Towards Autonomic Networks

Erol Gelenbe
Memb. Acad. Europ., FIEE FIEEE FACM
Dennis Gabor Chair
Intelligent Systems and Networks Group
Electrical & Electronic Engineering Dept.
Imperial College
London SW7 2BT
e.gelenbe@imperial.ac.uk

Abstract

The Internet is becoming an immense organism of composite, highly distributed, pervasive, communication intensive services. For such a system to operate effectively, a ensible dialogue between users, services and the network components must proceed constantly based on mutual observation, self-observation, and adaptive and distributed feedback control. We review issues such as network "situational awareness", self-organisation, and structure, and relate these concepts to research on autonomic communication systems. We discuss how this vision can benefit from techniques that have been experimented in the Cognitive Packet Network (CPN) test-bed at Imperial College, which dynamically routes traffic using on on-line monitoring, based on users' QoS needs and the network's objectives.

1 Introduction

Very soon, all everyday activities will be supported by a ubiquitous ITC environment or "networked service" that caters to our needs in a situation-aware manner. This service will [27, 46] autonomously detect and organize the knowledge necessary to understand the physical and social context, and adapt to the human, social and technical environment. Such services will enable us to interact with the world by providing us with any needed guidance and information as we go along with our work, family life or leisure. For instance, a traveller arriving at some location to go to a meeting at hotel X, then to have lunch with A at some restaurant Y, and later with stop at at coffee shop Z to get some work done [23] will be supported by his communication services to collect relevant information via local networks, web sites, the user's office system, the ITC system related to some local organisation that is hosting the meeting and from person B's comunication service. This information will be displayed on A's personal device, and will trigger other actions or information searches. We imagine that this will be done at the lowest cost and the best possible QoS [26, 38].

Such future services challenge today's networks and require that the networking world integrate technologies from other application areas as well as from advanced research in ITC, including: *Situational awareness* which is a concept borrowed from the defence area. Situational awareness in communications can be provided by sensors, location systems, user profiles, and tools for system and network monitoring [30] and provisioning [49, 45]. *Self-organization* will be required to automatically exploit the ability for situational awareness, going beyond the structural self-organisation of Peer-to-Peer networks [11], and borrowing techniques from swarms , reinforcement learning, and social networks [4, 5, 12, 21], and also bridge these robust but conceptually simple biological paradigms with the complex semantics [18, 43] of context-aware systems. The *Structure and Components* of such systems will move away from the top-down hierarchical protocol stack that has dominated networks for several decades, towards collections of agents or "autonomic components" that may use a common template [27, 37] as a software framework for the interactions between agents [28] and for programming this environment [7, 22].

2 The Search for Services

The network service will call upon directory services that will offer "how to get there" information providing a network path from the point where the to the system where

Proceedings of the 2005 Symposium on Applications and the Internet (SAINT'06)
0-7695-2262-9/06 $20.00 © 2006 IEEE

the service can be found. Directories will be updated proactively by the services or by the directories themselves, or on demand. They will be "smart" enought to offer information about faster or less congested paths, less expensive services, etc.

Smart search can be implemented using ideas similar to the Cognitive Packet Network (CPN) algorithm [5, 42] that finds a network path to a destination while optimising a QoS requirement (see $http://san.ee.ic.ac.uk$). CPN uses dumb packets (DPs) to carry the payload traffic. CPN routers are interconnected via portions of the Internet or by direct high-speed physical connections. Smart packets (SPs) in CPN find routes and collect measurements, but do not carry payload. DPs are source routed, using paths which best match the users' QoS requirements. On the other hand, SPs are routed using a Reinforcement Learning (RL) algorithm that uses the observed outcome of previous decisions to "reward" or "punish" the mechanism that lead to the previous choice, so that its future decisions are more likely to meet the QoS goal. When a SP arrives to its destination, an acknowledgement (ACK) packet is generated; the ACK stores the "reverse route" and the measurement data collected by the SP. It will travel along the "reverse route" which is computed by taking the corresponding SP's route, examining it from right (destination) to left (source), and removing any sequences of nodes which begin and end in the same node. For instance, the path $< a, b, c, d, a, f, g, h, c, l, m >$ will result in the reverse route $< m, l, c, b, a >$. Note that the reverse route is not necessarily the shortest reverse path, nor the one resulting in the best QoS. The route brought back by an ACK is used as the source route by subsequent DPs of the same QoS class having the same destination, until a newer and/or better route is brought back by another ACK. A *Mailbox (MB)* in each node is used to store QoS information. Each MB is organized as a Least-Recently-Used (LRU) stack, with entries listed by QoS class and destination, which are updated when an ACK is received.

