LNCS 4268

Gerard Parr
David Malone
Mícheál Ó Foghlú (Eds.)

Autonomic Principles of IP Operations and Management

6th IEEE International Workshop on IP Operations and Management, IPOM 2006
Dublin, Ireland, October 2006, Proceedings

Springer

Gerard Parr David Malone
Mícheál Ó Foghlú (Eds.)

Autonomic Principles of IP Operations and Management

6th IEEE International Workshop
on IP Operations and Management, IPOM 2006
Dublin, Ireland, October 23-25, 2006
Proceedings

Springer

Lecture Notes in Computer Science 4268

Preface

This volume presents the proceedings of the *6^{th} IEEE International Workshop on IP Operations and Management (IPOM 2006)*, which was held as part of Manweek 2006 in Dublin, Ireland from October 23rd to 25th, 2006. In line with its reputation as one ofthe pre-eminent venues for the discussion and debate of advances of management of IP networks and services, the 2006 iteration of IPOM brought together an international audience of researchers and practitioners from both industry and academia. The overall theme of Manweek 2006 was "Autonomic Component and System Management",with IPOM taking this to be the application of autonomic principles to the IP operations, administration, maintenance and provisioning (OAM&P) domain.

IPOM 2006 is more relevant than ever to the emerging communications infrastructure that is increasingly focused on "convergence" of networks and services. Although arguably over-hyped, there is a fundamental truth to this convergence story, and this is based on the fact that the TCP/IP protocol suite (IPv4 and IPv6) has become the common denominator for a plethora of such converged services. One good example in the period between IPOM 2005 and IPOM 2006 has been the large scale deployment of consumer VoIP, linked to the success of Skype and alternatives including SIP-based approaches. In many countries VoIP is driving broadband deployment for SMEs where real costs savings can be accrued, especially for companies with remote staff in the field. Many operators are now deploying Quality of Service (QoS) schemes to manage this VoIP (and other premium) traffic. This brings these issues from the research laboratory into the operations and management domain.

Being a relatively pragmatic workshop IPOM 2006 is focused on issues that matter to those managing such IP networks and services, both enterprise networks and telecommunications operators' networks. These issues include the complexity of interoperability between networks and service providers, the performance versus costs in operating IP-based networks, and the OAM&P challenges in next generation networks(NGNs) and related seamless service provision. Of particular interest in the telecommunications sector are issues related to Fixed-Mobile Convergence and the emerging IP Multimedia System (IMS). These issues were reflected in the issued call for papers.

In response to the IPOM 2006 call for papers a total of 45 paper submissions were received from the research community. Of these, 39 were full papers and 6 were short papers. After a comprehensive review process carried out by the technical programme committee and additional subject area experts all submissions were ranked based on review scores and the co-chair's view on their contribution and relevance to the conference scope. All submissions received at least 3 reviews, with most receiving 4. After lengthy discussions it was decided to accept 18 of the 39 submitted full papers (40% acceptance rate of the total submissions) and 4 short papers. These papers present novel and interesting contributions in topics ranging from OSPF weightings in intradomain QoS, to large scale topology discovery. We believe that, taken together, these papers provide a provocative insight into the current state of the art in IP operations and management.

There are many people whose hard work and commitment were essential to the success of IPOM 2006. Foremost amongst these are the researchers who submitted papers to the conference. The overall quality of submissions this year was high and we regret that many high quality papers had to be rejected. We would like to express out gratitude to both the IPOM steering committee and the technical committee, for their advice and support through all the stages of the conference preparation. We thank all paper reviewers, in particular those outside the technical programme committee, for their uniformly thorough, fair and helpful reviews. We thank the IEEE for their continued support and sponsorship of IPOM.

Most of the time-consuming practical and logistical organisation tasks for the conference were handled by the members of the Manweek Organisation Committee – this made our jobs significantly easier, and for that we are very grateful. Finally, we wish to acknowledge the financial support of both Science Foundation Ireland and the Manweek corporate sponsors, whose contributions were hugely instrumental in helping us run what we hope was a stimulating, rewarding and, most importantly, an enjoyable conference for all its participants.

