Washington, 2009.

[10] J. Li, B. Loo, J. Hellerstein, F. Kaashoek, D. Karger and R. Morris, "On the feasibility of peer-to-peer Web indexing and search," *Proceedings of the 2nd International Workshop on Peer-to-Peer Systems*, Lecture Notes in Computer Science 1735, 2003, pp. 207–215.

[11] J. Mischke and B. Stiller, "A methodology for the design of distributed search in P2P middleware," *IEEE Network* 18:1, January 2004, pp. 30–37.

[12] J. Risson and T. Moors, "Survey of research towards robust peer-to-peer networks: Search methods," Technical Report UNSW-EE-P2P-1-1, University of New South Wales, September 2007, RFC 4981, http://tools.ietf.org/html/rfc4981

[13] N. Sarshar, P. O. Boykin and V. P. Roychowdhury, "Percolation search in power law networks: Making unstructured peer-to-peer networks scalable," *Proceedings of the 4th International Conference on Peer-to-Peer Computing*, Zurich, Switzerland, August 2004, pp. 2–9.

[14] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for Internet applications," *Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures and Protocols for Computer Communications*, San Diego, CA, August 2001, pp. 149–160.

[15] W. W. Terpstra, J. Kangasharju, C. Leng and A. P. Buchman, "BubbleStorm: Resilient, probabilistic, and exhaustive peer-to-peer search," *Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures and Protocols for Computer Communications*, Kyoto, Japan, August 2007, pp. 49–60.

[16] D. A. Tran and C. Pham, "Enabling content-based publish/subscribe services in cooperative P2P networks," *Computer Networks* 52:11, August 2010, pp. 1739-1749.

[17] B. Wong and S. Guha, "Quasar: A probabilistic publish-subscribe system for social networks," *Proceedings of the 7th International Workshop on Peer-to-Peer Systems*, Tampa Bay, FL, February 2008.

[18] B. Yang and H. Garcia-Molina, "Improving search in peer-to-peer networks," *Proceedings of the 22nd IEEE International Conference on Distributed Computing Systems*, Vienna, Austria, July 2002, pp. 5–14.

[19] M. Zhong and K. Shen, "Popularity-biased random walks for peer-to-peer search under the square-root principle," Lecture Notes in Computer Science 4490, 2007, pp. 877–880.

Towards a Decentralized QoE Layer for the Mobile Internet

Martin Dobler, Jens Schumacher
Product and Process Engineering
Vorarlberg University of Applied Sciences
Dornbirn, Austria
{doma, scj}@fhv.at

Fikret Sivrikaya, Sebastian Peters
DAI Labor
TU Berlin
Berlin, Germany
{Fikret.Sivrikaya, Sebastian.Peters}@dai-labor.de

Eileen Dillon, Gemma Power
Telecommunications Software and Systems Group
Waterford Institute of Technology
Waterford, Ireland
{edillon, gpower}@tssg.org

Abstract — **Network selection on current mobile devices has to be done manually by the user and is furthermore strongly dominated by monopolistic telecom operators. A decentralized Quality of Experience (QoE) layer supported by a QoE knowledge base filled with automatically and user created QoE reports will offer a basis for user-centric and optimized network selection for users in the Future Internet. An automated handover from one networking interface to another can then be performed by a mobile or portable device automatically. This papers focuses on how a decentralized QoE layer for the mobile Internet can be achieved by describing how a QoE model is defined and QoE reports are gathered, shared and distributed. The content of this paper is based upon the results of the PERIMETER project. PERIMETER's main objective is to establish a new paradigm for user-centricity in advanced networking architectures. The PERIMETER middleware is briefly explained and testing methodologies for involving the user in the process of creating a user-centric QoE based mobile Internet are presented.**

Keywords - Quality of Experience; User-Centric; Seamless Mobility; Always Best Connected; Future Internet

I. INTRODUCTION

Telecommunication network management practices are strongly rooted in the monopolistic telecom operators. The liberalization of the operators has only changed the landscape in a way that there are multiple closed operators rather than one closed operator. As a result they are usually centrally managed, poorly integrated with outside components, and strictly isolated from external access. On the other hand the IP world has been about internet-working from its conception on. Furthermore the exposure of users to the prolific Internet services means that similar service models will have to be provided by the next generation telecom networks. The clash between these two opposite approaches poses important challenges for network operators. This is due to the fundamental risk associated with their networks turning into mere bit-pipes. In order for future telecom networks to be economically viable, they should provide similar user experience with Internet services, albeit in a more managed and reliable manner.

There lies the grand challenge of the so-called Telco 2.0 operators. The operators have to offer even more data intensive applications on their networks to make their operations profitable. This comes in a time, when the increasing data traffic is starting to hurt user experience, and pose itself as the biggest risk facing the operators [1].

Therefore, as we believe, a paradigm shift in the Future Internet is needed. Away from centralised, closed and single contract model towards an IP world where the user is consuming the services based upon their needs in the multiple-access multiple-operator networks of the Future Internet.

The approach presented in this paper is based upon the findings of the PERIMETER research project [2] which aims at establishing new paradigms for user-centricity in networking architectures. PERIMETER uses Quality of Experience (QoE) models to ensure user-centric optimal network usage. The finding of a user-centric QoE model is described in Section II. Section III focuses on the PERIMETER middleware and Section IV on how gathered QoE knowledge is spread in a mobile network. Section V describes the actual process on how QoE information is gathered and computed. The paper concludes with the testing methodologies as well as a summary and the future work ahead for the PERIMETER project.

II. DEFINING A USER-CENTRIC QOE MODEL FOR NETWORK USAGE – THE PERIMETER APPROACH

Nearly all portable and mobile devices nowadays contain a variety of network interfaces, among these GPRS, UMTS, WiMAX or WiFi. Additionally, today's devices allow the user to have a larger set of different contracts with different operators, e.g. with the use of multi-SIM-card-devices or by accessing WiFi hotspots. However, changing a connection from one networking interface to another is still the responsibility of the user. The PERIMETER approach targets at shifting this responsibility from the user to the device itself. Therefore PERIMETER is targeting a paradigm shift in the Future Internet where the user is in an Always Best Connected (ABC) state, where ABC is defined by the user's preferences and his or her environment. Additionally, if an unknown connection is encountered, e.g. a WiFi hotspot, the new connection is evaluated based upon the information and data gathered by other PERIMETER users – besides the physical data of the connection quality.

To define a QoE model of all connections available to the user, we need to identify parameters and user preferences which should be considered in the model. To identify these we use a scenario driven approach. In this approach real life scenarios are developed and the relevant elements are transformed into preferences and data input for the QoE model. The scenarios offer us sets of preferences and data input in the following categories:

- Connection cost
- Connection quality
- Security / Privacy
- Battery life

An excerpt of a common scenario is given below:

"Also, Linda, the lawyer that was conducting the transaction was concerned. It was not the cost of the call, but her privacy. She felt uneasy about how much private information could be disclosed from someone just knowing the existence of such a call. She really wanted no one to be able to trace her and to learn that she was not in her office but in another country where a huge deal was expected to finalise. And, even worse, if just one would know that she was talking with Helen at that time of the day…. She was aware that her calls leak location and identity, and could impart other information also. Her only hope was that nobody was keeping track of this call."

The scenarios offer also a possibility to test the PERIMETER approach in a Living Lab [3] environment (see Section VI).

III. THE PERIMETER MIDDLEWARE

The PERIMETER middleware architecture is based on the traditional layered architecture approach. There are two types of PERIMETER hardware nodes, the PERIMETER Terminal which is a mobile handheld device with certain resource restrictions, e.g. storage space, and a Support Node which has no resource restrictions, such as a server or laptop.

The architecture depicted in Figure 1 permits users to experience seamless connectivity while on the move. The PERIMETER components include:

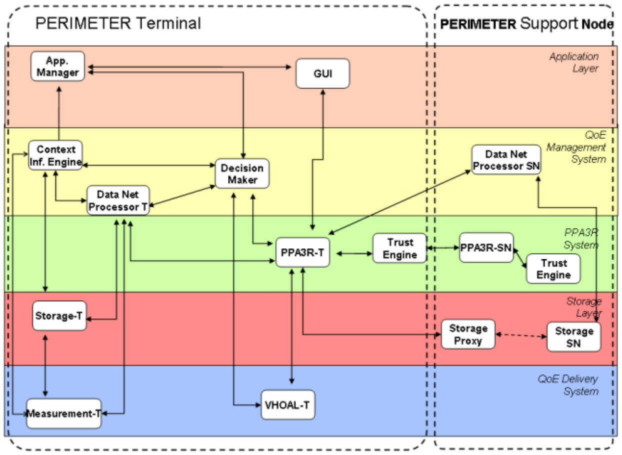


Figure 1. PERIMETER Middleware Architecture

- The *Application Layer* consisting of the Graphical User Interface (GUI) and Application Manager which provides the user with an intuitive interface to the entire PERIMETER system.
- The *Context Inference Engine (CIE)* which collects raw source data, such as geographical location and network information, and infers high level context information from this.
- The *Data Network Processor (DNP)* processes information relevant for making a decision about how satisfactory the current connection is for the user based on their context (from the CIE) and other contributing factors.
- The *Decision Maker* component decides whether a network switch is required based on information from the DNP and CIE. It also decides which network should be connected to.
- The *Privacy Preserving Authentication, Authorization, Accounting and Reputation (PPA3R)* module provides identity management, anonymisation and pseudonimization.
- The *Trust Engine (TE)* performs computations on data processed in the PERIMETER system, assigning trust and reputation values as appropriate.
- The *Vertical Handover Abstraction Layer (VHOAL)* and *Measurements* modules are charged with the task of seamless switching of networks.
- The Storage Layer takes care of storing and retrieving local and historical information using a peer-to-peer approach.

The interaction of these components provides a comprehensive architecture upon which the premise of the PERIMETER paradigm is built.

IV. A DISTRIBUTED QOE KNOWLEDGE BASE FOR MOBILE INTERNET

QoE reflects the collective effect of service performances that determine the degree of satisfaction of the end-user, e.g. what user really perceives in terms of usability, accessibility, retainability and integrity of the service [4]. Until recently, seamless communications has been mostly based on technical network Quality of Service (QoS) parameters, but a true end-user view of QoS is needed to link between QoS and QoE. While existing 3GPP or IETF specifications describe procedures for QoS negotiation, signaling and resource reservation for multimedia applications, such as audio/video communication and multimedia messaging, support for more advanced services involving interactive applications with diverse and interdependent media components is not specifically addressed. Such innovative applications, likely to be offered by 3rd party application providers and not the operators, include collaborative virtual environments, smart home applications and networked games. Additionally, although the QoS parameters required by multimedia applications are well known, there is no standard QoS specification enabling to deploy the underlying mechanisms in accordance with the application QoS needs.

For the Future Internet to succeed and to gain wide acceptance of innovative applications and service, not only QoS objectives but also QoE have to be met. Perceived quality problems might lead to acceptance problems, especially if money is involved. For this reason, the subjective quality perceived by the user has to be linked to the objective, measurable quality, which is expressed in application and network performance parameters resulting in QoE. Feedback between these entities is a prerequisite for covering the user's perception of quality [5].

The PERIMETER project investigates a user-centric networking paradigm for future telecommunication networks, where users not only make network selection decisions based on their local QoE evaluation but also share their QoE evaluations among each other for increased efficiency and accuracy in network selection, as depicted in Figure 2. In this paper we present the conceptual framework introduced by PERIMETER to achieve such user-centric network architecture for sharing and exploiting user quality of experience data. The focus here is on the utilization of a distributed *knowledge base (KB)* of QoE reports for improving network access selection decisions, while the actual implementation of the KB is out of the scope of this paper. The reader is referred to technical reports and public deliverables of the PERIMETER project for further details [2].

In order to make user-centric decisions and share user experiences based on the QoE, a software entity must first evaluate and quantify QoE for a given set of inputs including the network interface and the application running on the user terminal. Named as the Data Network Processor (DNP) in PERIMETER, this entity is responsible for calculating, from network performance measurements, user's context information and user's feedback, a QoE descriptor (QoED).

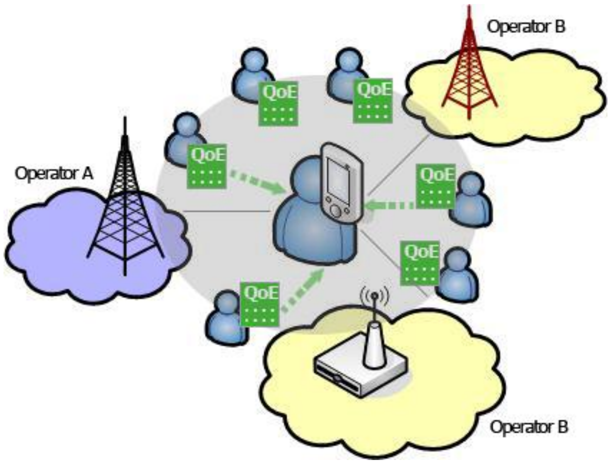


Figure 2. Future user-centric networking paradigm based on a QoE framework.

Each QoED item is an aggregate and synthetic description of the quality of the user's experience. It consists of a set of key parameters that summarize the quality of service from a user's point of view:

- Mean Opinion Score (MOS) for different types of applications
- Cost rating
- Security rating
- Energy saving issues

Once the QoED is calculated, it is uploaded onto a distributed *knowledge base (KB)*, which is a peer to peer storage module running on user terminals and on the so called support nodes specifically deployed by the operators with the incentive of obtaining user QoE reports more efficiently. The distributed knowledge base of QoE reports can then be probed with a QoED query (QoEDq) in order to obtain past QoE reports of other users for decision making, as will be described later in more detail. A QoEDq consists of a set of optional parameters that are used to filter network performance and user's context information stored both locally and globally. These filters apply to:

- Network connection, to get performance information and QoED items associated to it
- Application information, to get QoED items calculated for applications of the same class
- Geographical location, to get QoED items calculated at the same area
- User's id, to get QoED items calculated by a certain user

A QoEDq item may contain all or just a reduced set of parameters, allowing a wide variety of queries: QoEDs associated to a certain provider or a certain technology, etc. The calculated QoED items are mainly utilized by the

Decision Maker (DM), which will be described in the following section.

The DNP may generate QoED reports in two different ways: (i) Subscription based reports, where a certain component, which acts as a client from the DNP's point of view, subscribes to the reception of QoED reports according to a specific QoEDq. (ii) Unsolicited reports, where the DNP takes the initiative and sends a QoED report to all the components that offer a receiving interface for this type of events. The unsolicited reports are triggered by events that are related to an imminent handover action due to a significant change of network conditions, for example, signal loss. In this case, the QoED specifies the network that triggered the event and the actual user's context description (location, application under use, etc.).

V. User-Centric Decision Making for Improved QoE

The knowledge gathered by the DNP through local and remote QoE reports, user context information, and user preferences are all fed into a controller entity on the mobile device, named as the Decision Maker (DM) in PERIMETER. This is the entity that makes use of all those QoE related inputs to take allocation decisions for all the applications running on the terminal. The decisions that the DM is responsible for taking are what we call *allocation decisions*, where different applications running on the terminal are allocated to different access networks operated by different network providers. From this perspective the atomic decision is the movement of an application from a certain Point of Attachment (PoA) to another. This decision is made based on local and remote QoE reports, abstracting the network and subjective user satisfaction, context reports, and user preferences. The main purposes of the DM can be listed as follows:

- Take allocation decisions on which operator will be chosen for the applications
- Utilize local and remote QoE reports for the decisions
- Utilize context reports for the decisions
- Utilize user preferences for the decisions
- Infer the failure mode that has led to degradation in the QoE

The novel PERIMETER approach, in which users share their experiences, allows novel decision algorithms to be developed. Within this scope, the DM differentiates itself from the state of the art decision mechanism in the following aspects:

- *Failure Mode Inference*: The DM is able to discern the cause of the problem that has led to the degradation in QoE. The degradation can be due to a problem at the application service provider side, core network side, access network side, or at the air interface, as depicted in Figure 3. This novelty has two advantages. First of all, it minimizes the number of allocations that require handovers, which puts burden on network components, and degrades the QoE even more for their durations. Secondly, the users are not concerned with the actual cause of degradation in the QoE. They have a holistic view of the application and the service agreement. If an application is not running on an operator network properly, they will most likely blame the network operator, and give a bad MOS input. Thus there is an incentive for the operators to select decision mechanisms that are able to discern the causes of the connection problems. This information can also be used for network optimization purposes.

- *Reasoning*: The fact that users will be exchanging information about subjective measures on their applications requires a common understanding and agreement on the concepts that make up these subjective measures. This necessitates a semantically enhanced representation of the stored information. Reasoning algorithms will be used for performing Failure Mode Inference and taking the appropriate decisions based on the inferred failure mode.

- *Distributed Probing*: Thanks to the PERIMETER middleware, a distributed database of network performance data as experienced from different locations is available. This allows a practical implementation of the distributed probing of the network. This approach is used for Failure Mode Inference at the first stage, but it will be investigated for further utilization purposes that may benefit the network operators as well.

The DM requests sets of remote QoE reports, which are delivered in form of statistical distributions, a mathematical representation of the QoE reports. Within the Failure Mode Inference these distributions are fed into a Bayesian Network, which outputs the probability that a specific failure in some part of the network occurred. The comparison of user generated QoE reports is based on the assumption that users connected via the same Access Point share the same or at least parts of the route to a certain service and thus experience similar problems accessing their service or using a specific application.

In order to deduce which part of the network is affected by impairments (e.g. congestion), those QoE reports are requested from the distributed Storage that complement the view on the network. Following our assumption this includes remote user QoE reports of the currently used PoA and service. These two sets of QoE reports correspond to random distributions, one distribution of users' experience of our PoA and one of users of other PoAs using the same service or application.

Based on these two distributions from randomly selected users the most likely source of an impaired local QoE should be inferred. The events within each distribution are further categorized into two quality states that reflect users MOS, namely good and bad. These quality states can now serve as an input to the Bayesian Inference mechanism that is capable of performing inference in the face of incomplete knowledge (randomly chosen subset of users) and uncertainty (unconsidered causes for the QoE degradation). The outcome of the reasoning process in the Failure Mode Inference (FMI) component is either that a failure in a specific part of the network is most probable (in the Access Network (AN) or in the Service Domain (SD)) or the cause for impairments might remain unsolved.

While the FMI is used for the current allocation only, another inference mechanism called Prospective Network Analysis (PNA) makes use of remote QoE reports to determine the QoE that can be expected for a specific prospective PoA. Again remote QoE reports are requested, this time for the prospective PoA and the used service. The received distribution is then used to calculate the mean MOS for the prospective PoA. If the outcome of the PNA suggests that the prospective network has most probably no positive effect on the QoE the application will remain at its current PoA.

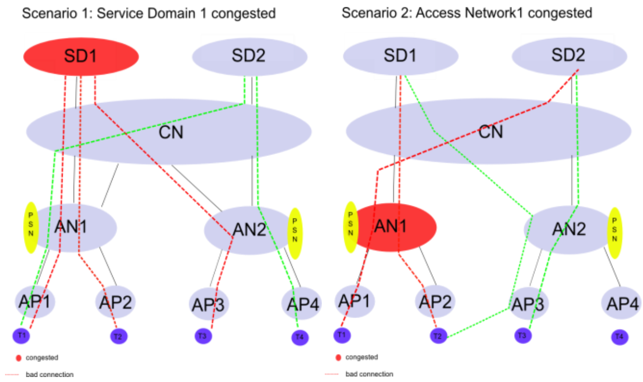

Scenario 1: Service Domain 1 congested
Scenario 2: Access Network1 congested
SD1
SD2
CN
AN1
AN2
PSN
AP1
AP2
AP3
AP4
T1
T2
T3
T4
congested
bad connection

Figure 3. Different modes of failure in a multi-operator, multi-access-technology environment.

VI. Testing the Approach

The proposal of a distributed knowledge base and user-centric allocation decisions based on this KB in PERIMETER requires an experimentally driven research approach in order to (i) fine-tune and optimize the decision parameters in the proposed framework, and (ii) ensure healthy progress of software development through the implementation-testing cycle. On these grounds, PERIMETER makes a distinction between the definition of *testing* and that of *experimentation*, and each is handled differently within the project.

A. PERIMETER Testing Process

In PERIMETER, testing is considered to be the continuous process needed in software development. The Agile software development methodology [6] was chosen for use within the project as it was considered to be the most adaptable, fluid and iterative methodology from those evaluated. Coupled with this, it has been successfully applied to large scale projects previously (for example [7] and [8]) and has been proven to lead to higher quality software when compared to using traditional methods [6]. In PERIMETER, the use of Agile is combined with a Test Driven Development (TDD) [9] approach to provide the adaptable, fluid and iterative development and testing cycle that is required [10].

The need for processes, supporting tools and automation is necessary for the success of most software projects, but is essential for the success of large scale experimental ventures. PERIMETER employs a number of processes and tools [11] to aid the processes of team collaboration and communication, testing, continuous integration (Hudson) and structured software development.

In this stage of development and testing, i.e. in a continuous build environment, a number of testing processes including unit (white-box), functional (black-box) and integration testing, are conducted to ensure that the developed software is brought to a level where it achieves its functional objectives.

B. PERIMETER Experimental Process

In PERIMETER, experimentation is needed to guarantee the success of the research, its innovation and the usability aspects of the project. These are verified in two large scale, state of the art testbeds, one in Waterford Institute of Technology (WIT) Ireland and one in Technical University of Berlin (TUB), Germany. These testbeds are interconnected over a Layer 3 connection. This interconnection was then matured to a Layer 2 federation in the final phase of the project. PERIMETER has also applied for, and was successful in procuring a FEDERICA slice (5 virtual nodes) for use in the project [12].

Conformance tests are performed on the PERIMETER system installed and running on the federated testbed to ensure that key components of the system are functioning as expected. These tests are complemented with interoperability tests to ensure the end-to-end functionality of the system is as expected. These processes are tested against the scenario under analysis, in the scenario-driven approach used in this project in order to demonstrate that the testing process is robust and to ensure the verifiability and reliability of the results. It is essential that both testing processes were repeatable and reproducible in order to achieve this. Application testing for the specific applications running on the federated testbeds is also performed.

This process is further complemented by the employment of a user-driven approach to the requirements specification and the determination of features and subsequent testing phases with the use of Living Labs [3] and dedicated usability sessions. Performance and scalability issues are addressed with the introduction of emulation and simulation of network conditions.

C. *Role of Testbeds in PERIMETER*

The distinction between testing and experimentation in PERIMETER allows the role of the federated testbed within this process to be further examined. The build environment which is used for the testing process allows a certain level of testing to be achieved. However, it is within the federated testbed environment where a greater level of realism can be determined with the testing and experiments conducted. The use of the testbeds allows the system to be tested from beginning to end in a realistic environment using real platforms, applications, devices and users. Without the federated testbed, the cost of achieving this level of realism would be greatly increased.

Testbeds can be successfully used to control the cost of achieving realism in experimental activities [13]. In PERIMETER, it was found that not only does the use of the federated testbed control the cost; it actually does this without actually increasing cost but whilst increasing the level of realism achieved.

VII. Summary and Outlook

The increasingly dynamic nature of the telecommunications scene is expected to go beyond the technical domain and also cover business models and socioeconomic aspects of telecommunications, eventually giving rise to the user-centric network vision foreseen by the PERIMETER project. There are many challenges, both technical and socioeconomic, that need to be addressed for this vision to come true, such as the need for a standardized view of QoE among all stakeholders that should act as a common performance and valuation criterion. In this paper we have focused on the exploitation of an open QoE knowledge base for more intelligent and user-centric interface selection, which can also provide benefits network operators in terms of resource utilization.

Currently the development PERIMETER system is in its final stage and Living Lab tests with real users are being conducted in two inter-connected testbeds, one in Waterford, Ireland and one Berlin, Germany. The tests will provide experimental results on user acceptance and about PERIMETER's usability in a real life environment.

Acknowledgment

This research activity is funded under the EU ICT FP7 project PERIMETER (Project No. 224024).

References

[1] Marvedis, R., "Mobile Operators Threatened More by Capacity Shortfalls than Growth of WiMAX," Sep 30, 2009. [Online]. Available online: http://maravedis-bwa.com/Issues/5.6/Syputa_readmore.html. [Last accessed: 16.02.2011].

[2] PERIMETER Website, Technical Reports and Delieverables. [Online] Available online: http://www.ict-perimeter.eu/ [Last accessed: 21.02.2011].

[3] M. Eriksson, V. Niitamo, and S. Kulki, "State-of-the-art in utilizing Living Labs approach to user-centric ICT innovation – a European approach", Centre for Distance-spanning Technology at Luleå University of Technology, 2005.

[4] DSL Forum Technical Report TR-126, Triple-play Services Quality of Experience (QoE) Requirements, December 2006, Produced by Architecture & Transport Working Group.

[5] E. Dillon, G. Power, M. O. Ramos, M. A. Callejo Rodríguez, J. R. Argente, M. Fiedler, and D. S. Tonesi, PERIMETER: A Quality of Experience Framework, Future Internet Symposium (FIS) 2009, Berlin (Germany), Sep. 1-3 2009.

[6] V. Subramaniam, and A. Hunt, "Practices of an Agile Developer", ISBN: 9780974514086, April 2006

[7] IBM promotes agile development. [Online] Available Online: http://www.infoworld.com/article/08/03/04/IBM-promotes-agile-development_1.html [Last accessed: 21.02.2011].

[8] J. Sutherland, A.Viktorov, J. Blount, and N. Puntikov, "Distributed Scrum: Agile Project Management with Outsourced Development Teams", Proc. Of 40th Annual Hawaii International Conference on System Sciences (HICSS'40), Hawaii: IEEE, 2007.

[9] K. Beck, "Test-Driven Development by Example", Addison Wesley, ISBN: 978-0321146533, 2003

[10] E. Dillon, F. Sivrikaya, C. Hammerle, and L. Salgarelli, "Agile Principles Applied to a Complex Long Term Research Activity -- The PERIMETER approach", Proc. 35th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA), Patras Greece, August 27-29, 2009.

[11] PERIMETER Toolset: Subversion, [Online] Available Online: http://subversion.tigris.org; Trac, Available: http://trac.edgewall.org; JUnit Testing Framework, Available:: http://www.junit.org/; Apache Ant, Available: http://ant.apache.org/; Cobertura Code Coverage Analysis Tool, Available: http://cobertura.sourceforge.net/; Hudson Continuous Integration Engine, Available: https://hudson.dev.java.net/ and Eclipse IDE, Available: http://www.eclipse.org/, [Last accessed: 21.02.2011].

[12] FEDERICA - Federated E-infrastructure Dedicated to European Researchers Innovating in Computing network Architectures. [Online]. Available Online: http://www.fp7-federica.eu/ [Last accessed: 21.02.2011].

[13] D. Papadimitriou, "Experimentation as a research methodology to achieve concrete results: where, how, when", FIA Stockholm, 23-24 November 2009, [Online] Available Online: http://www.future-internet.eu/ [Last accessed: 21.02.2011].

TCP Congestion Avoidance using Proportional plus Derivative Control

Dirceu Cavendish, Hikaru Kuwahara, Kazumi Kumazoe, Masato Tsuru, Yuji Oie
Department of Computer Science and Electronics
Kyushu Institute of Technology
Fukuoka, Japan 810-0004
Email: {cavendish,kuma,tsuru,oie}@ndrc.kyutech.ac.jp, kuwahara@infonet.cse.kyutech.ac.jp

***Abstract*—We introduce a transmission control protocol with a delay based congestion avoidance that utilizes a proportional plus derivative controller. The derivative part of the controller is shown to be well suited to effectively control TCP sessions, given the relative shallow buffer space of network elements. We demonstrate the competitive performance of the protocol via open source based network experiments over a research network and the Internet.**

***Keywords*—high speed networks; TCP congestion avoidance; Packet retransmissions; capacity estimation; path bottleneck; Proportional plus derivative controller.**

I. Introduction

Recent advances in TCP protocols have departed from window transmission regulation based on binary information into multi-bit information [2], [9]. Rich feedback information indeed holds the promise of better regulating packet transmission by reducing traffic oscillations typical of binary based control mechanisms. Moreover, recent advances in TCP flow probing have made available path estimators that may be useful for regulating TCP traffic transmission [10].

In our prior work, we have introduced a delay based TCP window flow control mechanism that uses path capacity and storage estimation, called Capacity and Congestion Probing - CCP [5]. The idea is to estimate bottleneck capacity and path storage space, and regulate the congestion window size using a proportional controller. We have shown that CCP has competitive performance as compared with widely known TCP protocols, such as Reno and Cubic, outperforming them in the presence of random packet loss scenarios such as in wireless bottlenecks.

However, quite often path storage space is limited, due to shallowness of network elements' buffers. In such cases, a large proportional gain parameter is needed to increase throughput performance, which increases the protocol aggressiveness. Motivated by the shallowness of buffers in the Internet network elements, in this paper we propose to add a derivative component to the proportional controller used to regulate TCP congestion window. The idea is to react to the derivative of the available path storage space, in addition to the absolute storage space value. When buffers are limited, a derivative component should help regulate better traffic input into the TCP session.

In this work, our contributions are as follows. We show the feasibility of a window flow control mechanism based on a proportional plus derivative controller, based on non intrusive path capacity and buffer storage estimators. As previously, we use a control theoretic framework that ensures stability regardless of session characteristics (path capacity and round trip time) and cross traffic activity; we design a congestion window regulation scheme within the framework of a TCP protocol, called TCP-Capacity and Congestion Proportional plus Derivative - CCPD; we demonstrate TCP-CCPD performance via a comprehensive set of open source based transpacific network experiments. The material is organized as follows. Related work discussion is provided on Section II. Section III introduces the modeling and control theoretic approach of the window regulation scheme, whereas section IV reviews the path estimators used to implement the window regulation. Section V describes the TCP-CCPD protocol, and section VI addresses its performance evaluation. Section VII addresses directions we are pursuing as follow up to this work.

II. Related Work

TCP protocols fall into two categories, delay and loss based. Advanced loss based TCP protocols, such as HS-TCP and Scalable TCP use packet loss as primary congestion indication signal, performing window regulation as $w_k = f(w_{k-1})$, being ack reception paced. Most f functions follow an Additive Increase Multiplicative Decrease strategy, with various increase and decrease parameters. TCP NewReno and Cubic are examples of AIMD strategies. Delay based TCP protocols, on the other hand, use queue delay information as the congestion indication signal, increasing/decreasing the window if the delay is small/large, respectively. Examples of these are TCP-Vegas and FAST TCP. CCP and CCPD fall into the second category, delay based protocols.

Although TCP-Vegas relies on estimates of round trip propagation delay, its window adjustment function is of the form $w_k = f(w_{k-1})$, paced by one rtt per adjustment [2]. TCP-Vegas aims at keeping a small number of packets buffered in the routers along the path. Fast TCP, on the other hand, tracks both minimum and average rtt values of a session, in order to update the window, still via a $w_k = f(w_{k-1})$ function. Although both of these algorithms can be tuned to convergence [6], parameter tuning depends on particular characteristics of each session, such as propagation delays and link speeds [7]. In contrast, CCP and CCPD both rely on a technique for dead-time delay systems to ensure stability regardless the characteristics of a TCP session [12]. This allows us to fine tune the algorithm's parameters to trade throughput for packet loss, without the danger of driving the system to instability. In addition, CCPD derivative component allows a faster reaction to transients when buffer space is limited. Unique to CCP and CCPD window control is that they do not follow a $w_k = f(w_{k-1})$ control law, which, together with built in stability mechanism, allows the protocols to be responsive to cross traffic disturbances at widely different network scenarios and traffic conditions.

III. Proportional plus Derivative Window Congestion Control

We make use of the control theoretic approach of [3] to design CCPD protocol. In what follows, we limit ourselves to summarize the results needed for CCPD protocol design, augmenting them with a derivative component.

A. Network and Queue Models

The network consists of $N = \{1, 2, \ldots, n\}$ nodes and $L = \{1, 2, \ldots, l\}$ links. Each link i is characterized by: transmission capacity $c_i = 1/t_i$ (segments/sec); propagation delay td_i. The network traffic is generated by source/destination pairs (S, D), where $S, D \in N$. To each (S, D) session, there is a number of TCP sessions associated, each of which having a fixed path $p(S, D)$ over which all segments of a given session travel. Each source is characterized by its maximum transmission speed, $c_s = 1/t_s$, dictated by its network interface card.

We further assume that each switch maintains a single queue for all sessions exiting a given outgoing link. Let $x_{i,j}(t)$ be the occupancy at time t of the queue associated with link i and $session_j$, and $B_{i,j}$ a corresponding buffer size.

For the model of the dynamic behavior of each queue, we assume a deterministic fluid model approximation of segment flow [11]. Considering the queue associated with the TCP session TCP_j at link i, if the level of queue occupancy at time t is $x_{i,j}(t)$, its input rate $u_{i,j}(t)$, and cross traffic $d_{i,j}(t)$, a fluid model of the queue system is given by:

$$\frac{dx_{i,j}(t)}{dt} = \begin{cases} u_{i,j}(t) + d_{i,j}(t) & \text{if } x_{i,j} > 0 \\ \max(0, u_{i,j}(t) + d_{i,j}(t)) & \text{if } x_{i,j} = 0 \end{cases} \quad (1)$$

B. Window Control Model

In order to control the queue level $x(t)$ for a specific session, we use a proportional plus derivative controller. Letting B be the size of the bottleneck buffer, we compute the difference between the buffer size and the current queue level $x(t)$. This difference, the error $e(t)$, is multiplied by a positive gain K_p, so that $K_p e(t)$ is the regulated input rate of the TCP source. In addition, a difference between the current queue level $x(t)$ and the previously received queue level $x(t - t_p)$ is multiplied by a positive parameter K_d and added to the input rate.

Figure 1 depicts the block diagram of the continuous time model of a TCP session. T_{ff} denotes the propagation delay from the flow controlled source to the bottleneck queue, the most congested intermediate node, whereas T_{fb} is the propagation delay incurred by traffic carrying feedback information from the intermediate node to the destination node, plus the delay incurred from the destination node back to the source node. Therefore, $RTT = T_{ff} + T_{fb}$ is the round trip propagation delay incurred by a segment carrying feedback information. In Figure 1, a generic proportional controller $Kp^*(t)$ is depicted, rather than a simple proportional controller Kp. In addition, a derivative component $Kd^*(t)$ is also depicted, acting on variations of the input signal. The cross traffic rate is $d(t)$, hence variations in $d(t)$ represent cross traffic variations, or disturbances, accounting for cross traffic impact on a TCP session, whereas B is the bottleneck queue size.

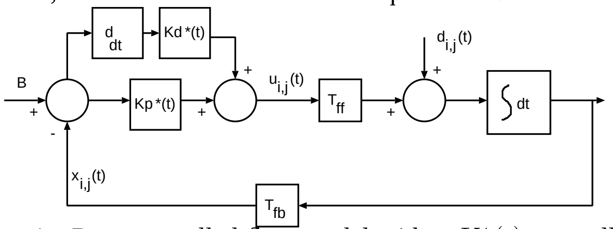


Fig. 1: Rate controlled flow model with a $K^*(t)$ controller

We use a Smith Predictor [12] to ensure stability in large bandwidth delay product paths regardless of the proportional and derivative gain parameters used. Therefore, we substitute the controllers $Kp^*(t)$ and $Kd^*(t)$ in Fig. 1 with controllers such that the resultant system has a delay free feedback loop in cascade with pure delays [3]. The proportional plus derivative controller plus the feedback loop predictor gives rise to the following input rate control equation:

$$\begin{aligned} u(t) = Kd[B - x(t) - in\_flight\_traffic(t)] + \quad (2) \\ Kd\frac{d}{dt}[B - x(t) - in\_flight\_traffic(t)] \end{aligned}$$

Eq. 2 implements a proportional plus derivative control action with the difference that the actual queue level is increased by the amount of data transmitted during the last round trip delay ($in\_flight\_traffic(t)$). Finally, a discretization of the above continuous system yields the window adjustment equation as:

$$\begin{aligned} w_k = Kp[B - x_k - in\_flight\_segs_k] + \quad (3) \\ \frac{Kd}{t_k - t_{k-1}}[x_{k-1} + in\_flight\_segs_{k-1} + \\ -x_k - in\_flight\_segs_k] \end{aligned}$$

where x_k the buffer level at discrete time k, and $in\_flight\_segs_k$ the number of segments transmitted in the last round trip delay.

Although our main interest is in TCP sender window regulation, Eq. 3 can also be used to account for retransmissions of lost segments. This is because the window regulation scheme itself does not distinguish between fresh and retransmitted segments, as long as they are both accounted for by in_flight_segs. Notice that Eq. 3 is not a pure recurrent equation, of the form $w_k = f(w_{k-1})$. Therefore, a current window size is not solely dependent on the value of the previous window size, as in most TCP protocols. [1] In addition, theoretically Eq. 3 would need to be recomputed at a given minimum frequency,

[1] To be precise, w_k depends on the values of all w_i within a full round trip time, due to in_flight_segs term.

which would require timers associated with transport sockets that require locks for safe access. Instead, a TCP congestion avoidance implementation of Eq. 3 does not require timers because, as soon as a control window worth of packets is transmitted, no new packets are allowed into the network until an acknowledgement is received. At that time, we can recompute Eq. 3 appropriately.

We use Kp parameter for throughput regulation, whereas Kd is used to quickly respond to variations in path buffer space. That is, although large values of Kp increase session throughput, segment loss may be experienced, depending on cross traffic behavior. Segment losses can be mitigated by responding quickly to queue build ups via Kd parameter of the derivative component of the controller.

Summarizing, window regulation through Eq. 3 allows for a trade off between segment loss and throughput via parameter Kp. Hence, tuning of Kp can be exercised for traffic engineering purposes. For paths through network elements with shallow buffers, Kd is tuned so as to provide quick response to variations in available space. To end this section, we mention that the control window prescribed by Eq. 3 allows the TCP sender to send a certain number of segments at line speed, if wished, without impairing controllability or stability of the session. This is indeed the typical behavior of a TCP session. Moreover, each TCP session sees its own buffer size B and a buffer level caused by its own traffic plus cross traffic on its path, caused by other TCP sessions and UDP traffic crossing its path.

IV. Path estimators

Eq. 3 requires estimators for bottleneck buffer size $\hat{B}$ and buffer level $\hat{x}$. As the estimators used in this paper are a more accurate version of the estimators used in [5], in this section we simply summarize the description of the estimators, for completeness.

A. Capacity estimation

The capacity estimation method of our choice is based on packet pair dispersion [10] techniques. The idea is to measure dispersion of the delay of packet pairs sent back to back . If both probing packets of size MSS of a packet pair sample do not suffer any queueing delay, and the dispersion between them is d, the slowest link capacity can be estimated as:

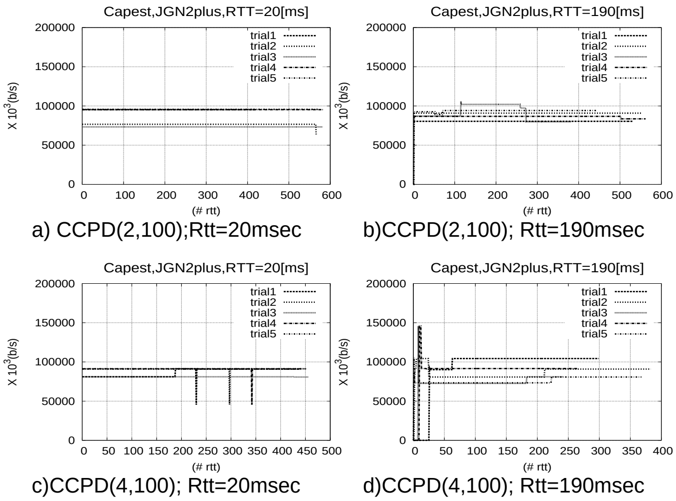
$$\hat{C} = \frac{MSS}{d} \tag{4}$$

The capacity estimation method is described in detail in [4]. In this paper, we implemented a version of this method with high resolution clocks, which allows us more precision, as well as the bottleneck capacity estimation of a wider range of speeds. Fig. 2 reports capacity estimation results for short and long rtt path scenarios, using two K_p parameter values. We can see that capacity estimation accuracy does not depend on the rtt nor the set of parameters used.

B. Buffer size estimation

Let rtt_{max} and rtt_{min} be the maximum and minimum rtts experienced by segments of a given session. A reasonable estimator for the bottleneck buffer size would then be:

$$\hat{B} = \hat{C} * (rtt_{max} - rtt_{min}) \tag{5}$$

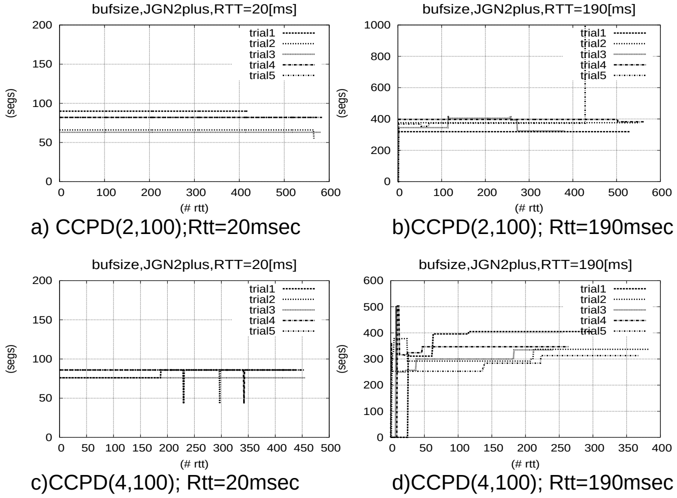


a) CCPD(2,100);Rtt=20msec b)CCPD(2,100); Rtt=190msec

c)CCPD(4,100); Rtt=20msec d)CCPD(4,100); Rtt=190msec

Fig. 2: Capacity estimation

However, a precise estimation would be achieved only when the bottleneck buffer is full, so that rtt_{max} is the rtt of the segment once stored at the last buffer slot of the buffer, otherwise the estimator will underestimate the buffer size. On the other hand, rtt_{min} may represent more than pure propagation delays, if during the estimation period the bottleneck buffer never empties. In this case, however, one may argue that the extra buffer space, taken by a persistent traffic, is never available anyways, so this extra space is perceived by a TCP session as an additional propagation delay. Fig. 3 report buffer size estimation results for short and long rtt path scenarios, using two K_p parameter values. Buffer size estimation accuracy does not depend on the parameter values used. However, long rtt sessions result in larger buffer size estimation, as expected.



a) CCPD(2,100);Rtt=20msec b)CCPD(2,100); Rtt=190msec

c)CCPD(4,100); Rtt=20msec d)CCPD(4,100); Rtt=190msec

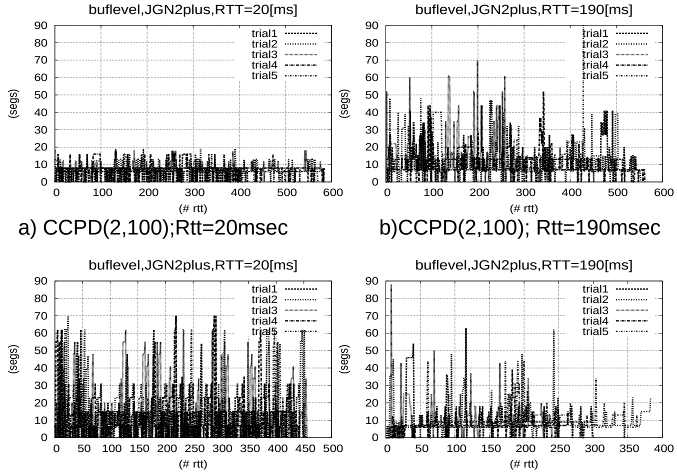
Fig. 3: Buffer Size Estimation

C. Buffer level estimation

If one tracks each segment rtt, the current buffer level $x(t)$ can be estimated by $\hat{x}(t) = (rtt(t) - rtt_{min}) \times \hat{C}$. Since sample rtt values typically include high frequency variations, a smoothed average rtt value $rtt_s(t)$ is used instead, so:

$$\hat{x}(t) = \hat{C} * (rtt_s(t) - rtt_{min}) \tag{6}$$

Fig. 4 report buffer level estimation results for short and long rtt path scenarios, using two K_p parameter values. Buffer level estimation does depend on both the parameter values used, as well as the session rtt.



a) CCPD(2,100);Rtt=20msec

b)CCPD(2,100); Rtt=190msec

c)CCPD(4,100); Rtt=20msec

d)CCPD(4,100); Rtt=190msec

Fig. 4: Buffer Level Estimation

V. TCP-CCPD PROTOCOL

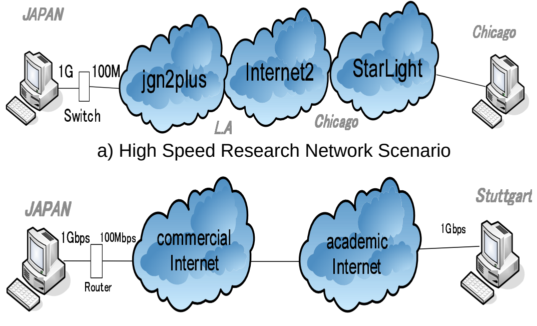
Our design follows the TCP framework: slow start, congestion avoidance, fast retransmit, and fast recovery phases, with adaptations to capacity and congestion probing as follows:

- **Slow Start :** We use a plain TCP slow start mechanism so as to focus on characterizing the performance of the congestion avoidance mechanism proposed in this paper. The only difference is that the bottleneck capacity and the buffer size estimation are passively performed during slow start as well as during congestion avoidance.
- **Congestion Avoidance :** In congestion avoidance, capacity estimators are updated continuously, so that the CCPD session can track changes in the path characteristics due to cross traffic dynamics. In particular, a capacity segment sample is set at every rtt interval to avoid interference between samples, provided that the control window is increased by at least two segments via Eq. 3. High accuracy clock helps accurate computation of the derivative component of the controller.
- **Fast Retransmit and fast recovery :** Duplicate acks cause segments to be retransmitted. During retransmission, the congestion window is maintained at the same size until all segments transmitted during that window are acknowledged. Moreover, for each duplicate ack received, the congestion window is increased to allow the retransmission of the missing segment. During recovery, rtt measurements become problematic, since segments may have to be retransmitted several times, artificially increasing their rtt. Since CCPD relies on rtt measurements, the protocol does not react to dupacks, avoiding estimators' contamination with inflated rtt values.

Regarding implementation cost, since no additional packets are used, there is no bandwidth overhead incurred by CCPD. Regarding scalability, the protocol requires OS kernel timers of small enough granularity to detect time differences that scale with bottleneck capacity speed. We have upgraded our previous estimators' implementation with high accurate clocks, where nanoseconds accuracy allows us to probe path bottleneck capacity in excess of 100 Gbps.

VI. PERFORMANCE EVALUATION

We now report on a series of open source based experiments on a high speed research network as well as the Internet (5. The research network is used to analyze our protocol properties and performance in detail, as we are able to control cross traffic and path routes. The Internet scenario is used to investigate protocol feasibility on paths with realistic cross traffic. We contrast the CCPD performance with two well known TCP protocols: NewReno, with a loss based congestion avoidance; Cubic, the Linux TCP algorithm of choice; and CCP [5], our previous delay based congestion avoidance protocol.



a) High Speed Research Network Scenario

b) Internet Network Scenario

Fig. 5: TCP Protocol Evaluation: Network Scenarios

A. PD controller parameters tuning

We use the research network scenario to tune CCPD parameters. We have selected a path with medium rtt, 40msecs, between two machines in Japan, as a commonplace path. However, the control theory behind equation 3 ensures system stability regardless the path rtts and bottleneck capacity speeds. Table I reports on 100MByte file delivery completion time for various Kp and Kd parameters. We have included CCP results, for comparison. From these results, we have selected $Kp = 2$ and $Kd = 100$ as default CCPD parameters, as a tradeoff between throughput performance and variance.

	CCPD(2,100)	CCPD(2,1000)	CCPD(4,100)	CCPD(4,1000)	CCP(2)	CCP(4)
trial 1	20791.4	17872.1	19364.5	20945.8	56636.3	25303.4
trial 2	19615.9	24074.5	17046.9	16810.4	38083.9	23647.3
trial 3	19162.0	15854.5	14419.3	20013.4	59176.1	22777.5
trial 4	20054.5	16103.8	15288.4	18280.9	30264.5	35769.7
trial 5	20818.6	18916.6	26469.8	20625.3	16608.3	33574.1
avg	20088.5	18564.3	18517.8	19335.2	40153.8	28214.4

TABLE I: 100MB delivery time(msec) : rtt=40msec

B. Transport protocols' performance with no packet loss

In this experiment set, we characterize the performance of the TCP protocols when session path is clear of congestion and packet losses. Tables II and III show the completion time of a file of 100MBytes delivered over the research network for short (20msecs) and long (180msecs) path scenarios. CCP is characterized for K_p parameter, whereas CCPD is

ISBN: 978-1-61208-141-0

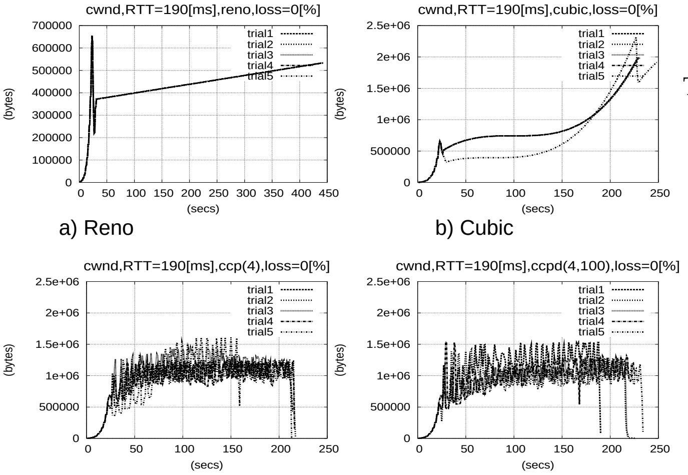
characterized for K_p and K_d parameters. In terms of transfer speed performance, Table II shows that all protocols perform similarly for the short rtt and no packet loss scenario. For long rtts (Table III), we see that Cubic, CCP(4) and CCPD(4,100) are the fastest protocols, being the most aggressive TCP congestion avoidance schemes. Hence, the least aggressive protocols (Reno and CCPD(2,100)) perform poorly in long rtt scenarios with no cross traffic. For better understanding of the dynamics of the congestion avoidance, we include a characterization of the cwnd control window for a single long rtt trial in Fig. 6. CCP(4) and CCPD(4,100) have similar cwnd dynamics over large rtts.

	Reno	Cubic	CCP(2)	CCP(4)	CCPD(2,100)	CCPD(4,100)
trial 1	1229.7	1227.3	1410.7	1226.2	1459.6	1230.8
trial 2	1221.2	1226.9	1219.3	1222.6	1225.9	1224.1
trial 3	1223.7	1455.8	1220.3	1219.7	1220.6	1224.4
trial 4	1220.0	1453.6	1458.0	1447.2	1222.6	1231.8
trial 5	1218.6	1223.2	1200.3	1222.9	1220.9	1228.3
avg	1222.64	1317.36	1301.72	1267.72	1269.92	1227.8

TABLE II: 100MB delivery time(msec): 0 PER ; rtt=20msec

	Reno	Cubic	CCP(4)	CCPD(2,100)	CCPD(4,100)
trial 1	44383.5	23114.0	21548.3	34490.6	19117.5
trial 2	44449.1	23197.8	21868.4	30846.7	22878.3
trial 3	44385.2	23134.4	19116.3	29957.6	21982.2
trial 4	44382.6	23117.9	21788.6	28430.8	22641.4
trial 5	44385.4	25043.9	21485.8	37542.3	23516.9
avg	44397.2	23521.6	21161.5	32253.6	22039.3

TABLE III: 100MB delivery time(msec): 0 PER ; rtt=190msec


cwnd,RTT=190[ms],reno,loss=0[%]

a) Reno

cwnd,RTT=190[ms],cubic,loss=0[%]

b) Cubic

cwnd,RTT=190[ms],ccp(4),loss=0[%]

c) CCP(4)

cwnd,RTT=190[ms],ccpd(4,100),loss=0[%]

d) CCPD(4,100)

Fig. 6: Cwnd(t) without random packet loss : rtt=190msec

C. Transport protocols' performance with random packet loss

In this experiment set, we characterize the performance of the TCP protocols when the session experiences random packet losses. Tables IV and V show the completion time of a file of 100MBytes delivered over the research network, when a 10^{-4} packet drop (PER) is exercised by a link emulator placed at the bottleneck link of the session, for 20msec and 180msec rtt scenarios, respectively. CCP is characterized for $K_p = 4$, whereas CCPD is characterized for $K_p = 2, 4$, and $K_d = 100$. In terms of transfer speeds, we see that all protocols completion time get severely affected by the packet loss, if compared with no loss results, except CCP and CCPD for long rtt scenario. For short rtt scenario, the protocols with most impacted file completion times are the most aggressive protocols, namely Cubic and CCP. The least affected protocol is CCPD(2,100), arguably the least aggressive protocol. Figure 7 depicts the cwnd dynamic behavior of one of the long rtt trial for all protocols. We see that for all protocols except CCP and CCPD, there is a large drop in cwnd size on every packet loss experienced. Because CCP and CCPD rely on estimators that are not related with packet loss, but rather packet delays, not affected by random losses, their performance do not get affected by random losses significantly.

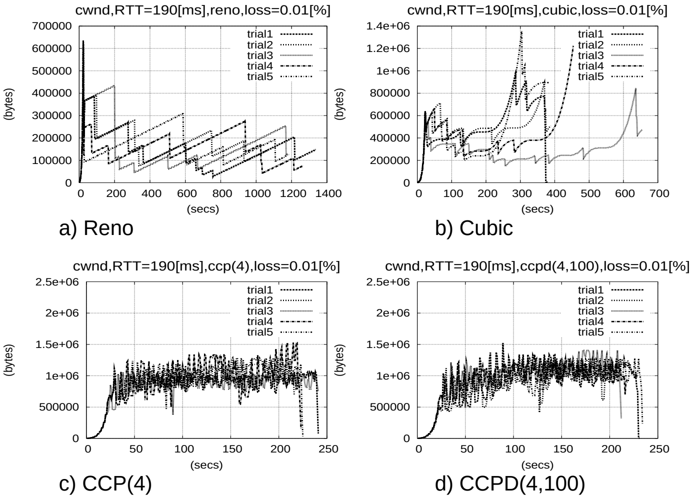
In summary, two factors significantly affect the performance of the protocols: packet loss level, and rtt size. For short rtt scenarios and no packet loss, all protocols deliver similar completion time performance. For high packet loss, long rtt scenarios require aggressive protocols for superior performance, while short rtt scenarios favor less aggressive protocols in delivering faster completion time.

	Reno	Cubic	CCP(2)	CCP(4)	CCPD(2,100)	CCPD(4,100)
trial 1	19781.9	1502.4	1849.3	2364.3	1295.4	1274.0
trial 2	17402.6	2082.8	1981.0	1728.1	1988.2	1711.2
trial 3	20899.0	3634.5	1596.4	1421.4	1623.7	1229.3
trial 4	7699.5	1726.7	2463.0	3603.7	1144.3	1519.9
trial 5	7047.1.2	1304.6	1198.5	1604.0	1508.7	1953.5
avg	14566.0	2050.2	1817.6	2144.3	1512.0	1537.6

TABLE IV: 100MB delivery t(msec): 10^{-4} PER; rtt=20msec

	Reno	Cubic	CCP(4)	CCPD(2,100)	CCPD(4,100)
trial 1	133872.7	38054.4	24262.5	33819.9	23247.3
trial 2	107217.8	39091.7	21772.8	32684.9	22374.0
trial 3	117845.4	65617.6	24245.2	30335.9	21342.4
trial 4	126682.7	45681.4	22435.5	29964.2	21837.4
trial 5	118829.2	38538.0	22616.5	31382.9	23557.9
avg	120889.6	45396.6	23066.5	31637.6	22471.8

TABLE V: 100MB delivery t(msec): 10^{-4} PER ; rtt=190msec


cwnd,RTT=190[ms],reno,loss=0.01[%]

a) Reno

cwnd,RTT=190[ms],cubic,loss=0.01[%]

b) Cubic

cwnd,RTT=190[ms],ccp(4),loss=0.01[%]

c) CCP(4)

cwnd,RTT=190[ms],ccpd(4,100),loss=0.01[%]

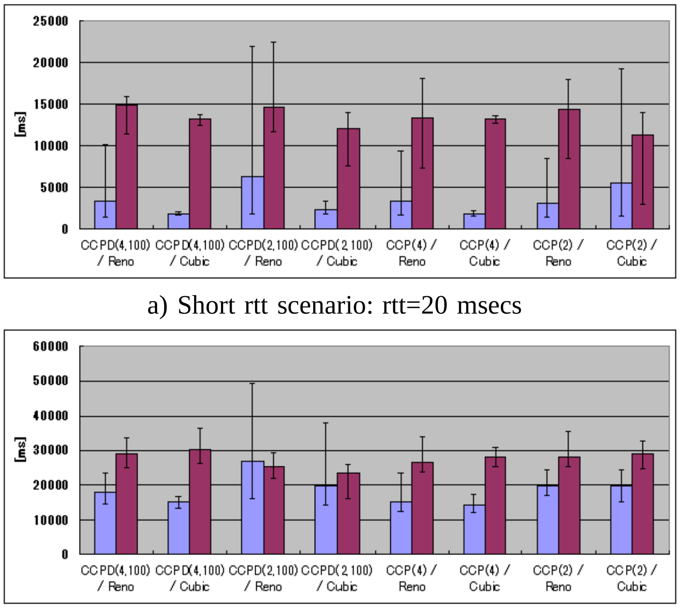
d) CCPD(4,100)

Fig. 7: Cwnd(t) with random packet loss

D. Benchmarking CCPD against other TCP protocols

In this subsection, we benchmark CCPD against Reno, Cubic, and CCP TCP protocols. Two parallel TCP sessions are initiated for the same file of 100MByte size, over the research network and Internet scenarios. We recall that the Research Network has very little cross traffic. We collect completion time performance for short and long rtt types of session. Results are shown in Figs. 8 and 9. Each pair of bars indicate average completion time over five trials for two competing TCP protocols. A range bar on top of the histogram bar indicates minimum and maximum values across the trials.

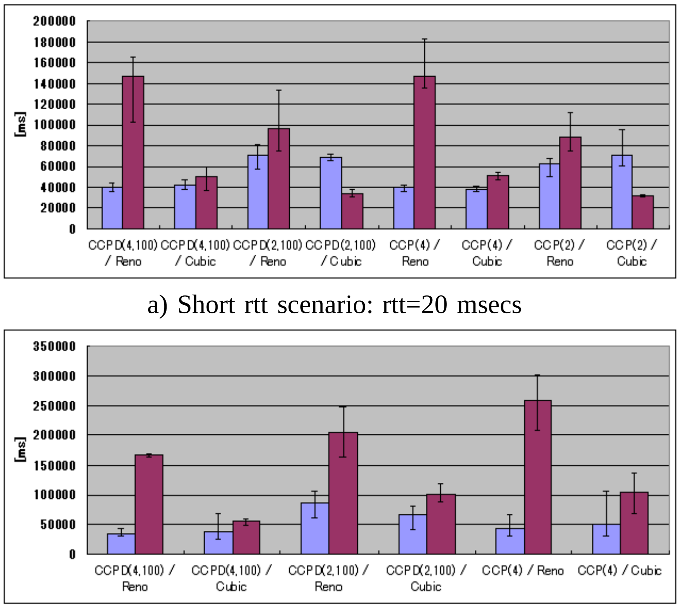
As the two sessions come simultaneously into the network, their completion time performance are comparable.



a) Short rtt scenario: rtt=20 msecs

b) Long rtt scenario: rtt=190 msecs

Fig. 8: Research Network: Completion time performance



a) Short rtt scenario: rtt=20 msecs

b) Long rtt scenario: rtt=300msecs

Fig. 9: Internet: Completion time performance

In the research network scenario, where we have practically no cross traffic, CCP and CCPD perform better than all other protocols except CCP(2) and CCPD(2,100), outperformed by Cubic on long rtt scenario. For short rtt sessions, CCPD outperforms CCP and other protocols (e.g. CCPD(2,100)/Cubic vs CCP(2)/Cubic). In the Internet scenario, CCP and CCPD outperform the other protocols on average completion time. When comparing CCPD(x,100)/otherTCP with CCP(x)/otherTCP, we see that CCPD outperforms CCP for long rtt scenario, whereas CCP outperforms CCPD for short rtt scenario. CCPD performance over the Internet needs further investigation.

VII. Future Work

In this paper, we have introduced TCP-CCPD, a transmission control protocol based on control theoretical concepts, and window regulation based on TCP session estimators. The protocol regulates traffic injection by tracking capacity and congestion along the session path during the lifetime of the session, an implementing a proportional plus derivative controller. The derivative component of the controller allows quick reaction to queue build ups. Preliminary experimental results have demonstrated CCPD competitiveness as compared to widely used TCPs. We are in the process of generating more extensive experimental results. In addition, we are currently studying a hybrid CCP/CCPD congestion avoidance mechanism, which activates the derivative part of the controller only in appropriate path scenarios. The goal is to guarantee best performance regardless of the network path characteristics.