By analogy with the above presentation, the steps needed to establish a connection between some user U and a service S can be listed as follows. U first searches for a directory; assuming he finds one, U formulates his request in the form of (SX, QY, PZ) meaning that he wants a service SX at QoS value QY for a price of PZ. The directory either is unable to answer the request, or it provides one or more paths $\pi(U, SX, QY, PZ)$ which best approximate this request for several possible locations of the service. Assuming that the directory does provide the information, U sends out (typically via the node) a sequence of smart packets SPs which have the desired QoS information, with several following each of the possible designated paths. The first SP for each of the paths will follow it to destination, with the purpose of verifying that the information provided by the directory is correct. Subsequent SPs on each route will be used to search for paths: they will invoke an optimisation algorithm at all or some of the Nodes they traverse so as to seek out the best path with respect to the user's QoS and pricing requirements. Nodes collect measurements and store them in mail boxes (MB). These can concern both short term measurements which proceed at a fast pace comparable to the traffic rates, and long term historical data. Nodes will measure packet loss rates on outgoing links and on complete paths, delays to various destinations, possibly security levels along paths (when security is part of a QoS requirement), available power levels at certain mobile nodes, etc.. This constant monitoring can be carried out using the SPs and other user related traffic, or using specific sensing packets generated by the Nodes. The network monitoring function can also be structured as a special set of users and services whose role is to monitor the network and provide advice to the users and to the directories. Each SP also collects measurements from the Nodes it visits which are relevant to its users QoS and cost needs, about the path from the Nodes which it visits. When a SP reaches a service SX, an acknowledgement ACK packet is sent back along the reverse path back to U; the ACK carries the relevant QoS information, as well as path information which was measured by the SP and by the ACK, back to the Nodes and to the user U. The ACK may thus be carrying back a new path which was unknown to the directory. For a variety of reasons, both SPs and ACKs may get lost. SPs or ACKs which travel through the network over a number of hops (ERs or total number including routers within the clouds) exceeding a predetrmined fixed number, will be destroyed by the routers to avoid congesting the IN with "lost" packets. Note that the SPs and ACKs may be emitted by the directory itself, rather than by U. This would be an additional service offered by certain directories. One could also imagine that both users and directories have this capability so as to verify that the request is being satisfied.

3 Individual versus Collective QoS Goals

The usual question that a telecommunications engineer will as at this point is what will happen when individual goals of users and services conflict with the superior goals of the system, or the perceived collective good. We are allowing for users to set up the best paths they can find, from a selfish perspective, with services, and for services to actually do the same, in parallel with the behaviour of users. This may overload the infrastructure, because services have an interest in maximising their positive response to user's needs, and they may even overdo it in terms of solliciting users; portions of the infrastructure itself may have an interest in getting overloaded. Also, traffic congestion and oscillations between hot spots can occur due to users and

services switching constantly to seemingly better ways to convey their traffic. Malicious traffic may also be generated, whose purpose is to deny service to legitimate users through the creation of overload in the services or the infrastructure (e.g. denial of service attacks). Some of these aspects can be handled through self regulation of the network. We could have a virtual regulating agency (VRA) which sets up a dialogue with an incoming user, service or component to provide it with its identity, and to ascertains its type, nature from its technical characteristics. The VRA then enables it with a set of parameters which in effect limit its access to other components of the network. Services and users are identified by the VRA. Just as a shop rents space in a building and on a particular street, the VRA can provide each service with a "footprint", depending on the rent it is willing to pay, and on the VRA's knowledge of currently available resources, that determines the amount of processing power and bandwidth that it is allowed inside the network and at any given node. The overall quality and seriousness of the VRA will make a particular network more or less desirable to users and services. The second point is related to dynamic behaviour. Each INR, in its role as a service support centre enabled by the VRA, will run the dynamic flow and workload control algorithms for each service and user that it hosts. However it will also run a monitoring algorithm which has IN-wide implications.