October 2006

Gerard Parr
David Malone
Mícheál Ó Foghlú

IPOM 2006 Organisation

Technical Programme Committee Co-chairs

Gerard Parr	University of Ulster, UK
David Malone	NUI Maynooth, Ireland
Mícheál Ó Foghlú	Waterford Institute of Technology, Ireland

Steering Committee

Tom Chen	Southern Methodist University, USA
Petre Dini	Cisco Systems, USA
Andrzej Jajszczyk	AGH University of Science and Technology, Poland
G.-S. Kuo	NCCU, Republic of China
Deep Medhi	University of Missouri-Kansas City, USA
Curtis Siller	IEEE ComSoc, USA

Organisation Co-chairs

Brendan Jennings	Waterford Institute of Technology, Ireland
Sven van der Meer	Waterford Institute of Technology, Ireland

Publication Chair

Tom Pfeifer	Waterford Institute of Technology, Ireland

Publicity Co-chairs

Sasitharan Balasubramaniam	Waterford Institute of Technology, Ireland
John Murphy	University College Dublin, Ireland

Treasurer

Mícheál Ó Foghlú	Waterford Institute of Technology, Ireland

Local Arrangements

Miguel Ponce de León	Waterford Institute of Technology, Ireland
Dave Lewis	Trinity College Dublin, Ireland
Dirk Pesch	Cork Institute of Technology, Ireland

Gabriel-Miro Muntean	Dublin City University, Ireland
Seán Murphy	University College Dublin, Ireland
Rob Brennan	Ericsson, Ireland

Manweek 2006 General Co-chairs

William Donnelly	Waterford Institute of Technology, Ireland
John Strassner	Motorola Labs, USA

Manweek 2006 Advisors

Raouf Boutaba	University of Waterloo, Canada
Joan Serrat	Universitat Politècnica de Catalunya, Spain

IPOM 2006 Technical Programme Committee

Nader Azarmi	BT Group Research, UK
John-Luc Bakker	Telcordia, USA
Saleem Bhatti	University of St Andrews, UK
Marcus Brunner	NEC Europe, Germany
Baek-Young Choi	University of Missouri-Kansas City, USA
Alexander Clemm	Cisco Systems, USA
Haitham Cruickshank	University of Surrey, UK
Laurie Cuthbert	Queen Mary University of London, UK
Timothy Gonsalves	IIT Madras, India
Abdelhakim Hafid	University of Montreal, Canada
Steve Hailes	University College London, UK
Masum Hasan	Cisco Systems, USA
David Hutchison	Lancaster University, UK
Wolfgang Kellerer	DoCoMo Eurolabs, Germany
G.S. Kuo	NCCU, Republic of China
Edmundo Madeira	UNICAMP, Brazil
Thomas Magedanz	Fraunhofer FOKUS, Germany
Manu Malek	Stevens Institute of Technology, USA
Dave Maltz	Microsoft, USA
Deep Medhi	University of Missouri-Kansas City, USA
Maurizio Molina	DANTE, UK
Donal O'Mahony	Trinity College Dublin, Ireland
Michal Pioro	Warsaw University of Technology, Poland
Caterina Scoglio	Kansas State University, USA
Bryan Scotney	University of Ulster, UK
Stephan Steglich	Fraunhofer FOKUS, Germany
Martin Stiemerling	NEC, Germany

John Strassner	Motorola Labs, USA
Vincent Wade	Trinity College Dublin, Ireland

IPOM 2006 Additional Paper Reviewers

Sasitharan Balasubramaniam	Waterford Institute of Technology, Ireland
Keara Barrett	Waterford Institute of Technology, Ireland
Prakash Bettadapur	Cisco Systems, USA
Dmitri Botvich	TSSG, Ireland
Ray Carroll	Waterford Institute of Technology, Ireland
Peter Clifford	NUI Maynooth, Ireland
Steven Davy	Waterford Institute of Technology, Ireland
Petre Dini	Cisco, USA
William Fitzgerald	Waterford Institute of Technology, Ireland
Paulo Freitas	UFSC, Brazil
Luciano Gaspary	UFRGS, Brazil
Celio Guimaraes	UNICAMP, Brazil
Paul Malone	Waterford Institute of Technology, Ireland
Joberto Martins	UNIFACS - Universidade Salvador, Brazil
Jimmy McGibney	Waterford Institute of Technology, Ireland
Sven van der Meer	Waterford Institute of Technology, Ireland
Niall Murphy	Amazon.COM Network Engineering, Ireland
Seán Murphy	University College Dublin, Ireland
Venet Osmani	Waterford Institute of Technology, Ireland
Tom Pfeifer	TSSG, Ireland
Miguel Ponce de Leon	TSSG, Ireland
John Ronan	Waterford Institute of Technology, Ireland
José Augusto Suruagy Monteiro	UNIFACS, Brazil
Fábio Verdi	Unicamp, Brazil
Rolf Winter	NEC Europe, Germany

Lecture Notes in Computer Science

For information about Vols. 1–4180

please contact your bookseller or Springer

Vol. 4272: P. Havinga, M. Lijding, N. Meratnia, M. Wegdam (Eds.), Smart Sensing and Context. XI, 267 pages. 2006.