Acknowledgment

Work supported in part by JSPS Grant-in-Aid for Scientific Research KAKENHI B (21300024).

References

[1] K.J.Astrom and B.Wittenmark, “Computer Controlled Systems.” *Englewood Cliffs, NJ*: Prentice Hall, 1990.

[2] L. S. Brakmo and L. L. Peterson, “TCP Vegas: End to End Congestion Avoidance on a Global Internet,” *IEEE J. Select. Areas Commun.*, Vol. 13, No. 8, pp. 1465-1480, 1995.

[3] D. Cavendish, M. Gerla, and S. Mascolo, “A Control Theoretical Approach to Congestion Control in Packet Networks,” *IEEE Transactions on Networking*, Vol. 12, No. 5, pp. 893-906, October 2004.

[4] D. Cavendish, K. Kumazoe, M. Tsuru, Y. Oie, and M. Gerla, “Capstart: An Adaptive TCP Slow Start for High Speed Networks,” *IEEE First International Conference on Evolving Internet*, best paper award, pp. 15-20, August 2009.

[5] D. Cavendish, K. Kumazoe, M. Tsuru, Y. Oie, and M. Gerla, “Capacity and Congestion Probing: TCP Congestion Avoidance via Path Capacity and Storage Estimation,” *IEEE Second International Conference on Evolving Internet*, best paper award, September 2010.

[6] M. Chen, J. Zhang, M. N. Murthy, and K. Premaratne, “TCP Congestion Avoidance: A Network Calculus Interpretation and Performance Improvements,” *Proceedings of INFOCOM05*, Vol. 2, pp. 914-925, March 2005.

[7] J-Y. Choi, K. Koo, J. S. Lee, and S. H. Low, “Global Stability of FAST TCP in Single-Link Single-Source Network,” *Proceedings of the 44th IEEE Conference on Decision and Control*, pp. 1837-1841, December 2005.

[8] S. Floyd and K. Fall, “Promoting the Use of End-to-End Congestion Control in the Internet,” *IEEE/ACM Transactions on Networking*, Vol. 7, No. 4, August 1999.

[9] J. Martin, A. Nilsson, and I. Rhee, “Delay-Based Congestion Avoidance for TCP,” *IEEE/ACM Transactions on Networking*, Vol. 11, No. 3, pp. 356-369, 2003.

[10] R. Kapoor, L-J Chen, L. Lao, M. Gerla, and M. Y. Sanadidi, “CapProbe: A Simple and Accurate Capacity Estimation Technique,” *Proceedings of SIGCOMM 04*, Portland, Oregon, pp. 67-78, Sept. 2004.

[11] L. Kleinrock, “Queueing Systems. Volume II: Computer Applications,” *Wiley*, 1976.

[12] O.J.Smith, “A controller to overcome dead time,” *ISA J.*, Vol.6, No.2, pp. 28-33, Feb. 1959.

Evaluating an Open Source eXtensible Resource Identifier Naming System for Cloud Computing Environments

Antonio Celesti, Francesco Tusa, Massimo Villari and Antonio Puliafito
Dept. of Mathematics, Faculty of Engineering, University of Messina
Contrada di Dio, S. Agata, 98166 Messina, Italy.
e-mail: {acelesti, ftusa, mvillari, apuliafito}@unime.it

***Abstract*—Clouds are continuously changing environments where services can be composed with other ones in order to provide many types of other services to their users. In order to enable cloud platforms to manage and control their assets, they need to name, identify, and retrieve data about their virtual resources in different operating contexts. These tasks can not be easily accomplished using only the DNS and this leads cloud service providers to design proprietary solutions for the management of their name spaces. In this paper, we discuss a possible cloud naming system based on the eXtensible Resource Identifier (XRI) technology. More specifically, we evaluate the performance of OpenXRI, one of its open source implementations, simulating typical cloud name space management tasks.**

***Keywords*-Cloud Computing, Naming System, XRI, OpenXRI.**

I. Introduction

Nowadays, cloud providers supply many kinds of Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) to their users, e.g., common desktop clients, companies, governments, organizations, and other clouds. Such services can be arranged composing and orchestrating several Virtual Environments (VEs) or Virtual Machines (VMs) through hypervisors.

The overwhelming innovation of cloud computing is that cloud platforms can react to events internally rearranging the VEs composing their services pushing down management costs, and the interesting thing is that cloud users are not aware of changes, continuing to use their services without interruptions according to a priori Service Level Agreements (SLAs). For example, when a physical server hosting an hypervisor runs out or is damaged, the cloud can decide to move or “migrate” one or more VEs into another server of the same cloud’s datacenter acting as virtualization infrastructure. Further migrations can be triggered for many other reasons including power saving, service optimization, business strategy, SLA violation, security, etcetera. In addition, if we consider the perspective of cloud federation where clouds cooperate sharing computational and storage resources, a VE might migrate also into a server of another cloud’s virtualization infrastructure. Another business model which can take place in federated scenarios might be the rent of a VEs from a cloud to another.

Such a dynamic, variegated, and continuously changing scenario involves no just cloud services and VEs, but also other cloud entities such as physical appliances and cloud users. All these entities need to be named and represented both in human-readable and in machine-readable way. Moreover, they need also to be resolved with appropriate data according to a given execution context. For example, as a VE needs to be identified by a name, it may happen that different entities (e.g., cloud software middlewares, cloud administrators, cloud user, etcetera) may be interested to resolve that name retrieving either data concerning general information on the VE (e.g., CPU, memory, kernel, operating system, virtualization format version), data regarding the performance of the VE (e.g., used CPU and memory), or by means of a Single-Sign-On (SSO) authentication service (e.g., using an Identity Provider (IdP) asserting the trustiness of the VE when it migrate from a place to another in order to avoid identity theft), and many others. In addition, the scenario becomes more complex if we consider the fact that these entities might hold one or more names and identifiers also with different levels of abstraction. In our opinion, for the aforementioned concerns the management and integration of cloud name spaces can be difficult because such a scenario raises several issues concerning naming and service location for all the involved entities and the traditional DNS-based systems along with URL, URI, and IRI standards are inadequate for cloud computing scenarios.

In order to discourage a possible evolving scenario where each cloud could develop its own proprietary cloud naming systems with compatibility problems in the interaction among different cloud name spaces, this paper aims to propose a standard approach for the designing of a seamless cloud naming system able to manage and integrate independent cloud name spaces also in a federated scenario.

The paper is organized as follows: Section II provides a brief description of cloud name spaces. Section III describes the state of the art of naming systems and the most widely adopted solutions in distributed systems and in ubiquitous computing environments. In Section IV, we provide an overview of the XRI technology motivating how it suits the management of cloud name spaces. In Section V we discuss OpenXRI [1], one of the XRI open source implementations, showing how to use it as cloud naming system. An analysis

ISBN: 978-1-61208-141-0

of the performances of OpenXRI managing a simulated cloud name space is discussed in Section VI. Conclusions and lights to the future are summarized in Section VII.

II. Cloud Name Space Issues

In this Section, we briefly summarize the main cloud name space issues which have already been analyzed in our previous work [2]. Despite the internal cloud structure, we think cloud entities have many logical representations in various contexts. In addition, there are many abstract, structured entities (e.g., a distributed cloud-service built using other services, each one deployed in a different VE). These entities are characterized by a high-level of dynamism: allocations, changes and deallocations of VEs may occur frequently. Moreover, these entities may have one or more logical representations in one or more contexts. But which are the entities involved in cloud computing? In order to describe such entities, we introduce the generalized concept of *Cloud Named Entity* (CNE). A CNE is a generic entity indicated by a name or an identifier which may refer both to real/abstract and simple/structured entity. As depicted in Figure 1, examples of CNE may be a cloud itself, a cloud federation, a virtualization infrastructure, a server running an hypervisor, a VE, a cloud service, or cloud users including companies, governments, universities, cloud technicians, and desktop clients.

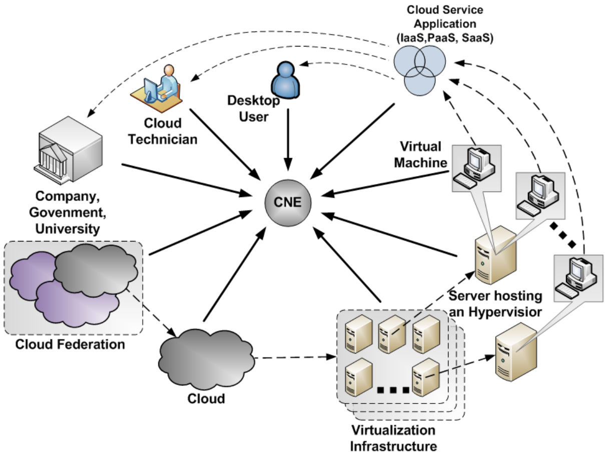

Cloud Service Application (IaaS,PaaS, SaaS)
Cloud Technician
Desktop User
Virtual Machine
Company, Govenment, University
CNE
Server hosting an Hypervisior
Cloud Federation
Cloud
Virtualization Infrastructure

Figure 1. Examples of generic CNEs.

In our abstraction, we assume that a CNE is associated to one or more identifiers. As a CNE is subject to frequent changes holding different representations in various *Cloud Contexts* (CCNTXs), the user-centric identity model [3] seems to be the most convenient approach. We define a CCNTX as an execution environment where a CNE is represented by one or more identifiers and has to be processed. In this work, we assume a CNE is represented by one or more *CCNTX Resolver Server(s)*, which are servers returning data or services associated to a CNE in a given CCNTX. Figure 2 depicts an example of CNE associated with six identities within four CCNTXs. The target CNE holds identity 1, 2 inside CCNTX A, identity 3 inside CCNTX B, identity 4 inside CCNTX C, and identity 5, 6 inside CCNTX D. We define a *Cloud Naming System (CNS)* as a system that maps one or more identifiers to a CNE. A CNS consists of a set of CNEs, an independent cloud name space, and a mapping between them. A cloud name space is a definition of cloud domain names. Instead, a name or identifier is a label used to identify a CNE. A client resolver which needs to identify a CNE in a given CCNTX performs a resolution task. Resolution is the function of referencing an identifier to a set of data or services describing the CNE in several CCNTXs.

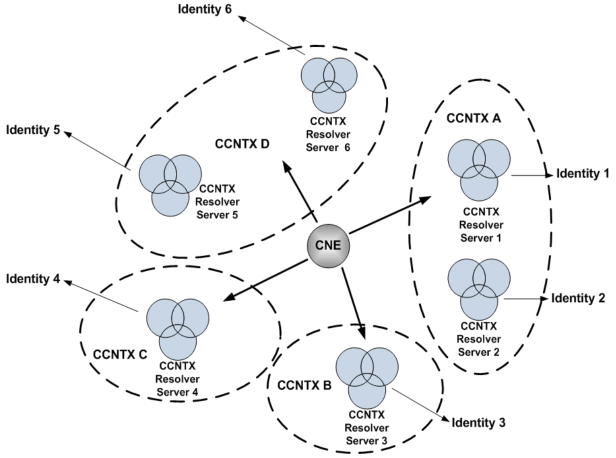


Figure 2. Examples of a generic CNE associated to several CCNTXs.

III. Related Work and Background

Cloud computing is generally considered as one of the more challenging research field in the ICT world. It mixes aspects of Utility Computing, Grid Computing, Internet Computing, Autonomic computing and Green computing [4], [5]. As previously discussed, in such new emerging environments, even though naming and resource location raise several issues, there have not been many related works in literature yet regarding naming systems managing cloud name spaces, as DNS is still erroneously considered the “panacea for all ills”. In fact, DNS presents some problems: it is host centric, unsuitable for complex data and services location, and it is not suited to heterogeneous environments. Possible improvements might come from the naming system works in high-dynamic, heterogeneous and ubiquitous environments. An alternative to the DNS is presented in [6]. The authors propose a Uniform Resource Name System (URNS), a decentralized solution providing a dynamic and fast resource location system for the resolution of miscellaneous services. Nevertheless, the work lacks of an exhaustive resource description mechanism. With regard to naming

ISBN: 978-1-61208-141-0

system in ubiquitous computing, in [7] the authors propose a naming system framework for smart space environments. The framework aims to integrate P2P independent cloud naming systems with the DNS, but appears unfitted to be exported in other environments. In addition it aims to localize and identify an entity that moves from a smart space to another using as description mechanism the little exhaustive DNS resource records. A hybrid naming system that combines DNS and Distributed Hash Table (DHT) is presented in [8]. The authors adopt a set of gateways executing a dynamic DNS name delegation between DNS resolver and DHT node.

Regarding naming, name resolution, and service location in federated cloud environments, in our previous work [2], besides highlighting the cloud name space issues previously discussed, we proposed a generic theoretical cloud naming framework for the management of cloud name spaces. As possible representation of the cloud naming framework we chose XRI [9] and the eXtensible Resource Descriptor Sequence (XRDS) [10] technologies which are also the focus of this work. How will be described in the following Sections the aim of this paper is to evaluate the performances of several operational tasks using the OpenXRI implementation by means of the simulation of typical cloud name space management tasks.

IV. An XRI Naming System for Cloud Computing

In this Section, after a brief description the XRI technology, we motivate how it can help the cloud name space management.

The XRI protocol provides a standard syntax for identifying entities, regardless any particular concrete representation. The XRI system is similar to DNS, including a set of hierarchical XRI authorities but more powerful. The protocol is built on URI (Uniform Resource Identifiers) and IRI (Internationalized Resource Identifiers) extending their syntactic elements and providing parsing mechanisms. Particular types of URI are URN and URL. Since an URL is also an URI, the protocol provides a parsing mechanism from XRI to URL. Therefore XRI is also compatible with any URN domain. XRI supports persistent and reassignable identifiers by means of i-numbers (Canonical ID) and i-names (Local ID). It also provides four types of synonyms (LocalID, EquivID, CanonicalID, and CanonicalEquivID) to provide robust support for mapping XRIs, IRIs, or URIs to other XRIs, IRIs, or URIs that identify the same target entity. This is particularly useful for discovering and mapping to persistent identifiers as often required by trust infrastructures. XRI enable organization to logically organize entities building XRI tree. According to the XRI terminology, each entity in the tree is named authority. The protocol provides two additional options for identifying an authority: Global Context Symbols (GCS) and cross-references. Common GCS are "=" for people, "@" for organization, and "+" for generic concepts. For example the xri://@XYZ*marketing indicates the marketing branch of an organization named XYZ, where the "*" marks a delegation.

An authority is resolved by means of an XRDS document representing a simple, extensible XML resource description format standard describing the features of any URI, IRI, or XRI-identified entity in a manner that can be consumed by any XML-aware system. Each XRDS describes which types of information are associated to an authority an the way in which they can be obtained. Using HTTP, XRI resolution involves two phases: authority resolution which is the phase required to resolve a XRI into a XRDS document from an XRI Authority Resolution Server (ARS), and Service End-Point Selection which is the phase of selection of the SEP server (e.g., web service, service provider, web application) returning the data describing the entity in a given context. The same SEP server can also return different data of the same authority.

In our opinion, as XRI meets the requirements of cloud name space management, it can be adopted to develop a seamless mechanism for retrieve data regarding CNEs. As XRI is compatible with IRI naming systems, there is not the need to use a unique global naming system, even though this would be possible. This feature allows clouds to manage their own XRI naming systems, mapping them on the global DNS maintaining the compatibility with the existing naming systems. Moreover, with XRI a cloud can keep different trees representing IaaS, PaaS, and SaaS. In addition, such a technology can be used for both identify and resolve VMs and whole *aaS by means of the resolution of XRI authorities. In addition, the XRDS document can be used to describe a XRI authority associated to a target service of VM, indicating how to resolve it by means of the corresponding SEP.

For example the cloud service provider may need to retrieve three types of information about an authority representing a VM, resolving it in three different ways. In the fist way the VM has to be resolved by means of general data (e.g., CPU, memory, kernel, operating system), in the second way the VM has to be resolved by means of real time performance data (e.g., amount of used CPU and memory used), in the third way instead the VM has to be resolved by means real time data regarding an internal running application (e.g., the percentage of processed data). Such a situation can be addressed by mean of three different XRDs inside the XRDS document corresponding to the VM, authority, each one pointing to a target SEP server.

V. How to Manage Cloud Name Spaces Using an OpenXRI Architecture

Regarding the implementation of the XRI technology, currently there are not many available solutions on the market. Nowadays, in our opinion, the OpenXri Project is one of the best open source initiatives which aims to promote the development of XRI-based applications. Therefore, in this Section we describe how to implement a CNS using the java libraries developed by the Openxri Project. Our practice of CNS includes the following components: the XRI Authority Resolution Server (ARS) 1.2.1, the XRI Client Resolver (CR), the XRI Cloud Name Space Management (CNSM) front-end and the SEP Server. The OpenXRI has provided the java libraries to arrange the following components:

- **The ARS 1.2.1**, a server for the resolution XRI authorities (i.e., in our scenario CNEs). It is provided along with a web application to allow administrators to create, move, and delete authorities and thus managing XRI trees.
- **The XRI CR**, a software client which resolves an authority queering a ARS, retrieving the corresponding XRDS document, and performing a "SEP Selection Task" choosing the right SEP Server acting in a given CCNTX for the resolution of an XRI authority. The XRI CR is used by each entity interested to resolve a CNE name, e.g., another cloud, a desktop client, a service provider, an IT society, and so on.

The XRI ARS developed by the Openxri Project provides an administration web interface where an user can interactively manage his name space. Such a condition is very penalizing for our scenario, as we assumed that the cloud name space should be also managed automatically by the cloud middleware according to the business model in force on the cloud. In fact, in our opinion there are circumstances where the interaction with an administrator is required and other cases where the cloud has to arrange its assets automatically by itself. For such reasons we have designed the **XRI CNSM front-end** offering both a standard SOAP web service interface in order to make the naming system controllable by any cloud platform, and additional specific utilities for the name space management of cloud computing environments. The choice of using SOAP is motivated by the fact that it provides a consolidated framework offering security an Quality of Service (QoS) support. At any rate, nothing prevents the possibility to use another web service technology. In addition, **SEP Servers** have been developed by means of Restfull web services. The aim of the SEP Server is to resolve CNE name in a given CCNTX sending data in XML format to the XRI CR. In this case, the choice of the Restfull technology has been motivated by the fact that it offers better performances than SOAP in term of response time (this is very useful especially when it is needed to retrieve real-time data, e.g., the performances of a VM). At any rate, also in this case, nothing prevents the possibility to use another web service technology.

Security, is a hot topic in cloud computing. For this reason, even though security is not the focus of this paper is worthwhile to spend a few words about the security of the proposed CNS. Regarding the CNE name resolution, the XRI technology natively supports secure resolution using the Security Assertion Markup Language (SAML) [11]. As far as it is concerned the interaction between the cloud middleware and the XRI CNSM front-end, it can be easily secured using the WS-Secutity features [12] of the SOAP protocol. In the end, considering the information retrieval of an XRI software client from the Restful SEP Server, security can be accomplished using https.

In Figure 3 is depicted an example of CNE name resolution. In step 1, the XRI CR software wants resolver a CNE name and contacts the XRI ARS making a resolution request. In step 2, the XRI ARS resolves the name and responds to the XRI CR sending the XRDS document corresponding to the the CNE name. In step 3 the XRI CR performs a SEP selection task choosing the server for the resolution of the CNE name in a given CCNTX to which performing a Restfull web service request. In step 4 SEP server responds with the data resolving the CNE name in the corresponding CCNTX, so that the XRI CR can process the obtained XML data. In the rest of the paper, as there are not limits to the type of CNE which our architecture is able to manage, we focus on the name space management and information retrieval of VMs.

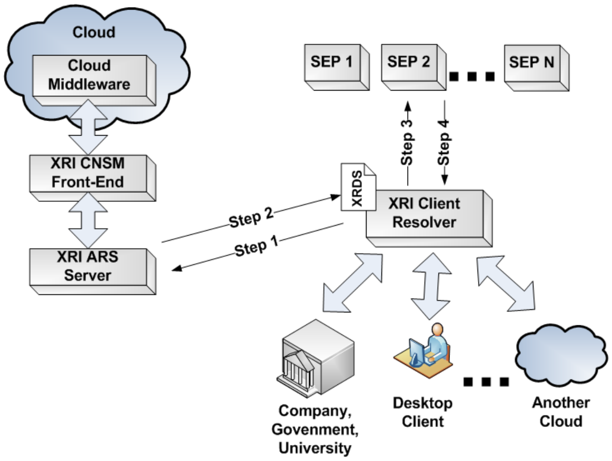


Figure 3. Example of CNE name resolution in a CNS.

As clouds are highly dynamic environments the logical and physical arrangement of its resources can continuously change. For example a VM can be either physically moved from a server running a hypervisor to another. Figure 4 depicts such a situation presenting an example of XRI name space changing due to a CNE name movement. In the XRI name space tree on the left of the Figure at t time the "VE2" CNE name is logically mounted under the "service2" CNE name, instead, on the right of the Figure, at $t + 1$ time is logically mounted under "service1".

VI. Authority Moving Performances

In this Section, we present several experiments on a real testbed in order to evaluate our XRI CNS implementation.

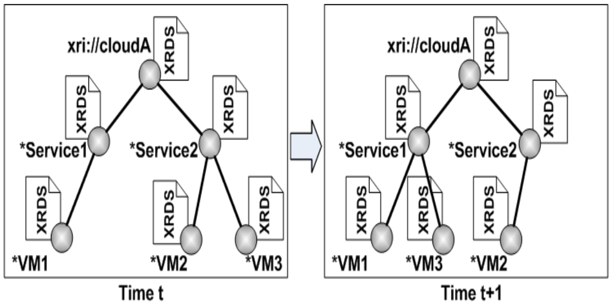


Figure 4. Example of CNE name movement.

More specifically, we focused on the evaluation of the costs due to the movement of XRI authorities (i.e. CNE names) within the XRI tree representing the cloud name space. In order to evaluate the goodness of the OpenXRI architecture for the management of cloud name spaces, in our opinion it is necessary to understand the overall behavior of the architecture under particular conditions of workload. As XRI follows a hierarchical approach for the management of name spaces using one or more tree structures, we decided to stress the operations of XRI authority movement in such structures considering two possible cases: “Wide Tree” and “Deep Tree”. Considering the “Wide Tree” case, we performed movement tasks with 10, 100, and 1000 XRI authority. Instead, as far as it is concerned the “Deep Tree” case, we performed tests with 10, 20, and 30 levels in the XRI tree structure. Experiments have consisted in the movement of the last authority on the right under a the first authority on the left of the XRI tree.

Our experiments have been performed considering a testbed deployed inside a computer running a Redhat Enterprise Linux AS Release 3.0.8 operating system having the following hardware features: Blade LS21 AMD Opteron Biprocessor Dual Core 2218 2.6GHz 8GB RAM. Instead, in order to emulate a cloud middleware interacting with the SOAP web service interfaces of the CNSM front-end, we used JMeter, an open source automatic client tool, which has been deployed within another computer. More specifically, we store a typical cloud behavior pattern and then we applied it repetitively.

To estimate the workload of OpenXRI ARS 1.2.1 we evaluated the Response Time of the system expressed in *msecs*. We have measured the time interval between the request phase to the XRI CNSM front-end at Ts and the response time at Tr, taking place in the receiving phase. In our graphs we reported the total time spent to accomplish each task: $Tt = Ts + Tr$. The exchange of requests and responses is measured in a local network (LAN, without any Internet connection), since the measurements are not affected from the network communication parameters (e.g., throughput, delays, jitter, etcetera). The series of tests executed (50 runs for each simulation) guarantee a wide coverage of possible results. The confidence interval (at 95%) indicates the goodness of our analysis. Figure 5 summarizes the response time trend regarding the authority movement tasks using the “Wide Tree” and considering 10, 100, and 1000 authorities. On the x-axis we have represented the number of considered nodes, whereas on the y-axis we have represented the response time expressed in milliseconds. Observing the graph, we notice that with 1000 nodes we have a response time of 40 seconds, a rather high value, but reasonable considering the presence of 1000 operating VEs. Instead, in the case of 10 and 100 nodes, the “Wide Tree” needs a time that ranges from about 1 to 5 seconds in order to perform authority movement tasks. This latter results are reasonable in cloud environments, in particular, if we assume a scenario where a single VE needs to be boot up, from an unrunning state. Usually, the time needed for the VE boot-up is higher than the time spent by OpenXRI ARS for any type of tree reconfiguration.

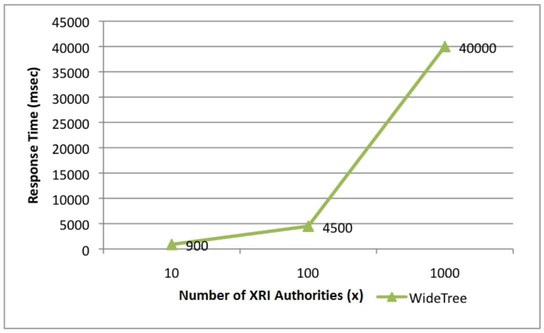


Figure 5. Authority movement tasks in the “Wide Tree” case with 10, 100, and 1000 authorities.

Instead, the graph depicted in Figure 6 shows the response time trend concerning authority movement tasks considering a “Deep Tree” with 10, 20, and 30 tree levels. On the x-axis we have represented the number of levels, whereas on the y-axis we have represented the response time expressed in milliseconds. Observing the graph the worst case (an authority movement within a tree with 30 levels) implies

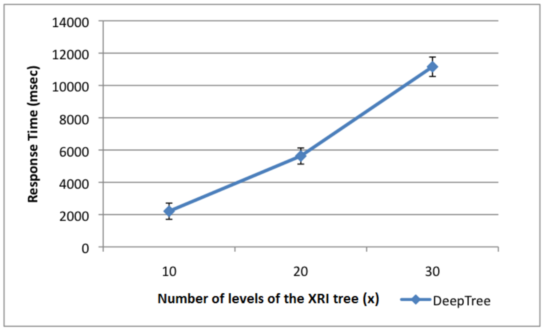


Figure 6. Authority movement tasks in the “Deep Tree” case with 10, 20, and 30 levels.

12 seconds. In reality, the case under analysis may be considered as an event with a low probability in cloud computing environments. In fact, it represents an hierarchical structure with 30 levels in which we should identify 30 CNEs with hierarchical relationships. However, with a few levels, and with our hardware configuration the response time we can achieved is rather low.

VII. Conclusions and Future Works

In this paper we presented a solution for identifying and resolving VMs and more in general various other CNEs in different operating cloud contexts. Particularly, we presented the XRI technology as a possible solution to these problems, evaluating one of its open implementation, that is OpenXRI. This works has highlighted how several tasks can be accomplished using OpenXRI for the management of cloud name spaces. The conducted experiments show the goodness of OpenXRI and how it is particularly suitable to our goals. In future works we are planning to apply some improvements to the OpenXRI Authotity Resolution Server 1.2.1 for the accomplishment of the tasks needed for the management of cloud name spaces.

Acknowledgements

The research leading to the results presented in this paper has received funding from the European Union's Seventh Framework Programme (FP7 2007-2013) Project RESERVOIR under grant agreement number 215605 and from the Union's Seventh Framework Programme (FP7 2007-2013) Project VISION-Cloud under grant agreement number 217019.

References

[1] OpenXRI Project, XRI applications and libraries, http://www.openxri.org/.

[2] A. Celesti, M. Villari, and A. Puliafito, "Ecosystem of cloud naming systems: An approach for the management and integration of independent cloud name spaces," (Los Alamitos, CA, USA), pp. 68–75, IEEE Computer Society, 2010.

[3] G.-J. Ahn, M. Ko, and M. Shehab, "Privacy-enhanced user-centric identity management," in *IEEE International Conference on Communications, ICC '09*, pp. 14–18, June 2009.

[4] I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud computing and grid computing 360-degree compared," in *Grid Computing Environments Workshop, 2008. GCE '08*, pp. 1–10, 2008.

[5] R. L. Grossman, "The case for cloud computing," in *IT Professional*, vol. 11, pp. 23–27, March 2009.

[6] D. Yang, Y. Qin, H. Zhang, H. Zhou, and B. Wang, "Urns: A new name service for uniform network resource location," in *Wireless, Mobile and Multimedia Networks, 2006 IET International Conference*, pp. 1–4, 2006.

[7] Y. Doi, S. Wakayama, M. Ishiyama, S. Ozaki, T. Ishihara, and Y. Uo, "Ecosystem of naming systems: discussions on a framework to induce smart space naming systems development," in *ARES*, p. 7, April 2006.

[8] Y. Doi, "Dns meets dht: treating massive id resolution using dns over dht," in *Applications and the Internet International Symposium*, pp. 9–15, January 2005.

[9] Extensible Resource Identifier (XRI) Syntax V2.0, Committee Specification, OASIS, 2005.

[10] Extensible Resource Identifier (XRI) Resolution V2.0, Committee Draft 03, OASIS, 2008.

[11] "Security assertion markup language, oasis, http://www.oasis-open.org/committees/security."

[12] "Web services security: Soap message security 1.0, oasis, http://www.oasis-open.org/committees/wss."

Signal Processing-based Anomaly Detection Techniques: A Comparative Analysis

Joseph Ndong
Université Pierre et Marie Curie
LIP6-CNRS, France
Email: joseph.ndong@lip6.fr

Kavé Salamatian
Université de Savoie Chambery Annecy
LISTIC PolyTech, France
Email: kave.salamatian@univ–savoie.fr

***Abstract*—In this paper, we present an analysis for anomaly detection by comparing two well known approaches, namely the Principal Component Analysis (PCA) based and the Kalman filtering based signal processing techniques. The PCA-based approach is coupled with a Karuhen-Loeve expansion (KL) to achieve higher improvement in the detection performance; on the other hand, based on a Kalman filter, we built a new method by combining statistical methods such as: gaussian mixture and a hidden markov modellers, which allows us to obtain performances better than those obtained with the PCA-KL expansion method. For this newer method, our approach consists of not assuming anymore that the Kalman innovation process is gaussian and white. In place, we are assuming that the real distribution of the process is a mixture of normal distributions and that, there is time dependency in the innovation that we will capture by using a Hidden Markov Model. We therefore derive a new decision process and we show that this approach results in an considerable decrease of false alarm rates. We validate the two comparative approaches over several different realistic traces.**

***Index Terms*—Anomaly Detection, Monitoring System, Kalman filter, GMM, HMM**

I. Introduction

The literature of the recent years has used two fundamental classes of monitoring techniques, to implement anomaly detection techniques for networking applications: PCA-based and Kalman-filter based methods. An anomaly detector consists essentially of two components: (i) an entropy reduction component and (ii) a decision component applying statistical tests to a *decision variable* issued from the first step. The entropy reduction step is here, to simplify the second step. When using statistical signal processing based techniques, entropy reduction is obtained by a predictive model that uses a model of normal behavior to forecast the values of the parameters to monitor. The prediction error obtained after filtering out the normal behavior model prediction has a smaller entropy than the initial signal, resulting in a considerable entropy reduction. A decision variable is thereafter derived as a function of the prediction error and fed to the decision stage. In the decision stage a statistical test is applied to the decision variable and an anomaly is detected if this variable exceed a threshold.

In PCA-based method, the predictive model used in the entropy reduction step is built by using a projection in a *low* dimensional *orthogonal* sub-space, that minimizes the approximation error. This subspace is derived using Principal Component Analysis (PCA) or more precisely a Karhunen-Loeve Transform (KLT). The decision variable in PCA-based techniques is obtained as, a *square sum of the prediction errors* made by projecting the observed signal in the PCA defined subspace (see [1], [2] for a detailed description).

Kalman-filter based techniques first, calibrate a Maximum-Likelihood based model for normal behavior modeling for the entropy reduction step. Thereafter the decision variable is obtained as the *innovation process* at the **output** of a Kalman-filter, that filters the normal behavior component from network observations [3].

In the two above cited methods, under the conditions that the observed signals follow a jointly gaussian distribution, the decision variable is known to converge to a zero mean, gaussian and white (uncorrelated) signal . However, realistic network traffic is well known to **not** follow a gaussian distribution. Moreover, anomalies can break the ***stationarity*** of the decision signal when they happen generally. These last two problems result in divergence from the basic assumptions (gaussianity) under which classical anomaly detectors are built. Therefore, the error probabilities bounds predicted by assuming a gaussian distribution at the decision variable are not tight enough to be used for a robust anomaly detection test. Prior work [1], [2], [4] have shown that, even after careful calibration of the normal behavior model, the decision variables still exhibit non-gaussian and correlated behaviors. This last problem explains the high false alarm rate observed when using classical anomaly detection approaches.

The fundamental issue in the area of anomaly detection in networks is the false alarm rate *vs.* detection rate trade-off. This trade-off is represented by the ROC (Receiver Operating Characteristics) curve [4]. In operational settings, one would like to attain a high detection rate (larger than 90%) with the lowest false alarm probability. The main aim of this paper is twofold: to see between the two methods, PCA or Kalman filtering, which one is the most robust towards deviation from the gaussianity assumptions, and to improve the false alarm versus detection rate trade-off by making a more robust anomaly detection decision. This will allow us to compare the two above methods and to show that the method based on Kalman filtering performs best.

The contribution of this paper is a careful comparative analysis of the decision step in PCA and Kalman-filter based

Copyright (c) IARIA, 2011. ISBN: 978-1-61208-141-0

anomaly detectors. Based on this analysis, we are proposing one approach to improve the robustness of the decision step. This approach is based on accepting that the distribution of the decision variable at the output of the anomaly detector is not an uncorrelated gaussian process. We will rather assume that, the decision variable follows a distribution that can be modelled by a Gaussian Mixture Model (GMM) with a *small* number of components.

The residual temporal correlation in the decision variable is modelled by a Hidden Markov Model (HMM) defined on the sequence of component index of the GMM model. Since anomalies might be rare, we assume that they might happen in some (but not all) components of the GMM. This means that, by inferring the states of the HMM calibrated on the residual variable, one is able to decide if an anomaly has happened.

The above model is not the only one that is able to capture the deviation from gaussian hypothesis as well as the residual correlation in the decision variable. However, we will show in this paper that the crude and relative complexity of the model is powerful enough to result in a considerable decrease in false alarm rate for a given detection ratio.

In summary our proposed robust decision scheme differs from previous work in the field of network anomaly detection in at least two ways: i) we do not assume that the innovation process does strictly remain a zero mean gaussian process. Instead, we assume that the real distribution of the process is a mixture of gaussians, and residuals can be split in different mixture components (where anomalies might or might not happen), ii) we use hidden markov models over the different mixture components to capture residual time dependencies that can be relevant to anomaly detection. Therefore, anomaly decision is not done as usual, using a simple threshold based decision but rather in a two step approach: we first use a Viterbi algorithm to estimate the Maximum a posteriori (MAP) estimates of the state sequence of the HMM; clearly the Viterbi path will select some of the HMM states (i.e some of the GMM components, because each HMM state contains part of the GMM components) able to capture all the low and high variations in the decision variable; thereafter we only apply thresholding in these selected states.

The organization of this paper is as follows. Section 2 describes the monitoring system we used. Section 3 deals with the methodology we adopt in our anomaly detection scheme. In section 4, we detail our calibration method and we validate our approach by showing efficient results, which we compare with the results obtained by the KL-PCA method. Section 5 concludes our work and fix some ideas for future works.

II. Architecture of the Monitoring System

We are assuming a novel network monitoring system shown in Figure 1. The monitoring system contains three functional blocks: data collection (by a NOC-Network Operations Center), a data analysis block and a decision process phase. The data collection block is responsible for receiving the flow of microscopic measurement coming from network equipment in form of SNMP or Netflow/IPFIX flows. This flow is aggregated and transformed to a vector of time sequences that are fed to other blocks. The Data analysis block calibrates the normal behavior model, by applying machine learning techniques like PCA or Maximum Likelihood analysis and generates the decision variable by filtering the expected normal behavior from the observations. The last block implements the decision process. Classically, the decision stage consists simply of a comparison with a threshold, *i.e.* if the decision variable exceeds the threshold, an anomaly is detected.

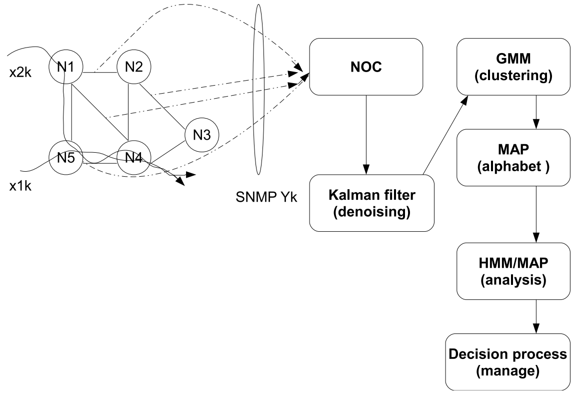


Fig. 1. A new monitoring system combining a Kalman filter for entropy reduction, a GMM for clustering, a HMM for time dependencies learning and finally the use of the Viterbi algorithm for decision manage.

We propose in this paper a more elaborate decision scheme containing three new components shown in Fig. 1. These three new stages need to calibrate two models: a GMM to account for deviation from gaussian distribution of the decision variable and a HMM to integrate the residual temporal correlation. The GMM is calibrated over a learning set of continuous valued decision variable time series coming directly from observations. Clearly, using the multi-dimensional Kalman innovation process, we form a *one-dimensional* residual vector, which we put as input of the GMM to build a family of gaussians. These normal distributions will be later transformed into discrete sequences using a Maximum A Posteriori criterion. Thereafter the HMM is calibrated over the discrete sequence of GMM Components membership, for time dependence learning. The three new components consists of a MAP (Maximum A Posteriori) phase that maps each observation to an index of the discrete mixture using a Maximum a Posteriori probability; thereafter a Viterbi algorithm is used to detect and **select** the state of the decision variable. Finally in the third and last step, threshold based anomaly detection is only applied if we are in these states detected by the Viterbi path. We will describe in detail these three steps in the forthcoming.

III. Methodology

A. Normal behavior modeling

The first step is to seek for a normal behavior model that captures traffic dynamics. For this sake, two classes of models have been proposed in the literature: PCA-based model and state-space model. In PCA-based method, we build the predictive model assuming the projection in an orthogonal subspace obtained through application of PCA in a good predictor of the signal. We refer the reader to a precise description of this class of model in [1]. State-space model is a classical approach to model dynamical signal and is shown in Eq. 1.

$$\begin{cases} X_{t+1}=C_tX_t+W_t \\ Y_t \ =A_tX_t+V_t \end{cases} \tag{1}$$

The model contains two equations: the first describes the dynamic of state variations and the second one the measurement dynamics that may result, in not directly observing the states but rather a linear combination of them. The noise W_t accounts for intrinsic noise in the state variations as well as modeling errors; the noise V_t accounts for measurement errors. Both are assumed to be uncorrelated zero-mean gaussian white-noise processes with covariance matrices Q_t and R_t, respectively. This class of models can be used to model a very large class of signals and in particular, they can be calibrated to model any signal if the number of states is large enough [4].

Normal behavior model calibration can be done for PCA models using the diagonalisation of the observation covariance matrix [1]. The calibration of the state-space model is presented in [4].

B. Decision variable generation

After calibrating the normal behavior model over a learning set, we can use this model to generate the decision variable. The decision variable in PCA is derived as the *square sum of the prediction error*, where the prediction error is derived as the difference between the observation and the projection in the PCA-based subspace; the reader can refer to [1] for an in detail analysis of the decision variable generation for PCA.

For state-space based approach, the decision variable is obtained through the application of a Kalman filter. The decision variable is derived as the *sum of squares of the innovation process weighted by its variance*. When the state space is a n-dimensional, the resulting decision variable becoming a χ^2 random variable with n degree of freedom. When n is large enough, this converges to a gaussian random variable with mean n and variance $2n$.

C. Kalman filter equations

The first problem to solve after building a simple model to monitor features (link counts and TCP/UDP metrics), is to find an optimal estimate ($\hat{X}_t$) of our unobservable network states X_t, given a set of past and current observations $\{Y_1, ..., Y_t\}$. To estimate the state of the system using only all information until time t, a robust method is the Kalman fixed-interval filtering algorithm. From the dynamical linear system, we refer to Y_t as the observation vector at a specific time t. And the state of the system at time t is given by X_t; let also $\hat{X}_{t|k}$ denotes the estimate of X_t using all the information available up to time k, i.e, $\forall\tau < k$. $\hat{X}_{t+1}$ denotes the estimate of X_{t+1} using all the information up to time t, (this constitutes the *phase predictor*). The quantity $\hat{X}_{t+1|t+1}$ denotes the estimate of X_{t+1} using all past information and the recently arrived data point at time t+1. On the other hand, $P_{t|t}$ denotes the variance of the *state estimate* and $P_{t+1|t}$ indicates the variance of *the state prediction*. As it is shown in its earlier elaboration, the Kalman filter addresses the problem of estimating a discrete state vector when the observations are only a linear combination of the underlying state vector. As an iterative algorithm, it estimates the system state using two steps: the filter runs as a *predictor-corrector* algorithm. Prediction comes in the *time update* phase, and correction in the *measurement update* phase.

- **Prediction step** (*time update equations*):
  In this step, the estimated state of the system at time t, $\hat{X}_{t|t}$, is used to predict the state at next time t+1, $\hat{X}_{t+1|t}$. And, as we know that the noise W_t influences the evolution of the system at each time t, we compute only the variance of the prediction, $P_{t+1|t}$ based on the updated variance at the previous time t, $P_{t|t}$, and the noise covariance at the same time, Q_t. The error covariance $P_{t+1|t}$ provides an indication of the uncertainty associated with the state estimate.

$$\begin{cases} \hat{X}_{t+1|t}= \quad C_t\hat{X}_{t|t} \\ P_{t+1|t}=C_tP_{t|t}C_t^T+Q_t \end{cases} \tag{2}$$

- **Correction step** (*measurement update equations*):
  This step updates (corrects) the state and the variance of the estimate in the previous step, using a combination of their predicted values and the new observations Y_{t+1}. The accuracy of this update depends on the Kalman innovation $Y_{t+1} - A_{t+1}\hat{X}_{t+1|t}$.

$$\begin{cases} \hat{X}_{t+1|t+1} =\hat{X}_{t+1|t} + K_{t+1}(Y_{t+1} - A_{t+1}\hat{X}_{t+1|t}) \\ \qquad P_{t+1} =(I - K_{t+1}A_{t+1})P_{t+1|t}(I - K_{t+1}A_{t+1})^T \\ \qquad\qquad + K_{t+1}R_{t+1}K_{t+1}^T \end{cases} \tag{3}$$

In the measurement equations, K_{t+1} denotes the Kalman gain. For more details in linear dynamical system, estimation and Kalman filtering techniques, we refer the reader to : [5],[6],[7] and [8]. The above equations with initial conditions of the state of the system $\hat{X}_{0|0} = E[X_0]$ and the associated error covariance matrix $P_{0|0} = E[(\hat{X}_0 - X_0)(\hat{X}_0 - X_0)^T]$ define the discrete-time sequential recursive algorithm for determining the linear *minimum variance* estimate known as the *Kalman filter*.

D. Gaussian Mixture Model

In Kalman filtering philosophy, it is more generally assumed that the residual remains a zero mean gaussian process, however this assertion is not always true in practice. Thus our motivation in using gaussian mixture model is based on our belief that the real distribution of the process is an ensemble (mixture) of gaussians and there is some time dependency in the innovation (which we will later study by using hidden markov model theory). This assertion allows us to build a method, which has the ability to find anomalies in different families built on a one-dimensional residual process. The GMM takes as input the 1-dimensional residual vector we formed with the multi-dimensional Kalman innovation process, and finds a few number (K) of families (clusters). Each gaussian component is fixed by its first order statistic. The variance obtained for each component will help to determine the suitable number of K. Each cluster contains data coming from one gaussian component. Thereafter, we aim to convert each cluster into discrete sequence of symbols (1,2,3,...) by means of a MAP criterion. In the next step, we propose the use of an hidden markov model (HMM) to classify these discrete groups into P states. Each hidden state could probably contain mixing symbols yielding in different clusters. This operation has the main advantage to discover the potential temporal dependencies in the innovation process.

Technically, to find the values of the model parameters μ_m and Σ_m , as well as the prior probability vector π, we are interested in maximizing the *likelihood* $\pounds(\theta|\mathbf{X}) = p(\mathbf{X};\theta)$ of generating the known observed data (X) given the model parameters $\theta = \{\mu_m, \Sigma_m, \pi_m\}$, 1≤m≤M. **X** denotes all the observation while θ contains all the parameters of the mixture. In other words, we hope to find $\hat{\theta}_{ML}$=argmax $p(x|\theta)$. This approach is called the *Maximum Likelihood* (ML) framework since it finds the parameter settings that maximize the likelihood of observing the data sets. To find the best parameters of the features of θ, the Expectation Maximization (EM) iterative algorithm can be used to simplify the mathematical routines considerably and numerically compute the unknown parameters. In the **E-step** (*Expectation phase*), the parameters are estimated given the observed data and current estimates of the model parameters (i.e EM comes with an initial guess of the model parameters, μ_m, Σ_m and π_m; we have used the K-means algorithm). In the **M-step** (*Maximization phase*), EM takes the expected complete log-likelihood and maximizes it w.r.t. the parameters to estimate (π_m, μ_m, Σ_m). For more details about mathematical routines for EM, see [9],[10],[11].

E. Hidden Markov Model

From the above learning phase, each GMM component is transformed into a sequence of a finite set of alphabet (symbols as 1,2,3,...), using a *maximum a posteriori* criterion. This discrimination phase will help for plugging the above clusters into different a priori unknown states, using hidden markov model. We represent these families of states by the following collection of unknown random variables $\{Q_1, Q_2, ...Q_T\}$ (where Q_t is a constant value with values in $\{1, 2, ..., K\}$). We also represent our alphabet by the known vector $\{O_1, O_2, ..., O_T\}$. Now, the problem is resumed to find a model to produce the states and to determine the probability of each symbol being in a state.

A well known model-based approach to tackle this problem, is the discrete *hidden markov model* (HMM) approach. Our choice of using HMM is based on the fact that: i) potential time dependencies in the innovation process can be modelled and captured using a finite set of a *priori hidden states*, each of them containing a subset of gaussian components ii) relatively efficient algorithm can be derived to solve the problems related to them, [9], [10], [12]. The full HMM model we used is defined by the quantity $\lambda = (A, B, \pi)$ (where A is the *transition matrix*, B is the *emission probabilities* matrix and pi the *prior* probabilities).

To find and estimate the best parameters of our model, we use the well-known *forward-backward* algorithm parameter estimation (or Baum-Welch algorithm). For more details for EM techniques related to hidden markov model, see, [9], [12]. Thereafter, we reuse the model to find the optimal state sequence associated with the given observation sequence. We believe that this final step of our approach will allow us to capture all the variations in the innovation process. An optimal criterion we have chosen here is to find the single best state sequence (path), $Q = \{q_1, q_2, ..., q_T\}$ for the given observation sequence $O = \{O_1, O_2, ..., O_T\}$,i.e., we aim to maximize $P(Q|O, \lambda)$. A formal technique for finding this unique best state sequence is the Viterbi algorithm. The Viterbi algorithm will find a unique path (containing some of the symbols) witch will be able to capture all the variations in the decision variable.

IV. Model Evaluation

A. Experimental Data

In this work we used two kinds of data coming from two different networks: the Abilene and the SWITCH networks. The Abilene backbone has 11 Points of Presence(PoP) and spans the continental US. The data from this network was collected from every PoP at the granularity of IP level flows. The Abilene backbone is composed of Juniper routers whose traffic sampling feature was enabled. Of all the packets entering a router, 1% are sampled at random. Sampled packets are aggregated at the 5-tuple IP-flow level and aggregated into intervals of 10 minute bins. The raw IP flow level data is converted into a PoP-to-PoP level matrix using the procedure described in [2]. Since the Abilene backbone has 11 PoPs, this yields a traffic matrix with 121 OD flows. Note that each traffic matrix element corresponds to a single OD flow, however, for each OD flow we have a seven week long time series depicting the evolution (in 10 minute bin increments) of that flow over the measurement period. All the OD flows have traversed 41 links. Synthetic anomalies are injected into the OD flows by the methods described in [2], and this resulted in 97 anomalies in the OD flows.

The second collection of data we used for our experiments is a set of three weeks of Netflow data coming from one of the peering links of a medium-sized ISP (SWITCH, AS559),[13], [1]. This data was recorded in August 2007 and comprise a variety of traffic anomalies happening in daily operation such as network scans, denial of service attacks, alpha flows, etc. For computing the detection metrics, we distinguish between incoming and outgoing traffic, as well as TCP and UDP flows. For each of these four categories, we computed seven used traffic features: *byte, packet, flow counts, source and destination IP address entropy, unique source and destination IP address counts*. All metrics were obtained by aggregating the traffic at 15 minute intervals resulting in 28x96 data matrix per measurement day. Anomalies in the data were identified using available manual labelling methods, as visual inspection and time series and top-n queries on the flow data. This resulted in 28 detected anomalous events in UDP and 73 detected in TCP traffic. The SWITCH data was collected in a single link and the observed metrics are correlated in time and in space.

B. Validation

1) Tuning of the System parameters.: Our method begins with a learning phase where we calibrate a Kalman filter for denoising, using our linear dynamical system (1), and the collection of observation data . In order to run the Kalman filter, we need the state and measurement matrices A, Q, C and R. For the Abilene data, the matrix A is available given the routing scheme of a network. We thus only need to obtain Q, C and R. It is sufficient to learn the model's parameters using a sample of one week measurements. We find the tuning parameters , using two days of consecutive samples of all link counts, and we do our calibration using the *EM algorithm* developed in [14],[15] and implemented (in R language) in [16]. In Practice to run and find the GMM and HMM parameters, we had used the HMM toolbox [17].

The way to find the tuning parameters for the Kalman filter for the SWITCH data, is slightly different from the case of the Abilene case, because for these data, we don't know a priori the matrix A and it should be estimated. For this case, we used the EM algorithm developed in [18], and implemented (in Matlab) in [19].

To build the model for the PCA and the KL expansion, we use the vector of metrics $X[1:192]$ and the vector of link counts $X[1:288]$ containing the first two days of data, respectively for the SWITCH network and for the Abilene backbone. And we follow exactly the method described in [1] (based on pole-zero diagram) to choose the components number to be included in the model.

To apply the KL expansion, the SVD (Singular Vector Decomposition) technique is applied to the data (resulting to a basis change matrix) and the squared prediction error $Q[k] = e[k]^T e[k]$ ($e[k]$ is the residual process)is computed from, which the decision variable D[k] is calculated. Thus, using the multi-dimensional data (all vectors of metrics and all vectors of link counts), we obtained the *one-dimensional* array $D[k]$ with, which we perform anomaly detection.

As for the KL expansion, the same samples of metrics and link counts are used for the calibration of the Kalman filter. Thereafter we use the multi-dimensional innovation process to form a *one-dimensional* innovation process used to perform anomaly detection. To obtain this one-dimensional process, we take into account the variance of the residual obtained after running the Kalman filter and we built a new process using the formulas: $e_{new}(t) = e(t)^T V e(t)$, where V is the inverse of the variance of the innovation process, e(t) is the multi-dimensional innovation process, and T denotes the transpose. This space reduction for the residual process allows to perform a good comparison between the PCA-KL expansion method and our method based on Kalman filtering. On the other hand, performing anomaly detection in a single one-dimentional residual vector is more simple and less complex than analysing a multi-dimensional array.

2) Summary of the results.: The first result to show is the ability of our method to track the behavior of link counts for the Abilene data (total byte per unit time) and the behavior of the different TCP and UDP metrics for the SWITCH data, over time. In Figure 2, we show the real and inferred link counts (Abilene) for our model. The evolution of the traffic and estimates are shown for a seven weeks duration for each observation vector. The calibration is applied only once. In Figure 3, we show the results obtained using SWITCH data for the TCP metrics; here too, the calibration is done once a time, for a three weeks of measurements. The results for the UDP metrics are quite similar.

Now we are looking at the compared performances of our two methods, for the Abilene network and also for the SWITCH network. First, to validate our model, one has to find the suitable number of components in the GMM. To do this, we calibrate a GMM with a set of *r* components (r=2,3,4,5...) using the EM algorithm as described above, and the decision to select the best model (i.e the suitable r) is done by analysing the variance performed for each component in the mixture. We compare the results obtained for the different values of r and the model with the lowest variance is chosen.

We have found that the data residual can be organized into three (r=3) distinct clusters for the Abilene data, and into five (r=5) clusters for the SWITCH data. Thereafter *a maximum a posteriori* criterion is used to build, taking as input the clusters, a finite alphabet of symbols, where we perform the *hidden markov modeller*. To train the *hidden markov model*, one must ensure that the different *hidden states* are clearly distinct. This means that one should have a *transition matrix* with higher probabilities in its **main diagonal**. For the datasets we used, we have discovered that an hidden markov model with two (2) states were able to capture the temporal dependencies in the datasets. We show below only the transition and observation matrices for the Abilene case:

$$transmat = \begin{bmatrix} 0.9662 & 0.0338 \\ 0.0162 & 0.9838 \end{bmatrix}$$

$$obsmat = \begin{bmatrix} 0.0113 & 0.0020 & 0.9867 \\ 0.4092 & 0.5904 & 0.0004 \end{bmatrix}.$$

These results show clearly that, the state 1 is composed with almost entirely the symbol$\#3$ and the state 2 is a mixture of the two remaining symbols (1 and 2) with 41% of probability of presence of symbol$\#1$ and 59% for symbol$\#2$. In the philosophy of anomaly detection theory, generally it is assumed, that anomalies might be rare; base on this assumption we can ask this question: if anomalous events occur, do they might come from state $\#1$ or state $\#2$ or both? It is not obvious to answer this question, but we believe that all the changes in the mean of the residual (abrupt changes, lower or higher variations) can be tracked by a combination of symbols yielding in the different states. In other words, one can think that some (but not all) a priori unknown states could be classified as *normal* states and the others as

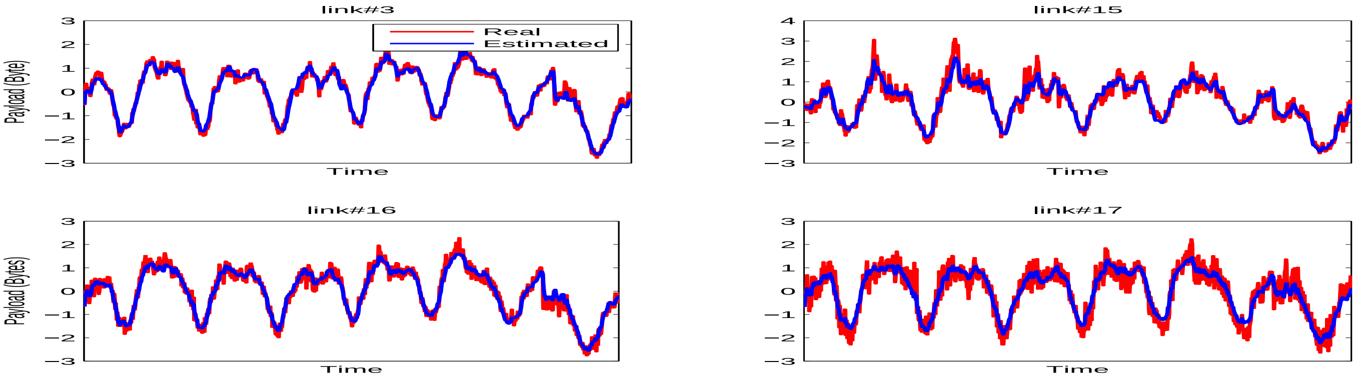


Fig. 2. Real(red) and estimated (blue) link counts obtained using Kalman filter.

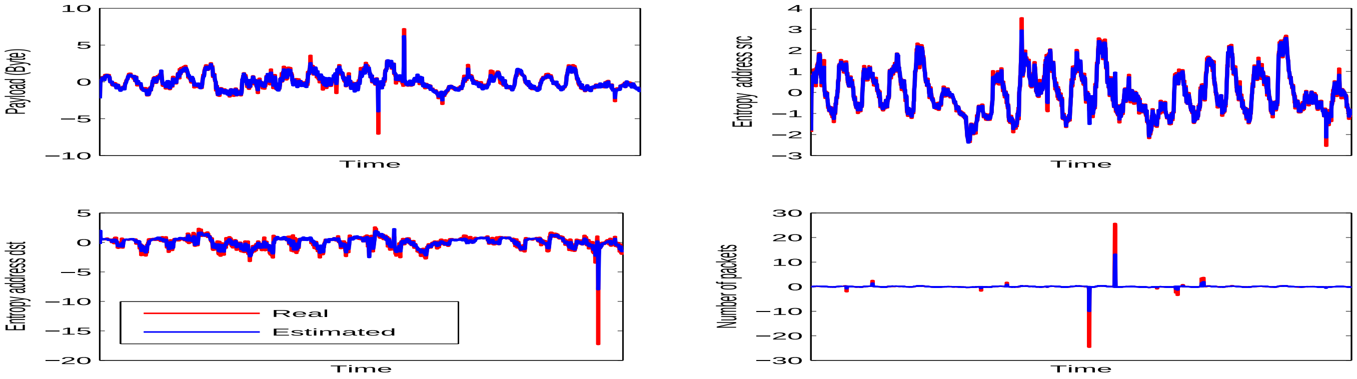


Fig. 3. Real(red) and estimated (blue) TCP metrics obtained using Kalman filter.

abnormal states. If one can find a combination of states to track all the lower, higher or/and abrupt variations in the decision variable, these states will participate in the detection of anomalous events when they occur. And then these states will be etiquetted as abnormal, and one should take attention to them. The remaining states that don't participate in the tracking operation will then be labelled as normal. Recall that PCA and Kalman-based anomaly detection techniques analyze residuals to perform the detection issue. So, if anomalies occur, they will appear in the residual process either in the form of low or high variations or in the form of abrupt changes in the mean of the residual process. We believe that it will be interesting to divide the residual into two parts corresponding to the low and high variations, and ***separately*** apply thresholds for each part. To confirm our intuition, we run the Viterbi algorithm for all the sequences of discrete alphabet and we obtain one unique sequence (path) composed by only the symbols in the state #2.

At this time, we can argue that, if anomalies exist they might be caught either by the two symbols simultaneously, or by one symbol only. One can observe that in Fig. 4, all the variations in the mean of the decision variable can be caught by these two symbols. The top graph corresponds to TCP, the middle to UDP and the bottom graph to Abilene. To track the anomalies, one has just to extract the residual corresponding to the symbol #1 and extract also the part of residual corresponding to symbol #2 and apply to each part thresholds. We reused the methods described in [1] to obtain these thresholds. In addition, in our study we discover that the injected anomalous events never evolve in the cluster with mean closely equal to zero(namely the cluster corresponding to symbol #3).

Another result is about the performance obtained in analyzing the trade-off between false positive and false negative rates. We examine the entire traffic for each method, namely the PCA-KL and the Kalman-based approaches. We then can compute one false positive percentage and one false negative percentage for each threshold configuration scheme. The performances of the two methods on the Abilene and the SWITCH data are depicted in the ROC (Receiver Operating Characteristic) curve of Figures 5, 6 and 7. For all of the results shown on these figures, one can see clearly that the method based on the Kalman fitering techniques, the gaussian mixture model and the hidden markov model performs better than the KL-PCA method, when the temporal correlation range is set to N=1,2,3. For the TCP traffic (SWITCH network), we obtain in Figure 5, 90% of detection rate with 8% of probability of false alarm, for KL expansion with N=2, while we have only 3.5% of false alarm rate with the same detection probability for our new method with N=2, and for N=3, we obtain 2.5% of false positive rate. We observe that, with the newer approach, the false alarm ratio decreases significantly

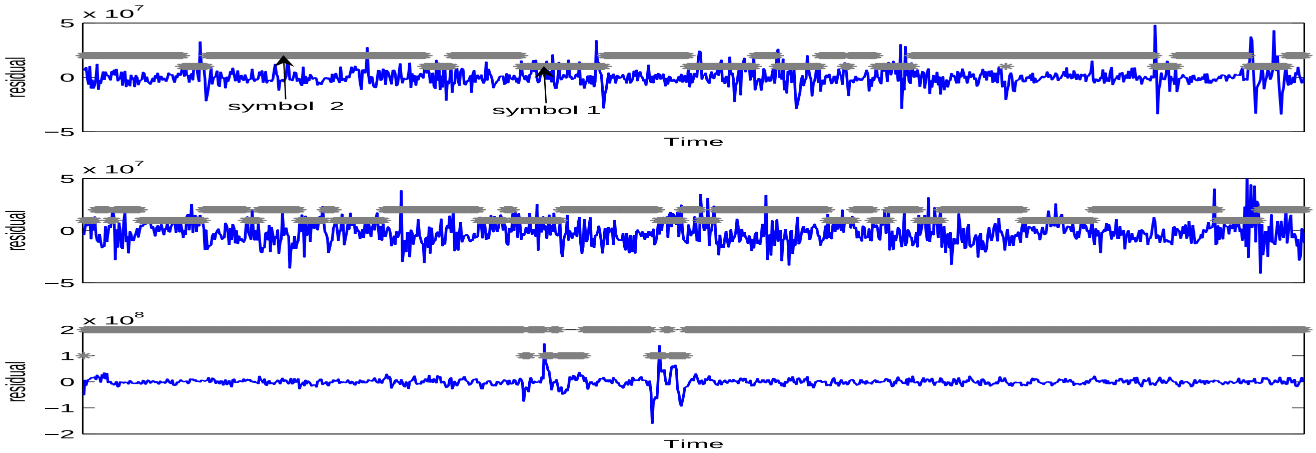


Fig. 4. Tracking the low and high variations in the decision variable by combination of mixing symbols extracted by the Viterbi algorithm.

for all values of N. In the same figure we note that the newer method can achieve 100% (0% of miss detection) of probability of detection with 3.2% of false positive rate while the KL expansion exhibits the same detection rate with 14% of false alarm rate. Also, for the UDP traffic, we obtain in Figure 6, for the KL expansion method for N=2, 96% of detection rate with 7% of false alarm rate versus 2.4% (for N=3) of false alarm rate with the same probability of detection. As for the TCP case, here too, our new method can achieve 100% of detection rate with 2.5% of false positive (N=3) when the KL expansion shows 12% of false positive with the same detection rate (for N=2).

