LNCS 3751

Thomas Magedanz
Edmundo R.M. Madeira
Petre Dini (Eds.)

Operations and Management in IP-Based Networks

5th IEEE International Workshop on IP Operations and Management, IPOM 2005
Barcelona, Spain, October 2005, Proceedings

Springer

Thomas Magedanz Edmundo R.M. Madeira
Petre Dini (Eds.)

Operations and Management in IP-Based Networks

5th IEEE International Workshop on
IP Operations and Management, IPOM 2005
Barcelona, Spain, October 26-28, 2005
Proceedings

Springer

Volume Editors

Thomas Magedanz
Technical University of Berlin/Fraunhofer Institute FOKUS
Kaiserin-Augusta-Allee 31, 10589 Berlin, Germany
E-mail: magedanz@fokus.fraunhofer.de

Edmundo R.M. Madeira
UNICAMP – University of Campinas, IC - Institute of Computing
13083-970 Campinas, SP, Brazil
E-mail: edmundo@ic.unicamp.br

Petre Dini
Cisco Systems, Inc.
10 West Tasman Drive, Bldg O/2, San Jose, CA 95134, USA
E-mail: pdini@cisco.com

Library of Congress Control Number: 2005933267

CR Subject Classification (1998): C.2, D.4.4, D.2, H.3.5, H.4, K.6.4

ISSN 0302-9743
ISBN-10 3-540-29356-6 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-29356-9 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springeronline.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Olgun Computergrafik
Printed on acid-free paper SPIN: 11567486 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 3751

Preface

The beginning of the 21st century is witnessing a drive to the convergence of fixed and mobile telecommunication networks and the increasing adoption of IP technologies for implementing seamless multimedia applications in next-generation networks. The IEEE International Workshop Series on IP Operations & Management (IPOM) is documenting this evolution by providing snapshots of the state of the art in the field of operations and management in IP-based networks.

The 5th IEEE International Workshop on IP Operations & Management (IPOM 2005), devoted to the "O&M Challenges in Next Generation Services and Networks", was held in Barcelona, Spain, October 26–28, 2005. Here IPOM was one of the five collocated events under the banner "First International Week on Management on Networks and Services (www.manweek2005.org)", together with the 16th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management (DSOM 2005), the 8th International Conference on Management of Multimedia Networks and Services (MMNS 2005), the 2005 Symposium on Self-stabilizing Systems (SSS 2005) and the 1st IEEE/IFIP International Workshop on Autonomic Grid Networking and Management (AGNM 2005).

This book contains the official proceedings of IPOM 2005. It features 21 high-quality papers grouped into seven technical sessions looking at O&M for VoIP, IMS and managed IP services, management of open interfaces, QoS and pricing in NGNs, autonomic communications, policy-based management, routing and topologies, routing and tools, as well as experiences from testbeds and trials. Additional papers presented in two short sessions are published separately.

We would like to thank the authors for all their efforts, as well as the members of the Technical Program Committee, and the reviewers. Without their support the high-quality program of this event would not have been possible. We are also indebted to many individuals and organizations that made the conference possible (IEEE, IARIA, Fraunhofer Institute FOKUS, JEMS's drivers, and the Universitat Politècnica de Catalunya).

October 2005

Thomas Magedanz
Edmundo R.M. Madeira
Petre Dini

Organization

General Chair

Petre Dini, Cisco Systems, USA

Program Co-chairs

Thomas Magedanz, TU Berlin / Fraunhofer FOKUS, Germany
Edmundo Madeira, Unicamp, Brazil