Vol. 4270: H. Zha, Z. Pan, H. Thwaites, A.C. Addison, M. Forte (Eds.), Interactive Technologies and Sociotechnical Systems. XVI, 547 pages. 2006.

Vol. 4269: R. State, S. van der Meer, D. O'Sullivan, T. Pfeifer (Eds.), Large Scale Management of Distributed Systems. XIII, 282 pages. 2006.

Vol. 4268: G. Parr, D. Malone, M. Ó Foghlú (Eds.), Autonomic Principles of IP Operations and Management. XIII, 237 pages. 2006.

Vol. 4267: A. Helmy, B. Jennings, L. Murphy, T. Pfeifer (Eds.), Autonomic Management of Mobile Multimedia Services. XIII, 257 pages. 2006.

Vol. 4265: N. Lavrač, L. Todorovski, K.P. Jantke (Eds.), Discovery Science. XIV, 384 pages. 2006. (Sublibrary LNAI).

Vol. 4264: J.L. Balcázar, P.M. Long, F. Stephan (Eds.), Algorithmic Learning Theory. XIII, 393 pages. 2006. (Sublibrary LNAI).

Vol. 4263: A. Levi, E. Savas, H. Yenigün, S. Balcisoy, Y. Saygin (Eds.), Computer and Information Sciences – ISCIS 2006. XXIII, 1084 pages. 2006.

Vol. 4260: Z. Liu, J. He (Eds.), Formal Methods and Software Engineering. XII, 778 pages. 2006.

Vol. 4257: I. Richardson, P. Runeson, R. Messnarz (Eds.), Software Process Improvement. XI, 219 pages. 2006.

Vol. 4254: T. Grust, H. Höpfner, A. Illarramendi, S. Jablonski, M. Mesiti, S. Müller, P.-L. Patranjan, K.-U. Sattler, M. Spiliopoulou (Eds.), Current Trends in Database Technology – EDBT 2006. XXXI, 932 pages. 2006.

Vol. 4253: B. Gabrys, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part III. XXXII, 1301 pages. 2006. (Sublibrary LNAI).

Vol. 4252: B. Gabrys, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part II. XXXIII, 1335 pages. 2006. (Sublibrary LNAI).

Vol. 4251: B. Gabrys, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part I. LXVI, 1297 pages. 2006. (Sublibrary LNAI).

Vol. 4249: L. Goubin, M. Matsui (Eds.), Cryptographic Hardware and Embedded Systems - CHES 2006. XII, 462 pages. 2006.

Vol. 4248: S. Staab, V. Svátek (Eds.), Engineering Knowledge in the Age of the Semantic Web. XIV, 400 pages. 2006. (Sublibrary LNAI).

Vol. 4247: T.-D. Wang, X. Li, S.-H. Chen, X. Wang, H. Abbass, H. Iba, G. Chen, X. Yao (Eds.), Simulated Evolution and Learning. XXI, 940 pages. 2006.

Vol. 4245: A. Kuba, L.G. Nyúl, K. Palágyi (Eds.), Discrete Geometry for Computer Imagery. XIII, 688 pages. 2006.

Vol. 4243: T. Yakhno, E.J. Neuhold (Eds.), Advances in Information Systems. XIII, 420 pages. 2006.

Vol. 4241: R.R. Beichel, M. Sonka (Eds.), Computer Vision Approaches to Medical Image Analysis. XI, 262 pages. 2006.

Vol. 4239: H.Y. Youn, M. Kim, H. Morikawa (Eds.), Ubiquitous Computing Systems. XVI, 548 pages. 2006.

Vol. 4238: Y.-T. Kim, M. Takano (Eds.), Management of Convergence Networks and Services. XVIII, 605 pages. 2006.

Vol. 4237: H. Leitold, E. Markatos (Eds.), Communications and Multimedia Security. XII, 253 pages. 2006.