In the aproach that we have suggested for finding services, a user formulates yhe request (SX, QY, PZ) for a service SX at quality level QY and for the price PZ. Both the quality of service value and the price constitute "goals" in the sense that the term is used in the CPN algorithm [39]. They may be treated as separate goals to be minimised, and combined in some manner as outlined above, or combined into some single common metric. For instance, if QY is some non-negative number such as "loss" or "delay", we could combine the two considerations in a single metric such as $G = QY/PZ$ (quality for a given price), or as

$$G = PZ\ 1[QY < Qmax] + \frac{QY}{PZ} 1[QY \geq Qmax] \quad (1)$$

where $1[x]$ is the function which takes the value one if the predicate $x = true$ and takes the value zero if $x = false$. Thus (1) means, for instance, that as long as the delay is less than some maximum acceptable value $Qmax$, we are happy to minimise the price; however if the delay is larger than this maximum value, we want simply to minimise the delay per price unit that we pay for the service.

Scalability is another important question that one needs to address. If each Internet router were enabled to deal with the QoS needs of each connection, it would have to identify and track the packets of each individual connection that is transiting through it. In the system we describe, we propose to avoid the scalability issue by making each network router responsible only for local users and services, much as a local telephone exchange handles its local users. Source routing removes the burden of routing decisions from all but the local router, reducing overhead, and removing the need of "per flow" information handling except at the router where the flows are resident. However, it comes at the price of being less rapidly responsive to changes that may occur in the network, which can be compensated by constant monitoring of the flows.

4 Protecting the network Against Denial of Service Attacks

Denial of Service (DoS) attacks are known to the network research community since the early 1980s. Although initially DoS attacks were the act of hackers who wanted to demonstrate their ability and power over computer systems, they have now become an important weapon in the hands of cyber-criminals and for cyber-warfare. DoS attacks have reportedly been used against business competitors, for extortion purposes, for political reasons, and even as a form of "legitimate" protest. It is this variety of targets and types of attack that dictate the need for flexible defence systems which can react according to both the attacker's aims and the defender's needs.

A DoS attack is very often distributed (DDoS): the attacker takes control of a large number of networked computers and orders them to send a large number of packets to a specific target node, server or web site. Typical targets of an attack may include servers or web sites which accomplish some function in the public interest, or the servers of e-commerce web-sites which can suffer significant financial loss. Other targets may be news web-sites, corporate networks, banks, etc. Often, the attacker needs to conceal its identity and the nodes that are used in the attack. It can do this does by introducing fake IP addresses into the packets ("IP spoofing"), providing a false identity for the nodes that generate the attack. Indeed, in a seminal paper on some of the weaknesses of the TCP/IP protocol [1] it is said that (quote)"The weakness in this scheme (the IP protocol) is that the source host itself fills in the IP source host id, and there is no provision to discover the true origin of the packet".

Since attacks occur rapidly and unexpectedly, it is essential to incorporate an autonomic approach to defence based on network self-observation and adaptive reaction. The effective protection of the network from DoS attacks requires the combination of different elements:

- ***Detection*** of the existence of an attack. The detection can be either anomaly-based or signature-based, or a hybrid of those two. In anomaly-based detection, the system recognises a deviation from the standard be-

haviour of its clients, while in the latter it tries to identify the characteristics of known attack types.

- ***Classification*** of the incoming packets into valid (normal packets) and invalid (DoS packets). As in detection, one can choose between anomaly-based and signature-based classification techniques.
- ***Response*** The protection system should either drop the attacking packets or redirect them into a trap for further evaluation.