Technical Program Committee

Alexander Clemm, Cisco Systems, USA
Andrzej Jajszczyk, AGH University of Science and Technology, Poland
Antonio Pescapé, University of Napoli "Federico II", Italy
Carlos Becker Westphall, UFSC, Brazil
Deep Medhi, University of Missouri-Kansas City, USA
G.-S. Kuo, NCCU, Republic of China
Gerard Parr, University of Ulster, UK
Iakovos Venieris, NTUA, Greece
Iraj Saniee, Bell Labs, USA
Joerg Heuer, T-Labs, Germany
Johan Zuidweg, Telefonica, Spain
John-Luc Bakker, Telcordia, USA
Karthikeyan Ramasamy, Juniper, USA
Manu Malek, Stevens Institute of Technology, USA
Marcus Brunner, NEC Europe, Germany
Mario Baldi, Politecnico di Torino, Italy
Martin Stiemerling, NEC, Germany
Masum Hasan, Cisco Systems, USA
Meng Luoming, BUPT, P.R. China
Michal Pioro, Warsaw University of Technology, Poland
Nail Akar, Bilkent University, Turkey
Peter Domschitz, Alcatel, Germany
Richard Schaedler, Tekelec, USA
Roberto Minerva, Telecom Italy Labs, Italy
Sascha Karlich, Siemens, Austria
Stamatios Kartalopoulos, University of Oklahoma, USA
Tassos Gavras, EURESCOM, Germany
Tom Chen, SMU, USA
William Donnelly, Waterford Institute of Technology, Ireland
Wolfgang Kellerer, DoCoMo Eurolabs, Germany
Wulf Bauerfeld, T-Systems, Germany

Panel Chair

Stephan Steglich, FOKUS, Germany

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

Lecture Notes in Computer Science

For information about Vols. 1–3648

please contact your bookseller or Springer

Vol. 3751: T. Magedanz, E.R. M. Madeira, P. Dini (Eds.), IP Operations and Management. X, 213 pages. 2005.

Vol. 3750: J. Duncan, G. Gerig (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2005, Part II. XL, 1018 pages. 2005.

Vol. 3749: J. Duncan, G. Gerig (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2005, Part I. XXXIX, 942 pages. 2005.

Vol. 3739: W. Fan, Z. Wu, J. Yang (Eds.), Advances in Web-Age Information Management. XXII, 930 pages. 2005.

Vol. 3738: V.R. Syrotiuk, E. Chávez (Eds.), Ad-Hoc, Mobile, and Wireless Networks. XI, 360 pages. 2005.

Vol. 3735: A. Hoffmann, H. Motoda, T. Scheffer (Eds.), Discovery Science. XVI, 400 pages. 2005. (Subseries LNAI).

Vol. 3734: S. Jain, H.U. Simon, E. Tomita (Eds.), Algorithmic Learning Theory. XII, 490 pages. 2005. (Subseries LNAI).

Vol. 3731: F. Wang (Ed.), Formal Techniques for Networked and Distributed Systems - FORTE 2005. XII, 558 pages. 2005.

Vol. 3728: V. Paliouras, J. Vounckx, D. Verkest (Eds.), Integrated Circuit and System Design. XV, 753 pages. 2005.

Vol. 3726: L.T. Yang, O.F. Rana, B. Di Martino, J. Dongarra (Eds.), High Performance Computing and Communcations. XXVI, 1116 pages. 2005.

Vol. 3725: D. Borrione, W. Paul (Eds.), Correct Hardware Design and Verification Methods. XII, 412 pages. 2005.

Vol. 3724: P. Fraigniaud (Ed.), Distributed Computing. XIV, 520 pages. 2005.

Vol. 3723: W. Zhao, S. Gong, X. Tang (Eds.), Analysis and Modelling of Faces and Gestures. XI, 423 pages. 2005.

Vol. 3722: D. Van Hung, M. Wirsing (Eds.), Theoretical Aspects of Computing – ICTAC 2005. XIV, 614 pages. 2005.

Vol. 3721: A. Jorge, L. Torgo, P. Brazdil, R. Camacho, J. Gama (Eds.), Knowledge Discovery in Databases: PKDD 2005. XXIII, 719 pages. 2005. (Subseries LNAI).

Vol. 3720: J. Gama, R. Camacho, P. Brazdil, A. Jorge, L. Torgo (Eds.), Machine Learning: ECML 2005. XXIII, 769 pages. 2005. (Subseries LNAI).

Vol. 3719: M. Hobbs, A.M. Goscinski, W. Zhou (Eds.), Algorithms and Architectures for Parallel Processing. XI, 448 pages. 2005.

Vol. 3718: V.G. Ganzha, E.W. Mayr, E.V. Vorozhtsov (Eds.), Computer Algebra in Scientific Computing. XII, 502 pages. 2005.

Vol. 3717: B. Gramlich (Ed.), Frontiers of Combining Systems. X, 321 pages. 2005. (Subseries LNAI).

Vol. 3715: E. Dawson, S. Vaudenay (Eds.), Progress in Cryptology – Mycrypt 2005. XI, 329 pages. 2005.