Vol. 4236: L. Breveglieri, I. Koren, D. Naccache, J.-P. Seifert (Eds.), Fault Diagnosis and Tolerance in Cryptography. XIII, 253 pages. 2006.

Vol. 4234: I. King, J. Wang, L. Chan, D. Wang (Eds.), Neural Information Processing, Part III. XXII, 1227 pages. 2006.

Vol. 4233: I. King, J. Wang, L. Chan, D. Wang (Eds.), Neural Information Processing, Part II. XXII, 1203 pages. 2006.

Vol. 4232: I. King, J. Wang, L. Chan, D. Wang (Eds.), Neural Information Processing, Part I. XLVI, 1153 pages. 2006.

Vol. 4229: E. Najm, J.F. Pradat-Peyre, V.V. Donzeau-Gouge (Eds.), Formal Techniques for Networked and Distributed Systems - FORTE 2006. X, 486 pages. 2006.

Vol. 4228: D.E. Lightfoot, C.A. Szyperski (Eds.), Modular Programming Languages. X, 415 pages. 2006.

Vol. 4227: W. Nejdl, K. Tochtermann (Eds.), Innovative Approaches for Learning and Knowledge Sharing. XVII, 721 pages. 2006.

Vol. 4225: J.F. Martínez-Trinidad, J.A. Carrasco Ochoa, J. Kittler (Eds.), Progress in Pattern Recognition, Image Analysis and Applications. XIX, 995 pages. 2006.

Vol. 4224: E. Corchado, H. Yin, V. Botti, C. Fyfe (Eds.), Intelligent Data Engineering and Automated Learning – IDEAL 2006. XXVII, 1447 pages. 2006.

Vol. 4223: L. Wang, L. Jiao, G. Shi, X. Li, J. Liu (Eds.), Fuzzy Systems and Knowledge Discovery. XXVIII, 1335 pages. 2006. (Sublibrary LNAI).

Vol. 4222: L. Jiao, L. Wang, X. Gao, J. Liu, F. Wu (Eds.), Advances in Natural Computation, Part II. XLII, 998 pages. 2006.

Vol. 4221: L. Jiao, L. Wang, X. Gao, J. Liu, F. Wu (Eds.), Advances in Natural Computation, Part I. XLI, 992 pages. 2006.

Vol. 4219: D. Zamboni, C. Kruegel (Eds.), Recent Advances in Intrusion Detection. XII, 331 pages. 2006.

Vol. 4218: S. Graf, W. Zhang (Eds.), Automated Technology for Verification and Analysis. XIV, 540 pages. 2006.

Vol. 4217: P. Cuenca, L. Orozco-Barbosa (Eds.), Personal Wireless Communications. XV, 532 pages. 2006.

Vol. 4216: M.R. Berthold, R. Glen, I. Fischer (Eds.), Computational Life Sciences II. XIII, 269 pages. 2006. (Sublibrary LNBI).

Vol. 4215: D.W. Embley, A. Olivé, S. Ram (Eds.), Conceptual Modeling - ER 2006. XVI, 590 pages. 2006.

Vol. 4213: J. Fürnkranz, T. Scheffer, M. Spiliopoulou (Eds.), Knowledge Discovery in Databases: PKDD 2006. XXII, 660 pages. 2006. (Sublibrary LNAI).

Vol. 4212: J. Fürnkranz, T. Scheffer, M. Spiliopoulou (Eds.), Machine Learning: ECML 2006. XXIII, 851 pages. 2006. (Sublibrary LNAI).

Vol. 4211: P. Vogt, Y. Sugita, E. Tuci, C. Nehaniv (Eds.), Symbol Grounding and Beyond. VIII, 237 pages. 2006. (Sublibrary LNAI).

Vol. 4210: C. Priami (Ed.), Computational Methods in Systems Biology. X, 323 pages. 2006. (Sublibrary LNBI).

Vol. 4209: F. Crestani, P. Ferragina, M. Sanderson (Eds.), String Processing and Information Retrieval. XIV, 367 pages. 2006.

Vol. 4208: M. Gerndt, D. Kranzlmüller (Eds.), High Performance Computing and Communications. XXII, 938 pages. 2006.

Vol. 4207: Z. Ésik (Ed.), Computer Science Logic. XII, 627 pages. 2006.

Vol. 4206: P. Dourish, A. Friday (Eds.), UbiComp 2006: Ubiquitous Computing. XIX, 526 pages. 2006.