For the Abilene network, we confirm the improvement in the performance of our method above the PCA-KL expansion. One must clearly observe, in Figure 7, that the KL expansion (N=2) shows 100% of probability of detection with 13% of false positive rate while the new method exhibits a false alarm rate of 5% (N=3) with the same detection rate. In summary, the ROCs curve exhibit different points, which can be served as good references to show the high performance we gain above the KL method.

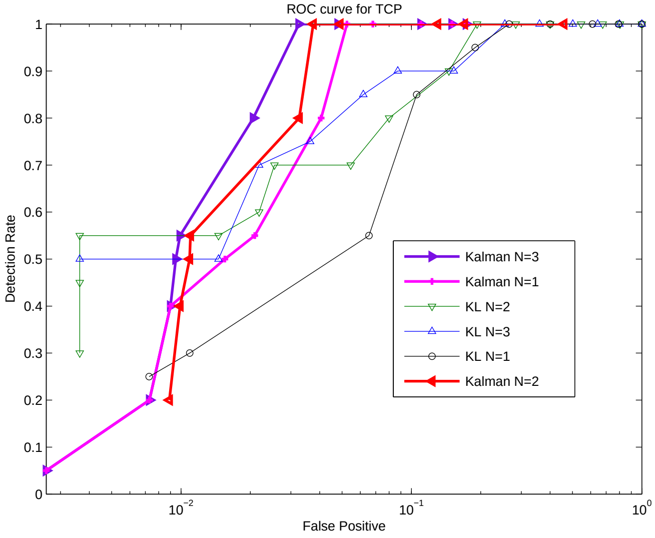


Fig. 5. ROC curve using SWITCH data (TCP)

V. Conclusion

In this work we proposed a profound analysis allowing us to show that an anomaly detection technique combining a panoply of different methodologies and based on Kalman filtering perform better than the PCA technique, which the performance is highly improve by the use of the Karuhen-Loeve expansion. Using a multi-dimensional residual process for each kind of network data, we built a one-dimensional innovation process used as a decision variable, and the comparison of the two schemes is done by analyzing the trade-off between false positive rate and the probability of detection. We found that the use of the Viterbi algorithm as a final tool to make possible to split the one-dimensional decision variable into several subset where we applied different thresholds is an important discovery. It has make possible to track the variability in the residual process using only a combination of symbols yielding in one state. The main drawback of

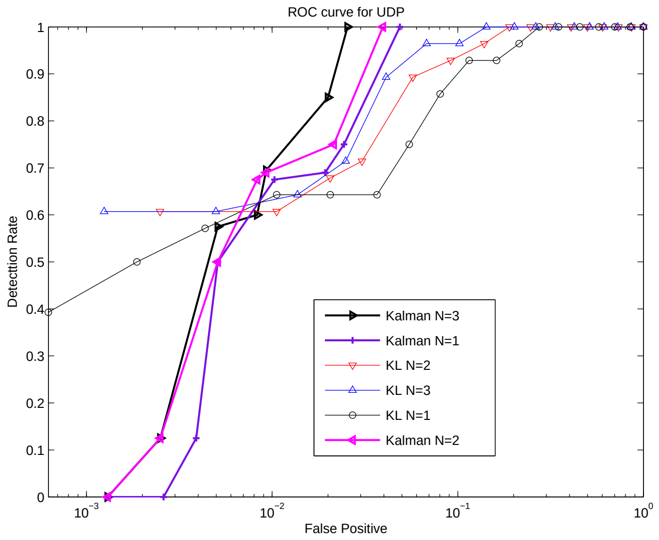


Fig. 6. ROC curve using SWITCH data (UDP).

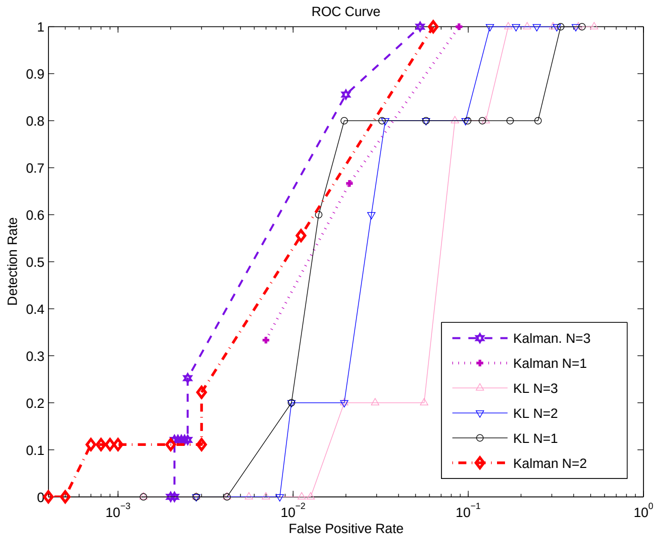


Fig. 7. ROC curve using Abilene data.

this method is the relative complexity introduced by the use of a gaussian mixture model followed by using an HMM, for time dependencies tracking. However, we obtained good performance by reducing considerably the false alarm rate. We had shown in this study that for the two kind of data, the temporal dependencies can be tracked with a hidden markov model with a few number of states (each state being composed by parts of the GMM component). At the other hand, when applying the Viterbi algorithm, we discover that all anomalies are detected in one state with only components 1 and 2, no anomaly were found in the remaining components (number 3 for Abilene and number 3 ,4 and 5 for SWITCH network). The state where no anomaly has been found is the one containing the clusters with mean closely equal to zero. At this moment, one could ask a question about the potential uncertainties about the possibility of these remaining components to capture anomalies in some situations.

References

[1] Brauckhoff, D., Salamatian, K. and May, M.: Applying PCA for Traffic Anomaly Detection: Problems and Solutions. Proceedings IEEE INFOCOM 2009, pp. 2866-2870.

[2] Lakhina, A., Crovella, M. and Diot, C.: Characterization of network-wide traffic anomalies. In Proceedings of the ACM/SIGCOMM Internet Measurement Conference (2004). pp. 201-206.

[3] Soule, A., Salamatian, K. and Taft, N.: Traffic Matrix Tracking using Kalman Filters. ACM LSNI Workshop (2005).

[4] Soule, A., Salamatian, K. and Taft, N.: Combining Filtering and Statistical Methods for Anomaly Detection. USENIX , Association, Internet Measurement Conference (2005). pp. 331344.

[5] Kalman, R. E. and Bucy R. S.: New results in linear filtering and predictions. Trans. ASM E., Series D, Journal of Basic Engineering, Vol. 83 (1961), pp. 95-107.

[6] Kailath, T., Sayed, A. H. and Hassibi B.: Linear Estimation. Prentice Hall, 2000.

[7] Wolverton, C.: On the Linear Smoothing Problem. IEEE Transactions on Automatic Control, vol. 14 Issue:1 pp. 116-117. February 1969.

[8] Raugh, H. E.: Solutions to the linear smoothing problem. IEEE Trans. Automatic Control AC-8 (October 1963) pp. 371372.

[9] Bilmes, J. A.:A Gentle Tutorial of the EM algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models, International Computer Science Institute, Berkeley CA. Technical Report TR-97-021, ICSI, 1997.

[10] McLachlan, G. and Krishnan, T.: The EM Algorithm and Extensions. John Wiley and Sons, New York, 1996.

[11] Dempster, A. P., Laird N. M., and Rubin D. B.: Maximum likelihood from in-complete data via the em algorithm. Journal of the Royal Statistical Society: Series B, 39(1): pp. 138, November 1977.

[12] Rabiner, L., R.,: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proc. IEEE, Vol. 77, No. 2, pp. 257-286, February 1989

[13] Brauckhoff, D., Dimitropoulos, X., Wagner, A. and Salamatian, K. : Anomaly Extraction in Backbone Networks using Association Rules. IMC09, November 46, 2009, pp. 28-34 Chicago, Illinois, USA.

[14] Shumway, R. H. and Stoffer, D. S.: An Approach to Time Series Smoothing And Forecasting Using the EM Algorithm. Journal of Time Series Analysis, vol.3, No 4,pp. 253-264.

[15] Shumway, R. H. and Stoffer, D. S.: Dynamic Linear Models With Switching. Journal of the American Statistical Association, 86, pp. 763-769, 1992.

[16] http://www.stat.pitt.edu/stoffer/tsa2/chap6.htm. 2011

[17] http://www.cs.ubc.ca/ murphyk/Software/HMM/hmm.html. 2011

[18] Ghahramani, Z. and Hinton, G. E.:Parameter Estimation for Linear Dynamical Systems. Technical Report CRG-TR-96-2. February 22, 1996.

[19] http://learning.eng.cam.ac.uk/zoubin/software.html. 2011

Dynamic Firewall Configuration: Security System Architecture and Algebra of the Filtering Rules

Vladimir Zaborovsky[*], **Vladimir Mulukha**[**], **Alexander Silinenko**[***], **Sergey Kupreenko**[****]
St. Petersburg state Polytechnical University
Saint-Petersburg, Russia
vlad@neva.ru[*], vladimir@mail.neva.ru[**], avs@neva.ru[***], ksw@neva.ru[****]

Abstract – **Internet is a global information infrastructure that stores information in the form of distributed digital resources, which have to be protected against unauthorized access. However, the implementations of this protection are far from simple due to dynamic nature of network environment and users activities. We offer a system approach providing a firewall configuration procedure based on new functional model, which includes network monitor, firewall rules generator and the means of rules aggregation. With the help of proposed algebra of filtering rules it is possible to standardize and optimize the dynamic firewall configuration.**

Keywords — dynamic firewall configuration, algebra of filtering rules, access policy, traffic security

I. INTRODUCTION

Internet as a global information infrastructure is used widely for business, education and research. This infrastructure keeps information in the form of distributed digital resources that have to be available for authorized use, while sensitive data should be protected against unauthorized access.

However, the implementations of this protection are far from simple due to the dynamic nature of the network environment state and impossibility of the “security perimeter” organization. Nowadays in virtual networks and clouds, besides securing external network connections, an access control system has to take into account the shared hardware resources and network environment state [1].

The information protection in computer systems has been discussed for almost 50 years. However, the well-known methods of protection of the local data from a remote attacker don’t take into account the specifics of modern computer networks such as [2]:

- territorial distribution and concurrency;
- the dual nature of access control procedures that doesn’t allow to form a "security perimeter" as a static requirement concerning network services;
- non-locality of network resources and characteristics;
- a semantic gap between security policy description and firewall configuration parameters.

Therefore many well-known security solutions of the past have become increasingly inadequate. That is why currently we need deeper and more detailed understanding of the security processes going on in computer networks. That is why we describe below a system approach to provide a firewall configuration procedure based on new functional model, which includes a network monitor, a firewall rules generator and the means of rules aggregation. The main advantage of the proposed approach is the possibility of constructing an algebra of filtering rules, which allows to aggregate and control the firewall configuration that implements the security requirements. The paper is organized as follows: in Section 2, the architecture of dynamic firewall configuration system and the descriptions of its main components are presented. In Section 3, the usage of the above mentioned algebra is described. Section 4 presents the conclusion and the discussion of the overall results.

II. SECURITY SYSTEM ARCHITECTURE

Internet security is a main issue of modern information infrastructure. This infrastructure stores information in the form of distributed digital resources, which have to be protected against unauthorized access. However, the implementations of this statement are far from simple due to the dynamic nature of the network environment and users activity [3]. Below we describe a new approach to configure the security network appliances, that allows an administrator to overcome the semantic gap between security policy requirements and the ability to configure the firewall filtering rules [1]. The architecture of the proposed system is presented in Fig.1.

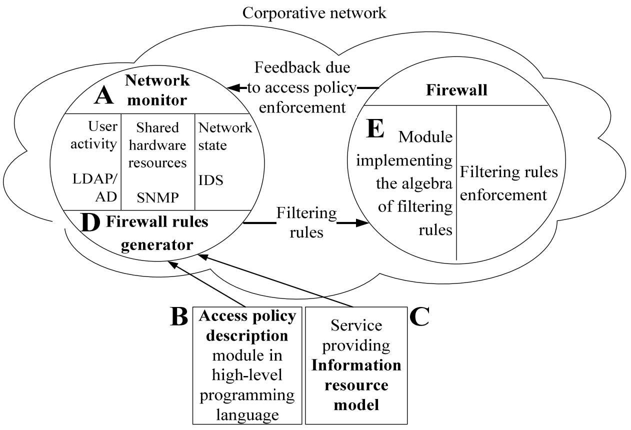

Corporative network
A
Network monitor
User activity
LDAP/ AD
Shared hardware resources
SNMP
Network state
IDS
D
Firewall rules generator
Feedback due to access policy enforcement
Firewall
E
Module implementing the algebra of filtering rules
Filtering rules enforcement
Filtering rules
B
Access policy description module in high-level programming language
C
Service providing Information resource model

Figure 1. Security system architecture

where:

A. *Network monitor*

Network monitor controls the whole system. Network environment state consists of three main parts:

- "User activity" is the information about what computer is currently used by which user. This information can be obtained from Microsoft Active Directory (AD) by means of LDAP protocol.
- "Shared hardware resources" is the information about network infrastructure and shared internal resources that can be described by network environment state vector X_k
- "Network state" is the information about external network channel received from Intrusion Detection Systems (IDS).

B. *Access policy description module*

Filtering rules of a firewall in itself are a formalized expression of an access policy. An access policy may simply specify some restrictions, e.g., "Mr. Black shouldn't work with Youtube" without the refinement of the nature of "Mr. Black" and "work" [4].

There is a common structure of access policy requirements, which uses the notions of subject, action and object. Thus, the informally described requirement "Mr. Black shouldn't work with Youtube" can be formally represented as the combination of the subject "Mr. Black", the action "read", the object "www.youtube.com" and the decision "prohibit". This base can also be augmented by a context, which specifies various additional requirements restricting the cases of rule's application, e.g.: time, previous actions of the subject, attributes' values of the subject or object, etc.

However, access rules, which are based on the notions of subject, action and object are not sufficient alone to implement complex real-world policies. As a result, new approaches have been developed. One of them, Role Based Access Control (RBAC) [5], uses the notion of role. A role replaces a subject in access rules and it's more invariant. Identical roles may be used in multiple information systems while subjects are specific to a particular system. As an example, remember the roles of a system administrator and unprivileged user that are commonly used while configuring various systems. Administrator-subjects (persons) may be being added or removed while an administrator-role and its rules are not changing.

However, every role must be associated with some subjects as only rules with subjects can be finally enforced. During policy specification roles must be created firstly, then access rules must be specified with references to these roles, then the roles must be associated with subjects.

The OrBAC [6] model expands the traditional model of Role Based Access Control. It brings in the new notions of activity, view and abstract context. An activity is to replace an action, i.e., its meaning is analogous to the meaning of a role for a subject. A view is to replace an object. "Entertainment resources" can be an example of view, and "read" or "write" can be examples of an activity. Thus, the notions of role, activity, view and abstract context finally make up an abstract level of an access policy. OrBAC model allows to specify the access rules only on an abstract level using the abstract notions. Those are called the abstract rules. For instance, an abstract rule "user is prohibited to read entertainment resources", where "user" is a role, "read" is an activity, and "entertainment resources" is a view. The rules for subjects, actions and objects are called concrete access rules.

To specify an OrBAC policy, a common language, XACML (eXtensible Access Control Markup Language) was introduced. The language maintains the generality of policy's specification while OrBAC provides additional notions for convenient editing.

C. *Firewall rules generator*

There is a feature common for all firewalls: they execute an access policy. In common representation the main function of access control device (ACD) is to decide whether a subject should be permitted to perform an action with an object. A common access rule "Mr. Black is prohibited to read www.youtube.com".

As was mentioned above, "Mr. Black" is a subject, "HTTP service on www.youtube.com" is an object, and reading is an action. So the configuration of ACD consists of common access rules that reference the subjects, actions and objects.

Although a firewall as an ACD must be configured with common access rules, each implementation uses its own specific configuration language. The language is often hardware dependent, reflecting the features of firewall's internal architecture, and usually being represented by a set of firewall rules. Each rule has references to host addresses and other network configuration parameters. An example of the verbal description of a firewall rule may go as follows:

> Host with IP address 10.0.0.10 is prohibited to establish TCP connections on HTTP port of host with IP address 208.65.153.238.

The main complexity of this approach is to find out how such elementary firewall rules could be obtained from common access rules.

Each firewall vendor reasonably aims at increasing its sales appeal while offering various tools for convenient editing of firewall rules. However, so far the problem of obtaining firewall rules from common access rules is not resolved in general. Moreover, this problem has not been paid much attention to.

The most obvious issue concerning this problem is that additional information beyond access rules is necessary in order to obtain the firewall rules. This information concerns the configuration of network services and the parameters of network protocols that are used for data exchange – "network configuration". In general, it can be stored among the descriptions of subjects, actions and objects. An example:

> Mr.Black: host with IP-address = 10.0.0.10;
>
> www.youtube.com: HTTP service (port 80) on host with IP-address = 208.65.153.238.

Thus, the final firewall rules can be obtained by addition of the object descriptions to the access rules. It should be noted that even for small and especially for medium and large enterprises it is necessary to store and manage the network configuration separately from the security policy. The suggested approach allows us to achieve this goal: the security officer can edit the access rules with reference to real objects while the network administrator can edit the parameters of the network objects [7].

It should also be noted that there is no need to specify any fixed rules regarding association of the network parameters with the objects. For instance, HTTP port may be a parameter of an object or it may be a parameter of an action. A criterion is that the most natural representation of access policy must be achieved.

While generating the rules, the parameters of network objects can be automatically retrieved from various data catalogs. DNS is the best example of a world-wide catalog, which stores the network addresses. Microsoft offers the network administrators the powerful means, Active Directory, to store information about users. Integration with the above mentioned technologies greatly simplifies the work of a security officer as he has only to specify the correct name of an object while forming firewall rules.

D. *Information resource model*

Interaction between subject and object in computer network can be presented as a set of virtual connections. Virtual connections can be classified as technological virtual connections (TVC) or information virtual connections (IVC). (See Fig.2).

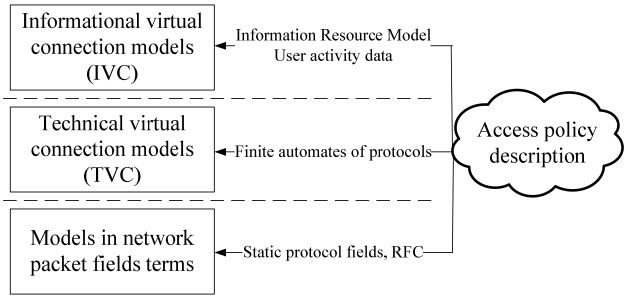

Informational virtual connection models (IVC)
Information Resource Model
User activity data
Technical virtual connection models (TVC)
Finite automates of protocols
Access policy description
Models in network packet fields terms
Static protocol fields, RFC

Figure 2. Layers of access control policies.

To implement the policy of access control, the filtering rules are decomposed in the form of TVC and IVC. These filtering rules can be configured for different levels of the data flow description based on the network packet fields at the levels of channel, transport, and application protocols.

At different layers of access control policy model, the filtering rules have to take into account various parameters of network environment and objects. At the packet filter layer, a firewall considers standards static protocol fields described by RFC. At the layer of TVC, firewall enforces the stateful inspection using finite automata describing states of transport layer protocols. On the upper layer of IVC firewall must consider a-priori information about subject and object of network interaction [8].

As was mentioned above, the information about subject can be obtained from catalog services by LDAP protocol, e.g. Microsoft Active Directory.

According to existing approach [9] a resource model can be presented in:

1) logical aspect – an N-dimensional resource space model [10];
2) representation aspect - the definition based on standard high-level description languages like XML or OWL [11];
3) location aspect – the physical storage model of the resource including resource address.

All these approaches describe the network resource as a whole but don't take into account the specific access control task. Any remote network resource can be fully classified when the connection between this resource and local user would be closed. So it is necessary to control all virtual connections in real time while monitoring traffic for security purpose.

In this paper we propose to implement a special service external to the firewall that would collect, store and renew information about remote network objects. It should automatically create information resource model, describing all informational virtual connections that have to be established to receive this resource. This service should periodically renew information about resource to keep it alive.

Firewall should cooperate with this external service to receive information resource model and enforce access policy requirements.

E. *Algebra of filtering rules*

As was mentioned above, the information security is defined by an access policy that consists of access rules. Each of these rules has a set of attributes; the basic ones among them are identifiers of subject and object and the rights of access from one to another. In TCP/IP-based distributed systems access rules have additional attributes that help to identify flows of packets (sessions) between the client and network application server. Generally these attributes identify the network subjects and objects at different layers of TCP/IP interaction model: MAC-addresses at link layer, IP-addresses at network layer, port numbers at transport layer and some parameters of application protocols.

The access policy in large distributed informational system consists of a huge number of rules that are stored and executed in different access control appliances. The generation of the access policy for such appliances is not very difficult: information must be made available for authorized use, while sensitive data must be protected against unauthorized access. However, its implementation and correct usage is a complex process that is error-prone. Therefore the actual problem of rule generation is representation, analysis and optimization of access policy for large distributed network systems with lots of firewall filtering rules. Below we propose an approach to description, testing and verification of access policy by the means of specific algebra with carrier being the set of firewall filtering rules.

According to proposed approach we define a ring as algebraic structure over set of filtering rules or R [12]. This ring consists of the following operations over the elements of the set R:

1. Commutativity of addition: $\forall a,b\in R\ \ a + b = b + a$.
2. Associativity of addition:
   $\forall a,b,c\in R\ \ a + (b + c) = (a + b) + c$.
3. Zero element of addition:
   $\forall a\in R\ \exists 0\in R\!:\ a + 0 = 0 + a = a$.
4. Inverse element of addition:
   $\forall a\in R\ \exists b\in R\!:\ \ a + b = b + a = 0$.
5. Associativity of multiplication:
   $\forall a,b,c\in R\ \ \ a \times (b \times c) = (a \times b) \times c$.
6. Distributivity: $\forall a,b,c \in R \begin{cases} a\times(b+c) = a\times b + a\times c \\ (b+c)\times a = b\times a + c\times a. \end{cases}$
7. Identity element: $\forall a\in R\ \exists 1\in R\!:\ \ \ a \times 1 = 1 \times a = a$.
8. Commutativity of multiplication:
   $\forall a,b\in R\ \ a \times b = b \times a$.

Let's define the algebra of filtering rules R = <R, Σ>, where R – the set of filtering rules, Σ – the set of possible operations over the elements of R. The set of filtering rules R = $\{r_j,\ j=\overline{1,|R|}\}$ – the carrier set of algebra R. Every rule $r_j = \{X_1,\ldots,X_N,\ A_j,\ B_1,\ldots B_M\}_j$ consists of a vector X_j of parameters, a binary variable A_j and a vector B_j of attributes. $A_i\in\{0,1\}$ is a mandatory attribute that defines the action of access control system over packets; A_j=0 means that packets must be dropped (access denied), A_j=1 means that packets must be passed to receiver (access allowed); $B_{ij}\in DB_j$ is a vector of attribute sets lengths to M (M can be 0). An example of elements of X_j: X_{j1} will be the set of client IP-addresses, and X_{j2} the set of server TCP-ports. The rule attributes B_j define the behavior of access control system that must be applied to corresponding flow of packets (session). The sets of possible values of parameter and attribute vectors are $DX_1,\ldots,DX_N$ and $DB_1,\ldots,DB_M$ in accordance with semantics of every parameter and attribute. For carrier set R the following expression is right (here "×" is the symbol of Cartesian product):

$$R \subset DX_1 \times DX_2 \times \ldots \times DX_N \times DA \times DB_1 \times \ldots \times DB_M$$

The set $\Sigma=\{\varphi_1,\varphi_2\}$ defines the operations that are possible over filtering rules , where φ_1 is the operation of addition, φ_2 is the operation of multiplication.

The operation of addition for filtering rules is defined by the following expressions [12]:

$$r_3 = r_1 + r_2 = \{X_{11}, X_{12},\ldots X_{1N}, A_1, B_{11},\ldots, B_{1M}\} +$$
$$+ \{X_{21}, X_{22},\ldots X_{2N}, A_2, B_{21},\ldots, B_{2M}\}$$
$$r_3 = \begin{cases} \{X_{11} \cup X_{21},\ldots X_{1N} \cup X_{2N}, A_1 \vee A_2, B_{11} \cup \\ \cup B_{21},\ldots, B_{1M} \cup B_{2M}\}, \text{if } A_1 = A_2; \\ \\ \{X_{11}\Delta X_{21},\ldots X_{1N}\Delta X_{2N}, A_1 \wedge \\ \wedge A_2, B_{11}\Delta B_{21},\ldots, B_{1M}\Delta B_{2M}\}, \text{if } A_1 \neq A_2, \end{cases}$$

where A_i is the the attribute "the action of rule", ∪ is the union of sets, Δ is the symmetrical difference of sets, ∨ and ∧ are the logical disjunction and conjunction respectively. In other words the sum of two filtering rules is

4) union of sets of the same name parameters and attributes if the attribute "the action of rule" is equivalent in both rules;
5) symmetrical difference of sets of the same name parameters and attributes if the attribute "the action of rule" is different in summand rules.

The operation of multiplication for filtering rules is defined by following expressions:

$$r_3 = r_1 \times r_2 = \{X_{11}, X_{12},\ldots X_{1N}, A_1, B_{11},\ldots, B_{1M}\} \times$$
$$\times \{X_{21}, X_{22},\ldots X_{2N}, A_2, B_{21},\ldots, B_{2M}\}$$
$$r_3 = \{X_{11} \cap X_{21}, X_{12} \cap X_{22},\ldots X_{1N} \cap X_{2N},$$
$$A_1 \wedge A_2, B_{11} \cap B_{21},\ldots, B_{1M} \cap B_{2M}\},$$

where ∩ – intersection of sets. In other words the product of two filtering rules is intersection of sets of the same name parameters and attributes; attribute "the action of rule" for result rule is a conjunction of corresponding attributes of initial rules.

Zero 0_r, identity 1_r and inverse –r elements of R are specifies by following expressions:

$$0_r = \{\varnothing, \varnothing,\ldots \varnothing, A, \varnothing,\ldots, \varnothing\},\ A = 0$$
$$1_r = \{DX_1, DX_2,\ldots, DX_N, A, DB_1,\ldots, DB_M\},\ A = 1$$
$$-r = \{X_1, X_2,\ldots, X_N, \overline{A}, B_1,\ldots, B_M\},$$

where $\overline{A}$ – logical inversion of A

The described algebra is distributive commutative ring with identity element that means execution of corresponding axioms.

III. Firewall Configuration Using Proposed Algebra

Let's specify the element of set R as r = {X_1, X_2, A} where X_1 – subset of source IP-addresses, DX_1 = [0.0.0.0, 255.255.255.255]; X_2 – subset of destination IP-addresses, DX_2 = [0.0.0.0, 255.255.255.255]; A – attribute "the action of rule", DA = {0,1}, 0 denies access, 1 allows access; let M=0, so there would be no B_{ij} attributes. It is necessary to define the full and consistent access policy that allows establishing of sessions from Internal network (see schema on Fig.3, a) to External subnetworks 0.0.0.0 – 9.255.255.255, 20.0.0.0 – 49.255.255.255 and from External subnetworks 40.0.0.0 – 49.255.255.255 to the whole Internal network.

For this task a convenient method of representation of access policy is 2-dimensional space x_1x_2. Every point of this space is specified by the coordinates (x_1, x_2). The set of points (x_1, x_2) is specified by Cartesian product of sets DX_1 and DX_2 (see on Fig.3, b).

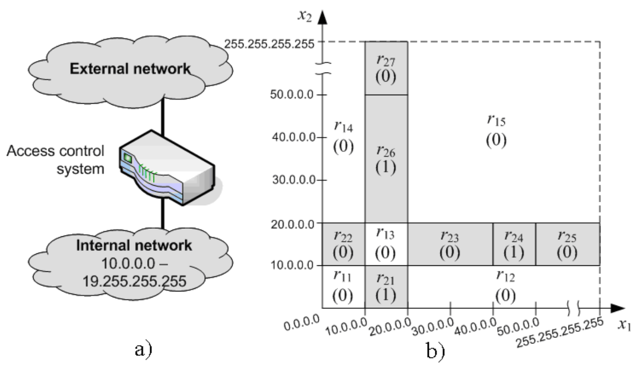


Figure 3. Access control system based on firewall (a) and access policy as a space of parameters (b)

Definition 1. The access policy is full if filtering rules specify the whole of space of parameters:

$$\forall x_1 \in DX_1, x_2 \in DX_2 ..., x_N \in DX_N \ (x_1, x_2, ..., x_N) \in$$
$$\in \bigcup_{i=1}^{|R|}(X_{i1}, X_{i2}, ..., X_{iN})$$

Definition 2. The access policy is consistent if any point of space of parameters belongs only to one filtering rule:

$$\bigcap_{i=1}^{|R|}(X_{i1}, X_{i2}, ..., X_{iN}) = \varnothing \ .$$

Obviously that for schema on Fig. 3 there are some forbidden areas that are incorrect from the point of view of IP-network functionality. The following rules describe such areas (in Fig. 9,b these areas are colored in white):

r_{11} = {0.0.0.0 – 9.255.255.255; 0.0.0.0 – 9.255.255.255; 0};
r_{12} = {20.0.0.0 – 255.255.255.255; 0.0.0.0 – 9.255.255.255; 0};
r_{13} = {10.0.0.0 – 19.255.255.255; 10.0.0.0 – 19.255.255.255; 0};
r_{14} = {0.0.0.0 – 9.255.255.255; 20.0.0.0 – 255.255.255.255; 0};
r_{15} = {20.0.0.0 – 255.255.255.255; 20.0.0.0 – 255.255.255.255; 0}.

Let us optimize this set of rules by applying the algebra's addition operation to rules r_{11} and r_{14}, r_{12} and r_{15}:

r_{17} = r_{11} + r_{14} = {0.0.0.0 – 9.255.255.255; 0.0.0.0 – 9.255.255.255, 20.0.0.0 – 255.255.255.255; 0};
r_{18} = r_{12} + r_{15} = {20.0.0.0 – 255.255.255.255; 0.0.0.0 – 9.255.255.255, 20.0.0.0 – 255.255.255.255; 0}.

For other areas (colored gray in Fig. 3,b) it is necessary to specify the filtering rules according to the task conditions:

r_{21} = {10.0.0.0 – 19.255.255.255; 0.0.0.0 – 9.255.255.255; 1};
r_{22} = {0.0.0.0 – 9.255.255.255; 10.0.0.0 – 19.255.255.255; 0};
r_{23} = {20.0.0.0 – 39.255.255.255; 10.0.0.0 – 19.255.255.255; 0};
r_{24} = {40.0.0.0 – 49.255.255.255; 10.0.0.0 – 19.255.255.255; 1};
r_{25} = {50.0.0.0 – 255.255.255.255; 10.0.0.0 – 19.255.255.255; 0};
r_{26} = {10.0.0.0 – 19.255.255.255; 20.0.0.0 – 49.255.255.255; 1};
r_{27} = {10.0.0.0 – 19.255.255.255; 50.0.0.0 – 255.255.255.255; 0}.

These rules may be optimized also by applying of algebra's addition operation:

r_{28} = r_{21} + r_{26} = {10.0.0.0 – 19.255.255.255; 0.0.0.0 – 9.255.255.255, 20.0.0.0 – 49.255.255.255; 1};
r_{29} = r_{22} + r_{23} = {0.0.0.0 – 9.255.255.255, 20.0.0.0 – 39.255.255.255; 10.0.0.0 – 19.255.255.255; 0}.

As a result the access policy describes by following filtering rule set:

R = {r_{13}, r_{17}, r_{18}, r_{24}, r_{25}, r_{27}, r_{28}, r_{29}}.

The dimension of R is the main attribute that describes firewall performance characteristics. Usage of the algebraic operations of addition and multiplication allows us to reduce dimensionality of R and thus to increase the firewall performance while fulfilling requirements of the specific security policy [12]. However the correctness of each rule depends on an environment condition, which can vary in real time. Therefore static description of access policy by means of proposed algebra is not enough and according to the telematics approach it is necessary to consider an environment condition with statistical parameters. Development of randomized model of the network environment considering these requirements, allows us to increase accuracy of the description of an access policy by means of filtering rules.

IV. CONCLUSION.

1. Each firewall is required to work in compliance with a security policy, user activities and network configuration. Policy requirements cannot be considered separately from methodology of proper firewall configuration and specified security characteristics. Based on OrBAC model it is possible to translate high-level abstract security requirements to low-level firewall configuration.

2. Firewall configuration can be largely automated based on specifying high-level access rules and parameters of corporate DNS, AD/LDAP, SNMP and IDS services. Proposed system architecture can be easily implemented due to consideration of role-based information access models and characteristics of specific firewalls.

3. Proposed algebra of filtering rules is a new mathematical description of access policy and a formal tool for firewall configuration. The system approach provides possibility to prove fullness and consistency of an access policy. The proposed algebra is the base of optimization of the set of filtering rules and of the design of dynamic firewall configuration.

REFERENCES

[1] V. Mulukha. Access Control in Computer Networks Based on Classification and Priority Queuing of the Packet Traffic, PhD. Thesis 05.13.19, SPbSPU, Russia, 2010

[2] V. Zaborovsky and V. Mulukha. Access Control in a Form of Active Queuing Management in Congested Network Environment // Proceedings of the Tenth International Conference on Networks, ICN 2011 pp.12-17.

[3] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia. 2010. A view of cloud computing. Commun. ACM 53, 4 (April 2010), pp.50-58.

[4] A. Titov and V. Zaborovsky. Firewall Configuration Based on Specifications of Access Policy and Network Environment // Proceedings of the 2010 International Conference on Security & Management. July 12-15, 2010.

[5] D.F. Ferraiolo and D.R. Kuhn. Role-Based Access Control. 15th National Computer Security Conference. (October 1992), pp. 554–563.

Copyright (c) IARIA, 2011. ISBN: 978-1-61208-141-0

(http://csrc.nist.gov/groups/SNS/rbac/documents/ferraiolo-kuhn-92.pdf)

[6] Organisation-based access control (OrBAC.org). Available: http://orbac.org/index.php?page=orbac&lang=en

[7] V. Zaborovsky and A. Titov. Specialized Solutions for Improvement of Firewall Performance and Conformity to Security Policy // Proceedings of the 2009 International Conference on Security & Management. v. 2. pp. 603-608. July 13-16, 2009.

[8] V. Zaborovsky, A. Lukashin, and S. Kupreenko Multicore platform for high performance firewalls. High performance systems // Materials of VII International conference – Taganrog, Russia.

[9] H. Zhuge, The Web Resource Space Model, Berlin, Germany: Springer-Verlag, 2007

[10] H. Zhuge, “Resource Space Grid: Model, Method and Platform,” Concurrency and Computation: Practice and Experience, vol. 16, no. 14, pp. 1385-1413, 2004

[11] D. Martin, M. Burstein, J. Hobbs, O. Lassila. et al. (November 2004) “OWL-S: Semantic Markup for Web Services,” [Online]. Available: http://www.w3.org/Submission/OWL-S/.

[12] A. Silinenko. Access control in IP networks based on virtual connection state models: PhD. Thesis 05.13.19: / SPbSTU, Russia, 2010.

Open IDM 2.0 Framework: a Unifying Gateway for Interoperable Identity Management

Brahim En-Nasry
Information Security Research Team
Ecole Nationale Supérieure d'Informatique et d'Analyse des Systèmes, ENSIAS, Université Mohammed V-Souissi
Rabat, Morocco
ennasri@ensias.ma

Mohamed Dafir Ech-Cherif El Kettani
Information Security Research Team
Ecole Nationale Supérieure d'Informatique et d'Analyse des Systèmes, ENSIAS, Université Mohammed V-Souissi
Rabat, Morocco
dafir@ensias.ma

Abstract—**Today, as Internet has brought individuals and organisms within easy discovery and reach of each other, the role of identity has taken on great importance in social interactions, commercial transactions and governance. Interoperability as the foundation and key enabler for cross-domain Identity Management is still a complex challenge to achieve. However, efforts to build a unified framework for interoperability between Identity Management systems, that maps to different contexts such as business, government, real and virtual communities, will bring the breath solution that we all need. We investigate this issue from stakeholder's perspectives and across many technological initiatives approaches. Moreover, we also discuss advantages and drawbacks of some Identity Management systems with respect to interoperability standards. Finally, we highlight the interoperability requirements towards a unified model and motivate the need of a mature model for Identity Management and interoperability.**

Keywords - Security; Identity Management; interoperability; framework.

I. Introduction

Digital Identity Management tools are designed to ensure effective use of the multiple facets of identity and identification data associated to individuals in Internet transactions. Digital identity is multifaceted and also context sensitive. It contains strong identifiers that uniquely describe a person, as well as non-exhaustive lists of other attributes ranged from weak to strong and from temporal to persistent: relationships, reputation, preferences, etc. The early purpose of Identity Management Systems is to facilitate the establishment of security mechanisms. The ultimate goal is the control of access to assets, by supplying access control systems with reliable, up to date and consistent information, while granting a tradeoff between security, usability and privacy. In other terms, within an organization, an Identity Management System integrates many processes (authentication, authorization, accounting, identification and personalization) to interact with central repositories. In open environments, the implementation of a Digital Identity System differs depending on the approach adopted to meet trust requirements and expectations of various stakeholders.

The former Identity Management models, named “in silo“, have been developed in closed environments, and are running with proprietary systems, without any possibility for interaction with each other. Within the rise in electronic data exchange in various contexts (such as business and consumer applications, Web 2.0, tele-declaration, etc.), it becomes necessary to bridge different Identity Management systems and manage different identities islets scattered in various accounts. Identity Management interoperability for networked and distributed applications continues to present several unique challenges for users and developers.

To take a look at the state of art, existing models are discussed in this paper, varying from centralized, federated to user-centric ones, reflecting their adaptation to Internet, through the evolution of service concept, and technologies to which they are associated. Each model requires some prerequisites and starts from a specific background [1].

Today, the storage and use of credentials (government issued credentials, credit card number, address, birth date, etc.) are controlled by the entity in possession of those credentials but in a confused manner. Without the distribution of defined roles and the delineation of the responsibility of each entity in all processes dealing with identity, interoperability can lead to the proliferation of solutions that spend the same problems.

A high level of interoperability can be reached if all entities in different domains can communicate to exchange identification information via a secure channel that permits strong and flexible authentication. This level must be reinforced by policies that define restricted roles and limits assigned to stakeholders. It is then useful to think about interoperability from a stakeholder perspective, including “user”, “relying parties”, and “ID providers” perspectives.

This paper is organized in 5 sections: after this introduction, Section 2 defines the global context of our study in terms of interoperability in current IDM approaches, especially under the scope of identity 2.0, in order to prepare the groundwork for an open interoperable IdM framework to access online services. Section 3 presents current approaches related to IDM existing frameworks. Section 4 proposes the model, consisting in a unified interoperable framework that will serve as a unifying gateway between all IdM solutions. Section 5 serves as a conclusion.

ISBN: 978-1-61208-141-0

II. Interoperability in Current IDM approaches

In this section, we will discuss this problematic through the analysis of current Identity Management solutions and we will see if they can allow a certain level of interoperability.

But let us first give some precisions about interoperability from a stakeholder perspective, including "user perspective", "relying parties perspective", and "ID providers perspective":

- *User perspective:* In context of exchange between different systems, privacy of identity attributes is thus a crucial problem. The privacy paradigm is that individuals will be able to protect their privacy if their information can only be collected, used, and disclosed with their consent. Thus, users would like to choose their Identity Provider so that they can efficiently define attributes of their identity and securely control how these attributes are gathered, stored, shared among multiple service providers, with at least some level of portability;
- *Relying parties perspective:* RPs aim to cooperate each other, exchange accurate, up to date, and relevant information about individuals from any source to propose personalized services to better serve users from wide communities. At the same time, they want to delegate some identity administration tasks to IDPs. This new trend extends the security perimeter. Hence, they need to build trust relationships, protect their users and also their assets.
- *ID providers perspective:* IDPs want to provide identity as a service to users and relying parties and reinforce their positions as safe guardians of identity;

Current Identity Management solutions can be classified into 3 approaches:

A. Centralized approaches

Most of Identity Management systems deployed early in the Internet were client/server-based and called *silo models*. A single entity which operates the Identity Management system can be either the service provider acting as both service provider and identity provider or a trusted identity provider mixed up with service provider controlling together the name space for a specific service domain, and allocating identifiers to users. A user gets separate unique identifiers from each service/identifier provider he transacts with.

This approach might provide simple Identity Management for service providers, but is rapidly becoming cumbersome for users who will have to remember many identifiers and credentials associated to each service.

This approach has several drawbacks because the IdP not only becomes a single point of failure, but it may also not be trusted. The silo model is not interoperable and many of its aspects present serious deficiencies.

B. Federated approaches

A federated Identity Management system consists in software components and protocols that manage the whole life cycle of identities. In such a model, we assume that user data are stored at various locations on the Internet. This model supports many identity providers with no centralized control point. The distributed storage locations linked together are also easily shared. A federated model is a group of sites or systems that establish a trust agreement where each entity trusts identification data coming from others.

Federation facilitates the use of user attributes across trust boundaries as this architecture gives the user the illusion that a single identifier authority exists. Even if the user has many identifiers, he doesn't need to know them all.

With Single Sign On (SSO) mechanism, users authenticate themselves once by a federation member they trust, so they can navigate to any of the member service providers and be granted appropriate permissions based on their unique identifier shared among multiple service providers. The process of establishing a shared identifier for each user is often referred to as federating user's identities.

The level of interoperability within a federation is often fairly high, as they work better with seamless data transfer. The openness of a federation to new relying parties is more variable and depends on trust agreements, rules and the technology choices made by its designers.

Having different types of institutions as part of the federation (each with its own policies regarding its own users) makes it difficult for administrators to properly determine the categories of users allowed to access to each resource: Scalability is a potential problem unless the federation is relatively homogeneous.

Federations can cooperate with each other since they start to identify partners beyond their initial offerings. In this case, offerings to end users are improved substantially: but if the technology and rules used by federations are different, it can be difficult to implement cross-federation initiatives. A base level of interoperability is needed in order to broaden service availability provided by federations.

However, privacy protection is a serious problem, as it is difficult to know to which extent and under which circumstances federations driven by for-profit corporations will offer benefits to consumers. No one can grant if a company that holds customer data will not sell access to user databases to other online companies. A wide variety of federated systems are possible, so the consequences for both corporations and consumers of federation in general are uncertain. Relevant proposals, such as Liberty Alliance, Shibboleth, and WS-Federation, are based on the notion of federated identity. In Liberty Alliance, a federation consists in a circle of trust including service providers (SPs) and IdPs with mutual trust relationships. The circle of trust enables single sign-on (SSO) across different SPs' websites. When an SP requests user authentication, the IdP authenticates the user and then issues an authentication assertion. The SP validates the assertion and determines whether to accept it. The unique first authentication of a user is enough to sign on to other service sites.

C. *User-centric approaches*

User-centric models [2] are driven by privacy concerns and aim to leave control with the user as to initiate or approve any transfer of personal information before it takes place, either directly or through a mediator with predefined rules for authorization. A user-centric model must have a basic level of interoperability in order for an individual to use their digital ID for multiple services.

Though data can still be stored with a relying party once data is given in a transaction, this model allows individuals to disclose minimal information. The information provided by the user can be easily checked with the Identity Provider, causing greater accuracy and less potential for fraud.

A major drawback of the user-centric model is its complexity. There are significant technical challenges related to creating a system that sufficiently satisfies all parties, so that they actually use it. One should not forget also social challenges in educating business owners and users. Most web businesses are accustomed to asking users to provide identifying information – often more than strictly necessary – and users are used to providing it, and setting up a username and password for each site. This situation is familiar, if cumbersome.

In contrast, a user-centric model requires both user and relying party to develop relationships with one or more trusted Identity Providers and possibly install and learn new software. This attitude could be a barrier to widespread adoption. Furthermore, businesses that currently collect identifying data may be reluctant to give up control over their customers' data, by using it for marketing or selling it to direct marketers.

Interoperability between user-centric and non user-centric systems is not always possible due to the preconditioned circle of trust and trust agreement requirements.

III. Current existing frameworks to interoperability

Many initiatives are currently under work to develop the Internet-based Identity Management services called Identity 2.0. They are based on the concept of user-centric Identity Management, supporting data mapping, authentication and identity verification protocols while protecting privacy by letting user with a margin freedom to express her consent and control her identities when doing Internet-based transactions. Until now, there are two categories of Identity 2.0 initiatives: URL-based and Infocard-based. The main difference among such proposals is the protocol they use to verify user identity. In CardSpace, the user selects from a set of information cards representing the digital identities that satisfy a relying party's (RP's) policy. The identity provider (IdP) that issued the card releases to the user a security token, encoding claims corresponding to the selected information card. The user then passes the card and the token to the RP. Credentica and CardSpace support similar identity verification protocols: The RP verifies the user's identity based on an IdP issued ID token, encoding claims about the identity presented by the user to the RP.

Contrariwise, OpenID is a URL-based protocol and when users access an RP's website, they provide an OpenID that is the URL of a webpage listing their IdPs. The RP selects an IdP, and the browser is redirected to the IdP's webpage.

If the IdP successfully verifies the user's identity, the browser is redirected to the designated return page on the RP website, along with an assertion of user authentication.

We should not forget that the frameworks listed below are unified solution to interoperability. They just propose initiatives to solve some aspects of interoperability within Identity Management approaches. Higgins is a model that will be useful, if modified to become a powerful bridge between many models.

A. *XRI*

The Organization for the Advancement of Structured Information Standards (OASIS) has developed a unified identifier scheme to help companies tackle today's rampant Identity Management interoperability problems.

The Extensible Resource Identifier [3] (XRI) specification establishes an interoperable framework for expressing, resolving and establishing equivalence between identifiers of any kind for any resource type, including people, applications, network devices and corporate assets. XRIs build on the ubiquitous Uniform Resource Identifier (URI) and Internationalized Resource Identifier (IRI) standards - widely used by Identity Management solutions - by defining standard ways to express characteristics such as type, language and date. The lightweight HTTP- and XML-based XRI resolution framework lets a consuming application quickly and easily discover metadata related to resources, such as an alternative synonym identifier that works better in the application's local Identity Management system.

Metadata isn't limited to alternative identifiers. Imagine that an XRI-identified resource is a technical manual, available as a PDF or Word document and retrievable from a variety of mirrored network locations via various protocols.

In a broad sense, the manual is the same document irrespective of where it is located, how it is retrieved or in which format it is represented. XRIs are ideally suited for identifying resources at this level of abstraction because the resolution process lets the consuming application choose the best network location, retrieval method and file format for its needs from the available options.

Like URIs, XRIs are composed of an authority portion and a path portion. XRI resolution converts the authority portion and the path portion of an XRI to an XML document called an XRI Descriptor. The XRI Descriptor describes the identified resource and the means by which the digital representation of the resource can be retrieved.

By providing an additional level of in direction away from concrete instances of a resource, XRIs provide a permanent, unbreakable reference on which stable business relationships can be based.

B. SAML

The initial versions of SAML [4] v1.0 and v1.1 define protocols for SSO, delegated administration, and policy management. The most recent version is SAML 2.0. It is now the most common language to the majority of platforms that need to change the unified secure assertion. It is very useful and simple because it is based on XML.

This protocol enables interoperability between security systems (browser SSO, Web services security, and so on). Other aspects of federated Identity Management as permission-based attribute sharing are also supported.

C. Identity Web Services Framework

In the second phase, the specifications offer enhancing identity federation and interoperable identity-based Web services. This body is referred to as the *Identity Web Services Framework* (ID-WSF). This framework involves support of the new open standard such as WS-Security developed in OASIS. ID-WSF is a platform for the discovery and invocation of identity services - Web services associated with a given identity. In the typical ID-WSF use case, after a user authenticates to an IdP, this fact is asserted to an SP through SAML-based SSO. Embedded within the assertion is information that the SP can optionally use to discover and invoke potentially numerous and distributed identity services for that user. Some scenarios present an unacceptable privacy risk because they suggest the possibility of a user's identity being exchanged without user's consent or even knowledge. ID-WSF has a number of policy mechanisms to guard against this risk. But ultimately, it is worth noting that many identity transactions (automated bill payments) already occur without user's active real-time consent (users appreciate this efficiency and convenience).

As a standard, SAML supports a standard syntax for the representation of assertions about identity attributes and IdP authentications but does not provide an identity verification protocol. SAML is important in our approach as it facilitates the exchange of identity tuples and mapping certificates across domains in a federation.

To build additional interoperable identity services such as registration services, contacts, calendar, geolocation services, and alert services, it's envisaged to use ID-WSF. This specification is referred to as the *Identity Services Interface Specification* (ID-SIS).

D. Shibboleth

Shibboleth [5] allowed interoperation between academic institutions by developing architectures, policy structure, practical technologies, and open-source implementation.

E. OpenID 2.0

OpenID authentication 2.0 [6] is becoming an open platform that supports both URL and XRI user identifiers. In addition, it would like to be modular, lightweight, and user oriented. Indeed, OpenID auth. 2.0 allows users to choose, control and manage their identity addresses. Moreover, the user chooses his identity provider and has a large interoperability of his identity and can dynamically use new services that stand out, such as attribute verification and reputation, without any loss of features. No software is required on the user's side because the user interacts directly with the identity provider's site. OpenID Authentication provides a way to prove that an end user controls an Identifier. It does this without the Relying Party needing access to end user credentials such as a password or to other sensitive information such as an email address. OpenID is decentralized. No central authority must approve or register Relying Parties or OpenID Providers. An end user can freely choose which OpenID Provider to use, and can preserve their Identifier if they switch OpenID Providers. OpenID Authentication is designed to provide a base service to enable portable, user-centric digital identity in a free and decentralized manner. It uses only standard HTTP(S) requests and responses.

The exchange of profile information, or the exchange of other information, can be addressed through additional service types built on top of protocol to create a framework.

F. InfoCards

CardSpace is Microsoft's code name for this new technology that tackles the problem of managing and disclosing identity information [7]. CardSpace implements core of identity meta-system, using open standard protocols to negotiate, request, and broker identity information between trusted IdPs and SPs. It is a technology that helps developers integrate consistent identity infrastructure into applications, Web sites, and Web services.

G. Higgins

Eclipse Foundation [8] has developed an open framework built around info-cards, to enable user's interaction with multiple authentication protocols. This framework allows software developers to use identity cards as a form of authentication to integrate and leverage multiple identification protocols within their applications. Three components provided by Higgins for enabling information-card authentication:

1) *Identity selector applications:* end-users can use to sign-in to web sites and systems that are compatible with Info-Card-based authentication.

2) *Complete code:* necessary for Identity Provider web services as well as for the "relying party", it enables websites and systems to be information card- and Open Id-compatible. Software developers can incorporate this code into their applications to make it easier for their users to login to their site. There are currently two web-site developer solutions available (STS, IdP-for, WS-Trust and SAML2 IdP –for SAML2)

Higgins Global Graph (HGG) data model and the Higgins Identity Attribute Service (IdAS): Developers now have a framework that provides an interoperability and portability abstraction layer over existing "silos" of identity data. The HGG/IdAS layer of Higgins offers integration opportunities between several identification protocols such as OpenID, WS-Trust, SAML, and LDAP.

IV. "Open IDM": a Unified Interoperable Framework

A. *Motivation*

To reach a basic level on interoperability, federated identity solution must, by its very nature, be standards-based. The key underlying standard for federated identity is SAML. SAML is the most mature and widely deployed identity federation protocol today and offers the highest potential for interoperability with federation partners. The latest version, SAML 2.0, marks the convergence of the SAML, Liberty ID-FF, and Shibboleth specifications into a single unified standard. A federated model must be user centric to allow the user to maintain control over its identity.

Distributed solutions are also interoperable and user centric if they use the same approach and technology. Level two is possible if at least two approaches are interoperable.

B. *Proposed Unified framework*

Interoperability between multiple Digital Identity systems has become an important and complex issue. Even if many initiatives exist, high level interoperability is far from being achieved outside circles of trust. Existing models are not clearly interoperable and are deficient in unifying standard-based protocols due to the conflicting requirements for each approach. In untrusted domains, privacy concerns and user's attribute controls are fundamental when offering identity to users; on the other hand, the same users ask for flexible access through homogenous user friendly interfaces. Taken as a single solution, it may not exhaust all possible solutions to the issue, but when bridging all solutions, this approach will federate all efforts currently under development.

From a logical point of view, high level of interoperability will be assured by a unified framework, an open user-centric bridge managed by a new entity so called Master Identity Provider (Figure 1). This framework serves as a unifying gateway between all existing models. In one side, as industry and other organizations continue to introduce capabilities and standards guided essentially by the approach adopted and suitable for only one specific community within a specific context, but not for all possible use cases. In other side, all identity problems come from the lack of visibility towards IDPs that work separately without any well-defined relationships between them.

Thus, a Master Identity Provider acting as a Root Authority to identities, will federate all relationships between IDPs in order to build the identity, during enrolment processes, piece by piece starting with the root while avoiding any duplication and any unnecessary information which represents nothing.

This unified framework serves as a metasystem to bridge all scenarios of Digital IDM Systems, which must be interoperable. This new arrangement of accepted standards enables decentralized identity infrastructure to work together as a single Identity Management system, including:

1) *External standards:* such as XML, SAML, etc.

2) *Open Software standards:* such as java or Linux

3) Hardware standards: they support interoperability

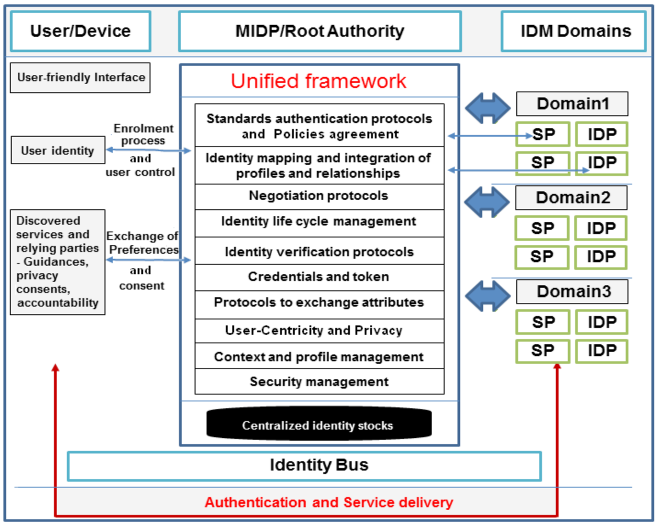

User/Device
MIDP/Root Authority
IDM Domains
User-friendly Interface
User identity
Enrolment process and user control
Discovered services and relying parties - Guidances, privacy consents, accountability
Exchange of Preferences and consent
Unified framework
Standards authentication protocols and Policies agreement
Identity mapping and integration of profiles and relationships
Negotiation protocols
Identity life cycle management
Identity verification protocols
Credentials and token
Protocols to exchange attributes
User-Centricity and Privacy
Context and profile management
Security management
Centralized identity stocks
Domain1
SP
IDP
SP
IDP
Domain2
SP
IDP
SP
IDP
Domain3
SP
IDP
SP
IDP
Identity Bus
Authentication and Service delivery

Figure 1. Architecture of the proposed framework

C. *"Open IDM 2.0" Framework modules*

1) *Policies agreement module:* it includes Service Provisioning Policies, Service Provider Privacy Policies, Privacy Preferences and Federation Agreement Policies.

2) *Identity mapping and integration of profiles and relationships module:* A first step toward achieving interoperability is the adoption of a standard to describe assertions and identity profiles.

3) *Negotiation protocols module:* it integrates all negotiation aspects especially trust negotiation when entities are not previously known to each other. Before meaningful interaction starts, a base level of trust must be established.

4) *Life cycle management module:* All federated identity solutions must provide management capabilities to perform required tasks to create, provision, manage and monitor it.

5) *Security module:* manages all security concerns.

6) *Heterogeneous identity verification:* heterogeneity among identity verification protocols and naming, especially in the context of the clients' identity verification process. It specifies such a set of identity attributes. If clients use names for the identity attributes from different vocabularies, after a client request for a resource or a service from an RP, they may not understand the adequate identity attributes.

7) *Security Credentials and Token module*: it supports different identity tokens and related encryption algorithms.

8) *Protocols to exchange attributes:* this module enables exchange of attributes in a cryptographic way

9) *User-Centric and Privacy module:* Users should have the maximum control possible over the release of their identity attributes. They should state under which conditions these attributes can be disclosed.

10) *Context and profile manager:* identity and context are closely related; during interoperability analysis, context issue must provide consistent experience across contexts.

11) *Centralized identity stocks:* playing the role of a repository in order to concentrate all IdM resources.

The proposed unified framework interacts with the global IDM architecture, through the following elements:

- *User Identity:* relates to a person, device or application, in order to define identity strength.
- *Identity Bus:* supports interoperability between varieties of IDM technologies available from different vendors, an Identity Bus that will provide interoperability functionalities is necessary.
- *Consistent user interface:* Lack of usability will make the control of identity by user almost impossible to take place. The model must facilitate the developer with adequate support for implementing usability through a user interface.

D. Analyzing the Framework

Trustworthiness of an identity depends on the initial enrollment process, the security token being issued, the level of collaboration and the depth of the relationship between entities. As identity providers and relying parties in current ecosystems don't directly communicate during enrollment process, identity islands remain as data silos between each other. This framework as a harmonized identity metasystem aims to solve the problem of consolidation of distributed identity and provide secure, privacy enhanced and seamless experiences. Reasonable-diligence of services needs to validate the identity of individuals or organizations requesting credentials that will enable them to participate in information exchanges. Trust will convey through inter-domain exchange of identity attributes as well as any useful information and policies to collaborate in tracking down all identity transactions.

To consolidate identity, we acknowledge that an Open IDM framework should integrate -but should not be limited to- the two main following processes :

Enrolment process required by a service*:* in Figure 2, (1) user contacts SP (2) user is redirected to the IDP which SP trusts. (3) SP specifies to his IDP what attributes it needs. (4) IDP contacts his direct MIDP where user is referenced and user is redirected to MIDP to be identified. (5) If the user is really referenced, a profile negotiated with user is generated with respect to the principle of minimal disclosure. (6) MIDP provides IDP with minimal attributes and new credentials are issued.

Access to service: a user attempts to gain authorization to do something online. User contacts MIDP to know if service is referenced and user already enrolled. Authentication is activated towards IDP with adequate protocol and credential.

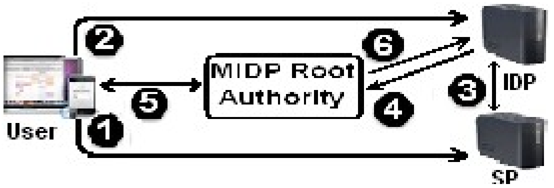


Figure 2. Enrollment process

This proposed framework presents many advantages: True interoperability will be possible with the open gateway serving as interface to standard protocols.

- Technologies like OpenID, SAML, Liberty ID-FF and WS-Trust should be supported. Data format and authentication systems at endpoints support new credential arrangements. Data is decoupling from application and IDM layer from application layer.
- User control empowerment: users have full knowledge regarding information they disclose.
- User preferences customize relying parties services

Our framework presents drawbacks, such as the unifying gateway. This point of failure will be a part of future work.

E. Implementation issues of the framework

This framework encompasses several modules. Research will now develop all those components and proceed towards the implementation and evaluation of associated prototype solution. Modules and interactions between platform components will be developed, implemented and tested successfully. Investigations will be conducted to select components supporting proposed aspects.