Vol. 3714: H. Obbink, K. Pohl (Eds.), Software Product Lines. XIII, 235 pages. 2005.

Vol. 3713: L. Briand, C. Williams (Eds.), Model Driven Engineering Languages and Systems. XV, 722 pages. 2005.

Vol. 3712: R. Reussner, J. Mayer, J.A. Stafford, S. Overhage, S. Becker, P.J. Schroeder (Eds.), Quality of Software Architectures and Software Quality. XIII, 289 pages. 2005.

Vol. 3711: F. Kishino, Y. Kitamura, H. Kato, N. Nagata (Eds.), Entertainment Computing - ICEC 2005. XXIV, 540 pages. 2005.

Vol. 3710: M. Barni, I. Cox, T. Kalker, H.J. Kim (Eds.), Digital Watermarking. XII, 485 pages. 2005.

Vol. 3709: P. van Beek (Ed.), Principles and Practice of Constraint Programming - CP 2005. XX, 887 pages. 2005.

Vol. 3708: J. Blanc-Talon, W. Philips, D. Popescu, P. Scheunders (Eds.), Advanced Concepts for Intelligent Vision Systems. XXII, 725 pages. 2005.

Vol. 3707: D.A. Peled, Y.-K. Tsay (Eds.), Automated Technology for Verification and Analysis. XII, 506 pages. 2005.

Vol. 3706: H. Fuks, S. Lukosch, A.C. Salgado (Eds.), Groupware: Design, Implementation, and Use. XII, 378 pages. 2005.

Vol. 3703: F. Fages, S. Soliman (Eds.), Principles and Practice of Semantic Web Reasoning. VIII, 163 pages. 2005.

Vol. 3702: B. Beckert (Ed.), Automated Reasoning with Analytic Tableaux and Related Methods. XIII, 343 pages. 2005. (Subseries LNAI).

Vol. 3701: M. Coppo, E. Lodi, G. M. Pinna (Eds.), Theoretical Computer Science. XI, 411 pages. 2005.

Vol. 3699: C.S. Calude, M.J. Dinneen, G. Păun, M. J. Pérez-Jiménez, G. Rozenberg (Eds.), Unconventional Computation. XI, 267 pages. 2005.

Vol. 3698: U. Furbach (Ed.), KI 2005: Advances in Artificial Intelligence. XIII, 409 pages. 2005. (Subseries LNAI).

Vol. 3697: W. Duch, J. Kacprzyk, E. Oja, S. Zadrożny (Eds.), Artificial Neural Networks: Formal Models and Their Applications – ICANN 2005, Part II. XXXII, 1045 pages. 2005.

Vol. 3696: W. Duch, J. Kacprzyk, E. Oja, S. Zadrożny (Eds.), Artificial Neural Networks: Biological Inspirations – ICANN 2005, Part I. XXXI, 703 pages. 2005.

Vol. 3695: M.R. Berthold, R. Glen, K. Diederichs, O. Kohlbacher, I. Fischer (Eds.), Computational Life Sciences. XI, 277 pages. 2005. (Subseries LNBI).

Vol. 3694: M. Malek, E. Nett, N. Suri (Eds.), Service Availability. VIII, 213 pages. 2005.

Vol. 3693: A.G. Cohn, D.M. Mark (Eds.), Spatial Information Theory. XII, 493 pages. 2005.

Vol. 3692: R. Casadio, G. Myers (Eds.), Algorithms in Bioinformatics. X, 436 pages. 2005. (Subseries LNBI).

Vol. 3691: A. Gagalowicz, W. Philips (Eds.), Computer Analysis of Images and Patterns. XIX, 865 pages. 2005.

Vol. 3690: M. Pěchouček, P. Petta, L.Z. Varga (Eds.), Multi-Agent Systems and Applications IV. XVII, 667 pages. 2005. (Subseries LNAI).

Vol. 3688: R. Winther, B.A. Gan, G. Dahll (Eds.), Computer Safety, Reliability, and Security. XI, 405 pages. 2005.

Vol. 3687: S. Singh, M. Singh, C. Apte, P. Perner (Eds.), Pattern Recognition and Image Analysis, Part II. XXV, 809 pages. 2005.

Vol. 3686: S. Singh, M. Singh, C. Apte, P. Perner (Eds.), Pattern Recognition and Data Mining, Part I. XXVI, 689 pages. 2005.