Vol. 4205: G. Bourque, N. El-Mabrouk (Eds.), Comparative Genomics. X, 231 pages. 2006. (Sublibrary LNBI).

Vol. 4204: F. Benhamou (Ed.), Principles and Practice of Constraint Programming - CP 2006. XVIII, 774 pages. 2006.

Vol. 4203: F. Esposito, Z.W. Raś, D. Malerba, G. Semeraro (Eds.), Foundations of Intelligent Systems. XVIII, 767 pages. 2006. (Sublibrary LNAI).

Vol. 4202: E. Asarin, P. Bouyer (Eds.), Formal Modeling and Analysis of Timed Systems. XI, 369 pages. 2006.

Vol. 4201: Y. Sakakibara, S. Kobayashi, K. Sato, T. Nishino, E. Tomita (Eds.), Grammatical Inference: Algorithms and Applications. XII, 359 pages. 2006. (Sublibrary LNAI).

Vol. 4200: I.F.C. Smith (Ed.), Intelligent Computing in Engineering and Architecture. XIII, 692 pages. 2006. (Sublibrary LNAI).

Vol. 4199: O. Nierstrasz, J. Whittle, D. Harel, G. Reggio (Eds.), Model Driven Engineering Languages and Systems. XVI, 798 pages. 2006.

Vol. 4198: O. Nasraoui, O. Zaiane, M. Spiliopoulou, B. Mobasher, B. Masand, P. Yu (Eds.), Advances in Web Minding and Web Usage Analysis. IX, 177 pages. 2006. (Sublibrary LNAI).

Vol. 4197: M. Raubal, H.J. Miller, A.U. Frank, M.F. Goodchild (Eds.), Geographic, Information Science. XIII, 419 pages. 2006.

Vol. 4196: K. Fischer, I.J. Timm, E. André, N. Zhong (Eds.), Multiagent System Technologies. X, 185 pages. 2006. (Sublibrary LNAI).

Vol. 4195: D. Gaiti, G. Pujolle, E. Al-Shaer, K. Calvert, S. Dobson, G. Leduc, O. Martikainen (Eds.), Autonomic Networking. IX, 316 pages. 2006.

Vol. 4194: V.G. Ganzha, E.W. Mayr, E.V. Vorozhtsov (Eds.), Computer Algebra in Scientific Computing. XI, 313 pages. 2006.

Vol. 4193: T.P. Runarsson, H.-G. Beyer, E. Burke, J.J. Merelo-Guervós, L.D. Whitley, X. Yao (Eds.), Parallel Problem Solving from Nature - PPSN IX. XIX, 1061 pages. 2006.

Vol. 4192: B. Mohr, J.L. Träff, J. Worringen, J. Dongarra (Eds.), Recent Advances in Parallel Virtual Machine and Message Passing Interface. XVI, 414 pages. 2006.

Vol. 4191: R. Larsen, M. Nielsen, J. Sporring (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2006, Part II. XXXVIII, 981 pages. 2006.

Vol. 4190: R. Larsen, M. Nielsen, J. Sporring (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2006, Part I. XXXVVIII, 949 pages. 2006.

Vol. 4189: D. Gollmann, J. Meier, A. Sabelfeld (Eds.), Computer Security – ESORICS 2006. XI, 548 pages. 2006.

Vol. 4188: P. Sojka, I. Kopeček, K. Pala (Eds.), Text, Speech and Dialogue. XV, 721 pages. 2006. (Sublibrary LNAI).

Vol. 4187: J.J. Alferes, J. Bailey, W. May, U. Schwertel (Eds.), Principles and Practice of Semantic Web Reasoning. XI, 277 pages. 2006.

Vol. 4186: C. Jesshope, C. Egan (Eds.), Advances in Computer Systems Architecture. XIV, 605 pages. 2006.

Vol. 4185: R. Mizoguchi, Z. Shi, F. Giunchiglia (Eds.), The Semantic Web – ASWC 2006. XX, 778 pages. 2006.

Vol. 4184: M. Bravetti, M. Núñez, G. Zavattaro (Eds.), Web Services and Formal Methods. X, 289 pages. 2006.

Vol. 4183: J. Euzenat, J. Domingue (Eds.), Artificial Intelligence: Methodology, Systems, and Applications. XIII, 291 pages. 2006. (Sublibrary LNAI).