V. Conclusion

This article discusses ongoing concerns with the interoperability between different Identity Management solutions. Current solutions are developed independently but their functionality complement each other. This unified gateway will exploit all specifications to define new standards to encapsulate different protocols.

In future work, as part of the PhD thesis, we'll tackle description for options and parameters of protocols and how parameters are interpreted and mapped to each other.

Bringing a Unifying Gateway for Interoperable Identity Management as a response to interoperability challenges will enhance trust and encourage the wide use of identity systems. But, Trust relationships have to be established.

Acknowledgment

Authors thank University Mohammed V-Souissi, for supporting this work (research project 006/ENSIAS/2011).

References

[1] E. Bertino et al. "Identity Management: Concepts, Technologies, and Systems", pp. 110-111, Artech House, 2011

[2] R. Marx et al. "Increasing Security and Privacy in User-centric Identity Management: The IdM Card Approach", *2010 International Conference on P2P, Parallel, Grid, Cloud and Internet Computing,* pp. 459-464, IEEE, 2010

[3] P. Mishra et al. "*Conformance Requirements for the OASIS Security Assertion Markup Language V2.0"* OASIS SSTC, 19 pages, March 2005

[4] OASIS (Organization for the Advancement of Structured Information Standards) project: http://www.oasis-open.org, 2011

[5] G. Connor, "Shibboleth: A Templar Monitor", Kessinger Publishing, 216 pages, 2010

[6] D. Recordon et al., "OpenID: The Definitive Guide: Identity for the Social Web", 225 pages, O'Reilly Editions 2011.

[7] V. Bertocci et al. "Understanding Windows CardSpace: An Introduction to the Concepts and Challenges of Digital Identities", 384 pages, Addison-Wesley Progessional, 2008

[8] *Higgins Personal Data Service:* http://www.eclipse.org/higgins*, 2009*

Performance Analysis of a Keyed Hash Function based on Discrete and Chaotic Proven Iterations

Jacques M. Bahi, Jean-François Couchot, and Christophe Guyeux*
University of Franche-Comté, Computer Science Laboratory (LIFC)
Belfort, France
Email: {jacques.bahi, jean-francois.couchot, christophe.guyeux}@univ-fcomte.fr

***Abstract*—Security of information transmitted through the Internet is an international concern. This security is guaranteed by tools like hash functions. However, as security flaws have been recently identified in the current standard in this domain, new ways to hash digital media must be investigated. In this document an original keyed hash function is evaluated. It is based on chaotic iterations and thus possesses various topological properties as uniform repartition and sensibility to its initial condition. These properties make our hash function satisfy the requirements in this field. This claim is verified qualitatively and experimentally in this research work, among other things by realizing simulations of diffusion and confusion.**

***Keywords*-Keyed Hash Function; Internet Security; Mathematical Theory of Chaos; Topology.**

I. Introduction

Hash functions are fundamental tools to guarantee the quality and security of data exchanges through the Internet. For instance, they allow to store passwords in a secure manner or to check whether a download has occurred without any error. SHA-1 is probably the most widely used hash functions. It is present in a large panel of security applications and protocols through the Internet. However, in the last decade, security flaws have been detected in SHA-1. As the SHA-2 variants are algorithmically close to SHA-1 and produce finally message digests on principles similar to the MD4 and MD5 message digest algorithms, a new hash standard based on original approaches is then eagerly awaited. In this context, we have proposed a new hash function in [1]. Based on chaotic iterations, this function behaves completely different from approaches followed until now.

However chaos insertion to produce hash functions is sometimes disputed [2], [3]. Indeed existing chaos-based hash functions only include "somewhere" some chaotic functions of real variables like logistic, tent, or Arnold's cat maps. It is then supposed that the final hash function preserves these properties [4], [5], [6], [7]. But, in our opinion, this claim is not so evident. Moreover, even if these algorithms are themselves proven to be chaotic, their implementations on finite machines can result to lost of chaos property. Among other things, the main reason is that chaotic functions (embedded in these researches) only manipulate real numbers, which do not exist in a computer. In [1], the hash function we have proposed does not simply integrate chaotic maps into algorithms hoping that the result remains chaotic; we have conceived an algorithm and have mathematically proven that it is chaotic. To do both our theory and our implementation are based on finite integer domains and chaotic iterations.

Chaotic iterations (CIs) were formerly a way to formalize distributed algorithms through mathematical tools [8]. Thanks to these CIs, it was thus possible to study the convergence of synchronous or asynchronous programs over parallel, distributed, P2P, grid, or GPU platforms, in a view to solve linear and non-linear systems. CIs have recently revealed numerous interesting properties of disorder formalized into the mathematical topology framework. These studies lead to the conclusion that the chaos of CIs is very intense and that chaos class can tackle the computer science security field [9]. As CIs only manipulate binary digits or integers, we have shown that they are amenable to produce truly chaotic computer programs. Among other things, CIs have been applied to pseudo-random number generators [10] and to an information hiding scheme [11] in the previous sessions of the International Conference on Evolving Internet. In this paper, the complete unpredictable behavior of chaotic iterations is capitalized to produce a truly chaotic keyed hash function.

The remainder of this research work is organized in the following way. In Section II, basic recalls concerning chaotic iterations and Devaney's chaos are recalled. Our keyed hash function is presented, reformulated, and improved in Section III. Performance analyses are presented in the next two sections: in the first one a qualitative evaluation of this function is outlined, whereas in the second one it is evaluated experimentally. This research work ends by a conclusion section, where our contribution is summarized and intended future work is given.

II. Discrete and Chaotic Proven Iterations

This section gives some recalls on topological chaotic iterations. Let us firstly discuss about domain of iterated functions. As far as we know, no result rules that the chaotic behavior of a function that has been theoretically proven on $\mathbb{R}$ remains valid on the floating-point numbers, which is the implementation domain. Thus, to avoid loss of chaos this research work presents an alternative, namely to iterate boolean maps: results that are theoretically obtained in that domain are preserved during implementations.

Let us denote by $[\![a;b]\!]$ the following interval of integers: $\{a, a+1, \ldots, b\}$. A *system* under consideration iteratively modifies a collection of n components. Each component $i \in [\![1;n]\!]$ takes its value x_i among the domain $\mathbb{B} = \{0,1\}$. A *configuration* of the system at discrete time t (also said at *iteration* t) is the vector $x^t = (x_1^t, \ldots, x_n^t) \in \mathbb{B}^n$. In what follows, the dynamics of the system is particularized with the negation function $\neg : \mathbb{B}^n \to \mathbb{B}^n$ such that $\neg(x) = (\overline{x_i}, \ldots, \overline{x_n})$ where $\overline{x_i}$ is the negation of x_i.

* Authors in alphabetic order

ISBN: 978-1-61208-141-0

In the sequel, the *strategy* $S = (S^t)^{t\in\mathbb{N}}$ is the sequence defining which component is updated at time t and S^t denotes its $t-$th term. We introduce the function $F_\neg$ that is defined for the negation function by:

$$\begin{array}{rcl} F_\neg : [\![1;n]\!] \times \mathbb{B}^n & \rightarrow & \mathbb{B}^n \\ F_\neg(s,x)_j & = & \left\{ \begin{array}{l} \overline{x_j} \text{ if } j = s \\ x_j \text{ otherwise.} \end{array} \right. \end{array}$$

With such a notation, configurations are defined for times $t = 0, 1, 2, \ldots$ by:

$$\left\{ \begin{array}{l} x^0 \in \mathbb{B}^n \text{ and} \\ x^{t+1} = F_\neg(S^t, x^t) \ . \end{array} \right. \tag{1}$$

Finally, iterations defined in (1), called "chaotic iterations" [8], can be described by the following system

$$\left\{ \begin{array}{rcl} X^0 & = & ((S^t)^{t\in\mathbb{N}}, x^0) \in [\![1;n]\!]^{\mathbb{N}} \times \mathbb{B}^n \\ X^{k+1} & = & G_\neg(X^k) \end{array} \right. , \tag{2}$$

such that

$$G_\neg\left(((S^t)^{t\in\mathbb{N}}, x)\right) = \left(\sigma((S^t)^{t\in\mathbb{N}}), F_\neg(S^0, x)\right)$$

where σ is the function that removes the first term of the strategy (*i.e.*, S^0). Let us remark that the term "chaotic" in the name of this tool is just an adjective, which has a priori no link with the mathematical theory of chaos.

In the space $\mathcal{X} = [\![1;n]\!]^{\mathbb{N}} \times \mathbb{B}^n$ we define the distance between two points $X = (S, E), Y = (\check{S}, \check{E}) \in \mathcal{X}$ by

$$\begin{array}{rcl} d(X,Y) & = & d_e(E,\check{E}) + d_s(S,\check{S}), \text{ where} \\ d_e(E,\check{E}) & = & \displaystyle\sum_{k=1}^{n} \delta(E_k, \check{E}_k), \text{ and} \\ d_s(S,\check{S}) & = & \displaystyle\frac{9}{n} \sum_{k=1}^{\infty} \frac{|S^k - \check{S}^k|}{10^k}. \end{array}$$

If the floor value $\lfloor d(X,Y) \rfloor$ is equal to j, then the systems $E, \check{E}$ differ in j cells. In addition, $d(X,Y) - \lfloor d(X,Y) \rfloor$ is a measure of the differences between strategies S and $\check{S}$. More precisely, this floating part is less than 10^{-k} if and only if the first k terms of the two strategies are equal. Moreover, if the k^{th} digit is nonzero, then the k^{th} terms of the two strategies are different.

With this material it has been already proven that [9]:

- $G_\neg$ is a continuous function on a suitable metric space $(\mathcal{X}, d)$,
- iterations as defined in Equ. 2 are regular (*i.e.*, periodic points of $G_\neg$ are dense in $\mathcal{X}$),
- $(\mathcal{X}, G_\neg)$ is topologically transitive (*i.e.*, for any pair of open sets $U, V \subset \mathcal{X}$, there exists some natural number $k > 0$ s. t. $G_\neg^k(U) \cap V \neq \varnothing$),
- $(\mathcal{X}, G_\neg)$ has sensitive dependence on initial conditions (*i.e.*, there exists $\delta > 0$ s.t. for any $X \in \mathcal{X}$ and any neighborhood V of X, there exist $Y \in V$ and $k \geqslant 0$ with $d(G_\neg^k(X), G_\neg^k(Y)) > \delta$).

To sum up, we have previously established that the three conditions for Devaney's chaos hold for chaotic iterations. So CIs behave chaotically, as it is defined in the mathematical theory of chaos [12], [13].

III. A Chaos-Based Keyed Hash Function

This section first recalls an informal definition [14], [15] of Secure Keyed One-Way Hash Function. We next present our algorithm. Finally, we establish relations between the algorithm properties inherited from topological results and requirements of Secure Keyed One-Way Hash Function.

A. Secure Keyed One-Way Hash Function

Definition 1 (Secure Keyed One-Way Hash Function) *Let Γ and Σ be two alphabets, let $k \in K$ be a key in a given key space, let l be a natural numbers which is the length of the output message, and let $h : K \times \Gamma^+ \rightarrow \Sigma^l$ be a function that associates a message in Σ^l for each pair of key, word in $K \times \Gamma^+$. The set of all functions h is partitioned into classes of functions $\{h_k : k \in K\}$ indexed by a key k and such that $h_k : \Gamma^+ \rightarrow \Sigma^l$ is defined by $h_k(m) = h(k,m)$ i.e., h_k generates a message digest of length l.*

A class $\{h_k : k \in K\}$ is a Secure Keyed One-Way Hash Function *if it satisfies the following properties:*

1) *the function h_k is keyed one-way. That is,*
   a) *Given k and m, it is easy to compute $h_k(m)$.*
   b) *Without knowledge of k, it is hard to find m when $h_k(m)$ is given and to find $h_k(m)$ when only m is given.*
2) *The function h_k is keyed collision free, that is, without the knowledge of k it is difficult to find two distinct messages m and m' s.t. $h_k(m) = h_k(m')$.*
3) *Images of function h_k has to be uniformly distributed in Σ^l in order to counter statistical attacks.*
4) *Length l of produced image has to be larger than* 128 *bits in order to counter birthday attacks.*
5) *Key space size has to be sufficiently large in order to counter exhaustive key search.*

Let us now present our hash function that is based on chaotic iterations as defined in Section II. The hash value message is obtained as the last configuration resulting from chaotic iterations of $G_\neg$.

We then have to define the pair $X^0 = ((S^t)^{t\in\mathbb{N}}, x^0)$, *i.e.*, the strategy and the initial configuration x^0.

B. Computing x^0

The first step of the algorithm is to transform the message in a normalized $n = 256$ bits sequence x^0. This size n of the digest can be changed, mutatis mutandis, if needed. Here, this first step is close to the pre-treatment of the SHA-1 hash function, but it can easily be replaced by any other compression method.

To illustrate this step, we take an example, our original text is: "*The original text*".

Each character of this string is replaced by its ASCII code (on 7 bits). Following the SHA-1 algorithm, first we append a "1" to this string, which is then

```
10101001 10100011 00101010 00001101 11111100
10110100 11100111 11010011 10111011 00001110
11000100 00011101 00110010 11111000 11101001.
```

Next we append the block 1111000, which is the binary value of this string length (120), and finally another "1" is added:

```
10101001 10100011 00101010 00001101 11111100
```

```
10110100 11100111 11010011 10111011 00001110
11000100 00011101 00110010 11111000 11101001
11110001.
```

The whole string is copied, but in the opposite direction:

```
10101001 10100011 00101010 00001101 11111100
10110100 11100111 11010011 10111011 00001110
11000100 00011101 00110010 11111000 11101001
11110001 00011111 00101110 00111110 10011001
01110000 01000110 11100001 10111011 10010111
11001110 01011010 01111111 01100000 10101001
10001011 0010101.
```

The string whose length is a multiple of 512 is obtained, by duplicating enough this string and truncating at the next multiple of 512. This string, in which the whole original text is contained, is denoted by D. Finally, we split our obtained string into blocks of 256 bits and apply to them the exclusive-or function, from the first two blocks to the last one. It results a 256 bits sequence, that is in our example:

```
11111010 11100101 01111110 00010110 00000101
11011101 00101000 01110100 11001101 00010011
01001100 00100111 01010111 00001001 00111010
00010011 00100001 01110010 01000011 10101011
10010000 11001011 00100010 11001100 10111000
01010010 11101110 10000001 10100001 11111010
10011101 01111101.
```

The configuration x^0 is the result of this pre-treatment and is a sequence of $n = 256$ bits. Notice that some distinct texts lead to the same string.

Let us build now the strategy $(S^t)^{t\in\mathbb{N}}$ that depends on both the original message and a given key.

C. Computing $(S^t)^{t\in\mathbb{N}}$

To obtain the strategy S, an intermediate sequence $(u^t)^{t\in\mathbb{N}}$ is constructed from D, as follows:

1) D is split into blocks of 8 bits. Let $(u^t)^{t\in\mathbb{N}}$ be the finite sequence where u^t is the decimal value of the t^{th} block.
2) A circular rotation of one bit to the left is applied to D (the first bit of D is put on the end of D). Then the new string is split into blocks of 8 bits another time. The decimal values of those blocks are added to (u^t).
3) This operation is repeated again 6 times.

Because of the function $\theta \longmapsto 2\theta \ (mod\ 1)$ is known to be chaotic in the sense of Devaney [12], we define the strategy $(S^t)^{t\in\mathbb{N}}$ with:

$$S^t = (u^t + 2 \times S^{t-1} + t) \mod 256,$$

which is then highly sensitive to initial conditions and then to changes of the original text. On the one hand, when a keyed hash function is desired, this sequence $(S^t)^{t\in\mathbb{N}}$ is initialized with the given key k (*i.e.*, $S^0 = k$). On the other hand, it is initialized to u^0 if the hash function is unkeyed.

D. Computing the digest

To construct the digest, chaotic iterations of $G_\neg$ are realized with initial state $X^0 = ((S^t)^{t\in\mathbb{N}}, x^0)$ as defined above. The result of these iterations is a $n = 256$ bits vector. Its components are taken 4 per 4 bits and translated into hexadecimal numbers, to obtain the hash value:

```
63A88CB6AF0B18E3BE828F9BDA4596A6
A13DFE38440AB9557DA1C0C6B1EDBDBD.
```

As a comparison if we replace "*The original text*" by "*the original text*", the hash function returns:

```
33E0DFB5BB1D88C924D2AF80B14FF5A7
B1A3DEF9D0E831194BD814C8A3B948B3.
```

We then investigate qualitative properties of this algorithm.

IV. Qualitative Analysis

We show in this section that, as a consequence of recalled theoretical results, this hash function tends to verify desired informal properties of a secure keyed one-way hash function.

A. The avalanche criteria

Let us first focus on the avalanche criteria, which means that a difference of one bit between two given medias has to lead to completely different digest. In our opinion, this criteria is implied by the topological properties of sensitive dependence to the initial conditions, expansivity, and Lyapunov exponent. These notions are recalled below.

First, a function f has a constant of expansivity equal to ε if an arbitrarily small error on any initial condition is *always* magnified till ε. In our iteration context and more formally, the function $G_\neg$ verifies the *expansivity* property if there exists some constant $\varepsilon > 0$ such that for any X and Y in $\mathcal{X}$, $X \neq Y$, we can find a $k \in \mathbb{N}$ s.t. $d(G_\neg^k(X), G_\neg^k(Y)) \geqslant \varepsilon$. We have proven in [16] that, $(\mathcal{X}, G_\neg)$ is an expansive chaotic system. Its constant of expansivity is equal to 1.

Next, some dynamical systems are highly sensitive to small fluctuations into their initial conditions. The constants of sensibility and expansivity have been historically defined to illustrate this fact. However, in some cases, these variations can become enormous, can grow in an exponential manner in a few iterations, and neither sensitivity nor expansivity are able to measure such a situation. This is why Alexander Lyapunov has proposed a new notion being able to evaluate the amplification speed of these fluctuations we now recall:

Definition 2 (Lyapunov Exponent) *Let be given an iterative system $x^0 \in \mathcal{X}$ and $x^{t+1} = f(x^t)$. Its* Lyapunov exponent *is defined by:*

$$\lim_{t\to+\infty} \frac{1}{t} \sum_{i=1}^{t} \ln \left| \ f' \left(x^{i-1}\right) \right|$$

By using a topological semi-conjugation between $\mathcal{X}$ and $\mathbb{R}$, we have proven in [9] that For almost all X^0, the Lyapunov exponent of chaotic iterations $G_\neg$ with X^0 as initial condition is equal to $\ln(n)$.

Let us now explain why the topological properties of our hash function lead to the avalanche effect. Due to the sensitive dependence to the initial condition, two close media can possibly lead to significantly different digests. The expansivity property implies that these similar medias mostly lead to very different hash values. Finally, a Lyapunov exponent greater than 1 lead to the fact that these two close media will always finish to have very different digests.

B. Preimage Resistance

Let us now discuss about the first preimage resistance of our unkeyed hash function denoted by h. Indeed, as recalled previously, an adversary given a target image D should not be able to find a preimage M such that $h(M) = D$. One reason (among many) why this property is important is that on most computer systems user passwords are stored as the cryptographic hash of the password instead of just the plaintext password. Thus an attacker who gains access to the password file cannot use it to then gain access to the system, unless it is able to invert target message digest of the hash function.

We now explain why, topologically speaking, our hash function is resistant to preimage attacks. Let m be the message to hash, (S, x^0) its normalized version (*i.e.*, the initial state of our chaotic iterations), and $M = h(m)$ the digest of m by using our method. So chaotic iterations with initial condition (S, M) and iterate function $G_{\neg}$ have x^0 as final state. Thus it is impossible to invert the hash process with a view to obtain the normalized message by using the digest. Such an attempt is equivalent to try to forecast the future evolution of chaotic iterations by only using a partial knowledge of its initial condition. Indeed, as M is known but not S, the attacker has an incertitude on the initial condition. he only knows that this value is into an open ball of radius 1 centered at the point M, and the number of terms of such a ball is infinite.

With such an incertitude on the initial condition, and due to the numerous chaos properties possessed by the chaotic iterations (as these stated in Section IV-A), this prediction is impossible. Furthermore, due to the transitivity property, it is possible to reach all of the normalized medias, when starting to iterate into this open ball. Indeed, it is possible to establish that, all of these possible normalized medias can be obtained in at most 256 iterations, and we iterate at least 519 times to obtain our hash value (*c.f.* Proposition 2 below). Finally, to find the normalized media does not imply the discovery of the original plain-text.

V. Quantitative and Experimental Evaluations

Let us first give some examples of hash values before discussing about the algorithm complexity.

A. Hash values

Let us now consider our hash function with $n = 128$. To give illustration of the confusion and diffusion properties, we will use this function to generate hash values in the following cases:

Case 1. The original text message is the poem *Ulalume* (E.A.Poe), which is constituted by 104 lines, 667 words, and 3754 characters.

Case 2. We change *serious* by *nervous* in the verse "*Our talk had been serious and sober*"

Case 3. We replace the last point '.' with a coma ','.

Case 4. In "*The skies they were ashen and sober*", skies becomes Skies.

Case 5. The new original text is the binary value of the Figure 1.

Case 6. We add 1 to the gray value of the pixel located in position (123,27).

Case 7. We substract 1 to the gray value of the pixel located in position (23,127).

The corresponding hash values in hexadecimal format are:

Case 1. C0EA2325BBF956D27C3561977E48B3E1,
Case 2. 6C4AC2F8579BCAB95BAD68468ED102D6,
Case 3. A538A76E6E38905DA0D35057F1DC1B14,

Figure 1: The original plain-image.

Case 4. 01530A057B6A994FBD3887AF240F849E,
Case 5. DE188603CFE139864092C7ABCD21AE50,
Case 6. FF855E5A626532A4AED99BACECC498B1,
Case 7. 65DB95737EFA994DF37C7A6F420E3D07.

These simulation results are coherent with the topological properties of sensitive dependence to the initial condition, expansivity, and Lyapunov exponent: any alteration in the message causes a substantial difference in the final hash value.

B. Algorithm Complexity

In this section is evaluated the complexity of the above hash function for a size l of the media (in bits).

Proposition 1 *The stages of initialization (Sections III-B and III-C) need $\mathcal{O}(l)$ elementary operations to be achieved.*

Proof: In this stage only linear operations over binary tables are achieved, such as: copy, circular shift, or inversion. ■

Let us consider the digest computation stage (Section III-D).

Proposition 2 *The digest computation stage requires less than $2l + 2\log_2(l+1) + 515$ elementary operations.*

Proof: The cost of an iteration is reduced to the negation operation on a bit, which is an elementary operation. Thus, the second stage is realized in t elementary operations, where t is the number of terms into the sequence S. But S has the same number of terms than u, and u and D have the same size (indeed, to build u, D has been copied 8 times, and bits of this sequence have been regrouped 8 per 8 to obtain the terms of u). To sum up, the size of D is equal to the total number of elementary operations of the digest computation stage.

The following operations are realized to obtain D.

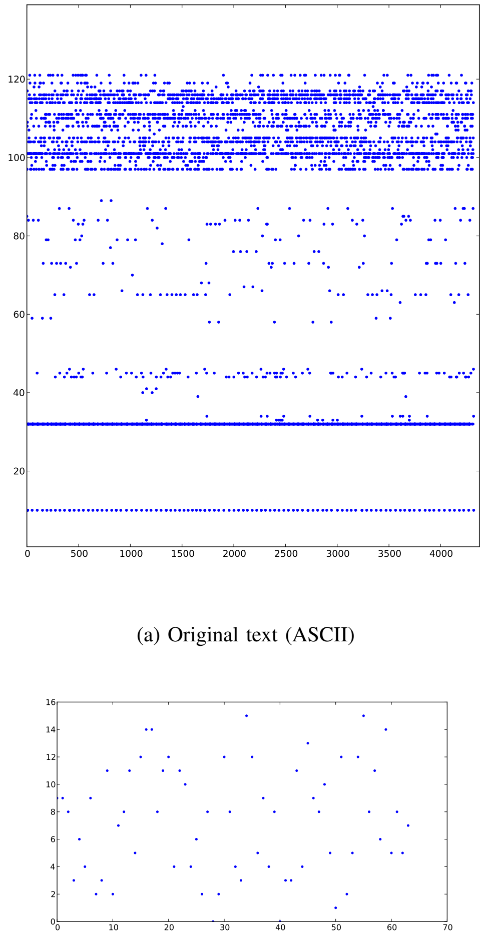
1) The digit 1 is added: D has $l+1$ bits.
2) The binary value of the size is added, followed by another bit: D has $l + 2 + \log_2(l+1)$ bits.
3) This string is copied after inversion: D has now $2 \times (l + 2 + \log_2(l+1))$ bits.
4) Lastly, this string is copied until the next multiple of 512: in the worst situation, 511 bits have been added, so D has in the worst situation $2l + 2\log_2(l+1) + 515$ bits.

■

We can thus conclude that:

Theorem 1 *The computation of an hash value is linear with the hash function presented in this research work.*

ISBN: 978-1-61208-141-0



(a) Original text (ASCII)

(b) Digest (Hexadecimal)

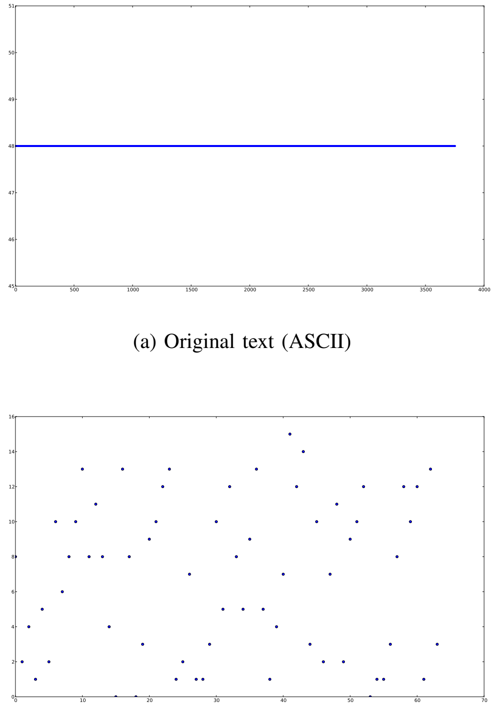
Figure 2: Values repartition of Ulalume poem

C. Experimental Evaluation

We focus now on the illustration of the diffusion and confusion properties [17]. Let us recall that confusion refers to the desire to make the relationship between the key and the ciphertext as complex and involved as possible, whereas diffusion means that the redundancy in the statistics of the plaintext must be "dissipated" in the statistics of the ciphertext. Indeed, the avalanche criterion is a modern form of the diffusion, as this term means that the output bits should depend on the input bits in a very complex way.

1) Uniform repartition for hash values: To show the diffusion and confusion properties verified by our scheme, we first give an illustration of the difference of characters repartition between a plain-text and its hash value when the original message is again the Ulalume poem. In Figure 2a, the ASCII codes are localized within a small area, whereas in Figure 2b the hexadecimal numbers of the hash value are uniformly distributed.

A similar experiment has been realized with a message having the same size, but which is only constituted by the character "*0*". The contrast between the plain-text message and its digest are respectively presented in Figures 3a and 3b. Even under this very extreme condition, the distribution of the digest still remains uniform. To conclude, these simulations tend to indicate that no information concerning the original message can be found into its hash value, as it is recommended by the Shannon's diffusion and confusion.



(a) Original text (ASCII)

(b) Digest (Hexadecimal)

Figure 3: Values repartition of the "*00000000*" message

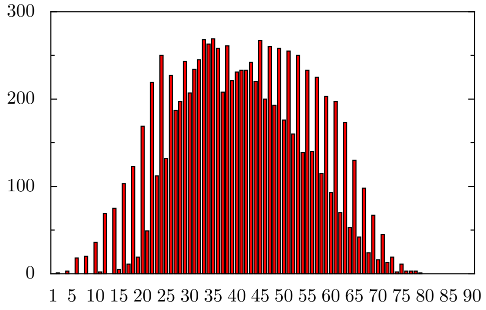


Figure 4: Histogram

2) Behavior through small random changes: We now consider the following experiment. A first message of 100 bits is randomly generated, and its hash value of size 80 bits is computed. Then one bit is randomly toggled into this message and the digest of the new message is obtained. These two hash values are compared by using the hamming distance, to compute the number B_i of changed bits. This test is reproduced 10000 times. The corresponding distribution of B_i is presented in Figure 4.

As desired, Figure 4 show that the distribution is centered around 40, which reinforces the confidence put into the good capabilities of diffusion and confusion of the proposed hash algorithm.

3) Statistic analysis of diffusion and confusion: Finally, we generate 1000 sequences of 1000 bits, and for each of these sequences, we toggle one bit, thus obtaining a sequence of 1000 couples of 1000 bits. As previously, the two digests of each couple i are obtained, and the hamming distance B_i

	B_{min}	B_{max}	$\overline{B}$	$P(\%)$	ΔB	$\Delta P(\%)$
N = 256	50	92	67.57	52.78	8.89	6.95
N = 512	47	82	65.13	51.11	7.65	5.87
N = 1024	47	81	63.01	52.10	7.51	5.71

Table I: Statistical performance of the proposed hash function

between these digests are computed. To analyse these results, the following common statistics are used.

- Mean changed bit number $\overline{B} = \frac{1}{N}\sum_{i=1}^{N} B_i$.
- Mean changed probability $P = \frac{\overline{B}}{128}$.
- $\Delta B = \sqrt{\dfrac{1}{N-1}\sum_{i=1}^{N}(B_i - \overline{B})^2}$.
- $\Delta P = \sqrt{\dfrac{1}{N-1}\sum_{i=1}^{N}(\frac{B_i}{128} - P)^2}$.

The obtained statistics are listed in Table I. Obviously, both the mean changed bit number $\overline{B}$ and the mean changed probability P are close to the ideal values (64 bits and 50%, respectively), which illustrates the diffusion and confusion capability of our algorithm. Lastly, as ΔB and ΔP are very small, these capabilities are very stable.

VI. Conclusion

MD5 and SHA-0 have been broken in 2004. An attack over SHA-1 has been achieved with only 2^{69} operations (CRYPTO-2005), that is, 2000 times faster than a brute force attack (that requires 2^{80} operations). Even if 2^{69} operations still remains impossible to realize on common computers, such a result based on a previous attack on SHA-0 is a very important one: it leads to the conclusion that SHA-2 is not as secure as it is required for the Internet applications. So new original hash functions must be found.

In this research work, a new hash function has been presented. The security in this case has been guaranteed by the unpredictability of the behavior of the proposed algorithms. The algorithms derived from our approach satisfy important properties of topological chaos such as sensitivity to initial conditions, uniform repartition (as a result of the transitivity), unpredictability, and expansivity. Moreover, its Lyapunov exponent can be as great as needed. The results expected in our study have been experimentally checked. The choices made in this first study are simple: compression function inspired by SHA-1, negation function for the iteration function, *etc.* The aim was not to find the best hash function, but to give simple illustrated examples to prove the feasibility in using the new kind of chaotic algorithms in computer science. Finally, we have shown how the mathematical framework of topological chaos offers interesting qualitative and qualitative tools to study the algorithms based on our approach.

In future work, we will investigate other choices of iteration functions and chaotic strategies. We will try to characterize topologically the diffusion and confusion capabilities. Other properties induced by topological chaos will be explored and their interest for the realization of hash functions will be deepened.

References

[1] J. M. Bahi and C. Guyeux, "Hash functions using chaotic iterations," *Journal of Algorithms & Computational Technology*, vol. 4, no. 2, pp. 167–181, 2010.

[2] C. song Zhou and T. lun Chen, "Extracting information masked by chaos and contaminated with noise: Some considerations on the security of communication approaches using chaos," *Physics Letters A*, vol. 234, no. 6, pp. 429 – 435, 1997.

[3] W. Guo, X. Wang, D. He, and Y. Cao, "Cryptanalysis on a parallel keyed hash function based on chaotic maps," *Physics Letters A*, vol. 373, no. 36, pp. 3201 – 3206, 2009.

[4] X. M. Wang, J. S. Zhang, and W. F. Zhang, "One-way hash function construction based on the extended chaotic maps switch," *Acta Phys. Sinici.*, vol. 52, No. 11, pp. 2737–2742, 2003.

[5] D. Xiao, X. Liao, and Y. Wang, "Improving the security of a parallel keyed hash function based on chaotic maps," *Physics Letters A*, vol. 373, no. 47, pp. 4346 – 4353, 2009.

[6] ——, "Parallel keyed hash function construction based on chaotic neural network," *Neurocomputing*, vol. 72, no. 10-12, pp. 2288 – 2296, 2009, lattice Computing and Natural Computing (JCIS 2007) / Neural Networks in Intelligent Systems Designn (ISDA 2007).

[7] D. Xiao, F. Y. Shih, and X. Liao, "A chaos-based hash function with both modification detection and localization capabilities," *Communications in Nonlinear Science and Numerical Simulation*, vol. 15, no. 9, pp. 2254 – 2261, 2010.

[8] D. Chazan and W. Miranker, "Chaotic relaxation," *Linear algebra and its applications*, pp. 199–222, 1969.

[9] C. Guyeux, "Le désordre des itérations chaotiques et leur utilité en sécurité informatique," Ph.D. dissertation, Université de Franche-Comté, 2010.

[10] Q. Wang, J. Bahi, C. Guyeux, and X. Fang, "Randomness quality of CI chaotic generators. application to internet security," in *INTERNET'2010. The 2nd Int. Conf. on Evolving Internet*. Valencia, Spain: IEEE Computer Society Press, Sep. 2010, pp. 125–130, best Paper award.

[11] C. Guyeux and J. Bahi, "An improved watermarking algorithm for internet applications," in *INTERNET'2010. The 2nd Int. Conf. on Evolving Internet*, Valencia, Spain, Sep. 2010, pp. 119–124.

[12] R. L. Devaney, *An Introduction to Chaotic Dynamical Systems*, 2nd ed. Redwood City, CA: Addison-Wesley, 1989.

[13] Knudsen, "Chaos without nonperiodicity," *Amer. Math. Monthly*, vol. 101, 1994.

[14] S. Bakhtiari, R. Safavi-Naini, and J. Pieprzyk, "Keyed hash functions," in *Cryptography: Policy and Algorithms*, ser. Lecture Notes in Computer Science, E. Dawson and J. Golic, Eds. Springer Berlin / Heidelberg, 1996, vol. 1029, pp. 201–214.

[15] J. Zhang, X. Wang, and W. Zhang, "Chaotic keyed hash function based on feedforward-feedback nonlinear digital filter," *Physics Letters A*, vol. 362, pp. 439–448, 2007.

[16] C. Guyeux, N. Friot, and J. Bahi, "Chaotic iterations versus spread-spectrum: chaos and stego security," in *IIH-MSP'10, 6-th Int. Conf. on Intelligent Information Hiding and Multimedia Signal Processing*, Darmstadt, Germany, Oct. 2010, pp. 208–211.

[17] C. E. Shannon, "Communication theory of secrecy systems," *Bell Systems Technical Journal*, vol. 28, pp. 656–715, 1949.

Virtual Internet Connections Over Dynamic Peer-to-Peer Overlay Networks

Telesphore Tiendrebeogo, Damien Magoni
University of Bordeaux – LaBRI
Bordeaux, France
{tiendreb,magoni}@labri.fr

Oumarou Sié
University of Ouagadougou
Ouagadougou, Burkina Faso
sie@univ-ouaga.bf

***Abstract*—Current Internet applications are still mainly bound to the state of their transport layer connections. This prevents many features such as end-to-end security and mobility from functioning smoothly in a dynamic network. In this paper, we propose a novel architecture for decoupling communications from their supporting devices. This enables the complete separation of the devices, applications and users. Our architecture is based on a peer-to-peer overlay network that provides its own distributed hash table system. Preliminary simulation results show that our proposal is feasible.**

***Keywords*-Overlay; virtual connection; distributed hash table.**

I. Introduction

Current Internet communications are still based on the paradigms set by the TCP/IP protocol stack 30 years ago and they are lacking several key features. Although many efforts have been done during the last decade to provide mobility, security and multicasting, those efforts have mainly been focused on the equipments themselves (e.g., computers, smart phones, routers, etc.) and not on the logical part of the communications. In fact, although we already have a lot of mobile equipments, it is still impossible to transfer a communication from one device to another without interrupting the communication (and thus start it all over again). In the same way, although we have the choice of many applications for carrying one task, it is also still impossible to transfer a communication from one application to another without interrupting the communication. Layer 2 device mobility (e.g., WiFi, WiMAX, 3G and beyond) is nowadays well supported but users still have a very limited access to upper layers mobility (e.g., MobileIP, TCP-Migrate).

In this paper we propose and describe a new architecture for using virtual connections setup over dynamic P2P overlay networks built on top of the TCP/IP protocol stack of the participating devices. We have called this architecture CLOAK (Covering Layers Of Abstract Knowledge). This architecture supports names for entities (i.e., users) and devices, virtual addresses for devices and logical sessions that enable a full virtualization of all kinds of Internet communications. The new semantics brought by our proposal open up many novel possibilities for Internet communications. The virtual connections setup and managed by our solution enable for instance the transparent handling of the breakdown and restore of transport layer connections (e.g., such as TCP or SCTP connections).

The remainder of this paper is organized as follows. Section II outlines the related previous work done on virtual connections. Section III presents the design and features of our architecture. Section IV describes its implementation. Section V presents some preliminary results obtained by simulations. Finally, we conclude the paper and present our future research directions.

II. Related work

Virtual connections, as we define them, can be considered as providing (among other benefits) transport layer connection mobility. Research on such transport layer connection mobility has mainly remained experimental up to now. Concerning the TCP connection management, several solutions have been proposed. TCP-Migrate [1], [2] developed at the Massachusetts Institute of Technology, provides a unified framework to support address changes and connectivity interruptions. Migrate provides mobile-aware applications with a set of system primitives for connectivity re-instantiation. Migrate enables applications to reduce their resource consumption during periods of disconnection and resume sessions upon reconnection. Rocks [3] developed at the University of Wisconsin, protect sockets-based applications from network failures, such as link failures, IP address changes and extended periods of disconnection. Migratory TCP [4], developed at Rutgers University, is a transport layer protocol for building highly-available network services by means of transparent migration of the server endpoint of a live connection between cooperating servers that provide the same service. The origin and destination servers cooperate by transferring the connection state in order to accommodate the migrating connection. Finally, the Fault-Tolerant TCP [5], [6] developed at the University of Texas, allows a faulty server to keep its TCP connections open until it either recovers or it is failed over to a backup. The failure and recovery of the server process are completely transparent to client processes. However, all these projects only deal with TCP re-connection. They do not enable the total virtualization of a communication. They also do not allow to switch both applications and/or devices from any communicating user at will.

III. Architecture

A. Design

In the context of our architecture, a *communication* is a set of interactions between several entities. It can be any form of simplex or duplex communication where information is processed and exchanged between the entities (e.g., talk, view video, check bank account, send mail, etc.). An *interaction* is simply a given type of action carried out between two or more entities by using an application protocol (e.g., FTP, HTTP, etc.). An *entity* is typically a human user but it can also be an automated service such as a server. A communication typically involves a minimum of two entities but it can involve many more in the case of multicast and broadcast communications. Finally, a device is a communication terminal equipment. On the device are running *applications* that are used by an entity to interact with other entities. Given this context, the aim of our architecture is to enable a communication to be carried out without any definitive unwanted interruption when some or all of its components (i.e., device, application or entity) are evolving (i.e., moving or changing) over space and time. Our architecture enables a communication to have a lifetime that only depends on the will of the currently implied entities. Changes in devices, applications and even entities (when it makes sense) will not terminate the communication.

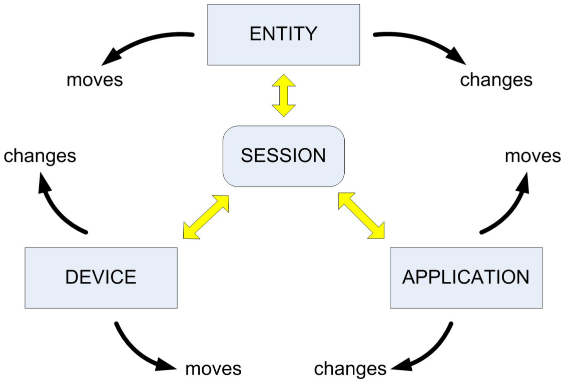


Figure 1. CLOAK communication paradigm.

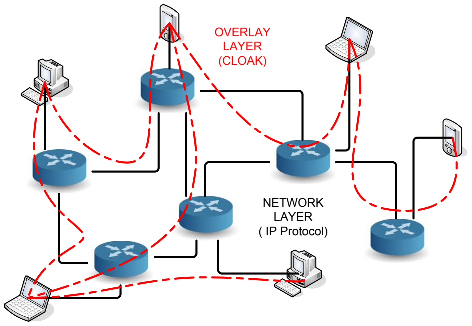


Figure 2. Overlay network.

Fig. 1 shows the CLOAK communication paradigm. In order to untie entities, applications and devices, CLOAK introduces the use of a *session*. A session is a communication descriptor that contains everything needed for linking entities, applications and devices together in a flexible way. A session can be viewed as a container storing the identity and the management information of a given communication. Thus the lifetime of a communication between several entities is equal to the lifetime of its corresponding session. As shown on Fig. 1, a device can move or be changed for another without terminating the session. Similarly, an application can be changed for another if deemed appropriate or even moved (i.e., mobile code) also without terminating the session. Finally, entities can move or change (i.e., be transferred to another entity) without terminating the session if this is appropriate for a given communication. We can see that in our new architecture, entities, applications and devices are loosely bound together (i.e., represented by yellow arrows in Fig. 1) during a communication and all the movements and changes of devices, applications and entities are supported. Note that in Fig. 1, only one instance of each part (device, application, entity) of a communication is shown, other instances would obey the same scheme.

B. Operation

In order to provide all the above mentioned features, our architecture sets up and maintains a P2P overlay network. All the devices that wish to share resources in order to benefit from the architecture join together to form an overlay. Fig. 2 shows an overlay example with the links shown in dotted red lines. The devices (i.e., end-hosts) connect to the others by creating virtual links (i.e., transport layer connections). Devices with two or more links play the role of overlay routers. We allow the overlay network to build up without any constraints. Network devices can connect arbitrarily to each other and join and leave the P2P network at any time.

When joining the overlay, each device obtains a unique overlay address. The method for addressing the peers and routing the packets inside the overlay is based on the groundbreaking work of Kleinberg [7] that assigns addresses equal to coordinates adequately taken from the hyperbolic plane (represented by the open unit disk). This method creates a greedy embedding of an addressing tree upon the overlay network. This addressing tree is a regular tree of degree k. However in Kleinberg's proposal, the construction of the embedding requires a full knowledge of the graph topology which is also considered static. This is required as the degree k of the addressing tree is equal to the highest degree found in the network. We have enhanced his proposal in order to manage a dynamic topology which is able to grow and shrink over time. Indeed, as we setup an overlay network, we are able to set the degree k of the addressing tree to an arbitrary value and as such, we are able to avoid the discovery of the highest degree node. This specificity renders our method scalable because unlike [7], we do not have to make a two-pass algorithm over the whole network to find its highest degree. The fixed degree that we choose determines how many addresses each peer will be able to give. The degree of the addressing tree is therefore set at the creation of the overlay for all its lifetime. In the overlay however, a peer can connect to any other peer at any time in order to obtain an address thus setting the degree does not prevent the overlay to grow. These hyperbolic addresses enable the use of a greedy routing based on the hyperbolic distance metric that is guaranteed to work. Thus, only the addresses of the neighbors of a peer are needed to forward a message to its destination. This is highly scalable as the peers do not need to build and maintain routing tables. Our dynamic method is fully described in our previous paper [8].

In order to set up the DHT (Distributed Hash Tables) structure needed by our architecture on top of the P2P overlay network, we only need to add a mapping function between a keyspace and the addressing space of the peers. When a peer wants to store an entry in the DHT, it first creates a fixed length key by hashing a key string with the SHA-1 algorithm. Then, the peer maps the key to an angle by a linear transformation. The peer computes a virtual point on the unit circle by using this angle. Next, the peer determines the coordinates of the closest peer to the computed virtual point. The peer then sends a store request to this closest peer. This request is routed inside the overlay by using the greedy routing algorithm presented above.

With the addressing, routing and mapping services provided by our architecture, any user/entity of the P2P overlay network can communicate with any other by setting up a virtual connection on top of the overlay. The steps for establishing a communication between two entities of an overlay are the following:

1) Bootstrap into the overlay by setting transport layer connections to one or more devices (i.e., neighbor peers).
2) Obtain an overlay address from one of those neighbor peers.
3) Identify oneself in the overlay with unique device and entity identifiers.
4) Create a session.
5) Invite in this session another entity to communicate with.
6) Set an overlay layer virtual connection to this entity as shown in Fig. 3.
7) Send the data stream through this connection.

To be able to implement our architecture, we need to introduce several new types of identifiers. More specifically we need to define the following new namespaces:

- Session namespace: any session should be attributed a unique identifier that defines the session during its lifetime in the overlay.
- Device namespace: any device should be attributed a unique identifier that permanently represents the device. The lifespan of this identifier should be as long as the lifespan of its corresponding device.
- Entity namespace: any entity should be attributed a unique identifier that represent the entity in a given context. It can be the name of a real person (John Smith) but it could also be the identifier of a professional function (Sales Manager) or the name of an organization (Michelin Company) or a specific service (Areva Accounting service). The lifespan of this identifier should be as long as the lifespan of its corresponding entity.
- Application namespace: any application used during a part or all of a session should be attributed a unique identifier that enables it to receive data from the other applications of this session. The lifespan of this identifier should be equal to the lifespan of the use of the application. If the entity switches to another application, this identifier should be updated.

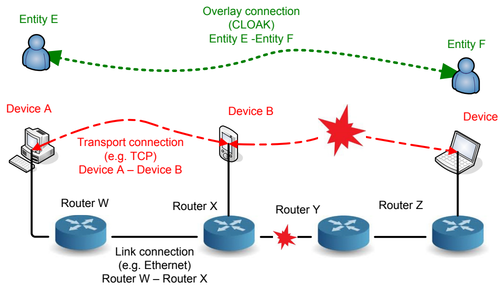


Figure 3. Virtual connections.

The identifiers will be stored in a DHT built on the P2P overlay network. Each peer will store a fraction of all the records in its naming module. There will be records for the devices (containing pairs like: device ID - overlay address), for the entities (containing pairs like: entity ID - device ID), for the applications (containing pairs like: application ID - session ID) and finally for the sessions (containing pairs like: session ID - session data information). An application using CLOAK will not directly open a connection with an IP address and a port number as with the usual sockets API but it will use the destination's entity ID as well as a stream ID. Fig. 4 shows a typical scenario relying on this naming system for solving an entity's location. The yellow oval represents the CLOAK DHT. An entity B registers itself in the DHT by providing the device identifier it is on and its overlay address. Any entity A can now retrieve the location of B by querying the DHT. It can then connect to B via the overlay. When B switches to another device during the same session, A can reconnect to B by using its new overlay address.

As defined earlier, a session is a communication's context container storing everything necessary to bind together entities, applications and devices that are involved in a given communication. Any device, application or entity can be changed or moved without terminating the session. In order to make this possible, the session will be stored in the DHT built by the peers of the overlay network. The DHT will ensure reliability by redundantly storing the sessions on several peers. This session management system will enable the survival of the session until all the entities involved decide to stop it. Fig. 5 shows a typical scenario relying on this session management

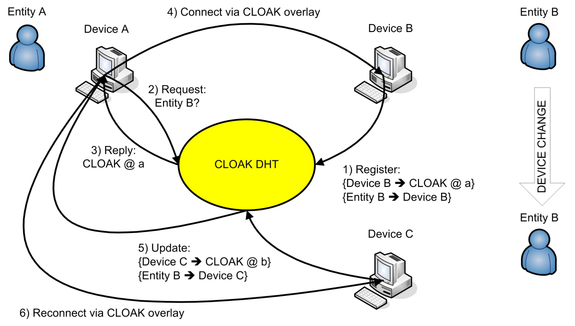

Entity A
4) Connect via CLOAK overlay
Entity B
Device A
Device B
2) Request: Entity B?
DEVICE CHANGE
3) Reply: CLOAK @ a
CLOAK DHT
1) Register: {Device B → CLOAK @ a} {Entity B → Device B}
Entity B
Device C
5) Update: {Device C → CLOAK @ b} {Entity B → Device C}
6) Reconnect via CLOAK overlay

Figure 4. Identification and localization.

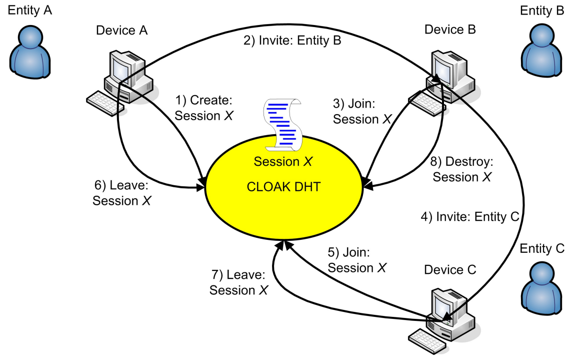


Figure 5. Session management.

system. The yellow oval represents the CLOAK DHT. Let us assume that an entity A wants to start a video conference communication with an entity B. It first creates a session called X describing the desired interaction (e.g., video conference) as well as the destination entity that it wants to communicate with (here the entity B). Then A sends an invite message to B that replies by joining the session X. Later on the entity B invites another entity C to participate in the video conference. C accepts and joins the session X. Three entities are now involved in the session X. Later on, the entity A leave the session X allowing the others to continue. This thus does not end the session X. Later on the entity C leaves the session X. The entity B being the last one involved decides to destroy the session and thus to end the communication.

C. Usage

Our architecture has a wide range of usages. It provides mechanisms for mobile and switchable applications, for adaptive transport protocol switching and enables the definition and use of new namespaces. It can build scalable and reliable dynamic Virtual Private Networks, define fully isolated Friend-to-Friend networks, serve as an anonymizing layer for Darknets or be used as a convergence layer for IPv4, NATs and IPv6. The Table I shows the benefits of *cloaked* applications. Applications are grouped by families. Messaging applications contain e-mail, talk and chat programs. Conferencing applications regroup real-time audio and video communications based on protocols such as SIP and H323. Sharing applications encompass file-sharing, blogging and social networking applications. Finally, streaming applications contain audio and video broadcasting services such as Internet radios, IPTV, and VoD. Most of the features are usually self-explaining but we give now a few examples to highlight possible scenarios. Reachability is the ability to be reached on whatever device the user is currently using. When someone sends a message to an entity, the CLOAK DHT can be used dynamically to determine on which device is the entity and the message is routed to the proper device. Mobility is the ability of CLOAK to hide the handovers of the lower layers to the applications. If an entity is moving or switching devices, real-time applications will be maintained without interruption at the application level. CLOAK uses security by using entity IDs, thus establishing End-to-End (E2E) privacy and authentication. Because CLOAK packets usually transit through several terminals before reaching destination, the IP address of the source is often unknown to the destination thus providing anonymity. Redirection is the ability to forward a message or a stream to another entity. Finally, multicasting support is provided by CLOAK as group addresses can be easily set up in the DHT. This feature is useful for saving bandwidth during group communications.

TABLE I
FEATURES FOR *cloaked* APPLICATIONS.

Application type	Messaging	Conferencing	Sharing	Streaming
Reachability	✓			
Mobility		✓		✓
E2E privacy	✓	✓		
E2E authentication	✓	✓		
Anonymity			✓	✓
Redirection	✓			✓
Multicasting		✓	✓	✓

IV. IMPLEMENTATION

Fig. 6 shows the OSI layers where the CLOAK architecture fits in. CLOAK uses the session layer and the presentation layer between the transport and application layers. These layers do not exist in the Internet stack model but they do already exist in the OSI model. In these two layers we add two new protocols. We add a CLOAK session protocol (CSP) at the session layer and a CLOAK interaction protocol (CIP) at the presentation layer. We also define new identifiers for these new protocols. These new identifiers enable data streams to be bound by virtual identifiers instead of the typical network identifiers (i.e., IP address, protocol n°, port n°) that are now able to change without breaking the communication. As shown in Fig. 1, identifiers of devices, applications and entities are interwoven together inside a session, but for the purpose of implementation, we have to order them. We chose to manage a session and its involved devices at the session layer. We also chose to manage the interactions between entities at the presentation layer. As previously said, an interaction is a type of action carried out between two or more entities. It is equal to the use of an existing application layer protocol (e.g., FTP, SMTP, HTTP, etc.). Indeed, our architecture will use the existing application layer protocols as well as the existing

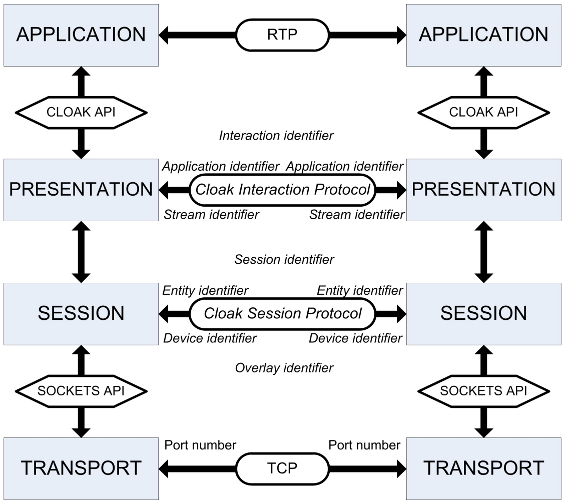


Figure 6. CLOAK architecture in the OSI model.

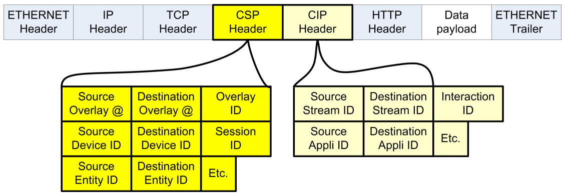


Figure 7. CLOAK protocol encapsulation.

transport layer protocols. Thus a file transfer (FTP [9]) client application will still use the FTP protocol to speak to a FTP server. Only the portion of code for establishing a session and thus a connection to the server will have to be rewritten for using the CLOAK API instead of the socket API [10]. The code implementing the application layer protocol will not have to be changed. Please note that the CLOAK API and the mapping of application connections to transport sockets inside the middleware are not defined yet. They will be presented in a future work.

We have shown in Fig. 6 how the CLOAK architecture fits in the network protocol stack. We will show now how this design translates into the format of the packet headers. Fig. 7 shows a CLOAK packet exchanged between a web client and a web server. The application header involving the HTTP protocol is now located after the CLOAK headers. We have added two additional headers. The CSP header is located directly above the TCP protocol managing the connection in the operating system of the device. It contains the overlay addresses for routing inside the overlay and enabling device mobility, the device identifiers for switching devices and enabling entity mobility and the entity identifiers for switching entities. The CIP header is located between the CSP and the application level header. It is used for identifying streams and applications. The stream identifiers allow for virtual port numbering on top of the entity. The application identifiers allow for selecting or switching applications when it makes sense in a communication.

The definition and implementation of the CLOAK additional protocols (CSP and CIP) and their corresponding headers enable our architecture to solve NAT issues because applications using CLOAK will not use IP addresses and ports numbers for setting up or managing connections. They will use unique permanent entity identifiers, thus restoring the end-to-end principle of the Internet communications. The CLOAK architecture will also solve firewall issues because any type and any number of transport layer connections can be used to connect a CLOAK overlay. A transport layer connection can act as a multiplex tunnel for the applications using CLOAK. Thus on a given device, the applications can even use only a single port number and a single transport protocol if this is required by the firewall of the device. Indeed, a CLOAK packet has a session ID field and two application ID fields that enable numerous applications to be multiplexed on a single transport connection if necessary. CLOAK also solves security issues because the security protocols can create security associations by using entity identifiers instead of IP addresses. The security is then by design independent from the devices and applications involved.

Fig. 8 shows the modules composing the CLOAK middleware. We can see that many are needed to enable the proper functioning of the CLOAK architecture. The functionality provided by each module is briefly described below:

- Bootstrap: primitives for creating a new or joining an existing CLOAK overlay.
- Link: primitives for managing overlay links (i.e., transport layer connections) with the neighbor peers.
- Address: primitives for obtaining an overlay address from an addressing tree parent and for distributing overlay addresses to the addressing tree children.
- Route: primitives for greedily routing the overlay packets with the hyperbolic distance metric.
- Steer: primitives for rerouting overlay packets by using their device or entity identifiers to update their overlay destination address.
- Connect: primitives for establishing and managing overlay virtual connections (i.e., CLOAK layer connections) to other entities.
- Bind: primitives for querying the DHT of the overlay.
- Name: primitives for managing the identifiers used by the peer.
- Interact: primitives for managing the bindings between the data streams and the applications.

For not overwhelming the paper with too much details, the functioning of the bootstrap, steering and interacting modules will be done in a future work.

V. Simulations

In this section, we present the preliminary results of the simulations that we have carried out to establish a proof-of-concept of our dynamic P2P overlay architecture. We have used our packet driven discrete event network simulator called *nem* [11] for obtaining all the results shown in this paper.

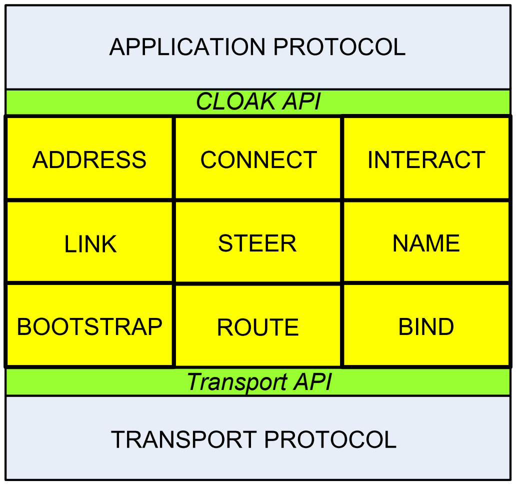


Figure 8. Modules of the middleware.

A. Parameters

In order to evaluate our overlay system on a realistic topology, we have used a 4k-node IP level Internet map created from real data measurements with the *nec* software [12]. In all simulations, the first peer creating the overlay is always a randomly picked node of the map. We have considered that only some part of the nodes of a map at any given time are acting as overlay peers. The simulator's engine manages a simulation time and each overlay peer starts at a given time for a given duration on a random node of the map. The peer that creates the overlay remains active for all the duration of a simulation. The packets are delivered between the nodes by taking the transmission time of the links into account. Peers bootstrap by contacting the node that holds the peer that created the overlay, search for other peers to which they can connect, obtain an address from one of the peers they are connected to and send data or requests messages. This process models the birth, life and death of the overlay.

In any dynamic simulations, there is a warm up phase at the beginning and a cool down phase at the end that must both be considered as transitory regimes. Indeed, at the beginning only the creator peer exists before new peers start and join it. Similarly, at the end, all peers are gradually leaving the overlay until only the creator peer is left and then it stops. Each simulation runs for 1 hour, thus only measurements in the middle of the simulation (around 30 minutes) can be considered as representing a steady state regime. This comment must be taken into account when looking at all the plots of the graphs shown below. Indeed, most of them show a curve with a typical plateau in the middle. The most significant measurements are those located in this flat part of the plots.

The number of new peers is set to 30 per minute with random inter-arrival times set with a probability following an exponential distribution. Each peer has a random lifetime set with a probability following an exponential distribution with $\lambda = 10e - 5$ which gives a median value of 300 seconds and a 90th percentile value of 1000 seconds. As each dynamic simulation lasts for 1 hour, this distribution of the peers' session lengths produces a lot of churn. The peers create overlay links with other peers by selecting those which are closer in terms of network hops. Finally, we collect measurements every 600 seconds.

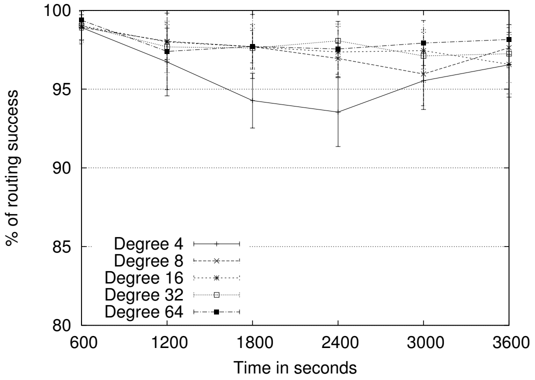


Figure 9. Average routing success rate

B. Results

We evaluate here the performances of the overlay routing depending on the chosen fixed addressing tree degree as explained in III-B. Data packets are sent by each peer at a rate of 1 every 10 seconds. We only want to evaluate routing success, query success and path lengths but not bandwidth or throughput for now that is why we do not use more realistic generated traffic patterns. The routing success rate for a given peer is equal to the number of data packets properly received by their destinations divided by those sent by the peer. Each point shown on the following graphs is the average value of 20 runs, and the associated standard deviation values are plotted as error bars. We observe the average routing success rate, the average path length and the 90th percentile path length as a function of the addressing tree degree of the overlay. In Fig. 9, we can see that the routing success rate is always above 90% which confirms the proper functioning of our system which maintains a high routing success rate despite the churn.