Vol. 3685: V. Gorodetsky, I. Kotenko, V. Skormin (Eds.), Computer Network Security. XIV, 480 pages. 2005.

Vol. 3684: R. Khosla, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part IV. LXXIX, 933 pages. 2005. (Subseries LNAI).

Vol. 3683: R. Khosla, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part III. LXXX, 1397 pages. 2005. (Subseries LNAI).

Vol. 3682: R. Khosla, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part II. LXXIX, 1371 pages. 2005. (Subseries LNAI).

Vol. 3681: R. Khosla, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part I. LXXX, 1319 pages. 2005. (Subseries LNAI).

Vol. 3679: S.d.C. di Vimercati, P. Syverson, D. Gollmann (Eds.), Computer Security – ESORICS 2005. XI, 509 pages. 2005.

Vol. 3678: A. McLysaght, D.H. Huson (Eds.), Comparative Genomics. VIII, 167 pages. 2005. (Subseries LNBI).

Vol. 3677: J. Dittmann, S. Katzenbeisser, A. Uhl (Eds.), Communications and Multimedia Security. XIII, 360 pages. 2005.

Vol. 3676: R. Glück, M. Lowry (Eds.), Generative Programming and Component Engineering. XI, 448 pages. 2005.

Vol. 3675: Y. Luo (Ed.), Cooperative Design, Visualization, and Engineering. XI, 264 pages. 2005.

Vol. 3674: W. Jonker, M. Petković (Eds.), Secure Data Management. X, 241 pages. 2005.

Vol. 3673: S. Bandini, S. Manzoni (Eds.), AI*IA 2005: Advances in Artificial Intelligence. XIV, 614 pages. 2005. (Subseries LNAI).

Vol. 3672: C. Hankin, I. Siveroni (Eds.), Static Analysis. X, 369 pages. 2005.

Vol. 3671: S. Bressan, S. Ceri, E. Hunt, Z.G. Ives, Z. Bellahsène, M. Rys, R. Unland (Eds.), Database and XML Technologies. X, 239 pages. 2005.

Vol. 3670: M. Bravetti, L. Kloul, G. Zavattaro (Eds.), Formal Techniques for Computer Systems and Business Processes. XIII, 349 pages. 2005.

Vol. 3669: G.S. Brodal, S. Leonardi (Eds.), Algorithms – ESA 2005. XVIII, 901 pages. 2005.

Vol. 3668: M. Gabbrielli, G. Gupta (Eds.), Logic Programming. XIV, 454 pages. 2005.

Vol. 3666: B.D. Martino, D. Kranzlmüller, J. Dongarra (Eds.), Recent Advances in Parallel Virtual Machine and Message Passing Interface. XVII, 546 pages. 2005.

Vol. 3665: K. S. Candan, A. Celentano (Eds.), Advances in Multimedia Information Systems. X, 221 pages. 2005.

Vol. 3664: C. Türker, M. Agosti, H.-J. Schek (Eds.), Peer-to-Peer, Grid, and Service-Orientation in Digital Library Architectures. X, 261 pages. 2005.

Vol. 3663: W.G. Kropatsch, R. Sablatnig, A. Hanbury (Eds.), Pattern Recognition. XIV, 512 pages. 2005.

Vol. 3662: C. Baral, G. Greco, N. Leone, G. Terracina (Eds.), Logic Programming and Nonmonotonic Reasoning. XIII, 454 pages. 2005. (Subseries LNAI).

Vol. 3661: T. Panayiotopoulos, J. Gratch, R. Aylett, D. Ballin, P. Olivier, T. Rist (Eds.), Intelligent Virtual Agents. XIII, 506 pages. 2005. (Subseries LNAI).

Vol. 3660: M. Beigl, S. Intille, J. Rekimoto, H. Tokuda (Eds.), UbiComp 2005: Ubiquitous Computing. XVII, 394 pages. 2005.

Vol. 3659: J.R. Rao, B. Sunar (Eds.), Cryptographic Hardware and Embedded Systems – CHES 2005. XIV, 458 pages. 2005.

Vol. 3658: V. Matoušek, P. Mautner, T. Pavelka (Eds.), Text, Speech and Dialogue. XV, 460 pages. 2005. (Subseries LNAI).