We have developed and tested an autonomic approach to DoS defence using the CPN protocol [50]. In this approach, a node being attacked must be able to sense the attack, or be informed about it. The nodes that are upstream from the victims towards the sources of the attack will be informed and will participate in the defence. The essence of the defence is to drop packets which belong to streams which have been identified as participating in the attack. However, both false alarms and detection errors can occur, so that some "innocent" traffic may be dropped by mistake and some "guilty" traffic may get through. An alternate to dropping the attacking traffic is to divert all or part of it to a "honey-pot" or sink where it is stored for analysis. Our approach to detection is to use QoS considerations, since an attack will provoke congestion and buffer overflows, and the attacking traffic may be perceived as exceeding some pre-assigned traffic rate or proportion of service.

5 Conclusions

Autonomic networks can offer a user-friendly and self-organising communication environment for users and services. In such systems, users will dynamically indicate their requests for services, and formulate needs in terms of Quality of Service (QoS) and price. They will also monitor the level at which which their requests are satisfied. Services will dynamically try to satisfy the users, while the network will act as a mediator to provide guidelines or constraints to avoid that different entities impede each others' progress. We suggest that these ideas can be supported by systems that incorporate some of the techniques offered by the CPN system. Other important issues, such as the energy efficient operation of networks [41], also need to be considered.

References

[1] R.T. Morris. A Weakness in the 4.2BSD Unix TCP/IP Software. *Technical Report Computer Science #117*, AT&T Bell Labs, February 1985.

[2] P. Ferguson and D. Senie. Network Ingress Filtering: Defeating Denial of Service Attacks which Employ IP Source Address Spoofing. *Tech. Rep. RFC 2267*, January 1998.

[3] D. Williams and G. Apostolopoulos. QoS Routing Mechanisms and OSPF Extensions. RFC 2676, Aug. 1999.

[4] E. Bonabeau, M. Dorigo, G. Theraulaz. Swarm Intelligence: From Natural to Artificial Systems. New York, NY, Oxford University Press, 1999.

[5] E. Gelenbe, Z. Xu, E. Şeref. Cognitive packet networks. *Proc. 11th IEEE Int. Conf. on Tools with Artificial Intelligence (TAI99)*, 47-54, Chicago, Ill., 1999.

[6] S. Savage, D. Wetherall, A. Karlin, and T. Anderson. Practical Network Support for IP Traceback. *Proc. ACM SIGCOMM*, 295-306, Stockholm, Sweden, August 2000.

[7] C. Tschudin, H. Lundgren, H. Gulbrandsen. Active Routing for Ad Hoc Networks. IEEE Communications Magazine, April 2000.

[8] D. Song and A. Perrig. Advanced and Authenticated Marking Schemes for IP Traceback. *Proc. Infocom 2001*, ISBN: 0-7803-7016-3, vol. 2, pp. 878-886, Anchorage, Alaska, USA, 22-26 April 2001.

[9] T. Berners-Lee, J. Hendler, O. Lassila. The Semantic Web. Scientific American, May 2001.

[10] G. P. Picco, A. L. Murphy, G. C. Roman. LIME: a Middleware for Logical and Physical Mobility. 22nd IEEE Intl. Conference Distributed Computing Systems, 2001.

[11] S. Ratsanamy,, P. Francis, M. Handley, R. Karp. A Scalable Content-Addressable Network. ACM SIGCOMM Conference, Aug. 2001.

[12] E. Gelenbe, E. Seref and Z. Xu. Simulation with learning agents. *Proceedings of the IEEE*, 89 (2), pp. 148-157, 2001.

[13] V. Paxson. An Analysis of Using Reflectors for Distributed Denial-of-Service Attacks. *ACM Computer Communications Review 31(3)*, July 2001.

[14] E. Gelenbe, R. Lent and Z. Xu. Design and performance of cognitive packet networks. *Performance Evaluation*, 46, pp. 155-176, 2001.

[15] G. Rice and J. Davis. A Genealogical Approach to Analyzing Post-Mortem Denial of Service Attacks. *Secure and Dependable System Forensics Workshop*, University of Idaho, September 23-25, 2002.

[16] BBC News. Mafiaboy hacker jailed. (September 13, 2001), http://news.bbc.co.uk/1/hi/sci/tech/1541252.stm.