Vol. 4182: H.T. Ng, M.-K. Leong, M.-Y. Kan, D. Ji (Eds.), Information Retrieval Technology. XVI, 684 pages. 2006.

Table of Contents

1. Modeling and Planning

Traffic Modeling and Classification Using Packet Train Length and Packet Train Size 1
Dinil Mon Divakaran, Hema A. Murthy, Timothy A. Gonsalves

Adaptive Bandwidth Allocation Method for Long Range Dependence Traffic 13
Bong Joo Kim, Gang Uk Hwang

Algorithms for Fast Resilience Analysis in IP Networks 25
Michael Menth, Jens Milbrandt, Frank Lehrieder

2. Quality of Service Routing

Efficient OSPF Weight Allocation for Intra-domain QoS Optimization . . . 37
Pedro Sousa, Miguel Rocha, Miguel Rio, Paulo Cortez

Probabilistic QoS Guarantees with FP/EDF Scheduling and Packet Discard in a Real Time Context: A Comparative Study of Local Deadline Assignment Techniques 49
Fadhel Karim Maïna, Leila Azouz Saïdane

A Quantitative QoS Routing Model for Diffserv Aware MPLS Networks 61
Haci A. Mantar

3. Quality of Service Issues

Experience-Based Admission Control with Type-Specific Overbooking . . . 72
Jens Milbrandt, Michael Menth, Jan Junker

Applying Blood Glucose Homeostatic Model Towards Self-management of IP QoS Provisioned Networks 84
Sasitharan Balasubramaniam, Dmitri Botvich, William Donnelly, Nazim Agoulmine

New Mathematical Models for Token Bucket Based Meter/Markers 96
Rafal Stankiewicz, Andrzej Jajszczyk

4. Management and Configuration

Unique Subnet Auto-configuration in IPv6 Networks 108
Reha Oguz Altug, Cuneyt Akinlar

An Efficient Process for Estimation of Network Demand for QoS-Aware IP Network Planning 120
Alan Davy, Dmitri Botvich, Brendan Jennings

A Protocol for Atomic Deployment of Management Policies in QoS-Enabled Networks 132
Rodrigo Sanger Alves, Lisandro Zambenedetti Granville, Maria Janilce Bosquiroli Almeida, Liane Margarida Rockenbach Tarouco

5. Autonomics and Security

Towards Autonomic Network Management for Mobile IPv4 Based Wireless Networks 144
Dong-Hee Kwon, Woo-Jae Kim, Young-Joo Suh, James W. Hong

A Comparison of Mobile Agent and SNMP Message Passing for Network Security Management Using Event Cases 156
Ching-hang Fong, Gerard Parr, Philip Morrow

Principles of Secure Network Configuration: Towards a Formal Basis for Self-configuration 168
Simon N. Foley, William Fitzgerald, Stefano Bistarelli, Barry O'Sullivan, Mícheál Ó Foghlú

6. Topology

Risk Assessment of End-to-End Disconnection in IP Networks Due to Network Failures 181
Jens Milbrandt, Ruediger Martin, Michael Menth, Florian Hoehn

Evaluation of a Large-Scale Topology Discovery Algorithm 193
Benoit Donnet, Bradley Huffaker, Timur Friedman, kc claffy

The Virtual Topology Service: A Mechanism for QoS-Enabled Interdomain Routing 205
Fábio Verdi, Maurício Magalhães, Edmundo Madeira, Annikki Welin

7. Short Papers

Correlating User Perception and Measurable Network Properties: Experimenting with QoE 218
Pál Varga, Gergely Kún, Gábor Sey, István Moldován, Péter Gelencsér

Towards Realization of Web Services-Based TSA from NGOSS TNA 222
Mi-Jung Choi, Hong-Taek Ju, James W.K. Hong, Dong-Sik Yun

An Efficient Queue Management (EQM) Technique for Networks........ 228
Debessay Fesehaye Kassa

Monitoring MIPv6 Traffic with IPFIX 232
Youngseok Lee, Soonbyoung Choi, Jaehwa Lee

Author Index 237

Traffic Modeling and Classification Using Packet Train Length and Packet Train Size

Dinil Mon Divakaran, Hema A. Murthy, and Timothy A. Gonsalves

Department of Computer Science and Engineering
Indian Institute of Technology, Madras
Chennai - 600036
{dinil, hema, tag}@tenet.res.in

Abstract. Traffic modeling and classification finds importance in many areas such as bandwidth management, traffic analysis, traffic prediction, network planning, Quality of Service provisioning and anomalous traffic detection. Network traffic exhibits some statistically invariant properties. Earlier works show that it is possible to identify traffic based on its statistical characteristics. In this paper, an attempt is made to identify the statistically invariant properties of different traffic classes using multiple parameters, namely packet train length and packet train size. Models generated using these parameters are found to be highly accurate in classifying different traffic classes. The parameters are also useful in revealing different classes of services within different traffic classes.