Vol. 3657: F.S. de Boer, M.M. Bonsangue, S. Graf, W.-P. de Roever (Eds.), Formal Methods for Components and Objects. VIII, 325 pages. 2005.

Vol. 3656: M. Kamel, A. Campilho (Eds.), Image Analysis and Recognition. XXIV, 1279 pages. 2005.

Vol. 3655: A. Aldini, R. Gorrieri, F. Martinelli (Eds.), Foundations of Security Analysis and Design III. VII, 273 pages. 2005.

Vol. 3654: S. Jajodia, D. Wijesekera (Eds.), Data and Applications Security XIX. X, 353 pages. 2005.

Vol. 3653: M. Abadi, L. de Alfaro (Eds.), CONCUR 2005 – Concurrency Theory. XIV, 578 pages. 2005.

Vol. 3652: A. Rauber, S. Christodoulakis, A M. Tjoa (Eds.), Research and Advanced Technology for Digital Libraries. XVIII, 545 pages. 2005.

Vol. 3651: R. Dale, K.-F. Wong, J. Su, O.Y. Kwong (Eds.), Natural Language Processing – IJCNLP 2005. XXI, 1031 pages. 2005. (Subseries LNAI).

Vol. 3650: J. Zhou, J. Lopez, R.H. Deng, F. Bao (Eds.), Information Security. XII, 516 pages. 2005.

Vol. 3649: W.M. P. van der Aalst, B. Benatallah, F. Casati, F. Curbera (Eds.), Business Process Management. XII, 472 pages. 2005.

¥396.48元

Table of Contents

Emergency Telecommunication Support for IP Telephony

Francesco Moggia, Mudumbai Ranganathan, Eunsook Kim, and Doug Montgomery

Advanced Networking Technologies Division,
National Institute of Standards and Technology,
Gaithersburg, MD 20899, USA
{fmoggia,mranga,eunah,dougm}@nist.gov
http://w3.antd.nist.gov

Abstract. As universal high speed internet access becomes a reality, phone calls are increasingly being made over the Internet rather than the conventional PSTN. The danger to this trend is the un-availability of priority mechanisms for communication between emergency response personnel during times of disaster. We define a proposed architecture to enable ETS support for SIP-based VOIP systems.

1 Introduction

The abundance of available bandwidth has spurred the growth of companies that provide telephony services over the public Internet at minimal cost. This has prompted many to abandon the Public Switched Telephone Network (PSTN) altogether and rely solely upon IP Telephony. IP Telephony offers some advantages for disaster response. For example, one may directly contact the Public Safety Access Point (PSAP) of a region by specifying the region in a URL or as URL parameters. The personal mobility offered by such technologies permits people to contact each other regardless of physical location. Integration with web technologies and enhanced methods of communication also permit richer interactions between emergency response personnel, resulting in an improved ability to react to the emergency scenario at hand. However there are some negative aspects to IP Telephony under such circumstances. Recent experience with disaster scenarios indicate that the public telephone network will become so overloaded that effective communication is no longer possible. Similarly, for IP telephony, without dedicated capacity, it is expected that an overload of the signaling servers and media gateways will occur.

The PSTN is regulated by Government and providers are required to allocate dedicated resources for emergency calling – allowing authorized personnel to complete even when the normal (unreserved) network is fully saturated. However, the same cannot be guaranteed for telephony over the Internet. Service providers are generally reluctant to accept any form of government regulation. How do we provide authorized personnel with priority calling under these circumstances? In this paper we outline a solution to this problem by proposing a coupling of signaling and Quality of Service (QOS) mechanisms for IP telephony.

There are two essential parts to IP Telephony – signaling or call setup and media (RTP). To devise an effective solution for emergency telecommunications both signaling and media issues should be addressed. During disasters, signaling services

T. Magedanz, E.R.M. Madeira, and P. Dini (Eds.): IPOM 2005, LNCS 3751, pp. 1–8, 2005.

may become saturated and media gateways may suffer overloads. Emergency calls may need to traverse and Broadband access networks and sections of the Internet located in regions of high congestion to reach other emergency workers. In this paper we make the assumption that QOS mechanism such as DIFFSERV is deployed in the network and propose architecture based on this assumption.

The rest of this paper is organized as follows. In section 2 we briefly outline the Emergency scenario to set the stage for the issues involved in contrast with how these issues are addressed in the PSTN. In section 3, we discuss these issues in greater detail. In section 4 we outline the architectural assumptions of our proposed solution. In section 5 and 6 we detail or proposed solution.