[17] E. Gelenbe, R. Lent, and Z. Xu. Cognitive Packet Networks: QoS and Performance. *Proc. IEEE MASCOTS Conference*, ISBN 0-7695-0728-X, pp. 3-12, Fort Worth, TX, Oct. 2002.

[18] I. Horrocks, P. Patel-Schneider, F. van Harmelen. Reviewing the design of DAML+OIL: An ontology language for the semantic web" National Conference on Artificial Intelligence, Edmonton, Alberta, Canada, 2002.

[19] R. Mahajan, S. Bellovin, S. Floyd, J. Ioannidis, V. Paxson, and S. Shenker. Controlling High Bandwidth Aggregates in the Network. *ACM SIGCOMM Computer Communication Review*, ISSN: 0146-4833, Vol. 32, Issue 3, pp. 62–73, July 2002.

[20] E. Gelenbe, R. Lent, and Z. Xu. Cognitive Packet Networks: QoS and Performance. *Proc. IEEE MASCOTS Conference*, ISBN 0-7695-0728-X, pp. 3-12, Fort Worth, TX, October 2002.

[21] R. Albert, A. Barabasi. Statistical Mechanics of Complex Networks", Rev. Mod. Phys. 74 (47), 2002.

[22] C. Borcea, et al.. Cooperative Computing for Distributed Embedded Systems", 22nd International Conference on Distributed Computing Systems, Vienna (A), IEEE CS Press, 227-238, 2002.

[23] D. Estrin, D. Culler, K. Pister, G. Sukjatme. Connecting the Physical World with Pervasive Networks", IEEE Pervasive Computing, 1 (1): 59-69, Jan. 2002.

[24] A. Snoeren, C. Partridge, L.A. Sanchez, C.E. Jones, F. Tchakountio, B. Schwartz, S. Kent, and W.T. Strayer. Single-Packet IP Traceback. *IEEE/ACM Transactions on Networking*, ISSN: 1063-6692, Vol. 10, no. 6, pp. 721-734, December 2002.

[25] W.G. Morein, A. Stavrou, D.L Cook, A.D. Keromytis, V. Mishra, and D. Rubenstein. Using Graphic Turing Tests to Counter Automated DDoS Attacks against Web Servers. *Proc. 10th ACM Int'l Conference on Computer and Communications Security (CCS '03)*, ISBN: 1-58113-738-9, pp. 8-19, Washington DC, USA, October 27-30, 2003.

[26] L. Capra, W. Emmerich, C. Mascolo. CARISMA: Context-Aware Reflective mIddleware System for Mobile Applications", IEEE Transactions of Software Engineering Journal (TSE) 29(10):929-945, 2003.

[27] J. Kephart, D. Chess. The Vision of Autonomic Computing", IEEE Computer, 36 (1), 2003.

[28] F. Zambonelli, N. Jennings, M. Wooldridge. Developing Multiagent Systems: the Gaia Methodology", ACM Transactions on Software Engineering and Methodology, 12 (3):317-370, 2003.

[29] M. Papazoglou, M. Aiello, M. Pistore, J. Yang. XSRL: A Request Language for Web Services (www.webservices.org)", 2003

[30] M. Philipose, K. Fishkin, M. Perkowitz, D. Patterson, D. Fox, H. Kautz, D. Hahnel. Inferring Activities from Interactions with Objects", IEEE Pervasive Computing, 3(4):50-57, 2004.

[31] S. Jing, H. Wang, and K. Shin. Hop-Count Filtering An Effective Defense Against Spoofed Traffic. *Proc. ACM Conference on Computer and Communications Security*, pp. 30-41, ISBN 1-58113-738-9, Washington DC, October 2003.

[32] G. Mori and J. Malik. Recognizing objects in adversarial clutter - Breaking a visual CAPTCHA. *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2003 (CVPR '03)*, ISSN: 1063-6919, ISBN: 0-7695-1900-8, vol. 1, pp. 134-141, Madison, WI, USA, June 18-20, 2003.

[33] M. Sung and J. Xu. IP Traceback-Based Intelligent Packet Filtering: A Novel Technique for Defending against Internet DDoS Attacks. *IEEE Transactions on Parallel and Distributed Systems*, vol. 14, pp. 861-872, September 2003.