1 Introduction

The phenomenal expansion of Internet has seen a rapid growth in the number and variety of applications. Many of such applications turn out to be bandwidth-hungry or delay-sensitive; and require, or at least benefit from specific service classes that prioritize packets in the Internet. Internet Service Providers and network operators need to classify traffic data within their network and evaluate their absolute and relative importance and subsequently create traffic policies to be enforced in the Internet routers [1]. Network administrators need to know the different classes or types of traffic that flow through their network so as to manage the network efficiently. Accurate traffic classification is essential for provisioning and bandwidth management. Floyd and Paxson [2] pointed out that it is important to capture the invariants of traffic to cope with the constantly changing nature of Internet traffic.

Conventional methods for traffic classification use the packet header information to find out the ports used for communication. Well known ports are supposed to be used by specific application protocols (e.g. port 80 is usually used by HTTP). But this method has become less and less accurate as more and more emerging applications use well know ports for relaying traffic, e.g. tunneling over HTTP port is very common [3,4]. Owing to the increasing traffic and application protocols, techniques based on statistical modeling are gaining importance. Recent works have made efforts to increase the accuracy in traffic classification [5,6,7,8].

G. Parr, D. Malone, and M. Ó Foghlú (Eds.): IPOM 2006, LNCS 4268, pp. 1–12, 2006.

In traffic modeling, parameters are extracted from packet headers. There are different parameters such as packet length, packet inter-arrival time, flow duration, packet train inter-arrival time, packet train length, packet train size etc that can be considered for modeling of traffic. While parameters can be modeled separately [9], we focus on modeling the traffic classes using multiple parameters.

In [10], it was shown that traffic characteristics are generally multimodal in nature. For example, the total number of packets transferred during mail transfer vary for small text messages to large picture attachments. To capture the multimodal characteristic of traffic, we employ clustering techniques based on Vector Quantization (VQ) [11] and Gaussian Mixture Models (GMM) [12]. The models obtained using these techniques are later used for classification of a given data set (not used during training) into one of the different traffic types.

The rest of the paper is organized as follows. In Sect.3 we explain the experimental setup. Modeling and classification using VQ is explained in Sect.4. Section 5 details modeling using GMM and Bayesian classification using the Gaussian mixtures obtained. Evaluation and verification of models follow in Sect.6. We conclude in Sect.7.

2 Related Work

System administrators have long being using port based classification method to identify different traffic classes flowing in the network. Tools such as tcpdump [13] read the header information to find the source and destination ports. Each server port associates itself with an application as per the IANA (Internet Assigned Numbers Authority) [14], which maintains a mapping of the server ports to application types. Such a method does not provide good accuracy for a number of reasons. For instance, with the proliferation of applications, not every application is registered with IANA. Users behind a firewall that permits packets to only a few ports, usually relay traffic through well known ports (eg. SSH over HTTP). Similarly, non-privileged users run HTTP servers on ports other than 80. Even for applications defined with IANA, some ports are used by different applications with entirely different QoS requirements (for example, SSH and SCP use same port 22).

Statistical traffic classification is an alternative to the less accurate port based classification. Past research works have focused on characterizing particular traffic classes. In [15], joint distribution of flow duration and number of bytes were used to identify DNS traffic. Paxson [16] examined the distribution of flow bytes and packets for a number of different applications. Roughan et al. used LDA (Linear Discriminant Analysis) and QDA (Qualitative Discriminant Analysis) to classify traffic into different classes of services for QoS implementation [5]. In [6], authors describe a method for visualisation of the attribute statistics that aids in recognizing cluster types. The method uses EM (Expectation-Maximistaion) for probabilistic clustering with parameters such as packet size and packet inter-arrival time. In [8], the authors explore Bayesian classifier using a number of per flow discriminators to classify network traffic.

The multimodal nature of traffic was highlighted in [9] using packet size as a promising parameter for traffic characterisation. We extend this work and use flow related information for modeling traffic.