2 The Emergency Scenario

The requirements for IP Emergency Telecommunication systems are outlined in RFC3690 When a public disaster occurs, emergency workers need to communicate with other emergency workers outside the disaster zone for effective coordination. Under typical circumstances they will be using the same telephony signaling servers and media gateways that are being used by regular phone users, thus resulting in an overload of these.

The PSTN has government mandated support for emergency calling. This service is known as GETS (Government Emergency Telecommunications Support). A GETS user is issued a special card that can be used from any public telephone. This gives the user a higher probability of call completion by dedicating resources to the call. Thus even when regular users are getting busy signals or "all circuits are busy" messages, the GETS user gets through. GETS support on PSTN relies on (1) the availability of alternate paths (2) preferential treatment at Signaling Control Points (SCPs) – a GETS call setup message is less likely to be dropped in an SS7 network. Similar to GETS, the Wireless Priority Service (WPS) provides resource prioritization for emergency call setup for wireless technologies. There is no equivalent emergency call setup mechanism over the IP network today.

3 Issues in Supporting ETS in IP Telephony

Since the PSTN is a managed network end-to-end measures that give priority to GETS calls are possible. Unlike the PSTN, the IP network uses dynamic routing of packets and the packet may traverse multiple IP service provider networks. A call may not be IP End to End; a call may traverse multiple segments – some of which are circuit switched and others of which are packet networks. Each such switching point will have a gateway that supports a fixed number of trunk lines. Some sections of the network may thus be supported by GETS/WPS and others may not. Such a hybrid scenario is in Figure 1.

During call setup, when the call transitions from IP to PSTN, the signaling server communicates with a server to potentially reserve a trunk line for the call. The signaling and media for an end-to-end call may in fact traverse several such gateways. When a given call transition between the PSTN and Internet or cellular network, there is a high probability of trunk blocking caused by the limited number of trunk lines supported by the gateway. It may thus become necessary to pre-empt trunk lines for use by emergency workers or reserve trunk lines for emergency use. The challenges at hand are (1) Authentication and authorization for the use of ETS over IP networks. (2) Agreed upon service policies with service providers (3) Identification of Emergency connection requests (4) Priority treatment call setup signaling and media (5) Reservation of resources for emergency calls (6) Priority reservation of trunks and bandwidth resources under emergency conditions.

In this paper, we restrict our attention to an end-to-end IP based solution and consider the support requirements of the access point to the IP network.

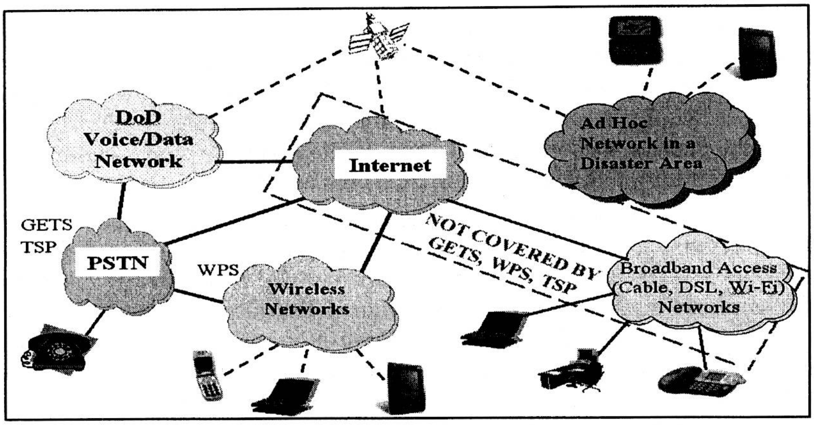

Fig. 1. Emergency Preparedness – GETS, WPS and TSP. An emergency call may have to traverse different networks operated by different providers

4 ETS Support of IP Telephony – Architectural Assumptions

For the purposes of this paper, we assume that SIP is the signaling protocol for call setup. There are two sub-problems that need to be addressed – (1) signaling prioritization for high priority calls and (2) preferential handling of media packets for emergency workers to communicate in a congested network. The main issues in achieving these objectives are (1) mechanisms resource prioritization at the service platform for preferential treatment of the call and (2) guarding against masquerading by regular users to get better quality of service. The main focus of this paper is to define an architecture to achieve these objectives.