[34] R. Thomas, B. Mark, T. Johnson, and J. Croall. NetBouncer: client-legitimacy-based high-performance DDoS filtering. *Proc. DARPA Information Survivability Conference and Exposition*, vol. 1, pp. 14-25, April 22-24, 2003.

[35] A. Hussain, J. Heidermann, and C. Papadopoulos. A Framework for Classifying Denial of Service Attacks. *Proc. ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication 2003*, ISBN: 1-58113-735-4, pp. 99-110, Karlsruhe, Germany, August 25-29, 2003.

[36] J. Mirkovic, P. Reiher, and M. Robinson. Forming Alliance for DDoS Defense. *Proc. 2003 workshop on New security paradigms*, 11-18, ISBN 1-58113-880-6, Ascona, Switzerland, August 2003.

[37] H. Liu, M. Parashar. Component-based Programming Model for Autonomic Applications. *Proc. First International Conference on Autonomic Computing*, New York, NY, USA, 2004.

[38] M. Mikic-Rakic, N. Medvidovic. Support for Disconnected Operation via Architectural Self-Reconfiguration. *Proc. First International Conference on Autonomic Computing*, (IEEE Computer Society), ISBN 0-7695-2114-2, New York, 2004.

[39] E. Gelenbe, M. Gellman, R. Lent, P. Liu, Pu Su. Autonomous smart routing for network QoS. *Proc. First International Conference on Autonomic Computing*, (IEEE Computer Society), ISBN 0-7695-2114-2, 232-239, New York, 2004.

[40] E. Gelenbe, R. Lent, A. Nunez. Self-aware networks and QoS. *Proceedings of the IEEE*, 92 (9), pp. 1478-1489, 2004.

[41] E. Gelenbe and R. Lent. Adhoc power aware Cognitive Packet Networks. *Ad Hoc Networks Journal*, Vol. 2 (3), pp. 205–216, 2004 (ISN: 1570-8705).

[42] E. Gelenbe. Cognitive Packet Network. *U.S. Patent No. 6,804,201 B1*, Oct. 12, 2004.

[43] J. Frey, G. Hughes, H. Mills, M. Schraefel, G. Smith, D. De Roure. Less is More: Lightweight Ontologies and User Interfaces for Smart Labs. *UK e-Science All Hands Meeting*, Nottingham, 2004.

[44] D.K.Y. Yau, J.C.S Lui, F. Liang, and Y. Yam. Defending Against Distributed Denial-of-Service Attacks With Max-Min Fair Server-Centric Router Throttles. *IEEE/ACM Transactions on Networking*, 13 (1): 29-42, February 2005.

[45] L. Tummolini, C. Castelfranchi, A. Ricci, M. Viroli, A. Omicini. Exhibitionists and Voyeurs do it better: A Shared Environment Approach for Flexible Coordination with Tacit Messages. *Environments for MultiAgent Systems*. LNAI 3374, Springer-Verlag, January 2005.

[46] F. Zambonelli, M.P. Gleizes, M. Mamei, R. Tolksdorf. Spray Computers: Explorations in Self Organization. *Journal of Pervasive and Mobile Computing*, 1 (1), May 2005.

[47] S. Kandula, D. Katabi, M. Jacob, and A. Berger. Botz-4-Sale: Surviving Organized DDoS Attacks that Mimic Flash Crowds. *Proc. 2nd USENIX Symposium on Networked Systems Design and Implementation (NSDI '05)*, Boston, MA, USA, May 2-4, 2005.

[48] J. Mirkovic and P. Reiher. D-WARD: A Source-End Defense against Flooding Denial-of-Service Attacks. *IEEE Transactions on Dependable and Secure Computing*, vol. 2, no. 3, pp. 216-232, July-September, 2005.

[49] P. Bouquet, L. Serafini, S. Zanobini. Peer-to-Peer Semantic Coordination. *Journal of Web Semantics*, 2 (1), 2005.

[50] E. Gelenbe, M. Gellman, and G. Loukas. An autonomic approach to denial of service defence. *Proc. of the IEEE Int. Symp. on a World of Wireless, Mobile and Multimedia Networks*, 537-541, June 2005.