Fig. 10 shows the average path length of the hyperbolic routing. The path length is measured as the number of IP hops covered by the packet from the source peer to the destination peer. We can see that values are larger than the ones measured in the static simulations because here only a subset of the nodes are peers belonging to the overlay thus statistically increasing the distances. In the static simulations, the paths from all pairs were evaluated and the overlay topology was the same as the map itself. Here the nodes form an overlay which may have a different topology and thus lower path length optimality. This remains true even though overlay peers always try to establish overlay links to hop-wise closer peers.

Fig. 11 shows the 90th percentile value of the path length. Here also, the path length is measured as the number of IP hops covered by the packet. This value gives an acceptable statistical upper bound on the path length by excluding ex-

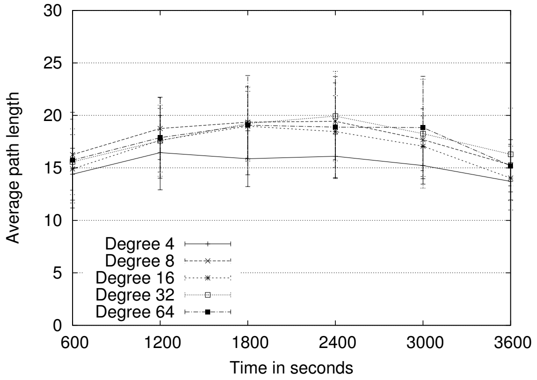


Figure 10. Average path length between peers

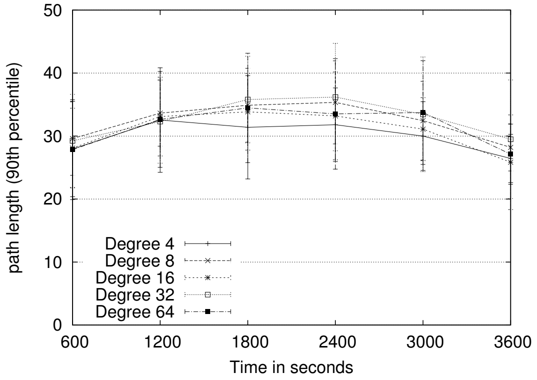


Figure 11. 90th percentile path length between peers

treme cases. We can observe that the path length, for degrees above 4, is around 35 compared to the average path length of 18 seen in fig. 10. We conclude that including the values from the median to the 90th percentile yields a path inflation of 100% which is important but still bearable.

We now evaluate the DHT efficiency. The only difference with the previous simulations is that now the peers do not send data packets but only storing and solving requests. The frequency of the storing requests generated in each peer is 1 every 30 seconds. The frequency of the solving requests generated in each peer is 1 every 5 seconds. We do not consider any redundancy parameters for now. Thus, a given pair is stored on one peer only. We observe the influence of the addressing tree degree of the overlay on the performances of the storing and the solving requests. More precisely we measure the rate of success as well as the average overlay path length of both storing and solving requests.

Fig. 12 shows the percentage of successful storing requests over the simulation duration. We assume here that only one copy of a given pair is stored in the system. We can see that given the parameters of the simulation, the rate of success is very high despite the churn.

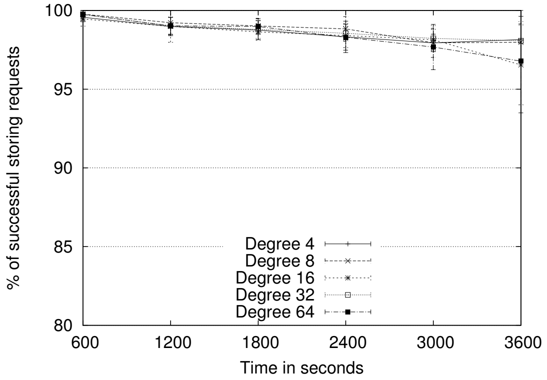


Figure 12. Percentage of successful storing requests.

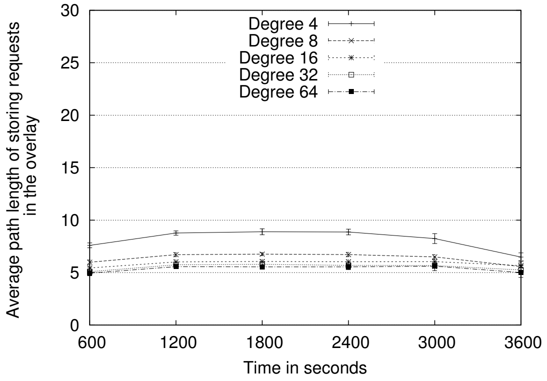


Figure 13. Average path length of the storing requests in the overlay network.

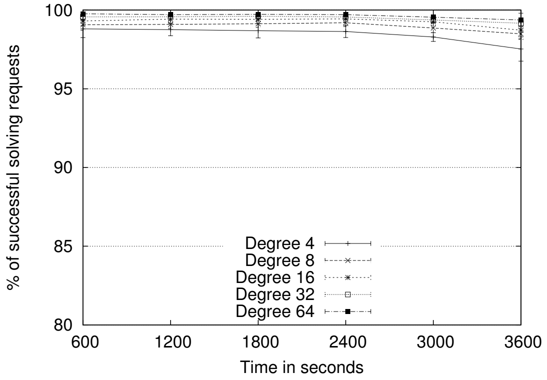


Figure 14. Percentage of successful solving requests.

Fig. 13 shows the average path length of the storing requests in the overlay network over the simulation duration. The number of peers to go through including the destination before storing a pair varies from 6 to 9 depending on the addressing tree degree. This number is decreasing when the degree is increasing with a diminishing return effect that can be seen starting at degree 16.

Fig. 14 shows the percentage of successful solving requests over the simulation duration. As for the storing request, we can see that given the parameters of the simulation, the rate of success is very high despite the churn.

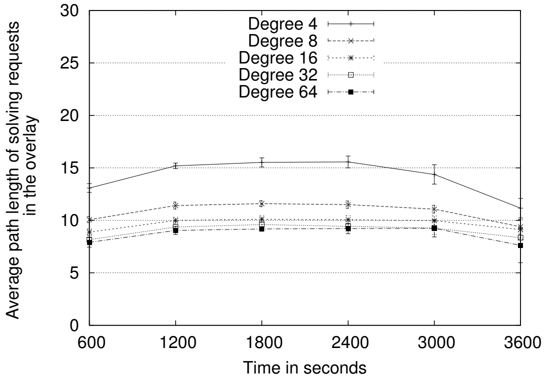


Figure 15. Average path length of the solving requests in the overlay network.

Fig. 15 shows the average path length of the solving requests in the overlay network over the simulation duration. The number of peers to go through to reach the holder of the pair and including the return trip to the sender of the request varies roughly from 9 to 16 depending on the addressing tree degree. A degree of 4 yields a typical path length of 16, a degree of 8 reduces the path length to 12 and degree values above 8 all yield path lengths between 9 and 10. Thus the number of hops is decreasing when the degree is increasing with a diminishing return effect around degree 16, similar to the storing requests path lengths of Fig. 13.

We can conclude that given those simulation results, our DHT shows encouraging performances whatever the degree chosen. The rate of success of both the storing and solving requests, for an overlay running for one hour with a total of 1800 peers, is very high. The average path lengths of the requests are also acceptable and show typical values for these kind of systems.

VI. Conclusion

In this paper, we have presented a new architecture called CLOAK designed for providing flexibility to Internet communications by using virtual connections set upon an overlay network. This architecture will be implemented as two protocols running on top of the transport protocols of the devices. The devices using the CLOAK middleware will freely interconnect with each other and thus will form a dynamic P2P overlay network. This overlay will enable the applications to maintain their communications even if some transport layer connections are subject to failures. The middleware will transparently restore the transport connections without killing the applications. The architecture, by giving identifiers to users and devices, will provide flexibility, security and mobility to applications despite the IP address changes suffered by the devices. We have implemented the overlay addressing and routing part as well as the DHT part of our middleware in a simulator and preliminary results are encouraging.

Our future work will be aimed at defining the CLOAK API, implementing the middleware as a library, modifying a relevant test application (such as a video streaming application) and testing it on a virtualized platform for studying the impact of transport layer connection pipelining created by the P2P overlay network.

References

[1] A. C. Snoeren, H. Balakrishnan, and M. F. Kaashoek, "Reconsidering ip mobility," in *Proceedings of the 8th HotOS*, 2001, pp. 41–46.

[2] A. Snoeren and H. Balakrishnan, "An end-to-end approach to host mobility," in *Proceedings of the 6th ACM MobiCom*, 2000, pp. 155–166.

[3] V. Zandy and B. Miller, "Reliable network connections," in *Proceedings of the 8th ACM MobiCom*, 2002, pp. 95–106.

[4] F. Sultan, K. Srinivasan, D. Iyer, and L. Iftode, "Migratory tcp: Connection migration for service continuity in the internet," in *Proceedings of the 22nd International Conference on Distributed Computing Systems*, 2002, pp. 469–470.

[5] D. Zagorodnov, K. Marzullo, and T. Bressoud, "Engineering fault tolerant tcp/ip services using ft-tcp," in *Proceedings of the IEEE International Conference on Dependable Systems and Networks*, 2003, pp. 393–402.

[6] T. Bressoud, A. El-Khashab, K. Marzullo, and D. Zagorodnov, "Wrapping server-side tcp to mask connection failures," in *Proceedings of the 20th IEEE INFOCOM*, 2001, pp. 329–338.

[7] R. Kleinberg, "Geographic routing using hyperbolic space," in *Proceedings of the 26th IEEE INFOCOM*, 2007, pp. 1902–1909.

[8] C. Cassagnes, T. Tiendrebeogo, D. Bromberg, and D. Magoni, "Overlay addressing and routing system based on hyperbolic geometry," in *Proceedings of the 16th IEEE Symposium on Computers and Communications*, to appear, 2011.

[9] J. Postel and J. Reynolds, "File transfer protocol (ftp)," Request For Comments 959, 1985.

[10] G. Wright and R. Stevens, *TCP/IP Illustrated, Volume 2: The Implementation*. Addison-Wesley, 1995.

[11] D. Magoni, "Network topology analysis and internet modelling with nem," *International Journal of Computers and Applications*, vol. 27, no. 4, pp. 252–259, 2005.

[12] D. Magoni and M. Hoerdt, "Internet core topology mapping and analysis," *Computer Communications*, vol. 28, no. 5, pp. 494–506, 2005.

Connectivity Services Management in Multi-domain Content-Aware Networks for Multimedia Applications

Eugen Borcoci, Mihai Stanciu, Dragoş Niculescu
Telecommunications Dept.
University POLITEHNICA of Bucharest
Bucharest, Romania
e-mails: {eugenbo, ms, dniculescu}@elcom.pub.ro

Daniel Negru
CNRS-LaBRI Lab.
University of Bordeaux, France
e-mail:daniel.negru@labri.fr

George Xilouris
NCSR Demokritos
Institute of Informatics and Telecommunications
Athens, Greece
e-mail: xilouris@iit.demokritos.gr

***Abstract*—This paper proposes a new framework, for connectivity services management in overlay Virtual Content Aware Networks (VCAN) built over multi-domain, multi-provider IP networks. The framework is part of a Future Internet-oriented Multimedia networked architecture developed inside a FP7 European ICT research project, ALICANTE. The VCAN new concept is a stronger coupling between network and applications. The VCANs are managed by CAN Providers, and the high level services by Service Providers (SP). The CANP offers to SPs enhanced connectivity services, including unicast, multicast, in a multi-domain networking context. The management framework is based on vertical and horizontal Service Level Agreements(SLA) negotiated and concluded between providers and possibly also on content/service description information (metadata) inserted in the media flow packets by the servers.**

Keywords—Content-Aware Networking; Network Aware Applications; Connectivity services; Management; Multimedia distribution; Future Internet

I. INTRODUCTION

The Future Internet has a strong orientation towards services and content, [1][2][3]. A new solution to make the Future Internet more content oriented [3][4][5][6], is to create virtualized *Content Aware Networks* (CAN) and *Network Aware Applications* (NAA) on top of the flexible IP. Additionally to routing, the CAN routers are optimized for filtering, forwarding, and transforming inter-application messages on the basis of their content and context.

The work of this paper is part of an activity performed in the framework of a new European FP7 ICT research project, "Media Ecosystem Deployment Through Ubiquitous Content-Aware Network Environments", ALICANTE [7][8][9]. The following inter-working multi-actor environments are defined: *User Environment (UE)*, to which some end users belong; *Service Environment (SE)*, to which Service Providers (SP) and Content Providers (CP) belong; *Network Environment (NE)*, to which the Network Providers (NP) belong. *Environment* is a generic name for a grouping of functions defined around the same common goal and which possibly vertically span one or more several architectural layers.

We propose a new framework, for connectivity services management in overlay VCANs built over multi-domain, multi-provider IP networks. The VCANs are managed by a CAN Provider (CANP), and the high level services by SPs. The CANP offers to SPs enhanced connectivity services including unicast and multicast. The management framework is based on vertical and horizontal SLAs negotiated and concluded between providers and possibly also on content/service description information (metadata) inserted in the media flow packets by the servers.

The paper continues the starting work on VCAN management presented in [8][10]. It is organized as follows. Section II presents samples of related work. Section III summarizes the overall ALICANTE architecture. Section IV presents the content awareness features of the system and QoS assurance solutions. Section V describes the peering approach to extend a VCAN over several domains. The proposed CAN management architecture and functionalities is presented in Section VI. Section VII contains some conclusions and future work outline.

II. RELATED WORK

A higher coupling between the Application and Network layers was recently proposed in order to make the IP network more adapted to content and services. In the framework of rethinking the architecture of the Future Internet, the concepts of CAN and NAA are proposed. CAN adjusts network layer processing based on limited examination of

Copyright (c) IARIA, 2011. ISBN: 978-1-61208-141-0

the nature of the content, and NAA implies processing the content based on limited understanding of the network conditions. The work presented in [1] emphasizes the strong orientation of the FI towards content and services and shows the importance of management. CAN/ NAA can offer a way of evolution of networks beyond IP, as presented in [6]. The implementation of such an approach can be supported by virtualization as a strong method to overcome the ossification of the current Internet [2][3][4][5].

The work in [11] discusses the content adaptation issues in the FI as a component of CAN/NAA approach. The CAN/NAA approach can also offer QoE (Quality of Experience) and QoS capabilities of the future networks, [6][12]. Context awareness is added to content awareness in [13]. However, the CAN approach requires a higher amount of packet header processing, similar to deep packet inspection techniques. The CAN/NAA approach can also help to solve the current networking problems related to the P2P traffic overload of the global Internet [14]. The Application Layer Traffic Optimization (ALTO) problem studied by IETF can be solved by the cooperation between the CAN layer and the upper layer. The management architecture of the CAN/NAA oriented networks is still an open research issue.

III. ALICANTE SYSTEM ARCHITECTURE

The main concepts and general ALICANTE architecture are defined in [7][8][9]. The business model is defined, composed of traditional SP, CP, NP - Providers and End-Users (EU). A new actor is the CAN Provider (CANP) offering virtual layer connectivity services. A new entity is also defined: Home-Box (HB)- partially managed by the SP, the NP, and the end-user, located at end-user's premises and gathering content/context-aware and network-aware information. The HB can also act as a CP/SP for other HBs, on behalf of the EUs. Two novel virtual layers exist: the CAN layer for network level packet processing and the HB layer for the actual content delivery, working on top of IP. The virtual CAN routers are called Media-Aware Network Elements (MANE) to emphasize their additional capabilities: content and context - awareness, controlled QoS/QoE, security and monitoring features, etc.

The SE [8] uses information from the CAN layer to enforce NAA procedures, in addition to user context-aware ones. Apart from VCANs provisioning, per flow adaptation can be deployed at both HB and CAN layers, as additional means for QoS, by making use of scalable media resources.

The management and control of the CAN layer is partially distributed; it supports CAN customization as to respond to the upper layer needs, including 1:1, 1:n, and n:m communications, and also allow efficient network resource exploitation. The rich interface between CAN and the upper layer allows cross-layer optimizations interactions, e.g., including offering distance information to HBs to help collaboration in P2P style. At all levels, monitoring is performed in several points of the service distribution chain and feeds the adaptation subsystems with appropriate information, at the HB and CAN Layers. Fig. 1 presents a partial view on the ALICANTE architecture, with emphasis on the CAN layer and management interaction. The network contains several NP domains (Autonomous Systems - AS) and access networks (AN). Each domain has an Intra-domain Network Resource Manager (IntraNRM), as the authority configuring the network nodes. The CAN layer cooperates with HB and SE by offering them CAN services. One CAN Manager (CANMgr) exists for each IP domain to assure the consistency of CAN planning, provisioning, advertisement, offering, negotiation installation and exploitation. However, autonomous CAN-like behavior of the MANE nodes can be also offered in a distributed way by processing individual flows.

The following contracts/interactions of SLA/SLS types performed in the Management and Control Plane and the appropriate interfaces are shown in Fig. 1:

SP-CANP(1): the SP requests to CANP to provision/ modify/ terminate new VCANs and the CANP to inform SP about its capabilities; *CANP-NP(2)* - through which the NP offers or commits to offer resources to CANP (this data is topological and capacity-related); *CANP-CANP(3)* - to extend a VCAN upon several NP domains; - *Network Interconnection Agreements (NIA) (4)* between the NPs or between NPs and ANPs; these are not new ALICANTE functionalities but are necessary for NP cooperation.

After the SP negotiates a desired VCAN with CANP, it will issue the installation commands to CANP, which in turn configures via IntraNRM (5) the MANE functional blocks (input and output).

IV. CONTENT AWARENESS AND QOS ASSURANCE AT CAN LAYER

The content awareness (CA) is realized in three ways:

(i) by concluding an SLA between SP and CANP, concerning different VCAN construction. The content servers are instructed by the SP to insert some special Content Aware Transport Information (CATI). This simplifies the media flow classification and treatment by the MANE.

(ii) the SLA is concluded, but no CATI information is inserted in the data packets. The MANE applies deep packet inspection for data flow classification and assignment to VCANs. The treatment of the flows is based on VCANs characteristics defined in the SLA.

(iii) no SLA exists between SP and CANP. No CATI is inserted in the data packets. The treatment of the data flows can still be CA, but conforming to the local policy established at CANP and IntraNRM.

An important issue related to multimedia flow transportation is the QoS assurance. The DiffServ philosophy can be applied to split the sets of flows in QoS classes (QC), with a mapping between the VCANs and the QCs.

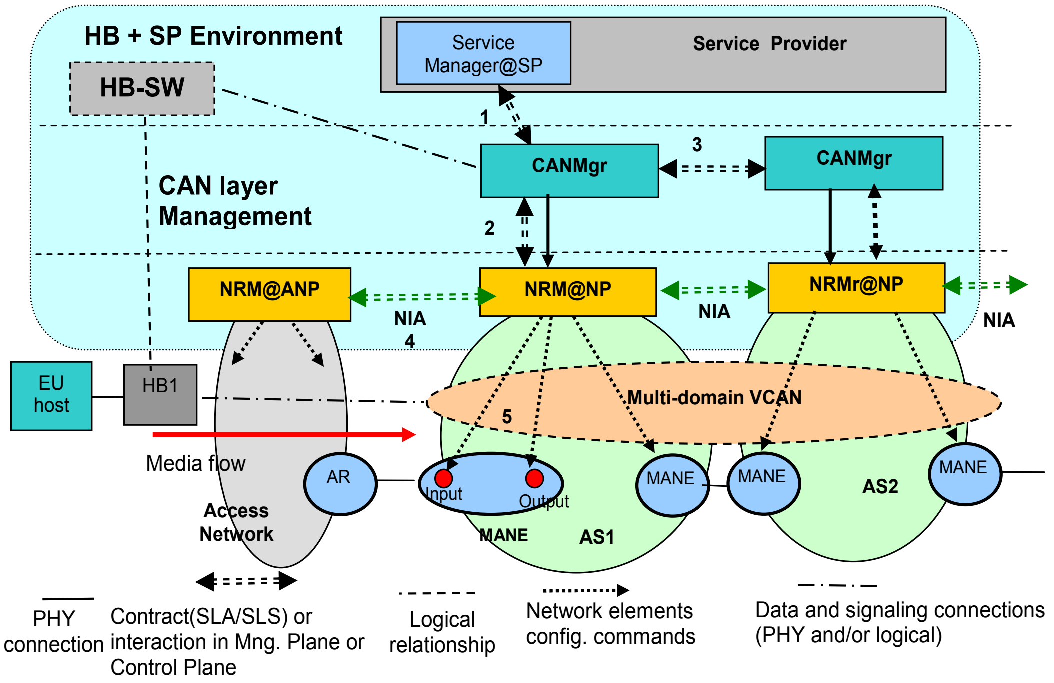


Figure 1. ALICANTE architecture: CAN management interactions

Several levels of QoS granularity can be established when defining VCANs. The QoS behavior of each VCAN is established inside the SLA between SP and CANP.

Actually, the CAN layer may offer to the SP, several parallel internets (PI) [15], specialized in different types of application content. We adopt the PI concept, enriching it with content awareness. A PI enables end-to-end service differentiation across multiple administrative domains. The PIs can coexist, as parallel logical networks composed of interconnected, per-domain, Network Planes. A given plane is defined to transport traffic flows from services with common connectivity requirements. The traffic delivered within each plane receives differentiated treatment, so that service differentiation across planes is enabled in terms of edge-to-edge QoS, availability and also resilience.

In ALICANTE, generally a one-to-one mapping between a VCAN and a network plane will exist. Specialization of CANs may exist in terms of QoS level of guarantees (weak or strong), QoS granularity, content adaptation procedures, degree of security, etc. A given network plane or VCAN can be realized by the CANP, by combining several processes, while being possible to choose different solutions concerning some dimensions: route determination, data plane forwarding, packet processing, and resource management.

The definitions of local QoS classes (QC) and extended QCs were adopted, to allow us to capture the notion of QoS capabilities across several domains [16][17][18]. For a simplified design, we also used the concept of Meta-QoS-Class [16]. A meta QC captures a common set of QoS ranges of parameters spanning several domains. It relies on a worldwide common understanding of application QoS needs. Foir example, VoD service flows need similar QoS characteristics whatever AS they transit. The meta QC concept offers the advantage that the existence of well known classes greatly simplifies the inter-domain signaling in the sequence of actions needed to establish domain peering in the multi-domain context. This concept simplifies the peering of different domains inside the same VCAN.

The types of VCANs for different QoS granularities based on QCs are described in [9]. In short, the following use cases have been defined for multi-domain VCANs: VCANs based on meta-QC, VCANs based on local QC composition, hierarchical CANs based on local QC composition.

The last case is the most efficient but also the most complex. Each domain may have its local QoS classes. Several local QCs can be combined to form an extended QC. Inside each CAN, several QCs are defined corresponding to platinum, gold, silver, etc. In such a case, the mapping between service flows at SP level and CANs can be done per type of the service: VoD, VoIP, Video-conference, etc.

V. CAN MULTI-DOMAIN PEERING

A given CAN may span one or several IP domains. Thus a peering problem appears: how to determine the intermediate and terminal domains to be chained in the resultant VCAN. The hub model is proposed, in which a CAN Manager is communicating with other managers in order to establish the multi-domain CANs.

A drawback is that each CAN Manager should know the complete graph of AS candidates to participate in every possible VCAN (overhead). However, given that the number of ASes involved in a VCAN communication cannot be high, and that they can be localized in an Internet region, the scalability problem is not so stringent.

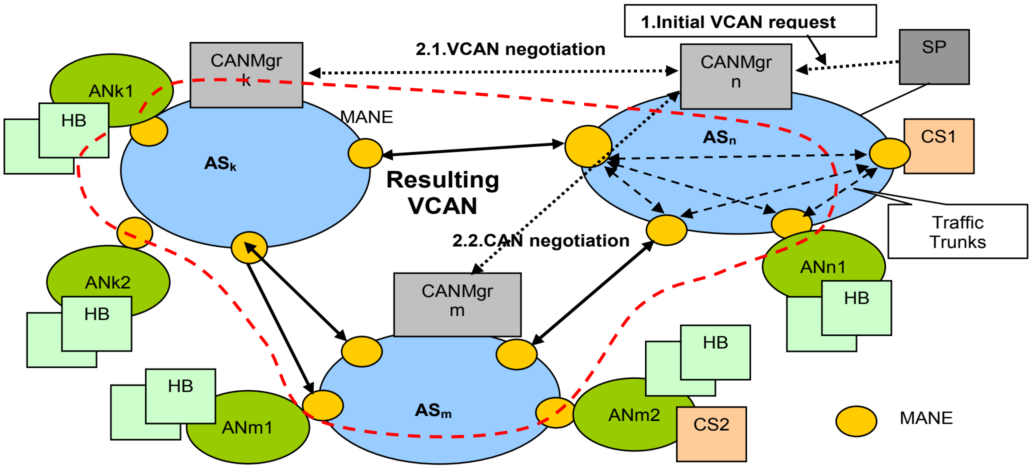


Figure 2. Example of a multi-domain CAN (hub model)

The initiator CAN Manager should discuss/negotiate with all other CAN Managers in order to establish the "international" VCAN = {CAN1 **U** CAN2 **U** CAN3 …}. The split of the SLS parameters should be done at the initiator (e.g. for delay).

Fig. 2 shows an example of a multi-domain VCAN. The infrastructure is composed of three domains AS*n*, AS*k*, AS*m*, each having a CAN Manager. Several Access Networks are connected to these domains, containing Home-Boxes or/and Content Servers (CS). The latter are controlled by the SP. The SP is requesting a CANMng *n* to construct a CAN, spanning several domains, i.e. AS*n*, AS*k* and AS*m*. It is supposed that the SP knows the edge points of this CAN, the MANEs where different sets of HB currently are, or they will be connected. Based on its inter-domain routing information, the CANMng_*n* determines that the components of the VCANs are AS*n*, AS*k*, AS*m*. Therefore, it negotiates in actions 2.1 and 2.2 the appropriate VCAN capabilities with CANMng_*k* and respectively CANMng_*m*. In a successful scenario, the multi-domain CAN is agreed and then it is instantiated in the network.

VI. CAN Resource Manager Architecture at Service Provider and CAN Provider

Fig. 3 presents the proposed architecture for CAN Management. This is a continuation and development of the one presented in [10]. At the Service Manager SM@SP level, the CAN Network Resources Manager (CAN_RMgr) component performs all the actions needed to assure the CAN support to the SP, in order to deploy its high level services in unicast or multicast mode. It is responsible to negotiate with CANP on behalf of the SP and perform all actions necessary for *VCAN* planning, VCAN provisioning and VCAN operation.

CNMgr@CANP performs at the CAN layer all actions related to VCAN provisioning and operation. The two entities interact based on the SLA contract initiated by the SP. The technical part of these contracts is the Service Level Specification (SLS).

Several points of view should be considered when defining/planning the services, planning the CAN and respectively when defining CAN_RMgr functionalities: the commercial optimization needs of the SP, CANP resources, CAN network engineering and implementation.

The CAN_RMgr interacts with the following modules supposed to exist and belonging to the SM:

Service Forecast and Planning - an *offline process* performing service predictions and their associated plans of deployment, considering the business as input.

Service Deployment Policy - can contain (in a data base) predefined rules for service planning. This information is derived from the high-level business interests of the SP and significantly influences the planning.

CAN_RMgr@SM contains the following functional blocks: CAN Planning, CAN Provisioning and CAN Operation and Maintenance, as main functional blocks. A CAN Repository data base keeps all data related to VCAN provisioning, installation and current status. Policies can intervene to guide the other blocks through the module *CAN Deployment and Operation Policies*.

Fig. 3 also shows the interfaces, defined below. Where possible, the interface implementation for data transport will be based on SOAP/Web Services interfaces, used for SOAP requests and responses.

1. CAN Planning at *CAN-RMgr@SM* - to - Service Forecast and *Planning@SM* at Service Life Cycle block. This input interface to CAN_RMgr delivers information from the service forecast module and from the policy block, to allow the high level CAN Planning.

2. CAN Operation and Maintenance at *CAN-RMgr@SM* - to - Service Life Cycle block. This interface delivers the current status data on active CANs to the Service Life Cycle block.

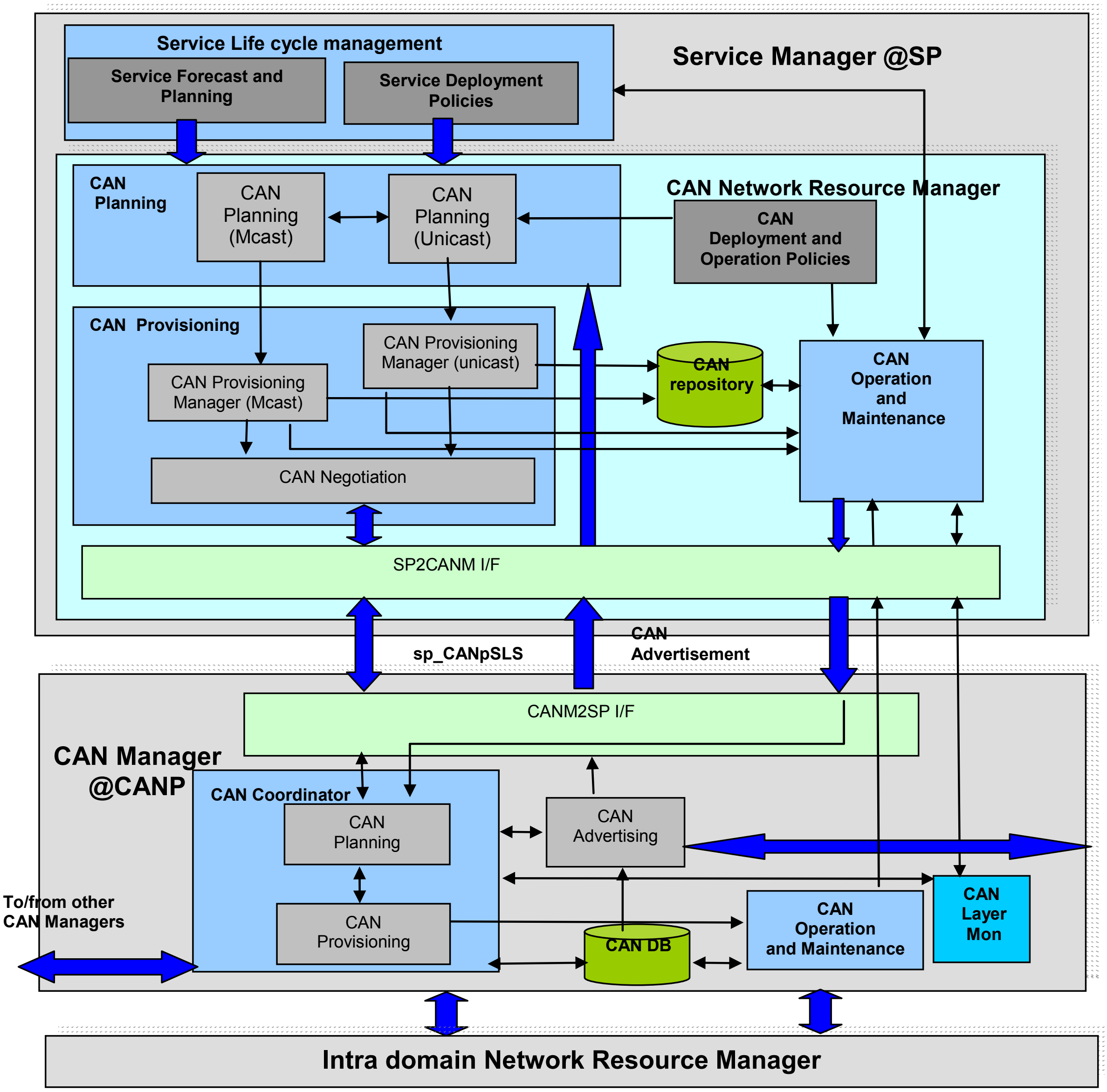

Service Manager @SP
Service Life cycle management
Service Forecast and Planning
Service Deployment Policies
CAN Network Resource Manager
CAN Planning
CAN Planning (Mcast)
CAN Planning (Unicast)
CAN Deployment and Operation Policies
CAN Provisioning
CAN Provisioning Manager (unicast)
CAN Provisioning Manager (Mcast)
CAN repository
CAN Operation and Maintenance
CAN Negotiation
SP2CANM I/F
sp_CANpSLS
CAN Advertisement
CANM2SP I/F
CAN Manager @CANP
CAN Coordinator
CAN Planning
CAN Provisioning
CAN Advertising
CAN DB
CAN Operation and Maintenance
CAN Layer Mon
To/from other CAN Managers
Intra domain Network Resource Manager

Figure 3. Architecture of the CAN Network Resource Manager at SP and CAN Manager at CAN layer

3. CAN-RMgr@SM – to – CAN Manager. This is a multiple interface necessary for CAN_RMgr at SM@SP to perform the following:

- request the CAN Manager to negotiate VCANs and perform negotiation (SLS contracts will be concluded for VCAN subscription, based on a negotiation protocol);
- command VCAN installation (invocation)
- receive advertisement information about available CANs constructed at the CANP's initiative
- request modification and/or termination of a CAN: according to the current situation and the evolution of the forecast, the SP can re-negotiate the network resources with CANP, which will imply to add/modify/delete VCANs;
- receive status and monitoring information about the active VCANs.

A. *CAN Provisioning*

The functional block for this is the CAN Provisioning Manager at SM@SP. The *CANProvMng@SM* has several main functions shortly presented below.

It performs all *sp_CANpSLS* processing - subscription (unicast/multicast mode) in order to assure the CAN transport infrastructure for the SP. For CAN subscription, the *CANProvMng@SM* receives requests for a *sp_CANpSLS* contract dedicated to a given CAN from *CAN Planning*. Then, it requests to the CAN Manager associated with its home domain, to subscribe for a new CAN. It negotiates the subscription and concludes an SLS denoted by: *SP-*

CAN_SLS-uni_sub for unicast, or *SP-CAN_SLS-mc_sub for* multicast. The results of the contract are stored in the *CAN repository.* Note that CAN subscription only means a logical resource reservation at the CAN layer, not real resource allocation and network node configuration.

The CAN subscription action may or may not be successful, depending on the amount of resources demanded by the SP and the available resources in the network. Note that at its turn the CAN Manager has to negotiate the CAN subscription with IntraNRM, and overbooking is an option, depending on the SP policy.

B. CAN Negotiation

The basic version of this negotiation protocol (SP-CANP-SLS-P) can support the negotiation process between several pairs of managers like *CANProvMng@SM* as a client and *CAN Manager* as a server. The main usage of this protocol is for establishing *sp_CANpSLSs* contracts, but it should have all the necessary properties of a general negotiation protocol, and could be adapted/used to serve CAN invocation.

The SP-CANP-SLS-P protocol is a client-server, 1-to-1 protocol, where the client initiates the negotiation sessions. In order to be able to serve the c/pSLS negotiations, it is completely distinct from the negotiation logic, which is located a layer above the protocol and acts as a user of the protocol. For SLS negotiation between SP/CANP, the logic is represented by the combined roles of: Service Provisioning and an SLS Request blocks at the client-side; SLS Request Handler and SLS Subscription Admission Control blocks as the server-side. The SP-CANP-SLS-P supports one of several negotiation actions: establishment/ modifications/ termination of SLS contracts.

The management system described is currently in the design phase in ALICANTE project. Validation and implementation results will be published in the near future.

VII. Conclusions and Future Work

The paper proposed an architectural solution for connectivity services management, in Content Aware Networks for a multi-domain and multi-provider environment. The management is based on vertical and horizontal SLAs negotiated and concluded between providers (SP, CANP, NP), the result being a set of parallel VCANs offering different classes of services to multimedia flows, based on CAN/NAA concepts. The approach is to map the QoS classes on virtual data CANs, thus obtaining several parallel QoS planes. The system can be incrementally built by enhancing the edge routers functionalities with content awareness features. Further work is going on to design and implement the system in the framework of the FP7 research project ALICANTE. A preliminary implementation and performance evaluation will appear in [19].

Acknowledgments

This work was supported partially by the EC in the context of the ALICANTE project (FP7-ICT-248652) and partially by the project POSDRU/89/1.5/S/62557.

References

[1] Schönwälder, J., Fouquet, M., Dreo Rodosek, G., and Hochstatter, I.C., "Future Internet = Content + Services + Management", IEEE Communications Magazine, vol. 47, no. 7, Jul. 2009, pp. 27-33.

[2] Baladrón, C., "User-Centric Future Internet and Telecommunication Services", in: G. Tselentis, et. al. (eds.), Towards the Future Internet, IOS Press, 2009, pp. 217-226.

[3] Turner, J. and Taylor, D., "Diversifying the Internet," Proc. GLOBECOM '05, vol. 2, St. Louis, USA, Nov./Dec. 2005, pp. 760-765

[4] Anderson, T., Peterson, L., Shenker, S., and Turner, J., "Overcoming the Internet Impasse through Virtualization", Computer, vol. 38, no. 4, Apr. 2005, pp. 34–41.

[5] Chowdhury, N. M. and Boutaba, R., "Network Virtualization: State of the Art and Research Challenges", IEEE Communications Magazine, vol. 47, no.7, Jul. 2009, pp. 20-26.

[6] Kourlas, T., "The Evolution of Networks beyond IP", IEC Newsletter, vol. 1, Mar. 2007. Available at http://www.iec.org/newsletter/march07_1/broadband_1.html (last accessed: Mar. 2010).

[7] FP7 ICT project, "MediA Ecosystem Deployment Through Ubiquitous Content-Aware Network Environments", ALICANTE, No248652, http://www.ict-alicante.eu/ (last accessed: Dec. 2010).

[8] Borcoci, E., Negru, D. and Timmerer, C., "A Novel Architecture for Multimedia Distribution based on Content-Aware Networking" Proc. of. CTRQ 2010, Athens, June 2010, pp. 162-168

[9] ALICANTE, Deliverable D2.1, ALICANTE Overall System and Components Definition and Specifications, http://www.ict-alicante.eu, Sept. 2010

[10] Borcoci, E. and Iorga, R., "A Management Architecture for a Multi-domain Content-Aware Network" TEMU 2010, July 2010, Crete.

[11] Zahariadis, T., Lamy-Bergot, C., Schierl, T., Grüneberg, K., Celetto, L., and Timmerer, C., "Content Adaptation Issues in the Future Internet", in: G. Tselentis, et al. (eds.), Towards the Future Internet, IOS Press, 2009, pp. 283-292.

[12] Liberal, F., Fajardo, J.O., and Koumaras, H., "QoE and *-awareness in the Future Internet", in: G. Tselentis, et al. (eds.), Towards the Future Internet, IOS Press, 2009, pp. 293-302.

[13] Baker, N., "Context-Aware Systems and Implications for Future Internet", in: G. Tselentis et. al. (eds.), Towards the Future Internet, IOS Press, 2009, pp. 335-344.

[14] Aggarwal, V., Feldmann, A., "Can ISPs and P2P Users Cooperate for Improved Performance?", ACM SIGCOMM Computer Communication Review, vol. 37, no. 3, Jul. 2007, pp. 29-40.

[15] Boucadair, M. et al., "A Framework for End-to-End Service Differentiation: Network Planes and Parallel Internets", IEEE Communications Magazine, Sept. 2007, pp. 134-143

[16] Levis, P., Boucadair, M., Morrand, P., and Trimitzios, P., "The Meta-QoS-Class Concept: a Step Towards Global QoS Interdomain Services", Proc. of IEEE SoftCOM, Oct. 2004.

[17] Howarth, M.P. et al., "Provisioning for Interdomain Quality of Service: the MESCAL Approach", IEEE Communications Magazine, June 2005, pp. 129-137

[18] MESCAL D1.2: "Initial Specification of Protocols and Algorithms for Inter-domain SLS Management and Traffic Engineering for QoS-based IP Service Delivery and their Test Requirements", January 2004, www.mescal.org (last accessed: Dec 2010)

[19] Dragoş S. Niculescu, Mihai Stanciu, Marius Vochin, Eugen Borcoci, Nikolaos Zotos, Implementation of a Media Aware Network Element for Content Aware Networks, CTRQ 2011, to appear.

Class of Trustworthy Pseudo-Random Number Generators

Jacques M. Bahi*, Jean-François Couchot*, Christophe Guyeux*and Qianxue Wang*
*University of Franche-Comte
Computer Science Laboratory LIFC, Belfort, France
Email:{jacques.bahi, jean-francois.couchot, christophe.guyeux, qianxue.wang}@univ-fcomte.fr

***Abstract*—With the widespread use of communication technologies, cryptosystems are therefore critical to guarantee security over open networks as the Internet. Pseudo-random number generators (PRNGs) are fundamental in cryptosystems and information hiding schemes. One of the existing chaos-based PRNGs is using chaotic iterations schemes. In prior literature, the iterate function is just the vectorial boolean negation. In this paper, we propose a method using Graph with strongly connected components as a selection criterion for chaotic iterate function. In order to face the challenge of using the proposed chaotic iterate functions in PRNG, these PRNGs are subjected to a statistical battery of tests, which is the well-known NIST in the area of cryptography.**

***Keywords*-Internet security; Chaotic sequences; Statistical tests; Discrete chaotic iterations.**

I. Introduction

Chaos and its applications in the field of secure communication have attracted a lot of attention in various domains of science and engineering during the last two decades. The desirable cryptographic properties of the chaotic maps such as sensitivity to initial conditions and random behavior have attracted the attention of researchers to develop new PRNG with chaotic properties. Recently, many scholars have made an effort to investigate chaotic PRNGs in order to promote communication security [5] [10] [14]. One of the existing chaos-based PRNGs is using chaotic iterations schemes.

A short overview of our recently proposed PRNGs based on Chaotic Iterations are given hereafter. In Ref. [1], it is proven that chaotic iterations (CIs), a suitable tool for fast computing iterative algorithms, satisfies the topological chaotic property, as it is defined by Devaney [7]. The chaotic behavior of CIs is exploited in [2], in order to obtain an unpredictable PRNG that depends on two logistic maps. The resulted PRNG shows better statistical properties than each individual component alone. Additionally, various chaos properties have been established. The advantage of having such chaotic dynamics for PRNGs lies, among other things, in their unpredictability character. These chaos properties, inherited from CIs, are not possessed by the two inputted generators. We have shown that, in addition of being chaotic, this generator can pass the NIST battery of tests, widely considered as a comprehensive and stringent battery of tests for cryptographic applications [13]. Then, in the papers [3], [4], we have achieved to improve the speed of the former PRNG by replacing the two logistic maps: we used two XORshifts in [3], and ISAAC with XORshift in [4]. Additionally, we have shown that the first generator is able to pass DieHARD tests [11], whereas the second one can pass TestU01 [9].

In spite of the fact that all these previous algorithms are parametrized with the embed PRNG, they all iterate the same function namely, the vectorial boolean negation later denoted as $\neg$. It is then judicious to investigate whether other functions may replace the $\neg$ function in the above approach. In the positive case, the user should combine its own function and its own PRNGs to provide a new PRNG instance. The approach developed along these lines solves this issue by providing a class of functions whose iterations are chaotic according to Devaney and such that resulting PRNG success statistical tests.

The rest of this paper is organized in the following way. In the next section, some basic definitions concerning CIs are recalled. Then, our family of generators based on discrete CIs is presented in Section III with some improvements. Next, Section IV gives a characterization of functions whose iterations are chaotic. A practical note presents an algorithm allowing to generate some instances of such functions. These ones are then embedded in the algorithm presented in Sect. V where we show why generator of Sect. III is not convenient for them. In Section VI, various tests are passed with a goal to decide whether all chaotic functions are convenient in a PRNG context. The paper ends with a conclusion section where our contribution is summarized and intended future work is presented.

II. Discrete Chaotic Iterations: recalls

Let us denote by $[\![a;b]\!]$ the interval of integers: $\{a, a+1, \ldots, b\}$. A boolean system (BS) is a collection of n components. Each component $i \in [\![1;n]\!]$ takes its value x_i among the domain $\mathbb{B} = \{0,1\}$. A *configuration* of the system at discrete time t (also called at *iteration* t) is the vector $x^t = (x_1^t, \ldots, x_n^t) \in \mathbb{B}^n$.

The dynamics of the system is described according to a function $f : \mathbb{B}^n \to \mathbb{B}^n$ such that: $f(x) = (f_1(x), \ldots, f_n(x))$.

Let be given a configuration x. In what follows the configuration $N(i,x) = (x_1, \ldots, \overline{x_i}, \ldots, x_n)$ is obtained by switching the $i-$th component of x. Intuitively, x and $N(i,x)$ are neighbors. The discrete iterations of the f function are represented by the so called graph of iterations.

Definition 1 (Graph of iterations) *In the oriented* graph of iterations, *vertices are configurations of* $\mathbb{B}^n$ *and there is an arc labeled* i *from* x *to* $N(i,x)$ *iff* $f_i(x)$ *is* $N(i,x)$ *(we consider 1-bit transitions).*

In the sequel, the *strategy* $S = (S^t)^{t\in\mathbb{N}}$ is the sequence of the components that may be updated at time t, S^t denotes the $t-$th term of the strategy S.

Let us now introduce two important notations. Δ is the *discrete Boolean metric*, defined by $\Delta(x,y) = 0 \Leftrightarrow x = y$, and the function F_f is defined for any given application $f : \mathbb{B}^n \to \mathbb{B}^n$ by

$$\begin{array}{rcl} F_f : [\![1;n]\!] \times \mathbb{B}^n & \to & \mathbb{B}^n \\ (s,x) & \mapsto & \Big(x_j.\Delta(s,j) + f_j(x).\overline{\Delta(s,j)}\Big)_{j\in[\![1;n]\!]}, \end{array}$$

where the point and the line above delta are multiplication and negation respectively. With such a notation, configurations are defined for times $t = 0, 1, 2, \dots$ by:

$$\left\{ \begin{array}{l} x^0 \in \mathbb{B}^n \text{ and} \\ x^{t+1} = F_f(S^t, x^t) \end{array} \right. \tag{1}$$

Finally, iterations of (1) can be described by the following system

$$\left\{ \begin{array}{l} X^0 = ((S^t)^{t\in\mathbb{N}}, x^0) \in [\![1;n]\!]^{\mathbb{N}} \times \mathbb{B}^n \\ X^{k+1} = G_f(X^k), \end{array} \right. \tag{2}$$

such that

$$G_f\left((S^t)^{t\in\mathbb{N}}, x\right) = \left(\sigma((S^t)^{t\in\mathbb{N}}), F_f(S^0, x)\right),$$

where σ is the function that returns the strategy $(S^t)^{t\in\mathbb{N}}$ where the first term (*i.e.*, S^0) has been removed. In other words, at the t^{th} iteration, only the S^t-th cell is modified; the resulting strategy is the initial one where the first t terms have been removed.

A previous work [1] has shown a fine metric space such that iterations of the map G_f are chaotic in the sense of Devaney [7] when f is the negation function $\neg$. This definition consists of three conditions: topological transitivity, density of periodic points, and sensitive point dependence on initial conditions. Topological transitivity is established when, for any element, any neighborhood of its future evolution eventually overlap with any other given region. On the contrary, a dense set of periodic points is an element of regularity that a chaotic dynamical system has to exhibit. This regularity "counteracts" the effects of transitivity. Finally, a system is sensitive to initial conditions if future evolution of any point in its neighborhood are significantly different. This result theoretically implies the "quality" of the randomness.

The next section formalizes with chaotic iterations terms the PRNG algorithm presented in [2].

III. Chaos based PRNG

This section aims at formalizing a PRNG algorithm already presented in [2] and gives some improvements.

First of all, Let us intorduce *XORshift*, generator. Xorshift is a category of pseudorandom number generators designed by George Marsaglia [12] that repeatedly uses the transform of exclusive or on a number with a bit shifted version of itself. A XORshift operation is defined as follows.

```
Input: the internal state $z$ (a 32-bits word)
Output: $y$ (a 32-bits word)
$z \leftarrow z \oplus (z \ll 13)$;
$z \leftarrow z \oplus (z \gg 17)$;
$z \leftarrow z \oplus (z \ll 5)$;
$y \leftarrow z$;
return $y$;
```

Algorithm 1: An arbitrary round of XORshift algorithm

Then the design procedure of this generator is summed up in Algorithm 2.

```
Input: an initial state $x^0$ ($n$ bits)
Output: a state $x$ ($n$ bits)
$x \leftarrow x^0$;
$k \leftarrow \textit{reallocate}(\textit{XORshift}() \mod (2^n - 1))$;
$x \leftarrow \textit{iterate\_G}(\textit{neg}, \textit{XORshift}, k, x)$;
return $x$;
```

Algorithm 2: An arbitrary round of the (*XORshift*,*XORshift*) generator

Let be given a seed as the internal state x. This algorithm outputs a random configuration x'. It is based on the *XORshift*, generator which is called in two situations. The first one occurs while generating the parameter of the *reallocate* function that aims at computing the number k of time a function has to be iterated. The second one occurs as a parameter of *iterate_G*, which executes the iterations of G as defined in (2), with $f = neg, S = XORshift, x$ as initial state, and k for the number of iterations.

Firstly, let us focus on the *reallocate* function, which is defined by:

$$\textit{reallocate}(k) = \left\{ \begin{array}{lllll} 0 & \text{if} & 0 & \leqslant k < & \binom{n}{0} \\ 1 & \text{if} & \binom{n}{0} & \leqslant k < & \displaystyle\sum_{i=0}^{1}\binom{n}{i} \\ \vdots & & & & \vdots \\ n & \text{if} & \displaystyle\sum_{i=0}^{n-1}\binom{n}{i} & \leqslant k \leqslant & 2^n - 1 \end{array} \right.$$

Formally, the set $[\![0, 2^n - 1]\!]$ is partionned into subsets $[\![\Sigma_{i=0}^{j}\binom{n}{0}, \Sigma_{i=0}^{j+1}\binom{n}{i}[\![$ where $j \in [\![0, n-1]\!]$. Each interval bound is a binomial coefficient: it gives the number of combinations of n things taken j. In our context, it is the number of configurations $(x_1, \dots, x_n)$ that can be built by negating j elements among n. The function *reallocate* allows to compute a distribution on $[\![0, n]\!]$ that permits to reach configurations in $[\![0, 2^n - 1]\!]$ uniformly.

Let us present now the *iterate_G* function. It starts with computing the strategy S of lenght k as the result of a usual *sample* (not detailled here) function that selects k elements among n following a PRNG r given as the first parameter. The loop next reproduces k iterations of G_f as define in Equ. (2)

```
Input: a function $f$, a PRNG $r$, an iterations number $k$,
       a binary number $x^0$ ($n$ bits)
Output: a binary number $x$ ($n$ bits)
$x \leftarrow x^0$;
S = $\textit{sample}(r, k, n)$;
for $i = 0, \dots, k-1$ do
    $s \leftarrow S[i]$;
    $x \leftarrow F_f(s, x)$;
end
return $x$;
```

Algorithm 3: The *iterate_G* function.

Compared to work [2], this algorithm is:

- close to the formal iterations of G_f: strategy is explicitely computed and there are as many iterations as the number of executed loops.
- more efficient: in the previous work, loops are executed untill k distinct elements have been switched leading to possibly more iterations. In the opposite, the function *iterate_G* exactly executes k loops when k iterations are awaited. However, this improvement moves the problem into the *sample* function, which is classically tuned to

	100	10000	100000	1000000	1000000
Speed up	10%	7.8 %	8.8 %	8.1%	9.5%

Table I: Speed up improvement from Algorithm [2]

speed up its global behavior. In such a context we take a benefit of this improvement. Table I compares these two algorithms in terms of execution time with respect to the number of generated elements. The improvement is about 9%.

However as noticed in introduction, the whole (theoretical and practical) approach is based on the negation function. The following section studies whether other functions can theoretically replace this one.

IV. Characterizing and Computing Functions for PRNG

This section presents other functions that theoretically could replace the negation function $\neg$ in the previous algorithms.

In this algorithm and from the graph point of view, iterating the function G_f from a configuration x^0 and according to a strategy $(S^t)^{t \in \mathbb{N}}$ consists in traversing the directed iteration graph $\Gamma(f)$ from a vertex x^0 following the edge labelled with S^0, S^1, ... Obviously, if some vertices cannot be reached from other ones, their labels expressed as numbers cannot be output by the generator. The *Strongly connected component of* $\Gamma(f)$ (*i.e.*, when there is a path from each vertex to every other one), denoted by SCC in the following [6], is then a necessary condition for the function f. The following result shows this condition is sufficient to make iterations of G_f chaotic.

Theorem 1 (Theorem III.6, p. 91 in [8]) *Let f be a function from $\mathbb{B}^n$ to $\mathbb{B}^n$. Then G_f is chaotic according to Devaney iff the graph $\Gamma(f)$ is strongly connected.*

Any function such that the graph $\Gamma(f)$ is strongly connected is then a candidate for being iterated in G_f for pseudo random number generating. Thus, let us show how to compute a map f with a strongly connected graph of iterations $\Gamma(f)$.

We first consider the negation function $\neg$. The iteration graph $\Gamma(\neg)$ is obviously strongly connected: since each configuration $(x_1, \ldots, x_n)$ may reach one of its n neighbors, there is then a bit by bit path from any $(x_1, \ldots, x_n)$ to any $(x'_1, \ldots, x'_n)$. Let then Γ be a graph, initialized with $\Gamma(\neg)$, the algorithm iteratively does the two following stages:

1) select randomly an edge of the current iteration graph Γ and
2) check whether the current iteration graph without that edge remains strongly connected (by a Tarjan algorithm [15], for instance). In the positive case the edge is removed from G,

until a rate r of removed edges is greater than a threshold given by the user.

Formally, if r is close to 0% (*i.e.*, few edges are removed), there should remain about $n \times 2^n$ edges (let us recall that 2^n is the amount of nodes). In the opposite case, if r is close to 100%, there are left about 2^n edges. In all the cases, this step returns the last graph Γ that is strongly connected. It is not then obvious to return the function f whose iteration graph is Γ.

Function f	$f(x)$, for x in $(0, 1, 2, \ldots, 15)$	Rate
$\neg$	(15,14,13,12,11,10,9,8,7,6,5,4,3,2,1,0)	0%
ⓐ	(15,14,13,12,11,10,9,8,7,6,7,4,3,2,1,0)	2.1%
ⓑ	(14,15,13,12,11,10,9,8,7,6,5,4,3,2,1,0)	4.1%
ⓒ	(15,14,13,12,11,10,9,8,7,7,5,12,3,0,1,0)	6.25%
ⓓ	(14,15,13,12,9,10,11,0,7,2,5,4,3,6,1,8)	16.7%
ⓔ	(11,2,13,12,11,14,9,8,7,14,5,4,1,2,1,9)	16.7%
ⓕ	(13,10,15,12,3,14,9,8,6,7,4,5,11,2,1,0)	20.9%
ⓖ	(13,7,13,10,11,10,1,10,7,14,4,4,2,2,1,0)	20.9%
ⓗ	(7,12,14,12,11,4,1,13,4,4,15,6,8,3,15,2)	50%
ⓘ	(12,0,6,4,14,15,7,15,11,1,14,2,7,4,7,9)	75%

Table II: Functions with SCC graph of iterations

However, such an approach suffers from generating many functions with similar behavior due to the similarity of their graph. More formally, let us recall the graph isomorphism definition that resolves this issue. Two directed graphs Γ_1 and Γ_2 are *isomorphic* if there exists a permutation p from the vertices of Γ_1 to the vertices of Γ_2 such that there is an arc from vertex u to vertex v in Γ_1 iff there is an arc from vertex $p(u)$ to vertex $p(v)$ in Γ_2.

Then, let f be a function, $\Gamma(f)$ be its iteration graph, and p be a permutation of vertices of $\Gamma(f)$. Since $p(\Gamma(f))$ and $\Gamma(f)$ are isomorphic, then iterating f (*i.e.*, traversing $\Gamma(f)$) from the initial configuration c amounts to iterating the function whose iteration graph is $p(\Gamma(f))$ from the configuration $p(c)$. Graph isomorphism being an equivalence relation, the sequel only consider the quotient set of functions with this relation over their graph. In other words, two functions are distinct if and only if their iteration graph are not isomorphic.

Table II presents generated functions that have been ordered by the rate of removed edges in their graph of iterations compared to the iteration graph $\Gamma(\neg)$ of the boolean negation function $\neg$.

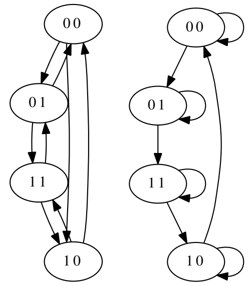
For instance let us consider the function ⓖ from $\mathbb{B}^4$ to $\mathbb{B}^4$ defined by the following images: $[13, 7, 13, 10, 11, 10, 1, 10, 7, 14, 4, 4, 2, 2, 1, 0]$. In other words, the image of 3 (0011) by ⓖ is 10 (1010): it is obtained as the binary value of the fourth element in the second list (namely 10). It is not hard to verify that Γ(ⓓ) is SCC. Next section gives practical evaluations of these functions.

V. Modifying the PRNG Algorithm

A coarse attempt could directly embed each function of table II in the *iterate_G* function defined in Algorithm 3. Let us show the drawbacks of this approach on a more simpler example.

Let us consider for instance n is two, the negation function on $\mathbb{B}^2$, and the function f defined by the list $[1, 3, 0, 2]$ (i.e., $f(0, 0) = (0, 1), f(0, 1) = (1, 1), f(1, 0) = (0, 0)$, and $f(1, 1) = (1, 0)$) whose iterations graphs are represented in Fig. 1. The two graphs are strongly connected and thus the vectorial negation function should theoretically be replaced by the function f.

In the graph of iterations $\Gamma(\neg)$ (Fig. 1a), let us compute the probability $P^t_\neg(X)$ to reach the node X in t iterations from the node 00. Let X_0, X_1, X_2, X_3 be the nodes 00, 01, 10 and 11. For $i \in \llbracket 0, 3 \rrbracket$, $P^1_\neg(X_i)$, are respectively equal to 0.0, 0.5, 0.0, 0.5. In two iterations $P^2_\neg(X_i)$ are 0.5, 0.0, 0.5, 0.0. It is obvious to establish that we have $P^{2t}(X_i) = P^0(X_i)$



(a) Negation (b) $(1, 3, 0, 2)$

Figure 1: Graphs of Iterations

Name	Deviation	Suff. number of it.
ⓐ	8.1%	167
ⓑ	1%	105
ⓒ	18%	58
ⓓ	1%	22
ⓔ	24%	19
ⓕ	1%	14
ⓖ	20%	6
ⓗ	45.3%	7
ⓘ	53.2%	14

Table III: Deviation with Uniform Distribution

and $P^{2t+1}(X_i) = P^1(X_i)$ for any $t \in \mathbb{N}$. Then in k or $k+1$ iterations all these probabilities are equal to 0.25.

Let us apply a similar reasoning for the function f defined by $[1, 3, 0, 2]$. In its iterations graph $\Gamma(f)$ (Fig. 1b), and with X_i defined as above, the probabilities $P_f^1(X_i)$ to reach the node X_i in one iteration from the node 00 are respectively equal to 0.5, 0.5, 0.0, 0.0. Next, probabilities $P_f^2(X)$ are 0.25, 0.5, 0.25, 0.0. Next, $P_f^3(X)$ are 0.125, 0.375, 0.375, 0.125. For each iteration, we compute the average deviation rate R^t with 0.25 as follows.

$$R^t = \frac{\Sigma_{i=0}^3 \mid P_f^t(X_i) - 0.25 \mid}{4}.$$

The higher is this rate, the less the generator may uniformly reach any X_i from 00. For this example, it is necessary to iterate 14 times in order to observe a deviation from 0.25 less than 1%. A similar reasoning has been applied for all the functions listed in Table II. The table III summarizes their deviations with uniform distribution and gives the smallest iterations number the smallest deviation has been obtained.

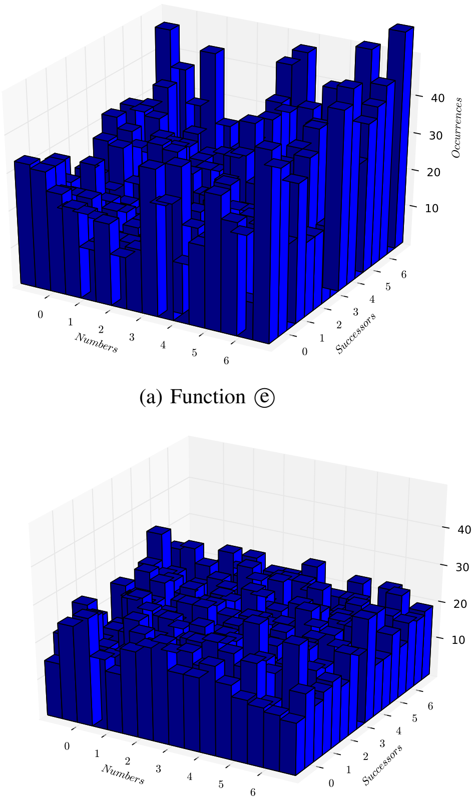
With that material we present in Algorithm 4 the method that allows to take any chaotic function as the core of a pseudo random number generator. Among the parameters, it takes the number b of minimal iterations that have to be executed to get a uniform like distribution. For our experiments b is set with the value given in the third column of Table III.