5 Supporting Resource Prioritization in a Service Platform

Services are fragments of code that intervene during the signaling for call setup. The Service Container or Service platform is a managed, middleware environment where these fragments of code are installed and execute in a contained environment. The Service Plane is where the network intelligence resides. Standards are just being defined for service architectures at present. The JAIN Service Logic Execution Environment (JSLEE) [3] is an emerging JAVA standard that is designed explicitly for high throughput, low latency asynchronous applications such as IP Telephony Services.

5.1 Architectural Overview of the Service Platform

To experiment with ideas on how to support ETS in managed networks, we picked JSLEE as a concrete standard because it offers a complete set of features and facilities. We now give a brief description of JSLEE in this section and outline the enhancements to support Signaling Prioritization.

The JSLEE application server is composed of three main parts:

- The Execution Logic: this is the core of the SLEE that runs the deployed application
- The Resource Adaptor: Resources are the way in which a SLEE application can interact with the outside world (Protocol Stacks, Databases, Network Devices)
- Management Interface and Facilities: The JSLEE standard provides a Management Interface that allows the system administrator to deploy a service inside the container, manage the operational state of the service, receive alarm and trace from the application that are running inside the SLEE.

A JSLEE Service is composed of elementary components called Service Building Blocks (SBB) that are composed together in a parent-child relationship with execution priorities that defines their execution order. The SLEE is responsible for instantiating services and instances of the components of a service and routing events to the components at run time. JSLEE follows a Publish Subscribe model i.e. applications are bundled with deployment descriptors that specify the conditions under which component instances receive events and the conditions under which Services are instantiated. A SBB is essentially an event handler. It is executed by the container when an event that it is interested in is received. A Service is a composition of these event handlers. The SLEE manages the instantiation and destruction of the SBBs in addition to routing events to it. A Service developer defines the conditions under which the Service is instantiated and installs a template of the service using a deployment descriptor. A Service instance can be seen as a tree composed of SBB instances. Event routing and priority is controlled by the parent child relationship defined in the Service deployment descriptor, which determines the delivery order of the events to the SBBs that are part of the same service.

JSLEE defines an abstraction for an event bus. Events are logically grouped and fired in FIFO order on an event bus or Activity by a *Resource Adaptor* (protocol stack). This is mapped by the SLEE to an internal data structure known as an *ActivityContext*. The ActivityContext has a corresponding application-visible data structure called an *ActivityContextInterface*. Protocol stacks (such as a SIP protocol stack) are pluggable *Resource Adaptors*. The primary function of such a Resource Adaptor is to identify streams of events and place the event on the SLEE abstraction for the related stream of events (i.e. the ActivityContext). The SLEE's primary job at runtime is to identify services that may need to be instantiated on receipt of these events and route events to the SBBs of such services (which may already be instantiated previously).

As a concrete example, the Java Call Control (JCC) Resource Adaptor may regard a phone call as an Activity with an accompanying ActivityContext and identify the incoming signaling messages related to the call as belonging to that Activity (in the case of SIP, it would use the SIP Call ID to do so). In order to receive events related to an Activity an SBB should be attached to the corresponding ActivityContext. This attachment is done by the SLEE runtime environment the SLEE Event Router.

5.2 Supporting Resource Prioritization in JAIN-SLEE

The Signaling server can become the choke point during times of high utilization. During busy periods, it can be expected to handle thousands of call attempts per second. Hence it is important to have mechanisms to prioritize CPU use and have mechanisms by means of which high priority signaling events get preferential treatment during times of emergency.

We propose mechanisms to deal with this. These mechanisms are geared towards increasing the platform resources available to handle calls from emergency response personnel and rely on the decoupled publish-subscribe model and the component model of the SLEE.

When a Service is specified, it is defined by the user as a set of Abstract Classes which comprise its SBBs and a deployment descriptor that ties these together. When the SLEE installs the Service are converted by the SLEE Deployable Unit loader to concrete Java classes and each SBB is assigned its own Java class-loader, thus providing environment isolation. This allows us the opportunity to insert monitoring code fragments using bytecode rewriting techniques to ensure that an SBB cannot exceed allocated CPU resources during periods of high CPU utilization. Second, we can prioritize events so that during high utilization, Services with higher priority get events routed to them in preference to lower priority services. Third, we need to support Authentication, Authorization and Roles for Services to prevent unauthorized triggering of Emergency Services.