3 Traffic Modeling

In this section, we detail the setup for experimentation in terms of the parameters used for identifying the various traffic classes considered for this work.

3.1 Parameters

The goodness of these models largely depends on the parameters that are used for modeling. The parameters used here for modeling and classification of traffic classes are *packet train length* and *packet train size.* The concept of packet train was introduced by Raj Jain and Shawn A. Routhier [17]. The two ends of a packet train are identified as two nodes in a network. As defined in [17], a packet train is essentially the flow of packets between two nodes in a network, where each packet forms the car of the train. Here, we modify the definition of a packet train to be the flow between two sockets, which is appropriately identified by the quadruple *(source host, source port, destination host, destination port).* Each end of such a packet train is identified by the node name and port number. Packet train length is then defined as the number of packets within a train; and packet train size is the sum of the sizes of all the packets that form a packet train. The advantage of using these parameters (as will be discussed later), is that the models generated using these parameters give information regarding the class of service within an application type (say HTTP), apart from classifying traffic.

Since we consider two parameters for modeling and testing, it is important to ensure that one parameter doesn't overshadow the other parameter. Appropriate normalization of parameters is therefore performed.

3.2 Traffic Classes

Five commonly used application protocols are selected for modeling and testing. These are HTTP, SMTP, DNS, SSH and POP3 which use ports 80, 25, 53, 22 and 110 respectively. DNS uses UDP, whereas HTTP, SMTP, SSH and POP3 use TCP as the transport layer protocol. The terms *traffic types* and *traffic classes* refer to these application protocols in general.

This work looks only at one UDP based traffic class, namely DNS traffic. Since all other traffic classes considered are TCP based, and hence connection oriented, they inherently have the packet train property. But, it should be noted that almost all UDP based applications can be viewed as a connection oriented traffic and therefore can be represented using packet trains. For example, a video conferencing tool running on top of UDP can be identified uniquely by *src host*, *src port*, *dst host* and *dst port*, and all such packets can be considered as part of a connection. To distinguish different connections of the same applications, the time between packets can be used; as the time between consecutive packets of one run of the application will be much less as compared to time between consecutive packets of different runs of the application. Hence, using all these information, packet train parameters can be extracted for the traffic generated by a video conferencing application, or in general, for traffic generated by almost any UDP based application.

3.3 Data Representation

Throughout the paper, data or packet trains are represented as a set of N vectors, $\boldsymbol{X} = \{\mathbf{x_1}, \mathbf{x_2}, ..., \mathbf{x_N}\}$ [1]. Each input vector has a dimension of two corresponding to the two parameters used in modeling.

4 Classification Using Vector Quantization

VQ is a very popular approximate method in the class of clustering algorithms that simplifies computations and accelerates convergence. VQ partitions d-dimensional vectors in the vector space, $\mathbf{R}^d$, into finite sets of vectors based on the *nearest-neighbour* criterion [18]. Such sets, called clusters, represent separate regions in the vector space. A vector, $\mathbf{x} \in \mathbf{R}^d$, belongs to cluster C_i, if

$$\|\mathbf{x} - \boldsymbol{\mu}_\mathbf{i}\| < \|\mathbf{x} - \boldsymbol{\mu}_\mathbf{j}\| \quad \text{for all } j \neq i \ . \tag{1}$$

where $\boldsymbol{\mu}_\mathbf{i}$ is the mean vector of the cluster, C_i. This equation states that a vector belongs to the nearest cluster. If there are two or more clusters to which the distance from the vector is minimum, one among them is chosen randomly. The clusters partition the vector space such that

$$\bigcup_{i=1}^{k} C_i = \mathbf{R}^d \qquad \textit{and} \qquad \bigcap_{i=1}^{k} C_i = \phi \ . \tag{2}$$

where k is the number of clusters.

4.1 Training

During training, we use VQ to partition the vector space, where the first dimension of each vector is *packet train length* and second dimension is *packet train size.* The algorithm used is as follows

1. Initialize the mean (vector) of each cluster by randomly selecting a vector from the given set of vectors, $\mathbf{X}$, such that no two clusters have the same mean.
2. Until the mean of each cluster converges
   - Classify each vector into one of the clusters using (1).
   - Recompute the mean of each cluster.

Using the above algorithm, the vectors in the given data set, $\mathbf{X}$, are classified into clusters. The models thus generated for each traffic type is used for testing.

[1] Boldface is used to denote vectors and matrices.