Compared to the algorithm 2 parameters of this one are the function f to embed and the smallest number of time steps G_f is iterated. First, the number of iterations is either b or $b+1$ depending on the value of the *XORshift* output (if the next value . Next, a loop that iterates G_f is executed.

In this example, n and b are equal to 4 for easy understanding. The initial state of the system x^0 can be seeded by the decimal part of the current time. For example, the current time in seconds since the Epoch is 1237632934.484088, so $t = 484088$. $x^0 = t \mod 16$ in binary digits, then $x^0 = 0100$. m and S can now be computed from *XORshift*.

```
Input: a function $f$, an iteration number $b$, an initial
        state $x^0$ ($n$ bits)
Output: a state $x$ ($n$ bits)
$x \leftarrow x^0$;
$k \leftarrow b + (\textit{XORshift}() \mod 2)$;
for $i = 0, \ldots, k-1$ do
    $s \leftarrow \textit{XORshift}() \mod n$;
    $x \leftarrow F_f(s, x)$;
end
return $x$;
```

Algorithm 4: modified PRNG with various functions



(a) Function ⓔ

(b) Function ⓕ

Figure 2: Repartition of function outputs.

- f = [14,15,13,12,11,10,9,8,7,6,5,4,3,2,1,0]
- k = 4, 5, 4,...
- s = 2, 4, 2, 3, , 4, 1, 1, 4, 2, , 0, 2, 3, 1,...

Chaotic iterations are done with initial state x^0, the mapping function f, and strategy s^1, s^2...The result is presented in Table IV. Let us recall that sequence k gives the states x^t to return: x^4, x^{4+5}, x^{4+5+4}...Successive stages are detailed in Table IV.

To illustrate the deviation, Figures 2a and 2b represent the simulation outputs of 5120 executions with b equal to 40 for ⓔ and ⓕ respectively. In these two figures, the point (x, y, z) can be understood as follows. z is the number of times the value x has been succedded by the value y in the considered generator. These two figures explicitly confirm that outputs of functions ⓕ are more uniform that these of the function ⓔ. In the former each number x reaches about 20 times each number y whereas in the latter, results vary from 10 to more that 50.

k	4					5						4				
s	2	4	2	3		4	1	1	4	2		0	2	3	1	
	$f(4)$	$f(0)$	$f(0)$	$f(4)$		$f(6)$	$f(7)$	$f(15)$	$f(7)$	$f(7)$		$f(2)$	$f(0)$	$f(4)$	$f(6)$	
f	1	1	1	1		1	**1**	**0**	1	1		1	1	1	**1**	
	0	1	**1**	0		0	0	0	0	**0**		1	**1**	0	0	
	1	1	1	**1**		0	0	0	0	0		**0**	1	**1**	0	
	1	**0**	0	1		**1**	0	0	**0**	0		1	0	1	1	
x^0					x^4						x^9					x^{13}
4	0	0	4	6	6	7	15	7	7	7	2	0	4	6	14	14
0					0		$\xrightarrow{1} 1$	$\xrightarrow{1} 0$			0				$\xrightarrow{1} 1$	1
1	$\xrightarrow{2} 0$		$\xrightarrow{2} 1$		1					$\xrightarrow{2} 0$	0		$\xrightarrow{2} 1$			1
0				$\xrightarrow{3} 1$	1						1	$\xrightarrow{3} 0$		$\xrightarrow{3} 1$		1
0		$\xrightarrow{4} 0$			0	$\xrightarrow{4} 1$			$\xrightarrow{4} 0$		0					0

Table IV: Application example

VI. Experiments

A convincing way to prove the quality of the produced sequences is to confront them with the NIST (National Institute of Standards and Technology) Statistical Test Suite SP 800-22 [13]. This is a statistical package consisting of 15 tests that focus on a variety of different types of non-randomness that could occur in a (arbitrarily long) binary sequences produced by a pseudo-random number generators.

For all 15 tests, the significance level α was set to 1%. If a p-value is greater than 0.01, the keystream is accepted as random with a confidence of 99%; otherwise, it is considered as non-random. For each statistical test, a set of p-values is produced from a set of sequences obtained by our generator (i.e., 100 sequences are generated and tested, hence 100 p-values are produced).

Empirical results can be interpreted in various ways. In this paper, we check whether $\mathbb{P}_T$ (P-values of p-values), which arise via the application of a chi-square test, were all higher than 0.0001. This means that all p-values are uniformly distributed over (0, 1) interval as expected for an ideal random number generator.

Table V shows $\mathbb{P}_T$ of the sequences based on discrete chaotic iterations using different "iteration" functions. If there are at least two statistical values in a test, the test is marked with an asterisk and the average value is computed to characterize the statistical values. Here, NaN means a warning that test is not applicable because of an insufficient number of cycles. Time (in seconds) is related to the duration needed by each algorithm to generate a 10^8 bits long sequence. The test has been conducted using the same computer and compiler with the same optimization settings for both algorithms, in order to make the test as fair as possible.

Firstly, the computational time in seconds has increased due to the growth of the sufficient iteration numbers, as precised in Table III. For instance, the fastest generator is ⓖ since each new number generation only requires 6 iterations. Next, concerning the NIST tests results, best situations are given by ⓑ, ⓓ and ⓕ. In the opposite, it can be observed that among the 15 tests, less than 5 ones are a successful for other functions. Thus, we can draw a conclusion that, ⓑ, ⓓ, and ⓕ are qualified to be good PRNGs with chaotic property. NIST tests results are not a surprise: ⓑ, ⓓ, and ⓕ have indeed a deviation less than 1% with the uniform distribution as already precised in Table III. The rate of removed edge in the graph $\Gamma(\neg)$ is then not a pertinent criteria compared to the deviation with the uniform distribution property: the function ⓐ whose graph Γ(ⓐ) is $\Gamma(\neg)$ without the edge $1010 \rightarrow 1000$ (*i.e.*, with only one edge less than $\Gamma(\neg)$) has dramatic results compared to the function ⓕ with many edges less.

Let us then try to give a characterization of convenient function. Thanks to a comparison with the other functions, we notice that ⓑ, ⓓ, and ⓕ are composed of all the elements of $[\![0;15]\!]$. It means that ⓑ, ⓓ, and ⓕ, and even the vectorial boolean negation function are arrangements of $[\![0;2^n]\!]$ ($n = 4$ in this article) into a particular order.

VII. Conclusion

In this work, we first have formalized the PRNG already presented in a previous work. It results a new presentation that has allowed to optimize some part and thus has led to a more efficient algorithm. But more fundamentally, this PRNG closely follows iterations that have been proven to be topological chaotic.

By considering a characterization of functions with topological chaotic behavior (namely those with a strongly connected graph of iterations), we have computed a new class of PRNG based on instances of such functions. These functions have been randomly generated starting from the negation function. Then an a posteriori analysis has checked whether any number may be equiprobabilistically reached from any other one.

The NIST statistical test has confirmed that functions without equiprobabilistical behavior are not good candidates for being iterated in our PRNG. In the opposite, the other ones have topological chaos property and success all the NIST tests. To summarize the approach, all our previous approaches were based on only one function (namely the negation function) whereas we provide now a class of many trustworthy PRNG.

Future work are mainly twofold. We will firstly study sufficient conditions to obtain functions with the two properties of equiprobability and strongly connectivity of its graph of iterations. With such a condition any user should choose its own trustworthy PRNG. Dually, we will continue the evaluation of randomness quality by checking other statistical series like DieHard[11], TestU01 [9]. . . on newly generated

Method	ⓐ	ⓑ	ⓒ	ⓓ	ⓔ	ⓕ	ⓖ	ⓗ	ⓘ
Frequency (Monobit) Test	0.00000	0.45593	0.00000	0.38382	0.00000	0.61630	0.00000	0.00000	0.00000
Frequency Test within a Block	0.00000	0.55442	0.00000	0.03517	0.00000	0.73991	0.00000	0.00000	0.00000
Cumulative Sums (Cusum) Test*	0.00000	0.56521	0.00000	0.19992	0.00000	0.70923	0.00000	0.00000	0.00000
Runs Test	0.00000	0.59554	0.00000	0.14532	0.00000	0.24928	0.00000	0.00000	0.00000
Test for the Longest Run of Ones in a Block	0.20226	0.17186	0.00000	0.38382	0.00000	0.40119	0.00000	0.00000	0.00000
Binary Matrix Rank Test	0.63711	0.69931	0.05194	0.16260	0.79813	0.03292	0.85138	0.12962	0.07571
Discrete Fourier Transform (Spectral) Test	0.00009	0.09657	0.00000	0.93571	0.00000	0.93571	0.00000	0.00000	0.00000
Non-overlapping Template Matching Test*	0.12009	0.52365	0.05426	0.50382	0.02628	0.50326	0.06479	0.00854	0.00927
Overlapping Template Matching Test	0.00000	0.73991	0.00000	0.55442	0.00000	0.45593	0.00000	0.00000	0.00000
Maurer's "Universal Statistical" Test	0.00000	0.71974	0.00000	0.77918	0.00000	0.47498	0.00000	0.00000	0.00000
Approximate Entropy Test	0.00000	0.10252	0.00000	0.28966	0.00000	0.14532	0.00000	0.00000	0.00000
Random Excursions Test*	NaN	0.58707	NaN	0.41184	NaN	0.25174	NaN	NaN	NaN
Random Excursions Variant Test*	NaN	0.32978	NaN	0.57832	NaN	0.31028	NaN	NaN	NaN
Serial Test* (m=10)	0.11840	0.95107	0.01347	0.57271	0.00000	0.82837	0.00000	0.00000	0.00000
Linear Complexity Test	0.91141	0.43727	0.59554	0.43727	0.55442	0.43727	0.59554	0.69931	0.08558
Success	5/15	15/15	4/15	15/15	3/15	15/15	3/15	3/15	3/15
Computational time	66.0507	47.0466	32.6808	21.6940	20.5759	19.2052	16.4945	16.8846	19.0256

Table V: NIST SP 800-22 test results ($\mathbb{P}_T$)

functions.

References

[1] J. M. Bahi and C. Guyeux. Topological chaos and chaotic iterations, application to hash functions. In *WCCI'10, IEEE World Congress on Computational Intelligence*, pages 1–7, Barcelona, Spain, July 2010. Best paper award.

[2] J. M. Bahi, C. Guyeux, and Q. Wang. A novel pseudo-random generator based on discrete chaotic iterations. In *INTERNET'09, 1-st Int. Conf. on Evolving Internet*, pages 71–76, Cannes, France, August 2009.

[3] J. M. Bahi, C. Guyeux, and Q. Wang. Improving random number generators by chaotic iterations. application in data hiding. In *ICCASM 2010, Int. Conf. on Computer Application and System Modeling*, pages V13–643–V13–647, Taiyuan, China, October 2010.

[4] J. M. Bahi, C. Guyeux, and Q. Wang. A pseudo random numbers generator based on chaotic iterations. application to watermarking. In *WISM 2010, Int. Conf. on Web Information Systems and Mining*, volume 6318 of *LNCS*, pages 202–211, Sanya, China, October 2010.

[5] S. Behnia, A. Akhavan, A. Akhshani, and A. Samsudin. A novel dynamic model of pseudo random number generator. *Journal of Computational and Applied Mathematics*, 235(12):3455–3463, 2011.

[6] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT press, 3rd ed. edition, 2009.

[7] R. L. Devaney. *An Introduction to Chaotic Dynamical Systems*. Redwood City: Addison-Wesley, 2nd edition, 1989.

[8] C. Guyeux. *Le désordre des itérations chaotiques et leur utilité en sécurité informatique*. PhD thesis, Université de Franche-Comté, 2010.

[9] P. L'Ecuyer and R. J. Simard. Testu01: A C library for empirical testing of random number generators. *ACM Trans. Math. Softw.*, 33(4), 2007.

[10] N. Liu. Pseudo-randomness and complexity of binary sequences generated by the chaotic system. *Communications in Nonlinear Science and Numerical Simulation*, 16(2):761–768, 2011.

[11] G. Marsaglia. Diehard: a battery of tests of randomness. *1414203*, 1996.

[12] G. Marsaglia. Xorshift rngs. *Journal of Statistical Software*, 8(14):1–6, 2003.

[13] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, M. Vangel, D. Banks, A. Heckert, J. Dray, and S. Vo. *A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*. National Institute of Standards and Technology, April 2010.

[14] Fuyan Sun and Shutang Liu. Cryptographic pseudo-random sequence from the spatial chaotic map. *Chaos, Solitons & Fractals*, 41(5):2216–2219, 2009.

[15] R. Tarjan. Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 1(2):146–160, 1972.

On the Applications of Deterministic Chaos for Encrypting Data on the Cloud

J. M. Blackledge
Information and Communications Security Research Group
Dublin Institute of Technology
url: http://eleceng.dit.ie/icsrg
Email: http://eleceng.dit.ie/blackledge

N. Ptitsyn
Department of Information Processing
and Management Systems
Moscow State Technical University
Email:nptitsyn@gmail.com

***Abstract*—Cloud computing is expected to grow considerably in the future because it has so many advantages with regard to sale and cost, change management, next generation architectures, choice and agility. However, one of the principal concerns for users of the Cloud is lack of control and above all, data security. This paper considers an approach to encrypting information before it is 'placed' on the Cloud where each user has access to their own encryption algorithm, an algorithm that is based on a set of iterated function systems that outputs a chaotic number stream, designed to produce a cryptographically secure cipher. We study cryptographic systems using finite-state approximations to chaos or 'pseudo-chaos' and develop an approach based on the concept of multi-algorithmic cryptography that exploits the properties of pseudo-chaos. Although such algorithms can be taken to be in the public domain in order to conform with the Kerchhoff-Shannon principal, i.e.** *the enemy knows the system*, **their combination can be used to secure data in a way that is unique to each user. This provides the potential for users of the Cloud to upload and transfer data in the knowledge that they are encrypting their data in a way that is algorithm as well key dependent, thereby defeating a known algorithm attack. This paper reports on one application of this approach called** *Crypstic* **in which the encryption engine is mounted on a USB memory stick and where the key is automatically generated by the characteristics of the plaintext/ciphertext file.**

***Keywords*-Cloud computing and Virtualization, Privacy, Security, Ownership and reliability, Data encryption, Deterministic chaos, Multi-algorithmicity**

I. Introduction

Current debates with regard to Cloud Computing assume that little will change for users that depend upon third party hosting for their servers. Further, there appears to be a view that standard security protocols will provide sufficient security in the future. These assumptions ignore the widely held view that the Cloud is insecure. This perception is being constantly reinforced in the mind of the user by the increasingly slow and complicated anti-malware software required and frequent stories in the media about major security breaches - often by hostile governments.

Most businesses rely on some proprietary know-how, process, design or other commercial secret to preserve their competitive position and to try and delay product cycle decay. Business, especially now, is very conscious of the need to avoid fixed and capital costs to reduce their vulnerability to volatility. Cloud Computing, as a capital and fixed cost free approach, is an obvious solution but perceived lack of security for commercially sensitive data is a major barrier to conversion from in-house information and communications technology.

A. The Role of Encryption

Conventional encryption, as a means of securing data, has several drawbacks for commercial users. These include the following: (i) Decision-makers do not understand exactly what encryption is or how to judge the relative strengths of different systems; (ii) Industry certification standards and legal regulations, which are relied upon by both governments and commercial organisations, seek to stratify encryption strength by key length while the underlying algorithms are judged by their resistance to standard attacks. This general approach is common to many industries and is not specific to encryption; (iii) The way in which certification is applied causes, as an unintended consequence, systemic risks to be inherent in any approved system. (iv) Certification is both expensive and slow creating a high barrier to entry for innovative encryption systems and making commercially available systems lag years behind the technologies available to hackers. (v) State regulation with regard to the sale of encryption technology can make the process of commercialising new concepts capital investment intensive [1]; (vi) It is clear that the certification process is valued by governments as a means of understanding, controlling and limiting the strength of encryption to meet their security needs in terms of surveillance. Unfortunately, this approach is fatally flawed as it wrongly assumes that hostile governments do not have equivalent or better capabilities to breach encryption.

B. Data Encryption on the Cloud

How are users going to use The Cloud? For practical purposes, commercial users need to process and store data and communicate with new data and output from stored data. In most cases what is needed is a combination of a Website and Database with secure communications. There is also a need to protect against Malware. This is where encryption faces difficulties as it is impossible to identify Malware if it infiltrates a data stream and is encrypted. Also,

ISBN: 978-1-61208-141-0

in spite of some claims to the contrary, a database cannot be encrypted and then used efficiently. For data to be used it has to be readable. The dilemma therefore is, how can data be securely processed within the cloud. With server hosting, the problem is dealt with by encrypting the communication channel, installing anti-malware, providing physical security and segregating a particular user's servers. Thus, the key is to physically and electronically protect the environment where live processing of data takes place and provide data security using encryption for all communication channels and when data needs to be stored.

The greatest danger from using conventional encryption within The Cloud is that the systemic risks inherent in such encryption methods with only key management to separate secret data contaminate The Cloud as a whole. In other words, a fundamental breach of the encryption engine can bring the whole edifice down. It is this issue that provides the focus for this paper which introduces an approach to encrypting data where all systemic risk can be minimised by replacing the issue of key management with the management of meta-encryption-engines using multiple encryption algorithms based on chaos theory - *multi-algorithmicity*. This is based on a Technology to License called *Crypstic* which is available from Hothouse at Dublin Institute of Technology http://www.dit.ie/hothouse/ and has been developed by the Information and Communications Security Research at the same Institute - http://eleceng.dit.ie/icsrg. The current version is designed specifically for the meta-encryption-engines to be mounted and executed on a USB memory 'key'. However, irrespective of where the engines are mounted, to be credible, their control and processing environment has to be undertaken within a cluster of physically and electronically secure hosting locations.

In the context of using *Crypstic* to secure data on the Cloud, each meta-engine is specific to an individual user and each individual must be properly validated and authorized to have a meta-engine which can be submitted to a user upon request. Each meta-engine device provides a secure entry point to the Cloud so that secure communication to and from the exchange can be achieved without the need for the parties to share their meta-engines. As each meta-engine is seeded for each file or packet differently so that the overall system can act like a 'one-time pad'.

The paper is structured as follows: Section II discusses general issues with regard to the role of encrypting data using chaos before it is uploaded onto the Cloud. Section III discusses the principal issues associated with using chaotic iterators for encrypting which leads to Sections IV and V which focus on the computational issues associated with floating-point approximation and state space partitioning respectively. Section VI is the central kernel of the paper which discusses different chaotic maps including multi-algorithmic maps and introduces some of the principal steps required to design chaotic maps suitable for encrypting data. Section VII presents an implementation of the approach discussed in Section VI and Section VII provides a summary of the work reported in this paper.

II. Cloud Computing and Encryption using Chaos

Cloud computing is set to become a dominating theme in security. The Cloud Security Alliance document *Security Guidance for Critical Areas of Focus in Cloud Computing V2.1* [2] provides an overview of the issues associated with security on the cloud and it is on the basis of this document that we present an approach to encrypting data on the Cloud using chaos.

Cloud computing is inevitable. For example, it avoids the need to acquire infrastructure, it decreases 'time to market' and gives flexibility to update in real time. It is instantly scaleable to meet unexpected increases or decreases in traffic volumes and it saves money by transforming the business model from capital expenditure and depreciation to predictable operating cost. Examples of early adopters to the Cloud include the New York Times who wanted to convert 70 years of articles into PDF format to store it electronically. Using the Cloud it achieved this within 24 hours with no residual unneeded IT infrastructure - a 'one-off' project cost. Start-up companies can use the Cloud to give them full IT capabilities with up-front costs and agility to change requirements and scale up at short notice if successful. The Cloud provides low revenue cost 'Customer Relationship Management' facilities without the need to customize data and process applications. However, there are a number of issues with regard to Cloud Computing which include: trust, loss of privacy, regulatory violation, data replication and erosion of integrity and coherence, application sprawl and dependencies. A general overview of the 'Pros and Cons' associated with Cloud Computing is given in Figure 1.

Of these 'pros and cons', security is a potential major problem for the Cloud. In other words, it is imperative to treat the Cloud as a hostile territory. Consequently user-based security is a likely solution and it is in this context that chaos based cipher generation may provide a solution as discussed in the following section.

A. Chaos Based Cipher Generation

The application of chaos to generating ciphers can create billions of different cryptographically secure encryption engines for users. The commercial solution is to generate a website where users can pay for a unique encryption engine to be produced that, upon a remote payment, can be downloaded and used to encrypt their data before 'storage' on the Cloud. This requires a large database of encryption engines to be created. Once created, a randomly selected sequence of these algorithms can be created on a user-by-user basis. The operational conditions under which this approach can be pursued on a commercial basis depends upon country in

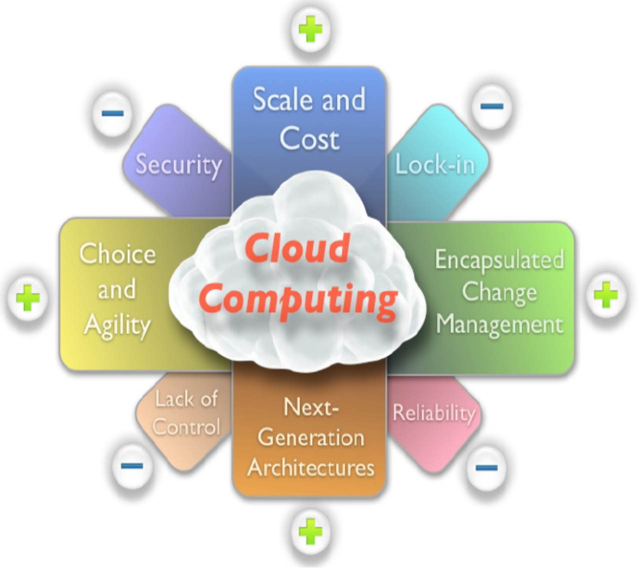


Figure 1. The Pros and Cons associated with Cloud Computing. It should be noted that 'Security' is a 'negative' which relates indirectly to a 'Lack of Control' in terms of possible unauthorised access to data that has been encrypted using standard encryption systems and may therefore be vulnerable to an attack.

which the company is registered. For example, in the UK, commercial operations must conform to the Regulation of Investigatory Powers Act 2000 [1] which inevitably requires an infrastructure to be established involving the employment of staff and is therefore capitalization and overheads intensive.

Chaos can be considered to be a superset of other random number generators used in standard encryption algorithms. There are many disadvantages in using chaos for cryptography but it is nevertheless an interesting application of nonlinear dynamics. The principal value of chaos is the ability to create many different algorithms. This is of course possible with conventional random number generators such as Knuth's M-algorithm [3] but chaos provides greater diversity in terms of the functions available (other than the mod function, for example). However, there are still some major theoretical/computational problems with this approach which include the following:

1) Structurally stable pseudo-chaotic systems: We ideally require a structurally stable cryptosystem, i.e. a system that has (almost) the same cycle length and Lyaponov exponent for all initial conditions. Most of the known pseudo-chaotic systems do not possess this property and there is no rigorous analytical method, as yet, for assessing this property. This is an important problem because without solving it, it is not possible to guarantee that a crypto system based on a deterministic chaotic algorithm or set of algorithms will always produce uncorrelated number streams for any and all keys.

2) Conditions of unpredictability for chaotic systems: What properties of a chaotic system guarantee its computational unpredictability? There is still no theoretically plausible method for evaluating a chaotic system in terms of the necessary/sufficient conditions and properties that will absolutely guarantee the unpredictability of the system to acceptable cryptographic standards. The approach currently being taken is based more on a trial and error approach without the use of an algorithm proving facility. The use of formal methods of software engineering may be of value with regard to this issue.

3) Natively Binary Chaos: While there are, in principle, an unlimited number of chaos based algorithms that can be invented, they currently rely on the use of floating point arithmetic and require high precision FP arithmetic to generate reasonably large cycles (deterministic chaotic algorithm have relatively low cycle lengths which is another disadvantage). These floating point schemes are time consuming given that the number streams they produce are usually converted into bit stream anyway. Designing algorithms that output bit streams directly would therefore be a significant advantage. No theoretical study of this natively binary chaos appears to have been undertaken to date.

4) Asymmetric chaos-based cryptographic: Asymmetric systems are based on trapdoor functions, i.e. functions that have a one-way property unless a secret parameter (trapdoor) is known. One of the best known examples of this is the RSA algorithm that makes use of the properties of prime numbers to design the trapdoor. There is currently no counterpart of a trapdoor transformation, as yet, known in chaos theory.

III. Applications of Chaos for Digital Cryptography

From a theoretical point of view, chaotic systems produce infinite random strings that are asymptotically uncorrelated. However, for applications to digital cryptography, a finite-state systems approach is required which places certain constraints on the design of the algorithm(s). The notion of *pseudo-chaos* introduced in [4], for example, involves a numerical approximation of chaos. The fundamental differences between chaos and pseudo-chaos include the following: (i) the state variable has a finite length (i.e. stores the state with finite precision) and the system has a finite number of states; (ii) the iterated function is evaluated with approximation methods where the result is rounded (or truncated) to a finite precision; (iii) the system may be observed during a finite period of time. The basic problem is that rounding is applied during iteration and the error accumulation causes the original and the approximated processes to diverge. Thus, in general, pseudo-chaos is a poor approximation of chaos because the approximated model does not converge to the original model, and, formally, may exhibit non-chaotic properties including trajectories that eventually become periodic (i.e. contain patterns) and cycles that appear as soon as two states are rounded to the same approximate value. Consequently, the Lyapunov exponent and the Kolmogorov-Sinai information entropy discussed earlier may approach 0. For this reason, it is not possible to directly transform continuous chaotic generators to numerically based generators that require numerical approximations to be made as as summarized in Figure 2. Thus, to use chaos theory for applications in cryptography, a study must be undertaken of pseudo-chaotic systems. This study forms the remit of this paper which is concerned with the question of what are the minimal, typical and maximal periods of the orbits (i.e. string lengths) generated by a pseudo chaotic system?

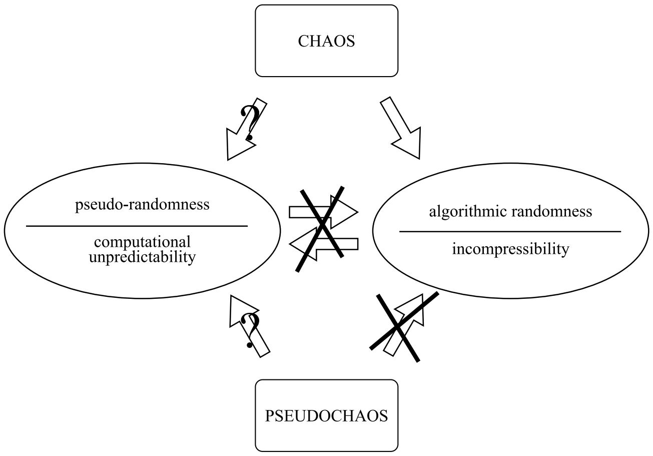


Figure 2. The fundamental properties of chaotic and pseudo-chaotic systems.

Such questions are important in most cryptographic systems. In general, a pseudo-chaotic system produces orbits with different lengths (sometimes called random-length orbits) as illustrated in Figure 3a. Of course, such patterns constitute serious vulnerability as a system may have weak plaintexts and weak keys resulting in recognizable ciphertexts.

If a system has a stable attractor for all initial conditions and parameters, and all orbits have (almost) the same length (Figure 3c), there are more chances to develop a secure encryption scheme. Nevertheless, multiple orbits reduce the search space required for cryptanalysis. An ideal cryptosystem has a single orbit passing through the whole state space (Figure 3b). Another important step in the evaluation of a pseudo-chaotic system is to estimate the Lyapunov exponent of a typical orbit for a time not exceeding its period. However, the analysis of periodic orbits depends critically on the order in which the orbits are considered [5]. Two ordering criteria are considered in the literature, both corresponding to a Lebesgue measure: ordering according to the system size and ordering according to a minimal period or within a period on a lexicographical basis. If the pseudo-chaotic system has a finite precision σ, then the exponential divergence given by

$$e^{n\lambda} = \frac{\left|f^n\left(x_0+\varepsilon\right) - f^n\left(x_0\right)\right|}{\varepsilon}, \qquad n \to \infty, \quad \varepsilon \to 0, \tag{1}$$

will eventually be limited by $\varepsilon = \sigma$. Usually the fraction (1) grows exponentially during the first few iterations and then increases linearly until it finally levels off at a certain finite value.

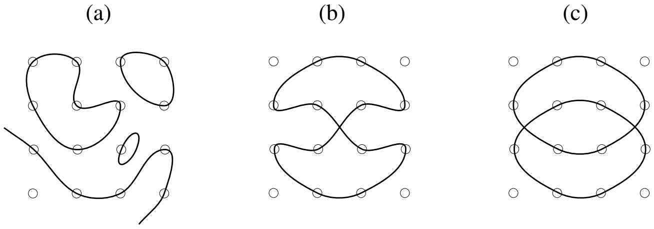


Figure 3. Examples of orbits of a pseudo-chaotic system. (a) Dangerously short orbits (unsuitable for cryptography); (b) A single orbit (the best choice for cryptography); (c) Multiple orbits with the same length (also suitable for encryption).

IV. Floating-point Approximations

Floating-point and fixed point arithmetic are the most straightforward solutions for approximating a continuous system on a finite state machine [6]. Both approaches imply that the state of a continuous system is stored in a program variable with a finite resolution. A state variable x can be written as a binary fraction $b_m b_{m-1} \dots b_1 \,.\, a_1 a_2 \dots a_s$, where a_i, b_j are bits, $b_m b_{m-1} \dots b_1$ denotes the integer part and $a_1 a_2 \dots a_s$ is the fractional part of x. Under a finite resolution, instead of $x_{n+1} = f\left(x\right)$, we write

$$x_{n+1} = round_k\left(f\left(x_n\right)\right),$$

where $k \leq s$ and $round_k\left(x\right)$ is a rounding function defined as

$$round_k\left(x\right) = b_m b_{m-1} \dots b_1 \,.\, a_1 a_2 \dots a_{k-1}\left(a_k + a_{k+1}\right).$$

The iterative rounding is accumulative and results in surprisingly different behavior of pseudo-chaos compared with the continuum counterpart. Figure 4 shows how fast the original and approximated trajectories diverge. For cryptographic applications, the rounding off function exposes another danger. Rounding or truncating the state (e.g. to zero values) can lead to the process dropping out of the chaotic attractor and the system state typically remaining at a certain constant value or infinity. Thus, it is necessary to exclude some forbidden initial conditions and parameters which yield short orbits or patterns of behavior after a small number of iterations. Figure 5 is a plot of the average cycle length verses floating-point precision and shows that high precision does not guarantee a sufficiently long trajectory.

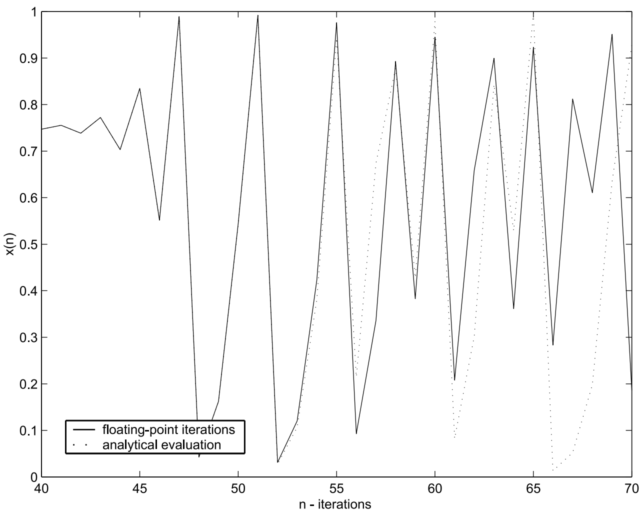


Figure 4. Example trajectories of a continuous-state chaotic system (2) and its 64-bit floating-point approximation. The first curve is obtained by means of the analytical solution (3). The rounding off error is amplified at each iteration and the trajectories diverge exponentially.

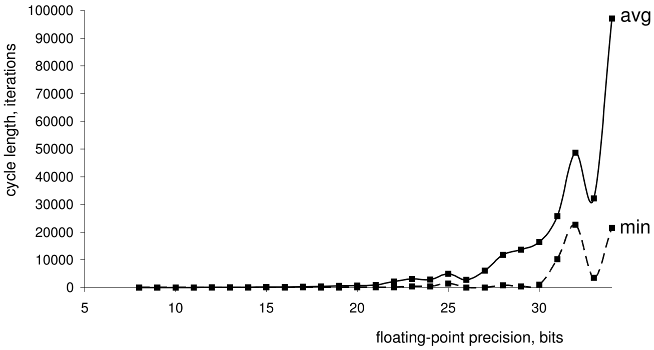


Figure 5. The average (avg) and the minimal (min) cycle length of the logistic system (2) verses floating-point precision obtained from 10 samples of the logistic system.

Another problem associated with the application of pseudo-chaos to encryption is the sensitivity to floating-point processor implementations. Diversified mathematical algorithms or internal precisions in intermediate calculations can lead to a situation where the same encryption application code can generate different cryptographic sequences leading to an incompatibility between software environments. A chaos-based string with two different seeds produces two different sequence with probability 1. This is true for chaotic systems with an infinite state space, where the probability $\Pr\left(f(x_n) = f(x'_n)\right) \to 0$ with $x_n \neq x'_n$ (despite of the fact that f^{-1} is multi-valued). In finite-state approximations, the probability of mapping two points into one is much higher. Furthermore, this can occur at each iteration so that a significant number of trajectories may have identical end routes.

In spite of these shortcomings, a number of investigators have explored the applications of continuous chaos to digital cryptography and in the following sections, an overview of encryption schemes based on a floating-point approximation to chaos is given.

V. Partitioning the State Space

Floating-point cryptographic systems require a mapping from the plaintext alphabet $\{0,1\}^m$ (e.g. 8 bit symbols) to the state space X (e.g. 64 bit floating-point numbers) and, sometimes, from the state space to the ciphertext alphabet. A partition can be defined by a partitioning function $\sigma : X \to \{0,1\}^m$ as with symbolic dynamics. For example, a simple function for two subsets can be designed by taking the last significant bit:

$$\sigma(b_m b_{m-1} \ldots b_1 \,.\, a_1 a_2 \ldots a_s) = a_s.$$

If a floating-point system is a pseudo-random generator, the function σ must be irreversible as with a hard-core predicate. This can be archived with an equiprobable mapping where partitions are selected in such a way that each symbol occurs with the same probability. However, it is *not* obligatory to cover all the state space or assign symbols to all partitions. On the contrary, we can change the statistical properties of the resulting symbolic trajectory by assigning symbols in a particular way. For example, Figure 6 shows a discrete probability distribution of state points in the attractor of the logistic system. By choosing regions with almost the same probability mass, we obtain better statistics in the output, i.e. avoid any statistical bias associated with a cipher. The number of subsets can be increased, for example, up to 4, 8, 16 etc. In this case the generator will produce more pseudo-random bits per iteration ($m = 2, 3, 4$). However, increasing m reduces the cryptographic strength of the generator since it becomes easier to invert σ.

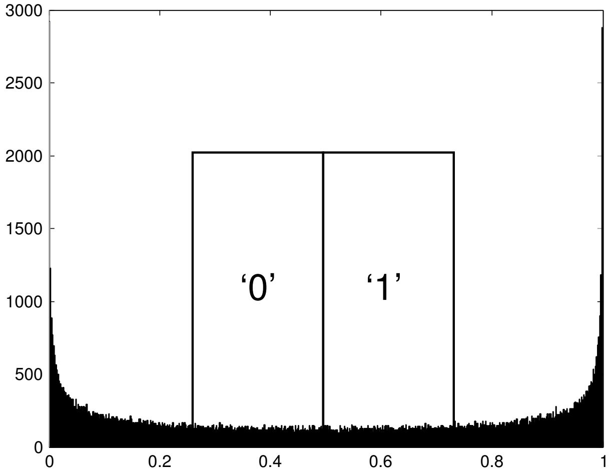


Figure 6. The (discrete) Probability Density Function of a state sequence produced by the logistic system with an incomplete partition.

VI. Example Chaotic Maps

We consider some example chaotic maps which illustrate the principles of using pseudo-chaos for encrypting data.

A. Logistic Map

In 1976, Mitchell Feigenbaum studied the complex behavior of the so-called logistic map given by

$$x_{n+1} = 4rx_n\left(1 - x_n\right), \tag{2}$$

where $x \in (0, 1)$ and $r \in (0, 1)$. For any long sequence of N numbers generated from the seed x_0 we can calculate the Lyapunov exponent given by

$$\lambda\left(x_0\right) = \frac{1}{N}\sum_{n=1}^{N} \log\left|r\left(1 - 2x_n\right)\right|.$$

For example, the numerical estimation for $r = 0.9$ and $N = 4000$ is $\lambda\left(0.5\right) \approx 0.7095$.

With certain values of the parameter r, the generator delivers a sequence, which *appears* pseudo-random. The Freigenbaum diagram (Figure 7) shows the values of x_n on the attractor for each value of the parameter r. As r increases, the number of points in the attractor increases from 1 to 2, 4, 8 and hence to infinity. In this area ($r \rightarrow 1$) it may be considered difficult to estimate the final state of the system (without performing n iterations) given an initial conditions x_0, or vice-versa - to recover x_0 (which can be a key or a plaintext) from x_n. This complexity is regarded as a fundamental advantage in using continuous chaos for cryptography. However, for the boundary value of the control parameter $r = 1$ the analytical solution [7], [8] is:

$$x_n = \sin^2\left(2^n \arcsin\sqrt{x_0}\right). \tag{3}$$

When $n = 1$ we have the initial equation (2). Hence, the state x_n can be computed directly from x_0 without performing n iterations.

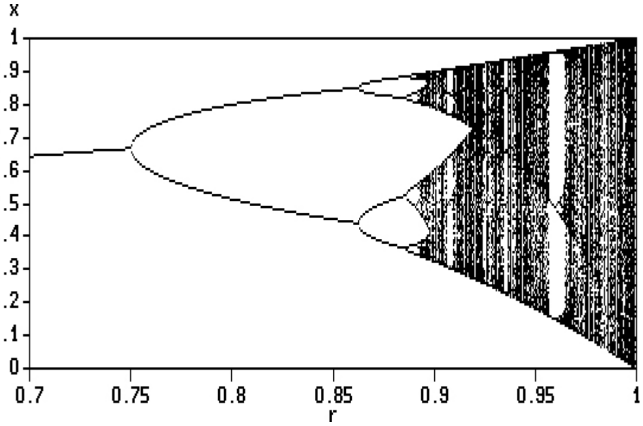


Figure 7. Bifurcation of the logistic map. The most 'unpredictable' behavior occurs when $r \rightarrow 1$.

Bianco *et al.* [9] used the logistic map (2) to generate a sequence of floating point numbers which are then converted into a binary sequence. The binary sequence is XOR-ed with the plaintext, as in a one-time pad cipher where the parameter r together with the initial condition x_0 form a secret key. The conversion from floating point numbers to binary values is done by choosing two disjoint interval ranges representing 0 and 1. The ranges are selected in such a way, that the probabilities of occurrence of 0 and 1 are equal (as illustrated in Figure 6). Note, that an equiprobable mapping does not ensure a uniform distribution. Though the numbers of zeros and ones are equal, the order is not necessarily random.

It has been pointed out by Wheeler [10] and Jackson [11] that computer implementations of chaotic systems yield surprisingly different behavior, i.e. it produces very short cycles and trivial patterns (a numeric example in this paper being given in Figure 5).

B. Matthews Map

Matthews [12] generalizes the logistic map with cryptographic constraints and develops a new map to generate a sequence of pseudo-random numbers based on the iteration

$$x_{n+1} = (r + 1)\left(\frac{1}{r} + 1\right)^r x_n\left(1 - x_n\right)^r, \quad r \in (1, 4).$$

The Matthews system exhibits chaotic behavior for parameter values within an extended range (Figure 8) thereby stretching the key space. However, no robust cryptographic system has been created using this map because of the general floating-point issues discussed previously.

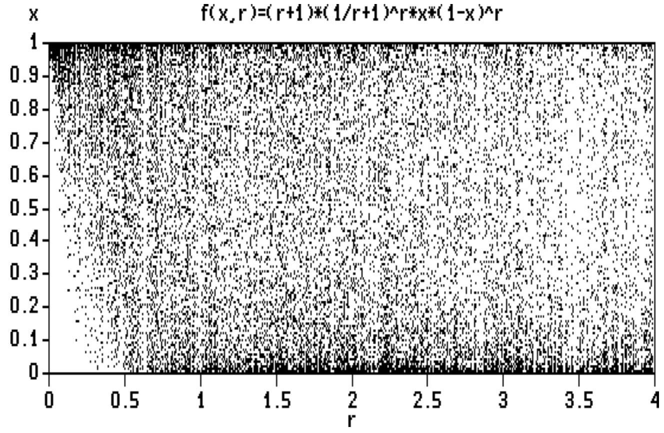


Figure 8. Attractor points corresponding to different values of the parameter r in the Matthews map.

C. Other Examples of Chaotic Maps

Gallagher *et el.* [13] developed a chaotic stream cipher based on the transformation

$$f\left(x\right) = \left(a + \frac{1}{x}\right)^{\frac{x}{a}}, \quad x \in \left(0, 10\right), \quad a \in \left[0.29, 0.40\right].$$

Copyright (c) IARIA, 2011. ISBN: 978-1-61208-141-0

Both the initial condition x_0 and the parameter a represent the key. After $n_0 = 200$ iterations, the system encrypts the plaintext byte p_1 into the ciphertext float $c_1 = f^{n_0+n_1}(x_0)$, i.e. the chaotic map is applied $p_1 \in [0, 255]$ times. Subsequent plaintexts are encrypted using the same trajectory. Clearly, the disadvantages of such an encryption scheme are: (i) the data expansion (the floating-point representation of c_i is considerably larger that the source byte p_i; (ii) unstable cycles incident to floating-point chaos generators.

Kotulski [14] proposes a two dimensional map matching the reflection law of a geometric square and defines conditions under which the system is chaotic and mixing. In addition to a range of specific maps suggested by a wealth of authors, there are, in principle, an unlimited number of iteration functions available or that can be invented to generate cryptographic sequences where the nonlinear transformation can be more or less complex, e.g.

$$rx\left[1 - \tan\left(\frac{1}{2}x\right)\right] \quad \text{or} \quad rx\left[1 - \log\left(1 + x\right)\right]$$

Although each system has a particular state distribution in the phase space, qualitatively, its behavior is similar to a basic chaotic system such a logistic map. To increase unpredictability (i.e. the number of states, nonlinearity, complexity) high-order multi-dimensional chaotic system can be used [15]. However, to date, no known systems have been implemented as a working encryption algorithm. This is principally due to the relatively complex numerical integration schemes that are required and the non-uniform distribution of state variables. However, by considering a number of randomly selected pseudo-chaotic algorithms (all of which meet the appropriate design criteria) that operate on randomly selected plaintext blocks, it is possible to produce a multi-algorithmic approach to data encryption which is the principal concept presented in this paper.

D. Pseudo-Chaos and Conventional Cryptosystems

Existing pseudo-random generators can be viewed as pseudo-chaotic systems. For example, consider the Blum-Blum-Shub system [16] given by the iterated function $x_{n+1} = x_n^2 \bmod M$ where $M = pq$, where p, q are two distinct prime numbers each congruent to 3 modulo 4. The output bit b_n is obtained from a predicate $\sigma(x_n)$, which is the last significant bit of x_n. Besides the sensitivity to the initial condition and the topological transitivity, a pseudo-random generator has to be computationally unpredictable. The last property is ensured by a *one-way* iterated function and a hard-core predicate. A one-way transformation is based on a certain mathematical problem, which is considered unsolved. For example, the Blum-Blum-Shub function works under the assumption that integer factorization is intractable. Chaos theory is not focused on the algorithmic complexity of the iterated function, whereas in cryptography the complexity is the key issue, i.e. security.

E. Symmetric Block Ciphers

All classical iterative block ciphers, at least with regard to our notation, are pseudo-chaotic or combinations of several pseudo-chaotic systems. As an example, consider the Rijndael algorithm which form the basis for the Advanced Encryption Standard [17]. The system state x is a two-dimensional array of bits. The plaintext is assigned to the initial conditions x_0 and, after a fixed number of iterations ($n = 10 \dots 14$), the ciphertext is obtained from the final state x_n. The encryption transformation is a combination of several pseudo-chaotic maps: (i) the substitution phase is a composition of multiplicative inverse and affine transformations; (ii) the mixing phase includes cycle shifts and column multiplication over a finite field; (iii) the round key is obtained from another pseudo-chaotic system.

If we consider the substitution and mixing phases as a single iterated function, the encryption scheme will represent two linked pseudo-chaotic systems (Figure 9).

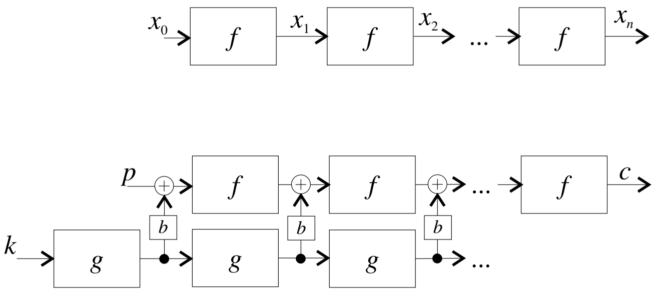


Figure 9. A typical block cipher which is a combination of several pseudo-chaotic systems.

F. Multi-Algorithmic Generators

Protopopescu [18] proposes an encryption scheme based on multiple iterated functions: m different chaotic maps are initialized using a secret key. If the maps depend on parameters, these too are determined by the key. The maps are iterated using floating point arithmetic and m bytes are extracted from their floating point representations, one byte from each map. These m numbers are then combined using an XOR operation. The process is repeated to create a one time pad which is finally XOR-ed with the plaintext. In this paper, we extend the Protopopescu scheme to include a multi-algorithmic approach based on the following properties: (i) Chaotic systems can be connected to each other (i.e. the state of each system influences the states of all other systems) to increase the average orbit length and form a single chaotic system with a large state space and more stable orbits. (ii) The set of chaotic systems (iterated functions) can be different for each encryption session, implemented by supplying an iterated function set with the key. (iii) The output bit can be generated in each q^{th} iteration to increase the independence of bits. (iv) Chaotic systems can be permuted in a complex manner, in particular, the order in which they are utilized or 'turned on' by a key.

We can define this extended cryptographic system as

$$\begin{cases} x^1_{n+1} = f_1(x^2_n, k^1), & b^1_j = \sigma_1(x^1_{qj}) \\ x^2_{n+1} = f_2(x^2_n, k^2), & b^2_j = \sigma_2(x^2_{qj}) \\ \dots & \dots \\ x^m_{n+1} = f_m(x^m_n, k^m), & b^m_j = \sigma_m(x^m_{qj}) \end{cases}$$

$$b_j = b^1_j \oplus b^2_j \oplus \ldots \oplus b^m_j,$$

where $f_1, f_2, \ldots, f_m$ are iterated functions of the session set, $\langle x^1_0, k^1, x^2_0, k^2, \ldots, x^m_0, k^m \rangle$ are initial conditions, $b^1_j, b^2_j, \ldots b^m_j$ are the internal state bits in the $(n = qj)^{\text{th}}$ moment of time, b_j is the generator output and where the mixing component providing property (i) is given by

$$\begin{cases} x^1_n = mix_1(x^1_n, x^2_n, \ldots, x^m_n) \\ x^2_n = mix_2(x^1_n, x^2_n, \ldots, x^m_n) \\ \dots \\ x^m_n = mix_m(x^1_n, x^2_n, \ldots, x^m_n) \end{cases}$$

A demonstration encryption system - *Crypstic* - based on multiple chaotic systems with extended properties (i)-(iv) is available from [19]. The system solves the problems relating to the floating-point arithmetic to provide $(m-1)$ redundant systems. In practice, an encryption engine can be based on any number of algorithms, each algorithm having been 'designed' with respect to the required (maximum entropy) performance conditions through implementation of appropriate conditional parameters T and $\Delta_\pm$ where T is the threshold defining the partition between bits as shown in Figure 6 and $\Delta_\pm$ defines the extent of each partition. The basic steps are as follows:

Step 1: Invent a (non-linear) function f and apply the iteration $x_{n+1} = f(x_n, p_1, p_2, ...)$
Step 2: Normalise the output of the iteration so that $\mathbf{x}_\infty = 1$.
Step 3: Graph the output x_n and adjust parameters $p_1, p_2, ...$ until the output 'looks' chaotic.
Step 4: Graph the histogram of the output and observe if there is a significant region of the histogram over which it is 'flat'.
Step 5: Set the values of the thresholds T and $\Delta_\pm$ based on 'observations' made in Step 4.

Analysing of the iteration using a Feigenbaum diagram can also be undertaken but this can be computationally intensive and each function can be categorised in terms of parameters such as the Lyapunov Dimension and information entropy, for example. It should be noted that many such inventions fail to be of practical value because their statistics may not be suitable (e.g. the histogram may not be flat enough or is flat only over a very limited portion of the histogram), chaoticity may not be guaranteed for all values of the seed x_0 between 0 and 1 and the numerical performance of the algorithm may be poor. The aim is to obtain an iteration that is numerically relatively trivial to compute, provides an output that has a broad statistical distribution and is valid for all floating point values of x_0 between 0 and 1.

The functions used for the demo system available at [19] are given in the following table where the values of T, Δ_+ and Δ_- apply to the normalised output stream generated by each function.

Function $f(x)$	r	T	Δ_+	Δ_-
$rx(1-\tan(x/2))$	3.3725	0.5	0.3	0.3
$rx[1-x(1+x^2)]$	3.17	0.5	0.25	0.35
$rx[1-x\log(1+x)]$	2.816	0.6	0.3	0.2
$r(1- \mid 2x-1 \mid^{1.456})$	0.9999	0.5	0.3	0.3
$\mid \sin(\pi r x^{1.09778}) \mid$	0.9990	0.6	0.25	0.25

The functions given in the table above produce outputs that have a relatively broad and smooth histogram which can be made flat by application of the values of T and $\Delta_\pm$ as illustrated in Figure 6 Some functions, however, produce poor characteristic in this respect. For example, the function

$$f(x) = r \mid 1 - \tan(\sin x) \mid, \quad r = 1.5$$

has a highly irregular histogram which is not suitable in terms of applying values of T and $\Delta_\pm$ and, as such, is not an appropriate IFS for this application.

VII. Systems Implementation - *Crypstic*

Crypstic is a generic USB utility for encrypting single data files and can be used as such before a file is uploaded to the Cloud on a file-by-file basis. In conventional encryption systems, it is typical to provide a Graphical User Interface (GUI) with fields for inputting the plaintext and outputting the ciphertext where the name of the output (including file extension) is supplied by the user. *Crypstic* [19] outputs the ciphertext by overwriting the input file. This allows the file name, including the extension, to be used to 'seed' the encryption engine and thus requires that the name of the file remains unchanged in order to decrypt. The seed is used to initiate the session key. The file name is converted to an ASCII 7-bit decimal integer stream which is then concatenated and the resulting decimal integer used to seed a hash function whose output is of the form (d, d, f, f, f) where d is a decimal integer and f is a 32-bit precision floating point number between 0 and 1.

The executable file is camouflaged as a *.dll* file which is embedded in a folder containing many such *.dll* files. The reason for this is that the structure of a *.dll* file is close to that of a *.exe* file. Nevertheless, this requires that the source code must be written in such a way that all references to its application are void. This includes all references to the nature of the data processing involved including words such as *Encrypt* and *Decrypt* (strings that are replaced by E and D respectively in a GUI), so that the compiled file, although camouflaged as a *.dll* file, is forensically inert. This must include the development of a run time help facility. Clearly, such criteria are at odds with the 'conventional wisdom' associated with the development of applications but the purpose of this approach is to develop a forensically inert executable file that is obfuscated by the environment in which it is placed. This is based on the forensically inert approach to software engineering.

A. Procedure

The approach to loading the application to encrypt/decrypt a file is based on renaming the *.dll* file to an *.exe* file with a given name as well as the correct extension. Simply renaming a *.dll* file in this way can lead to a possible breach of security by a potential attacker using a key logging system. In order to avoid such an attack, *Crypstic* uses an approach in which the name of the *.dll* file can be renamed to a *.exe* file by using a 'deletion dominant' procedure. For example, suppose the application is called *enigma.exe*, then by generating a *.dll* file called *engine_gmax_index.dll*, renaming can be accomplished by deleting (in the order given) *lld.* followed by *dni_x* followed by *_en* followed by *g* and then inserting a . between *ae* and including *e* after *ex*. A further application is required such that upon closing the application, the *.exe* file is renamed back to its original *.dll* form. This includes ensuring that the time and date stamps associated with the file are not updated.

The procedure described above is an attempt to obfuscate the use of passwords which are increasingly open to attack especially with regard to password protected USB memory sticks. Many manufacturers break all the rules when attempting to implement security. Checking the password and unlocking the stick are two separate processes, both initiated from the PC. Thus, from the point of view of the stick, they are both separate processes, but this is a major flaw. The best USB sticks handle all the encryption to and from the flash memory themselves and do not keep a password at all. The fact that the data cannot be decrypted without it makes it safe. Many USB sticks store a password inside the flash-controller and check it against a password sent by the PC before unlocking the flash-memory. This way, the password cannot be found by reading out the flash-chip manually. Other USB sticks do the same but store the password on flash. Some sticks even store the password on flash and let the PC do the validation.

In addition to the procedures associated with password validation, the concept of password protection is becoming increasingly redundant. For example, Elcomsoft Limited recently filed a US patent for a password cracking technique that relies on the parallel processing capabilities of modern graphics processors. The technique increases the speed of password cracking by a factor of 25 using a GeForce 8800 Ultra graphics card from Nvidia. '*Cracking times can be reduced from days or hours to minutes in some instances and there are plans to introduce the technique into password cracking products*' (http://techreport.com/discussions.x/13460).

B. Protocol

Crypstic is a symmetric encryption system that relies on the user working with a USB memory stick and maintaining a protocol that is consistent with the use of a conventional set of keys, typically located on a key ring. The simplest use of *Crystic* is for a single user to be issued with a *Crypstic* which incorporates an encryption engine that is unique (through the utilisation of a unique set of algorithms). The user can then use the *Crypstic* to encrypt/decrypt files and/or folders (after application of a compression algorithm such as *pkzip*, for example) on a PC before closure of a session. In this way, the user maintains a secure environment using a unique encryption engine with a 'key' that includes a covert access route. If any crypstic, by any party, is lost, then a new pair of sticks are issued with new encryption engines unique to both parties. In addition to a two-party user system, crypstics can be issued to groups of users in a way that provides an appropriate access hierarchy as required.

VIII. Conclusion

There is a fundamental relationship between cryptography and chaos. In both cases, the object of study is a dynamic system that performs an iterative nonlinear transformation of information in an apparently unpredictable but deterministic manner. In terms of sensitivity to initial conditions together with the mixing properties of chaotic systems, with appropriate entropy conscious post-processing, it is possible to ensure cryptographic confusion and diffusion. However, there are a number of conceptual differences between chaos theory and cryptography which include the following: (i) Chaos theory is often concerned with the study of dynamical systems defined on an infinite state space whereas cryptography relies on a finite-state machine and all chaos models implemented on a computer are approximations, i. e. digital computers can only generate pseudo-chaos. (ii) Chaos theory typically studies the asymptotic behaviour of a nonlinear system (i.e. the behaviour of the system as the number of iterations approach infinity when the Lyapunov dimension can be quantified), whereas cryptography focuses on the effect of a small number of iterations that are typically determined by the size of the plaintext. (iii) Chaos theory is not necessarily concerned with the algorithmic complexity but in the interpretation of a physical model from which it has been derived; in cryptography, complexity is the key issue and thus, the concepts of cryptographic security and efficiency have no counterparts in chaos theory. (iv)Classical chaotic systems usually have recognizable attractors whereas in cryptography, we attempt to eliminate any structure by post processing the output to produce a maximum entropy cipher. (v) Unlike chaos in general, cryptographic systems use a combination of independent variables to provide an output that is unpredictable to an observer. (vi) Chaos theory is often associated with the mathematical model used to quantify a physically significant problem, whereas in cryptography, the physical model is of no importance. Point (vi) is of particular importance with regard to the design of chaos based encryption engines. Whereas previous publications in this field (e.g. [12], [9], [20] and [21]) have considered variations on a theme of established chaotic systems, in this paper, we have considered the idea that, in principal, an unlimited number of systems can be 'invented' by a designer in order to provide a limitless range of multi-algorithmic encryption engines.

Cloud computing only represents 4% of current IT spend and is expected to more than double by 2012. Software as a Service (SaaS) by itself is projected to nearly double from $9B to $17B (less than 10% of the total market). However, user-security underpins acceptance of cloud architecture. The approach consider in this paper is based on each user having their own encryption engine enabling both protection and control, e.g.

$$\text{PC} + \textit{Crypstic} = \text{Cloud}\ \ \text{Security}$$

The approach to encrypting data discussed in this paper represents a 'paradigm shift' with regard to single algorithm based ciphers that are in the public domain. The importance of this paradigm shift with regard to cryptography in general and, in particular, security on the cloud, may be appreciated in light of the following text taken from Patrick Mahon's secret history of Hut 8 - the naval section at Bletchly Park from 1941-1945 [22]: *The continuity of breaking Enigma ciphers was undoubtedly an essential factor in our success and it does appear to be true to say that if a key has been broken regularly for a long time in the past, it is likely to continue to be broken in the future, provided that no major change in the method of encypherment takes place*.

Acknowledgment

Jonathan Blackledge is supported by the Science Foundation Ireland Stokes Professorship.

References

[1] Office of Public Sector Information, *Regulation of Investigatory Powers Act 2000*, 2000 CHAPTER 23. http://www.opsi.gov.uk/acts/acts2000/ukpga_20000023_en_1

[2] Cloud Security Alliance http://www.cloudsecurityalliance.org/ Security Guidance for Critical Areas of Focus in Cloud Computing V2.1 http://www.cloudsecurityalliance.org/csaguide.pdf

[3] D. Knuth. The Art of Computer Programming: Volume 2, Seminumerical Algorithms, Second Edition Addison-Wesley, 1981.

[4] B. V. Chirikov and F. Vivaldi. An algorithmic view of pseudorandomness. *Physica D*, (129), 1999.

[5] L. Kocarev. Chaos-based cryptography: a brief overview. *Circuits and systems*, 1(3), 6-21, 2001.

[6] S. Hollasch. Ieee standard 754: floating point numbers, 1998. http://research.microsoft.com/~hollasch/cgindex/coding/ieeefloat.html.

[7] S. Katsura and W. Fukuda. Exactly solvable models showing chaotic bahavior. *Physica*, (130A):597–605, 1985.

[8] J. A. González and R. Pino. Chaotic and stochastic functions. *Physica*, 276A:425–440, 2000.

[9] M. E. Bianco and D. Reed. An encryption system based on chaos theory. US Patent No. 5048086, 1991.

[10] D. D. Wheeler. Problems with chaotic cryptosystems. *Cryptologia*, (12):243–250, 1989.

[11] E. A. Jackson. Perspectives in nonlinear dynamics. Cambridge University Press, 1991.

[12] R. Matthews. On the derivation of a chaotic encryption algorithm. *Cryptologia*, (13):29–42, 1989.

[13] J. B. Gallagher and J. Goldstein. Sensitive dependence cryptography, 1996. http://www.navigo.com/sdc/.

[14] Z. Kotulski and J. Szczepański. Discrete chaotic cryptography. new method for secure communication. In *Proc. NEEDS'97*, 1997. http://www.ippt.gov.pl/~zkotulsk/kreta.pdf.

[15] N. Paar. Robust encryption of data by using nonlinear systems, 1999. http://www.physik.tu-muenchen.de/~npaar/encript.html.

[16] T. Ritter. The efficient generation of cryptographic confusion sequences. *Cryptologia*, (15):81–139, 1991. http://www.ciphersbyritter.com/ARTS/CRNG2ART.HTM.

[17] V. Rijmen and J. Daemen. Rijndael algorithm specification, 1999. http://www.esat.kuleuven.ac.be/~rijmen/rijndael/.

[18] V. A. Protopopescu, R. T. Santoro, and J. S. Tolliver. Fast and secure encryption-decryption method. US Patent No. 5479513, 1995.

[19] http://eleceng.dit.ie/arg/downloads/crypstic

[20] M. S. Baptista, *Cryptography with Chaos*, Physics Letters A, **240**(1-2), 50-54, 1998.