The SLEE specification mandates that events belonging to a given Activity must be queued and delivered in first-come first-served order. The Activity priority extension to the SLEE is based on prioritizing activities or related streams of events. All the events fired on a given Activity are given the same priority value. The Event Router consumes events from multiple queues by selecting the event to be processed from the higher-priority non-empty queue as suggested in [draft-ietf-resource-priority-08]. To make the suggestion concrete we explain the operation in terms of SIP.

When a call is set up, the User Agent (IP Phone initiating the call) issues a SIP INVITE Message. The Call-ID header of the outgoing SIP INVITE identifies the call is associated with a SLEE Activity that is created for the call. A high-priority signaling message will have a Resource-Priority header. However, this is clearly subject to abuse by ordinary (non-emergency) users. Thus the SIP Resource Adaptor needs to authenticate and the SIP Message and authorize the user before believing the Resource-Priority value. After successful authentication, the event is assigned to the appropriate high priority queue according to the value of the header, otherwise the dropped as a spurious call setup attempt. After authentication all subsequent signaling messages related to the call may receive priority treatment. Note that these messages will all be identified with the same Activity and that the activity lasts for the duration of the call. (If mobility is supported at the Signaling layer, there could be several such messages as the emergency responder moves around in the affected area). As recommended in [draft-ietf-resource-priority-08] six different priority queues are supported in order to implements the "ETS" namespace. In addition there is one default priority queue for ordinary users.

During periods of high utilization, it may become necessary to pre-empt the execution of certain low-priority SBBs to allow higher priority SBBs more execution cycles. To do this, we devise mechanisms that limit the CPU utilization of an individual SBB by placing monitoring calls directly in the SBB code at the end of every basic block or non-branching sequence of bytecode instructions. We can accomplish this using bytecode re-writing at the time the SBB is installed.

Certain sensitive information such as location data and location of other emergency personnel may only be accessible by Authorized emergency personnel. These resources may implement their own authentication mechanisms at an application layer. However, we would like to support an overall policy framework that can be administered by the SLEE Administrator in a uniform way and without detailed knowledge of the authentication mechanisms of each of the resources accessed by the SBB. To support this we add the notion of Roles and Permissions to SLEE. Each SLEE service is assigned Permission. Roles and Permissions are comparable with each other. Each call is assigned a Role after the user is authenticated. The Service may only be instantiated if the Role of the user exceeds the Permission of the Service. Because the SLEE is protocol agnostic, we need to support this feature in such a way that our mechanisms will apply to a variety of Resource Adaptors and Services. For this we use the Java Authentication and Authorization Framework (JAAS). Details of our design are deferred to a more detailed paper.

6 Improving Quality of Service for Emergency Calls

The IETF document [4] describes call prioritization via the resource priority header and explains integration of resource management and SIP. A SIP User Agent (i.e. IP Phone or IM agent) must distinguish an emergency call and make arrangements for better quality of service for that call. We need an efficient and secure coupling between QoS supporting networks and the User Agent that originates the emergency call. As a minimal mechanism, we can simply drop the normal calls currently in progress to improve the chances of acquiring enough bandwidth for the high priority emergency call. Signaling protocols such as SIP support third party call control [5] which can be used to accomplish this. A BYE signal can be sent to both communicating parties resulting in the call being dropped. This is clearly a simplistic solution which users may find objectionable. If the underlying network supports a QOS mechanism such as DIFFSERV [6], a better solution may be devised.

With DIFFSERV, routers give preferential treatment to marked IP Packets. However, such a scheme is subject to abuse. Only authorized personnel should be allowed to originate IP Packets that get priority treatment at routers. To deal with this, we introduce an IP to IP media gateway as shown in the Figure 2.

The job of the media gateway is to mark packets originating from authenticated IP addresses that it knows about. The core network will only accept packets marked by the media gateway. The media gateway is in turn informed by the signaling server (i.e. service inside the SLEE) about authenticated and authorized users (i.e. IP Source addresses) from which it may accept incoming packets for marking. Non-emergency call packets are not marked and hence enter the core network as such and receive best effort service. The core network rejects any IP Packets that are marked for priority handling which do not originate from the gateway. The gateway thus functions as an admission controller – only allowing packets from authorized emergency personnel to be marked for better quality of service.