[21] E. Alvarez, A. Fernandez, P. Garcia, J. Jimenez and A. Marcano, *New Approach to Chaotic Encryption*, Physics Letters A, **263**(4-6), 373-375, 1999.

[22] O. Hoare, Enigma: Code Breaking and the Second World War. The True Story through Contemporary Documents, introduced and selected by Oliver Hoare. UK Public Record Office, Richmond, Surrey, 2002.

Target Tracking in the Recommender Space

Toward a new recommender system based on Kalman filtering

Samuel Nowakowski, Anne Boyer
LORIA-KIWI
Campus scientifique – BP239
F-54506 Vandœuvre-lès-Nancy Cedex
Samuel.nowakowski@loria.fr
Anne.boyer@loria.fr

Cédric Bernier
Alcatel-Lucent Bell Labs France – LORIA-KIWI
Route de Villarceaux
F-91600 Nozay
Cedric.bernier@loria.fr

Abstract— **We assume that users and their consumptions of television programs are vectors in the multidimensional space of the categories of the resources. Knowing this space, we propose an algorithm based on a Kalman filter to track the user's profile and to foresee the best prediction of their future position in the recommendation space. The approach is tested on data coming from TV consumptions.**

Keywords-recommender system; user profile; group profile; Kalman filter; target tracking

I. Introduction

In Web-based services of dynamic content, recommender systems face the difficulty of identifying new pertinent items and providing pertinent and personalized recommendations for users.

Personalized recommendation has become a mandatory feature of Web sites to improve customer satisfaction and customer retention. Recommendation involves a process of gathering information about site visitors, managing the content assets, analyzing current and past user interactive behavior, and, based on the analysis, delivering the right content to each visitor.

Recommendation methods can be distinguished into two main approaches: content based filtering [9] and collaborative filtering [10]. Collaborative filtering (CF) is one of the most successful and widely used technology to design recommender systems. CF analyzes user ratings to recognize similarities between users on the basis of their past ratings, and then generates new recommendations based on like-minded users' preferences. This approach suffers from several drawbacks, such as cold start, latency, sparsity [11], even if it gives interesting results.

The main idea of this paper is to propose an alternative way for recommender systems. Our work is based on the following assumption: we consider Users and Web resources as a dynamic system described in a state space. This dynamic system can be modeled by techniques coming from control system methods. The obtained state space is defined by state variables that are related to the users. We consider that the states of the users (by states, we understand « what are the resources they want to see in the next step ») are measured by the grades given to one resource by the users.

In this paper, we are going to present the effectiveness of Kalman filtering based approach for recommendation. We will detail the backgrounds of this approach, i.e., state space description and Kalman filter. Then, we expose the applied methodology. Our conclusion will give some guidelines for future works.

II. Principles

Kalman filter is an optimal state estimator of a linear system. It can estimate the state of the system using a priori knowledge of the evolution of the state and the measurements.

A. Target tracking in the cyberspace

Hypothesis: the user is a target which is moving along an a priori unknown trajectory in the multidimensional space of the categories. Figure 1 shows the principle of our approach.

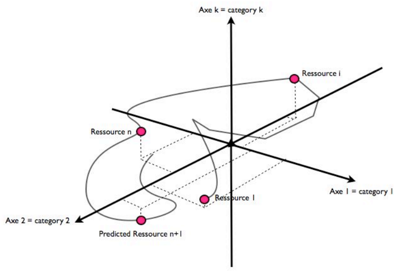


Figure 1. Trajectory in the recommender space

B. Kalman filter: equations

How can we know about a target moving in the recommender space?

Using its position, speed and acceleration, we choose as the state vector the following form:

$$X_k = \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \end{bmatrix}_k \qquad (1)$$

Copyright (c) IARIA, 2011. ISBN: 978-1-61208-141-0

where :

x is the vector containing the position vector

$\dot{x}$ is the vector containing the speed vector,

$\ddot{x}$ is the vector containing the acceleration vector.

The estimation of this state vector will give the necessary knowledge of the trajectory in the recommender space. We use the following state space model:

$$\begin{cases} X_{k+1} = AX_k + w_k \\ Z_k = HX_k + v_k \end{cases} \qquad (2)$$

where matrix A includes the relationship between the position, speed and acceleration where T represents the time period. In our case, we consider T = 1. w_k and v_k are random noises which takes into account unexpected variations in the trajectories.

$$A = \begin{bmatrix} \alpha & T & \frac{1}{2}T^2 \\ 0 & \alpha & T \\ 0 & 0 & \alpha \end{bmatrix} \qquad (3)$$

Matrix H will have the following structure, as shown in the Figure 2.

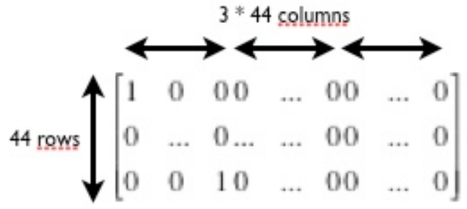


Figure 2. Structure of Matrix H

The Kalman filter equations are then given [6]:

Prediction at time k knowing k+1 ($\hat{X}_{k/k-1}$)

$$\begin{cases} \hat{X}_{k+1/k} = \hat{X}_{k/k-1} + K_k\left(Z_k - H\hat{X}_{k/k-1}\right) \\ \qquad = \left(A - K_kC\right)\hat{X}_{k/k-1} + K_kZ_k \end{cases} \qquad (4)$$

Kalman gain:

$$K_k = AP_{k/k-1}H^T\left(HP_{k/k-1}H^T + R\right)^{-1} \qquad (5)$$

The evolution of the uncertainty on the estimation is then given by:

$$P_{k+1/k} = AP_{k/k-\&}A^T - AP_{k/k-1}H^T\left(HP_{k/k-1}H^T + R\right)^{-1}HP_{k/k-1}A^T \qquad (6)$$

where the initial conditions are given by:

$$\hat{X}_{0/-1} = X_0 \text{ , } P_{0/-1} = P_0$$

and the state prediction by:

$$\hat{X}_{k+1/k}$$

III. Application

A. Description of the experiment

This experiment is based on TV consumption. The dataset is the TV consumption of 6423 English households over a period of 6 months (from 1st September 2008 to 1st March 2009) (Broadcaster Audience Research Board, [7]), [8]. This dataset contains information about the user, the household and about television program. Each TV program is labelled by one or several categories. In the experiment, a user profile build for each person. The user profile is the set of categories associated to the value of interest of the user. This user profile is elaborated in function of the quality of a user's TV consumption: if a TV program is watched entirely, the genre associated to this TV program increases in the user profile. Several logical rules are applied to estimate the interest of a user for a TV program.

The methodology of the experimentation is the following:

- Each user profile is computed at different instants (35) from the TV viewing data.
- The Kalman filter is applied iteratively to estimate the following positions of the user profile in the recommender space.

The entire consumption is described by 44 types which will define the 44 dimensions space where users are "moving".

B. Numerical results

The obtained results can be exposed as follows:

- Kalman filter predicts the interest of a specific user for one gender knowing his past.

Using this prediction, we can propose a new recommendation strategy:

- If the Quantity of Interest (QoI) of the user is predicted to be in one specific region of the space, we can recommend something inside this specific region:
- For example, if the specific region is defined by dimensions Documentary and Drama, we can recommend contents related to these two dimensions
- If the predicted quantity of interest (QoI) changes to another dimension of the space, we can automatically recommend content from this new region of the space.

C. Results

The results can be analyzed as follows: Kalman filter predicts the specific interest for a category of contents of one user.

Figures 3 and 4 show Estimation / Prediction computed by Kalman filtering. Doted-lines show the evolution of the real values. Continuous lines show the obtained predictions. We can see that the prediction curve given by the filter fits very well the real data.

Figure 5 shows the results of the cosine distance which has been computed between the true values and the prediction by the filter. It shows that the prediction will quickly converge with the true values.

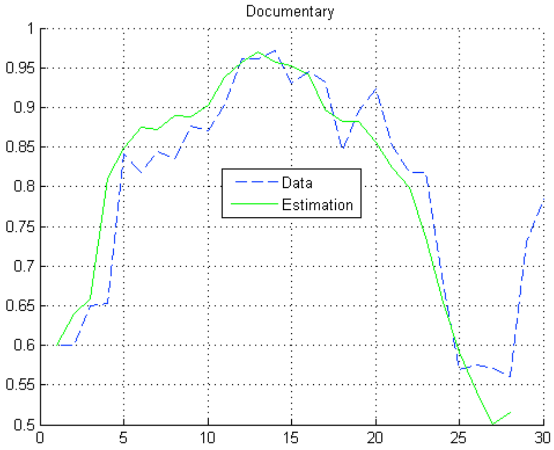


Figure 3. Prediction for Drama

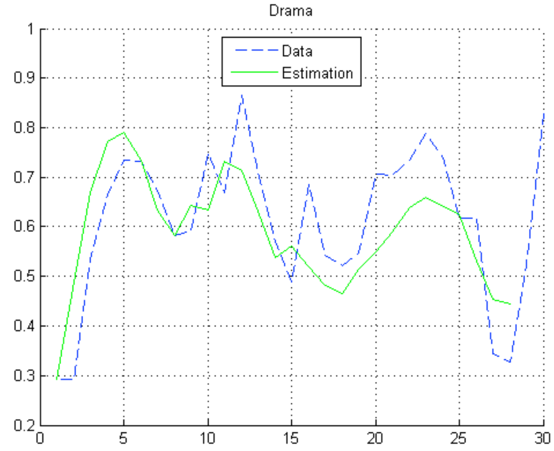


Figure 4. Prediction for Documentary

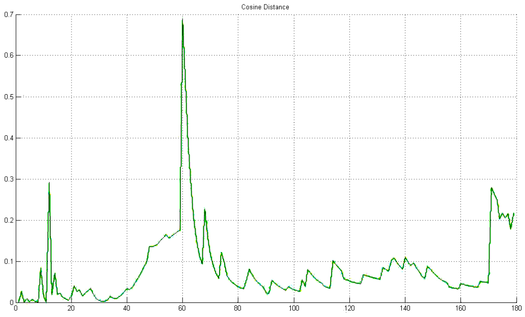


Figure 5. Cosine distance

IV. RECOMMENDATION STRATEGY

In this approach, we can build a recommendation by analyzing the estimation provided by Kalman filter.

The profile is built from the consumption of TV programs. Each TV program is defined by categories such as entertainment, science fiction, talk show, etc. The analysis of the way different TV programs are watched allows us the possibility to estimate the interest of a user for each category. Hence, the user profile is calculated from the TV consumption and it is represented by a vector of valuated concepts.

The user profile is considered as a point in the 44 dimensions-space. This point moves at each different time in the space along a trajectory. With the Kalman filter, we predict the future position of the user profile. The prediction shows the evolution of the trajectory in subspaces restricted to specific dimensions.

For our new recommending strategy (see Figure 6), we observe the difference between the predicted category and the computed one. The rule can be derived as follows:

- If the computed QoI for one category is superior to the estimated QoI (noted negative difference in figure 6), then the user's interest for this category is decreasing.
- If the predicted QoI is superior to the computed one (noted positive difference in Figure 6), then the user's interest for this category is increasing.

Our strategy will could be :

- QoI for specific categories with an important positive difference will influence the recommendation towards these categories
- QoI with an important negative difference discourage the recommendation towards these categories.

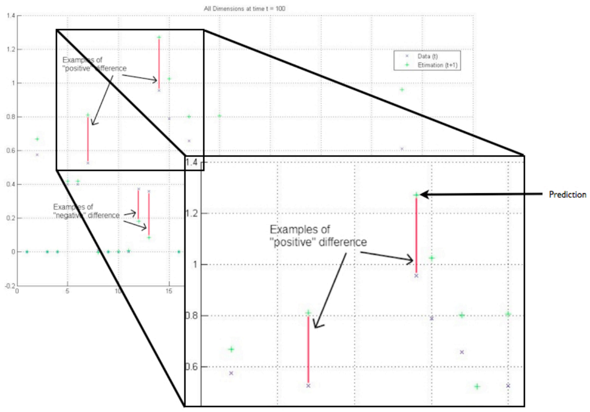


Figure 6. Analysis of the evolution of the prediction for recommendation

Conversely to existing methods which recommend specific contents for a given user, this method takes into account the user's state of mind and will recommend a set of categories of movies inside a subspace of the whole recommender space. Our method performs on the macroscopic level. We find out the type of content the user appreciates and can determine some dimensions that can deliberately be closed out.

The recommendation is based on the two preceding arguments.

- the user's actual state of mind
- the subset of identified dimensions.

From the analysis of these “positive” or “negative” dimensions and from the TV program, we will define the recommendation strategy for a set of TV programs. According to what the user watched during the day, we can refine our recommendation:

- if the user is interested in contents of types x, y and z and if he has already watched content of type x and y, the recommendation would essentially concentrate on content of type z.

The recommender strategy will recommend contents belonging to the categories corresponding to the selected dimensions of the recommender space.

V. Conclusion

In this paper, the main idea is to consider that the one who chooses films as a target which moves along a trajectory in the recommender space. The recommender space is seen as a 44 dimensions space based on the main categories describing the movies. The position of the target is measured by the Quantity of Interest (QoI) for certain categories of movies. Then, the Kalman filter applied using a tracking state space model predicts the “positions” in the recommender space. Knowing the past positions of the user in this space along the different axis of the 44 dimensions space, our Kalman filter based recommender system will suggest:

- if the user is interested in contents of types x, y and z and if he has already watched content of type x and y that day, the recommendation would essentially concentrate on content of type z
- knowing the position in the space, the best prediction for his future positions in the recommender space, i.e., his best index of interest related to the favorite contents.

The strength of our approach is in its capability to make recommendations at a higher level which fit users habits, i.e., given main directions to follow knowing the trajectory in the space and not to suggest specific resources.

Future works will be focused on tracking groups of users and on the definition of the topology of the recommendation space as a space including specific mathematical operators.

References

[1] Anderson, B., and Moore, J. B., 1977. Optimal filtering. Prentice Hall – Information and System Sciences Series

[2] Gibson, W, 1988. Neuromancien. Collection J'ai Lu, La découverte, ISBN 2-7071-1562-2

[3] Söderström, T., 1994. Discrete-time stochastic systems : estimation and control. Prentice Hall International.

[4] Box, G.E.P., and Jenkins, G.M., 1970. Time series analysis : forecasting and control. Holden Day.

[5] Bernier C., Brun A., Aghasaryan A., Bouzid M., Picault J., Senot C., and Boyer A., 2010. Topology of communities for the collaborative recommendations to groups, SIIE 2010 conference, (Sousse, Tunisia, February 17 – 19, 2010).

[6] Gevers, M., and Vandendorpe, L. Processus stochastiques, estimation et prediction. DOI=http://www.tele.ucl.ac.be/EDU/INMA2731/].

[7] BARB: Broadcaster Audience Research Board, DOI=http://www.barb.co.uk/.

[8] Senot, C., Kostadinov D., Bouzid M., Picault J., Aghasaryan A., and Bernier C., 2010. Analysis of strategies for building group profiles. User Modeling, Adaptation and Personalization.

[9] M Pazzani, D. Billsus, 2007. Content-Based Recommendation Systems. In Brusilovsky, P. Kobsa, A. Nejdl, W. (réds) The Adaptive Web : Methods and Strategies of Web Personalization, pp. 325–341.

[10] D. Goldberg, D. Nichols, B. Oki, and D. Terry, 1992. Using Collaborative Filtering to Weave an Information Tapestry. Communications of the ACM, 35(12), pp. 61–70

[11] M. Grcar, D. Mladenic, B. Fortuna, and M. Groblenik, 2006. Data Sparsity Issues in the Collaborative Filtering Framework. Advances in Web Mining and Web Usage Analysis, pp. 58–76

[12] Smith, J., 1998. The book, The publishing company. London, 2nd edition

A Cooperative Game for Distributed Wavelength Assignment in WDM Networks

Horacio Caniza Vierci*, Alexander Wich Vollrath*, Benjamín Barán* †
* *Universidad Católica Nuestra Señora de la Asunción, Asunción, Paraguay*
horacio@tejupy.info, aw@tejupy.info
†*Universidad Nacional de Asunción, San Lorenzo, Paraguay*
bbaran@pol.una.py

***Abstract*—The use of game theory in networking problems is becoming more popular given its potential to model real commercial situations where different agents try to optimize their profit. In this context, this paper proposes a novel cooperative game theory based distributed wavelength assignment method for WDM (Wavelength Division Multiplexing) networks. Experimental results show a clear improvement of the proposed method over a well recognized state of the art distributed wavelength assignment algorithm known as DIR (Destination Initiated Reservation). Thus, this new approach inspired in game theory, provides a first baseline for future work considering cooperation among competing long haul providers that may benefit from collaboration.**

***Keywords*-Game Theory; Nash Bargaining Problem; distributed RWA.**

I. Introduction

At the very core of modern telecommunications resides a very complex: the growing user base requires a solution to provide everyone with enough transmission capacity. Every solution provided to accommodate this ever growing user base, must satisfy contrasting (and in some cases contradictory) objectives: simply supplying bandwidth is not a desirable solution, one must find a way to use the bandwidth in an *efficient* way.

Emerging optical systems are deployed using WDM (Wavelength Division Multiplexing) [1]. In WDM systems, connection requests are satisfied by establishing all-optical channels between source and destination. Given a set of connection requests between two nodes and the paths connecting them, the RWA (Routing and Wavelength Assignment) problem models, as an ILP (Integer Linear Programming) [1], the assignment of every connection request between the two nodes to a free channel in a path joining them. Every resulting pair is known as a lightpath. The *wavelength continuity constraint* imposes a restriction on the lightpaths: the lightpath *must* be established using the same wavelength along the entire path [2]. Wavelength converters could be placed at the nodes to weaken this restriction; however, this is a very expensive alternative [1].

The prohibitively high computational cost of an ILP ($\mathcal{NP}$ (Non-Deterministic Polynomial)-complete) [3] discourages its use in large networks, as well as in networks with bursty traffic patterns. This non-trivial problem drives, in some sense, the research in distributed RWA schemes.

These distributed RWA schemes are, usually, based on message passing, allowing the nodes to establish the needed wavelengths by themselves, thus rendering the existence of a central node unnecessary. There are several proposed distributed RWA schemes: DIR (Destination Initiated Reservation) [4], SIR (Source Initiated Reservation) [4], among other's. The following sections present a very brief description of these schemes.

In the DIR method, the source node sends a *reservation request* message that will travel to the destination node; this message gathers information on the availability of wavelengths along the way. Once this message arrives at the destination node, an available wavelength will be chosen, and a *reservation message* will be sent. This reservation message traverses the reverse path of the *reservation request*, reserving the selected wavelength [4]. One inherent problem of DIR is that, due to the fact that information gathering and wavelength reservation are decoupled, outdated information could result in trying to reserve a wavelength that is no longer available, thus resulting in a blocked connection request.

The differences between DIR and SIR are subtle but important. The SIR method follows a somewhat more aggressive approach. There is no *information gathering* stage *per se*. A *reservation message* is sent to the destination node reserving the available wavelengths on the way. Once it reaches the destination, one of the previously reserved wavelengths is selected. This selection is announced to the source of the connection request in a message, which traverses the reverse path of the *reservation message*, announcing the selection and releasing the unused reserved wavelengths. The number of wavelengths reserved by the *reservation message* varies depending on whether a greedy or moderate approach is used. In the greedy case, every single available wavelength is going to be reserved. In the case a more moderate approach is chosen, a single wavelength is going to be reserved.

The greedy approach improves the chances for the requested connection to be established, while imposing a higher blocking risk for competing connection requests. A more moderate approach, where a single wavelength is reserved, reduces the effect on competing connection requests.

Game Theory constitutes one of the first attempts at formalizing Economic Science. Presented as an integral work for the first time by Von Neumann and Morgenstern [5], it provides a formal framework for describing the behavior of rational and intelligent individuals. Cooperative Game Theory studies voluntary coalitions and negotiations within the Game Theoretical Framework [6].

Considering Metcalfe's law [7], one can envision a non-distant future, where coalitions among competing telecommunication companies are the norm. Pushing the idea even further, one could envision occasional negotiations, resulting in ad-hoc agreements between different companies to relay each others traffic. This convergence of once-competitors is by no means a simple task. Cooperative Game Theory provides a rich set of tools that could be used to prescribe the behavior in this ideal environment.

This paper presents a novel distributed wavelength assignment scheme based in the Cooperative aspects of Game Theory, particularly the $\mathcal{NBP}$ (Nash Bargaining Problem) [8].

This work is organized as follows: a brief introduction to the elements of Game Theory is presented in Section II, Section III maps the Distributed RWA problem to a cooperative game, Section IV presents an illustrative example and in Section V the experimental results are shown. The conlusions and further reaserch section can be found at the end.

II. Game Theory

Game Theory can be defined as the study of mathematical models of conflict and cooperation among intelligent rational decision-makers [9].

Despite the fact that the $\mathcal{NBP}$ belongs to a cooperative game theory approach, it is built upon a non-cooperative game. Therefore a distinction between both approaches is relevant. Non-cooperative game theory explores situations where players do not take into account the possibility to coordinate with each other, thus making communication between players of no benefit at all. On the other hand, cooperative Game Theory analyzes situations where players could benefit from communicating and establishing coalitions among themselves [9].

We now briefly introduce essential concepts required by the model, as presented by Myerson in [9]. A game is defined as a 3-tuple

$\Gamma = \left\langle N, (C_i)_{i \in N}, (f_i)_{i \in N} \right\rangle$ where:

- N is the set of players,
- C_i represents the set of strategies with $i \in N$,
- f_i denotes the utility function with $i \in N$.

In a game (denoted by Γ), each player $i \in N$ has a utility function (denoted by f_i) that represents their own preferences, and a set of strategies (denoted by C_i) from which to choose.

A general behavioral archetype is assumed by all models presented by Von Neumann and Morgestern: every player in a game is going to act in a way as to maximize his utility function [5]. In this context, a strategy is a complete plan of action considering every possible situation that might arise during the course of a game [5].

A. Nash Bargaining Problem

In his work, Nash presents the bargaining solution for a situation in which all players are: *i.* rational, *ii.* intelligent, *iii.* free to choose among the various possible agreements, *iv.* are not going to repudiate any choice made, and, *v.* are perfectly informed, *i.e.* every player knows everything about the game in question [8].

Nash defined the bargaining procedure for a two-player interaction explicitly. He described the negotiation as a two step game: the $\mathcal{TG}$ (Threat Game) and the $\mathcal{DG}$ (Demand Game). The first is a non-cooperative game, while the second depends on the first games and is played cooperatively as described next.

The Threat Game ($\mathcal{TG}$):

Each player values all jointly achievable plans of actions, while expecting a non-cooperative behavior of each other. From a players perspective, a threat is a strategy he is forced to choose in case the negotiation is not favorable [8].

Among various possible solution concepts for a non-cooperative game the $\mathcal{NE}$ (Nash Equilibrium) is perhaps the most widely used [6]. Nash's Equilibrium captures the stable state of a situation, considering the actions that the players take when they act rationally [10].

Formally, according to Osborne and Rubinstein [6]:

$$f_i\left(c_i^*, c_{-i}^*\right) \geq f_i\left(c_i, c_{-i}^*\right) \ \forall c_i \in C_i \qquad (1)$$

where: c_i^* denotes the equilibrium strategy for player i, c_{-i}^* denotes the equilibrium strategies for all other players in the game and c_i denotes a unilateral deviation by player i. These actions are the best, in the sense that there is no possible unilateral deviation by any of the players involved [9].

In a two player context, the resulting equilibrium strategies $n_1 = f_1\left(c_1^*, c_2^*\right)$ and $n_2 = f_2\left(c_1^*, c_2^*\right)$ obtained with equation (1) determine the threats for player 1 and 2 respectively (threat point (n_1, n_2)).

The Demand Game ($\mathcal{DG}$): Given the threat point (n_1, n_2), it is possible to form the set of utilities for the jointly achievable set of strategies for the players in case they cooperate (set B) [8]. Now among all cases where both players could benefit mutually (reflected by set B), each player demands a strategy denoted by d_i $i \in \{1, 2\}$ with utility denoted by $b_1 = f_1\left(d_1, d_2\right) \in B$ for player 1, with the corresponding definition for player 2.

The rationality assumption forces each player to make a demand resulting in the highest possible payoff. Formally, both players choose their demands according to [8]:

$$argmax_{(n_1;n_2)\leq(b_1;b_2)}\left(b_1-n_1\right)\left(b_2-n_2\right) \tag{2}$$

The solution obtained by solving (1), and then (2), is known as the $\mathcal{NBS}$ (Nash Bargaining Solution). It has the following interesting properties: *i.* it is unique, *ii.* it is Pareto efficient, *iii.* it is based on Von Neumann and Morgenstern utilities, *iv.* it is symmetric, in the sense that it does not matter which of the players is known as 1 or 2, and, *v.* is independent of irrelevant alternatives, that is, it is not affected by alternatives that would not have been chosen [9], [8]. For a more detailed exposition, the reader may refer to [11], [8].

III. RWA AS A COOPERATIVE GAME

The $\mathcal{NBP}$ allows different network operators to cooperate with each other without neglecting their own interest. In order to produce a "game" the description below is going to establish analogies between the various elements of Game Theory and those pertaining the RWA problem. After these analogies are described, the $\mathcal{NBS}$ is going to be determined.

We start the construction of the model by defining the following elements: *i.* The set N of players, *ii.* the set C of strategies and, *iii.* the Von Neumann and Morgenstern utility function f.

In this paper, the set of players is mapped to the set of optical nodes in the network, considering that each node may be operated by a different company interested in its own benefit. Of course, when cooperation is good for a company it will be willing to cooperate. This is clearly a case suited for the $\mathcal{NBP}$.

Definition 1 – *The set N of players*:
Let V be the set of vertices and E the set of links in an optical network represented by $G=(V,E)$. An enumeration of the set V is produced: every $v\in V$ is assigned an index i such that $1\leq i\leq|V|$, where i represents a player in the game. ∎

The bargaining process proposed by Nash in [8] is based in a barter scheme *i.e.*, in an exchange of goods between the players. For the purposes of this work, one can derive the set of objects by closely observing the way an optical WDM network operates. In the proposed model, the players are two adjacent nodes in the network which have cross-flow traffic to be serviced.

Definition 2 – *The set C of strategies.*:
Let $v_k,v_{k'}\in V$ represent two nodes of an optical network $G=(V,E)$. Let $e=\langle v_k,v_{k'}\rangle$ be a link between adjacent nodes, and let $\lambda\in\Lambda_e$ be a free wavelength on link e. ∎

Given a connection request t_o that involves v_k and $v_{k'}$, the available lightwaves $\lambda\in\Lambda_e$ shared by link e constitute the strategies for player v_k and $v_{k'}$. This establishes a correspondence between λ and a barter object. Each player will accept as his own the objects received, and (possibly) trade with them in a later instance. Thus, each player must decide if he is interested in *taking* an object in exchange. The chosen utility function must represent, in a reasonable manner, the interests of the players in the game. Since the the problem at hand requires the minimization of the used wavelengths (*min*-RWA), the behavior of the players should reflect this objective. In order to achieve this behavior from the players, a non-negative cost is assigned to every available wavelength in the network. This non-negative cost, motivates each user to use the set of wavelengths representing the lowest possible cost, thus maximizing his utility.

Definition 3 – *Cost function*:
Given a set Λ_{e_j} of wavelengths for a link $e_j\in E$ in the network represented by G, the cost function is defined as:

$$\eta_{e_j}:\Lambda_{e_j}\mapsto\mathbb{N} \tag{3}$$

This function defines a mapping between every wavelength and its position in the radioelectric spectrum for the L window (1565 nm to 1625 nm) defined in ITU-T G.696.1. ∎

The link load, represents the common cost incurred by the players in sharing a particular link. This is similar to the concept of tolls in public roads: the cost is shared between the drivers. The link load is defined as the number of paths that, given a set of pending connection requests, share a particular link. Formally:

Definition 4 – *Link load*:
given a set of paths P in the network represented by G, the load ξ_{e_j} of a link $e_j\in E$ is defined as the number of paths $p_m\in P$ using link e_j

$$\xi_{e_j}=\left|\{p_m\in P\mid e_j\in p_m\}\right| \tag{4}$$

∎

The concept of neighborhood, which is defined as the set of links in a path joining the origin and destination of the connection request, represents the concept of *bounded rationality* [9].

Definition 5 – *Neighborhood*:
Let $p_m\in P$ be a path joining nodes $v_k,v_{k'}\in V$. The neighborhood for node v_k is defined as a subset $\omega_{p_m}\subseteq p_m$. ∎

The utility presented below is adapted for our work from the one presented by Bilò, Moscardelli and Flammini in [12]:

Definition 6 – *Utility function*:
The utility function is defined as $f_{t_0}:T_{v_k}\mapsto\mathbb{N}$, where:

$$f_{t_o}=\sum_{e_j\in\omega_{p_m}}\frac{\eta_{e_j}\left(\lambda\right)}{\xi_{e_j}} \tag{5}$$

where: t_o represents a pending connection request, e_j represents a link in the network G, ω_{p_m} represents the neighborhood, $\eta_{e_j}\left(\lambda\right)$ represents the cost function for wavelength λ and ξ_{e_j} denotes the load of link e_j ∎

The correspondence between the $\mathcal{NBP}$ and an optical network is now complete. However, merely defining the game between a pair of nodes is not enough. In order to complete the proposed model, it is necessary to define which of all possible nodes in G will have the opportunity to barter. For the purposes of this work, the matching of two players that conforms a game is based on Dijkstra's shortest path algorithm [13].

Definition 7 – *Matching technology*:
Let the nodes $v_1, v_2, v_3, \ldots, v_q \in V$ be in graph G, and given a connection request $t_0 = \langle v_1, v_q \rangle$ with v_1 and v_q as origin and destination nodes respectively. The shortest path $p_m = < v_1, v_2, v_3, \ldots, v_q > \in P$ joining the nodes v_1 and v_q produces a "game" confronting v_1 and v_2. After this game is played, v_2 is paired with v_3, and so on. Thus, given a connection request, a sequence of barter games is created. $\blacksquare$

Once a player is matched, the bargaining process follows the one defined by Nash in [8]. Once the solution for the $\mathcal{TG}$ is calculated, the $\mathcal{NBS}$ gives the solution for the $\mathcal{DG}$, and based upon it, the bartered wavelength are assigned.

In summary, when a node belonging to a company (a player of the game) has traffic to be serviced, it will look for a path (using Dijkstra's shortest path algorithm). Once the player found a path (and therefore, its neighbor), he will negotiate the needed wavelengths.

IV. Illustrative example

The following example was extracted from a particular simulation run, which was chosen with the specific intent to reveal some details of the proposed model. In particular, the initial instance corresponds to a simulation of the network represented in Figure 1, with 120 Erlangs of uniformly-distributed traffic, under the assumption that the traffic does not end during the entire simulation run. In order to simplify the following example, the capacity of the fiber links was increased until 0% blocking probability was reached.

According to the proposed matching technology, two adjacent nodes in the network can interact if at least one of them has cross-flow traffic to be serviced, as illustrated in Figure 2.

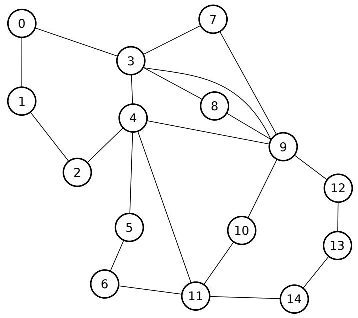


Figure 1. Pacific Internet.

Since nodes 5 and 6 share a link (Figure 1), one needs to analyze the set of pending lightpaths requests for both nodes, represented by multisets T_5 and T_6 respectively, and the set of paths in the network. Multisets are required, since multiple connection requests to the same destination may coexist in the same node; however, the establishment of one such connections, is of no relevance to the others.

$$T_5 = \{6, 9, 13, 4, 9\} \tag{6}$$

$$T_6 = \{3, 0, 3, 1, 0, 3, 4, 4, 8, 13, 8\} \tag{7}$$

Each request is represented by the intended destination node of the pending connection request (see Figure 2).

The set of paths (calculated using Dijkstra's algorithm as shown in [13]) for each node (P_5 and P_6 for nodes 5 and 6, respectively) is a set of ordered n-tuples representing the nodes a lightpath will have to traverse on its way to its destination. The paths needed by nodes 5 and 6 to satisfy their respective outstanding connection requests (T_5, T_6) are shown in (8) and (9). The only paths required are those that connect the source node with the destination of every pending lightpath request.

$$P_5 = \{\langle 5, 6\rangle, \langle 5, 4, 9\rangle, \langle 5, 6, 11, 14, 13\rangle, \langle 5, 4\rangle\} \tag{8}$$

$$\begin{aligned} P_6 = \; & \{\langle 6, 5, 4, 3\rangle, \langle 6, 5, 4, 3, 0\rangle, \langle 6, 5, 4, 2, 1\rangle, \\ & \langle 6, 5, 4\rangle, \langle 6, 5, 4, 3, 8\rangle, \langle 6, 11, 14, 13\rangle\} \end{aligned} \tag{9}$$

The set of barter objects in the game is defined considering sets T_5, T_6, P_5, P_6. In this case, for example, node 6 has three pending connection requests to node 3 (shown in T_6), each requiring one individual lightwave. According to the paths in set P_6, these pending requests need to go through node 5. Thus, node 6 requires three lightwaves passing through node 5 in order to fulfill all three requests to node 3. Node 5 takes advantage of this need, and uses the required lightwaves as object of barter. The full set of barter objects for node 5 and 6 are shown in (10) and (11).

$$Objects_5 = \{\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6, \lambda_7, \lambda_8\} \tag{10}$$

$$Objects_6 = \{\lambda_1, \lambda_2\} \tag{11}$$

The strategies are based in these barter objects, as defined in the previous section.

The size of the neighborhood was defined as 1 for this example, which means that for set P_5, the neighborhood includes only the first link of every path used by node 5. For example, for path $p_1 = \langle 5, 6\rangle \in P_5$, $\omega_{p_1} = \{\langle 5, 6\rangle\}$, for path $p_2 = \langle 5, 4, 9\rangle \in P_5$, $\omega_{p_2} = \{\langle 5, 4\rangle\}$. The other neighborhoods, as well as those for node 6, are obtained following a similar reasoning.

Since the only links that are going to be included when calculating the utility function are those in the neighborhood of each path the connection traverses, only the load of the links in the neighborhood are going to be considered. In this particular example (according to set T_5), link $\langle 5,6\rangle$ has to carry the traffic destined to nodes 6 and 13 (see Figure 2). This would result in a link load of 2 (*i.e.* $\xi_{\langle 5,6\rangle} = 2$).

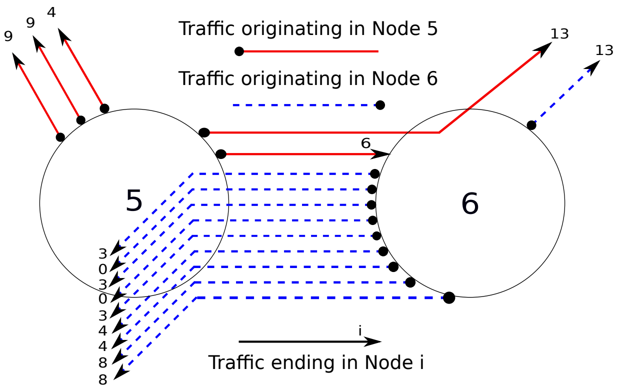


Figure 2. Traffic example.

There is one item left in the model, in order to have the full data needed to compute the utility function: costs have to be assigned to every object of barter for every player. As an example, the costs for both lightwaves needed by node 5 (*i.e.* node's 6 objects of barter) are: $\eta_{\langle 5,6\rangle}(\lambda_1) = 1, \eta_{\langle 5,6\rangle}(\lambda_2) = 2$, according to the definition of η previously introduced in (3).

The utility function for a connection request of node 5, $t_1 = 6 \in T_5$, is:

$$f_{t_1} = \sum_{\langle 5,6\rangle} \frac{\eta_{\langle 5,6\rangle}}{\xi_{\langle 5,6\rangle}} = \frac{\eta_{\langle 5,6\rangle}}{\xi_{\langle 5,6\rangle}} = \frac{1}{2}$$

The corresponding utilities for the pending connection requests for node 6 are calculated in a similar way. It is important to notice, that a player may choose to barter one or more items simultaneously; therefore, if player 5 exchanges the lightwaves needed for two connection requests (*i.e.* to nodes 6 and 13) in one encounter, the resulting utility is the sum of the individual utilities.

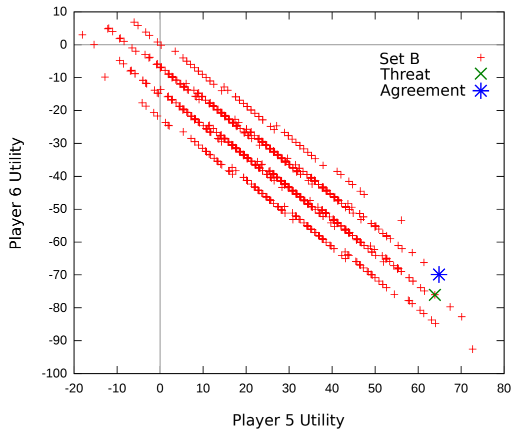


Figure 3. Agreement.

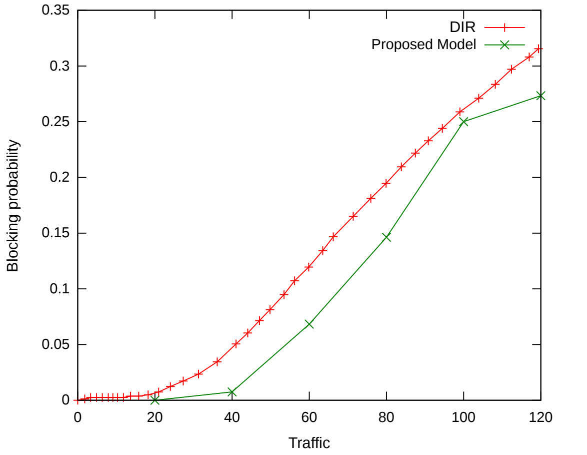


Figure 4. Comparison between DIR and the proposed model - PACnet.

Figure 3 shows the feasible set B, obtained by calculating all pairs of utilities for every pending connection request of both players in the Game. The green and blue points represent the threat point and the achieved agreement, respectively.

V. Experimental results

Given that we could not find in the literature, any work presenting a scenario where networks from different operators coexist, the comparisons were made using a single network. This does not introduce any bias in the possible extension of the presented model. The following simulation results where obtained by simulating lightpath establishment using the PACnet (Pacific Internet) network. This network is shown in Figure 1 and was extracted from [14].

The following simulation results where obtained by simulating lightpath establishment using the PACnet network. Connection request pairs (origin and destination nodes) where chosen using a uniform probability distribution.

The problem solved corresponds to the well-known static-RWA problem, where all requests are known in advance and they are assumed to exist for the whole duration of a particular simulation. Traffic sets range from 0 to 120 Erlangs in 20 Erlangs step increment. For every step, 10 uniformly distributed traffic sets where generated, *i.e* 10 sets of 20 Erlangs, 10 sets of 40, and so on. Figure 4 shows the average of the blocking probability obtained by simulating the network with each of the 10 sets of connection requests for every traffic increment in the network.

Comparison with DIR Figure 4 presents a comparison between the proposed method and state of the art algorithm DIR, as presented by Lu, Xiao and Chlamtac in [4] and According to the results presented in [15], DIR outperforms SIR, making it necessary only to compare the performance of the proposed model with DIR.

The parameters used for the comparison presented in Figure 4 are the same as those used by Lu, Xiao and Chlamtac in [4], *i.e.*: *i.* each link is composed of two opposed

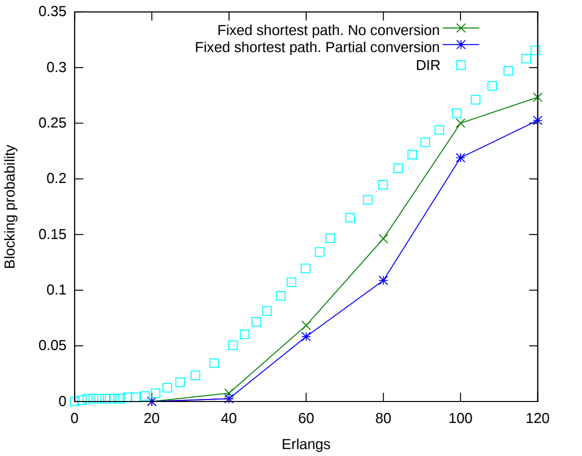


Figure 5. Comparison between DIR and the proposed model. Wavelength conversion - PACnet.

unidirectional fibers, with 8 lightwaves per fiber, *ii.* static traffic, and, *iii.* fixed shortest path routing.

As shown in Figure 4, the proposed model presents a clear improvement, in terms of blocking probability, over SIR and DIR for non-bursty traffic. As an informal model validation, one can observe in Figure 5, the improvement when using wavelength conversion, under the same simulation instance as that of Figure 4.

VI. Conclusion

A novel model, which is based upon the principles of Cooperative Game Theory, has been presented for the first time to our knowledge.

By comparing simulation results with the performance obtained from DIR, a non-trivial improvement was found. However, the most important contribution obtained from this work does not necessarily lie on quantitative blocking probability improvement, but rather on the model itself. The distributed RWA problem, where nodes from competing companies can benefit from cooperation, can clearly be modeled as a cooperative game, particulary a $\mathcal{NBP}$.

Most state of the art Distributed Wavelength Assignment algorithms account for some sort of *good faith* from other nodes in the network for assigning lightpaths. It is the case of both DIR and DRCLS (Distributed Relative Capacity Loss), proposed by Zang, Jue and Mukherjee in [16]. In a strictly interconnected scenario, this assumption would delay, and in some cases even impede, the detection of ill-intentioned nodes in the network.

One has to keep in mind that this work is a first attempt at introducing Game Theory concepts, not only to solve the current problem of Wavelength Assignment, but also to account for the inevitable evolution of the deployed networks. A plethora of work lies ahead to obtain a fully tested procedure, to cite a few:

- Exploring utility functions using genetic algorithms.
- Extending simulations with nodes using different utility functions.
- Exploring non-symmetric solutions.
- Considering topologies formed by interconnected long haul providers.
- Analyzing the dynamics and complexity of the proposed method and comparing it to state of the art distributed wavelength assignment methods.

References

[1] B. Mukherjee: Optical WDM Networks. Springer Verlag (2006).

[2] B. Jaumard, C. Meyer ,and B. Thiongane: Comparison of ILP formulations for the RWA problem. Optical Switching and Networking. 4, 157-172 (2007).

[3] M. Sipser: Introduction to the Theory of Computation. Course Technology. 2. Edition (2006).

[4] K. Lu, G. Xiao, and I. Chlamtac: Analysis of blocking probability for distributed lightpath establishment in WDM optical networks. IEEE/ACM Trans. Netw. , 13, 187–197. (2005).

[5] J. von Neumann and O. Morgenstern: Theory of Games and Economic Behavior. Princeton University Press (2004).

[6] M. J. Osborne and A. Rubinstein: A course in Game Theory. The MIT Press (1994).

[7] J. A. Hendler, J. Golbeck: Metcalfe's law, Web 2.0, and the Semantic Web. J. Web Semantics.1, 14–20 (2008).

[8] J. F. Nash, Jr. : Two-Person Cooperative Games. Econometrica. 21, 128-140 (1953).

[9] R. B. Myerson: Game Theory: Analysis of Conflict. Harvard University Press (1997).

[10] J. F. Nash, Jr. : Non-Cooperative Games. The Annals of Mathematics. 54, 286-295 (1951).

[11] J. F. Nash, Jr. : The Bargaining Problem. Econometrica. 33, 155-162 (1950).

[12] V. Bilò, M. Flammini, and L. Moscardelli: On Nash Equilibria in Non-cooperative All-Optical Networks. STACS 2005 LNCS 3404 (2005).

[13] E. W. Dijkstra: A note on two problems in connexion with graphs. Numerische Mathematik. 1, 269–271 (1959).

[14] J. P. Jue and G. Xiao: An Adaptive Routing Algorithm for Wavelength-Routed Optical Networks with a Distributed Control Scheme. Proc. Ninth International Conference on Computer Communications and Networks. 192–197. (2000).

[15] X. Yuan, R. Melhem, R. Gupta, Y. Mei, and C. Qiao: Distributed Control Protocols For Wavelength Reservation and Their Performance Evaluation. Photonic Network Communications. 1, 207-218 (1998).

[16] H. Zang, J. P. Jue, and B. Mukherjee: A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks. Optical Networks Magazine. 1, 47-60 (2000).

Manual Multi-Domain Routing for Géant E2E Links with the I-SHARe Tool

Mark Yampolskiy[1,2,3], Wolfgang Fritz[1,2], Wolfgang Hommel[2,3], Gloria Vuagnin[4], Fulvio Galeazzi[4]
[1] *German Research Network (DFN), Alexanderplatz 1, 10170 Berlin, Germany*
[2] *Leibniz Supercomputing Centre (LRZ), Boltzmannstraße 1, 85748 Garching, Germany*
[3] *Munich Network Management (MNM) Team, Oettingenstraße 67, 80538 München, Germany*
[4] *Italian Research and Education Network (Consortium GARR), Via dei Tizii 6, I00186 Rome, Italy*
{yampolskiy,fritz,hommel}@lrz.de, {gloria.vuagnin,fulvio.galeazzi}@garr.it

***Abstract*—The term *routing* is usually associated with the fully automated routing in computer networks. Various routing techniques, protocols, and strategies have been established in LANs, WANs, Internet, and PSTN networks. However, all these routing approaches rely on a pre-installed, pre-configured, and well-maintained network infrastructure. The process of planning the network infrastructure remains the stronghold of manual planning. Whereas the planning within a single provider domain is a very common task for all network service providers (SPs), the planning of multi-domain backbone connections introduces several additional challenges, such as the coordination of connection options among multiple SPs' planning teams. In this paper we present the *I-SHARe* (Information Sharing across Heterogeneous Administrative Regions) tool, which has been developed in the pan-European collaboration Géant in order to foster inter-provider collaboration during the planning and operation of multi-domain backbone connections.**

***Keywords*-manual routing; backbone connections; management processes; tool-support**

I. Introduction

Géant is a collaboration of over 30 European National Research and Education Networks (NRENs). Whereas the purpose of every NREN is to provide network connections for national research and education institutions, the purpose of Géant is to interconnect NRENs and consequently to foster international research projects where participating organizations are connected to different NRENs. The portfolio of Géant includes various services among other conventional IP connections. However, such services cannot always fulfil all the challenging requirements of modern research collaborations. The Large Hadron Collider (LHC) project provides a very good example. Experiments performed in the CERN center near Geneva produce over 15 petabytes of raw data per year [1]. Mainly to store all this vast amount of data, it has to be transferred to 11 so called Tier-1 (T1) supercomputing centres spread across Europa and North America [2]. The analysis of the data is then performed in 160 T2 supercomputing centres spread around the entire world. CERN backups all the data on tape but has the local high speed storage capacity sufficient only to save experimental data of few days. Therefore, a bad network quality or long term outages of network connections between the T0 centre at CERN and the T1 centres might lead to an inability to process real time the data taken, with a negative impact on the analysis results. A bad quality of connections between T1 and T2 centres is not that critical, but nevertheless might lead as well to significant delays of the data analysis. Therefore, LHC needs permanent high-bandwidth and high availability connections between involved research organizations. Other good examples are Grids, e.g., WLCG [3] or EGEE [4], where the involved supercomputing centres require network connections as a means for job-transfers between the collaborating partners.

Realizing high-quality high-bandwidth connections in general purpose IP networks is very difficult, as communication flows can interfere with each other and therefore lead to bad connection quality. In order to cope with the challenging customer requirements, a novel service, End-to-End (E2E) Links, has been introduced in Géant. E2E Links are dedicated optical point-to-point connections realized at ISO/OSI levels 1 and 2, with connection segments provided by one or more NRENs [5]. Regarding their quality requirements, the used network technologies as well as the geographical dimensions, E2E Links are nothing else but multi-domain backbone connections, in which—in opposite to classical backbones—multiple network providers are involved and heterogeneous network technologies can be used.

The E2E Links service has been first introduced mid 2005. The speciality of the service is that a new connection can be ordered regardless of whether the required infrastructure is already installed or not. If new infrastructure is needed to fulfil the customer's request, it can be procured, installed, and configured according to the requirements to the new E2E Link. Consequently, all route planning procedures can only be done manually and require intensive interactions between the involved NRENs.

The experience made in the first years of the E2E Links service has revealed that information exchange and information management are key factors determining the time needed for manual connection planning and installation. Information exchange via email and planning via Excel sheets has proven to be error prone with a high probability of losing necessary information or missing various events, e.g., the delivery of the procured infrastructure by neighbour NRENs. This results in a high fluctuation of the time needed to plan and install new connections. In order to improve the outlined situation, a tool supporting the information exchange among participating NRENs should be introduced for the E2E Links service.

The design and development of the *I-SHARe* (Information Sharing across Heterogeneous Administrative Regions) tool has been performed by an international team of researchers working for different NRENs. The *I-SHARe* tool covers information exchange for the whole life cycle of an E2E Link service instance, from its planning through installation and operation till decommissioning. In this paper, we focus on the tool support for the planning of a new E2E Link. We outline the most important challenges of such planning procedures in Section II. The manual routing procedure with *I-SHARe* is presented in Section III. In Section IV, we provide a brief history of the tool's design and development. In Section V we present our future plans. The presented paper aims to promote the gained knowledge about the *I-SHARe* tool, so that network operators facing similar challenges can use it as well.

II. SPECIFIC CHALLENGES

Manual route planning has to overcome a combination of technical and organizational challenges. Both types are caused by the organizational independence of the involved NRENs.

Due to various domain-internal reasons, it is typical for organizations to have very restrictive information policies. This means that the amount of information, which is allowed to be shared with other project partners, is very limited. For instance, it is typically prohibited to share detailed information about the physical network topology or total capacity available on the already installed infrastructure. On the other hand, as NRENs have to collaborate with each other on service instance planning and realization, it is broadly acceptable to share service-instance-bound information.

Further, independent organizations tend to have different procurement policies and various preferences regarding hardware vendors and technologies. This is often caused by legal issues like procurement rules, by past experience, and contractual conditions with various hardware vendors as well as by the competence of organizational members with particular technology. Consequently, this results in a high level of heterogeneity of hardware and networking technologies used by different NRENs. The interconnection of different technologies is not an easy but very well understood task. This, however, requires the consideration of the compatibility of the used hardware, e.g., network interfaces, and network parameters like the maximal supported frame size. Sharing such information is essential in order to prevent problems caused by incompatibility or misconfiguration.

The planning of the network infrastructure is done manually in each NREN. It is inevitable that some infrastructure might have to be procured and information about both financial conditions and hardware properties should be first requested from the hardware vendor(s). This introduces unpredictable delays and an uncertainty of properties that NRENs will be able to provide, as for instance some hardware used in the past might become obsolete and is not supplied anymore. As all NRENs are independent organizations, changes of the planning conditions, e.g., the ordered infrastructure should be delivered at a certain time, are not automatically known to other involved partners. This raises the necessity to notify other involved partners about the completion of the own planning part and about the properties available for the connection.

In conjunction with the information exchange, a reduction of the information flood is also needed. This is especially important as the tool has to support manual processes. The reduction of information means, for instance, that only relevant partners should be involved and not all NRENs involved in the collaboration and the necessary information do not need to be resent many times. Furthermore, especially for the planning of interconnecting interfaces it is important to know the plans of the neighbour NREN for the particular connection instance, in opposite to all interfaces planned for all instances.

Last but not least, the coordination aspect has to be mentioned. As network planning teams of all involved NRENs operate independently, some sort of coordination is needed in order to achieve the common goal—planning of a new E2E Link. The lack of such information can result in unresolved deadlocks, if, for example, two neighbour NRENs have simultaneously planned incompatible infrastructure. Finally, the outlined information exchange has to be embedded in multi-domain operational procedures.

III. PLANNING A NEW ROUTE WITH I-SHARE

If a project like LHC requires a new E2E Link, for example, between CERN and Brookhaven National Laboratory (BNL) in USA, a corresponding request can be submitted to one of the NRENs to which end-points are connected. The project has to specify two end-points and the required properties of a new connection. If the contacted NREN approves the request, network planning team(s) start to work on the planning of a new connection and consequently on its installation. The *I-SHARe* tool is dedicated to support this work.

In this section we first briefly describe the system architecture of *I-SHARe*. After that, we outline the support of the service instance life cycle. Finally, we present the usage functionality of *I-SHARe* by planning and setting up a new connection.

A. Data separation in I-SHARe's system architecture

The handling of single-domain and multi-domain information is clearly distinguished in the *I-SHARe* system architecture (see Figure 1). Information such as operational groups and group members, their responsibility areas and contact data are handled in the *domain part*. This information can be maintained in the *I-SHARe* domain part, or in an NREN's domestic management tool. In both cases the single-domain information is propagated to *I-SHARe* via the *I-SHARe Domain Interface*. The *I-SHARe Central Server* stores the copy of the provided information, so that it can be incorporated in the supported processes.

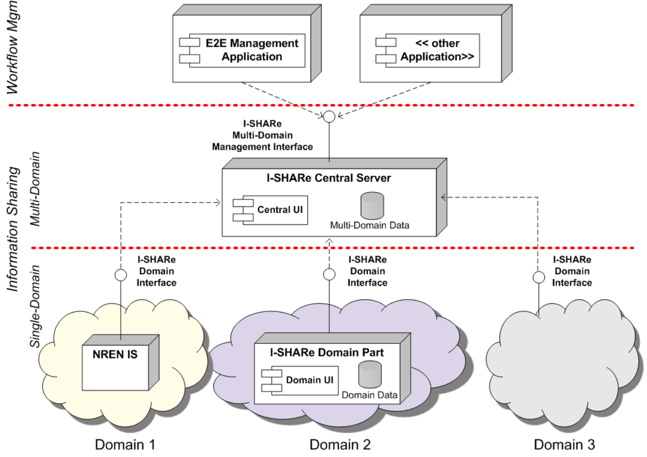


Figure 1. System architecture of *I-SHARe* [6]

Multi-domain information like the route of an E2E Link through NRENs, interfaces of the adjacent connection parts provided by neighbor NRENs, and states of various operations are stored directly in the *I-SHARe Central Server*. This information can be accessed and edited via a web-based GUI (see Sections III-C and III-D for the detailed description).

All information stored in the *I-SHARe Central Server* can be accessed from other applications through the *I-SHARe Multi-Domain Interface*. This north-bound interface provides the means for integrating *I-SHARe* with other tools, e.g., with workflow management or analysis tools.

B. Life cycle coverage in I-SHARe

I-SHARe is designed to support the multi-domain manual management processes during the whole life cycle of E2E Links. According to the specifics of the E2E Links service, one has to distinguish between four phases: (i) *Ordering* of a new E2E Link, (ii) *Setting up* of the ordered E2E Link, (iii) *Operation* of E2E Links in-service, and (iv) *Decommissioning* of no longer needed E2E Links. E2E Links in different phases can be accessed through different views (top-level tabs in the GUI) of the *I-SHARe Central Server*. The distinguishing between views is needed, as the tasks in the corresponding phases require different knowledge, skills, and competences, which generally are provided by different teams. Furthermore, also the information needed in various phases overlaps only partially.

The GUI views and the corresponding tasks are defined as follows:

- The **Ordered** view contains all links that have been requested. During the first phase the general feasibility of the requested E2E Link with the specified quality parameters is investigated. Among other tasks, this includes the selection of NRENs, which should participate in the realization of E2E Link segments.
- The **Set Up** view incorporates information about all ordered E2E Links, whose general feasibility has been approved. The purpose of this phase is to oversee and coordinate all steps needed for the establishing of the planned connection, i.e., the installation and configuration of the needed equipment, the interconnection of all adjacent connection segments, and the accomplishment of all integration tests needed for the allowance for service.
- The **Operational** view provides access to information about all E2E Links, which are delivered as a service to the end-customers. This view incorporates all plans and details elaborated in the previous phases. The changes in this phase can be used, e.g., in order to plan an upgrade of used infrastructure.
- The **Decommissioned** view provides access to all formerly operational, now obsolete connections. This view can be used, e.g., in order to reuse solutions elaborated for E2E Links, which are not in service anymore.

In the remainder of this section we will present, how the *I-SHARe* tool can support the manual work performed in the first two phases of the service instance life cycle.

C. The ordering process

After the contacted domain approved the request for a new E2E Link, one of the domain's experts needs to login to *I-SHARe* and open a new link request. First, he specifies the end points given by the request and both connecting domains (see Figure 2). The rectangles in the figure represent organizational domains. If domains connecting the end-points are not neighbors, one or more transit-domains can be inserted by clicking the ”+”-button on the right hand of the vertically arranged route. The acronyms of the domains can be selected from a drop-down box in the middle of the rectangle. Furthermore, for each domain the *Point of Presence* (PoP) can be specified, at which it should be connected to the neighbor domain. The connection at the end-point site is not specified, as it is the end-customer's responsibility. In the GUI, the domain-specific PoP list is represented as a drop-down box at the top or bottom edge of the rectangle. The list is provided to the *I-SHARe Central Server* by each domain via the *I-SHARe Domain Interface* (see Section III-A).

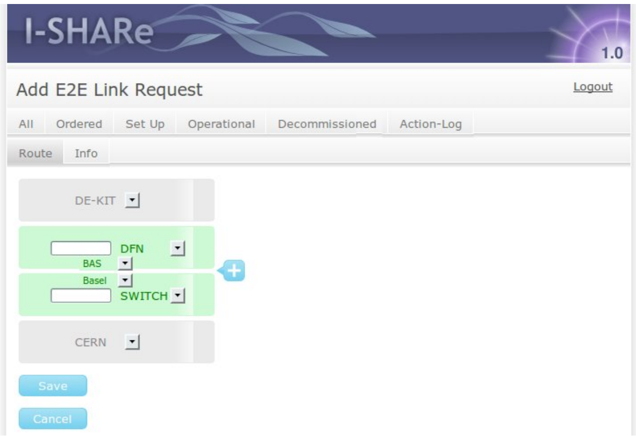
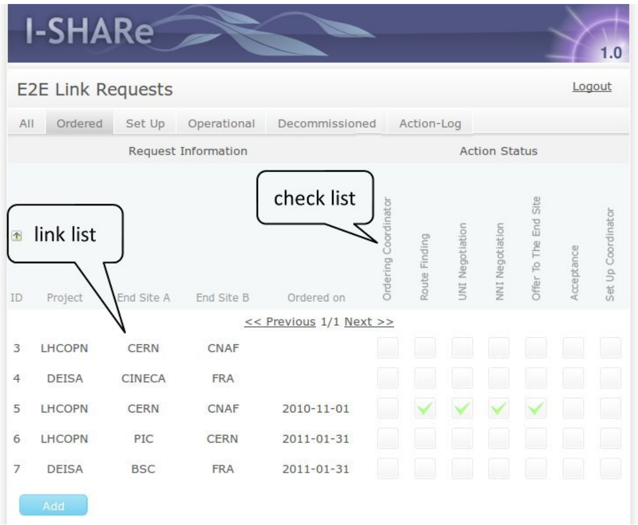
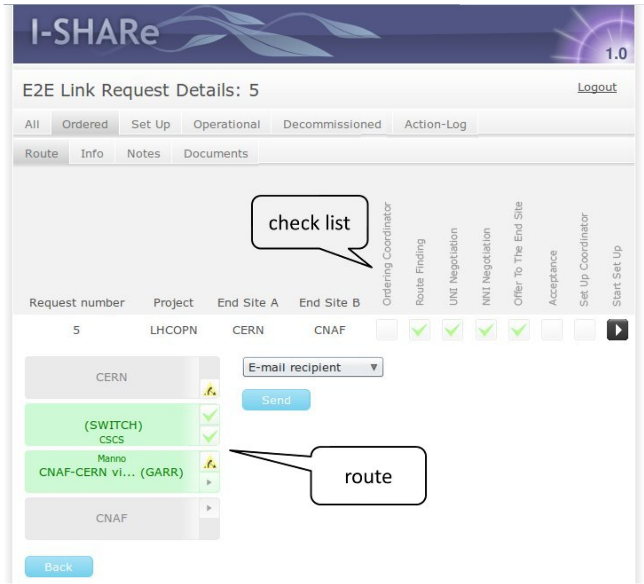


Figure 2. Define end points and connecting domains [7]



ID	Project	End Site A	End Site B	Ordered on	Ordering Coordinator	Route Finding	UNI Negotiation	NNI Negotiation	Offer To The End Site	Acceptance	Set Up Coordinator
3	LHCOPN	CERN	CNAF		☐	☐	☐	☐	☐	☐	☐
4	DEISA	CINECA	FRA		☐	☐	☐	☐	☐	☐	☐
5	LHCOPN	CERN	CNAF	2010-11-01	☐	✓	✓	✓	✓	☐	☐
6	LHCOPN	PIC	CERN	2011-01-31	☐	☐	☐	☐	☐	☐	☐
7	DEISA	BSC	FRA	2011-01-31	☐	☐	☐	☐	☐	☐	☐

Figure 3. I-SHARe's list of ordered links [7]



Request number	Project	End Site A	End Site B	Ordering Coordinator	Route Finding	UNI Negotiation	NNI Negotiation	Offer To The End Site	Acceptance	Set Up Coordinator	Start Set Up
5	LHCOPN	CERN	CNAF	☐	✓	✓	✓	✓	☐	☐	▶

Figure 4. Detailed view of an ordered link [7]

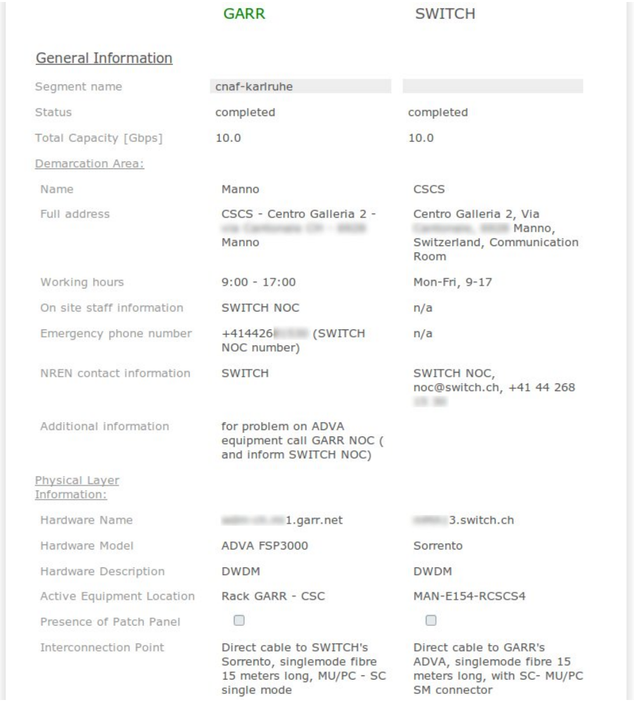
At this stage, the *I-SHARe* tool also asks for further relevant information, such as the assigned project's name and the customer requirements, e.g., the guaranteed bandwidth. Other relevant entries, like the request date, are added automatically. Besides the information already published by the domain part, more contact details may be specified then as well. Furthermore, already at this stage it is possible to assign an *Ordering Coordinator* (OC)—a special role responsible for coordinating the actual ordering process among different steps, persons, and institutions. The OC makes sure everyone takes the right actions and keeps track of the overall progress. The OC is selected from the list of all network specialists (reported via the domain interface), whose domain designated him as qualified to take the responsibility for this role. Usually, the OC is selected from members of the connecting domain.

After a new link request is saved, it will appear in the *Ordered* view (top-level tab in the GUI). This view is divided into two parts—a link list showing the pending link requests and a check list indicating the progress (see Figure 3).

By clicking at an ordered link, its detailed—alike divided—view can be accessed (see Figure 4). The route part shows all involved end sites and domains, whose detailed (contact) information people can access by clicking at them.

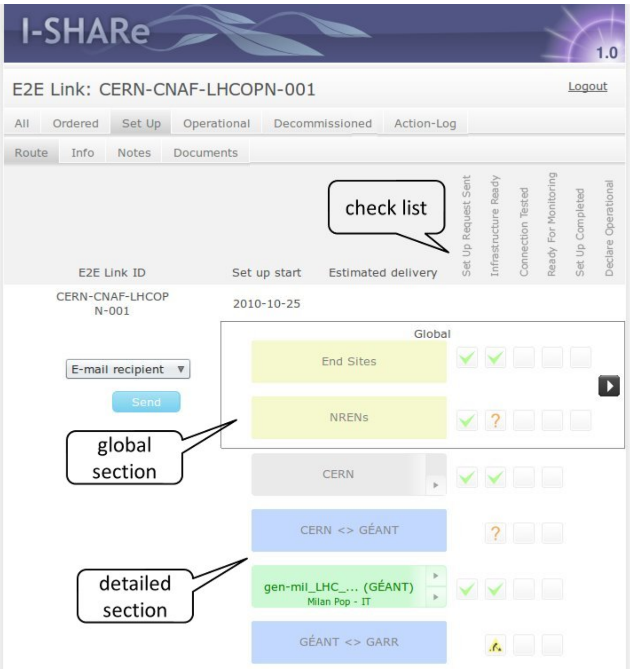
The check list contains the state of all steps that the experts have to take before the link may be "promoted" to the next life cycle phase. It covers the selection of an *ordering coordinator*, *route finding*, *UNI negotiation*, *NNI negotiation*, *offer to the end site*, *acceptance*, and selection of *set up coordinator*. First, all involved domains need to agree on an OC, which is the first action. After that, they have to work together to find a feasible route from the requesting customer to the destination. The OC and the experts are supposed to set the individual states so that other people can easily keep track of the link negotiation's progress. During both, *UNI and NNI negotiation*, the experts provide general and technical information about their interfaces, such as its capacity, hardware, and interface type, where it is located, transceiver and media types or even fine grained parameters such as the MTU size or the used wavelength. Figure 5 outlines a typical view during NNI negotiation. In this view, a member of one domain can edit various parameters and at the same time see the corresponding parameters its neighbor specified.

In order to notify neighboring domains about different events, *I-SHARe* provides user friendly email functionalities, too. The tool supports the selection of recipients relevant for the particular connection. Further, at any time notes and documents can be stored in the system, for example, plans for patch panel interconnection.



	GARR	SWITCH
General Information		
Segment name	cnaf-karlruhe	
Status	completed	completed
Total Capacity [Gbps]	10.0	10.0
Demarcation Area:		
Name	Manno	CSCS
Full address	CSCS - Centro Galleria 2 - [redacted] Manno	Centro Galleria 2, Via [redacted] Manno, Switzerland, Communication Room
Working hours	9:00 - 17:00	Mon-Fri, 9-17
On site staff information	SWITCH NOC	n/a
Emergency phone number	+414426[redacted] (SWITCH NOC number)	n/a
NREN contact information	SWITCH	SWITCH NOC, noc@switch.ch, +41 44 268 [redacted]
Additional information	for problem on ADVA equipment call GARR NOC (and inform SWITCH NOC)	
Physical Layer Information:		
Hardware Name	[redacted]1.garr.net	[redacted]3.switch.ch
Hardware Model	ADVA FSP3000	Sorrento
Hardware Description	DWDM	DWDM
Active Equipment Location	Rack GARR - CSC	MAN-E154-RCSCS4
Presence of Patch Panel	□	□
Interconnection Point	Direct cable to SWITCH's Sorrento, singlemode fibre 15 meters long, MU/PC - SC single mode	Direct cable to GARR's ADVA, singlemode fibre 15 meters long, with SC- MU/PC SM connector

Figure 5. NNI negotiation page [7]



E2E Link: CERN-CNAF-LHCOPN-001

E2E Link ID	Set up start	Estimated delivery	Set Up Request Sent	Infrastructure Ready	Connection Tested	Ready For Monitoring	Set Up Completed	Declare Operational
CERN-CNAF-LHCOPN-001	2010-10-25							

Global	Set Up Request Sent	Infrastructure Ready	Connection Tested	Ready For Monitoring	Set Up Completed
End Sites	✓	✓	□	□	□
NRENs	✓	?	□	□	□

	Set Up Request Sent	Infrastructure Ready	Connection Tested	Ready For Monitoring
CERN	✓	✓	□	□
CERN <> GÉANT		?	□	□
gen-mil_LHC_... (GÉANT) Milan Pop - IT	✓	✓	□	□
GÉANT <> GARR		⚠	□	□

Figure 6. Install and configure the network infrastructure [7]

After all planning steps have been completed successfully and the customer accepted the particular offer, the link may be put to the next life cycle phase, the set-up phase. Prior to that a *Setup Coordinator* (SC) has to be selected— a role similar to OC, but responsible for the coordination of all activities related to the link set up. If the domains could not find a feasible route for a particular link request, they will reject it and inform the customer.

D. The set up process

Similar to the ordering phase, *I-SHARe* provides a separated list for each link that is currently in the process of being established. By clicking at a link, the experts can access a more detailed view that is alike divided. On the one hand, there is a more detailed description of the selected route—now not only containing sites, but also interconnecting links. It is itself divided into two parts—one with detailed check boxes, where the actual states can be set depending on the type of the entry and a global section that computes the overall states automatically, based on the segments' states (as indicated in Figure 6). The top part presents a check list (containing *E2E link ID assigned*, *set up request sent*, *infrastructure ready*, *connection tested*, ready for monitoring, *set up completed*—see Figure 6). The first thing *I-SHARe* will ask for is a unique name for the link (until now, the identifier was just a number). After that, the SC asks all involved domains to start the actual hardware installation and configuration (apply connectors, fibres, etc.) and indicate that by setting *infrastructure ready* properly (done, work in progress or delayed, see Figure 6). This step can be done by various domains simultaneously to save time. As soon as all involved NRENs have completed that step successfully, they need to test the new connection. This may not only include local test, but also tests in cooperation with some or all of the other partners to guarantee the whole link is working. Last but not least, the experts have to include that new link in their local monitoring systems and export information about it to the Géant multi-domain monitoring tool. This allows the experts to keep track of the link's current status. After all previous steps have been completed, the SC marks the last step, *set up completed*, as done, and then declares the link operational. That means the whole process was successful and the customer's request lead to a new E2E link, which can be then used by the end users.

IV. Short history of I-SHARe

As the *I-SHARe* tool had to be developed within and provided to an international collaboration of different NRENs, the whole *I-SHARe* team has also been assembled from members of the participating NRENs. In order to separate responsibilities for key aspects and hereby avoid conflict of interests, the *I-SHARe* team consisted of a designer and a development team from the beginning.

The design of the *I-SHARe* tool has been led by the Italian and the German NRENs (GARR and DFN respectively) with a strong participation of the Swiss NREN (SWITCH). During different phases members of RENATER (French NREN) and DANTE (operator of the Géant network) have also participated in the efforts of the designer team. Only NREN members with experience in requirements analysis, system design and other key project management tasks have been assigned to the designer team. The *I-SHARe* development team consists of members of the Polish supercomputing centre PSNC (working for the Polish NREN PIONIER), who have proven to be experienced in the development of web-based applications.

The major difficulty of *I-SHARe's* design was to gather and analyse the operations' requirements. The interviews with network planning and operational teams have begun in the mid of 2008 and revealed different and sometimes even controversial needs of these teams. The de-facto multi-domain process was defined by the *I-SHARe* designer team in the Géant deliverable DS3.16 [8]. After the NRENs had approved it, this deliverable has been used to identify necessary information that have to be exchanged during the different steps planning, setting up, and operation of E2E links. In order to evaluate gathered requirements, an *I-SHARe* prototype with reduced functionality has been designed. Implemented by the *I-SHARe* development team, this prototype has been evaluated by the NRENs' operational teams. The received feedback has been used to improve the *I-SHARe* system specification, which has been finished in summer 2009. The implementation of the first fully functional version of *I-SHARe* tool has been finished in summer 2010.

After the quality assurance performed by the designer team, *I-SHARe* v 1.0 was approved for the 6 month long pilot phase. During this phase, the operations of selected NRENs evaluate the suitability of the developed tool for their daily work regarding the planning and the management of E2E Links. For the pilot phase the *I-SHARe* installation is hosted at Leibniz Supercomputing Centre (LRZ), a tight partner of DFN. The pilot phase started at the end of 2010. NRENs participating in the pilot phase are (in alphabetical order) DANTE, DFN, GARR, PIONIER, REDIRIS and SWITCH. In order to introduce *I-SHARe*, an online training course has been delivered to the network operation teams of these NRENs.

V. First Experiences and Further Steps

During the requirement analysis, system design, development, and quality assurance stages only one team (either designers or developers) was in charge of a particular task at a time. Interactions with potential users and among these teams took place with clear responsibilities and were of rather simple nature. The start of the pilot phase has introduced the necessity of communication not only among these two teams, but also with end-users and the hosting provider. In order to overcome possible misunderstandings and deadlocks, the designer team is now in charge for developing a proposal for operational procedures covering the whole life cycle of *I-SHARe*. Among other, procedures are about to be defined for the treatment of user-feedback, planning new releases, rollout by the hosting provider, and Incident & Problem management during the *I-SHARe* operation. These procedures will define roles, their rights and responsibilities as well as the way of interaction in different situations.

Another development is planned after the *I-SHARe* operation is settled and broadly used by NRENs. The main goal of *I-SHARe* is to support the information exchange between manually performed operational processes. In case these processes become settled and may be even standardized, the development of another tool for workflow management is planned. This tool should reuse *I-SHARe* as an information exchange platform and access it via the "north bound" interface already implemented in the tool.

Acknowledgment

The authors would like to thank all organizations participating in the design, development and operation of *I-SHARe*, especially DFN, GARR, PIONIER, SWITCH, LRZ and PSNC. The authors also wish to thank the members of the Munich Network Management (MNM) Team for valuable comments on previous versions of this paper. The MNM Team directed by Prof. Dr. Dieter Kranzlmüller and Prof. Dr. Heinz-Gerd Hegering is a group of researchers at Ludwig-Maximilians-Universität München (LMU), Technische Universität München (TUM), the University of the Federal Armed Forces and the Leibniz Supercomputing Centre (LRZ) of the Bavarian Academy of Science.

References

[1] Knobloch, J. and Robertson, L., "LHC computing Grid, Technical design report," CERN, http://lcg.web.cern.ch/LCG/tdr/, Tech. Rep., 2006 (last accessed 2011/04/13).

[2] "LHC website," http://public.web.cern.ch, (last accessed 2011/04/13).

[3] "WLCG website," http://lcg.web.cern.ch/LCG/public/, (last accessed 2011/04/13).

[4] "EGEE website," http://www.eu-egee.org/, (last accessed 2011/04/13).

[5] Schauerhammer, K. and Ullmann, K., "Operational Model for E2E links in the NREN/GÉANT2 and NREN/Cross-Border-Fibre supplied optical platform," Géant, Tech. Rep., 2006.

[6] Cesaroni, G. and Hamm, M. and Simon, F. and Vuagnin, G. and Yampolskiy, M. and Labedzki, M. and Wolski, M., "I-SHARe: Prototype specification," Géant, Tech. Rep., 2008.

[7] "I-SHARe pilot installation," cs.ishare.geant.net/, (last accessed 2011/04/13).

[8] De Marinis, E. and Hamm, M. and Hanemann, A. and Vuagnin, G. and Yampolskiy, M. and Cesaroni, G. and Thomas, S.-M., "Deliverable DS3.16: Use Cases and Requirements Analysis for I-SHARe," Géant, Tech. Rep., 2008.

Dynamic Decision Engine for Data Connections Routing

Sajjad Ali Musthaq, Christophe Lohr, Annie Gravey
TELECOM Bretagne Department of Computer Science Brest, France
Email: {sajjad.musthaq, christophe.lohr, annie.gravey}@telecom-bretagne.eu

***Abstract*—Companies nowadays are subscribing links to several Internet Service Providers (ISPs) for reliability, redundancy and better revenues underlying the service extension, while providing good Quality of Service (QoS). A dynamic decision-making framework is presented for SOCKS based data services over a multihomed platform that is primarily architectured for multimedia services. The decision engine takes multiple criteria (attributes from context of the request, platform's latest conditional parameters, business objectives of the company, etc.) into account while computing the routing decision. Two Multi-Criteria Decision Making (MCDM) methods, namely Analytical Hierarchy Process (AHP) and Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) are used for weight calculation and decision-making respectively. The system supports outsourcing and provisioning decision enforcement modes. The proposed solution gives higher throughput and lower connection dropping probability with an add-on susceptible delay while fulfilling the desired goals, taking into account the multiple attributes for choosing the best alternative.**

***Index Terms*—Decision Engine, Multi-Criteria Decision Making (MCDM), Connection Dropping Probability, Throughput**

I. Introduction

Legacy network infrastructures and technologies cannot guarantee the Quality of Service (QoS), Quality of Experience (QoE) and performance requirements of voice/video and data services (FTP, Web, Mail) all together as they need diverse resources with varying set of QoS parameters. Companies use Internet to deliver these services with desired QoS. Traditionally, companies purchase multiple links to the Internet from different service providers (termed as multihoming) to address the versatile QoS requirement issue. Companies with mulihoming support also require the ability to ensure that connections in their networks are routed according to the optimal route to maximize their income and to ensure the required level of service performance. Although the primary purpose of multihoming is to enhance reliability of the network, it is also desirable to use multihoming for Load Balancing (LB) and latency reduction. However, intelligent route control/selection allows companies to take advantage of the path diversity that multihoming provides, to improve network performance while using the resources effectively and efficiently. Policy-based Border Gateway Protocol (BGP) [1] deployment is used to address the intelligent route control issue in multihomed environments. However, BGP deployment is costly and requires lots of administration effort and hence does not suit small-to-medium business. Decision-making in the intelligent route control/selection plays a crucial role in mulihomed systems. The system must take into account context of the request, platform's environmental conditions/parameters, state of the links, predefined routing rules and business objectives of the company. Multimedia sessions/connections (voice/video and other quadruple services) carry enough information about the context of the request during the signaling phase (e.g., Session Initiation Protocol (SIP) [2]) as opposed to data (FTP, Web, Email) connections. This information is exploited in decision-making for routing the request to an appropriate link in mulithomed network. The information availability with certain limitation can have an impact on dynamic decision-making for request routing in mulithomed environment.

The objective of this work is to provide a dynamic policy controlled decision-making framework (decision computation and its enforcement) for Transmission Control Protocol (TCP) based SOCKetS (SOCKS) [3] connections/session routing in consent with ongoing multimedia services over the same platform. There are mechanisms for controlling the traffic at private-public network border (e.g., Connection/Call Admission Control (CAC), Least Cost Routing (LCR), etc.). However, the decision-making mechanisms involved in these systems are usually static and/or semi-dynamic. Moreover, these systems take into account few attributes among the set of available parameters over the platform, while calculating the decision (service profile, reliability information, time of the day, business objectives of the company, latest state of the links, user profiles and Service Level Agreement (SLA) etc.). It is important to mention here that the scope of an SLA is limited to exploit the relevant Service Level Specification (SLS) information extracted from the direct and/or reciprocal agreement between the company and service provider within Policy Server (PS) for decision computation.

The underlying information stated above (which has to be taken into account for request routing) comes from different sources with different dimensions, hence formulating a multi-criterion problem. The first challenge is to utilize the available information over the platform maximally, so that the final decision for link selection reflects dynamic control and effective resource utilization with good QoS. Another objective is to enforce the calculated decision using existing technologies (e.g., Network Address Translation (NAT)ing, Domain Name Service (DNS) Cycling, Hashing, Proxying etc.) without introducing overheads in the protocol stack. Multi Criteria Decision Making (MCDM) theory has been applied in order to use this multidimensional info for routing decision computation. Internet Engineering Task Force (IETF)

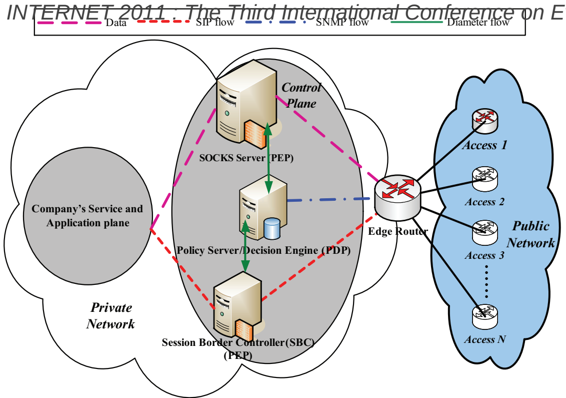


Fig. 1. Proposed Architecture.

conventional Policy-Based Network Management (PBNM) [4] framework involving Policy Decision Point (PDP), Policy Enforcement Point (PEP) and Local PDP (LPDP) is followed for decision enforcement in outsourcing and provisioning modes.

The architecture shown in Fig. 1 is proposed in Companym@ges [5] project which provides a platform where companies are linked to the rest of the world via two or more network accesses offering data and multimedia services. PEP functionality for multimedia and data services is splitted and is being performed at two distinct locations (i-e., SBC and SOCKS server respectively). Data connections are routed to the external links by PEP, i-e., SOCKS server while taking into account business intelligence of the company, dynamics about the resources, network issues etc. Admission Control (AC) function has LCR objective and is split into profile and resource based AC and is distributed among PEP and PDP i-e., PS respectively. Communication between PDP and PEP is carried out over IMS 'Gq' [6] interface using Diameter [7] protocol. The private-public traffic management issues at the border-line regarding multimedia services are addressed in our previous work [8], [9]. The present paper is an extension to this work for data traffic. The proposed framework proposes an efficient solution by enhancing and extending the existing standards. Dynamic decision engine computes decisions by taking multiple criteria into account. In this context, tweaking of SOCKS server to act as PEP, support for decision enforcement in outsourcing (on-the-fly) and provisioning modes (off-line) respectively and the introduction of MCDM theory to solve the multi-criterion network problem are the main contributions from our side. It is an ultra lightweight solution for dynamic LCR implementation in SOCKS communication framework under the control of policy decisions.

The remainder of this paper is organized as follows. In the following Section, we describe the proposed architecture. Section III elaborates the decision theory and the application of MCDM methods. Section IV presents realization of the framework, its functionality, tools tweaked and the decision enforcement modes. In Section V, the test bed for the validity of the proposed solution is presented. Section VI outlines related work. Finally, in Section VII, concluding remarks are made while outlining the future work.

II. SYSTEM ARCHITECTURE

QoS-centered architecture integrates devices and modules from different vendors over a single platform while offering multimedia and data services for public and private (local) networks. One of the objectives of the proposed architecture shown in Fig. 1 is the accommodation of dynamic modifications/variations into the decision-making criteria for request routing to different links by using enhanced general methods/techniques. Service, control and transfer planes issues posing a multi-criteria problem are handled together without affecting the standard mechanisms and classical layered approach. The platform supports the enhanced standard protocols (e.g SIP) [2], SOCKS [3], Diameter [7], etc.) without employing overheads while the dynamics over those three planes are taken into account in decision-making.

A SOCKS based framework for the control and management in multihoming scenario is presented in order to provide better than Best Effort (BE) QoS for data services. The underlying Companym@ges [5] project stems from competitivity cluster for handling traffic management issues at the private-public network border. Components of this platform (Fig. 1) are provided by partners: the platform's service and application plane is realized by modules from Alcatel-Lucent whereas SBC, PS and SOCKS server are/will be developed and tweaked by two different teams at TELECOM Bretagne Brest. Moreover, these modules can be integrated in a single box; however, there are different teams/partners involved in this project with their dedicated solutions/packages (stand-alone boxes).

Policy Server (PS) is the main controller in the proposed architecture. It acts as a PDP. It computes all the decisions by taking into account the static configurations and dynamics taking place over the platform, in addition to the policy enforcement supervision. Decision engine proposed here while using MCDM theory partly constitutes the core of PS. It is worthwhile to mention here that functional decision engine is embedded within the architecture but PS is in development phase, so the rules are entered manually via web-based front end. Session Border Controller (SBC) in the offered framework is primarily dedicated to multimedia communication. It provides a number of vendor specific functionalities depending on the requirements and its deployment. More details are available in [8], [9].

SOCKS server is implemented in the application layer while the client is a shim-layer between application and transport layer (TCP/IP Layer Model). It allows hosts located on one side of a SOCKS server to gain controlled access to hosts located on the other side with configured rules and policies. It acts as a TCP forwarder on demand. The control, management and rule administration/application is static but we introduce dynamicity in decision-making for link selection by taking into account various parameters and attributes (user profile, QoS profile, latest state of the links/SLAs associated with these links, predefined configuration over the platform). TCP based SOCKS data connections are targeted here in this work (User Datagram Protocol (UDP) support is available in SOCKS5).

SOCKS is an application independent transport-level forwarder offering Authentication, Authorization (AA). It works in client-server mode and provides NAT/Port Address Translation (PAT) traversal and firewall services. SOCKS native AA

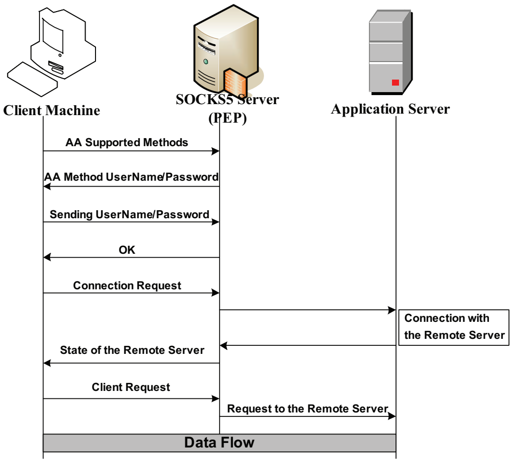

Client Machine
SOCKS5 Server (PEP)
Application Server
AA Supported Methods
AA Method UserName/Password
Sending UserName/Password
OK
Connection Request
Connection with the Remote Server
State of the Remote Server
Client Request
Request to the Remote Server
Data Flow

Fig. 2. SOCKS (Client, Server) Communication with Application Server.

functionality and its bi-directional proxy characteristics facilitates CAC but with static implementation. Its firewall traversal capability complements the routing at higher layers (Open System Interconnection (OSI) Application, Session layers) but with static rules and configurations. Typical communication between SOCKS (Client-Server) and the application server is shown in Fig. 2. After the AA mechanism, the connection established from client to SOCKS server carries client data to be forwarded by the SOCKS server to the application server through a simple TCP connection.

SOCKS is chosen for the introduction of dynamic decision-making framework due to the fact as network application stacks may integrate SOCKS capabilities for managing TCP connections (e.g., web browser, but some of the most popular ones lacks AA mechanism). If they don't facilitate SOCKS client then we may use a wrapper to insert SOCKS request at the top of TCP connection between the application and transport layer (TCP/IP layer model). For each socksified client request, there will be a decision to be enforced at SOCKS Server while routing the request to an appropriate link. The aforementioned data connection routing mechanism at network layer can be made workable by configuring the routers manually (access-list, route-map, static routing, etc.). However, this is not an elegant and efficient way due to its static nature and performance issues. Conventional proxy (Web) working above network layer is performing almost the similar functionality but the rules and policy enforcement are the same for all the traffic (unless a manual change is carried out), and are dedicated to web applications. There are some SOCKS proxy solutions available (e.g., Dante [10], one of the most sophisticated) but they do not take into account the latest dynamics of the platform from application, control and network point of view.

The protocol chosen to communicate the information/decisions between PDP and PEP is Diameter with newly defined and developed Attribute Value Pairs (AVPs). Diameter is natively an Authentication Authorization Accounting (AAA) protocol. Due to its AAA characteristics, its enhancement orientations are becoming natural for decision-based network management. It has large AVP space and supports large number of pending requests. Common Open Policy Service (COPS) [11], a strong candidate for PBNM has not been chosen for decision (policy) provisioning and dissemination, as it is specifically designed for device-level configuration and management. However, dynamic session/call/data-connection management is required while taking into account the variations and latest dynamics. SNMP has sometimes been proposed in the literature to be a candidate for PBNM [12]. SNMP-based information in our system is exploited to gauge the QoS parameters of access router interfaces. In case of communication failure between PDP and PEP, pre computed default rules are enforced depending on the context of the request offering ordinary QoS.

This paper addresses the private-public border traffic management issues for request routing decisions at the application layer (OSI). It supports dynamicity by using Multi-Criteria Decision Making (MCDM) theory. The calculated decisions are enforced in outsourcing and provisioning modes by using existing mechanisms mentioned subsequently.

III. Multi-Criteria Decision Making Theory and its Application in Dynamic Routing

MCDM involves choosing the best alternative, given a set of alternatives (available links here in the architecture) and a set of criteria (context of the request and predefined configurations/settings over the proposed platform). These alternatives are ranked on the basis of multiple criteria using some specific MCDM method. MCDM methods have been used to help solve a wide variety of problems in many different applications such as telecommunications, manufacturing, transportation and software engineering [13], [14]. There is not a single MCDM technique to deal with all multi-criteria problems. Indeed each situation requires a specific MCDM technique. The choice of technique and its impact on the decision-making is not within the scope of this work and reader is referred to [15] for an overview of this particular domain. However the abnormal behavior shown by certain MCDM methods for particular scenarios and the complexity involved in those methods complements our choice of the presented method for the posed problem. The targeted objectives in the multi-criteria decision-making problems might sometimes be conflicting and/or overlapping. In the underlying problem, SLA includes Delay (one-way delay is computed by dividing the round-trip delay by 2), Jitter (computed by polling the trapped values in SNMP MIB tables) and Packet Loss (calculated by counting the number of retransmissions in a particular session) (DJPL) which falls under the business objectives of the company when they sign the direct or reciprocal agreements with partners or companies. However, the same sets of parameters (DJPL) are used to grade the QoS of the available links (a link has to be chosen). The triplet (DJPL) can be used to gauge the authorization and authentication of a particular user class (e.g., Gold user must have the best QoS profile, while Silver can be assigned either a good or a satisfactory QoS profile) while executing the context of the request. There are various approaches to deal with such sort of problems each having its pros and cons but we will not address this issue due to space limitations.

Each MCDM problem is associated with multiple attributes. These attributes are linked to the goals and are referred to as decision criteria. Since different criteria represent different dimensions of alternatives, they may conflict with each other (e.g., Cumulative Bandwidth may be confused with Total Bandwidth, traffic measurements, granularity (connection/session/packet level) obsession, cost, etc). The criteria are assigned different weights according to context of the request and the rules defined over the platform. Conventional algorithms used for link selection in multihomed networks are either user-centric or motivated for efficient resource utilization over the platform and/or they are centered towards application optimization for desired QoS. However, to cope with all these multi-criteria goals and objectives, MCDM is chosen. Two MCDM methods have been chosen to address the problem. Analytical Hierarchy Process (AHP) [16] is used to calculate the corresponding weights of the attributes (termed as criteria in terms of MCDM) involved decision-making. The calculated weight values illustrate the relative importance of each attribute and they are used in the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) to rank the links (alternatives). These ranked links will be used to route the SOCKS connections in consent with the business objectives of the company, user profile and platform's configurations/conditions.

A. Problem Formulation and Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) MCDM Method

TOPSIS was developed by Yoon and Hwang [17]. It is an alternative to ELECTRE [18] and is considered to be one of its variants. It is known as a double standard method that evaluates alternatives through two basic criteria. First, the chosen alternative should have the shortest distance from the positive ideal solution and secondly it must be farthest from the negative-ideal solution for a MCDM problem. The perceived positive and negative ideal solutions are based on the range of attribute values available for the alternatives. The distances are measured in Euclidean terms. The Euclidean distance approach is proposed to evaluate the relative closeness of the alternatives to the ideal solution. The reason for choosing TOPSIS is that it will rank/grade the available alternatives (links) whenever applied by taking all the variations/dynamics and static configurations of the platform into account. Moreover, TOPSIS is extended to be applied on interval data (i.e. lower and upper values of an attribute) over the proposed architecture. Moreover, TOPSIS is extended to be applied in the scenario when the exact value of an attribute is not known, then these bounds (upper and lower) are used. The best link among the available alternative links (ranked by the application of an extended TOPSIS) is assigned to request by following the predefined set of criteria. Due to space limitations and to avoid the complexity, TOPSIS is applied using the standard approach.

The system is capable of accommodating large number of links (n) with enormous set of attributes associated to those alternatives (links). But, for brevity and to avoid the complexity of stringent mathematics, 5 attributes are chosen for the application of MCDM methods on 4 alternative links. Figure 3 illustrates the hierarchy of the desired goal, the criteria and the available alternative links. There are four links L_1, L_2, L_3 and L_4 and for the sake of simplicity, Decision Matrix (DM) contains 5 attributes (link Utilization (U), QoS Profile of the Link (QPL), Cost (C), Total Bandwidth (TB) and Allowed Bandwidth (AB)).

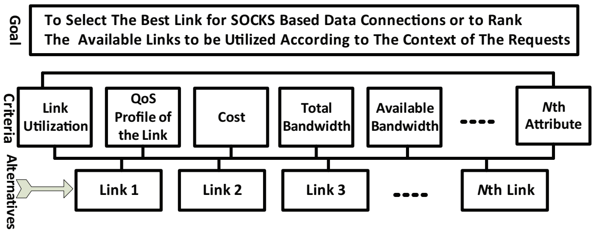


Fig. 3. Candidate Links, Attributes and Objectives Hierarchy.

$$DM = \begin{bmatrix} U_1 & QPL_1 & C_1 & TB_1 & AB_1 \\ U_2 & QPL_2 & C_2 & TB_2 & AB_2 \\ U_3 & QPL_3 & C_3 & TB_3 & AB_3 \\ U_4 & QPL_4 & C_4 & TB_4 & AB_4 \end{bmatrix} \begin{matrix} L_1 \\ L_2 \\ L_3 \\ L_4 \end{matrix} \tag{1}$$

The values of these attributes are obtained from the SNMP traps and the SLAs of the corresponding links over the platform. Moreover the QoS Profile of the link is dependent on DJPL and it is computed by following a predefined criterion embedded by the administrator of the platform. As the parameters involved in the DM come from different sources, the units representing the values are different. We need to normalize these parameters in order to make them unit-less. The attributes having bigger values (e.g., TB is in Mega) are divided by the largest value in the corresponding column vector while the smaller range attribute (e.g., U represented in %age) is divided by the smallest value in the corresponding column vector. The normalized Decision Matrix is given by

$$\widetilde{DM} = \begin{bmatrix} \widetilde{U_1} & \widetilde{QPL_1} & \widetilde{C_1} & \widetilde{TB_1} & \widetilde{AB_1} \\ \widetilde{U_2} & \widetilde{QPL_2} & \widetilde{C_2} & \widetilde{TB_2} & \widetilde{AB_2} \\ \widetilde{U_3} & \widetilde{QPL_3} & \widetilde{C_2} & \widetilde{TB_3} & \widetilde{AB_3} \\ \widetilde{U_4} & \widetilde{QPL_4} & \widetilde{C_4} & \widetilde{TB_4} & \widetilde{AB_4} \end{bmatrix} \begin{matrix} L_1 \\ L_2 \\ L_3 \\ L_4 \end{matrix} \tag{2}$$

Next step is to construct the weighted normalized DM: it cannot be assumed that each evaluation criterion is of equal importance because the evaluation criteria have various meanings. AHP is used to calculate the weight of the corresponding column vector (laying out the criteria) representing an attribute column in the DM. AHP is a MCDM methodology in itself. But its ability to elicit accurate ratio scale measurements and combine them across multiple criteria has led us to use it in conjunction with TOPSIS for ranking the links (alternatives) dynamically. The integration of AHP and TOPSIS is illustrated in Fig. 4. The weighted normalized entities in the DM are represented by subscript wn (e.g., for U will be U_{wn})

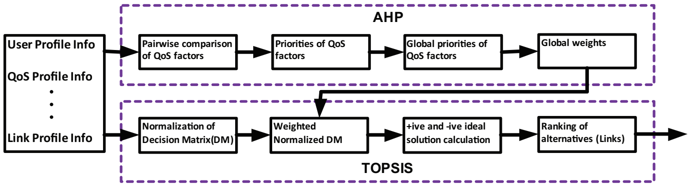

AHP
User Profile Info
QoS Profile Info
Link Profile Info
Pairwise comparison of QoS factors
Priorities of QoS factors
Global priorities of QoS factors
Global weights
Normalization of Decision Matrix(DM)
Weighted Normalized DM
+ive and -ive ideal solution calculation
Ranking of alternatives (Links)
TOPSIS

Fig. 4. TOPSIS and AHP Integration for Ranking of Candidate Links.

Now positive and negative ideal solutions for each attribute are computed: the positive ideal solution indicates the most preferable alternative, and the negative ideal solution indicates the least preferable alternative as follows (e.g., link Utilization, U)

$$U^{+} = \left(Max\left(U_{w-norm}\right)_i\right) \parallel \left(Min\left(U_{w-norm}\right)_i\right), i = 1, 2, 3, 4 \tag{3}$$

and

$$U^{-} = \left(Min\left(U_{w-norm}\right)_i\right) \parallel \left(Max\left(U_{w-norm}\right)_i\right), i = 1, 2, 3, 4 \tag{4}$$

The Euclidean distance method is applied to measure the separation from the positive and negative ideal for each alternative

$$S_i^{+} = \sqrt{\begin{gathered}\left((U_{wn})_i - U^{+}\right)^2 + \left((QPL_{wn})_i - QPL^{+}\right)^2 + \left((C_{wn})_i - C^{+}\right)^2 + \\ \left((TB_{wn})_i - TB^{+}\right)^2 + \left((AB_{wn})_i - AB^{+}\right)^2\end{gathered}} \tag{5}$$

and

$$S_i^{-} = \sqrt{\begin{gathered}\left((U_{wn})_i - U^{-}\right)^2 + \left((QPL_{wn})_i - QPL^{-}\right)^2 + \left((C_{wn})_i - C^{-}\right)^2 + \\ + \left((TB_{wn})_i - TB^{-}\right)^2 + \left((AB_{wn})_i - AB^{-}\right)^2\end{gathered}} \tag{6}$$

Finally, the candidate links are ranked by measuring the relative closeness of an alternative (candidate links L_1, L_2, L_3 and L_4 under consideration represented by a row vector in the Decision Matrix) to the ideal solution S^{+} as follows

$$R_i = \frac{S_i^{+}}{S_i^{+} + S_i^{-}} \tag{7}$$

The links L_1, L_2, L_3 and L_4 characterized by attributes link Utilization U , QoS Profile of the Link (QPL), Cost (C), Total Bandwidth (TB) and Available Bandwidth (AB) respectively are represented by the values shown in table I. For the application of TOPSIS on the links represented by the corresponding row vectors in table I, all the steps mentioned subsequently in this section are gone through in order. The links are ranked with R values as mentioned in table II.

	U	QPL	C	TB	AB
	%	1-10	Cost per byte(Cents)	Megabits per second(Mbps)	Mbps
$\mathbf{L_1}$	66.65	5	0.50	100	65
$\mathbf{L_2}$	53.84	7	0.25	100	71
$\mathbf{L_3}$	81.81	6	0.30	100	81
$\mathbf{L_4}$	25.00	9	0.15	100	46

TABLE I
LINKS WITH CORRESPONDING PARAMETRIC VALUES

	$\mathbf{L_1}$	$\mathbf{L_2}$	$\mathbf{L_3}$	$\mathbf{L_4}$
R Value	0.4025	0.5835	0.4605	0.6344
Rank	4	2	3	1

TABLE II
R VALUES AND THE CORRESPONDING GRADING OF LINKS

IV. COMMUNICATION FRAMEWORK AND ITS FUNCTIONING

A. Realization

An open source SOCKS proxy server is tweaked and enhanced to route data requests according to the context of external links, user information, and resource conditions under the control of decision engine. This article presents an add-on feature within the ongoing Companym@ges [5] project outlining the implementation of well known and existing mechanisms but with novel methodology framing the competitivity and dynamicity of the platform. Jsocks [19] is chosen as the base SOCKS package for modification and addition accordingly. It supports Internet Engineering Task Force's (IETF) SOCKS5 standard and in turn allows more adaptability, flexibility and compatibility. Moreover it requires small enough code analysis and modifications to meet the proof of concept requirements within the framework. Traffix OpenBlox [20] Diameter stack has been adapted to act as IMS Gq interface. It is an implementation of the IETF's Request For Comments (RFC) 3588. It is used for the communication between PS (Decision Engine) and SOCKS server. Diameter Attribute Value Pairs (AVPs) have been developed and used for the required mechanism following the standardized header format. However, the AVP numbers adopted here are non-registered, i-e., these AVPs are understandable onto the platform and within the partners environment only. This methodology has been adopted to avoid the delayed and long AVP registration and approval process however, in the near future, the administrative requirements will be followed. The communication between the Diameter client at SOCKS server and Diameter server within the PS is initiated by Capability Exchange Request (CER) and Capability Exchange Answer (CEA) messages. Negotiation for secured connection (TCL or IPSec) is then performed. The communication starts immediately after the negotiation using Diameter protocol over the Gq interface. WatchDog request/answer messages are often sent to check the keep-alive status of the Diameter client and server. The peers must disconnect formally by sending/receiving the Disconnect Peer Request/Answer (DPR/A). The communication flow graph is shown in Fig. 5. Non-standard AVP identifiers 2221, 2222 and 2223 are chosen for service IP, service port and username respectively. The triplet service IP, service port and username is the reference information for choosing the ranked links (graded by MCDM theory) at PS. The underlying decision is going to be enforced at SOCKS server while routing the request to external link.

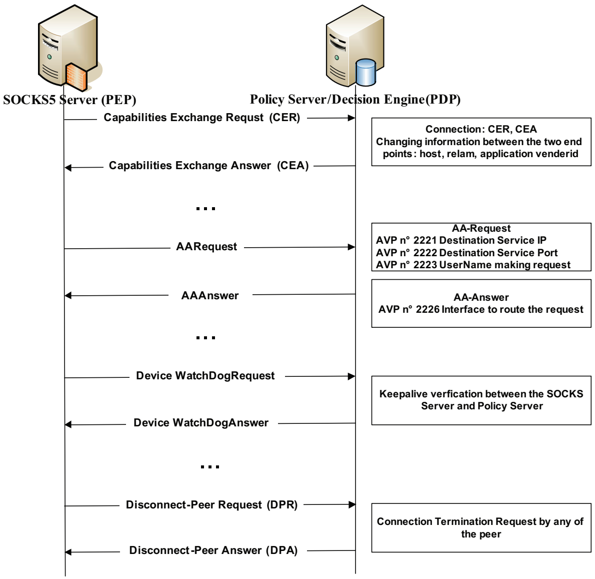

SOCKS5 Server (PEP)
Policy Server/Decision Engine(PDP)
Capabilities Exchange Requst (CER)
Connection: CER, CEA
Changing information between the two end points : host, relam, application venderid
Capabilities Exchange Answer (CEA)
...
AARequest
AA-Request
AVP n° 2221 Destination Service IP
AVP n° 2222 Destination Service Port
AVP n° 2223 UserName making request
AAAnswer
AA-Answer
AVP n° 2226 Interface to route the request
...
Device WatchDogRequest
Keepalive verfication between the SOCKS Server and Policy Server
Device WatchDogAnswer
...
Disconnect-Peer Request (DPR)
Connection Termination Request by any of the peer
Disconnect-Peer Answer (DPA)

Fig. 5. Communication Between Diameter Server and Client at SOCKS Server and PS Respectively.

When a user wishes to access the service over the SOCKS communication platform, a query is sent to the SOCKS server inquiring about the supported methods (Authentication and Authorization (AA) methods. After getting the response, the user then uses an appropriate method for sending the user name and password for AA process to take place at the SOCKS server. A response (an OK) is sent back to the client. The client is now eligible to send the data connection request with the triplet (service type, service IP and service port) info. This info is extracted by the Diameter client at the SOCKS server and is then sent to the PS (Diameter server). In response, the PS sends the decision (one of the ranked links is picked) in consent with the user info and the platform's configurational parameters. Enforcement of the decision takes place to be at SOCKS server. The data connection request is routed to the remote server using that particular interface number (link). The information communication and message exchange between different entities is shown Fig. 6. The remote server answers to the SOCKS server. SOCKS server informs the client about the status of the remote machine. In case it is an OK message, the client then initiates the connection request, which is being routed over the same interface sent previously by the PS in response to the former request.

B. Policy Enforcement Modes

The framework supports two decision enforcement modes namely provisioning and outsourcing. The System however functions in outsourcing mode by default. Whenever a request arrives at the SOCKS server, it extracts the required information and sends this information to Local Policy Decision Point (LPDP) situated at SOCKS server. An appropriate rule from the rule base is mapped and one of ranked links (already available at PEP) is chosen for routing the request in provisioning enforcement mode. It is up to the administrator of the platform to choose either provisioning or outsourcing enforcement mode. However, the policy enforcement irrespective of the two modes is ultimately done at SOCKS server (PEP). In provisioning mode, the pre-computed rules and the ranked links are available at SOCKS server. In outsourcing mode, the extracted information from the pending request is fetched to the PS (PDP) using Diameter Gq interface. PS, in outsourcing mode is delegated to compute/use online/off-line policy/decision depending on the request and the system configuration and conditions (system state). List of ranked links in the provisioning mode are fetched at SOCKS server, a-priori irrespective of online or off-line policy computation. The two enforcement mechanisms are in contrast with each other but they are not mutually exclusive. The policy-based management system is capable of handling both data and multimedia services. However, we are explaining the two policy enforcement modes while considering TCP based SOCKS data connections. The self-explanatory flow graphs give an illusion of the two enforcement modes in Figs. 7 and 8 representing provisioning and outsourcing mode respectively.

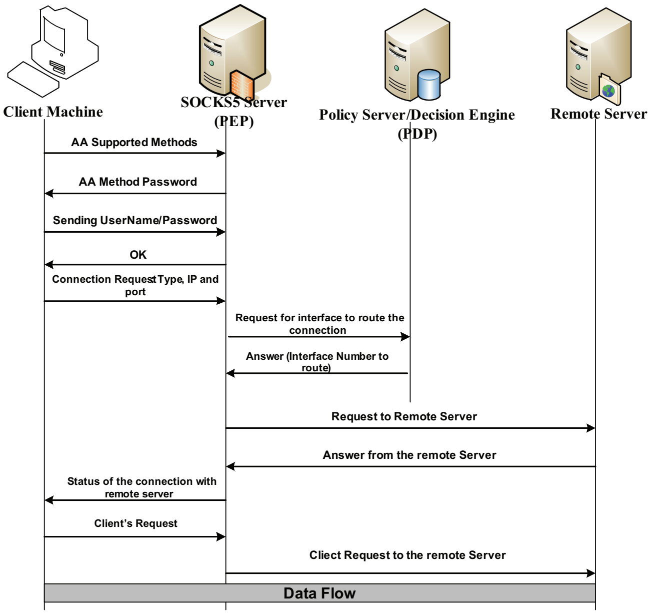

Client Machine
SOCKS5 Server (PEP)
Policy Server/Decision Engine (PDP)
Remote Server
AA Supported Methods
AA Method Password
Sending UserName/Password
OK
Connection RequestType, IP and port
Request for interface to route the connection
Answer (Interface Number to route)
Request to Remote Server
Answer from the remote Server
Status of the connection with remote server
Client's Request
Cliect Request to the remote Server
Data Flow

Fig. 6. SOCKS Communication Framework Flow Graph.

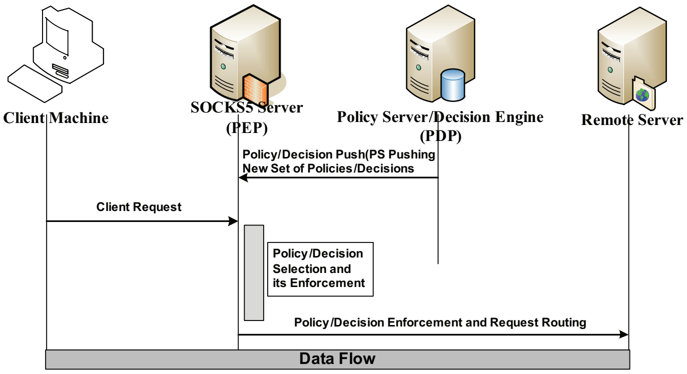

Client Machine
SOCKS5 Server (PEP)
Policy Server/Decision Engine (PDP)
Remote Server
Policy/Decision Push(PS Pushing New Set of Policies/Decisions
Client Request
Policy/Decision Selection and its Enforcement
Policy/Decision Enforcement and Request Routing
Data Flow

Fig. 7. Policy Decision and Enforcement in Provisioning Mode.

C. Provisioning and Outsourcing Mode Comparison

Data traffic is more immune to delay as compared to delay sensitive real time multimedia traffic. So susceptible delay in the two enforcement modes might not make any difference. More resources and computational power are required in outsourcing mode as opposed to provisioning mode. The former mode introduces higher delay than the latter one. Outsourcing mode takes latest platform conditions and network information into account. Provisioning mode, on the other hand, may have conflicts with the platform conditions and/or resource info due to the fluent dynamics onto the platform. Outsourcing mode supports both online and off-line policy/decision computation while provisioning mode has to rely on pre-ranked links.

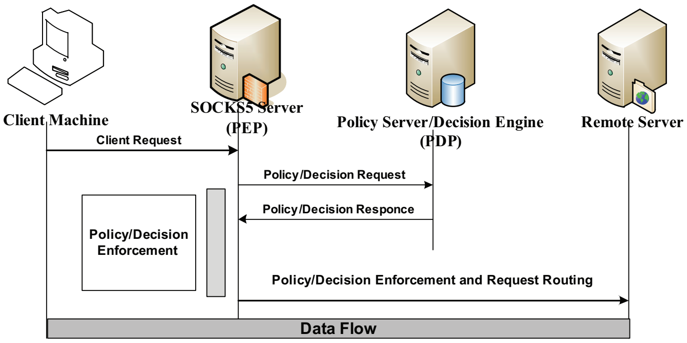


Fig. 8. Policy Decision and Enforcement in Outsourcing Mode.

ISBN: 978-1-61208-141-0

V. Test and Solution validity

The test environment for validating the proposed framework is shown in Fig. 9. The SOCKS server has four interfaces ranging from 1 to 4 for connection to the public network (external links). It has one internal interface for inter-communication within the platform. The proxy server (SOCKS server which is connected to four external interfaces marked as 1, 2, 3 and 4 as shown in Fig. 9) has four different IP subnets. Web Client on the left hand side of the Fig. 9 is connected having IP addresses from those mentioned four subnets. BitTwist [21], an open source traffic generator is used to generate extensive Ethernet-based traffic for stress testing and analysis. It is designed to compliment tcpdump or wireshark packet captures supporting captured file replay. PS is configured to listen to all the requests from SOCKS server on an interface as we are emulating the outsourcing enforcement mode. The links (4 interfaces) are ranked by using the combination of TOPSIS and AHP as explained earlier and the appropriate ranked link (interface) is chosen by following a predefined set of criteria and is ultimately disseminated to the SOCKS server. The remote web server, which is configured to listen on all those interfaces, (linked with SOCKS), displays a webpage showing the IP address of the chosen link. We then calculated the delay introduced by the system with and without decision engine. The graph shown in Fig. 10 indicates that addition of decision engine in the system introduces a minor overhead (delay). This calculation is performed in outsourcing enforcement mode due to more dynamics involved in that particular mode. The delay increases almost linearly as the number of connections increases and the delay is small enough having very little impact on services due to delay-prone and sustainable nature of data traffic. The factors involved in this minor delay are: firstly decision engine is not populated with complete data sets, so the decision computation introduces negligibly small delay, secondly the TCP also contributes to this delay due to its native connection oriented approach. Thirdly, the test is performed using Personal Computers (PCs) with 100 Megabits per second (Mbps) Ethernet interfaces, so the carrier grade hardware with giga speed inter-communication interfaces/channels can make a difference. Finally, the information extraction from the request at SOCKS server which is going to be sent to the PS for decision-making and the policy enforcement mechanism also contributes to this delay. The testing of the same platform while using UDP may be an interesting future work. Throughput of each link is plotted with and without decision engine as shown in Fig. 11 It is observed that there is a significant improvement in the throughput for each link with decision engine. This improvement illustrates the effective and efficient utilization of resources by decision system by taking all the dynamics and variations along with business rules of the platform. Decision system supports connection level granularity so the connection dropping probability is also plotted with and without the underlying Decision Engine. It is observed that the aggregated connection dropping probability with decision engine of the four links has lower value than without it as shown in Fig. 12. The presented decision engine is relatively simple and easy to realize the computer programming so can be easily embedded into systems with little complexity.

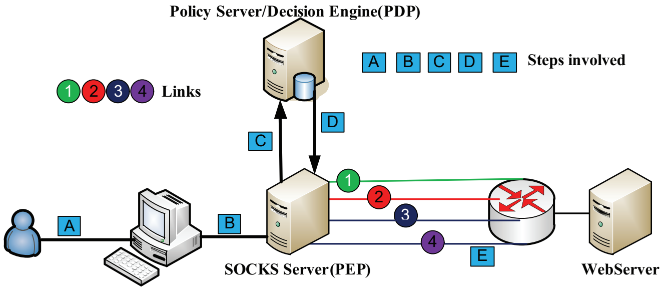


Fig. 9. SOCKS Communication Framework Test Environment.

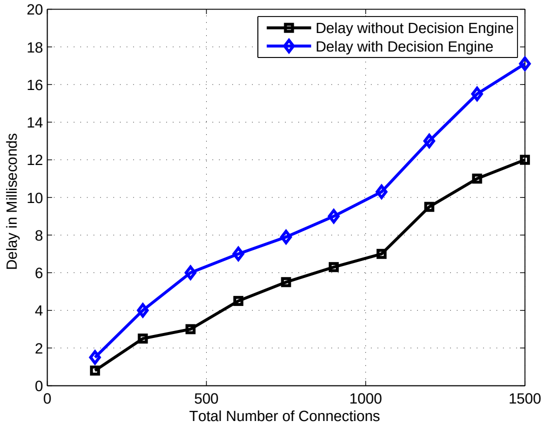


Fig. 10. Delay (Millisecond) Introduced by the System with and without Decision Engine.

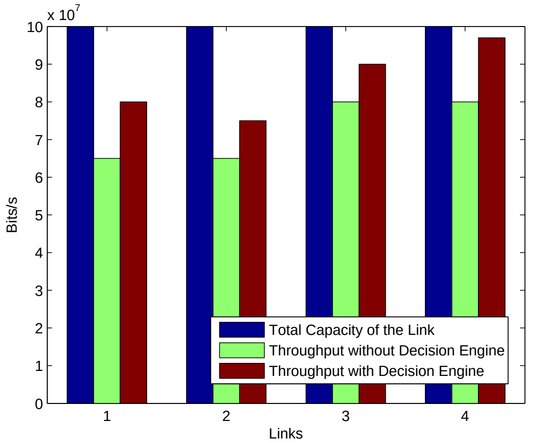


Fig. 11. Throughput of each Link with and without Decision Engine.

VI. Related Work

Currently there are growing number of research and proprietary efforts related to Multimedia Load Balancing focusing SIP [22], [23]. The core design and lower-level functionality are hidden because of commercial implications. Some vendors offer partial dynamicity with limited controls, while others are enforcing static decisions/rules [24]. A dynamic framework for load balancing in multi-homing scenario is presented by taking into account multiple criteria involving the service, control and network issues all together. MCDM theory is used to address the issue of dynamic variations and configuration from different planes with different set of objectives. This

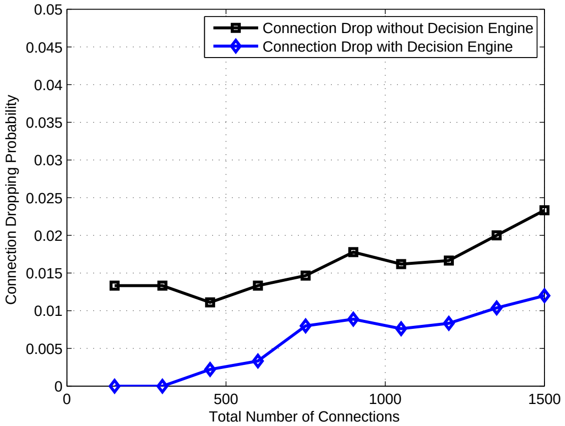


Fig. 12. Connection Dropping Probability with and without Decision Engine.

theory is used for access network technology (UMTS, GSM, WLAN etc) selection during the handoff based on user preferences [25]. A user priority scheme for admission control using Analytic Hierarchy Process (AHP) is proposed in [26]. Two MCDM methods namely AHP and TOPSIS are used in this framework for weight calculation and ranking of candidate link in multihomed network.

VII. Conclusion

QoS profile of the links, user authentication and authorization profiles, business objectives of the company and fluent dynamics over the multihomed platform constitutes a multidisciplinary problem. The information coming from different sources with different dimensions reflects the complexity of the underlying problem when a single decision has to be taken on the basis of multidimensional and multidisciplinary information. Conventional algorithms used for dynamic routing at higher layers in multihoming setups are either application oriented or are service dependent. Performance optimization is the ultimate goal in some cases while the others are technology specific. To address all these multi-facet goals in addition to the dynamics and fluctuations over the platform, MCDM methodology is required. A dynamic decision engine for SOCKS-based routing is presented. The system is capable of accommodating the fluent dynamics while handling a large set of attributes representing the underlying criteria in MCDM. Analytical Hierarchy Process (AHP) is used to calculate the weight of the corresponding attributes. These weight values are exploited in Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) to rank the alternatives (links). The system supports two decision enforcement modes. Decisions are computed on-the-fly in outsourcing mode while one of the pre-ranked links is chosen to route the request in provisioning mode (off-line). Existing standards and mechanisms are followed without involving overheads in the protocol stack. A test bed is developed to validate the solution. Throughput of the individual links improved significantly mentioning that the resources are being used efficiently and effectively at the cost of susceptible delay. Aggregated connection dropping probability has lower values than without the Decision Engine. Future work includes the interconnection of MCDM and conventional Policy-Based Network Management through the development of an automated linguistics in order to specify goals, criteria and alternatives.

References

[1] Y. Rekhter, T. Li, and S. Hares, “A Border Gateway Protocol 4 (BGP-4),” Jan. 2006.

[2] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, “SIP: Session Initiation Protocol,” RFC 3261, Jun. 2002.

[3] M. Ganis, Y. Lee, R. Kuris, D. Koblas, and L. Jones, “SOCKS Protocol Version 5,” RFC 1928, Mar. 1996.

[4] M. Brenner, “Diameter Policy Processing Application,” RFC 5224, Mar. 2008.

[5] “https://www.companymages.eu/ (12/01/2011, Last Visited (LV)).”

[6] “3GPP TS 29.207/209, Policy Control Over Gq interface,” Dec. 2004.

[7] P. Calhoun, J. Loughney, E. Guttman, G. Zorn, and J. Arkko, “Diameter Base Protocol,” RFC 3588, Sep. 2003.

[8] S. Mushtaq, O. Salem, C. Lohr, and A. Gravey, “Policy-based QoS Management for Multimedia Communication,” in *14th EUNICE Open European Conference*, 2008.

[9] S. A. Mushtaq, O. Salem, C. Lohr, and A. Gravey, “Distributed Call Admission Control in SIP Based Multimedia Communication,” in *NEM Summit 2008 : International Conference on Networked Electronic Media, october 13-15, Saint Malo, France*, 2008.

[10] “Dante,” http://www.inet.no/dante/ (15/12/2010, LV).

[11] D. Durham, J. Boyle, R. Cohen, S. Herzog, R. Rajan, and A. Sastry, “The COPS (Common Open Policy Service) Protocol,” RFC 2748, Jan. 2000.

[12] S. Boros, “Policy-Based Network Management With SNMP,” 2000.

[13] S. Önüt, S. S. Kara, and E. Işik, “Long term supplier selection using a combined fuzzy MCDM approach: A case study for a telecommunication company,” *Expert Syst. Appl.*, vol. 36, pp. 3887–3895, March 2009.

[14] P. P. Bonissone, R. Subbu, and J. Lizzi, “Multicriteria decision making (mcdm): a framework for research and applications,” *Comp. Intell. Mag.*, vol. 4, pp. 48–61, August 2009.

[15] E. Kornyshova and C. Salinesi, “MCDM Techniques Selection Approaches: State of the Art,” in *Computational Intelligence in Multicriteria Decision Making, IEEE Symposium on*, april 2007, pp. 22 –29.

[16] T. Saaty, *The Analytic Hierarchy Process, Planning, Piority Setting, Resource Allocation*. New york: McGraw-Hill, 1980.

[17] C. Hwang and K. Yoon, *Multiple attribut decision making : Methods and applications*. Springer-Verlag, 1981.

[18] R. Benayoun, B. Roy, and N. Sussmann, “Manual de reference du programme electre, Note de Sythese et Formation,” 1966.

[19] “JAVA SOCKS Server. http://jsocks.sourceforge.net/ (10/19/2010, (LV)).”

[20] “http://www.traffixsystems.com/ (05/12/2010, LV).”

[21] “http://bittwist.sourceforge.net/ (09/11/2010, LV).”

[22] H. Jiang, A. Iyengar, E. Nahum, W. Segmuller, A. Tantawi, and C. Wright, “Load Balancing for SIP Server Clusters,” in *INFOCOM 2009, IEEE*, april 2009, pp. 2286 –2294.

[23] “http://www.acmepacket.com/ (13/01/2011, LV).”

[24] “http://f5.com/products/ (15/01/2011, LV).”

[25] A. Sehgal and R. Agrawal, “QoS based network selection scheme for 4G systems,” *Consumer Electronics, IEEE Transactions on*, vol. 56, no. 2, pp. 560 –565, may 2010.

[26] H. Pervaiz, “A Multi-Criteria Decision Making (MCDM) network selection model providing enhanced QoS differentiation to customers,” in *Multimedia Computing and Information Technology (MCIT), 2010 International Conference on*, march 2010, pp. 49 –52.